DOI: 10.15439/2015F296
ACSIS, Vol. 5

Learning Fuzzy Cognitive Maps using Structure Optimization Genetic Algorithm

Katarzyna Poczęta, Alexander Yastrebov
Kielce University of Technology,
al. Tysiąclecia Państwa Polskiego 7
25-314 Kielce, Poland
Email: {k.piotrowska, a.jastriebow}@tu.kielce.pl

Elpiniki I. Papageorgiou
Center for Research and Technology Hellas, CERTH
6th km Charilaou-Thermi Rd., Thermi 57001, Greece
and
Technological Educational Institute (T.E.I.)
of Central Greece
3rd Km Old National Road Lamia-Athens,
Lamia 35100, Greece
Email: epapageorgiou@iti.gr, epapageorgiou@teilam.gr

***Abstract*—Fuzzy cognitive map (FCM) is a soft computing methodology that allows to describe the analyzed problem as a set of nodes (concepts) and connections (links) between them. In this paper a new Structure Optimization Genetic Algorithm (SOGA) for FCMs learning is presented for modeling complex decision support systems. The proposed approach allows to automatic construct and optimize the FCM model on the basis of historical multivariate time series. The SOGA defines a new learning error function with an additional penalty for highly complexity of FCM understood as a large number of concepts and a large number of connections between them. The aim of this study is the analysis of usefulness of the Structure Optimization Genetic Algorithm for fuzzy cognitive maps learning. Comparative analysis of the SOGA with other well-known FCM learning algorithms (Real-Coded Genetic Algorithm and Multi-Step Gradient Method) was performed on the example of prediction of rented bikes count. Simulations were done with the ISEMK (Intelligent Expert System based on Cognitive Maps) software tool. The obtained results show that the use of SOGA allows to significantly reduce the structure of the FCM model by selecting the most important concepts and connections between them.**

***Index Terms*—Fuzzy Cognitive Maps, Structure Optimization Genetic Algorithm, Real-Coded Genetic Algorithm, Multi-Step Gradient Method**

I. Introduction

FUZZY cognitive map (FCM) [16] is a soft computing methodology combining the advantages of fuzzy logic and artificial neural networks. It allows to visualize and analyze problem as a set of nodes (concepts) and links (connections between them). One of the most important aspect connected with the use of fuzzy cognitive maps is their ability to learn on the basis of historical data [22]. Supervised [14], [15], [25] and population-based [1], [7], [8], [20], [28] methods allow to evaluate the weights of connections.

Fuzzy cognitive maps are an effective tool for modeling dynamic decision support systems [18]. They were applied to many different areas, such as prediction of pulmonary infection [19], scenario planning for the national wind energy sector [2] or integrated waste management [4]. Carvalho discussed possible use of FCM as tool for modeling and simulating complex social, economic and political systems [5]. The use of FCMs as pattern classifiers is presented in [23]. An innovative method for forecasting artificial emotions and designing an affective decision system on the basic of fuzzy cognitive map is proposed in [26]. The application of fuzzy cognitive maps to univariate time series modeling is discussed in [13], [12], [17]. Prediction of work of complex and imprecise systems on the basis of FCM is described in [27].

In practical applications to solve certain classes of problems (e.g. data analysis, prediction or diagnosis), finding the most significant concepts and connections plays an important role. It can be based on expert knowledge at all stages of analysis: designing the structure of the FCM model, determining the weights of the relationships and selecting input data. Supervised and population-based algorithms allow the automatic construction of fuzzy cognitive map on the basis of data selected by the experts or all available input data. However, modeling of complex systems can be difficult task due to the large amount of the information about analyzed problem. Fuzzy cognitive maps with the large number of concepts and connections between them can be difficult to interpret and impractical to use as the number of parameters to be established grows quadratically with the size of the FCM model [13]. In [12] nodes selection criteria for FCM designed to model univariate time series are proposed. Also some simplifications strategies by posteriori removing nodes and weights are presented [13].

In [21] the Structure Optimization Genetic Algorithm allowing selection of the crucial connections is proposed. In this paper, we present an extension of this algorithm that allows to significantly reduce the size of FCM by automatic selection not only the most important connections but also the most important concepts from all possible nodes. The proposed approach enables fully automatic construction of the FCM model by selection of crucial concepts and determining the relationships between them on the basis of available historical data. The SOGA is compared with well-known methods: the Multi-Step Gradient Method (MGM) [14] and the Real-Coded Genetic Algorithm (RCGA) [28] on the example of system for prediction of count of rented bikes. Learning and testing of FCMs are based on historical multivariate time series, taken from the UCI Machine Learning Repository [6]. Simulations are accomplished with the use of the developed ISEMK (Intelligent Expert System based on Cognitive Maps) software tool [24].

The aims of this paper are:

- to introduce a new FCM learning algorithm that allows to automatic optimize the structure of fuzzy cognitive map by finding the most important concepts and connections between them on the basic of historical data,
- to perform a comparative analysis of the proposed method of FCM learning with other well-known algorithms on the example of system for prediction of count of rented bikes.

The paper is organized as follows. In Section II fuzzy cognitive maps are described. Section III introduces the proposed method for fuzzy cognitive maps learning. Section IV describes the developed software tool ISEMK. Section V presents selected results of simulation analysis of the proposed approach. The last Section contains a summary of the paper.

II. Fuzzy Cognitive Maps

The structure of FCM is based on a directed graph:

$$< X, W > , \tag{1}$$

where $X = [X_1, ..., X_n]^T$ is the set of the concepts significant for the analyzed problem, W is weights matrix, $W_{j,i}$ is the weight of the connection between the j-th concept and the i-th concept, taking on the values from the range $[-1, 1]$. Value of -1 means full negative influence, 1 denotes full positive influence and 0 means no causal effect [16].

Concepts obtain values in the range between $[0, 1]$ so they can be used in time series prediction. The values of concepts can be calculated according to the formula:

$$X_i(t+1) = F\left(X_i(t) + \sum_{j \neq i} W_{j,i} \cdot X_j(t)\right) , \tag{2}$$

where t is discrete time, $t = 0, 1, 2, ..., T$, T is end time of simulation, $X_i(t)$ is the value of the i-th concept, $i = 1, 2, ..., n$, n is the number of concepts, $F(x)$ is a transformation function, which can be chosen in the form:

$$F(x) = \frac{1}{1+e^{-cx}} , \tag{3}$$

where $c > 0$ is a parameter.

