



## Concept by Concept Learning of Fuzzy Cognitive Maps

M. Dodurka, Engin Yesil, Cihan Ozturk, Ahmet Sakalli, Cagri Guzey

► **To cite this version:**

M. Dodurka, Engin Yesil, Cihan Ozturk, Ahmet Sakalli, Cagri Guzey. Concept by Concept Learning of Fuzzy Cognitive Maps. Harris Papadopoulos; Andreas S. Andreou; Lazaros Iliadis; Ilias Maglogiannis. 9th Artificial Intelligence Applications and Innovations (AIAI), Sep 2013, Paphos, Greece. Springer, IFIP Advances in Information and Communication Technology, AICT-412, pp.577-586, 2013, Artificial Intelligence Applications and Innovations. <10.1007/978-3-642-41142-7\_58>. <hal-01459649>

**HAL Id: hal-01459649**

**<https://hal.inria.fr/hal-01459649>**

Submitted on 7 Feb 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

# Concept by Concept Learning of Fuzzy Cognitive Maps

M. Furkan Dodurka<sup>1,2</sup>, Engin Yesil<sup>1</sup>, Cihan Ozturk<sup>1</sup>,

Ahmet Sakalli<sup>1</sup>, Cagri Guzey<sup>1</sup>

<sup>1</sup>Istanbul Technical University, Faculty of Electrical and Electronics Engineering,  
Control Engineering Department, Maslak, TR-34469, Istanbul, Turkey

<sup>2</sup>GETRON Bilişim Hizmetleri A. Ş., Yıldız Teknik Üniversitesi Davutpaşa Kampüsü,  
Teknopark Binası B1 Blok, Esenler, 34220, Istanbul, Turkey

{dodurkam, yesileng, ozturkci, sakallia, guzey}@itu.edu.tr  
furkan.dodurka@getron.com

**Abstract.** Fuzzy cognitive maps (FCM) are fuzzy signed directed graphs with feedbacks; they are simple and powerful tool for simulation and analysis of complex, nonlinear dynamic systems. However, FCM models are created by human experts mostly, and so built FCM models are subjective and building a FCM model becomes harder as number of variables increases. So in the last decade several methods are proposed providing automated generation of fuzzy cognitive maps from data. The main drawback of the proposed automated methods is their weaknesses on handling with large number of variables. The proposed method brings out a new strategy called concept by concepts approach (CbC) approach for learning of FCM. It enables the generation of large sized FCM models with a high precision and in a rapid way using the historical data.

**Keywords:** Fuzzy cognitive maps, learning, density, global optimization.

## 1 Introduction

Cognitive maps were introduced for the first time by Axelrod [1] in 1976 in order to signify the binary cause-effect relationships of the elements of an environment. Fuzzy cognitive maps (FCM) are fuzzy signed directed graphs with feedbacks, and they can model the events, values, goals as a collection of concepts by forging a causal link between these concepts [2]. FCM nodes represent concepts, and edges represent causal links between the concepts. Most widely used aspects of the FCMs are their potential for use in learning from historical data and decision support as a prediction tool.

The main advantages of FCMs are their flexibility and adaptability capabilities [3]. As stated in [4], [5] and [6], there is an enormous interest in FCMs and this interest on the part of researchers and industry is increasing, especially in the areas of control [7], political and social sciences [8], business [9], medicine [10], robotics [11], environmental science [12], agriculture [13] and information technology [14].

Mainly, there are two types of FCMs called manual FCMs and automated FCMs. The unique difference between them is the way used for forming the FCMs. Manual FCMs are produced by experts manually and automated FCMs are produced by other information sources numerically [15]. Even sometimes, producing a FCM manually becomes difficult when the experts' interference could not be enough to solve the problem. Because of difficulties in manual FCM generation, the development of computational methods for learning FCM is required for automated FCMs. Lately, a large number of methods for learning FCM model structure have been proposed. These proposed methods can be summed in three groups named Hebbian-type learning methods, population-based (evolutionary) learning methods and hybrid learning algorithms [4].

