A Prolog Based System That Assists Experts to Construct and Simulate Fuzzy Cognitive Maps

Athanasios Tsadiras

Department of Information Technology
Alexander Technological Educational Institute of Thessaloniki
P.O.BOX 141, 57400, Thessaloniki, Greece
tsadiras@it.teithe.gr

Abstract. The method of Fuzzy Cognitive Map (FCM) is a combination of Fuzzy Logic and Artificial Neural Networks that is heavily used by experts and scientists of a diversity of disciplines, for strategic planning, decision making and predictions. A system that would assist decision makers to represent and simulate their own developed Fuzzy Cognitive Maps would be highly appreciated by them, especially from those that do not possess adequate computer skills. In this paper, a Prolog based system is designed and implemented to assist experts to both construct and simulate of their own FCMs. The representation capabilities of the system and the design choices are discussed and a variety of examples are given to demonstrate the use of the system.

Keywords: Fuzzy Cognitive Maps; Prolog; Simulation; Decision Making; Predictions.

1 Introduction

Fuzzy Cognitive Map (FCM) is a formal method for making predictions and taking decisions, that is used by scientists from various disciplines such as economy & management [1-3], industry [4], medicine [5], political science [6,7], and ecology [8,9]. The success of the construction of an FCM is heavily depended on the degree of the expertise of the domain experts involved in the FCM construction. Making decision through FCM, requires the simulation of the FCM model, which is a difficult task especially from those scientists that do not possess the necessary computer skills. The need of a system that would assist scientists to construct and simulate their own FCMs led us to the creation of a Prolog-based such system, which we will discuss in the following sections.. After a short introduction to FCMs in section 2, the representation capabilities of the system are discussed (section 3). The design of the system is presented in section 4, followed by a section concerning the implementation and the demonstration of the system's capabilities. Finally, in section 6, a summary is presented, accompanied by a number of conclusions.

2 Fuzzy Cognitive Maps

Based on Axelord's work on Cognitive Maps [10], Kosko introduced in 1986, Fuzzy Cognitive Maps (FCMs) [11,12]. FCMs are considered a combination of fuzzy logic

J. Darzentas et al. (Eds.): SETN 2008, LNAI 5138, pp. 288–300, 2008.

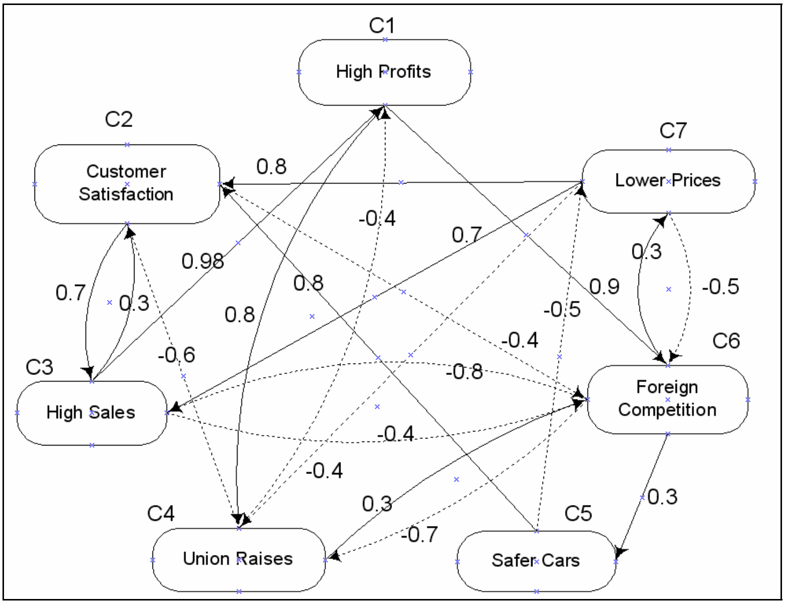


Fig. 1. An FCM concerning a car industry (modified version of original taken from [18])

and artificial neural networks and many researchers have made extensive studies on their capabilities (see for example [13-17]). An example of an FCM, concerning a car industry [18], is given in Figure 1.

FCMs create models as collections of concepts and the various causal relations that exist between these concepts. So in figure 1, the nodes represent the concepts that are involved in the model and the directed arcs between the nodes represent causal relationships between the corresponding concepts. The type of the causal relation between the two nodes is determined by the weight that each arc is accompanied. A positive (negative) causal relation between two concepts C_i and C_j means that an increase of the activation level of concept C_i will increase (decrease) C_j and also a decrease of concept C_i will decrease (increase) C_j.

