Available online at www.sciencedirect.com

SCIENCE DIRECT®

ELSEVIER

Int. J. Human-Computer Studies 64 (2006) 727–743

International Journal of
Human-Computer
Studies

www.elsevier.com/locate/ijhcs

Unsupervised learning techniques for fine-tuning fuzzy cognitive map causal links

Elpiniki I. Papageorgiou[a,*], Chrysostomos Stylios[b], Peter P. Groumpos[a]

[a]Laboratory for Automation and Robotics, Department of Electrical & Computer Engineering, Artificial Intelligence Research Center (UPAIRC), University of Patras, Rion 26500, Greece

[b]Department Communications, Informatics and Management, Technological Educational Institute (TEI) of Epirus, Artas

Received 16 March 2004; received in revised form 20 January 2006; accepted 24 February 2006
Communicated by E. Motta
Available online 18 April 2006

Abstract

Fuzzy Cognitive Maps (FCMs) constitute an attractive knowledge-based methodology, combining the robust properties of fuzzy logic and neural networks. FCMs represent causal knowledge as a signed directed graph with feedback and provide an intuitive framework which incorporates the experts' knowledge. FCMs handle available information and knowledge from an abstract point of view. They develop behavioural model of the system exploiting the experience and knowledge of experts. The construction of FCMs is based mainly on experts who determine the structure of FCM, i.e. concepts and weighted interconnections among concepts. But this methodology may not be a sufficient model of the system because the human factor is not always reliable. Thus the FCM model of the system may requires restructuring which is achieved through adjustment the weights of FCM interconnections using specific learning algorithms for FCMs. In this article, two unsupervised learning algorithms are presented and compared for training FCMs; how they define, select or fine-tuning weights of the causal interconnections among concepts. The implementation and results of these unsupervised learning techniques for an industrial process control problem are discussed. The simulations results of training the process system verify the effectiveness, validity and advantageous characteristics of those learning techniques for FCMs.

Keywords: Fuzzy cognitive maps; Learning algorithms; Hebbian learning; Process modeling and control

1. Introduction

Fuzzy Cognitive Map (FCM) is a soft computing technique capable of dealing with situations including uncertain descriptions using similar procedure such as human reasoning does. FCM is a modeling method based on knowledge and experience for describing particular domains using concepts (variables, states, inputs, outputs) and the relationships between them. The advantageous modelling features of FCMs, such as simplicity, adaptability and capability of approximating abstractive structures encourage us to enhance their structure using learning techniques, so that to broaden the FCMs functionality for complex problems.

In general, there is a great demand for modelling complex systems that can be achieved taking advantage of human like reasoning. There is also a need for advanced techniques which can take into consideration the various requirements of complex systems such as high autonomy and intelligence.

FCM was introduced by Kosko (1986), who expanded cognitive maps introducing causal algebra operating in the range of [0, 1] for propagating causality. Kosko proposed that negative influences be converted into positive ones by using the idea of *dis-concepts*. But this solution doubles the size of the concept set and increases computation time and space, particularly for large cognitive maps. In the same vein, Zhang and his colleagues proposed the POOL2 (Zhang et al., 1989), which is a generic system FCM for

[*]Corresponding author. Tel. +30261097293; fax: +3026120997309
E-mail addresses: epapageo@ee.upatras.gr (E.I. Papageorgiou), stylios@teleinfom.teiep.gr (C. Stylios), groumpos@ee.upatras.gr (P.P. Groumpos).

1071-5819/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ijhcs.2006.02.009

decision analysis. This system uses an approach in which both negative and positive assertions are weighted and kept separately based on the negative–positive–neutral (NPN) interval $[-1, 1]$ instead of values in $[0, 1]$. The same team went on to propose the D-POOL system (Zhang et al., 1992). The NPN causal inference was proposed to study a fuzzy time cognitive map with time lag on each arrow (Park and Kim, 1995).

FCMs have been used for representing knowledge (Taber, 1991), as artificial intelligence techniques appropriate for engineering applications, (Jain, 1997); for fault detection (Pelaez and Bowles, 1996), and modelling process control and supervision of distributed systems (Groumpos and Stylios, 2000; Stylios et al., 1999). FCMs have been used to model complex dynamical systems with chaotic characteristics, such as social and psychological processes and the organizational behaviour of a company (Craiger et al., 1996). FCMs have been also used for several tasks such as web-mining inference amplification (Lee et al., 2002), medical decision in radiotherapy, which is a complex process and is characterized by hard nonlinearities (Papageorgiou et al., 2003a), and computer-aided medical diagnosis for tumor characterization (Papageorgiou et al., 2003b).

Liu and Satur (1999) conducted extensive research on FCMs investigating their inference properties and they proposed contextual FCMs based on the object-oriented paradigm of decision support and applied contextual FCMs to geographical information systems (Liu, 2000). Other research efforts proposed FCMs to support the esthetical analysis of urban areas (Xirogiannis et al., 2004), and the management of relationships among organizational members in airline service (Kang et al., 2004). Furthermore, evaluation procedure for specifying and generating a consistent set of magnitudes for the causal relationships of a FCM, utilizing pair-wise comparison techniques have been presented (Muata and Bryson, 2004).

The development of a FCM requires that the expert provide information on both the sign and magnitude of each causal relationship. Although it is relatively easy to determine the relevant sign, experts often have difficulty in specifying the relevant magnitude. Thus, simple FCMs are often used to provide a first cut analysis of the given problem, but their value is often limited by the coarse granularity of the input information.

The methodology of developing FCMs is easily adaptable and relies on human expert experience and knowledge. However, it exhibits weaknesses in utilization of learning methods. The external intervention (typically from experts) for the determination of FCM parameters, the recalculation of the weights and causal relationships every time a new strategy is adopted, as well as the potential convergence to undesired regions for concept values are significant FCM deficiencies. It is necessary to overcome these deficiencies in order to improve efficiency and robustness of FCM. Weight adaptation methods are very promising as they can alleviate these problems by allowing the creation of less error prone FCMs where causal links are adjusted through a learning process.

FCM learning involves updating the strengths of causal links so that FCM concept values converge in a desired equilibrium region. A learning strategy is to modify FCM by fine-tuning its initial causal links based on ideas coming from the field of artificial neural networks (ANNs) training.

Learning methodologies for FCMs need to be developed in order to update the initial knowledge of human experts and to enhance the human experts' structural knowledge using training. So far there have been attempts to investigate and propose learning technique suitable for FCMs (Kosko, 1986; Koulouriotis et al., 2001; Aguilar, 2002; Papageorgiou et al., 2003c, d, 2004a; Papageorgiou and Groumpos, 2004; Khan et al., 2004; Stach et al., 2005).

Here two learning techniques have been proposed to adapt the cause–effect relationships of the FCM model improving the efficiency and robustness of FCMs. The introduction of FCM weight adaptation technique eliminates the deficiencies in the usage of FCM, enhances the dynamical behaviour and flexibility of the FCM model and enables it to learn nonlinear mappings. The aim of this paper is to present and compare the two proposed unsupervised learning algorithms for fine-tuning FCM causal links.

In this paper Section 2 describes the theoretical aspects of FCMs, while Section 3 presents a literature review on the learning algorithms for FCMs. Section 4 proposes the two learning algorithms, the Active Hebbian Learning (AHL) and Nonlinear Hebbian Learning (NHL) for FCM and how these learning techniques are implemented in general problems/case-studies. In Section 5, an industrial process control problem is described; the simulation results on modelling and controlling the process problem, using the proposed weight adaptation methods are presented in Section 6. Section 7 compares the two proposed learning techniques with each other as well as with other learning techniques for the same problem and concludes the paper.

2. Theoretical aspects of fuzzy cognitive maps

FCM is a soft computing technique used for causal knowledge acquisition and supporting causal knowledge reasoning process. FCM permits the necessary cycles for knowledge expression within their feedback framework of systems. FCMs are useful methods for exploring and evaluating the impact of inputs on dynamical systems that involve a set of objects such as processes, policies, events and value as well as the causal relationships between those objects.

More specifically, a FCM illustrates the whole system by a graph showing the effect and the cause among concepts. FCM is a simple way to describe the system's model and behaviour in a symbolic manner, exploiting the accumulated knowledge for the system. A FCM integrates the knowledge and experience with the operation of the system, as a result of the method by which it is constructed, i.e. using human experts who monitor, supervise and know the system behaviour in different circumstances. Moreover the FCM can utilize learning techniques to fine-tune the FCM causal links.

Nodes for the FCM stand for the concepts used to describe the behaviour of the system and are connected by sighed and weighted arcs representing the causal relationships that exist between the concepts (Fig. 1). All the values in the graph are fuzzy, thus concepts take values in the range between [0, 1] and the weights of the arcs are in the interval [−1, 1]. These weighted interconnections represent the direction and degree with which one concept influences the interconnected concepts.

The interconnection strength between two nodes C_j and C_i is w_{ji}, with w_{ji} taking on any value in the range −1 to 1. There are three possible types of causal relationships between concepts: $w_{ji} > 0$, which indicates positive causality between concepts C_j and C_i, $w_{ji} < 0$, which indicates negative causality between concepts C_j and C_i, and $w_{ji} = 0$, which indicates no relationship between C_j and C_i. The directional influences are presented as all-or-none relationships, thus the FCMs provide qualitative information about these relationships.

The value of each concept is calculated, computing the influence of other concepts to the specific concept, by applying the following calculation rule:

$$A_i^{(k)} = f\left(A_i^{(k-1)} + \sum_{j \neq i} A_j^{(k-1)} \cdot w_{ji}\right), \tag{1}$$

where $A_i^{(k)}$ is the value of concept C_i at iteration step k, $A_j^{(k-1)}$ the value of the interconnected concept C_j at iteration step $k-1$, w_{ji} the weighted arc from C_j and C_i, and f is a threshold function. Two threshold functions are usually used. The unipolar sigmoid function where $\lambda > 0$ determines the steepness of the continuous function $f(x) = 1/(1 + \mathrm{e}^{-\lambda x})$. When concepts can be negative and their values belong to the interval [−1, 1], threshold function $f(x) = \tanh(x)$ is used.

