A Generic Tool for Building Fuzzy Cognitive Map Systems

Maria Papaioannou, Costas Neocleous, Anastasis Sofokleous, Nicos Mateou,
Andreas Andreou, Christos N. Schizas

A Generic Tool for Building Fuzzy Cognitive Map Systems

Maria Papaioannou[1], Costas Neocleous[2], Anastasis Sofokleous[1],
Nicos Mateou[3], Andreas Andreou[1,2], Christos N. Schizas[1],

[1] 75 Kallipoleos Str, Department of Computer Science,
University of Cyprus, Nicosia, Cyprus
{cs03pm2, asofok, aandreou, schizas}@ucy.ac.cy
[2] 31 Archbishop Kyprianos Str, Cyprus University of Technology, Limassol, Cyprus,
{costas.neocleous, andreou}@cut.ac.cy
[3] Ministry of Defence, Cyprus, NMateou@mod.gov.cy

Abstract. A generic system for simulating complex dynamical systems along the paradigm of fuzzy cognitive maps (FCM) has been created and tested. The proposed system enables a user to design appropriate FCM structures, by specifying the desired concepts and the various parameters such as sensitivities, as well as a variety of shaping functions. The user is able to see the results, change the parameters, modify the functions, and rerun the system using an alteration of the final results and make new conclusions. The system is introduced and demonstrated using a simple real case. The results of a usability test of the system suggest that the system is capable of simulating complicated FCM structures in an effective manner, helping the user to reduce the degree of risks during decision making.

Keywords: Fuzzy Cognitive Maps, Soft Computing, Intelligent Systems.

1   Introduction and Background Theory

Fuzzy Cognitive Maps (FCMs) constitute a powerful soft computing modeling method that emerged from the combination of Fuzzy Logic and Neural Networks as an extension of the Cognitive Maps [3]. FCMs were introduced first by Kosko [4] and since then a wide range of applications in modeling complex dynamic systems have been reported such as medical [7], environmental [2], supervisory [9] and political [1, 5, 6] systems. Essentially, a Fuzzy Cognitive Map is developed by integrating the existing experience and knowledge regarding a system. This can be achieved by using a group of human experts to describe the system's structure and behavior in different conditions.

   In a graphical form, the FCMs are typically signed fuzzy weighted graphs, usually involving feedbacks, consisting of nodes and directed links connecting them. The nodes represent descriptive behavioral concepts of the system and the links represent cause-effect relations between the concepts [8]. In the context of FCM theory, the fuzzy value of a concept denotes the degree in which the specific concept is active in the general system, usually bounded in a normalized range of [0, 1]. Furthermore, the

weights (or sensitivity) of the system's interrelations reflect the degree of influence between two concepts and they are usually made to vary in the interval [-1,+1][8].

Considering the mathematical formulation of the FCM, an FCM system can be represented by a concept state vector A and an adjacency matrix W. The dimension of the concept vector is equal to the total number of distinct concepts n that compose the modeled system. Moreover, the values of its elements correspond to the states of the concepts. The nxn adjacency matrix represents the cause – effect interrelations among the concepts. The value of each cell $[i,j]$ corresponds to the weight of the cause – effect relation between the ith concept and the jth concept.

The basic FCM inference is an iterative process of estimating the new values of the concept vector until they converge. The change of the value of a concept depends on the values of the connected concepts along with the corresponding weights of their interrelations. In practice, at each iteration, the produced concept state vector A^{t+1} emanates from a vector – matrix multiplication, using the previous concept vector A^t and the adjacency matrix W.

After the system interacts for a number of iterations there are 3 possible system behaviors to appear: 1. *Fixed point attractor*, where the activation levels of all concepts become steady and don't alternate through iterations. 2. *Limit Cycle*, where an identical sequence of activation levels appears every fixed number of iterations. 3. *Chaotic attractor* where the activation levels keep changing for ever through iterations.

Additionally, the variation of a concept's impact on another concept may be defined through a manner expressed by a variety of functions (e.g. linear, stepwise, sigmoid, Gaussian and exponential) [11].

