Control and Cybernetics

vol. **37** (2008) No. 3

Fuzzy modelling of temperature difference in 200 MW power unit condenser using genetic fuzzy systems*

by

Michał Pająk

Radom University of Technology
Faculty of Mechanical Engineering
Department of Thermal Technology
Krasickiego 54, 26-600 Radom, Poland

Abstract: Energy generation is one of the most complicated industrial processes. Because of its complexity, there is no accurate conventional model of a power unit. Fuzzy logic concepts might be effectively implemented in this field. In the paper a universal method of creating a fuzzy logic model is presented. To check the usefulness of the method in the case of real industrial issues, a fuzzy model of temperature difference in a condenser was automatically generated. The modelling experiment and the assessment of model quality are presented in the paper.

Keywords: learning, fuzzy models, genetic algorithms, fuzzy statistics and data analysis, large-scale systems, linguistic modelling.

1. Introduction

Energy generation is a very complex production technology. To carry it in the proper way, people who are responsible for power plant supervision need to have access to a great number of data from each part of the plant (Pająk, 2001). For a power unit containing the 13K215 turbine it is necessary to analyse about 1000 pieces of information in real time to drive the unit with high efficiency (Laudyn, Pawlik, Strzelczyk, 1995). Only computerized systems of power plant monitoring can make it possible. Before power industry had been computerised, the power units were operated in an approximate way. Of course, efficiency was very poor. As an example, failure rate could be considered. Failure rate is calculated according to the formula

$$FAR = \frac{\sum_{i=1}^{n} T_{ai}}{\sum_{i=1}^{n} (T_{pi} + T_{ai})} \cdot 100\% \qquad (1)$$

*Submitted: March 2007; Accepted: April 2008

where: FAR – failure rate [%],
T_{ai} – length of outage because of failure for power unit no i [h/a],
T_{pi} – operation time of power unit no i [h/a],
n – number of analysed power units.

Before the implementation of computer monitoring the rate was about 20.1%, while now it is at the level of 2.4% (*Long-term* ..., 2003).

Unfortunately, a lot of data are of very low quality. In a situation like this it could be very useful to have a model of a power unit, which can qualify the collected data and generate the missing information.

The energy generation process is carried out in the power plant exploitation system. The system is very complex in view of high number of input data and the cross connections between them. There are three different levels of information. Each level is characterised by different range, frequency of changes and importance period (Kalotka, Pająk, 2006). Additionally, the processes carried out in the power unit exploitation system are highly differentiated (thermodynamical, electrical, mechanical). Therefore, it is very hard to develop a conventional model able to simulate the operation of the whole power unit. In the literature one can find studies meant to create a mathematical model of a power unit (Janiczek, 1992; Szlęk, Wilk, 1998; Badyda, Niwiński, 2007; Głodek, 2004). All these models can estimate power unit operation only in a limited range of processes and require wide range of measured parameters. In case of a real industrial plant only a limited number of operation parameters are accessible on-line. Therefore, a problem with model implementation arises (Pająk, 2007). So, the problem described offers a typical case for implementation of the fuzzy logic theory (Pająk. 2002).

The set of the model input data could be divided into two groups. The first group consists of the data described in the form of continuously measured values. The second one consists of the data assessed digitally (Muślewski, Woropay, 2005). To combine both types of data in one coherent assessment system, fuzzy modelling should be implemented. Especially, the Mamdani model makes it possible very easily (Ruano, 2005).

To obtain the accurate fuzzy model it is necessary to optimise its database and rule base. So, it is necessary to implement one of the optimisation techniques. The optimisation problems in the domain of fuzzy model structure optimisation are described by discrete (often integer) values. Such problems are very difficult to optimise from the computational point of view (Smutnicki, 2002). The reason is the lack of such analytical features as differentiability or linearity. Often, multimodality and high dimensionality of solution space are observed. All of that, and the limitation of the response time causes the uselessness of implementation of such algorithms as B&B (Branch-and-Bound), dynamic programming or linear and non-linear programming in these cases. Therefore, a good solution in this kind of optimisation problems is to implement the approximate methods. These methods enable to find the approximate solution and the quality of solution increases with calculation time.

The approximate methods can be divided into two groups. The first group is constituted by the construction and correction ones. The construction methods are fast and can be easily implemented, but generated solutions are not very accurate. The correction methods are slower and they need a starting point, which is corrected during the algorithm operation, but they yield a solution of very good quality. Owing to the implementation of the correction method it is possible to create the compromise between response time and solution quality in a flexible way. This kind of optimisation method is currently being strongly developed, and there is a lot of promising algorithms making it possible to get high quality solutions. Unfortunately, these new methods are not stable yet so that they should be examined before implementing them in practice.

In case of a real industrial problem the implemented optimisation method should be stable and well examined. All the enumerated above requirements are fulfilled by the genetic algorithms (Goldberg, 2003).

Due to the reasons mentioned above, the artificial intelligence technology in the form of fuzzy logic and genetic algorithms was used in this field.

The main objective of presented study was implementation of a universal method of fuzzy model generation in the case of a real industrial problem. In the paper, generation of a fuzzy model of temperature difference in a 13K215 steam turbine condenser is presented. The model is the first step to creation of a fuzzy model of a power unit.

2. Preliminaries: genetic fuzzy systems

Genetic fuzzy systems are a class of fuzzy systems. The structure and form of genetic fuzzy systems are generated by using the genetic algorithms.