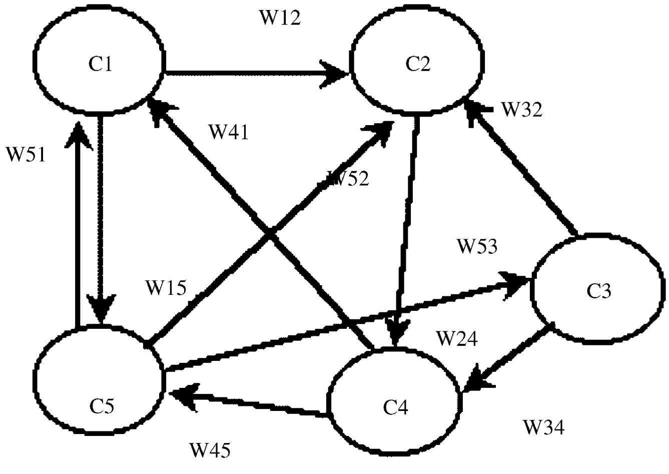

Fig. 1. A simple Fuzzy Cognitive Map.

The simplicity of the FCM model becomes apparent from its mathematical representation and operation. Suppose that a FCM consists of n-concepts. An $1 \times n$ matrix **A** represents the values of the n concepts and an $n \times n$ matrix **W** represents the causality of the relationships. Each element e_{ij} of the matrix **W** indicates the value of the weight w_{ji} between concept C_j and C_i. Eq. (1) can be transformed as follows to describe the FCM operation with a compact mathematical equation:

$$\mathbf{A}^{(k)} = f\left(\mathbf{A}^{(k-1)} + \sum \mathbf{A}^{(k-1)} \cdot \mathbf{w}\right) \tag{2}$$

where $\mathbf{A}^{(k)}$ is the matrix with values of concepts at iteration step k, and f is the threshold function.

The FCM model of the system takes the initial values of concepts based on measurements from the real system and it is free to interact. The interaction is also caused by the change in the value of one or more concepts.

This interaction continues until the model:

- Reaches equilibrium at a fixed point, with the output concept values stabilizing at fixed numerical values.
- Exhibits limit cycle behaviour, with the concept values falling in a loop of numerical values under a specific-time period.
- Exhibits a chaotic behaviour, with each value reaching a variety of numerical values in a non-deterministic, random way.

Simplest FCMs act as asymmetrical networks of threshold or continuous concepts and converge to an equilibrium point or limit cycles. They differ from neural networks in the way they are developed as they are based on extracting knowledge from experts. FCMs have nonlinear structure of their concepts and differ in their global feedback dynamics (Kosko, 1992, 1997).

2.1. Constructing fuzzy cognitive maps

The development and construction method of FCM has great importance for its potential to sufficiently model a system. The method used is dependent on the group of experts who operate, monitor, supervise the system and they develop the FCM model. This methodology extracts the knowledge on the system from the experts and exploits their experience of the system's model and behaviour (Stylios and Groumpos, 2000).

The group of experts determines the number and kind of concepts that comprise the FCM. An expert from his/her experience knows the main factors that describe the behaviour of the system; each of these factors is represented by one concept of the FCM. Experts know which elements of the system influence other elements; for the corresponding concepts they determine the negative or positive effect of one concept on the others, with a fuzzy degree of causation. In this way, an expert transforms his/her knowledge in a dynamic weighted graph, the FCM. According to the developing methodology experts are forced to think about and describe the existing relationship between the concepts and thus justify their suggestions. Each expert, indeed, determines the influence of one concept on another as "negative" or "positive" and then evaluates the degree of influence using a linguistic variable, such as "strong influence", "medium influence", "weak influence", etc.

More specifically, the causal interrelationships among concepts are declared using the variable *Influence* which is interpreted as a linguistic variable taking values in the universe $U = [-1, 1]$. Its term set T(*influence*) is suggested to comprise nine variables. Using nine linguistic variables, an expert can describe in detail the influence of one concept on another and can discern between different degrees of influence. The nine variables used here are: T(*influence*) = {negatively very strong, negatively strong, negatively medium, negatively weak, zero, positively weak, positively medium, positively strong and positively very strong}. The corresponding memberships functions for these terms are shown in Fig. 2 and they are μ_{nvs}, μ_{ns}, μ_{nm}, μ_{nw}, μ_z, μ_{pw}, μ_{pm}, μ_{ps} and μ_{pvs}.

Thus, every expert describes each interconnection with a fuzzy linguistic variable from the set, which describes the relationship between the two concepts and determines the grade of causality between the two concepts. Then, all the proposed linguistic variables suggested by experts, are aggregated using the SUM method and an overall linguistic weight is produced, which with the defuzzification method of Centre of Gravity (COG) (Lin and Lee, 1996), is transformed to a numerical weight w_{ji}, belonging to the interval $[-1, 1]$. A detailed description of the development of FCM model is given in (Stylios and Groumpos, 2004).

3. Learning methods for fuzzy cognitive maps

Utilization of appropriate learning algorithms can overcome the most significant weaknesses of the FCMs, namely the potential convergence to undesired regions and the recalculation of the weights when new strategies are adopted. The learning procedure is a technique which increases the efficiency and robustness of FCMs, contributing to more intelligent methods by modifying the FCM weight matrix. Moreover, the learning rules supply FCMs with useful characteristics such as the ability to learn arbitrary nonlinear mappings, capability to generalize situations, adaptively.

Experts involved in the construction of FCM determine concepts and causality among them. This approach may yield to a distorted model, since it is possible that experts have not considered appropriate factors and they may have assigned inappropriate causality weights among FCM concepts. The best conductance of FCMs is obtained by combining them with neural network characteristics and integrating their advantages. Specifically, neural learning techniques are used to train the FCM and determine appropriate weights of interconnections among concepts. The result is a hybrid neurofuzzy system. Learning methods have been proposed for FCM training, where the gradient for each weight is calculated by the application of the general rule:

$$w'_{ij} = g(w_{ij}, A_i, A_j, A'_i, A'_j). \tag{3}$$

Learning rules can train FCMs, meaning adjusting the interconnections between concepts, as in the case of synapses of neural networks.

The learning algorithms, proposed for FCMs are mostly based on ideas coming from the field of ANNs training. Adaptation and learning methodologies based on unsupervised Hebbian-type rules to adapt the FCM model and adjust its weights were proposed for the first time by Kosko (1986). Kosko presented the Differential Hebbian Learning (DHL), as a form of unsupervised learning, to train FCM, but without mathematical formulation and implementation in any problem (Kosko, 1992; Dickerson and Kosko, 1994). The balanced differential learning algorithm to train FCM from data, based on the DHL has been proposed (Huerga, 2002). This algorithm is a modified version of the

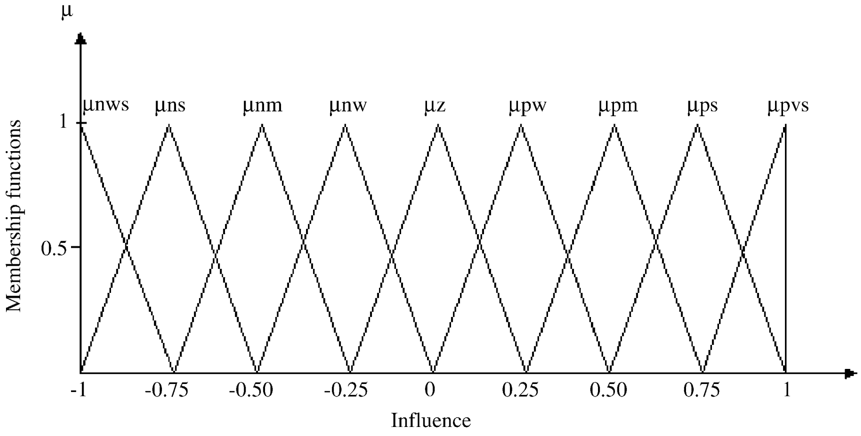

Fig. 2. Membership functions of the linguistic variable *Influence*.

DHL and seems to work better in learning patterns and modelling a given domain than the classical approach. Another proposed approach for FCMs training is the Adaptive Random FCMs based on the theoretical aspects of Random Neural Networks (Aguilar, 2002).

Recently, there have been proposed two approaches for FCM training, the AHL and the NHL algorithms. The AHL algorithm have been proposed and implemented successfully in a practical control problem (Papageorgiou et al., 2004a). This algorithm takes into consideration the initial experts' knowledge and experience starting from the initial values of elements of the weight matrix as they are derived from the summation of experts' opinions. The AHL introduced a sequence of activation concepts depending on the specific problem's configuration and characteristics. The AHL is described briefly in the following section.

The initial idea and description of using the nonlinear Hebbian rule in FCMs has been proposed in (Papageorgiou et al., 2003b), but the mathematical formalism for incorporating the NHL into FCM structure as well as the methodology for implementing this algorithm in a case-study scenario are introduced and presented in this article.

In addition to this some methods based on Evolutionary Computation techniques have been proposed for FCMs. Particle Swarm Optimization (PSO) methods were proposed for FCM learning giving very promising results (Papageorgiou et al., 2003d, 2004b). PSO algorithms belong to swarm intelligence, a rapidly growing area of artificial intelligence. This method provides a search procedure, which optimizes a problem-depended fitness function $\varphi(\,)$, by maintaining and evolving a swarm of candidate solutions. The individual of the swarm yielding the best fitness value throughout all generations gives the optimal solution. Using this learning approach a number of appropriate weight matrices were derived leading the system to desired convergence regions. This approach is very fast and efficient to calculate the optimum cause–effect relationships of the FCM model and to overcome a main drawback of the FCM, which is the recalculation of the weights every time a new real case is adopted.

Another approach for learning FCM connection matrix involves application of Evolution Strategies (ESs) (Koulouriotis et al., 2001). This technique is exactly the same used for neural networks training. One of its main drawbacks is that it does not take into consideration the initial structure and experts' knowledge for the FCM model, but uses data sets determining input and output patterns in order to define the cause–effect relationships which satisfy the fitness function. Another main drawback is the need for multiple state vector sequences (input/output pairs), which might be difficult to obtain for many real-life problems. The calculated weights appear as large deviations from the actual FCM weights. In real problems they do not appear to have any accepted physical meaning.

Recently, two different approaches based on application of genetic algorithms for learning FCM connection matrix have been proposed. The first approach involving genetic algorithms performs a goal-oriented analysis of FCM (Khan et al., 2004). This learning method did not aim to compute the weight matrix, but to find the initial state matrix, which leads the predefined FCM (with a fixed weight matrix) to converge to a given fixed-point attractor or limit cycle solution. They viewed the problem of the FCM backward inference as one of optimization, and they applied a genetic algorithm-based strategy to search for the optimal stimulus state.

The second more powerful genetic algorithm-based method has been proposed to develop FCM connection matrix which is based on historical data consisting of one sequence of state vectors. It uses a real-coded genetic algorithm (RCGA) which allows eliminating expert involvement during development of the model and learns the connection matrix for a FCM that uses continues transformation function, which is a more general problem that the one considered in (Stach et al., 2005). The main advantage of this method is the absence of human intervention but the RCGA method needs investigation in terms of its convergence and more investigation towards to associate the GA parameters with the characteristics of a given experimental data. Its usefulness is restricted in only specific problem domains. Learning algorithms for FCMs based on evolutionary computation methods need more investigation.