A simple differential Hebbian learning law (DHL) for FCM is stated in [16]. This has been extended in [17] as a balanced differential learning algorithm for FCM. Further extensions, called nonlinear Hebbian learning (NHL) and Active Hebbian learning algorithm (AHL) are presented in [18] and [19], respectively. An improved version of the NHL method named data driven NHL (DDNHL) is proposed in [20]. Another study to train a FCM is proposed in which a new model for unsupervised learning and reasoning on a special type of cognitive maps that are realized with Petri nets [21]. All Hebbian-type learning methods have the goal to learn the connection matrix with single historical data set.

Rather than Hebbian-type learning methods, population-based learning methods are more in demand. The learning goal of population-based methods can be connection matrix with optimal weights or matching input pattern. Obtaining the connection matrix with optimal weights will lead FCM to its desired activation state values for each concept. Population based learning algorithms with connection matrix goal of learning that are recently studied can be listed as: Particle Swarm Optimization (PSO) [22], Genetic Strategy (GS) [23], Real-coded Genetic Algorithm (RCGA) [24], Simulated Annealing (SA) [15], tabu search [25], immune algorithm [26], Big Bang-Big Crunch (BB-BC) optimization algorithm [27], Extended Great Deluge Algorithm (EDGA) [28], Artificial Bee Colony (ABC) algorithm [29]. In addition, GA [30] and BB-BC [31] learning are used for goal oriented decision support systems on FCM.

The hybrid learning methods are implemented by combining the first two mentioned learning types (Hebbian-based learning (HL) and the population-based learning) for FCMs. There are two hybrid algorithms studied, one has combined NHL and differential evolution (DE) [32] and the other algorithm has combined RCGA and NHL algorithms [33]. The learning goals of these two the hybrid learning methods are the connection matrix and they use single historical data set.

In this study, a novel and comprehensive learning approach called concept by concept (CbC) for the development of fuzzy cognitive maps is proposed. The existing optimization based learning approaches try to find the weights between the concepts at once. The main difference of the proposed approach from the existing learning methods is focusing on only one concept and the links (arcs) to this concept first, and then learning the weights of these connection weights. Then the algorithm searches the next concept and its links and learns the weights. In order to use proposed CbC the historical data of all the concepts must be known. The proposed approach is able to

generate a FCM model from input data consisting of a single or multiple sequences of concept state vector values. Proposed CbC is applicable with any of population based optimization algorithms proposed in literature for learning of FCMs. The benefit of this learning approach is presented with two simulation examples.

## 2 A Brief Overview of Fuzzy Cognitive Maps

A fuzzy cognitive map  $F$  is a 4-tuple  $(N, W, C, f)$  [30] where;

$N = \{N_1, N_2, \dots, N_n\}$  is the set of  $n$  concepts forming the nodes of a graph.

$W: (N_i, N_j) \rightarrow w_{ij}$  is a function of  $N \times N$  to  $K$  associating  $w_{ij}$  to a pair of concepts  $(N_i, N_j)$ , with  $w_{ij}$  denoting a weight of directed edge from  $N_i$  to  $N_j$ . Thus  $W(N \times N) = (w_{ij}) \in K^{n \times n}$  is a connection matrix.

$C: N_i \rightarrow C_i$  is a function that at each concept  $N_i$  associates the sequence of its activation degrees such as for  $t \in N$ ,  $C_i(t) \in L$  given its activation degree at the moment  $t$ .  $C(0) \in L^n$  indicates the initial vector and specifies initial values of all concept nodes and  $C(t) \in L^n$  is a state vector at certain iteration  $t$ .

$f: R \rightarrow L$  is a transformation function, which includes recurring relationship on  $t \geq 0$  between  $C(t+1)$  and  $C(t)$ .

The sign of  $w_{ij}$  expresses whether the relation between the two concepts is direct or inverse. The direction of causality expresses whether the concept  $C_i$  causes the concept  $C_j$  or vice versa. Thus, there are three types of weights [22]:

$W_{ij} > 0$ , indicates positive causality,

$W_{ij} < 0$ , indicates negative causality,

$W_{ij} = 0$ , indicates no relation.