The main part of a fuzzy system is a rule base. The rule base is a set of rules having the following form:

> IF a set of conditions are satisfied THEN a set of consequents can be inferred.

The rule base covers information about the modelled object or process.

At the beginning of fuzzy system operation, crisp input data are transformed to their fuzzy form. The process is called fuzzification and is based on database of a fuzzy system. A database contains the linguistic term sets considered in the linguistic rules and the membership functions defining the semantics of the linguistic labels (Herrera, 2008). Next, during the inference process, the fuzzy output is generated. The inference process depends on input data fuzzy values and rule base of the system. The latest step of fuzzy system operation is the defuzzification process. On the basis of fuzzy system database, according to the used defuzzification operator, the fuzzy output is transformed to its crisp form.

Generation of the rule base and definition of the database are the main goals of the fuzzy system learning process. This could be accomplished automatically using the training data set (input and output values). For this purpose, the learning process could be interpreted as an optimisation or search problem. From this point of view, the learning process attempts to find the optimal structure of the rule base and database of fuzzy system in the space of all possible rule base and database structures.

In the case of genetic fuzzy systems, defined above, the search problem is being solved using genetic algorithms.

Genetic algorithms are general purpose search algorithms, which use the principles inspired by natural genetics to evolve solutions to problems. The basic idea is to maintain a population of chromosomes (representing candidate solutions to the concrete problem being solved) that evolves over time through a process of competition and controlled variation. During successive iterations, called generations, chromosomes in the population are rated for their adaptation as solutions, and on the basis of these evaluations, a new population of chromosomes is formed using a selection mechanism and specific genetic operators such as crossover and mutation (Cordon et al., 2004).

Owing to genetic fuzzy systems implementation it is possible to create automatically the fuzzy models of considered problems on the basis of the input-output data set.

3. Characteristics of analysed process

A power unit condenser is the place where isobaric condensation occurs (Głodek, 2004). The pressure of an isobaric process depends on cooling conditions in the condenser.

For a condenser we can write down the following temperature equation:

$$T_{ps} = T_{w1} + \Delta T_w + \delta T \tag{2}$$

where:
- T_{ps} – temperature of condenser incoming steam [K]
- T_{w1} – temperature of condenser incoming cooling water [K]
- ΔT_w – temperature rise of cooling water [K]
- δT – difference between saturation temperature in condenser and outgoing cooling water [K].

The mean value of the cooling water temperature rise is about 8 to 12 K (Miller, 1998) and is described by the formula:

$$\Delta T_w = T_{w2} - T_{w1} \tag{3}$$

where:
- T_{w1} – temperature of condenser incoming cooling water [K]
- T_{w2} – temperature of condenser outgoing cooling water [K].

So, it is possible to establish the following formula:

$$T_{ps} = T_{w2} + \delta T. \tag{4}$$

The mean value of the temperature difference in a condenser is about 2 to 4 K (Zębaty, 1990). From (4), the temperature difference in a condenser is

$$\delta T = T_{ps} - T_{w2}. \tag{5}$$

The above enables the calculation of the temperature difference in a condenser, based on measured data. So, it is possible to learn and check the quality of the fuzzy model created in the next stages of the project.

4. Analysis of data collected from a real industrial object

Real object data were collected within the biggest hard coal fired power plant in Poland. In the analysis, the data covering fifteen months of power plant operation were considered (*Computerised System* ..., 1999). In this data set there were over 600,000 data vectors, each consisting of 34 parameters describing the operation state of the power unit.

The analysed data were of different quality. To exclude poor quality measurement data, a filter system was designed. The system reflects the correct value ranges of the exploitation parameters for a power unit 200MW (OP-650 boiler and 13K215 turbine). The size of the filter window was expressed as the function of the active load of the power unit. Below, the filtering condition for the amount of main steam is presented as an example:

$$\dot{M}_{ms}[t/h] = \begin{cases} 0 \div 300 & \text{for } P < 110MW \\ f(x) \pm 20 \text{ where } x = \dfrac{P}{200} & \text{for } P \geqslant 110MW \end{cases} \tag{6}$$

$$f(x) = \frac{0.943238 + 567098.904568x - 944124.470704x^2}{1000} + \frac{1026960.170082x^3 - 358343.972234x^4}{1000}$$

where: $\dot{M}_{ms}$ – amount of main steam [t/h]
P – active load [MW].

By applying the filtering system, the poor quality parameters and the vectors with the data, whose values were not included in the filtering windows, were removed. In the end, 61,168 data records were left. Unfortunately, the temperature difference in the condenser was not a measured value. So, it was calculated using the formula (5), where the temperature of condenser incoming steam was calculated as the saturation temperature for the current pressure of condenser incoming steam. The obtained data vectors were divided into two equal sets. The first set consisted of the data for the earlier time and was used for fuzzy model learning. The second one consisted of the data recorded later and was used to test the generated model.

5. Fuzzy analysis of measured parameters

The fuzzy modelling process was based on the analysis of the measurement data recorded on a real industrial object. The main problem was to choose only significant inputs for the fuzzy model. This was performed according to the fuzzy curves theory (Lin and Cunningham, 1995). According to this theory, for the chosen fixed values of each parameter a part of the surface is calculated:

$$\delta T = f(z_1, ..., z_n) \wedge n = 34 \tag{7}$$

where: z_n – n-index parameter.