4. Unsupervised learning techniques

4.1. The active hebbian learning algorithm

A new unsupervised learning algorithm has been proposed recently suitable for training FCMs, namely AHL algorithm, which introduces the sequence of activation concepts (Papageorgiou et al., 2004a). The novelty of this algorithm is based on accepting sequence of influence from one concept to another, in this way the interaction cycle is dividing in steps. When the experts develop the FCM, they are asked to determine the sequence of activation concepts, the activation steps and the activation cycle. At every activation step, one (or more) concept(s) becomes Activated concept(s), triggering the other interconnecting concepts, and in turn, at the next simulation step, may become Activation concept. When all the concepts have become Activated concepts, the simulation cycle has closed and a new one starts until the system converges in an equilibrium region. An activation cycle consists of steps; at each activation step one or more concepts are the Activation concepts that influence the interconnected concepts until the termination of the sequence of activation closes the cycle.

In addition to the determination of sequence of activation concepts; experts select a limited number of concepts as outputs for each specific problem which are defined as the Activation Decision Concepts (ADCs).

These concepts are in the centre of interest; they stand for the main factors and characteristics of the system, known as outputs and their values represent the final state of the system.

Suppose there is the FCM shown on Fig. 3, where experts determined the following activation sequence: $C_1 \to C_2$, $C_j \to C_i \to C_m \to C_n$. At second step of the cycle, according to the activation sequence concept C_j is the triggering concept that influences concept C_i, as shown in Fig. 3. This concept C_j is declared the Activation concept, with the value A_j^{act} and triggers the interconnected corresponding concept C_i, which is the Activated concept. At the next iteration step, concept C_i influences the other interconnected concepts C_m and so forth. The learning algorithm has asynchronous stimulation mode that means when concept C_j becomes the Activation concept that triggers C_i the corresponding weight w_{ji} of the causal interconnection is updated and the modified weight $w_{ji}^{(k)}$ is derived for each iteration step k.

Fig. 3 is a shot of the FCM model during the activation sequence. The FCM-model consists of n-nodes and it is the 2nd activation step where the Activation concept C_j, influences the Activated concept C_i. The following parameters are depicted:

C_i: is the ith concept with value $A_i(k)$, $1 \leqslant i \leqslant n$.
w_{ji}: is the weight describing the influence from C_j to C_i.
$A_j^{\text{act}}(k)$: is the activation value of concept C_j, which is triggering the interconnected concept C_i.
γ: is the weight decay parameter.
η: is the learning rate parameter, depending on simulation cycle c.
$A_i(k)$: is the value of Activated concept C_i, at iteration step k.

The value $A_i(k+1)$ of the Activated concept C_i, at iteration step $k+1$, is calculated, computing the influence of other Activation concepts with values A_l^{act} to the specific concept C_i due to modified weights $w_{li}(k)$ at iteration step k, through the equation:

$$A_i(k+1) = f\Big(A_i(k) + \sum_{l \neq i} A_l^{\text{act}}(k) \cdot w_{li}(k)\Big), \tag{4}$$

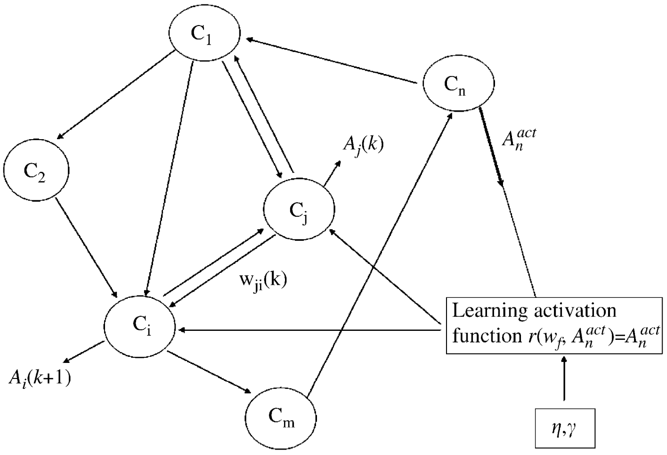

Fig. 3. The activation weight-learning process for FCMs.

where A_l are the values of concepts C_l that influence the concept C_i, and $w_{li}(k)$ are the corresponding weights that describe the influence from C_l to C_i.

For example, in Fig. 3, l takes values 1,2 and j, and A_1, A_2 and A_j are the values of concepts C_1, C_2 and C_j that influence C_{i}. Thus value A_i of concept, after triggering at step $k+1$, is calculated:

$$\begin{aligned} A_i(k+1) = f(A_i(k) &+ A_1^{\text{act}}(k) \cdot w_{1i}(k) \\ &+ A_2^{\text{act}}(k) \cdot w_{2i}(k) + A_j^{\text{act}}(k) \cdot w_{ji}(k)). \end{aligned} \tag{5}$$

The AHL algorithm relates the values of concepts and values of weights to the FCM model. We introduced a mathematical formalism for incorporating the learning rule, with the learning parameters and the introduction of the sequence of activation (Papageorgiou et al., 2004a, b).

The proposed rule has the general mathematical form:

$$w_{ji}(k) = (1-\gamma) \cdot w_{ji}(k-1) + \eta \cdot A_j^{\text{act}}(k-1) \cdot A_i(k-1), \tag{6}$$

where the coefficients η, γ are positive learning factors called learning parameters.

In order to prevent indefinite growing of weight values, we suggest normalization of weight at value 1, $\|\mathbf{W}\| = 1$, at each step update:

$$w_{ji}(k) = \frac{(1-\gamma) \cdot w_{ji}(k-1) + \eta \cdot A_j^{\text{act}}(k-1) \cdot A_i(k-1)}{\left[\sum_{\substack{j=1 \\ j \neq i}} ((1-\gamma) \cdot w_{ji}(k-1) + \eta \cdot A_j^{\text{act}}(k-1) \cdot A_i(k-1))^2\right]^{1/2}}, \tag{7}$$

where the addition in the denominator covers all of the interconnections from the Activation concepts C_j to the Activated concepts C_i.

For low learning rates of parameters η, γ, Eq. (7) can—without any loss of precision—be simplified to:

$$\begin{aligned} w_{ji}(k) = (1-\gamma) \cdot w_{ji}(k-1) &+ \eta \cdot A_j^{act}(k-1) \\ &\times [A_i(k-1) - w_{ji}(k-1) \cdot (A_j^{act}(k-1))]. \end{aligned} \tag{8}$$

Eq. (1) that calculates the value of each concept of FCM takes the form of Eq. (4), where the value of weight $w_{ji}(k)$ is calculated using Eq. (8).

The learning parameters η and γ are positive scalar factors. The learning rate parameter η is exponentially attenuated with the number of activation–simulation cycles c so that the trained FCM converges fast. Thus $\eta^{(c)}$ is selected to be decreased where the rate of decrease depends on the speed of convergence to the optimum solution and on the updating mode. Thus, the following equation is proposed:

$$\eta^{(c)} = b_1 \exp(-\lambda_1 \cdot c). \tag{9}$$

Depending on the problem's constraints and the characteristics of the specific case, the parameters b_1 and λ_1 may be values within the following bounds $0.01<b_1<0.09$ and $0.1<\lambda_1<1$, which are determined using experimental trial and error method for fast convergence.

The parameter γ is the weight decay coefficient which may decrease depending on the number of activation cycles c. The parameter γ can be selected for each specific problem to ensure that the learning process converges in a desired steady state. If the parameter γ is selected as a decreasing function at each activation cycle c, the following form is proposed:

$$\gamma^{(c)} = b_2 \exp(-\lambda_2 \cdot c), \tag{10}$$

where b_2 and λ_2 are positive constants determined by a trial and error experimental process. These values influence the rate of convergence to the desired region and the termination of the algorithm.

In addition, in AHL algorithm, two criteria functions have been proposed (Papageorgiou et al., 2004a). The first one is the criterion function J that examines the desired values of outputs concepts, which are the values of Activation Concepts we are interested in.

The criterion function J has been suggested as

$$J = \sqrt{\sum_{j=1}^{m}([ADC_j - A_j^{\min}]^2 + [ADC_j - A_j^{\max}]^2)}, \tag{11}$$

where $A_j^{\min}$ is the minimum target value of concept ADC_j and $A_j^{\max}$ is the corresponding maximum target value of ADC_j. At the end of each cycle, the value of J calculates the Euclidean distance of ADC_j value from the minimum and maximum target values of the desired ADC_j, respectively. The minimization of the criterion function J is the ultimate goal, according to which we update the weights and determine the learning process.

One more criterion for this learning algorithm of FCMs has been proposed. This second criterion is determined by the variation of the subsequent values of ADC_j concept, for simulation cycle c, yielding value e, which has to be minimum and takes the form:

$$|ADC_j^{(c+1)} - ADC_j^{(c)}| < e, \tag{12}$$

where ADC_j is the value of jth concept.

The term e is a tolerance level keeping the variation of values of ADC(s) as low as possible and it is proposed as equal to $e=0.001$, satisfying the termination of iterative process.

Thus, for training FCM using the asynchronous AHL algorithm two criteria functions have been proposed. The first one is the minimization of the criterion function J and the second one is minimization of the variation of the two subsequent values of ADCs, represented in Eqs (11) and (12), respectively, in order to determine and terminate the iterative process of the learning algorithm.

The proposed algorithm is based on defining a sequence of concepts that means distinction of FCM concepts as inputs, intermediates and outputs, which depends on the modeled system and the focus of experts. During the training phase a limited number of concepts are selecting as outputs (those we want to estimate their values). The expert's intervention is the only way to address this definition. This learning algorithm extracts the valuable knowledge and experience of experts and can increase the operation of FCMs and implementation in real case problems just by analysing existing data, information and experts' knowledge about the given systems.

The training process implementing the AHL into an n-concept FCM is described analytically in (Papageorgiou et al., 2004a). The schematic representation of this training process is given in Fig. 4. This learning algorithm drives the system to converge in a desired region of concepts values within the accepted-desired bounds for ADCs concepts.

4.2. Nonlinear Hebbian Learning algorithm for Fuzzy Cognitive Maps

The second proposed algorithm for training FCMs is based on the nonlinear Hebbian-type learning rule for ANNs learning (Hebb, 1949; Oja, 1989; Hassoun, 1995). This unsupervised learning rule has been modified and adapted for the FCM case, introducing the NHL algorithm for FCMs.