Values of concepts change as simulation goes on are calculated by the following formula [34]:

$$C_j(t+1) = f\left(\sum_{i=1}^N e_{ij} C_i(t)\right) \quad (1)$$

where  $C_i(t)$  is the value of  $i$ th node at the  $t^{\text{th}}$  iteration,  $e_{ij}$  is the edge weight (relationship strength) from the concept  $C_i$  to the concept  $C_j$ ,  $t$  is the corresponding iteration,  $N$  is the number of concepts, and  $f$  is the transformation (transfer) function.

In general, there are two kinds of transformation functions used in the FCM framework. The first one is the unipolar sigmoid function, where  $\lambda > 0$  decides the steepness of the continuous function  $f$  and transforms the content of the function in the interval  $[0, 1]$ .

$$f(x) = \frac{1}{1+e^{-\lambda x}} \quad (2)$$

The second transformation function, hyperbolic tangent, that has been used and which transforms the content of the function is in the interval  $[-1, 1]$ ,

$$f(x) = \tanh(\lambda x) = \frac{e^{\lambda x} - e^{-\lambda x}}{e^{\lambda x} + e^{-\lambda x}} \quad (3)$$

where  $\lambda$  is a parameter used to determine proper shape of the function. Both functions use  $\lambda$  as a constant for function slope.

### 3 Concept by Concept Learning Methodology

Most of the proposed population based learning approaches become time consuming when the number of concepts of FCM is relatively high. The main reason of this is the quadratic growth of the number of the parameters to be found as concept numbers increasing [35]. For a FCM with  $N$  concept, there will be  $N^2$  of weights to be found with learning algorithm if there is no prior knowledge. Therefore, in order to reduce the computational dimension of learning, concept by concept (CbC) method, which is usable with any of the population based algorithms, is proposed.

There are some factors that make the learning of FCM difficult or limit learning method. These factors can be listed as types of input data [5, 29], number of nodes ( $N$ ), knowledge of concepts' links, density measure, algorithms' search space, input nodes in FCM.

The proposed concept by concept (CbC) learning method develops a candidate FCM with any suitable population based global optimization algorithm from input and output data as given in Fig. 1. The input and output data are given as time series and that consist of a sequence of state vectors which describe a given system at consecutive iteration. The number of these successive iterations of the given historical data is called as the data length,  $K$ . Given a FCM with connection matrix ( $W_{N \times N}$ ) and collected data consisting output state vectors matrix of the FCM ( $Output_{K \times N}$ ), related to input state vectors matrix ( $Input_{K \times N}$ ). According to FCM iterative calculation formula (1), output state vector of  $t_{th}$  iteration is  $(t + 1)_{th}$  iteration of input state vector can be seen by (4)

$$Output(t)_n = Input(t + 1)_n \quad (4)$$

Output of a FCM can be found from (1) and this also can be expressed by (5).

$$Output_{K \times N} = f(Input_{K \times N} \times W_{N \times N}) \quad (5)$$

The goal of learning is to determine candidate connection matrix ( $\widehat{W}_{N \times N}$ ). As mentioned previously, assuming that all real input and output data is available, so (5) can be stated as follows:

$$Output_{K \times N} = f(Input_{K \times N} \times \widehat{W}_{N \times N}) \quad (6)$$

Therefore, learning problem becomes  $N \times N$  dimensional optimization problem. Since all real input and output data are known, this problem can be reduced to  $N$  total of optimization problem with  $N$  dimensional as given in Fig. 1. For example, given  $\widetilde{output}_n$  is a  $n_{th}$  column of  $Output_{K \times N}$  is a vector which consists outputs of  $n_{th}$  concept  $C_n$ . So (6) can be reduced to (7) given as

$$\widetilde{output}_n = f(Input_{K \times N} \times \widehat{w}_n) \quad (7)$$

where  $\hat{w}_n$  is a vector of  $n_{th}$  column of  $\hat{W}_{N \times N}$ . Concept  $C_n$  is only influenced from the weights in  $\hat{w}_n$ , thus, for determining candidate connection matrix  $\hat{W}_{N \times N}$  problem can be separated into  $N$  sub-learning problems; aiming to determine  $\hat{w}_n$  for  $n = 1, 2, \dots, N$ . Accordingly,  $\hat{W}_{N \times N}$  matrix can be found by carrying out  $N$  learning problem in (7) for  $Output_{(K,n)}$ .