Fuzzy cognitive map can be automatic constructed with the use of supervised and population-based learning algorithms. In the next section, selected methods of FCMs learning are described.

III. Fuzzy Cognitive Maps Learning

The aim of the FCM learning process is to estimate the weights matrix W. In the paper a new population-based approach for fuzzy cognitive maps learning is analyzed. Performance of the Structure Optimization Genetic Algorithm is compared with the Real-Coded Genetic Algorithm and the Multi-Step Gradient Method. Description of these methods is presented below.

A. Real-Coded Genetic Algorithm

Real-Coded Genetic Algorithm defines each chromosome as a floating-point vector, expressed as follows [28]:

$$W' = [W_{1,2}, ..., W_{1,n}, W_{2,1}, W_{2,3}, ..., W_{2,n}, ..., W_{n,n-1}]^T , \tag{4}$$

where $W_{j,i}$ is the weight of the connection between the j-th and the i-th concept.

Each chromosome in the population is decoded into a candidate FCM and its quality is evaluated on the basis of a fitness function according to the objective [9]. The aim of the analyzed learning process is to optimize the weights matrix with respect to the prediction accuracy. Fitness function can be described as follows:

$$fitness_p(J_l) = \frac{1}{a \cdot J(l) + 1} , \tag{5}$$

where a is a parameter, $a > 0$, p is the number of chromosome, $p = 1, ..., P$, P is the population size, l is the number of population, $l = 1, ..., L$, L is the maximum number of populations, $J(l)$ is the learning error function, described as follows:

$$J(l) = \frac{1}{(T-1)n_o} \sum_{t=1}^{T-1} \sum_{i=1}^{n_o} (Z_i^o(t) - X_i^o(t))^2 , \tag{6}$$

where t is discrete time of learning, T is the number of the learning records, $Z(t) = [Z_1(t), ..., Z_n(t)]^T$ is the desired FCM response for the initial vector $Z(t-1)$, $X(t) = [X_1(t), ..., X_n(t)]^T$ is the FCM response for the initial vector $Z(t-1)$, n is the number of the all concepts, n_o is the number of the output concepts, $X_i^o(t)$ is the value of the i-th output concept, $Z_i^o(t)$ is the reference value of the i-th output concept.

Each population is assigned a probability of reproduction. According to the assigned probabilities parents are selected and new population of chromosomes is generated. Chromosomes with above average fitness tend to receive more copies than those with below average fitness [9]. The basic operator of selection is a roulette wheel method. For each chromosome in population the probability of including a copy of such chromosome into the next population can be calculated according to the formula [11]:

$$P(p) = \frac{fitness_p(J_l)}{\sum\limits_{i=1}^{P} fitness_i(J_l)} , \tag{7}$$

where p is the number of the chromosome, P is the population size.

The population is mapped onto a roulette wheel, where each chromosome p is represented by a space that proportionally corresponds to $P(p)$. In the analysis a more effective ranking method of selection was used [3]. Selecting a copy of the chromosome into the next population is based on ranking by fitness [9].

The crossover operator is a method for sharing information between parents to form new chromosomes. It can be applied to random pairs of chromosomes and the likelihood of crossover depends on probability defined by the crossover probability P_c. The popular crossover operators are the single-point crossover and the uniform crossover [11].

The mutation operator modifies elements of a selected chromosome with a probability defined by the mutation probability P_m. The use of mutation prevents the premature convergence of genetic algorithm to suboptimal solutions [11]. In the analysis Mühlenbein's and random mutation were used. To ensure the survival of the best chromosome in the population, elite strategy was applied. It retains the best chromosome in the population [9].

The learning process stops when the maximum number of populations L is reached or the condition (8) is met.

$$fitness_{best}(J_l) > fitness_{max} \ , \tag{8}$$

where $fitness_{best}(J_l)$ is the fitness function value for the best chromosome, $fitness_{max}$ is a parameter.

B. Structure Optimization Genetic Algorithm

In this paper a new Structure Optimization Genetic Algorithm is proposed, which allows to select the most important for prediction task concepts and connections between them. SOGA defines each chromosome as a floating-point vector type (4) and a binary vector expressed as follows:

$$C' = [C_1, C_2, ..., C_n]^T \ , \tag{9}$$

where C_i is the information about including the i-th concept to the candidate FCM model, whereas C_i =1 means that the candidate FCM model contains the i-th concept, C_i =0 means that the candidate FCM model does not contain the i-th concept.

The quality of each population is calculated based on an original fitness function, described as follows:

$$fitness_p(J'_l) = \frac{1}{a \cdot J'(l) + 1} \ , \tag{10}$$

where a is a parameter, $a > 0$, p is the number of the chromosome, l is the number of population, $l = 1, ..., L$, L is the maximum number of populations, $J'(l)$ is the new learning error function with an additional penalty for highly complexity of FCM understood as a large number of concepts and non-zero connections between them [10]:

$$J'(l) = J(l) + b_1 \cdot \frac{n_r}{n^2} \cdot J(l) + b_2 \cdot \frac{n_c}{n} \cdot J(l) \ , \tag{11}$$

where t is discrete time of learning, T is the number of the learning records, b_1, b_1 are the parameters, $b_1 > 0$, $b_2 > 0$, n_r is the number of the non-zero weights of connections, n_c is the number of the concepts in the candidate FCM model, n is the number of the all possible concepts, $J(l)$ is the learning error function type (11).

Fig. 1 illustrates the steps of the learning and analysis of the FCM in modeling prediction systems with the use of population-based algorithms (SOGA and RCGA).

