Modelling Complex Systems using the Pliant Cognitive Map

József D. Dombi[1] and József Dombi[2]

[1]Department of Software Engineering, University of Szeged, Arpad ter 2., Szeged, Hungary

[2]Department of Computer Algorithms and Artificial Intelligence, University of Szeged, Arpad ter 2., Szeged, Hungary

Keywords: Dynamic System, Pliant System, Cognitive Map, Fuzzy Operators.

Abstract: Here, we present a tool for describing and simulating dynamic systems. Our starting point is the aggregation concept, which was developed for multicriteria decision making. Using a continuous logic operator and a proper transformation of the sigmoid function, we build positive and negative effects. From the input data we can calculate the output effect with the help of the aggregation operator. Our approach is similar to that of the Fuzzy Cognitive Map. We shall introduce a new technique that is more efficient than the FCM method. The applicability of PCM is discussed and simulation results are presented.

1 INTRODUCTION

When we have to deal with a sophisticated system, we are confronted by certain difficulties as we have to represent it as a dynamic system. Using a dynamic system model can be hard computationally. In addition, formulating a system using a mathematical model may be difficult, or even impossible. Developing a model requires effort and specialized knowledge. Usually a system involves complicated causal chains, which might be non-linear. It should also be mentioned that numerical data may be hard to obtain, or it may contain certain errors, noise and incomplete values. Our approach seeks to overcome the above-mentioned difficulties. It is a qualitative approach where it is sufficient to have a rough description of the system and deep expert knowledge is not necessary. A similar approach was proposed by Kosko (Kosko, 1986; Kosko, 1994; Kosko, 1992), and it is called the Fuzzy Cognitive Map (FCM). FCMs are hybrid methods that lie in some sense between fuzzy systems and neural networks (Glykas, 2010; Salmeron et al., 2012; Yaman and Polat, 2009; Salmeron and Lopez, 2012; Maio et al., 2011). Knowledge is represented in a symbolic way using states, processes and events. Every piece of information has a numerical value. In Figure 1 we can see a typical FCM model, which is a directed graph.

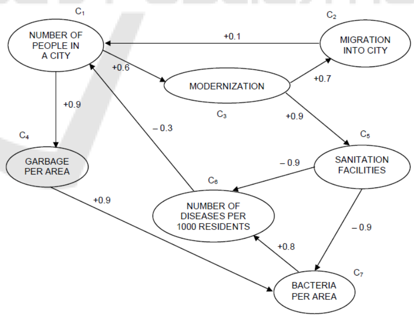

Figure 1: The FCM model.

The FCM approach allows us to perform qualitative simulations and experiment with a dynamic model. It has better properties than expert systems or neural networks, since it is relatively easy to use, it represents structured knowledge and inferences can be computed by numeric matrix operations instead of applying rules. In this paper, we will use a modification of the FCM concept so that ours better matches real-world modeling, which we call Pliant Cognitive Maps (Jozsef Dombi, 2005) (J. Dombi, 2005). We use cognitive maps to represent knowledge and to model decision making, which was first introduced by Axelrod (Axelrod, 1976). Kosko used fuzzy values and matrix multiplication to calculate the next state of a system. Here instead of values, we use time-dependent functions that are similar to impulse functions to represent positive and negative influences. Another improvement is that we drop the concept of matrix multiplication. On the one hand, matrix multiplication is not well suited in continuous logic (or fuzzy logic), where the true value is one and the false value is zero. On the other hand, general operators are more efficient for calculating the next step of a simulation. Logic and the cognitive map model correspond

D. Dombi J. and Dombi J..
Modelling Complex Systems using the Pliant Cognitive Map.
DOI: 10.5220/0004478205060512
In *Proceedings of the 15th International Conference on Enterprise Information Systems* (ICEIS-2013), pages 506-512
ISBN: 978-989-8565-59-4

to each other in the PCM case. It is easier to construct a PCM and after we have run PCM simulations and compared them with the real world, extracting knowledge is much easier. Combining cognitive maps with logic helps us to extract knowledge more efficiently, in contrast to when we use rule-based systems. The standard knowledge representation in expert systems is achieved through a decision tree. This form of knowledge representation in most cases cannot model the dynamic behaviour of the real world. The cognitive map describes the whole system by a graph showing the cause-effects that connect concepts. It is a directed graph with feedback that describes the real world concepts and the causal influences between them. From a logic point of view, causal concepts are unary operators of a continuous valued logic containing negation operators in the case of inhibition effects. The value of the node reflects the degree of system activity at any given time. Concept values are expressed on a normal [0,1] range. Values do not denote exact quantities, but the degree of activation. The inverse of the normalization might express the values coming from the real world, i.e. using a sigmoid function. Unlike the Fuzzy Cognitive Map, we do not use thresholds to force it to take values between zero and one. The mapping is a variation of the ”fuzzification” process in fuzzy logic, and it always hinders our desire to get quantitative results. In Pliant logic we map the real world into the logical model. These maps are continuous, strictly monotonous increasing functions, and so the inverse of these functions yields data about the real world. This paper is organized as follows. Section II describes the representation and mathematical formulation of the PCM concept compared to the FCM concept. Section III describes the components of the PCM, while Section IV describes how to create the PCM model. In Section V we discuss the development of a FCM model for the heat exchanger system that is common in the process industry. Section VI presents the features and potential use of the PCM for modeling complex systems. Lastly, in Section VII., we summarize our findings.