At the beginning, it is necessary to fuzzify the part of the fixed value. This should be done to avoid problems resulting from irregularly spaced measurement points in the solution area. As the membership function the Gaussian curve is used:

$$\mu(z_m^*) = \exp\left(-\left(\frac{z_m^* - z_m}{b_m}\right)^2\right). \tag{8}$$

For each section, the mean value is calculated according to the following formula:

$$\delta T_{mv}(z_m) = \frac{\sum\limits_{k=1}^{t} \mu_m(z_{mk}) \cdot \delta T_k}{\sum\limits_{k=1}^{t} \mu_m(z_{mk})} \tag{9}$$

where: δT_{mv} – mean value
m – index of input
k – index of parameter.

Mean values of each part define the curve. The range of the curve is the factor of the dependence between the input and output parameters. During the study the power unit measurement data were analysed, by changing the parameters of calculations. To shorten the time of calculations the measurement data were reduced with the reduction radius equal 1%. The reduction process separated the groups of measurement vectors placed closer than the assumed distance in the solution space according to the formula:

$$\underset{m_i m_j \in M}{\forall} |m_i - m_j| < \tau \tag{10}$$

where: m – set of measurements
τ – reduction radius.

The groups of measurement vectors were substituted by one vector. This process decreased the number of measurement vectors to 2,250. To simplify the process of analysis, the results of the calculations were presented in the form of gradient and spectrum diagrams. The spectrum diagram presents the range of fuzzy curves for all the parameters (Fig. 2). The gradient diagram could help in making a decision about the significance of the fuzzy model inputs (Fig. 1). Fig. 3 presents the diagram of fuzzy curves for one of the most significant inputs. Thanks to the experiments carried out the most significant inputs were separated according to the following list:

1. T_{pwn} – temperature of output steam from the low pressure turbine [K]
2. p_{sk} – pressure in condenser [MPa]
3. T_{wchd} – temperature of input water to the condenser [K]
4. T_{wchw} – temperature of output water from the condenser [K]
5. T_{kond} – temperature of condensate [K].