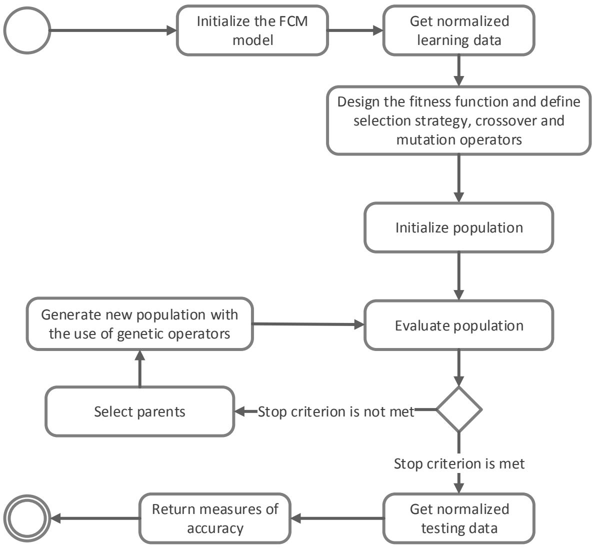

Initialize the FCM model
Get normalized learning data
Design the fitness function and define selection strategy, crossover and mutation operators
Initialize population
Generate new population with the use of genetic operators
Evaluate population
Select parents
Stop criterion is not met
Stop criterion is met
Return measures of accuracy
Get normalized testing data

Fig. 1. Activity diagram for population-based learning algorithm

In the paper the proposed algorithm was compared with the Real-Coded Genetic Algorithm and also with supervised learning based on Multi-Step Gradient Method.

C. Multi-Step Gradient Method

Multi-step algorithms of FCM learning are some kind of generalization of known one-step methods. Effectiveness of these methods in modeling of decision support systems was presented in [15], [25]. Multi-step supervised learning based on gradient method is described by the equation [14]:

$$W_{j,i}(t+1) = P_{[-1,1]}(\sum_{k=0}^{m_1} \alpha_k \cdot W_{j,i}(t-k) + \sum_{l=0}^{m_2}(\beta_l \cdot \eta_l(t) \cdot (Z_i(t-l) - X_i(t-l)) \cdot y_{j,i}(t-l))) \ , \tag{12}$$

where $\alpha_k, \beta_l, \eta_l$ are learning parameters, $k = 1, ..., m_1; l = 1, ..., m_2$, m_1, m_2 are the number of the steps of the method, t is a time of learning, $t = 0, 1, ..., T-1$, T is end time of learning, $X_i(t)$ is the value of the i-th concept, $Z_i(t)$ is the reference value of the i-th concept, $y_{j,i}(t)$ is a sensitivity function, $P_{[-1,1]}(x)$ is an operator of design for the set [-1,1], described as follows:

$$P_{[-1,1]}(x) = \frac{1 - e^{-x}}{1 + e^{-x}} \ , \tag{13}$$

Sensitivity function $y_{j,i}(t)$ is described by the equation:

$$y_{j,i}(t+1) = (y_{j,i}(t) + X_j(t)) \cdot F'(X_i(t) + \sum_{j \neq i} W_{j,i} \cdot X_j(t)) \ , \tag{14}$$

where $F'(x)$ is derivative of the stabilizing function.

Termination criterion can be expressed by the formula:

$$J(t) = \frac{1}{n} \sum_{i=1}^{n} (Z_i(t) - X_i(t))^2 < e \ , \tag{15}$$

where e is a level of error tolerance.

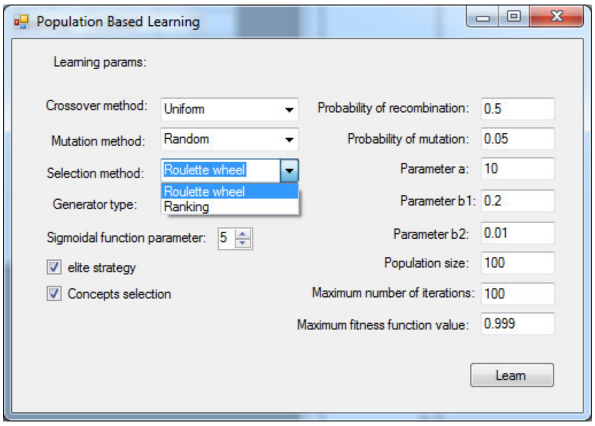
Simulation analysis of the presented algorithms performance was done with the use of ISEMK software tool. The basic features of ISEMK are described below.

IV. ISEMK Software Tool

ISEMK is a universal tool for modeling decision support systems based on FCMs [24]. It allows to:

- initialize the structure of the FCM model historical data (reading from .data files),
- visualize the structure of the FCM model,
- learn the FCM model based on Multi-Step Gradient Method and historical data (reading from .data files),
- learn the FCM model with the use of population-based learning algorithms (RCGA, SOGA) and historical data (reading from .data files),
- test the accurace of the learned FCMs operation based on historical data (reading from .data files) by calculating Mean Squared Error measure,
- export the results of learning and testing to .csv files,
- visualize the results of learning and testing in the form of charts.

Figure 2 shows an exemplary initialization of SOGA. Figure 3 shows an exemplary visualization of testing of the learned FCM operation.



Learning params:

Crossover method: Uniform

Mutation method: Random

Selection method: Roulette wheel (Roulette wheel, Ranking)

Generator type:

Sigmoidal function parameter: 5

☑ elite strategy

☑ Concepts selection

Probability of recombination: 0.5

Probability of mutation: 0.05

Parameter a: 10

Parameter b1: 0.2

Parameter b2: 0.01

Population size: 100

Maximum number of iterations: 100

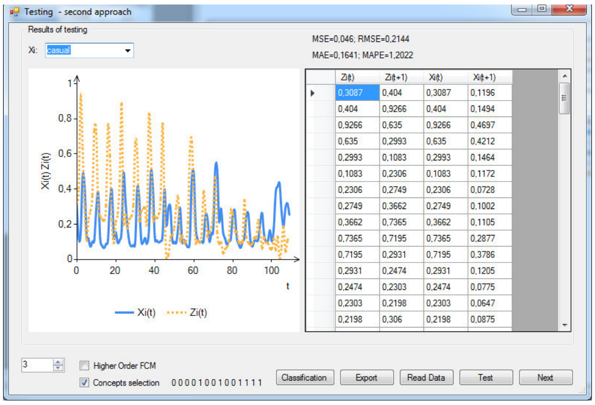
Maximum fitness function value: 0.999

Fig. 2. Exemplary visualization of population-based learning results

V. Simulation Results

To evaluate the proposed Structure Optimization Genetic Algorithm, historical data taken from the UCI Machine Learning Repository [6] were used. The dataset contains bike sharing counts aggregated on daily basis (731 days) and has the following fields:

- season (1:springer, 2:summer, 3:fall, 4:winter),
- year (0: 2011, 1:2012),
- month (1 to 12),
- hour (0 to 23),
- holiday : weather day is holiday or not,
- weekday : day of the week,
- working day : if day is neither weekend nor holiday is 1, otherwise is 0,



Xi: casual

MSE=0,046; RMSE=0,2144

MAE=0,1641; MAPE=1,2022

Zi(t)	Zi(t+1)	Xi(t)	Xi(t+1)
0,3087	0,404	0,3087	0,1196
0,404	0,9266	0,404	0,1494
0,9266	0,635	0,9266	0,4697
0,635	0,2993	0,635	0,4212
0,2993	0,1083	0,2993	0,1464
0,1083	0,2306	0,1083	0,1172
0,2306	0,2749	0,2306	0,0728
0,2749	0,3662	0,2749	0,1002
0,3662	0,7365	0,3662	0,1105
0,7365	0,7195	0,7365	0,2877
0,7195	0,2931	0,7195	0,3786
0,2931	0,2474	0,2931	0,1205
0,2474	0,2303	0,2474	0,0775
0,2303	0,2198	0,2303	0,0647
0,2198	0,306	0,2198	0,0875

3

□ Higher Order FCM

☑ Concepts selection 0 0 0 0 1 0 0 1 0 0 1 1 1 1

Fig. 3. Exemplary visualization of testing of learned FCM

- weather situation (1: Clear, Few clouds, Partly cloudy, Partly cloudy, 2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist, 3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds, 4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog),
- temperature (normalized temperature in Celsius),
- feeling temperature (normalized feeling temperature in Celsius),
- humidity (normalized humidity),
- wind speed (normalized wind speed),
- casual (count of casual users),
- registered (count of registered users),
- count (count of total rented bikes including both casual and registered).

The core data set is related to the two-year historical log corresponding to years 2011 and 2012 from Capital Bikeshare system, Washington D.C., USA (http://capitalbikeshare.com/system-data). The corresponding weather and seasonal information (http://www.freemeteo.com) were added. Normalization of the available data in the $[0, 1]$ range is needed in order to use the FCM model. Conventional min-max normalization (16) can be used.

$$f(x) = \frac{x - min}{max - min}\ , \tag{16}$$

where x is an input numeric value, min is the minimum of the dataset, max is the maximum of the dataset.

The aim of the study is one-step-ahead prediction of daily count of rented bikes based on the current values, environmental and seasonal settings. The dataset was divided into two subsets: learning (621 records) and testing (110 records) data. The learning process was accomplished with the use of Multi-Step Gradient Method, Real-Coded Genetic Algorithm and Structure Optimization Genetic Algorithm. Mean Squared Error for the output concepts (17) was used to estimate the performance of the FCM learning algorithms.

$$MSE = \frac{1}{n_o(T-1)} \sum_{t=1}^{T-1} \sum_{i=1}^{n_o} \left(Z_i^o(t) - X_i^o(t)\right)^2\ , \tag{17}$$

where t is time of testing, $t = 0, 1, ..., T - 1$, T is the number of the test records, $Z(t) = [Z_1(t), ..., Z_n(t)]^T$ is the desired FCM response for the initial vector $Z(t-1)$, $X(t) = [X_1(t), ..., X_n(t)]^T$ is the FCM response for the initial vector $Z(t-1)$, $X_i^o(t)$ is the value of the i-th output concept, $Z_i^o(t)$ is the reference value of the i-th output concept, n_o is the number of the output concepts.

The learning process was accomplished for various learning parameters, determined using experimental trial and error method. Optimal parameters of all analyzed algorithms were chosen based on minimization of the objective function, described as follows:

$$f_c = f(MSE, n_r, n_c) = 10000MSE + n_r + 10n_c \ , \quad (18)$$

where n_r is the number of the non-zero weights of the connections, n_c is the number of the concepts.

The best results of all three approaches were selected. Fig. 4 presents the structure of the FCM learned with the use of MGM (for the following parameters: $m_1 = 1$, $m_2 = 0$, $\alpha_0 = 1.5$, $\alpha_1 = -0.5$, $\beta_0 = 50$, $\lambda_0 = 100$, $e = 0.001$, $c = 3$). Table I shows the weights matrix for this map. Fig. 5 presents the structure of the FCM learned with the use of RCGA (for the following parameters: $P = 100$, $L = 200$, ranking selection, uniform crossover, Mühlenbein's mutation, $a = 10$, $h_{max} = 0.999$, $c = 3$). Table II shows the weights matrix for this map. Fig. 6 presents the structure of the FCM learned with the use of SOGA (for the following parameters: $P = 100$, $L = 500$, ranking selection, uniform crossover, random mutation, $a = 100$, $h_{max} = 0.999$, $c = 5$, $b_1 = 0.1$, $b_2 = 0.01$). Table III shows the weights matrix for this map.