The level of activation of each concept C_i at time step t is determined by number A_i^t, i=1,...,n with n to be the number of concepts of the FCM. We define w_{ij} as the weight of the arc that connects C_i and C_j. There is synchronous updating to the FCM system which means that A_i^{t+1}, i=1,...,n is calculated by one of the following formulas:

i) $A_i^{t+1} = f_M\,(A_i^t, S_i^t) - d\ A_i^t$ (using Certainty Neurons [15]) or (1)

ii) $A_i^{t+1} = f_M(w_{1i}A_1^t, f_M(w_{2i}A_2^t, f_M(\ldots, f_M(w_{n-1,i}A_{n-1}^t, w_{n,i}A_n^t)))) - dA_i^t$ (2)
(using Recursive Certainty Neurons [19])

where,

- $S_i^t = \sum_j w_{ji} A_j^t$ is the sum of the weight influences that concept C_i receives at time step t from all other concepts,
- d is a decay factor and
- $$f_M(A_i^t, S_i^t) = \begin{cases} A_i^t + S_i^t(1 - A_i^t) = A_i^t + S_i^t - S_i^t A_i^t & \text{if } A_i^t \geq 0, S_i^t \geq 0 \\ A_i^t + S_i^t(1 + A_i^t) = A_i^t + S_i^t + S_i^t A_i^t & \text{if } A_i^t < 0, S_i^t < 0 \quad \left|A_i^t\right|, \left|S_i^t\right| \leq 1 \\ (A_i^t + S_i^t) \big/ \left(1 - \min(\left|A_i^t\right|, \left|S_i^t\right|)\right) & \text{if } A_i^t S_i^t < 0 \end{cases} \tag{3}$$

is the function that was used for the aggregation of certainty factors at the MYCIN expert system [20].

3 Representation Capabilities

As it is discussed above, an FCM contains a number of arcs that represent causal relationships between the concepts of the FCM. For example, in figure 1, the arc between concept C_2:"Customers Satisfaction" and C_3:"High Sales" with weight 0.7 can be interpreted as the following statement

customers satisfaction influences sales with weight 0.7.

This means that in order to completely represent the FCM we will need as many such statements as many arcs/causal relationships exist in the FCM. Each of these statements would be in the following format:

Concept1 "influences" Concept2 "with weight" W.

Statements that are also useful for the simulation of the FCM are such as the ones below:

foreign competition is set to 0.8. or
foreign competition is set to high.

Statements like those would mean that the activation level of the specific concept should be kept steady to that certain level through the whole simulation process. Such statements can have the following form:

Concept "is set to" Value.

Another parameter that must be specified is that of the decay factor d, described in section 2. That parameter can be set to a value by a statement such as the following:

"decay is set to" Value.
(e.g. *decay is set to 0.1.)*

The type of transfer function that should be used in the simulation can be either a) that described in equation (1) or b) that of equation (2). This can be stated in the following way.

"*use mycin.*"	for equation (1) or
"use recursive."	for equation (2)

From the all the above, we define a grammar, written in valid Prolog syntax [21], for assisting the representation of FCMs, in figure 2.

```
statement-->concept,[influences], concept,[with,weight],weight.
statement-->concept,[influence], concept,[with,weight],weight.
statement -->[decay, is,set, to], number05.
statement-->concept,[is,set,to],value.
statement-->concept,[are,set,to],value.
statement-->[use, recursive].
statement-->[use,mycin].

concept-->[X].
concept-->[X],concept.

weight-->[X],{number(X)},{X=<1}, {-1=<X}.

value-->weight.
value--> [X], {member(X,[low,average, high])}.
value-->[very,high].
value-->[very,low].

number05--> [X],{number(X)},{X=<0.5}, {0=<X}.
```

Fig. 2. Grammar for representing FCMs

In the above grammar, special treatment exists for allowing the use of plural in statements. So

a) “influences” and “influence” are equivalent
b) “is set to” and “are set to” are equivalent

Also “concept” is defined as a series of one or more words and “weights” are only numbers in the interval [-1,1]. Correct “value” is considered not only any number in the interval [-1,1], but also a linguistic value such as “very low”, “low”, “average”, “high” and “very high” in a manner closer to fuzzy logic. These linguistic values during simulation are translated to the values 0.1, 0.3, 0.5, 0.7, 0.9, respectively. Finally, the decay factor is allowed to take only a value of a number in the interval [0, 0.5]. The above syntax contains all the necessary facilities for describing easily and clearly an FCM structure but also contains limitations in order to prevent the user from presenting ambiguities or misunderstandings.

Using the above syntax, the FCM of figure 1 can be represented, as it is shown in figure 3.