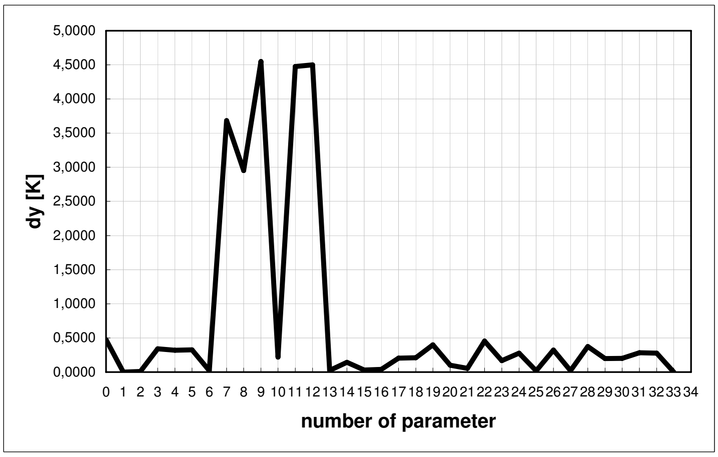

Figure 1. Gradient diagram of input parameters

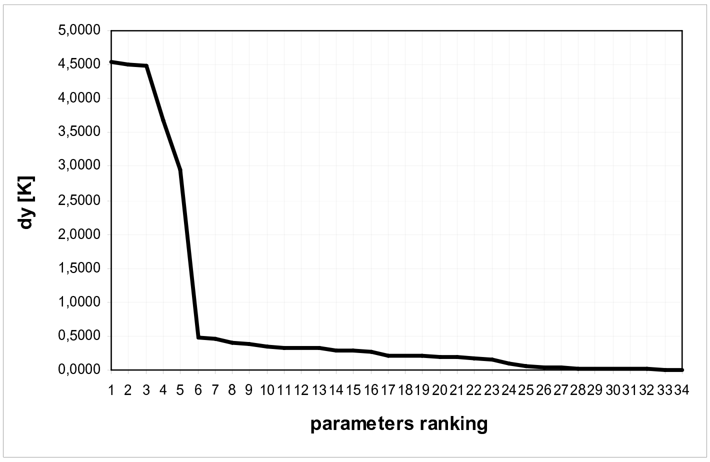

Figure 2. Spectrum diagram of input parameters

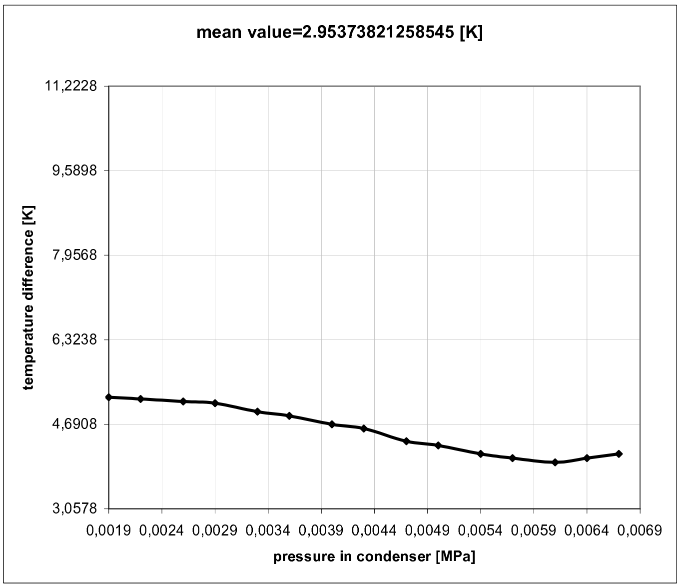

Figure 3. Fuzzy curve diagram for pressure in condenser

6. Fuzzy model generation algorithm

Fuzzy model generation was carried out on the basis of the measurement data, using a genetic algorithm (Herrera, Lozano, Verdegay, 1998). The model generation algorithm consists of three separate steps: rule base generation, rule base simplification and rule base tuning. The rule base generation is an iterative process, which makes a definite number of genetic algorithm steps in each iteration. As a result of functioning of the genetic algorithm a fuzzy rule is obtained. For each sample from the learning data set the compatibility degree is calculated on the basis of the generated rule. Then, for each sample from the learning data set, the covering degree is calculated as a sum of compatibility degrees for all fuzzy rules. The data samples having a greater or equal covering degree than the fixed value are removed from the data set. When the generation process creates a fuzzy rule, for which the compatibility degree for each sample is zero, the algorithm removes such a rule and increases the mutation probability of the genetic algorithm according to the following formula:

$$P'_m = P_m + \frac{(1 - P_m)}{n} \tag{11}$$

where: P_m – mutation probability
P'_m – modified mutation probability
n – fixed number of useless steps of generation process.

When the generating process creates the fuzzy rule, for which the compatibility degree of any sample is not zero, then the mutation probability is decreased to the start value and the process is repeated. However, if in n steps there are no results, the algorithm enables us to exchange the genetic material on the basis of the remaining samples and restarts the generation process, keeping the already generated rules, or generates the rules based on the remaining samples using the same generation method as in the initial population. The generation process is continued until all the data samples are removed.

The population in the genetic algorithm consists of chromosomes representing individual rules. The chromosomes are evaluated from the point of view of the fitness function. To increase the quality of the algorithm, the evolution strategy is applied as a tool for local tuning of generated rules.

To describe the input data, several linguistic values are established for each parameter. Each linguistic value is a fuzzy set defined over the range of the measured value. The first fuzzy set is of L type and the last one is of Γ type. All the remaining sets are of Λ type (Łachwa, 2001). All the input and output variables are partitioned in the same way. Thus, it is possible to fuzzify each measurement vector. For the real value of a variable the value of membership function for each fuzzy set is calculated. The fuzzy value is constituted by the set of obtained membership values. In the rule base generation process the real coded genetic algorithms are used.

The knowledge base consists of fuzzy rules in the following form:

$$\text{If } z_1 \text{ is } L_{i1}(z_1)\dots \text{ and } z_n \text{ is } L_{in}(z_n) \text{ then } y \text{ is } L_{in+1}(y) \tag{12}$$

where: z_i – model input
L_i – linguistic value of parameter
y – model output.

A chromosome consists of two parts. The first part represents linguistic values of the input and output of the model and the second one represents characteristic values for each fuzzy set placed in the first part:

$$C = C_1C_2 \tag{13}$$

$$C_1 = (L_{i1}, \dots, L_{in}, L_{in+1}) \tag{14}$$

$$C_2 = (a_{Li1}, b_{Li1}, c_{Li1}, \dots, a_{Lin+1}, b_{Lin+1}, c_{Lin+1}). \tag{15}$$

The initial generation consists of three equal parts. The first part consists of the chromosomes generated on the basis of the measured data assuming the values of the first chromosome part as numbers of fuzzy sets whose membership value is the highest. The second part of the initial population consists of chromosomes having the same first chromosome part and the randomly generated second one. The third part is entirely randomly generated.

The fitness function is described by the following formula:

$$Q(R_i) = CZ_M(R_i) \cdot SPZ(R_i) \cdot WWNZ(R_i) \cdot WIW(R_i) \tag{16}$$

where: M – measurement data set
R_i – analysed rule.

The components of fitness function are described below:

$CZ_M(R_i)$ – frequency of fuzzy rule

$$CZ_M(R_i) = \frac{\sum\limits_{l=1}^{p} WZ_i(m_l)}{p} \tag{17}$$

where: p – number of measurement samples
m_l – value of measurement vector $l \in (1, ..., p)$;

$WZ_i(m_l)$ – compatibility degree between R_i and sample m_l

$$WZ_i(m_l) = \circ(\mu_{Li1}(z_1(m_l)), \ldots, \mu_{Lin}(z_n(m_l)), \mu_{Ln+1}(z_{n+1}(m_l)) \tag{18}$$

where: $\circ\mu_{Li}$ – selected t-norm;

$SPZ(R_i)$ – average covering degree over positive examples

$$SPZ(R_i) = \sum_{m_l \in M_+(R_i)} \frac{WZ_i(m_l)}{p_+(R_i)} \tag{19}$$

where: $p_+(R_i)$ – number of positive examples;

$M_+(R_i)$ – set of positive examples

$$M_+(R_i) = \{m_l \in M : WZ_i(m_l) \geqslant \omega\} \tag{20}$$

where: ω – compatibility degree;