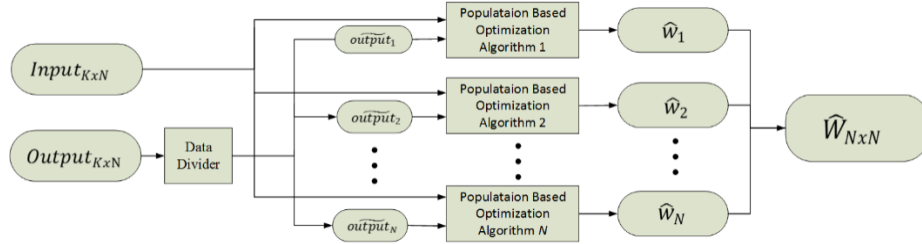


Fig. 1. Concept by Concept (CbC) FCM Learning Scheme

## 4 Simulation Examples

In this study, two systems with different characteristics are studied. The historical data is generated from the FCMs to use for the proposed CbC learning approach. The first system is a real-world FCM being a relatively easy problem with 13 nodes to show the proposed method's power of convergence accuracy. The second system is a synthetic (randomly generated) FCM with 100 nodes. This FCM is generated with 100 nodes and 100% density in order to show that the proposed method is able to cope with large scaled FCMs. For optimization algorithms in system 1 and 2, different types of cost functions that will be discussed in this section are selected to show the adaptiveness of the proposed method. In the simulation studies, BB-BC optimization algorithm [37] has been used as the global optimization method for CbC learning approach because of its fast convergence.

### 4.1 System 1: Suspension Viscosity FCM

Suspension viscosity FCM [36] is given in (8), stating what the factors affect suspension viscosity. The 13 concepts (C1 and C2 concepts are the input concepts) of suspension viscosity FCM model are given in Table 1. The transformation function given in (3) is utilized since the values of the nodes may fall within the range  $[-1, 1]$  and  $\lambda$  is chosen as 2 for generating the values of all concepts.

Table 1. Concepts of Suspension Viscosity FCM Model

<b>C1:</b> Gravity	<b>C7:</b> Liquid viscosity
<b>C2:</b> Mechanical properties of particles	<b>C8:</b> Effective particle shape
<b>C3:</b> Physicochemical interaction	<b>C9:</b> Effective particle size
<b>C4:</b> Hydrodynamic interaction	<b>C10:</b> Temperature
<b>C5:</b> Effective particle concentration	<b>C11:</b> Inter-particle attraction
<b>C6:</b> Particle-particle contact	<b>C12:</b> Floc structure formation
	<b>C13:</b> Shear rate

The connection matrix of the FCM built by the experts [36] is as in (8). The density value of this studied FCM is 39%. As mentioned above, it is assumed that the links of concepts nodes are known.

$$W = \begin{bmatrix} 0 & 0 & 0 & 0.03 & 0 & 0 & 0 & 0 & 0 & 0 & -0.13 & 0 & 0 \\ 0 & 0 & 0 & 0.23 & 0 & 0.17 & 0 & 0.33 & 0.37 & 0 & 0.10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.17 & 0.17 & 0 & 0.13 & 0.13 & 0 & 0.23 & 0.27 & 0 \\ 0 & 0 & 0.17 & 0 & 0.17 & 0.03 & 0 & 0 & 0 & 0 & 0 & -0.10 & 0 \\ 0 & 0 & 0 & 0.5 & 0 & 0.60 & 0.23 & 0 & 0 & 0 & 0.33 & 0.13 & -0.13 \\ 0 & 0 & 0.17 & 0.17 & 0.33 & 0 & 0 & 0 & 0.13 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.23 & 0 & -0.33 & 0 & 0 & 0 & 0 & -0.13 & -0.23 & -0.10 \\ 0 & 0 & 0 & 0.30 & 0 & 0 & -0.13 & 0 & 0 & 0 & -0.23 & 0.37 & 0 \\ 0 & 0 & 0 & 0.17 & 0 & 0.13 & 0.10 & 0 & 0 & 0 & 0.13 & 0.23 & 0 \\ 0 & 0 & 0.27 & 0.10 & 0.1 & 0.27 & -0.63 & 0.10 & 0 & 0 & 0.10 & 0.10 & 0.10 \\ 0 & 0 & 0.23 & 0 & 0.07 & 0.10 & 0.03 & 0 & 0 & 0 & 0 & 0.53 & 0 \\ 0 & 0 & 0 & 0 & 0.1 & 0 & 0.13 & 0.33 & 0.10 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.03 & 0.13 & -0.33 & -0.13 & 0 & 0.10 & 0 & -0.33 & 0 \end{bmatrix} \quad (8)$$