2 PLIANT COGNITIVE MAPS

In the FCM, a causal relationship is expressed by either positive or negative functions that have different weights. As we mentioned earlier, this will be replaced by unary operators in the PCM. First, let $\{C_1,\ldots,C_m\}$ be a set of concepts. Define a directed graph over the concepts. A directed edge has a weight w_{ij} from conceptC_i to concept C_j. This weight measures the influence of C_i on C_j, where

- 0.5 is the neutral value,
- 0 is the maximum negative and
- 1 is the maximal positive influence or causality.

In the FCM, the weight value $w_{ij} \in [-1,0,1]$. In our case,

- $w_{ij} > 0.5$ means there is a direct (positive) causal relationship between concepts C_i and C_j. That is, the increase (decrease) in the value of C_i leads to an increase (decrease) in the value of C_j.
- $w_{ij} < 0.5$ means there is an inverse (negative) causal relationship between concepts C_i and C_j. That is, the increase (decrease) in the value of C_i leads to a decrease (increase) in the value of C_j.
- $w_{ij} = 0.5$ means there is no causal relationship between C_i and C_j.

During the simulation, the activation level a_i of concept C_i is calculated in an iterative way. In the FCM, the calculation rule that was initially introduced to calculate the value of each concept based only on the influence of the interconnected concepts is

$$A_i^t = f\left(\sum_{i\neq j} A_j^{t-1}\cdot W_{ji}\right),$$

where A_i^t is the value of concept C_i for time step t, A_j^{t-1} is the value of concept C_j for time step $t-1$, W_{ji} is the weight of the causal interconnection from the jth concept toward the ith concept and f is a threshold function. One of the most popular threshold functions is the sigmoid function, where $\lambda > 0$ determines the steepness of the continuous function f and squashes the contents of the function into the interval [0,1]:

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

. A more general FCM formula was proposed By Stylios et al. (Stylios and Groumpos, 2004) to calculate the values of concepts for each time step. Namely,

$$A_i^t = f\left(k_1^i \sum_{i\neq j} A_j^{t-1}\cdot W_{ji} + k_2^i A_i^{t-1}\right)$$

The coefficients k_1^i and k_2^i must satisfy the conditions $0 \le k_1^i \le 1$ and $0 \le k_2^i \le 1$. The coefficient k_1^i expresses the influence of the interconnected concepts in the configuration of the new value of concept A_i. The coefficient k_2^i represents the proportion of the contribution of the previous value of the concept in the computation of the new value. The FCM approach has the advantage that we get a new state vector by multiplying the previous state vector a by the edge matrix W, which shows the effect of the change in the activation level of one concept on another. In the Pliant concept, we aggregate the influences instead of summing up the values. The result always lies between 0 and 1, so we do not need normalization as an additional step. Aggregation in Pliant logic is a general operation, which contains conjunctive operators and disjunctive operators as well. Depending on the parameter called the neutral value of the aggregation operator, we can build logical operators like Dombi operators. Using PCMs (Pliant Cognitive Maps) we answer ”what if” questions based on an initial scenario. Now let A_i be the initial state vector. The new state is calculated repeatedly with the aggregation operator until the system converges $\left|A_i^t - A_i^{t-1}\right| < \varepsilon$. Eventually we get the equilibrium vector, which provides a set of answers to our ”what-if” questions. The PCM approach can be used in every area covered by the FCM approach.

3 COMPONENTS OF THE PCM

Now we will introduce the components of the Pliant Cognitive Maps.

3.1 Aggregator Operator

Besides the logical operators constructed in fuzzy theory, a non-logical operator also appears. The reason for this is the insufficiency of using either conjunctive or disjunction operators for real-world situations (Zimmermann and Zysno, 1980). The rational form of an aggregation operator is (Dombi, 1982):

$$a(x_1,\ldots,x_n) = \frac{1}{1+\frac{1-\nu_0}{\nu_0}\cdot\left(\frac{\nu}{1-\nu}\right)^{\sum w_i-1}\cdot\prod_{i=1}^{n}\left(\frac{1-x_i}{x_i}\right)^{w_i}}$$

We can model conjunctive and disjunctive operators with the aggregation operator. If v is close to 0 then the operation has a disjunctive characteristic and if v is close to 1 then the operation has a conjunctive characteristic. From this property it can be seen that by using aggregation we have more possibilities than by simply using the sum function in FCM. By varying the neutral values at the nodes, different operations can be performed.