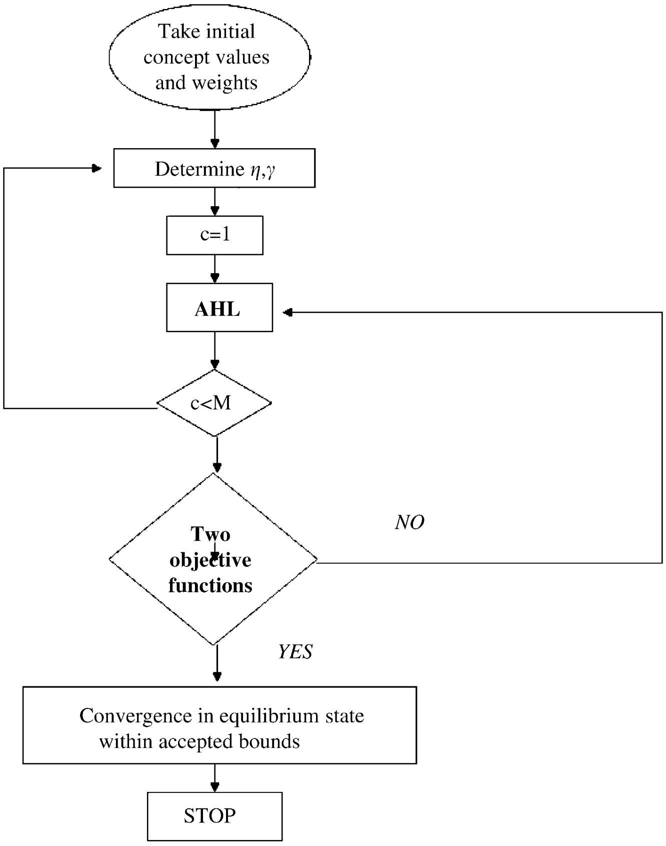
Take initial concept values and weights
Determine η,γ
c=1
AHL
c<M
Two objective functions
NO
YES
Convergence in equilibrium state within accepted bounds
STOP

Fig. 4. Flowchart of the training process using the AHL technique.

The NHL algorithm is based on the premise that all the concepts in FCM model are synchronously triggering at each iteration step and change their values synchronously. During this triggering process all weights w_{ji} of the causal interconnections of the concepts are updated and the modified weight $w_{ji}^{(k)}$ are derived for iteration step k.

The value $A_i^{(k+1)}$ of concept C_i at iteration step $k+1$ is calculated, computing the influence of interconnected concepts with values A_j to the specific concept C_j due to modified weights $w_{ji}^{(k)}$ at iteration step k, through the following equation:

$$A_i^{(k+1)} = f\left(A_i^{(k)} + \sum_{\substack{j \neq i \\ j=1}}^{N} A_j^{(k)} \cdot w_{ji}^{(k)}\right). \tag{13}$$

Taking the advantage of the general nonlinear Hebbian-type learning rule for neural networks (Oja et al., 1991; Xu, 1994; Hassoun, 1995), we introduce the mathematical formalism incorporating this learning rule for FCMs. This algorithm relates the values of concepts and values of weights in the FCM model, and it may take the general mathematical form:

$$\Delta w_{ji} = \eta A_i^{(k-1)}(A_j^{(k-1)} - w_{ji}^{(k-1)} A_i^{(k-1)}), \tag{14}$$

where the coefficient η is a very small positive scalar factor called learning parameter and is determined using experimental trial and error method in order to optimize the final solution.

Eq. (14) is modified and adjusted for FCMs and the following form of the nonlinear weight-learning rule for FCMs is proposed:

$$w_{ji}^{(k)} = \gamma \cdot w_{ji}^{(k-1)} + \eta A_i^{(k-1)}(A_j^{(k-1)} - sgn(w_{ji}^{(k-1)}) w_{ji}^{(k-1)} A_i^{(k-1)}), \tag{15}$$

where the γ is the weight decay learning coefficient.

The value of each concept of FCM is updated, through Eq. (13) where the value of weight $w_{ji}^{(k)}$ is calculated using Eq. (15).

Indeed, when experts develop a FCM they usually propose a quite sparse weight matrix **W**. Using the NHL algorithm the initially non-zero weights are updating synchronously at each iteration step through Eq. (15), until the termination of the algorithm. The NHL algorithm does not assign new interconnections and all the zero weights do not change value. When the algorithm termination conditions are met, the final weight matrix $\mathbf{W}_{\text{NHL}}$ is derived.

Implementation of NHL algorithm requires determination of upper and lower bounds for the learning parameter η; using trial and error experiments the values of learning rate parameter η was determined to belong in $0<\eta<0.1$. For any specific case-study problem a constant value for η is calculated (Papageorgiou et al., 2003b).

4.2.1. Two termination conditions

During the FCM development stage, experts define the Desired Output Concepts (DOCs). These concepts stand for the main characteristics and outputs of the system that we want to estimate their values, which reflect the overall state of the system. The distinction of FCM concepts as inputs and outputs is determined by the group of experts for each specific problem. Experts select the output concepts and they consider the rest as initial stimulators or interior concepts of the system. The proposed learning algorithm extracts hidden and valuable knowledge of experts and it can increase the effectiveness of FCMs and their implementation in real problems.

Two complementary termination conditions of the NHL process have been proposed: The first termination condition is the minimization of the following cost function F_1:

$$F_1 = \sqrt{\left\|DOC_j^{(k)} - T_j\right\|^2}, \tag{16}$$

where T_j is the mean target value of the concept DOC_j. At each step, the value of F_1 calculates the square of the Euclidean distance of actual DOC_j value and mean target value T_j of the DOC_j values.

Let us assume that we want to calculate the cost function F_1 of concept C_j. It is required that DOC_j take values in the range $DOC_j = [T_j^{\min}, T_j^{\max}]$. Then the target value T_j of the concept C_j is determined as

$$T_j = \frac{T_j^{\min} + T_j^{\max}}{2} \tag{17}$$

If we consider the case of an FCM-model, where there are m DOCs, then for the calculation of F_1, we take the sum of the square differences between the m-DOCs values and the m-T's mean values of DOCs, and Eq. (16) takes the following form:

$$F_1 = \sqrt{\sum_{j=1}^{m}(DOC_j^{(k)} - T_j)^2} \tag{18}$$

The objective of the training process is to find the set of weights that minimize function F_1.

In addition to the previous statements, one more criterion for the NHL has been introduced so as to terminate the algorithm after a limited number of steps. This second criterion is based on the variation of the subsequent values of DOC_j concepts, for iteration step k, yielding a very small value e, taking the form:

$$F_2 = |DOC_j^{(k+1)} - DOC_j^{(k)}| < 0.002, \tag{19}$$

where $DOC_j^{(k)}$ is the value of jth concept at iteration step k.

The constant value $e = 0.002$ has been proposed after a large number of simulations for different FCM cases. When the variation of two subsequent values of DOC_j is less than this number then it is pointless for the system operation to continue the training process.

```
Algorithm: "Nonlinear Hebbian Learning"

Step 1: Read input concept state $\mathbf{A}^0$ and initial weight matrix $\mathbf{W}^0$
Step 2: For iteration step k.
Step 3:     Update the weights:
    $w_{ij}^{(k)} =+ w_{ij}^{(k-1)} + \eta A_j^{(k-1)}(A_j^{(k-1)} - sgn\,(w_{ij})\, w_{ij}^{(k-1)} A_j^{(k-1)})$
Step 4:     Calculate $A_j^{(k)}$ according to the eq. (13)
Step 5:     Calculate the two termination functions
Step 6: Until both the terminati on conditions are met, go to step 2
Step 7: Return the final weights $\mathbf{W}_{NHL}$.
```

Fig. 5. NHL algorithm for Fuzzy Cognitive Maps.

When both terminations functions F_1 and F_2 are satisfied, the learning algorithm terminates and the desired equilibrium region for the DOCs is reached.

A generic description of the proposed NHL Algorithm for FCMs is given in Fig. 5.

A series of steps are performed to simulate a case-study scenario, implementing the proposed learning method. These steps are described in the following section.

4.2.2. Simulation steps of the NHL-based training technique

Let us assume that concept C_i was defined as DOC_i and it has the initial value A_i. We want to train FCM using NHL so that the DOC_i to reach a target value T_i. The series of steps to simulate a hypothetical scenario are as follows:

Step 1: Using the initial weight matrix and following the FCM model implications in Eq. (1) we calculate the value A_i.

Step 2: The proposed bounds for DOC_i are set.

Step 3: IF the calculated value A_i is within the accepted bounds THEN the process STOPs and the initial weight matrix is sufficient and it is not requiring training. IF the calculated value for DOC_i is non-accepted THEN GO TO next step.

Step 4: The NHL algorithm is applied and updated the weights until the DOC_i converges in the target value T_i.

Step 5: When the NHL algorithm has been applied for over 100 times AND both the termination conditions are not met THEN experts are asked to reconstruct the FCM model. The new weight matrix $\mathbf{W}_{\text{new}}^{\text{initial}}$ of the reconstructed FCM is used in Eq. (1) and calculates the A_i again, GO TO step 3.

Step 6: IF DOC_i does not reach an accepted value THEN GO TO step 4, determine the parameter η and implement the NHL technique again. ELSE the process STOPs and the updated weight matrix is appropriate for the case-study scenario.

The flowchart in Fig. 6 describes the NHL-based algorithmic procedure.

5. Illustrative comparison example—an industrial process control problem

In this section the two proposed unsupervised learning algorithms, the AHL and the NHL are implemented to train the FCM that models a simple process control problem encountered in chemical industry.

A simple process example is considered where there is one tank and three valves that influence the amount of liquid in the tank; Fig. 7 shows an illustration of the system. Valve 1 and Valve 2 empty two different kinds of liquid into Tank 1 and during the mixing of the two liquids a chemical reaction takes place in the tank. A sensor is located inside the tank that measures the specific gravity of the produced liquid in the tank. When the value of the specific gravity is in the range between ($G_{\max}$) and ($G_{\min}$), this means that the desired liquid has been produced in the tank. Moreover, there is a limit on the height of liquid into tank, which cannot exceed an upper limit ($H_{\max}$) and a lower limit ($H_{\min}$). The control target is to keep these variables in the following range of values:

$$G_{\min} \leqslant G \leqslant G_{\max},$$
$$H_{\min} \leqslant H \leqslant H_{\max}.$$

A FCM that models and controls this system is developed and depicted in Fig. 8. Three experts constructed the FCM. They jointly determined the concepts of the FCM and then each expert drawn the interconnections among concepts and assigned fuzzy weight for each interconnection (Stylios and Groumpos, 2000). The ADCs in this problem, as experts proposed them, are the concept C_1 representing the height of liquid and C_5 representing the specific gravity of the produced liquid into the tank.