For input data 5 different historical data sets gathered for different initial 5 state vectors. All weights are searched with in [-1 1] bounds. Chosen cost function for learning is given as

$$\hat{J}_n = \frac{1}{(K-1)S} \sum_{z=1}^S \sum_{t=1}^K \|C_n(t) - \hat{C}_n\| \quad (9)$$

where  $C_n(t)$  is the given system response,  $\hat{C}_n$  is the candidate FCM response of the  $n^{\text{th}}$  concept for the initial state vector,  $S$  is number of gathered historical data sets, which is 5 for this example.

## 4.2 System 2: Randomly Generated FCM

Rather than, relatively small size FCMs with relatively small value of density parameter, a randomly generated FCM with 100 nodes and 100% density parameter is studied for this example, where no input concept exists. In addition, it is assume that there is no prior knowledge about links between concept nodes. The transformation function given in (3) is utilized since the values of the nodes may fall within the range [-1, 1] and  $\lambda$  is chosen 0.3.

In this simulation example, BB-BC optimization algorithm has been used as in System 1 and single input historical data is gathered. All weights are searched with in [-1 1] bounds. The cost function is selected as follows:

$$\hat{J}_n = \frac{1}{(K-1)S} \sum_{z=1}^S \sum_{t=1}^K (C_i(t) - \hat{C}_i)^2 \quad (10)$$

## 5 Results and Discussions

In order to show the effectiveness of the Concept by Concept (CbC) learning method, two FCMs with different sizes are studied. The study is divided into three phases: Learning phase, generalization capability testing phase and closeness of weights test-

ing phase. All the parameters used in these three phases are summarized in Table 2 where  $\theta$  is the fitness function coefficient, R is the randomly selected initial state vector and D is the number of parameters to be optimized.

**Table 2.** Parameters Used in BB-BC Learning and Generalization Capability Test Phases

System No	Population number (N)	Number of iterations	$\theta$	R	D
1	50	4000	$10^3$	100	66
2	100	10000	$10^3$	100	10000

For giving the results of the learning phase, the following criteria given in (11) is used.

$$J_1 = \frac{1}{N} \sum_{n=1}^N \hat{J}_n \quad (11)$$

where  $J_n$  is the selected cost function for using in learning phase given in (9) and (10).

In learning phase, 30 simulations are performed via FCM-GUI [38] for the 1<sup>st</sup> system, and for the 2<sup>nd</sup> system 10 simulations are performed via FCM-GUI.

In order to normalize, visualize and determine the convergence of the cost function given in (11) and the final value of the cost function, the following fitness function convergence performance which has the value [0, 1] is used [24]:

$$f = \frac{1}{\theta J_1 + 1} \quad (12)$$

where parameter  $\theta$  is a positive fitness function coefficient. In Fig. 2, the average, worst and best fitness convergence performances of these two different FCM simulations are illustrated.

For the best candidate FCM's due to learning phase results, their generalization capabilities are tested. For this purpose, the real FCM and the candidate FCM are simulated for R randomly chosen initial state vectors [24]. The new criterion is selected as follows:

$$J_2 = \frac{1}{R(K-1)NS} \sum_{r=1}^R \sum_{z=1}^S \sum_{t=1}^K \sum_{n=1}^N (C_n^r(t) - \widehat{C}_n^r)^2 \quad (13)$$

where  $C_n^r(t)$  is the value of n<sup>th</sup> node at iteration t for the data generated by original FCM model started from the r<sup>th</sup> initial state vector, similarly, is the value of n<sup>th</sup> node at iteration t for the data generated by candidate FCM model started from the r<sup>th</sup> initial state vector.