A wide range of real life dynamical systems are characterized by causality through their interrelations. Most of the times, experienced and knowledgeable users can identify the parameters and the conditions of their system. However, the complexity of these systems acts as a prohibitive factor on prognosis of their future states. Nevertheless, it is crucial for the users to make a prognosis on the cost of changing a state of a concept, in taking a decision whether they will take such a risk of change or not. FCM modeling constitutes an alternative way of building intelligent systems providing exactly this opportunity to the users; predicting the final states of the system caused by a change on initial concept states. The proposed tool implements the FCM inference mechanism in a simple manner by combining in a simple, easy to use and friendly environment the mechanics of FCM and a user's knowledge. This leads to the modeling of the system in an intelligent framework. An attempt has also been reported in [14].

The main purpose of designing this tool was to create a user-friendly software for building and simulating complex dynamic systems via FCM, with little computer literacy or experience from the part of a user. The only requirement from the user's part is to go through the fuzzification process for the concept values and the weights of their interrelations. The main goal is to help users to design and simulate FCM systems even when they have only the basic knowledge of FCM modeling, provided that they have a good and clear understanding of the parameters and conditions of their system, as well as the corresponding fuzzy values. The long-term objective is to develop a multi-function tool, incorporating certain modeling procedures and algorithms, such as genetic algorithms and neural networks, as well as fuzzification

and defuzzification functions. The first edition of the tool implementation, which is limited to simple simulation and testing, is presented in this paper.

2 A Generic Tool for Running FCM systems

In FCM model, the causality propagation is accomplished through the configuration of the new concept values at each iteration. The proposed system allows the user to choose out of three functions for computing the new concept values.

$$A_i^t = f\left(\sum_{\substack{j=1 \\ j \neq i}}^{n}\left(A_j^{t-1}W_{ji}\right)\right). \tag{1}[4]$$

$$A_i^t = f\left(\sum_{\substack{j=1 \\ j \neq i}}^{n}\left(A_j^{t-1}W_{ji} + A_i^{t-1}\right)\right). \tag{2}[9]$$

$$A_i^t = f\left(\sum_{\substack{j=1 \\ j \neq i}}^{n}\left(k_1\,A_j^{t-1}W_{ji} + k_2 A_i^{t-1}\right)\right). \tag{3}[9]$$

In all the above equations A_i^t is the concept value of concept C_i at iteration t, n is the total number of the concepts affecting concept C_i, W_{ji} is the value of the sesnsitivity connecting concept C_j with (to) concept C_i, k_1 is the coefficient which controls the influence of the interconnected concepts in the calculation of the new concept value A_i^t, k_2 is the coefficient that expresses the contribution of the previous concept value in the calculation of the new one, $f(.)$ is a transformation function which maps each concept value into the space of either [-1,1] or [0,1]. The user is able to choose between sigmoid, hyperbolic, bivalent or trivalent function $f(.)$.

2.1 Interaction with the system

The graphical environment of the tool is created using the available interface features that MATLAB 7.9 (R2009 b) provides. The software tool incorporates Microsoft Excel for facilitating the storing and retrieving of data from files.

By invoking the tool, a window with a tabbed menu navigation system appears as shown in Figure 1.
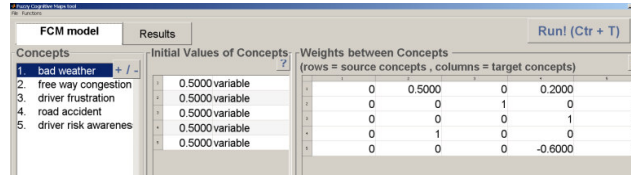

Fig. 1. The initial screen of the main window

The tabs divide the window into two screens. By selecting the firs tab *"Set FCM Model Parameters"* the user is lead to a screen where one can set or modify the FCM

model's concepts, their initial values, and the weights of their interrelations. The second tab named *Results* brings out a screen with information about the final values of the concepts. In the case where no results are available for any simulation of any model, the tab *Results* is disabled. The user has the flexibility to switch between *FCM model's parameters settings* and the *Results* screens whenever one desires. A button named *Run* is provided for executing the FCM model

2.1.1 Model Building and editing

The user can either build an FCM model from scratch or modify an existing one through the "Set FCM mode's Parameters" screen. The names of the concepts are listed in a list-box, while the initial values of the concepts and their weights are exposed in tables. The first step of building a model is the definition of the names of the concepts. The user can insert, delete, edit the name of a concept or re-order the list of concepts. For each new concept added to the model, the system adds a new row to the "Initial Values of Concepts" and the "Weights between Concepts" table and initializes their numeric values to 0. Each numbered row of the table corresponds to the same numbered concept of the "Concepts" list. The user can edit the numeric values of the tables. There is also a pop-up menu next to each initial concept value that allows the user to lock the corresponding value, by choosing the option "constant" (i.e. execute this concept out of any calculations that alter its initial value).