$WWNZ(R_i)$ – penalty on negative examples

$$WWNZ(R_i) = \begin{cases} 1 & \text{for } p_-(R_i) \leqslant k \cdot p_+(R_i) \\ \dfrac{1}{p_-(R_i) - k \cdot p_+(R_i) + \exp(1)} & \text{for } p_-(R_i) > k \cdot p_+(R_i) \end{cases} \tag{21}$$

where: k – factor of ratio of negative to positive examples $k \in [0, 1]$
$p_-(R_i)$ – number of negative examples,

$$\begin{aligned} &p_i(R_i) = |M_-(R_i)| \\ &M_-(R_i) = \{m_l \in M : WZ_i(m_l) = 0 \wedge IN_i(m_l) > 0\} \\ &IN_i(m_l) = \circ(\mu_{Li1}(z_1(m_l)), \ldots, \mu_{Lin}(z_n(m_l))) \end{aligned} \tag{22}$$

$WIW(R_i)$ – niche iteration rate

$$WIW(R_i) = 1 - \max\{WZ_i(W_h)\} \tag{23}$$

where: W_h – an already generated rule $h = 1, \ldots, H$
H – number of already generated rules.

The genetic algorithms maximize the value of the fitness function.

One step of genetic algorithm consists of three operations: crossing, mutation and evolution strategy. The crossing process uses different crossing operators depending on crossed chromosomes. If we want to cross chromosomes with equal C_1 parts then to cross C_2 parts we use the arithmetical min-max crossing operator. The offspring consist of C_1 part of the parent chromosome and the modified C_2 part. Four chromosomes are generated:

$$H_k = (h_1^k, ..., h_i^k, ..., h_n^k) \quad \text{where } k = 1,2,3,4 \tag{24}$$

$$h_i^1 = \lambda c_i^1 + (1-\lambda) c_i^2 \tag{25}$$

$$h_i^2 = \lambda c_i^2 + (1-\lambda) c_i^1 \lambda \tag{26}$$

$$h_i^3 = \min\{c_i^2, c_i^1\} \tag{27}$$

$$h_i^4 = \max\{c_i^2, c_i^1\} \tag{28}$$

and the two best ones replace their parents. If C_1 parts are different, then simple crossing takes place on C_1 parts and appropriate genes of C_2 are copied. Two offspring are generated and they replace their parents. C_1 parts of the offspring are generated according to the following formula:

$$H_1 = (c_1^1, c_2^1, ..., c_i^1, c_{i+1}^2, ..., c_n^2) \tag{29}$$

$$H_2 = (c_1^2, c_2^2, ..., c_i^2, c_{i+1}^1, ..., c_n^1). \tag{30}$$

C_2 parts are appropriate genes of the original chromosomes.

Mutation also uses two operators. If the chosen mutation point is a gene from C_2 part, then the Michalewicz non-uniform mutation operator is used. If the mutation point is in C_1 part then its value is randomly increased or decreased by one. If this is not possible because of the interval of performance of the gene, the opposite operation is carried out. The changing value of the C_1 part is connected with the adequate genes from modification of C_2.

The last operation of the genetic algorithm is evolution strategy. It is applied to the best chromosomes in the population. The operation modifies the values of the genes from the second part of the chromosome according to the following formula:

$$c_i' = c_i + g_i \tag{31}$$

where: g_i – randomly generated value based on uniform distribution from interval $[0, \sigma_i]$

$$\sigma_i = \sigma \cdot \frac{c_r - c_l}{4} \tag{32}$$

where: c_r, c_l – extreme values of the gene;

σ value is mutated according to the formula:

$$\sigma' = \begin{cases} \frac{\sigma}{\sqrt[n]{c}} & \text{for } p > \frac{1}{5} \\ \sigma \cdot \sqrt[n]{c} & \text{for } p < \frac{1}{5} \\ \sigma & \text{for } p = \frac{1}{5} \end{cases} \tag{33}$$

where: p – relative frequency of positive mutations,
n – number of mutations,
c – evolution strategy coefficient.

If the modified chromosome is better from the fitness function point of view, it replaces the parent one. Otherwise it is removed. If the process does not generate better chromosomes during more than a given number of steps, it is stopped.

The result of the generation process is a set of fuzzy rules. Because of its iterative characteristic, two similar or conflicting rules might exist in the generated set. Therefore, the next step of the fuzzy model generation is the rule base simplification.

For the purpose of simplification, rule bases are expressed in the form of binary coded chromosomes. Each chromosome represents a complete rule base. If a given rule exists in a described rule base, a corresponding gene of the chromosome has value 1, otherwise its value is 0. The initial population consists of randomly generated chromosomes except for the chromosome representing the rule base obtained as a result of the previous part of calculations. The binary coded genetic algorithm runs a fixed number of generations. One step of the genetic algorithm consists of crossing and mutation using an elitist selection and a uniform random sampling mechanism. The mutation operator changes the value of the gene to the opposite number. The two point crossing operator is used. It divides chromosomes into three parts and changes one of them. The algorithm tries to minimize the value of the fitness function, which could be expressed in the following form:

$$F(C_j) = \begin{cases} E(C_j) & \text{for } SZ(C_j) \geqslant \tau \\ \frac{1}{2} \cdot \sum_{m_l \in M} y(m_l)^2 & \text{for } SZ(C_j) < \tau \end{cases} \tag{34}$$

where: τ – fixed threshold value of compatibility degree between rule base and the training data set;

$SZ(C_j)$ – compatibility degree between the rule base and the training data set:

$$SZ(C_j) = \min\left\{ \sum_{i=1}^{h} WZ_i(m_1), \ldots, \sum_{i=1}^{h} WZ_i(m_p) \right\} \tag{35}$$

where: h – number of generated rules,
p – number of data samples;

$E(C_j)$ – the mean square error is calculated as follows:

$$E(C_j) = \frac{1}{2 \cdot p} \sum_{m_l \in M} (y(m_l) - Y(m_l))^2 \tag{36}$$

where: $Y(m_l)$ – fuzzy model output value for measurement vector m_l.