As the 3<sup>th</sup> phase, accuracy of the weights is tested. For this criteria, (14) is used where  $\widehat{W}_{ij}$  is weights of candidate FCM and  $W_{ij}$  is the weights of real FCM.

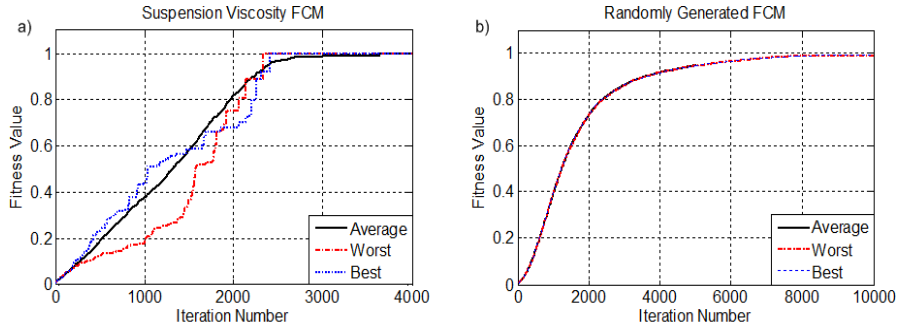
$$J_3 = \frac{1}{N^2} \sum_{n=1}^N (W_{ij} - \widehat{W}_{ij})^2 \quad (14)$$



In Table 3, minimum, maximum, mean and standard deviation values of error functions for learning, generalization capability and accuracy of weights phases are given. It is obvious that the obtained error function values are satisfactory.

**Table 3.** Learning and Generalization Capability and Accuracy of Weights Phases Results

	System #	Min $J_1$	Max $J_1$	Mean and Std. Dev.
<b>Learning Phase</b>	1	0	0	$0 \pm 0$
	2	$6.66 \times 10^{-6}$	$6.98 \times 10^{-6}$	$6.87 \times 10^{-5} \pm 1.33 \times 10^{-7}$
		Min $J_2$	Max $J_2$	Mean and Std. Dev.
<b>Generalization Phase</b>	1	0	0	$0 \pm 0$
	2	$3.95 \times 10^{-3}$	$1.42 \times 10^{-2}$	$8.97 \times 10^{-3} \pm 3.81 \times 10^{-3}$
		Min $J_3$	Max $J_3$	Mean and Std. Dev.
<b>Accuracy of Weights Phase</b>	1	0	0	$0 \pm 0$
	2	$1.26 \times 10^{-4}$	$1.39 \times 10^{-4}$	$1.34 \times 10^{-6} \pm 1.93 \times 10^{-6}$



**Fig. 2.** Fitness functions performance (12) (a) First system; (b) Second system

## 6 Conclusion

A new approach called concept by concept (CbC) for learning of FCMs is proposed. This approach is applicable when the full historical data of all concepts are available and the approach is independent from the used population based optimization learning method and the chosen cost function. In order to show the benefit of CbC learning two simulation examples are presented. The results show that the proposed learning method is very effective, and able to learn large scaled FCMs with a high accuracy. For the future work, the proposed CbC will be used with popular population based learning methods to compare the performances of these algorithms.

**Acknowledgments.** This project was supported by TÜBİTAK TEYDEB Industrial Research Funding Program Grant Number 7120842, awarded to GETRON Bilişim Hizmetleri A.Ş.