The FCM model for this process control problem consists of five concepts:

Concept 1—the amount of the liquid that Tank 1 contains, it depends on the operational state of Valves 1, 2 and 3.

Concept 2—the state of Valve 1 (it may be closed, open or partially opened).

Concept 3—the state of Valve 2 (it may be closed, open or partially opened).

Concept 4—the state of Valve 3 (it may be closed, open or partially opened).

Concept 5—the specific gravity of the liquid into the tank.

For this specific chemical control problem, experts asked to describe the behaviour of the system as well as to infer the sequence of activation. They explained that they usually monitor the height of liquid in tank (C_1) and according to this value, the operator of the system close Valve 1 (C_2) and

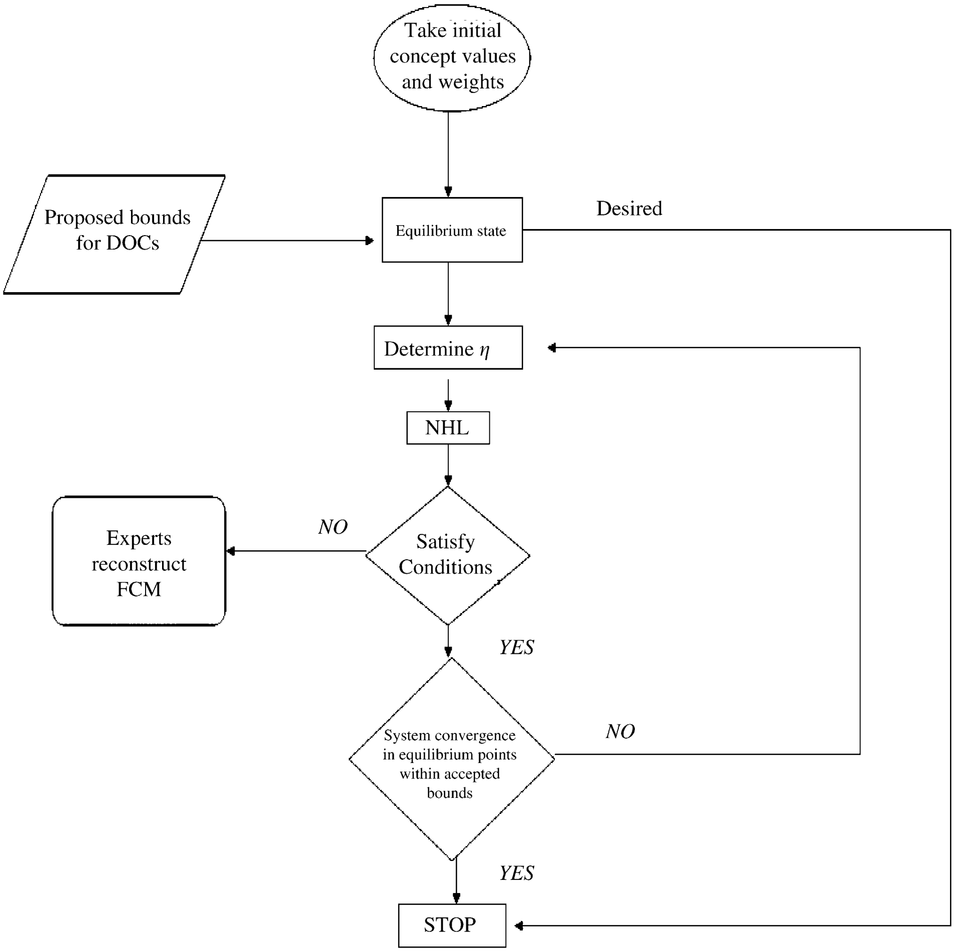

Fig. 6. Flowchart of the training process using the NHL technique.

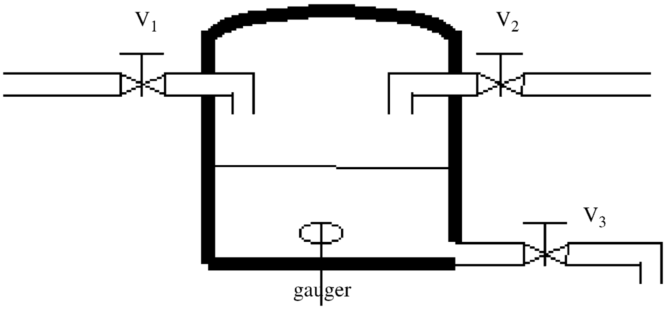

Fig. 7. The illustration for the process control example.

Valve 2 (C_3). Then, experts monitor the value of specific gravity of produced liquid (C_5) and according to this value open Valve 3 (C_4). Thus, concept C_1 is defined as the first Activated concept. Concepts C_2 and C_3 are the synchronously Activated concepts, at next sub-step, that are the second Activation concepts. Concept C_4 is the third Activated concept and concept C_5 is the fourth Activated concept. Then, the *c*-cycle consists of 4 sub-steps.

The sign and the weight of each interconnection have been determined by three experts, using the methodology described in Section 2.1. All the experts agreed on the direction of the interconnections between the concepts and then every one of the experts proposed a linguistic variable for each weight. These linguistic values for each one of the

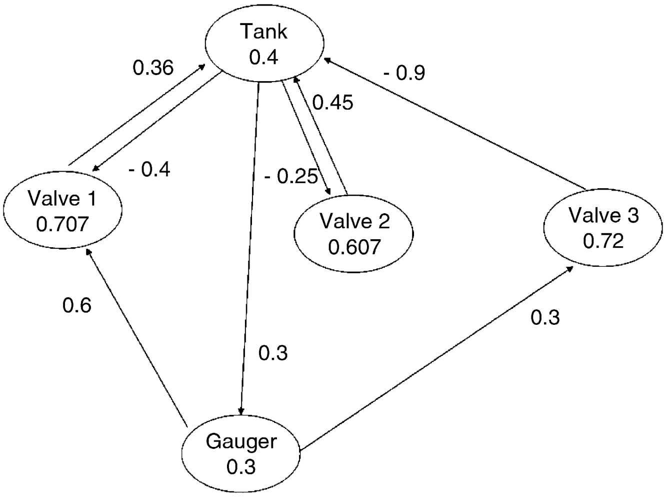

Fig. 8. The FCM model of the practical process problem.

corresponding weights are the following:

$$\begin{aligned}
&w_{12} \mapsto \textit{negatively} - \textit{weak}(\mu_{\mathrm{nw}})\\
&w_{13} \mapsto \textit{negatively} - \textit{weak}(\mu_{\mathrm{nw}})\\
&w_{21} \mapsto \textit{positively} - \textit{weak}(\mu_{\mathrm{pw}})\\
&w_{15} \mapsto \textit{positively} - \textit{weak}(\mu_{\mathrm{pw}})\\
&w_{31} \mapsto \textit{positively} - \textit{medium}(\mu_{\mathrm{pm}})\\
&w_{41} \mapsto \textit{negatively} - \textit{small}(\mu_{\mathrm{ns}})\\
&w_{52} \mapsto \textit{positively} - \textit{medium}(\mu_{\mathrm{pm}})\\
&w_{54} \mapsto \textit{positively} - \textit{medium}(\mu_{\mathrm{pm}})
\end{aligned}$$

The fuzzy linguistic weights proposed for each interconnection are aggregated and then defuzzified through the COG defuzzification so that the crisp weights values are produced:

$$\begin{aligned}
&-0.5 \leqslant w_{12} \leqslant 0\\
&-0.5 \leqslant w_{13} \leqslant 0\\
&0 \leqslant w_{21} \leqslant 0.5\\
&0 \leqslant w_{15} \leqslant 0.5\\
&0.25 \leqslant w_{31} \leqslant 0.75\\
&-1 \leqslant w_{41} \leqslant -0.5\\
&0.25 \leqslant w_{52} \leqslant 0.75\\
&0.25 \leqslant w_{54} \leqslant 0.75
\end{aligned} \tag{20}$$

Thus, the weight matrix of the FCM model is the following:

$$\mathbf{W}^{\mathrm{initial}} = \begin{bmatrix} 0 & -0.4 & -0.25 & 0 & 0.3 \\ 0.36 & 0 & 0 & 0 & 0 \\ 0.45 & 0 & 0 & 0 & 0 \\ -0.9 & 0 & 0 & 0 & 0 \\ 0 & 0.6 & 0 & 0.3 & 0 \end{bmatrix}. \tag{21}$$

Experts also determined the desired regions for the output concepts, which reflect the proper operation of the modelled system:

$$\begin{aligned}
&0.68 \leqslant A_1 \leqslant 0.74,\\
&0.74 \leqslant A_5 \leqslant 0.80.
\end{aligned} \tag{22}$$

For this FCM model of the process problem we will examine two scenarios. The first one supposes that all the values of concepts are changed synchronously in the same time units and this is referred to as an iteration step, so the NHL algorithm can be implemented. The second scenario supposes that the values of concepts are changed asynchronously through the sequence of activation, so that the AHL algorithm is applied.

6. Experimental results

First, before the implementation and testing of the proposed learning algorithms, for the chemical process control problem described in Section 5, we apply the typical Eq. (1) to find the equilibrium final state after the FCM modelling interactions. The initial values of concepts given in matrix $\mathbf{A}^0 = [0.4\,0.7\,0.6\,0.7\,0.3]$ represent the measured data of the physical process (after thresholding). These values and the initial weight matrix $\mathbf{W}^{\mathrm{initial}}$ of Eq. (21) are used in Eq. (1) to calculate the equilibrium region of the process when no learning algorithm is applied. After 9 iteration steps, the FCM concept values do not change, so the equilibrium region is reached (Table 1 gives the subsequent values of calculated concepts). It is observed that the values of concepts C_1 and C_5 are out of the suggested desired regions by Eq. (22).

Thus, learning algorithms are required to train the FCM so as to converge in the desired regions. The AHL and NHL algorithms can be applied to fine-tuning FCM causal links by modifying the weight values. Two different training scenarios will be considered. In the first scenario, experts either propose the initial values of concepts or concepts' values are derived from real measurement data of the physical process. Thus, two updating weight matrices are derived when the proposed learning techniques are implemented for the asynchronous triggering mode (applying AHL) and synchronous triggering mode (applying NHL). In the second scenario, the AHL and NHL algorithms are implemented for an initial set of random concepts values, calculating new values for ADCs and DOCs and examining the convergence of the FCM model.