The last step of the process is the rule base tuning. It is done using a genetic algorithm again, but in this step the real coded version is used. Each chromosome describes a complete rule base. Characteristic parameters of linguistic values for each real variable are coded in the form of chromosome genes. So, an individual chromosome can be presented in the forms mentioned earlier, i.e. (13, 14, 15). The initial population consists of randomly generated chromosomes, except for one chromosome, which represents the rule base obtained as a result of the previous part of calculations. The arithmetical min-max crossing operator and the Michalewicz non-uniform mutation operator are implemented. The algorithm tries to minimize the value of the fitness function (36).

7. Generation of fuzzy model of temperature difference in a condenser

The algorithm presented above was used to create a fuzzy model of temperature difference in a condenser. The parameters of model generation process are given in Table 1. As it was already mentioned (in Section 4) the training data set consisted of 2,250 samples. The number of training samples was the result of the reduction process carried out to decrease the time of calculations. To check the influence of the reduction process on quality of generated model, the first step of model generation was carried out on the basis of the full training set (15,420 samples) and the reduced one (2,250 samples). The quality measures of generated models are presented in Tables 2 and 3. The differences in the error values are not significant but the correlation value is much better in the case of the reduced training data set. Additionally, the reduction process decreased the calculation time (about ten times). So, the next steps of model generation were carried out on the basis of the reduced training set.

In the case of the reduced data set the generation algorithm removed 2,246 samples. Next the exchange of the genetic material was carried out. The algorithm removed the next three samples. On the basis of the last sample a fuzzy rule was generated. Then, the obtained rule base was simplified and tuned according to the algorithm.

To make the generation process possible, each data vector was fuzzified using linguistic values, defined for each variable. The range of each variable was partitioned into seven linguistic values. As it was already mentioned (Section 5) the first of the linguistic values was described by the L type fuzzy set, the last one by the Γ type fuzzy set, and the other ones by Λ type fuzzy sets. In the same way the fuzzy model output value was defuzzified. During the

Table 1. Parameters of model generation process

No	Parameter name	Value
1.	Compatibility degree	1.0
2.	Threshold of sample	0.05
3.	Negative to positive samples factor k	0.1
4.	Threshold of compatibility degree between rule base and samples set	0.25
5.	T-norm used in rule base generation process	MINIMUM
6.	Number of genetic algorithm generations runs in one iteration step	50
7.	Number of useless evolution strategy modifications runs to stop	25
8.	Number of chromosomes modified by evolution strategy	20% of population
9.	Mutation parameter c of evolution strategy	0.9
10.	Number of generations in simplification process	500
11.	Number of generations in tuning process	1000
12.	Number of simplification process initial population chromosomes	61
13.	Number of tuning process initial population chromosomes	61
14.	Parameter b of non-uniform Michalewicz mutation	5
15.	Crossing probability of generation process	0.6
16.	Crossing probability of simplification process	0.6
17.	Crossing probability of tuning process	0.6
18.	Mutation probability of generation process	0.1
19.	Mutation probability of simplification process	0.1
20.	Mutation probability of tuning process	0.1
21.	Min-max arithmetical operator coefficient	0.35
22.	Aggregation operator of fuzzy model	MINIMUM
23.	Implication operator of fuzzy model	MINIMUM
24.	Accumulation operator of fuzzy model	MAXIMUM
25.	Defuzzyfication operator of fuzzy model	centre of weight

Table 2. Quality measures of temperature difference fuzzy model after first step of the generation process (full data set)

Maximum error [%]	73.4014
Minimum error [%]	0.0065
Correlation [%]	43.86
Mean square error	0.4247
Mean square error [%]	0.1330
Mean absolute error [%]	15.1918
Number of maximum error examples	1
Number of rules	37

Table 3. Quality measures of temperature difference fuzzy model after first step of the generation process (reduced data set)

Maximum error [%]	88.4636
Minimum error [%]	0.0196
Correlation [%]	66.0760
Mean square error	0.6515
Mean square error [%]	0.1647
Mean absolute error [%]	19.5801
Number of maximum error examples	1
Number of rules	32