The statements of figure 3 are not close to a computer programming language, but closer to natural language, requiring limited computer skills from its author. This means that the author can concentrate on the FCM itself and not to its computer representation. The simplicity of the grammar used leads the domain experts themselves to describe and execute their own FCMs.

```
profits influences strikes with weight 0.8.
profits influences foreign competition with weight 0.9.
customers satisfaction influences sales with weight 0.7.
customers satisfaction influences foreign competition with weight -0.4.
sales influence profits with weight 0.98.
sales influence  customers satisfaction with weight 0.6.
sales influence foreign competition with weight -0.4.
strikes influence profits with weight -0.4.
strikes influence  customers satisfaction with weight -0.3.
strikes influence foreign competition with weight 0.7.
safe cars influence customers satisfaction with weight 0.8.
safe cars influence foreign competition with weight 0.3.
safe cars influence low prices with weight -0.5.
foreign competition influences sales with weight -0.8.
foreign competition influences strikes with weight -0.7.
foreign competition influences safe cars with weight 0.3.
foreign competition influences low prices with weight 0.3.
low prices influence customers satisfaction with weight 0.8.
low prices influence profits with weight 0.7.
low prices influence strikes with weight -0.4.
low prices influence foreign competition with weight -0.5.

foreign competition is set to 0.5.
decay is set to 0.1.
use recursive.
```

Fig. 3. Statements representing the FCM of figure 1

4 Designing the FCM Assisting System

Looking the statements in figure 3 we can see that they resemble facts of the Prolog language. The type of these statements suits to the declarative nature of Prolog. Furthermore, the grammar of figure 2 can be directly handled by Prolog which has been used successfully in Natural Language Processing [22]. These are the main reasons for choosing the Prolog programming language for implementing the FCM Assistance System. The design of the system is shown in figure 4.

The system has the following three components/stages:

<u>Stage 1 - Lexical Analysis.</u> Statements such as those of figure 3, concerning the structure of the FCMs are written in a simple text file having ".fcm" extension. This file is read character by character and valid words, numbers and punctuation marks, such as the full stop, are identified. If this stage succeeds, the system continues with stage 2, otherwise the system ends its process and returns a corresponding error message.

In this stage, regular expressions [23] as the following are used:

- Natural number = [0-9]+
- Integer=(+|-)?{natural}
- Decimal= {integer}("."{natural})?(E {natural})?

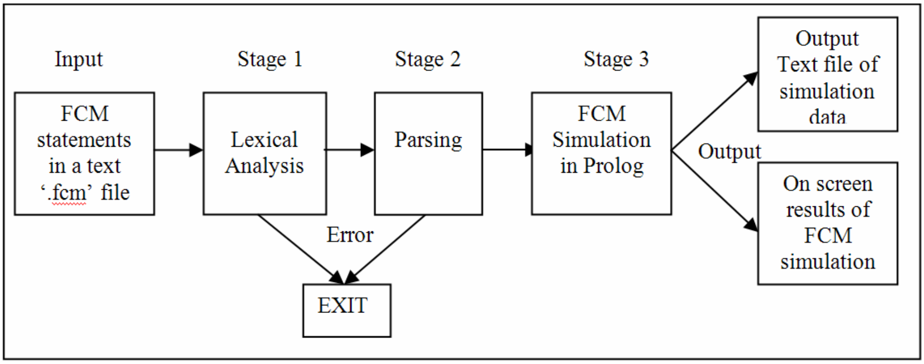


Fig. 4. The design of the FCM Assistance System

Stage 2 - Parsing: In this stage a parser is required. After the identification of words, numbers and punctuation marks, the parser checks whether or not the statements follow the syntactical rules defined in section 3. If the parser succeeds then the system continues with stage 3, otherwise it ends its process and returns a relevant error message in order to help the user to identify the syntax error and correct it.

Stage 3 - FCM Simulation in Prolog: After passing Stage 2, the system has all the necessary information for constructing and simulating the FCM. Prolog code is written in order to simulate the FCM imposed to the system. The results of the simulation are either a) shown on the computer screen or b) stored in a data file. FCM can reach a) an equilibrium point, b) a limit circle periodical behaviour or c) a chaotic behaviour [24]. If an equilibrium point is reached, this point is shown on the computer screen and the activation levels of the concepts of the FCM model through the transition phase towards this equilibrium point are saved in a ".txt" data file for further examination. If the FCM model reaches a limit circle behaviour or a chaotic behaviour then the system simulates 5000 time steps of interaction, and saves the activation levels of the concepts of the FCM model through these 5000 time steps to a ".txt" file for further examination to a graph drawing software tool.

5 Implementation and Demonstration of the FCM Assistance System

The FCM assistance system is implemented in Prolog, using SWI-Prolog [25], following the design described in section 4. To demonstrate its implementation, the FCM of figure 1 will be used. To do that, the statements of figure 3 are saved in a file called 'car_industry.fcm' and inserted to the system. The output is shown in figure 5, where seven (7) different concepts are identified and one of them, concept "Foreign Competition", is set constant to 0.5. This means that the imposed scenario #1, attempts to predict the consequences of having a moderate increase in foreign competition. After 70 time steps, FCM reaches an equilibrium point to the state that is presented in figure 5. From that, we can conclude that according to the FCM predictions, the reaction of the car industry to the increase of foreign competition is to create safer cars (0.6). This leads to increase in prices (low prices=-0.638) so sales are significantly decreased (-0.880) and the same applies to profits (-0.898) and movements toward strikes (-0.883). All the above, cause customers satisfaction to decrease (-0.798). The data of the interactions during the 70 time steps are saved to an output text file. This file can be processed by a graph drawing software tool and the transition phase towards equilibrium can be exhibited. This transition phase for the case above is shown in figure 6.