## References

1. Axelrod, R.: *Structure of Decision: the Cognitive Maps of Political Elites*. Princeton University Press, Princeton, New Jersey (1976).
2. Kosko, B.: Fuzzy cognitive maps. *International Journal of Man-Machine Studies*, vol. 24, pp. 65–75 (1986)
3. Aguilar, J.: A survey about fuzzy cognitive maps papers. *International Journal of Computational Cognition*, vol. 3 no. 2, pp. 27–33 (2005)
4. Papageorgiou, E. I.: Review study on fuzzy cognitive maps and their applications during the last decade. In: 2011 IEEE International Conference on Fuzzy Systems, IEEE Computer Society, Taipei, Taiwan, pp. 828-835 (2011)
5. Papageorgiou, E. I.: Learning algorithms for fuzzy cognitive maps: A review study. *IEEE Trans. Syst., Man Cybern. C Appl. Rev.*, vol. 42 no. 2, pp.150 -163 (2011)
6. Papageorgiou, E. I., Salmeron, J. L.: A Review of Fuzzy Cognitive Map research during the last decade. *IEEE Transactions on Fuzzy Systems*, vol. 21 no. 1, pp. 66-79 (2013)
7. Gonzalez, J.L., Aguilar, L.T., Castillo, O.: A cognitive map and fuzzy inference engine model for online design and self fine-tuning of fuzzy logic controllers. *International Journal of Intelligent Systems* 24 (11), pp. 1134-1173 (2009)
8. Andreou, A. S., Mateou, N. H., Zombanakis, G. A.: Soft computing for crisis management and political decision making: the use of genetically evolved fuzzy cognitive maps. *Soft Computing Journal*, vol. 9, No. 3, 194-210 (2006)
9. Glykas, M.: Fuzzy cognitive strategic maps in business process performance measurement. *Expert Systems with Applications*, vol. 40 no. 1, pp. 1–14 (2013)
10. Papageorgiou, E. I., Froelich, W.: Application of Evolutionary Fuzzy Cognitive Maps for Prediction of Pulmonary Infections. *IEEE Transactions on Information Technology in Biomedicine*, vol. 16 no. 1, pp. 143-149 (2012)
11. Motlagh, O., Tang, S. H., Ismail, N., Ramli, A. R.: An expert fuzzy cognitive map for reactive navigation of mobile robots. *Fuzzy Sets and Systems*, vol. 201, pp. 105–121 (2012)
12. Acampora, G., Loia, V.: On the Temporal Granularity in Fuzzy Cognitive Maps. *IEEE Transactions Fuzzy Systems*, vol 19 no 6, pp. 1040-1057 (2011)
13. Papageorgiou, E., Markinos, A., Gemtos, T.: Soft Computing Technique of Fuzzy Cognitive Maps to connect yield defining parameters with yield in Cotton Crop Production in Central Greece as a basis for a decision support system for precision agriculture application. In book: *Fuzzy Cognitive Maps*, edited by M. Glykas. Springer Verlag (2010)
14. Lee, K. C., Lee, S.: A causal knowledge-based expert system for planning an Internet-based stock trading system. *Expert Systems With Applications*, vol. 39 no. 10, pp. 8626-8635 (2012)
15. Alizadeh S., Ghazanfari, M.: Learning FCM by chaotic simulated annealing. *Chaos, Solutions & Fractals*, vol. 41 no. 3, pp. 1182-1190 (2009)
16. Dickerson, J. A., Kosko, B.: Fuzzy virtual worlds. *Artif. Intell. Expert*, vol. 7, pp. 25–31 (1994)
17. Vazquez, A.: A balanced differential learning algorithm in fuzzy cognitive maps. Technical report, Departament de Llenguatges I Sistemes Informatics, Universitat Politcnica de Catalunya (UPC) (2002)
18. Papageorgiou, E. I., Stylios, C. D., Groumpos, P.P.: Fuzzy cognitive map learning based on nonlinear Hebbian rule. In: *Australian Conference on Artificial Intelligence*, pp. 256–268 (2003)
19. Papageorgiou, E. I., Stylios, C. D., Groumpos, P.P.: Active Hebbian learning algorithm to train fuzzy cognitive maps. *Int. J. Approx. Reason.*, vol. 37 no. 3, pp. 219–249 (2004)