6.1. First case-study

This case-study scenario examines the implementation of the two proposed algorithms for the same initial set of concepts values and weights of the FCM. This scenario concerns the weight adaptation of the FCM model so as to converge in the desired region, determining the appropriate values of weights among concepts (weights).

6.1.1. Implementation of the AHL technique

The AHL algorithm is applied to modify the weights and control the process. The AHL steps were presented at Section 4.1; at first step, it takes the initial vector $\mathbf{A}^0$ and the initial weight matrix $\mathbf{W}^{\mathrm{initial}}$.

At second learning step, the learning parameters η, γ have been determined for the specific process control

Table 1
The values of concepts at each step of FCM interaction

Steps	Tank 1	Valve 1	Valve 2	Valve 3	Cauger
1	0.40	0.7077	0.612	0.717	0.30
2	0.5701	0.6743	0.6253	0.6921	0.6035
3	0.6157	0.6918	0.6184	0.7054	0.6845
4	0.6244	0.7019	0.6141	0.7132	0.7046
5	0.6252	0.7058	0.6125	0.7160	0.7093
6	0.6249	0.7071	0.6121	0.7168	0.7103
7	0.6248	0.7075	0.6120	0.7171	0.7105
8	0.6247	0.7076	0.6120	0.7171	0.7105
9	0.6247	0.7077	0.6120	0.7171	0.7105

problem after experimental trials and the followings have been suggested:

$$\eta^{(c)} = 0.02\ \exp(-0.1 \cdot c), \tag{23}$$

$$\gamma^{(c)} = 0.04\ \exp(-c). \tag{24}$$

At third learning step (which is the first sub-step) and for iteration number $k = 1$, as described in the AHL procedure (Fig. 4), the concept C_1 is defined as the first Activated concept, triggered by the corresponding interconnected concepts C_2, C_3, C_4, which behave as Activation concepts, using their previous values of $\mathbf{A}^0_{\text{first}} = [A_1^{(0)} A_2^{(0)} A_3^{(0)} A_4^{(0)} A_5^{(0)}]$. The value of Activation concept $A_1^{(1)}$, at iteration step $k = 1$, is calculated by Eq. (4), where the updated weight values are calculated by Eq. (8). At second sub-step, the concept C_1, with its new value $A_1^{(1)}$, triggers the other interrelated concepts and becomes the Activation concept. The concepts C_2 and C_3 are now the second Activated concepts, affected by the Activation concept C_1, and their activation values $A_2^{(1)}$ and $A_3^{(1)}$, for iteration number $k = 2$, are also calculated by Eq. (4). At the third sub-step the previously activated concepts affect the concept C_5, which now is the Activated concept with value $A_5^{(1)}$, for $k = 3$. All the previously Activated concepts C_1, C_2, C_3, C_5, with their calculated values $A_1^{(1)}$, $A_2^{(1)}$, $A_3^{(1)}$, $A_5^{(1)}$, respectively, act as Activation concepts, triggering the concept C_4. The C_4 as the fourth Activated concept, takes the value $A_4^{(1)}$, using Eq. (4) for iteration $k = 4$. Notably, those only the weights connect the Activation Concepts to the Activated concepts are updated at each iteration step using Eq. (8). All other weights remain unchanged at each iteration sub-step. When a cycle closes, then all the weights become updated.

The activation process implementing the AHL algorithm stops when the two proposed criteria, Eqs. (11) and (12) are satisfied, for this example it happens after 18 cycles.

When the AHL algorithm terminates and the output concepts values are within the desired regions, which mean that the values of decision-output concepts are accepted, a new updated weight matrix is derived, determining new cause–effect relationships between the concepts of FCM model. The equilibrium region for this scenario is reached, after 18 recursive cycles, and here is the equilibrium concept values $\mathbf{A}^1_{\text{AHL}} = [0.7270\ \ 0.7708\ \ 0.7065\ \ 0.7807\ \ 0.7771]$. Fig. 9 shows the values of concepts for 18 activation cycles after implementing the AHL algorithm.

The updated weight matrix after the 18 activation cycles is the following:

$$\mathbf{W}^1_{\text{AHL}} = \begin{bmatrix} 0 & -0.1822 & -0.0855 & 0.1055 & 0.316 \\ 0.3528 & 0 & 0.101 & 0.115 & 0.110 \\ 0.4134 & 0.102 & 0 & 0.105 & 0.100 \\ -0.5038 & 0.114 & 0.102 & 0 & 0.111 \\ 0.1052 & 0.532 & 0.098 & 0.322 & 0 \end{bmatrix}. \tag{25}$$

A new FCM model for the chemical process has been produced with this updated weight matrix. It is noticeable that the initial zero weights no longer exist and new interconnections have been assigned. This means that all concepts affect the related concepts, and the weighed arcs show the degree of this relation. If we examine some of the new assigned weights, we see that a weight interconnection between concepts C_1 and C_4 has been assigned—meaning that a new influence among these concepts exists. This interconnection means that the concept C_1 which represents the height of the tank influences positively low the concept C_4 which represents the "Valve 3". This is a reasonable interconnection, with engineering meaning. In the same way the following interconnections have been added between concepts C_2 and C_3 (weight w_{23}), C_2 and C_4 (weight w_{24}), C_2 and C_5, C_3 and C_2, C_3 and C_4, C_3 and C_5, C_4 and C_2, C_4 and C_3, C_4 and C_5, C_5 and C_1, and between C_5 and C_3.

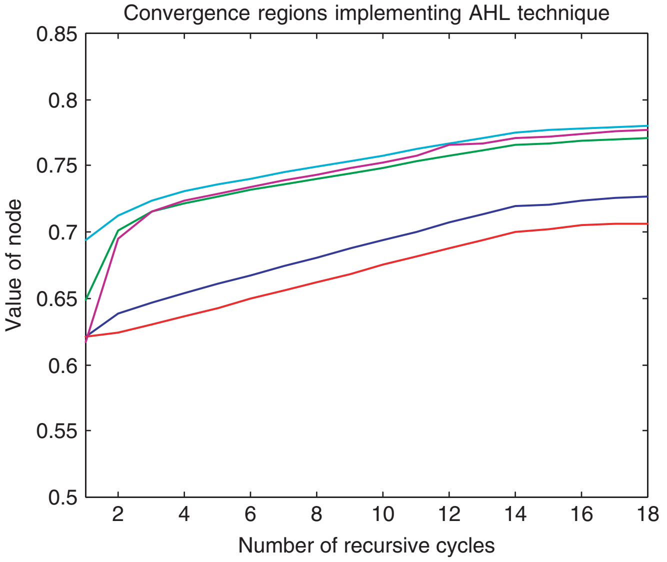

Fig. 9. Variation of concepts values implementing the AHL algorithm.

In addition to this the AHL algorithm modified some of the initially proposed cause–effect relationships. As an example the influence of concept C_3 towards C_1 (initial value -0.4) has been modified after 18 cycles taking the value -0.4134, which is a very small decrease. In the same way, the weighted arc $w_{12}$with initial value -0.4, after 18 cycles, takes the value -0.1822, which means that the initial influence from C_1 towards C_2 decreases negatively at a small amount. The initial zero value of weights w_{23} has changed and after 18 cycles is 0.101, which means that the concept C_2 affects the concept C_3. The same happens with all the other cause–effect relationships among concepts. Thus, this training affects the dynamical behaviour of the system.

Then, we tested this new produced weight matrix for 1000 different cases using random initial values. For all these cases the results for the ADCs where within the intervals:

$$\begin{aligned} &0.68 \leqslant ADC_1 \leqslant 0.74, \\ &0.74 \leqslant ADC_5 \leqslant 0.79. \end{aligned} \tag{26}$$

It may be observed that the two ADCs, the ADC_1 and ADC_5, take values in the desired regions suggested in Eq. (22).

Therefore, it is proved that using the AHL algorithm we have improved the FCM model, which exhibits equilibrium behaviour within the desired regions. With the proposed procedure the experts suggest the initial weights of the FCM and the sequence of activation, and then using the AHL algorithm a new weight matrix is derived which can be used for any set of initial values of concepts.

6.1.2. Implementation of the NHL algorithm

Here, it is the second phase of the first scenario where the NHL algorithm is applied to train the FCM and control the process. To implement the NHL steps given in Section 4.2.2 are followed. The training process starts by applying the initial values of concepts $\mathbf{A}^0$ and the weight matrix $\mathbf{W}^{\text{initial}}$. Notably, only the initially non-zero weights that connect the triggering concepts are updated at each simulation step using Eq. (15). All other weights remain zero.

For this specific control problem, the suggested value of learning rate parameter η is 0.04, which was defined after trial and error experiments, so as to fulfill the requirements of the specific system. If the learning parameter η, takes any other value than the proposed one, then the FCM will converge in undesired regions.

The proposed NHL procedure continues until the two termination criteria are met, which in these experimental tests is achieved after 17 iteration steps. The result of training the FCM is a new set of connection weights at the weight matrix $\mathbf{W}^1_{\text{NHL}}$:

$$\mathbf{W}^1_{\text{NHL}} = \begin{bmatrix} 0 & -0.1736 & -0.0265 & 0 & 0.479 \\ 0.5103 & 0 & 0 & 0 & 0 \\ 0.5753 & 0 & 0 & 0 & 0 \\ -0.90 & 0 & 0 & 0 & 0 \\ 0 & 0.707 & 0 & 0.493 & 0 \end{bmatrix}. \tag{27}$$

This updated weight matrix leads the FCM in the equilibrium region given in: $\mathbf{A}^1_{\text{NHL}} = [0.6830\ 0.7632\ 0.6538\ 0.7534\ 0.7440]$.

We examine the influence of NHL algorithm by comparing $\mathbf{W}^{\text{initial}}$ and the new produced matrix $\mathbf{W}^1_{\text{NHL}}$. The weight w_{15} (initial value equal to 0.3) takes the value 0.479 after training which mean that the influence of the height of tank (concept C_1) to the Gauger (concept C_5) increases positively at an important large amount. Also the weight w_{13} (initial value equal to -0.4) takes the value -0.0265 which means that the influence of the height of tank (concept C_1) to the valve 2 (concept C_3) decreases at a large amount. In practice this means that the influence is almost zero. Some of the weights have changed significantly their values from the initial ones in order to obtain the desired output values of concepts. More specifically, the weights w_{31}, w_{52} and w_{47}, have updated their values at a significant amount to satisfy the constraints for *DOCs*.

Fig. 10 represents the variation of values of concepts after 17 simulation steps implementing the NHL algorithm. We observe that the values of *DOCs* in the equilibrium state are within the desired regions.

Thus the weight adaptation method adjusts the FCM cause–effect relationships and controls the system's output concepts in accepted-desired regions.