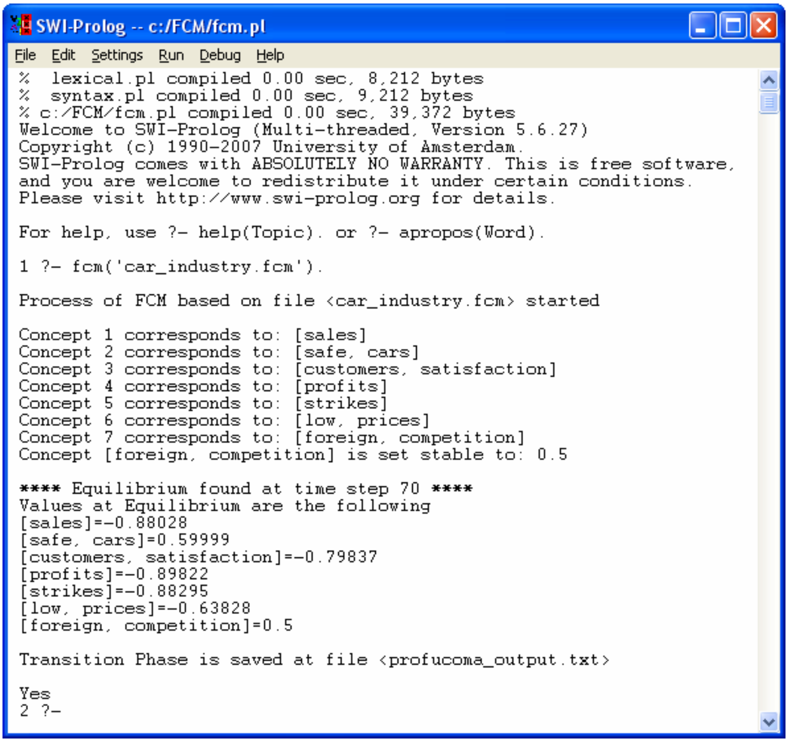


Fig. 5. Simulation of FCM concerning a Car Industry. Scenario #1: Foreign Competition is set to 0.5.

We can conclude the car industry is vulnerable to foreign competition. Other "what-if" scenarios can be imposed to the system, in order the decision maker to examine the predicted outcomes according to the FCM. For example, a decision maker would like to see the prediction of the system in the case, the direct way that foreign competition affect movements towards safer cars is lowered. This means that the initial statement