2.1.2 Running the system and Results

The user can run the FCM model by using the button "Run" located at the right top of the main window. When the algorithm stops executing, the "Results" screen appears (Figure 2).
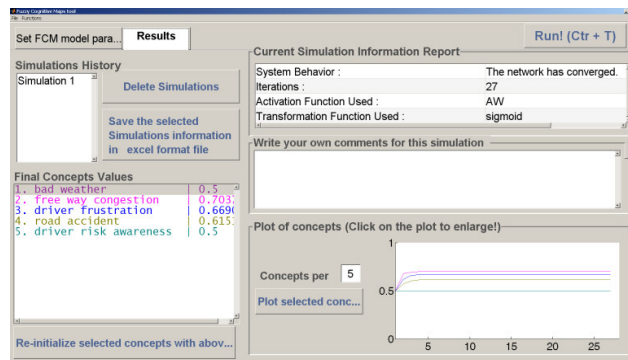

Fig. 2. The results screen

The final activation levels of the concepts appear on the "Final Concepts Values" list. Each concept in that list is colored and so do the values of the corresponding plot on the graph right next to the list. In the "Results" screen, the main graph presents the variation of the states of the model concepts. It is also possible to select the desired concepts to plot, from the "Final Concept Values" list and then click on the "Plot selected concepts" pushbutton right next to the graph. There is also an edit-box where the user can write comments and conclusions about a simulation. There is also a table displaying information about a simulation (functions used and the number of required iterations). There is also another list on this screen, named "Simulations History". Each time the user runs a model, the system saves all the information about the simulation and the "Simulations History" list adds one more "Simulation" at the bottom of the list. When the user selects a saved Simulation, all the corresponding information is retrieved and displayed. One can delete one or more simulations from the list. The user has also the ability to save a group of simulations in excel file format. The user is thus able to visit the results of each simulation of an FCM model easily and quickly, while one can transfer this simulations file to any computer running WINDOWS, without the need of the FCM tool.

3 Evaluation of the Tool

The credibility of the tool was tested by modeling and simulating a real process control problem taken from [9]. The results of the FCM model built with the tool agree with the results reported in [9].

For evaluating the usability of the tool, an evaluation questionnaire was developed. The evaluators had to carry out 5 tasks. The tasks cover all the functions a user is able to execute with the tool. After finishing the tasks the user had to answer 10 usability questions suggested in [12] and one extra question. Twenty senior computer science, mathematics and computer engineering students were invited to evaluate the tool. Seventeen of the selected students had never used tools for simulating graph models. All of the students had no knowledge about FCM modeling so an essential description of the FCM theory took place just before the evaluation.

Analytically, the 10 questions taken from [12] were:
1. I think that I would like to use this system frequently
2. I found the system unnecessarily complex
3. I though the system was easy to use
4. I think I would need the support of a technical person to be able to use this system
5. I found the various functions in this system were well integrated
6. I thought there was too much inconsistency in this system
7. I would imagine that most people would learn to use this system very quickly
8. I needed to learn a lot of things before I could get going with this system
9. I would easily learn how to user this tool
10. I found the system flexible to use

And the extra question included in the questionnaires was:

I think I would save time by using the tool instead of developing the FCM model without the tool.