We evaluate the derived weight matrix $\mathbf{W}^1_{\text{NHL}}$ using a set of random values (random vector $\mathbf{A}^0 = [0.27\,0.01\,0.08\,0.63\,0.23]$). The calculated values of concepts in equilibrium point are given in $\mathbf{A}_{\text{equil}} = [0.6837\,0.7632\,0.6538\,0.7543\,0.7451]$. Fig. 11 represents variation of concepts values starting with random initial concepts values. We have tested this FCM model for 1000 test cases with

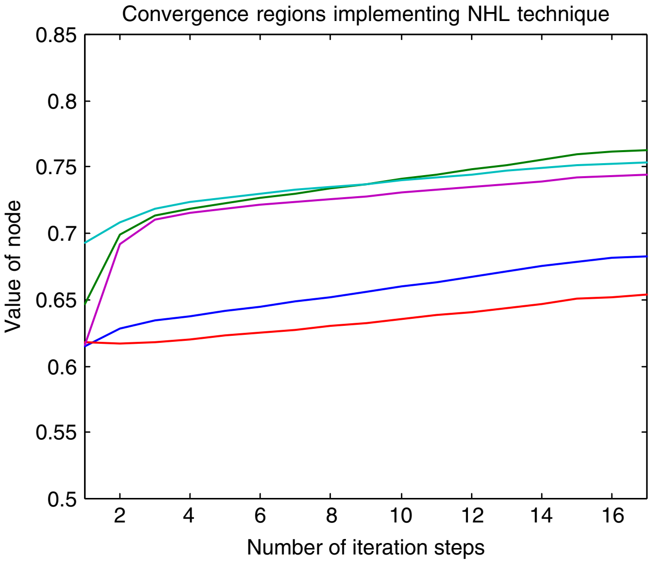

Fig. 10. Variation of concepts values implementing the NHL algorithm.

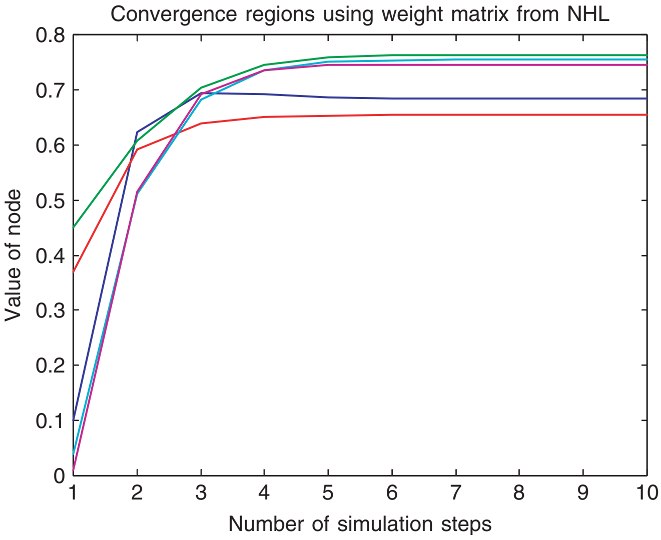

Fig. 11. Equilibrium state for concepts using the weight matrix $\mathbf{W}^1_{\text{NHL}}$.

random initial values of concepts, using the weight matrix $\mathbf{W}_{NHL}^{1}$ and we come up with the same result for concepts values.

6.2. Second case-study

6.2.1. Implementation of the AHL algorithm

In this scenario, initial random values of concepts are suggested to adapt the weights of the FCM model. Let us consider a set of initial random values $\mathbf{A}^0 = [0.13\ \ 0.405\ \ 0.20\ \ 0.85\ \ 0.04]$. Here the following values of $\eta^{(c)} = 0.02\ \exp(-0.1 \cdot c)$ and $\gamma^{(c)} = 0.04\ \exp(-c)$ are used which were used in Section 6.1. The AHL algorithm is applied following the proposed procedure in the Section 4.1 as it was implemented in the first scenario.

Implementing the AHL algorithm the FCM concepts converge after 17 recursive cycles and the following weight matrix is derived:

$$\mathbf{W}_{\mathrm{AHL}}^{2} = \begin{bmatrix} 0 & -0.1695 & -0.076 & 0.106 & 0.315 \\ 0.3520 & 0 & 0.0839 & 0.115 & 0.114 \\ 0.4116 & 0.106 & 0 & 0.104 & 0.105 \\ -0.4953 & 0.114 & 0.0830 & 0 & 0.115 \\ 0.1101 & 0.5275 & 0.102 & 0.320 & 0 \end{bmatrix}. \qquad (28)$$

The interpretation of the obtained cause–effect relationships among concepts is similar to the first case study. The equilibrium values of concepts are given in matrix $\mathbf{A}_{\mathrm{AHL}}^{2} = [0.7283\ \ 0.7724\ \ 0.7101\ \ 0.7793\ \ 0.7779]$. It is observed that the values for all *ADCs* are within the accepted bounds. The same results are taken using any other random set of initial values (It was tested for 1000 different cases using initial random values).

6.2.2. Implementation of the NHL algorithm

In the second phase of the second test case scenario, the NHL algorithm is implemented for the same initial values of concepts as above. The first set of initial random values is $\mathbf{A}^0 = [0.13\ 0.405\ 0.20\ 0.85\ 0.04]$. The following value for the learning parameter $\eta = 0.04$ is used as in the first scenario.

Implementing the NHL algorithm the system's concepts converge after 15 iteration steps and the following weight matrix is derived:

$$\mathbf{W}_{\mathrm{NHL}}^{2} = \begin{bmatrix} 0 & -0.177 & -0.0246 & 0 & 0.472 \\ 0.509 & 0 & 0 & 0 & 0 \\ 0.5751 & 0 & 0 & 0 & 0 \\ -0.877 & 0 & 0 & 0 & 0 \\ 0 & 0.702 & 0 & 0.481 & 0 \end{bmatrix}. \qquad (29)$$

The values of concepts at converge region are given in $\mathbf{A}_{\mathrm{NHL}}^{2} = [0.6885\ 0.7619\ 0.65418\ 0.7523\ 0.7446]$. It is clearly shown that the values of *DOCs* in the equilibrium state are within the desired regions (accepted bounds). Similar results are derived using any other random set of initial values and running the algorithm for 1000 different cases.

The weight values have been modified according to the system's characteristics retaining the meaning of the results. Some of the weights have changed significantly their values from the initial ones to obtain the desired values for output concepts. The weights w_{21}, w_{13}, w_{15} and w_{54}, have also updated their values at a significant amount to satisfy the accepted bounds for *DOCs*. These new values for weights after training describe new cause–effect relationships between the concepts of FCM. Thus the weight adaptation method adjusts the FCM cause–effect relationships and controls the system convergence in accepted regions.

Both the proposed AHL and NHL algorithms have been implemented and tested successfully to a process control problem, adapting the cause–effect relationships between concepts of FCM and eliminating some of deficiencies that appear in operation of FCMs. Through the iteration training processes, the weights keep their signs and direction and their values, after redefinition of cause–effect relationships, are within the initially determined weight ranges in Eq. (20). In this way, we take into account the dynamic characteristics of the learning process and the environment and finally, help the output concept values to converge in desired bounds.

7. Comparison and discussion of the two unsupervised learning techniques

The main difficulty of proposing learning algorithms for FCMs is that the FCM allows feedback. The first proposed unsupervised learning algorithms for FCMs, the DHL algorithm (Kosko, 1988) and the Balanced Differential Hebbian Learning (BDHL) algorithm (Huerga, 2002) were applied only to FCMs with binary concepts values which significantly restrict their application areas and can not be used in practical problems. Some other learning approaches are based on conventional learning methods and they introduce the broken of the feedback causal links of the FCM. It is clear that there is a limitation of transplanting existing learning methods from Neural Network domain into FCMs. The AHL and NHL algorithms have been proposed for fine tuning FCM causal links increasing the effectiveness and reliability and eliminating the main drawbacks of FCMs.

In this direction, we proposed two suitable learning methods for adjusting FCMs weights, depending on each specific problem's characteristics and constraints. The AHL algorithm introduce the asynchronous adaptation mode for weights, it requires the definition of Activation sequence and introduces the distinction of Activated and Activation concepts. The AHL adds causal links between all the concepts so as to succeed the desired behaviour of the system, and not only modify the initial causal links. In this manner, all the weights are updated at the end of learning cycle and new causal interconnections are assigned. Moreover, two criteria have been proposed for the termination of the algorithm and the convergence of the output concepts within desired values.

The second algorithm, the NHL suppose synchronous adaptation mode, all the non-zero weights of FCM are adjusted at the same learning step. These weights are updated synchronously at each iteration step through Eq. (15), until the satisfaction of the two proposed error function that terminates the algorithm. In the case of NHL, not new interconnections are assigned and the entire initial zero weights remain.

The AHL and NHL algorithms are problem-dependent and they use the initial weight matrix. However, both the processes are independent from the initial values for concepts and the system's output concepts manage to converge in desired equilibrium points for appropriate learning parameters. Table 2 gathers the main characteristics and differences between the two learning techniques.

Here, we compare the two proposed learning algorithms with the swarm-based learning approach, that is the PSO algorithm, because the PSO algorithm has been proposed and implemented for the same practical process control problem (Papageorgiou and Groumpos, 2004). The PSO algorithm has the advantage of being independent of the initial concepts values. It takes into consideration all the initial knowledge suggested by human experts and not only the initial elements of the weight matrix, and for every scenario PSO determines appropriate ranges for the non-zero weights in order to satisfy the requirements. For the same process problem, PSO algorithm was implemented for two different scenarios and a large number of appropriate weight matrices were derived. All the derived weight matrices drove the output concepts to converge in desired regions. The algorithm was applied in a search space that is restricted to certain FCM concepts values and imposes constrains on the connection matrix, all of which are specified by domain expert(s). Experimental results have shown that PSO algorithm can determine optimum ranges for weight values for specific problems in real domain, in order to drive the output concepts within desired regions.

Here we compare the proposed learning algorithms with the genetic-based learning algorithms for FCMs. A recently proposed genetic algorithm-based learning FCM approach (Khan et al., 2004) did not aim to compute the weight matrix, but rather to find initial state vector and its usefulness is restricted in real problems' domain. The RCGA aims to develop FCM weight matrix (Stach et al., 2005) and it has been implemented for four different FCM models with increasing number of nodes and the results have been promising. The large number of tests which involve ten cross-validation experiments for FCMs of varying sizes and densities show the first trial for optimization of FCM weight matrix, but this method needs investigation in terms of the sequence of input data used and the large number of weight matrices produced as well as its convergence and the determination of its GA parameters.