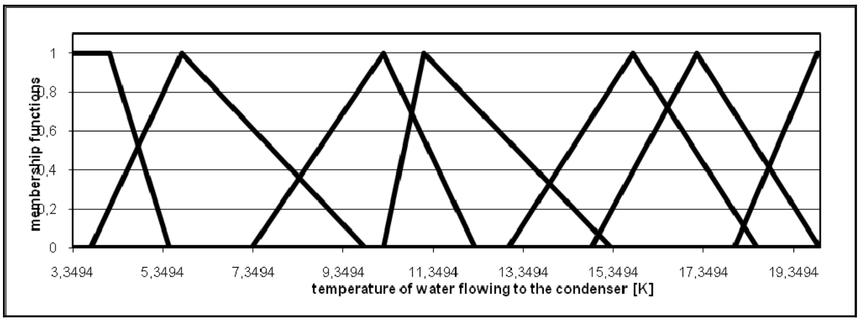

Figure 4. Linguistic values of temperature of water inflow to the condenser

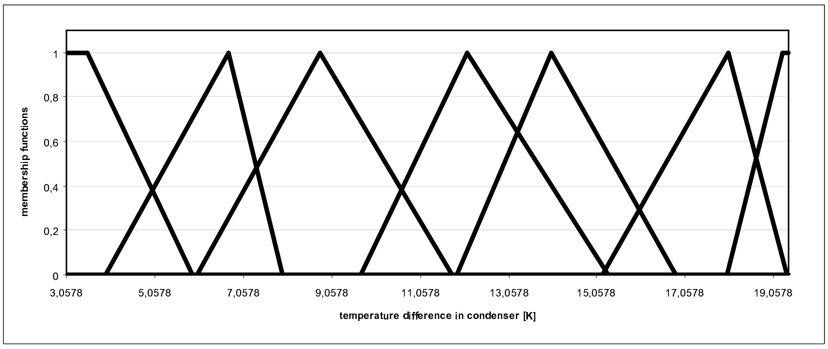

Figure 5. Linguistic values of temperature difference in the condenser

tuning step of the generation process the characteristic points of the linguistic values were changed to increase the quality of the model. For illustration, the linguistic values for temperature of water inflowing to the condenser (Fig. 4) and temperature difference in condenser (Fig. 5) are presented.

As a result of the complete generation process the fuzzy model was obtained. The rule base of the model consists of 21 rules. All the rules have the same form. They are built of five premises and one consequent. All five premises are connected with the AND operator (12). Some exemplary rules are presented below

$$
\begin{aligned}
T_{pwn} &= L^1_{Tpwn} \wedge p_{sk} = L^1_{psk} \wedge T_{wchd} = L^2_{Twchd} \wedge T_{wchw} \\
&= L^2_{Twchw} \wedge T_{kond} = L^2_{Tkond} \Rightarrow \delta T = L^1_{\delta T} \\
T_{pwn} &= L^3_{Tpwn} \wedge p_{sk} = L^3_{psk} \wedge T_{wchd} = L^6_{Twchd} \wedge T_{wchw} \\
&= L^5_{Twchw} \wedge T_{kond} = L^5_{Tkond} \Rightarrow \delta T = L^0_{\delta T} \\
T_{pwn} &= L^5_{Tpwn} \wedge p_{sk} = L^5_{psk} \wedge T_{wchd} = L^3_{Twchd} \wedge T_{wchw} \\
&= L^3_{Twchw} \wedge T_{kond} = L^4_{Tkond} \Rightarrow \delta T = L^5_{\delta T}
\end{aligned}
\tag{37}
$$

where: L^i_{Tpwn} – linguistic value no. i of temperature of steam flowing out of low pressure turbine [K]
L^i_{psk} – linguistic value no. i of pressure in condenser [MPa]
L^i_{Twchd} – linguistic value no. i of temperature of input water to the condenser [K]
L^i_{Twchw} – linguistic value no. i of temperature of output water from the condenser [K]
L^i_{Tkond} – linguistic value no. i of temperature of condensate [K]
$L^i_{\delta T}$ – linguistic value no. i of temperature difference in condenser [K].

According to the Mamdani model characteristic (Piegat, 1999) the rules are not evenly distributed in the solution space. The regions, where the solution space changes its shape fast are covered by a higher number of fuzzy rules. Thanks to this the model of the form considered can decrease the number of rules and keep high accuracy of the response.

The fuzzy model developed was tested using a testing data set. The testing data set consisted of 15,420 samples. The testing data set was as big as possible to guarantee certainty of the testing results. The results of model testing are presented in Table 4.

Table 4. Quality measures of temperature difference fuzzy model

Maximum error [%]	44.1898
Minimum error [%]	0.0030
Correlation [%]	83.3495
Mean square error	0.1366
Mean square error [%]	0.0754
Mean absolute error [%]	11.2652
Number of maximum error examples	1
Number of rules	21

To check the quality of the generated model the following measures were applied:

– the relative maximum error:

$$\delta_{\max} = \max\left\{ \underset{m_l \in M}{\forall} \frac{|y(m_l) - Y(m_l)|}{y(m_l)} \right\} \tag{38}$$

where: $y(m_l)$ – output of object for m_l sample [K]
$Y(m_l)$ – output of model for m_l sample [K];

– the number of examples generated with the maximum error;
– the relative minimum error:

$$\delta_{\min} = \min\left\{ \underset{m_l \in M}{\forall} \frac{|y(m_l) - Y(m_l)|}{y(m_l)} \right\}; \tag{39}$$

– the number of examples generated with no error
– the relative absolute error

$$\delta_{ma} = \frac{\sum_{l=1}^{p} \frac{|y(m_l) - Y(m_l)|}{y(m_l)}}{p}. \tag{40}$$

However, the most important measures of model quality are the mean square error (36), the value of the correlation function (41) and the relative mean square error (42):

$$r_{yY} = \frac{\sum_{l=1}^{p} (y(m_l) - \bar{y}) \cdot (Y(m_l) - \bar{Y})}{\sqrt{\sum_{l=1}^{p} (y(m_l) - \bar{y})^2 \cdot \sum_{l=1}^{p} (Y(m_l) - \bar{Y})^2}}, \tag{41}$$

$$\delta_{rw} = \frac{1}{(y_{\max} - y_{\min})} \sqrt{\frac{\sum_{m_l \in M} (y(m_l) - Y(m_l))^2}{p \cdot (p+1)}}. \tag{42}$$

8. Conclusions

Upon analysing the results obtained we can say that the quality of the fuzzy model is pretty good, especially the value of the relative mean square error, which is very small. The experiment carried out proves the correctness of the used algorithms and methods. Looking at the results of modelling of the temperature difference in a condenser we can say that it is possible to put into practice the software created and to use it to model real industrial objects.