The user had to express the degree of agreement or disagreement with each statement on a seven point Likert scale. Each questionnaire was first analyzed by calculating the score from each item (statement). For items 1, 3, 5, 7, 9, 10, 11, as listed above, the score was the scale position minus one. On the other hand, for items 2, 4, 6, 8 the score was seven minus the scale position [12]. As a result, there is a mapping between the Likert style user's choice, which ranges in [1,7], to a score ranging in [0,6] where, {0,1,2,3,4,5,6} stand for {"very bad", "bad", "tend to be bad", "moderate", "tend to be good", "good", "very good"} system in terms of usability, respectively. As shown in Figure 3, none of the questions was scored as "very bad". Only one question had 5% scores in "bad" category. Only the 5% of four questions were scored below "moderate". Considering the worst scores, *Question 6*
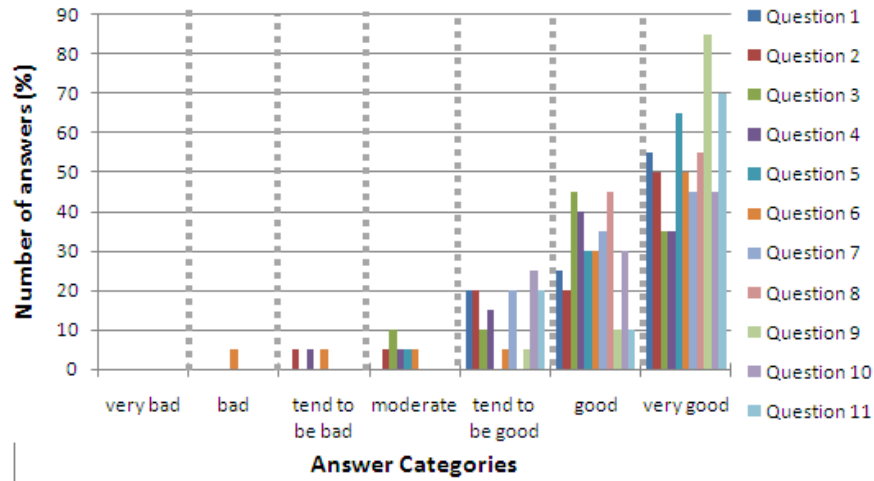

Fig. 3. Histogram presenting the percentage of the answers distribution per each question. The grey dashed lines divide the histogram into the 7 answer categories.had 10% scores

is the only one that had 10% scores falling in both "bad" and "tend to be bad" categories. After the first analysis of the questionnaires the students were asked why they thought there wasn't satisfactory consistency in the system. The users stated that they couldn't understand the word "consistency" and they got confused. *Question 2,* had also answers in the group of "tend to be bad","moderate" and "tend to be good" categories. A possible reason for that could be that the users felt they had to memorize many ways for utilizing data, since the first tab screen is about editing tables and a listbox, while the second one is more about pushing buttons and utilizing plots. 25% of the answers of *Question 4* were scored as "tend to be bad", "moderate" and "tend to be good". A possible reason why users felt that way is that they didn't have any experience with graph models simulation tools or FCM. The same reason applies for the "moderate" and "tend to be good" scores of *Question 3*. Hence, it is absolutely

reasonable that some of the users would face some difficulties using for the first time the tool and they would ask for help from a technical person. Besides, 95% of the answers of *Question 9* fall in "good" and "very good" categories, stating that 95% of the users believe they would easily learn how to use the tool. This implies that the users understand that the difficulties they had to resolve during the evaluation came from their inexperience and lack of knowledge. Furthermore, it is noteworthy that all eleven questions had 70% - 100% of their answers falling into categories "good" and "very good" indicating that most of the users thought the system was indeed successful in terms of usability. This fact is also reflected in Figure 4, whereas all question medians range between five and six (good and very good) and approximately 50% of them equal to very good which is the optimum score.
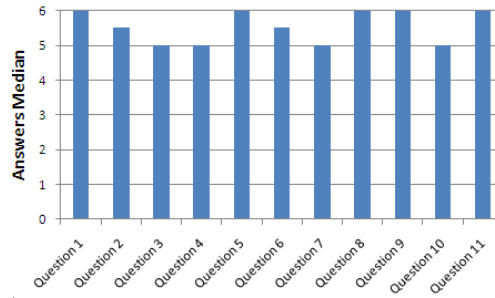

Fig. 4. Histogram presenting the median of the answers of each question

4 Conclusions and Future Work

Dynamical complex systems are met in most human decision making activities. Decision making problems concerning such systems can be solved by utilizing FCM modeling methodology. The proposed tool enables a user, who is not familiar with computer literacy, to design and simulate an FCM modeled system and benefit from the advantageous features of FCM modeling.