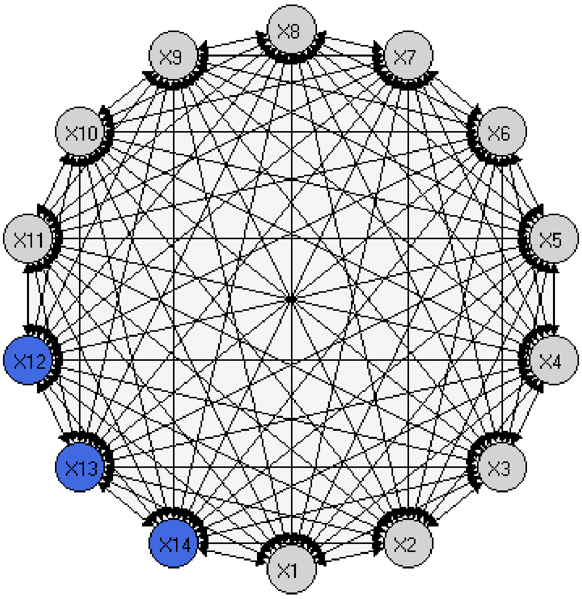


Fig. 4. The structure of the FCM learned with the use of MGM, where: X_1 – season, X_2 – year, X_3 – month, X_4 – holiday, X_5 – day of the week, X_6 – working day, X_7 – weather situation, X_8 – temperature, X_9 – feeling temperature, X_{10} – humidity, X_{11} – wind speed, X_{12} – count of casual users, X_{13} – count of registered users, X_{14} – count of total rented bikes

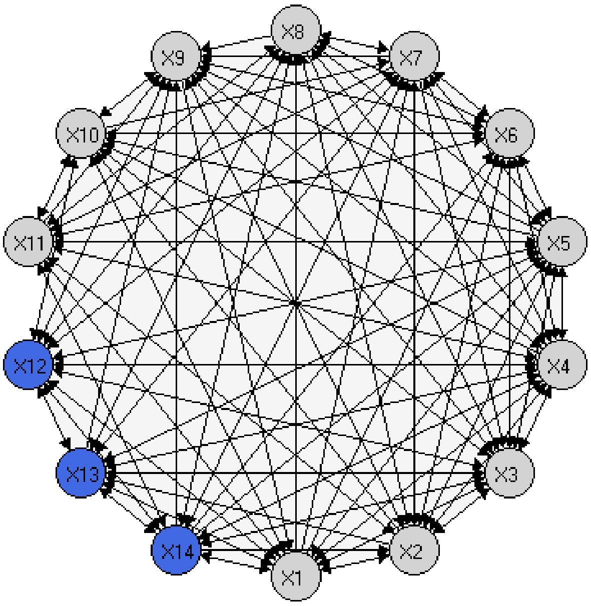


Fig. 5. The structure of the FCM learned with the use of RCGA, where: X_1 – season, X_2 – year, X_3 – month, X_4 – holiday, X_5 – day of the week, X_6 – working day, X_7 – weather situation, X_8 – temperature, X_9 – feeling temperature, X_{10} – humidity, X_{11} – wind speed, X_{12} – count of casual users, X_{13} – count of registered users, X_{14} – count of total rented bikes

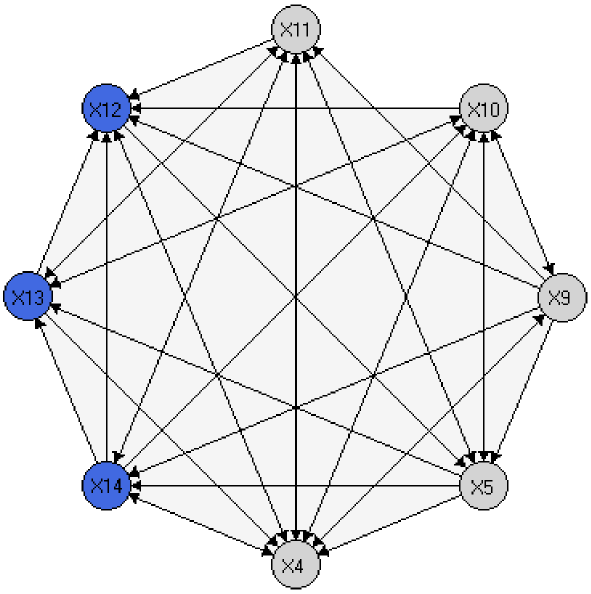


Fig. 6. The structure of the FCM learned with the use of SOGA, where: X_4 – holiday, X_5 – day of the week, X_9 – feeling temperature, X_{10} – humidity, X_{11} – wind speed, X_{12} – count of casual users, X_{13} – count of registered users, X_{14} – count of total rented bikes

Figures 7-9 show the exemplary results of testing of the learned FCMs operation. Table IV shows selected results of the comparative analysis of the Multi-Step Gradient Method, the Real-Coded Genetic Algorithm and the Structure Optimization Genetic Algorithm. Ranking selection and uniform crossover were used.

TABLE I
EXEMPLARY WEIGHTS MATRIX FOR THE MAP LEARNED WITH THE USE OF MGM

	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	X_{11}	X_{12}	X_{13}	X_{14}
X_1	0	-0.02	-0.01	-0.12	-0.06	0.06	-0.12	-0.07	-0.06	-0.08	-0.09	-0.11	-0.01	-0.03
X_2	-0.07	0	-0.07	-0.14	-0.06	-0.03	-0.13	-0.07	-0.09	-0.08	-0.11	-0.11	-0.07	-0.03
X_3	-0.08	-0.01	0	-0.12	-0.09	0.05	-0.12	-0.07	-0.07	-0.08	-0.09	-0.13	-0.02	-0.03
X_4	-0.05	-0.02	-0.04	0	0.08	0.05	-0.06	-0.02	0.07	0.02	-0.06	-0.08	0.08	0.07
X_5	-0.12	0.07	-0.12	-0.11	0	-0.04	-0.13	-0.11	-0.13	-0.11	-0.11	-0.17	-0.1	-0.09
X_6	-0.04	0.08	-0.08	-0.13	-0.08	0	-0.11	-0.04	-0.09	-0.07	-0.09	0.11	-0.12	-0.05
X_7	-0.06	-0.02	-0.05	-0.08	0.05	0.08	0	-0.01	-0.02	-0.09	-0.07	-0.09	0.04	0.05
X_8	-0.06	0.04	-0.06	-0.13	0.03	-0.03	-0.12	0	-0.08	-0.08	-0.09	-0.11	-0.03	-0.03
X_9	-0.07	0.03	-0.05	-0.12	-0.06	0.06	-0.12	-0.09	0	-0.08	-0.09	-0.12	0	-0.03
X_{10}	-0.04	0.07	-0.04	-0.13	0.01	-0.01	-0.12	-0.06	-0.07	0	-0.09	-0.08	-0.03	-0.02
X_{11}	-0.05	-0.01	-0.04	-0.08	0.04	0.06	-0.08	0	0.03	0.02	0	-0.07	0.04	0.04
X_{12}	-0.06	0	0.01	-0.12	0.04	0.11	-0.11	-0.08	0.06	-0.05	-0.09	0	0.11	0.04
X_{13}	-0.07	0.06	-0.07	-0.13	-0.09	-0.08	-0.12	-0.08	-0.09	-0.06	-0.1	-0.1	0	-0.07
X_{14}	-0.07	0.06	-0.07	-0.13	-0.08	0.03	-0.12	-0.08	-0.08	-0.06	-0.1	-0.12	-0.06	0