References

Badyda, K. and Niewiński, G. (2007) Model matematyczny turbozespołu parowego dla symulatora bloku energetycznego. Wybrane zagadnienia (Mathematical model of the steam turbine for the simulator of the power generation block. Selected issues; in Polish). *XLVI Symposium "Modelling in mechanics"*, Wisła, Poland.

Computerised System of Power Plant Activity Monitoring (1999) Advertising materials, P.W.P.T. Tekom, Radom.

Cordon, O., Gomide, F., Herrera, F., Hoffman, F. and Magdalena, L. (2004) Ten Years of Genetic Fuzzy Systems: Current Framework and New Trends. *Fuzzy Sets and Systems* **141** (1), 5-31.

Głodek, E. (2004) *Energetyczne i ekologiczne aspekty wprowadzenia skojarzenia w bloku energetycznym* (*Power generation and ecological aspects of introducing co-generation in a power block*; in Polish). PhD thesis, University of Technology in Opole.

Goldberg, D.E. (2003) *Algorytmy genetyczne i ich zastosowanie* (*Genetic Algorithms in Search, Optimization and Machine Learning*; Polish translation). WNT, Warszawa.

Herrera, F. (2008) Genetic Fuzzy Systems: Taxonomy, Current Research Trends and Prospects. *Evolutionary Intelligence* **1**, 27-46.

Herrera, F., Lozano, M. and Verdegay, J.L. (1998) A learning process for fuzzy control rules using genetic algorithms. *Fuzzy Sets and Systems* **100**, 143-158.

Janiczek, R.S. (1992) *Eksploatacja elektrowni parowych* (*Exploitation of steam power generating plants*; in Polish). WNT, Warszawa.

Kalotka, J. and Pająk, M. (2006) *Gospodarka remontowa elektrowni cieplnych* (*Repair economy of thermal power plants*; in Polish). ITE, Radom.

Laudyn, D., Pawlik, M. and Strzelczyk, F. (1995) *Elektrownie* (*Power plants*; in Polish). WNT, Warszawa.

Lin, Y. and Cunningham, G.A. (1995) A new approach to fuzzy-neural system modelling. *IEEE Transactions on Fuzzy Systems*, **3** (2), 190-198.

Long-term analyses of reliability indices of Polish power units. (2003) (In Polish.) Agencja Rynku Energii, Warszawa.

Łachwa, A. (2001) *Rozmyty świat zbiorów, liczb relacji, faktów, reguł i decyzji* (*Fuzzy world of sets, numbers, relations, facts, rules and decisions*; in Polish). EXIT, Warszawa.

Miller, A. (1998) *Maszyny i urządzenia cieplne i energetyczne* (*Thermal and energy machines and devices*; in Polish). WSIP, Warszawa.

Muślewski, Ł. and Woropay, M. (2005) Theoretical grounds to evaluate of the transport system operation. In: C.G. Soares, V. Garbatov and N. Fonseca, eds., *Proceedings of the 12th International Congress of the International Maritime Association of Mediterranean – IMAM2005, Maritime Transportation and Exploitation of Ocean and Costal Resources.* Taylor

& Francis Group, London.

PAJĄK, M. (2001) Computerised system of power plant monitoring and supervising ERO_TKE. *Proceedings of the V Conference Computerised Systems of Science, Industry and Transport TRANSCOMP,* Zakopane, Poland.

PAJĄK, M. (2002) Zastosowania logiki rozmytej w praktyce (Application of fuzzy logic in practice; in Polish). *Proceedings of the VI Conference Computerised Systems of Science, Industry and Transport TRANSCOMP,* Zakopane, Poland.

PAJĄK, M. (2007) Computerized support of WR-25 heat boiler control and supervision process. 11^{th} *International Scientific Seminary on Developments in Machinery Design and Control,* Cerveny Klastor, Slovakia.

PIEGAT, A. (1999) *Modelowanie i sterowanie rozmyte* (*Fuzzy modelling and control*; in Polish). EXIT, Warszawa.

RUANO, A. (2005) Intelligent Control Systems Using Computational Intelligence Techniques, IEE.

SMUTNICKI, C. (2005) *Algorytmy szeregowania* (*Sequencing algorithms*; in Polish). EXIT, Warszawa.

SZLEK, A. and WILK, R. (1998) Algorytm sterowania pracą kotła rusztowego (Algorithm for controlling the grill boiler; in Polish). *V Conference "The technical, economic and ownership problems of Polish thermal industry",* Korbielów, Poland.

ZĘBATY, W. (1990) *Systemy i urządzenia chłodzące elektrowni cieplnych* (*Cooling systems and devices of thermal power plants*; in Polish). WNT, Warszawa.