The tool and its capabilities were presented and described in detail in the presented paper. The friendly graphical interface helps the user to easily insert, save, retrieve and display data. The main advantages of the tool is the preservation of the information of multiple simulations of different FCM models at the same time, the display of the plot of each model final states in different windows and the feature of saving information of multiple simulations in an spreadsheet file so that the user is able to go back to the results of any simulation of any model without the need to re-build and re-simulate the system on the tool.

The usability of the system was evaluated by developing an evaluation questionnaire. The conclusions drawn from the results of the evaluation are (1) the tool meets the usability standards as defined in [12] since the majority of the students gave very good scores for the usability of the tool through their answers (2) users

having the basic knowledge of FCM theory are capable of using the tool successfully in designing and simulating FCM systems even if they had never interacted with FCM modeling systems in the past (3) the tool is easy to learn and easy to use.

The tool is expected to introduce common people to the challenging world of FCMs, by offering them a user-friendly functional environment which allows them to create and simulate FCM models.

In the future, the tool will be enhanced with pre-processing functions, helping and enabling the user to extract the fuzzy rules which characterize a system and embed this knowledge to the tool. The user will be also able to "run" a demonstration of built examples of the tool. In addition, the notion of time, in terms of time dependencies, as well as an algorithm inspired by the theory of evaluation of species will be introduced in the model.

References

1. A. Andreou, Nicos H. Mateou and G. Zombanakis.: The Cyprus Puzzle and the Greek-Turkish Arms Race: Forecasting Developments Using Genetically Evolved Fuzzy Cognitive Maps. Journal of Defence and Peace Making 14, 293-310 (2003)
2. K. Kok.: The potential of Fuzzy Cognitive Maps for semi-quantitative scenario development, with an example from Brazil. Global Environ. Change 19, 122-133 (2009)
3. B. Kosko.: A dynamic systems approach to Machine Intelligence, Neural Networks and Fuzzy Systems 2nd edn. Prentice Hall London (1992)
4. B. Kosko.: Fuzzy cognitive maps. International Journal of Man-Machine Studies 24, 65-75 (1986)
5. C. Neocleous and C. Schizas.: Application of Fuzzy Cognitive Maps to the Political-Economic Problem of Cyprus. pp. 340-349. (2004)
6. C. Neocleous, C. Schizas and C. Yenethlis.: Fuzzy Cognitive Models in Studying Political Dynamics - The case of the Cyprus problem.
7. E. I. Papageorgiou, N. I. Papandrianos, D. J. Apostolopoulos and P. J. Vassilakos.: Fuzzy Cognitive Map based decision support system for thyroid diagnosis management. pp. 1204-1211. (2008)
8. C. D. Stylios and P. P. Groumpos.: Modeling Complex Systems Using Fuzzy Cognitive Maps. IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans. 34, 155-162 (2004)
9. Chrysostomos D. Stylios and Peter P. Groumpos.: Fuzzy Cognitive Maps in modeling supervisory control systems. Journal of Intelligent and Fuzzy Systems 8, 83-98 (2000)
10. E. I. Papageorgiou and P. P. Groumpos.: A new hybrid method using evolutionary algorithms to train Fuzzy Cognitive Maps. Applied Soft Computing Journal 5, 409-431 (2005)
11. D.E. Koulouriotis, I.E. Diakoulakis, D.M. Emiris, E. N. Antonidakis, I.A. Kaliakatsos.: Efficiently modeling and controlling complex dynamic systems using evolutionary fuzzy cognitive maps (invited paper). International Journal of Computational Cognition 1, 41—65 (2003)
12. Brooke John.: SUS – A quick and dirty usability scale. In: P. W. Jordan, B.Thomas, B.A. Weerdmeester & A. L. McClelland. Usability Evaluation in Industry. London: Taylor and Francis (1996)
13. N. H. Mateou, C. Stylianou, A. S. Andreou.: Hybrid Fuzzy Cognitive Map Modeller, Soft computing as Transdisciplinary Science and Technology 29, p.p. 851—862. Springer Berlin/Heidelberg (2005)