TABLE II
EXEMPLARY WEIGHTS MATRIX FOR THE MAP LEARNED WITH THE USE OF RCGA

	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	X_{11}	X_{12}	X_{13}	X_{14}
X_1	0	-0.99	0	0.99	-0.4	0	0	0	0.85	0.58	0.37	0.03	0.1	0.01
X_2	-0.31	0	0.11	-0.7	-0.22	-0.77	0.41	-0.99	0.93	-0.31	0	0	0.01	0.05
X_3	-0.4	0	0	0.16	0.65	-0.73	-0.07	0	0.17	0.34	-0.94	0.09	-0.03	0
X_4	0	-0.19	0.91	0	-0.43	-0.05	0.21	0.03	0.98	0.82	0	0	0	-0.79
X_5	-0.11	0	0.41	0	0	0	0	0.2	-0.35	0.08	0	0.1	-0.1	-0.08
X_6	0.55	0.93	-0.72	-0.15	0.86	0	-0.73	-0.52	0.59	-0.01	0	0.16	0	0
X_7	0	0.23	0.64	-0.19	-0.27	0	0	0.45	0.5	0	0	0.11	0.1	0
X_8	-0.81	0	0.65	0.27	0.89	-0.57	-0.39	0	0.58	0.47	0.09	-0.4	0	0.24
X_9	0.83	-0.1	0	0.42	0	-0.27	-0.68	0.01	0	0.08	0	0	-0.68	0.24
X_{10}	0.91	0	0	0	0.14	-0.86	0.18	-0.92	0	0	0	-1	-0.28	-0.87
X_{11}	-0.15	0	0	0.49	0.21	-0.58	0	0.77	0	-0.94	0	0	-0.55	-0.75
X_{12}	0.59	0.63	-0.5	0	-0.2	0.08	0.33	0	-0.27	0	0	0	0.42	-0.11
X_{13}	0	0	0.88	0	-0.77	0.8	-0.47	-0.64	0	0.45	0.51	0	0	0
X_{14}	-0.53	0.6	0.04	0.03	-0.46	0	0.32	-0.27	-0.6	0	0.85	0	0.03	0

TABLE III
EXEMPLARY WEIGHTS MATRIX FOR THE MAP LEARNED WITH THE USE OF SOGA

	X_4	X_5	X_9	X_{10}	X_{11}	X_{12}	X_{13}	X_{14}
X_4	0	0	-0.95	0.27	0.85	-0.58	0	-0.6
X_5	-0.51	0	0	0.9	-0.13	0	-0.38	-0.01
X_9	0.89	0.68	0	0.16	-0.76	-0.36	0	-0.77
X_{10}	0.79	0.11	-0.99	0	0	-0.15	-0.12	0
X_{11}	0.72	0.16	0	0	0	-0.63	-0.41	-0.51
X_{12}	0.19	0.55	0	0	0	0	0	0
X_{13}	0.97	0	0	-0.22	0.38	-0.57	0	0
X_{14}	-0.35	0	0	0.66	0.56	0.25	-0.29	0

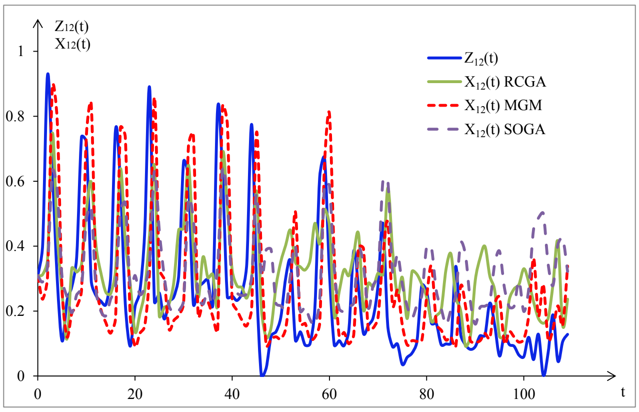


Fig. 7. Obtained values $X_{12}(t)$ and the desired values $Z_{12}(t)$ during testing

The lowest values of the objective function were obtained for the FCMs learned with the use of the Structure Optimization Genetic Algorithm. The minimum of $f_c = 471$ was achieved for the following parameters: $P = 100$, $L = 200$, ranking selection, uniform crossover, Mühlenbein's mutation, $a = 10$, $h_{max} = 0.999$, $c = 3$, $b_1 = 0.1$, $b_2 = 0.01$. Moreover, the second case of SOGA gives value of MSE lowest than Multi-Step Gradient Method and very similar to Real-Coded Genetic Algorithm. Also, from the fig. 6, we can say that bike-sharing rental process is highly related to holiday, day of the week, feeling temperature, humidity and wind speed. The results show the superiority of the SOGA. It allows to significantly simplify the FCM model by selecting the most important for prediction task concepts (e.g. 8 out of 14 possible) and connections (e.g. 41 out of 182 possible) with keeping the low values of MSE.