20. Stach, W., Kurgan, L., Pedrycz, W.: Data driven nonlinear Hebbian learning method for fuzzy cognitive maps. In: Proc. IEEE World Congr. Comput. Intell., pp. 1975–1981 (2008)
21. Konar A., Chakraborty, U. K.: Reasoning and unsupervised learning in a fuzzy cognitive map. *Inf. Sci.*, vol. 170, pp. 419–441 (2005)
22. Parsopoulos, K. E., Papageorgiou, E. I., Groumpos, P. P., Vrahatis, M. N.: A first study of fuzzy cognitive maps learning using particle swarm optimization. In: Proc. IEEE Congr. Evol. Comput, pp. 1440–1447 (2003)
23. Koulouriotis, D. E., Diakoulakis, I. E., Emiris, D. M.: Learning fuzzy cognitive maps using evolution strategies: a novel schema for modeling and simulating high-level behavior. In: Proc. IEEE Congr. Evol. Comput., pp. 364–371 (2001)
24. Stach, W., Kurgan, L., Pedrycz, W., Reformat, M.: Genetic learning of fuzzy cognitive maps. *Fuzzy Sets Syst.*, vol. 153, no. 3, pp. 371–401 (2005)
25. Alizadeh S., Ghazanfari, M., Jafari, M., Hooshmand, S.: Learning FCM by tabu search. *Int. J. Comput. Sci.*, vol. 2, no. 2, pp. 142–149 (2007)
26. Lin, C., Chen, K., He, Y.: Learning fuzzy cognitive map based on immune algorithm. *WSEAS Trans. Syst.*, vol. 6, no. 3, pp. 582–588 (2007)
27. Yesil, E., Urbas, L.: Big Bang - Big Crunch Learning Method for Fuzzy Cognitive Maps. In: International Conference on Control, Automation and Systems Engineering (2010)
28. Baykasoglu, A., Durmusoglu, Z. D. U., Kaplanoglu, V.: Training fuzzy cognitive maps via extended great deluge algorithm with applications. *Comput. Ind.*, vol. 62, no. 2, pp. 187–195 (2011)
29. Yesil, E., Ozturk, C., Dodurka, M. F., Sakalli, A.: Fuzzy Cognitive Maps Learning Using Artificial Bee Colony Optimization. In: IEEE Int. Conf. Fuzzy Systems (2013)
30. Khan, M. S., Chong, A.: Fuzzy cognitive map analysis with genetic algorithm. In: Ind. Int. Conf. Artif. Intell. (2003)
31. Yesil, E., Dodurka, M. F.: Goal-Oriented Decision Support using Big Bang-Big Crunch Learning Based Fuzzy Cognitive Map: An ERP Management Case Study. In: IEEE Int. Conf. Fuzzy Systems (2013)
32. Papageorgiou, E. I., Groumpos, P. P.: A new hybrid learning algorithm for fuzzy cognitive maps learning. *Appl. Soft Comput.*, vol. 5, pp. 409–431 (2005)
33. Zhu, Y., Zhang, W.: An integrated framework for learning fuzzy cognitive map using RCGA and NHL algorithm. In: Int. Conf. Wireless Commun., Netw. Mobile Comput. (2008)
34. Stach, W., Kurgan, L., Pedrycz, W.: A survey of fuzzy cognitive map learning methods In: P. Grzegorzewski, M. Krawczak, and S. Zadrozny, (Eds.), *Issues in Soft Computing: Theory and Applications*, Exit, pp. 71–84 (2005)
35. Stach, W., Kurgan, L., Pedrycz, W.: A divide and conquer method for learning large fuzzy cognitive maps. *Fuzzy Sets Syst.*, vol. 161, no. 19, pp. 2515–2532 (2010)
36. Banini, G.A., Bearman, R.A.: Application of fuzzy cognitive maps to factors affecting slurry rheology. *International Journal of Mineral Processing* vol. 52, pp. 233–244 (1998)
37. Erol, O. K., Eksin, I.: A new optimization method: Big Bang-Big Crunch. *Advances in Engineering Software*, vol. 37, pp. 106–111 (2006)
38. Yesil, E., Urbas, L., Demirsoy, A.: FCM-GUI: A graphical user interface for Big Bang-Big Crunch Learning of FCM, Fuzzy Cognitive Maps for Applied Sciences and Engineering – From Fundamentals to Extensions and Learning Algorithms. Ed. Elpiniki Papageorgiou. Vol. *Intelligent Systems Reference Library*. Springer (2013)