foreign competition influences safe cars with <u>weight 0.3.</u>

should be changed to

foreign competition influences safe cars with <u>weight 0.1.</u>

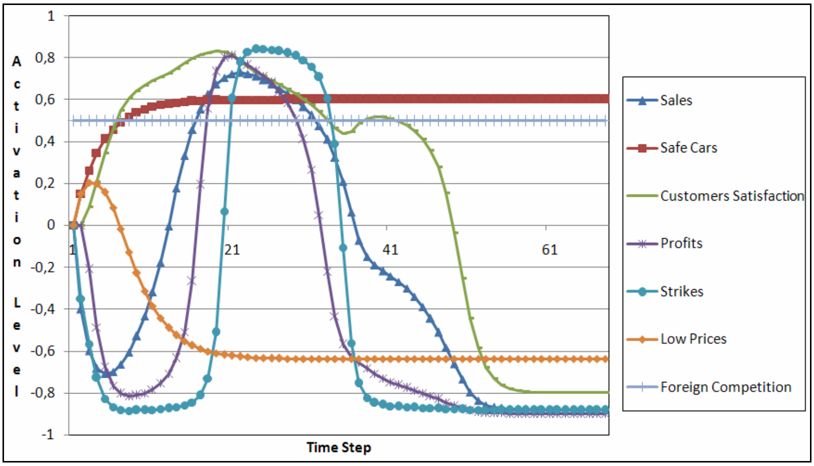
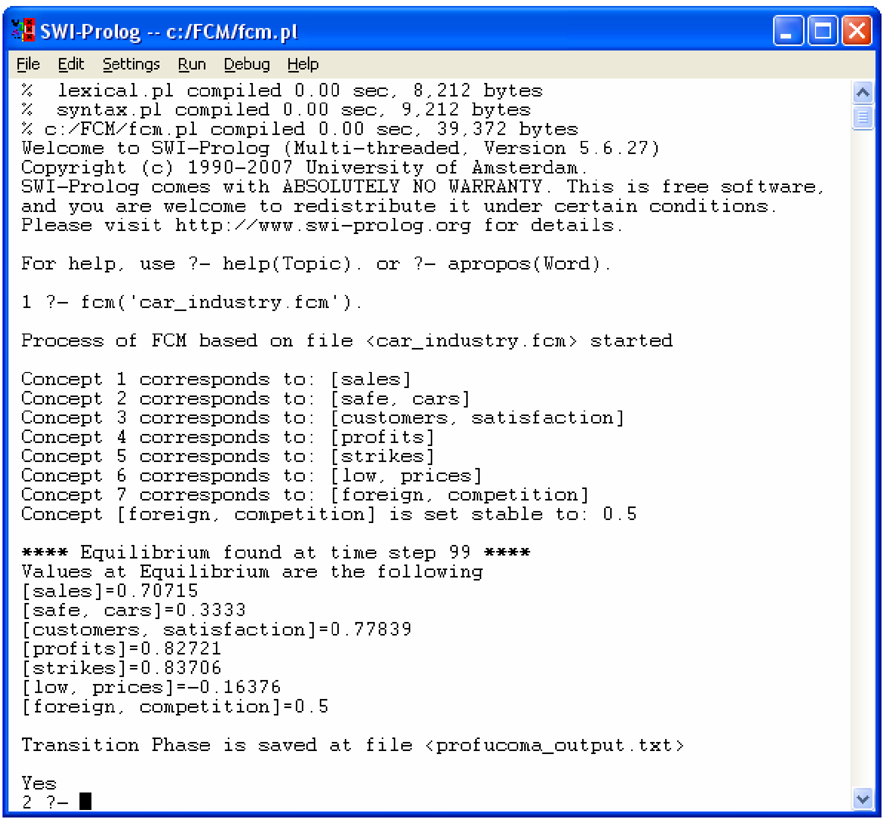


Fig. 6. Transition phase towards equilibrium for scenario #1



```
SWI-Prolog -- c:/FCM/fcm.pl
File Edit Settings Run Debug Help
%  lexical.pl compiled 0.00 sec, 8,212 bytes
%  syntax.pl compiled 0.00 sec, 9,212 bytes
% c:/FCM/fcm.pl compiled 0.00 sec, 39,372 bytes
Welcome to SWI-Prolog (Multi-threaded, Version 5.6.27)
Copyright (c) 1990-2007 University of Amsterdam.
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software,
and you are welcome to redistribute it under certain conditions.
Please visit http://www.swi-prolog.org for details.

For help, use ?- help(Topic). or ?- apropos(Word).

1 ?- fcm('car_industry.fcm').

Process of FCM based on file <car_industry.fcm> started

Concept 1 corresponds to: [sales]
Concept 2 corresponds to: [safe, cars]
Concept 3 corresponds to: [customers, satisfaction]
Concept 4 corresponds to: [profits]
Concept 5 corresponds to: [strikes]
Concept 6 corresponds to: [low, prices]
Concept 7 corresponds to: [foreign, competition]
Concept [foreign, competition] is set stable to: 0.5

**** Equilibrium found at time step 99 ****
Values at Equilibrium are the following
[sales]=0.70715
[safe, cars]=0.3333
[customers, satisfaction]=0.77839
[profits]=0.82721
[strikes]=0.83706
[low, prices]=-0.16376
[foreign, competition]=0.5

Transition Phase is saved at file <profucoma_output.txt>

Yes
2 ?-
```

Fig. 7. Simulation of FCM concerning scenario #2

Imposing this new scenario #2, the outcome is that of figure 7. Once again seven (7) different concepts are identified, and one of them, concept "Foreign Competition", is set constant to 0.5. After 99 time steps, FCM reached an equilibrium point, with the concepts to be activated to the degree shown in figure 7. It can be concluded that according to the FCM predictions, now the reaction of the car industry to the increase of foreign competition is to create safer cars (0.333) but to a smaller degree than that of scenario #1 where safer cars=0.6. This leads to smaller increases in prices (low prices=-0.163) so sales are increased (0.707), profits are also increased (0.827) and the same applies to movements toward strikes (0.837). All the above, cause customers satisfaction to increase (0.778). This is a much better scenario for the car industry than that of scenario #1. An output text file contains the data of the interactions during the 99 time steps. Figure 8 exhibits the transition phase towards equilibrium, based on this data.

It should also be mentioned that any lexical or syntax error in the statements of the ".fcm" file will lead to the end of the compilation. A corresponding error message is produced that will help the user to identify the lexical or syntax error. For example if in the 'car_industry.fcm' file the statement

customers satisfaction <u>influences</u> sales with weight 0.7.

was mistakenly written as

customers satisfaction <u>inffluences</u> sales with weight 0.7.

then the error will be identified and an error message will be shown. Such an error message for the case above is shown in figure 9. The user can see where the error is and an example of a correct similar statement is given.