TABLE IV
CHOSEN RESULTS OF ANALYSIS OF THE MGM, RCGA, SOGA

Method	Learning parameters	MSE	n_r	n_c	f_c
MGM	$m_1 = 1$, $m_2 = 0$, $\alpha_0 = 1.4$, $\alpha_1 = -0.4$, $\beta_0 = 10$, $\lambda_0 = 100$, $e = 0.001$, $c = 5$	0.0429	174	14	743
MGM	$m_1 = 1$, $m_2 = 0$, $\alpha_0 = 1.5$, $\alpha_1 = -0.5$, $\beta_0 = 50$, $\lambda_0 = 100$, $e = 0.001$, $c = 3$	0.0418	179	14	737
MGM	$m_1 = 1$, $m_2 = 0$, $\alpha_0 = 1.5$, $\alpha_1 = -0.5$, $\beta_0 = 10$, $\lambda_0 = 10$, $e = 0.001$, $c = 5$	0.043	181	14	751
RCGA	$P = 100$, $L = 500$, $a = 100$, $fitness_{max} = 0.999$, $c = 5$, random mutation, $P_c = 0.8$, $P_m = 0.2$	0.0366	141	14	647
RCGA	$P = 100$, $L = 200$, $a = 10$, $fitness_{max} = 0.999$, $c = 3$, Mühlenbein's mutation, $P_c = 0.5$, $P_m = 0.1$	0.0333	127	14	601
RCGA	$P = 100$, $L = 500$, $a = 100$, $fitness_{max} = 0.999$, $c = 3$,random mutation, $P_c = 0.5$, $P_m = 0.1$	0.0533	135	14	808
SOGA	$P = 100$, $L = 500$, $a = 100$, b_1=0.2, b_2=0.05, $fitness_{max} = 0.999$, $c = 5$, random mutation, $P_c = 0.8$, $P_m = 0.2$	0.0463	37	8	580
SOGA	$P = 100$, $L = 200$, $a = 10$, b_1=0.1, b_2=0.01, $fitness_{max} = 0.999$, $c = 3$, Mühlenbein's mutation, $P_c = 0.5$, $P_m = 0.1$	0.035	41	8	**471**
SOGA	$P = 100$, $L = 500$, $a = 100$, b_1=0.2, b_2=0.005, $fitness_{max} = 0.999$, $c = 3$,random mutation, $P_c = 0.5$, $P_m = 0.1$	0.046	19	6	539

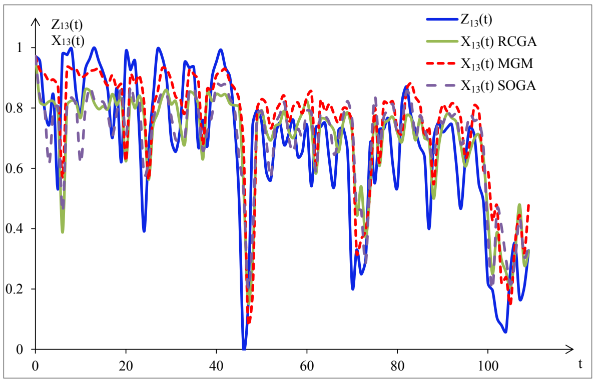


Fig. 8. Obtained values $X_{13}(t)$ and the desired values $Z_{13}(t)$ during testing

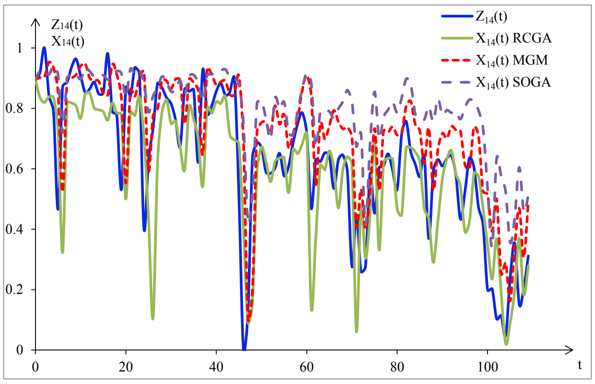


Fig. 9. Obtained values $X_{14}(t)$ and the desired values $Z_{14}(t)$ during testing

VI. CONCLUSION

In this paper we present a new approach for fuzzy cognitive maps learning allowing selection of the most important for the analyzed tasks concepts and connections between them. The proposed Structure Optimization Genetic Algorithm is described together with well-known methods for FCM learning. Comparative analysis of SOGA, RCGA and MGM was performed on the example of prediction of count of rented bikes. Simulation research was done in ISEMK. Selected results of simulation analysis of the developed algorithms performance are presented. The obtained results show that the proposed approach can significantly reduce the size of FCM by selecting the most important concepts and connections between them. The SOGA algorithm seems to be promising and effective method for modeling complex decision support systems based on fuzzy cognitive maps. There are plans of further analysis of the use of the Structure Optimization Genetic Algorithm.

REFERENCES

[1] S. Ahmadi, N. Forouzideh, S. Alizadeh, and E. Papageorgiou, "Learning Fuzzy Cognitive Maps using Imperialist Competitive Algorithm," *Neural Computing and Applications*, in press, http://dx.doi.org/10.1007/s00521-014-1797-4.

[2] M. Amer, A. J. Jetter, and T. U. Daim, "Scenario planning for the national wind energy sector through Fuzzy Cognitive Maps," *Proceedings of PICMET'13*, 2013, pp. 2153–2162.

[3] J. Arabas, *Lectures on genetic algorithms*, WNT, Warsaw, 2001.

[4] A. Buruzs, M. F. Hatwágner, A. Torma, and L. T. Kóczy, "Expert Based System Design for Integrated Waste Management," *International Scholarly and Scientific Research & Innovation*, vol. 8, no. 12, 2014, pp. 685–693.

[5] J. P. Carvalho, "On the semantics and the use of fuzzy cognitive maps and dynamic cognitive maps in social sciences," *Fuzzy Sets and Systems*, vol. 214, 2013, pp. 6–19, http://dx.doi.org/10.1016/j.fss.2011.12.009.

[6] H. Fanaee-T and J. Gama, "Event labeling combining ensemble detectors and background knowledge," *Progress in Artificial Intelligence*, Springer Berlin Heidelberg 2013, pp. 1–15, http://dx.doi.org/10.1007/s13748-013-0040-3.

[7] W. Froelich, E.I. Papageorgiou, M. Samarinasc, and K. Skriapasc, "Application of evolutionary fuzzy cognitive maps to the long-term prediction of prostate cancer," *Applied Soft Computing*, vol. 12, 2012, pp. 3810–3817, http://dx.doi.org/10.1016/j.asoc.2012.02.005.