Here, it was presented and compared the two proposed unsupervised weight-learning techniques for FCMs. It is proved that using the AHL and NHL algorithms we improve the FCM model and the output concept values can converge to desired values. With the proposed procedures the experts suggest the initial weights of the FCM, and then using these algorithms new weight matrices are derived, respectively, that can be used for any set of initial values of concepts.

8. Conclusions

In this paper, two unsupervised weight adaptation techniques, namely AHL and NHL have been introduced to fine-tune FCM causal links. These algorithms accompany the good knowledge of a given system or process can contribute towards the establishment of FCMs as a robust technique. They update the initial information and experts' knowledge so that to keep the values of output concepts of FCM model within desired bounds.

The two proposed unsupervised learning techniques sustain the following advantageous features:

- Strengthen the dynamical behaviour of the FCM model.
- Provide the FCM developers with learning parameters to adjust the influence of concepts and drive the system in convergence.
- Enhances the FCM adaptability and functionality and effectiveness.
- Improves the functional FCM reliability.

Table 2
The main differences between the AHL and NHL algorithms

	AHL	NHL
1	Asynchronous sequence of Activation concepts	Synchronous triggering and interaction of all concepts
2	Asynchronous updating of weights	Synchronous updating mode for weights
3	All weights are updated	Only the initially non-zero weights are updated
4	Arise new cause–effect relationships between concepts of FCM	No new cause–effect relationships between concepts
5	Introduction of activation cycle consisting of Activation steps	No cycle, just one step
6	Two criterion functions based on constraints for concepts	Two termination conditions for the algorithm
7	Exponential attenuation of learning parameters η, γ	Constant values for learning rate parameter, η after trial and error experiments

These types of learning rules accompanied with the good knowledge of the given system guarantee the successful implementation of the proposed processes. Moreover, these are suitable methodologies for practical applications as the obtained results can be interpreted and have direct relation to the successful system operation. They can also contribute towards the convergence of FCM's output concepts in desired regions.

The proposed learning techniques contribute towards the establishment of FCM as a robust technique, as they can efficiently update the cause–effect relationships among FCM concepts and their effectiveness in real problems have been proved through a number of simulations for different case studies.

Our future work will concern the improvement of the proposed learning methods in terms of their computational complexity and convergence. Furthermore, future work will be directed towards the investigation of robust evolutionary FCM learning techniques.

Acknowledgement

This work was supported by the "PYTHAGORAS II" research grant co funded by the European Social Fund and National Resources.

References

Aguilar, J., 2002. Adaptive random fuzzy cognitive maps. In: Garijio, F.J., Riquelme, J.C., Toro, M. (Eds.), IBERAMIA 2002. Lecture Notes in Artificial Intelligence, vol. 2527. Springer, Berlin, Heidelberg, pp. 402–410.

Craiger, J.P., Goodman, D.F., Weiss, R.J., Butler, A., 1996. Modeling organizational behavior with fuzzy cognitive maps. Journal of Computational Intelligence and Organizations 1, 120–123.

Dickerson, J., Kosko, B., 1994. Fuzzy Virtual Worlds. AI Expert, pp. 25–31.

Groumpos, P., Stylios, C., 2000. Modeling supervisory control systems using fuzzy cognitive maps. Chaos, Solitons and Fractals 11, 329–336.

Hassoun, M., 1995. Fundamentals of Artificial Neural Networks. MIT Press, Bradford Book, MA.

Hebb, D.O., 1949. The Organization of Behaviour: A Neuropsychological Theory. Wiley, New York.

Huerga, A.V., 2002. A Balanced Differential Learning algorithm in Fuzzy Cognitive Maps. In: Proceedings of the 16th International Workshop on Qualitative Reasoning 2002, poster.

Jain, L., 1997. Soft Computing Techniques in Knowledge-Based Intelligent Engineering Systems: Approaches and Applications. Studies in Fuzziness and Soft Computing, vol. 10. Springer, Berlin.

Kang, I.I., Lee, S., Coi, J., 2004. Using fuzzy cognitive map for the relationship management in airline service. Expert Systems with Applications 26 (4), 545–555.

Khan, M.S., Khor, S., Chong, A., 2004. Fuzzy cognitive maps with genetic algorithm for goal-oriented decision support. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 12, 31–42.

Kosko, B., 1986. Fuzzy cognitive maps. International Journal of Man–Machine Studies 24, 65–75.

Kosko, B., 1988. Hidden patterns in combined and adaptive knowledge networks. International Journal of Approximate Reasoning 2, 377–393.

Kosko, B., 1992. Fuzzy associative memory systems. In: Kandel, A. (Ed.), Fuzzy Expert Systems. CRC Press, Boca Raton, FL, pp. 135–162.

Kosko, B., 1997. Fuzzy Engineering. Prentice-Hall, New Jersey.

Koulouriotis, D.E., Diakoulakis, I.E., Emiris, D.M., 2001. Learning Fuzzy Cognitive Maps using evolution strategies: A novel schema for modeling a simulating high-level behavior. In: Proceedings of the IEEE Congress on Evolutionary Computation, vol. 1, pp. 364–371.

Lee, K.C., Kin, J.S., Chung, N.H., Kwon, S.J., 2002. Fuzzy cognitive map approach to web-mining inference amplification. Journal of Experts Systems with Applications 22, 197–211.

Lin, C.T., Lee, C.S.G., 1996. Neural Fuzzy Systems: A Neuro-Fuzzy Synergism to Intelligent Systems. Prentice-Hall, Upper Saddle River, NJ.

Liu, Z.Q., 2000. Fuzzy Cognitive Maps: Analysis and Extension. Springer, Tokyo.

Liu, Z.Q., Satur, R., 1999. Contextual fuzzy cognitive map for decision support in geographical information systems. Journal of IEEE Transaction on Fuzzy Systems 7, 495–507.

Muata, K., Bryson, O., 2004. Generating consistent subjective estimates of the magnitudes of causal relationships in fuzzy cognitive maps. Computers and Operations Research 31 (8), 1165–1175.

Oja, E., 1989. Neural networks, principal components and subspaces. International Journal of Neural Systems 1, 61–68.

Oja, E., Ogawa, H., Wangviwattana, J., 1991. Learning in nonlinear constrained Hebbian networks. In: Kohonen, T., et al. (Eds.), Artificial Neural Networks. North-Holland, Amsterdam, pp. 385–390.

Papageorgiou, E.I., Groumpos, P.P., 2004. A weight adaptation method for Fuzzy Cognitive Maps to a process control problem. In: Budak, M., et al. (Eds.), Lecture Notes in Computer Science 3037, vol. II (International Conference on Computational Science, ICCS 2004, Krakow, Poland, 6–9 June). Springer Publications, Berlin, pp. 515–522.

Papageorgiou, E.I., Stylios, C.D., Groumpos, P.P., 2003a. An integrated two-level hierarchical decision making system based on Fuzzy Cognitive Maps (FCMs). IEEE Transactions on Biomedical Engineering 50 (12), 1326–1339.

Papageorgiou, E.I., Stylios, C.D., Spyridonos, P., Nikiforidis, G., Groumpos, P.P., 2003b. Urinary bladder tumor grading using nonlinear Hebbian learning for Fuzzy Cognitive Maps. Proc. of IEE Int. Conf. on Systems Engineering (ICSE 2003) 2, 542–547.

Papageorgiou, E.I., Stylios, C.D., Groumpos, P.P., 2003c. Fuzzy Cognitive Map learning based on nonlinear Hebbian rule. In: Gedeon, T.D., Fung, L.C.C. (Eds.), AI 2003, Lecture Notes in Artificial Intelligence 2903:254–266. Springer-Verlag, Berlin Heidelberg.

Papageorgiou, E.I., Parsopoulos, K.E., Groumpos, P.P., Vrahatis, M.N., 2003d. A first study of Fuzzy Cognitive Maps learning using particle swarm optimization. In: Proceedings of the IEEE 2003 Congress on Evolutionary Computation. IEEE Press, New York, pp. 1440–1447.

Papageorgiou, E.I., Stylios, C.D., Groumpos, P.P., 2004a. Active Hebbian Learning algorithm to train fuzzy cognitive maps. International Journal of Approximate Reasoning 37 (3), 219–247.

Papageorgiou E.I., Parsopoulos K.E., Groumpos P.P., Vrahatis M.N., 2004b. Fuzzy Cognitive Maps learning through swarm intelligence. In: Proceedings of the 16th International Conference on Artificial Intelligence and Soft Computing (ICAISC) 2004, Zakopane, Poland, 7–11 June, Lecture Notes in Artificial Intelligence, vol. 3070, Springer Verlag, Berlin, Heidelberg, pp. 344–349

Park, K.S., Kim, S.H., 1995. Fuzzy cognitive maps considering time relationships. International Journal of Human-Computer Studies 42, 157–168.

Pelaez, C.E., Bowles, J.B., 1996. Using fuzzy cognitive maps as a system model for failure modes and effects analysis. Information Sciences 88, 177–199.

Stach, W., Kurgan, L., Pedrycz, W., Reformat, M., 2005. Genetic learning of fuzzy cognitive maps. Fuzzy Sets and Systems 153 (3), 371–401.

Stylios, C.D., Groumpos, P.P., 2000. Fuzzy cognitive maps in modeling supervisory control systems. Journal of Intelligent & Fuzzy Systems 8, 83–98.

Stylios, C.D., Groumpos, P.P., 2004. Modeling complex systems using fuzzy cognitive maps. IEEE Transactions on Systems, Man & Cybernetics, Part A 34 (1), 155–162.

Stylios, C.D., Groumpos, P.P., Georgopoulos, V.C., 1999. An fuzzy cognitive maps approach to process control systems. Journal of Advanced Computational Intelligence 3 (5), 409–417.

Taber, R., 1991. Knowledge processing with fuzzy cognitive maps. Expert Systems with Applications 2, 83–87.

Xu, L., 1994. Theories for unsupervised learning: PCA and its nonlinear extensions. In: Proceedings of the IEEE International Conference on Neural Networks, vol. II, New York, pp. 1252–1257.

Xirogiannis, G., Stefanou, J., Glykas, M., 2004. A fuzzy cognitive map approach to support urban design. Expert Systems with Applications 26 (2), 257–268.

Zhang, W.R., Chen, S.S., Bezdek, J.C., 1989. Pool 2: a generic system for cognitive map development and decision analysis. IEEE Transactions on Systems, Man and Cybernetics 19, 31–39.

Zhang, W.R., Chen, S.S., Wang, W., King, R.S., 1992. A cognitive-map-based approach to the coordination of distributed cooperative agents. IEEE Transactions on Systems, Man and Cybernetics 22, 103–114.