Foreign Competition is set to 0.5. Now "foreign competition" influences construction of "safer cars" with weight 0.1. Equilibrium found after 99 time steps.

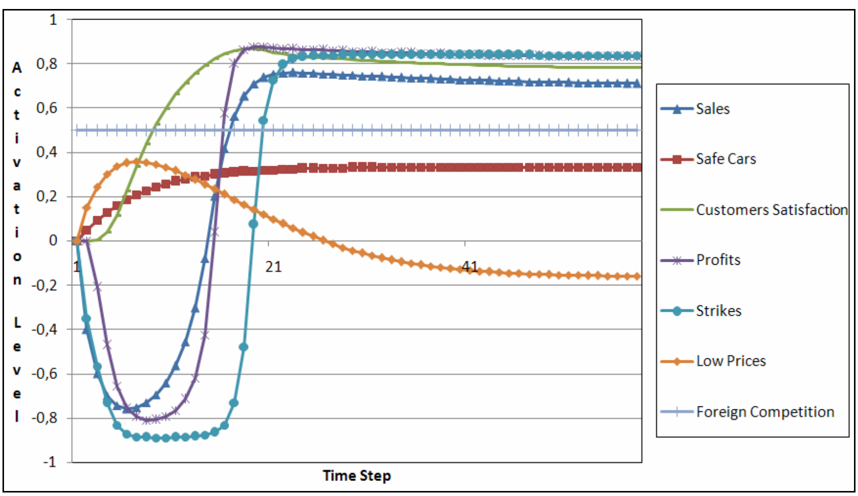
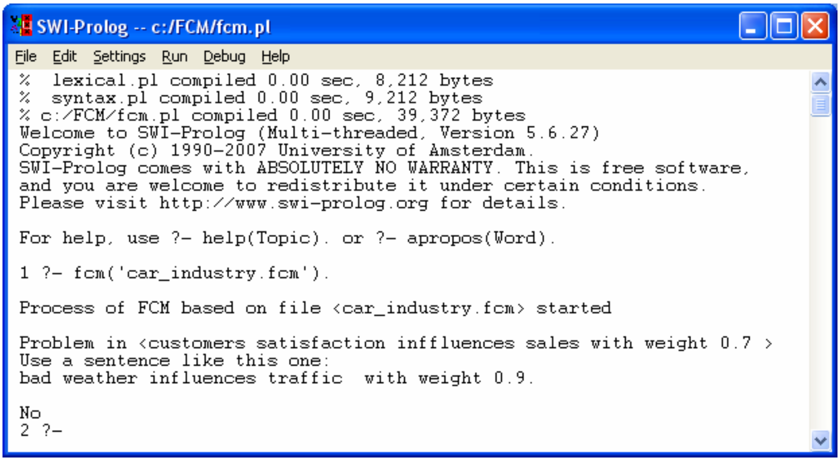


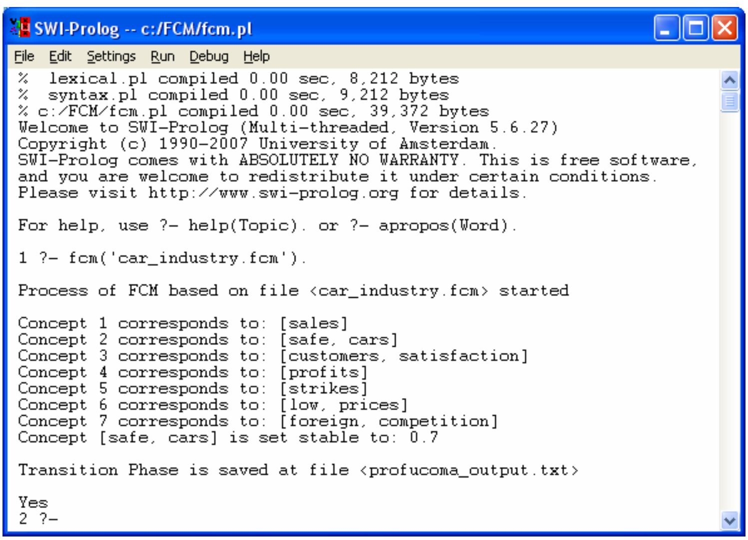
Fig. 8. Transition phase towards equilibrium for scenario #2




```
%  lexical.pl compiled 0.00 sec, 8,212 bytes
%  syntax.pl compiled 0.00 sec, 9,212 bytes
% c:/FCM/fcm.pl compiled 0.00 sec, 39,372 bytes
Welcome to SWI-Prolog (Multi-threaded, Version 5.6.27)
Copyright (c) 1990-2007 University of Amsterdam.
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software,
and you are welcome to redistribute it under certain conditions.
Please visit http://www.swi-prolog.org for details.

For help, use ?- help(Topic). or ?- apropos(Word).

1 ?- fcm('car_industry.fcm').

Process of FCM based on file <car_industry.fcm> started

Problem in <customers satisfaction infl uences sales with weight 0.7 >
Use a sentence like this one:
bad weather influences traffic  with weight 0.9.

No
2 ?-
```

Fig. 9. Syntax error found. User is guided by a similar correct statement.