[8] W. Froelich and J. Salmeron, "Evolutionary learning of fuzzy grey cognitive maps for the forecasting of multivariate, interval-valued time series," *International Journal of Approximate Reasoning*, vol. 55, 2014, pp. 1319–1335, http://dx.doi.org/10.1016/j.ijar.2014.02.006.

[9] D. B. Fogel, *Evolutionary Computation. Toward a new philosophy of machine inteligence* 3rd edition, John Wiley & Sons, Inc., Hoboken, New Jersey, 2006.

[10] L. Grad, “An example of feed forward neural network structure optimisation with genetic algorithm,” *BIULETYN INSTYTUTU AUTOMATYKI I ROBOTYKI*, no. 23, 2006, pp. 31–41.

[11] F. Herrera, M. Lozano, and J. L. Verdegay, “Tackling Real-Coded Genetic Algorithms: Operators and Tools for Behavioural Analysis,” *Artificial Intelligence Review*, vol. 12, 1998, pp. 265–319, http://dx.doi.org/10.1023/A:1006504901164.

[12] W. Homenda, A. Jastrzebska, and W. Pedrycz, “Nodes Selection Criteria for Fuzzy Cognitive Maps Designed to Model Time Series,” *Advances in Intelligent Systems and Computing*, vol. 323, 2015, pp. 859–870, http://dx.doi.org/10.1007/978-3-319-11310-4_75.

[13] W. Homenda, A. Jastrzebska, and W. Pedrycz, “Time Series Modeling with Fuzzy Cognitive Maps: Simplification Strategies. The Case of a Posteriori Removal of Nodes and Weights,” *Lecture Notes in Computer Science LNCS*, vol. 8838, 2014, pp. 409–420, http://dx.doi.org/10.1007/978-3-662-45237-0_38.

[14] A. Jastriebow and K. Piotrowska, “Simulation analysis of multistep algorithms of relational cognitive maps learning,” in: A. Yastrebov, B. Kuźmińska-Sołśnia and M. Raczyńska (Eds.) *Computer Technologies in Science, Technology and Education*, Institute for Sustainable Technologies - National Research Institute, Radom, 2012, pp. 126–137.

[15] A. Jastriebow and K. Poczęta, “Analysis of multi-step algorithms for cognitive maps learning,” *BULLETIN of the POLISH ACADEMY of SCIENCES TECHNICAL SCIENCES*, vol. 62, Issue 4, 2014, pp. 735–741, http://dx.doi.org/10.2478/bpasts-2014-0079.

[16] B. Kosko, “Fuzzy cognitive maps,” *International Journal of Man-Machine Studies*, vol. 24, no.1, 1986, pp. 65–75, http://dx.doi.org/10.1016/S0020-7373(86)80040-2.

[17] W. Lu, W. Pedrycz, X. Liu, J. Yang, and P. Li, “The modeling of time series based on fuzzy information granules,” *Expert Systems with Applications*, vol. 41, 2014, pp. 3799–3808, http://dx.doi.org/10.1016/j.eswa.2013.12.005.

[18] E. I. Papageorgiou , “Fuzzy Cognitive Maps for Applied Sciences and Engineering From Fundamentals to Extensions and Learning Algorithms,” *Intelligent Systems Reference Library*, vol. 54, Springer Verlag, 2014.

[19] E. I. Papageorgiou and W. Froelich, “Multi-step prediction of pulmonary infection with the use of evolutionary fuzzy cognitive maps,” *Neurocomputing*, vol. 92, 2012, pp. 28–35, http://dx.doi.org/10.1016/j.neucom.2011.08.034.

[20] E. I. Papageorgiou, K. E. Parsopoulos, C. D. Stylios, P. P. Groumpos, and M. N. Vrahtis, “Fuzzy Cognitive Maps Learning Using Particle Swarm Optimization,” *Journal of Intelligent Information Systems*, 25:1, 2005, pp. 95–121, http://dx.doi.org/10.1007/s10844-005-0864-9.

[21] E. I. Papageorgiou, K. Poczęta and C. Laspidou, “Application of Fuzzy Cognitive Maps to Water Demand Prediction,” *2015 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, Istanbul, Turkey, in press.

[22] E. I. Papageorgiou and J. L. Salmeron, “A Review of Fuzzy Cognitive Maps Research during the last decade,” *IEEE Transactions on Fuzzy Systems*, vol.21 , Issue: 1, 2013, pp. 66–79, http://dx.doi.org/10.1109/TFUZZ.2012.2201727.

[23] G. A. Papakostas, D. E. Koulouriotis, A. S. Polydoros, and V. D. Tourassis, “Towards Hebbian learning of Fuzzy Cognitive Maps in pattern classification problems," *Expert Systems with Applications*, vol. 39, 2012, pp. 10620–10629, http://dx.doi.org/10.1016/j.eswa.2012.02.148.

[24] K. Piotrowska, “Intelligent expert system based on cognitive maps,” *STUDIA INFORMATICA*, vol. 33, no 2A (105), 2012, pp. 605–616.

[25] K. Poczęta and A. Yastrebov, “Analysis of Fuzzy Cognitive Maps with Multi-Step Learning Algorithms in Valuation of Owner-Occupied Homes,” *2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, Beijing, China, 2014, pp.1029–1035, http://dx.doi.org/10.1109/FUZZ-IEEE.2014.6891587.

[26] J. L. Salmeron, “Fuzzy cognitive maps for artificial emotions forecasting," *Applied Soft Computing*, vol. 12, 2012, pp. 3704–3710, http://dx.doi.org/10.1016/j.asoc.2012.01.015.

[27] G. Słoń, “Application of Models of Relational Fuzzy Cognitive Maps for Prediction of Work of Complex Systems," *Lecture Notes in Artificial Intelligence LNAI*, vol. 8467, Springer Verlag, 2014, pp. 307–318, http://dx.doi.org/10.1007/978-3-319-07173-2_27.

[28] W. Stach, L. Kurgan, W. Pedrycz, and M. Reformat, “Genetic learning of fuzzy cognitive maps,” *Fuzzy Sets and Systems*, vol. 153, no. 3, 2005, pp. 371–401, http://dx.doi.org/10.1016/j.fss.2005.01.009.