Another scenario (scenario #3) is imposed to the system, to examine the consequences of a decision to manufacture much safer cars ("safer cars" is set stable to 0.7). Imposing this scenario to the system, the outcome is that of figure 10.




```
%  lexical.pl compiled 0.00 sec, 8,212 bytes
%  syntax.pl compiled 0.00 sec, 9,212 bytes
% c:/FCM/fcm.pl compiled 0.00 sec, 39,372 bytes
Welcome to SWI-Prolog (Multi-threaded, Version 5.6.27)
Copyright (c) 1990-2007 University of Amsterdam.
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software,
and you are welcome to redistribute it under certain conditions.
Please visit http://www.swi-prolog.org for details.

For help, use ?- help(Topic). or ?- apropos(Word).

1 ?- fcm('car_industry.fcm').

Process of FCM based on file <car_industry.fcm> started

Concept 1 corresponds to: [sales]
Concept 2 corresponds to: [safe, cars]
Concept 3 corresponds to: [customers, satisfaction]
Concept 4 corresponds to: [profits]
Concept 5 corresponds to: [strikes]
Concept 6 corresponds to: [low, prices]
Concept 7 corresponds to: [foreign, competition]
Concept [safe, cars] is set stable to: 0.7

Transition Phase is saved at file <profucoma_output.txt>

Yes
2 ?-
```

Fig. 10. Simulation of FCM concerning a Car Industry for scenario#3. "Safer cars" is set to 0.7. No equilibrium point found.

No equilibrium point is found even after making 5000 time steps. Conclusions can only be drawn by studying the transition phase, which is saved in the output text file. The transition phase according to the data of this output file is shown in figure 11. It

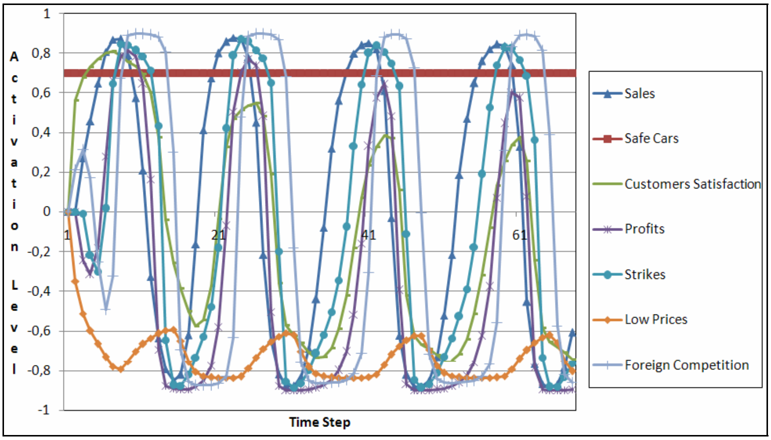


Fig. 11. Transition phase of an FCM concerning a car industry for scenario #3. A limit cycle - periodical behaviour is reached

is apparent that FCM enters a limit cycle, periodical behaviour where all concepts are getting high and low periodically. We can conclude for the car industry that these concepts will constantly increase and decrease, meaning that there will be strong interactions in the car industry and no equilibrium for this scenario. More scenarios can be inserted and the predicted outcome will be received in seconds. This is highly appreciated by decision makers that are willing to examine a number of "what-if" scenarios and check the predicted consequences of these imposed scenarios.

6 Summary – Conclusions

After a short introduction to Fuzzy Cognitive Maps, a Prolog based system is designed and implemented in order to:

a) read statements that represent causal relationships among concepts of the FCM
b) simulate the scenario written in the form of the above statements
c) present the equilibrium point of the imposed scenario (if such a point exists)
d) produce an output file that contain the data of the transition phase to visualize it if the user is interested in it.

The use of the system above is illustrated using a number of examples, to prove that it is useful for decision makers working with FCMs. The main reasons for that are:

a) the representation of FCMs is simple using close to natural language statements.
b) no programming skills are required to create and simulate FCMs,
c) a number of different scenarios can be tested easily and quickly, having the decision maker focused on the FCM itself and not to computer technicalities.

References

1. Billman, B., Courtney, J.F.: Automated Discovery in Managerial Problem Formulation: Formation of Causal Hypotheses for Cognitive Mapping. Decision Sciences 24, 23–41 (1993)
2. Bougon, M.G.: Congregate Cognitive Maps: A Unified Dynamic Theory of Organization and Strategy. Journal of Management Studies 29, 369–389 (1992)
3. Çoban, O., Seçme, G.: Prediction of socio-economical consequences of privatization at the firm level with fuzzy cognitive mapping. Information Sciences: an International Journal 169(1-2), 131–154 (2005)
4. Papageorgiou, E., Groumpos, P.: A Weight Adaptation Method for Fuzzy Cognitive Maps to a Process Control Problem. In: Bubak, M., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2004. LNCS, vol. 3037, pp. 515–522. Springer, Heidelberg (2004)
5. John, R.I., Innocent, P.R.: Modeling uncertainty in clinical diagnosis using fuzzy logic. IEEE Transactions on Systems, Man and Cybernetics 35(6), 1340–1350 (2005)
6. Andreou, A.S., Mateou, N.H., Zombanakis, G.A.: The Cyprus Puzzle And The Greek - Turkish Arms Race: Forecasting Developments Using Genetically Evolved Fuzzy Cognitive Maps. Defence and Peace Economics 14(4), 293–310 (2003)
7. Tsadiras, A.K., Kouskouvelis, I., Margaritis, K.G.: Using Fuzzy Cognitive Maps as a Decision Support System for Political Decisions. In: Manolopoulos, Y., Evripidou, S., Kakas, A.C. (eds.) PCI 2001. LNCS, vol. 2563, pp. 172–182. Springer, Heidelberg (2003)
8. Ramsey, D., Veltman, C.: Predicting the effects of perturbations on ecological communities: what can qualitative models offer? Journal of Animal Ecology 74, 905–916 (2005)
9. Hobbs, B.F., Ludsin, S., Knight, R.L., Ryan, P.A., Biberhofer, J.: Fuzzy Cognitive Mapping as a Tool to Define Management Objectives for Complex Ecosystems. Ecological Applications 12(5) (October 2002)
10. Axelrod, R.: Structure of Design. Princeton University Press, Princeton (1976)
11. Kosko, B.: Fuzzy Cognitive Maps. Inter. Jour. of Man-Machine Studies 24, 65–75 (1986)
12. Kosko, B.: Neural Networks and Fuzzy Systems. Prentice-Hall, Englewood Cliffs (1992)
13. Khan, M.S., Chong, A., Gedeon, T.: A Methodology for Developing Adaptive Fuzzy Cognitive Maps for Decision Support. Journal of Advanced Computational Intelligence 4(6), 403–407 (2000)
14. Khan, M.S., Quaddus, M.: Group Decision Support Using Fuzzy Cognitive Maps for Causal Reasoning. Group Decision and Negotiation 13, 463–480 (2004)
15. Tsadiras, A.K., Margaritis, K.G.: Cognitive Mapping and Certainty Neuron Fuzzy Cognitive Maps. Information Sciences 101, 109–130 (1997)
16. Taber, R., Yager, R.R., Helgason, C.M.: Quantization effects on the equilibrium behavior of combined fuzzy cognitive maps. International Journal of Intelligent Systems 22(2), 181–202 (2006)
17. Stach, W., Kurgan, L., Pedrycz, W., Reformat, M.: Genetic learning of fuzzy cognitive maps. Fuzzy Sets and Systems 153(3), 371–401 (2005)
18. Eberhart, R.C., Dobbins, R.W.: Neural Network PC Tools. Academic Press, London (1990)
19. Tsadiras, A.K., Margaritis, K.G.: Recursive Certainty Neurons and an Experimental Study of their Dynamical Behaviour. In: Proceedings of the European Congress on Intelligent Techniques and Soft Computing (EUFIT 1997), Aachen, Germany, September 1997, pp. 510–515 (1997)
20. Buchanan, B.G., Shortliffe, E.H.: Rule-Based Expert Systems. The MYCIN Experiments of the Stanford Heuristic Programming Project. Addison-Wesley, Reading (1984)

21. Bratko, I.: Prolog–Programming for Artificial Intelligence, 3rd edn. Addison Wesley, Reading (2000)
22. Sharp, R.: CAT2: An Experimental Eurotra Alternative. Machine Translation 6, 215–228 (1991)
23. Aho, V., Sethi, R., Ullman, J.D.: Compilers: Principles, Techniques and Tools. Addison-Wesley, Reading (1986)
24. Tsadiras, A.K., Margaritis, K.G.: An Experimental Study of Dynamics of the Certainty Neuron Fuzzy Cognitive Maps. NeuroComputing 24, 95–116 (1999)
25. SWI-Prolog, Free Software Prolog environment, licensed under the Lesser GNU Public License, `http://www.swi-prolog.org`