3.2 Creating Influences

In the Pliant Cognitive Map, we define influences. The sigmoid function naturally maps the values to the (0,1) interval. Positive (negative) influences can be built with the help of two sigmoid functions and the conjunctive operator. Hence we get the generalized positive impulse function

$$c(t,u,v,a,b) = \frac{1}{1+u*e^{-\lambda_1(t-a)}+v*e^{-\lambda_2(t-b)}},$$

where u and v are weights. In Figure 2 we can see a basic influence, as mentioned in (Jozsef Dombi, 2005). If the influence is neutral, we can represent it by a 1/2 value. If there are no influences, then we continuously order 1/2 values in the system. If we want to model positive influences, we order a value which is larger than 1/2, and whose maximal value is 1. The negative influence is the negation of the positive influence. To create these influences, we use the following transformations:

$$P(t,u,v,a,b) = \frac{1}{2}\left(1+c(t,u,v,a,b)\right)$$

$$N(t,u,v,a,b) = \frac{1}{2}\left(1-c(t,u,v,a,b)\right)$$

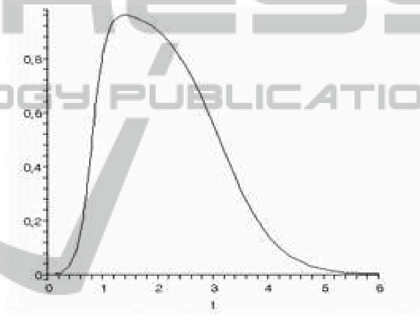

Figure 2: An average influence.

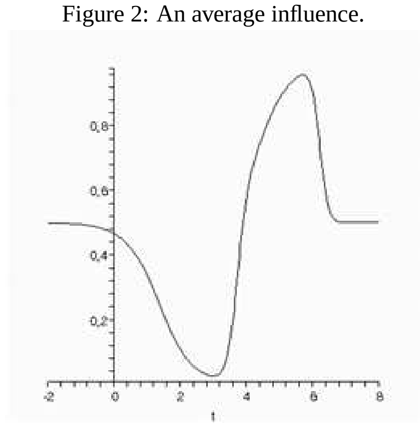

Figure 3: The aggregation of two influences.

In Figure 3, we have plotted the aggregation of a positive and a negative effect.

4 CONSTRUCTION OF THE PCM

To simulate the system, the only thing we have to do is to aggregate the influences. The aggregation operator is a guarantee that we will use influences in the right way. The following steps should be carried out to simulate the system:

1. Collect the concepts.
2. Define the expectation values of the nodes (i.e. threshold values of the aggregations).
3. Build a cognitive map (i.e. draw a directed graph for the concepts).
4. Define the influences (i.e. whether they are positive or negative)

In the iterative method we

1. Use the proper function or supply a data list for the input nodes.
2. Calculate the positive and negative influences using step 4.
3. Aggregate the positive and negative influences, where the aggregation parameter ν_0 value is the previous value of the concept C_j.

5 HEAT EXCHANGER APPLICATIONS

A heat exchanger is a standard device in the chemical and process industries (M. Fischer and Isermann, 2000). In this task, the temperature control is still a major challenge as the heat exchanger is used over a wide range of operating conditions. The behaviour of the system, which is non-linear, strongly depends on the flow rates and on the temperature of the medium. A cross-flow water/air heat exchanger is envisaged, which is subject to immeasurable or non-modeled disturbances that require the use of knowledge-based techniques. In this problem, we wish to develop a behavioural model for the heat exchanger system, that will control the water outlet temperature by manipulating the flow rate of the air.

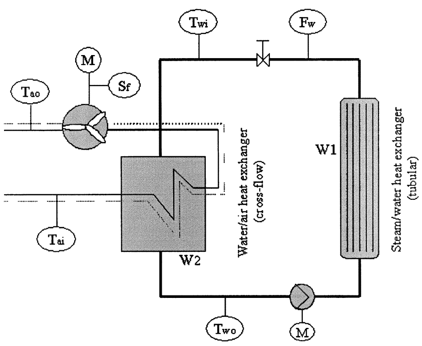

Figure 4: Typical heat exchanger system.

Here, in Figure 4 we can see a typical set-up for the system. It is well known that the FCM can be used to model and control the heat exchanger process (Stylios and Groumpos, 2004). In most process industries the thermal plant comprises two heat exchangers, but in our example (see Figure 2) we will just consider the secondary circuit. Our system contains two circuits W_1 and W_2. W_1 is a circuit, which is a tubular steam/water heat exchanger, and W_2 is the cross-flow water/air exchanger. The water in this circuit is heated by means of W_1. On the left hand side, the water is cooled in the cross-flow water/air heat exchanger W_2. A fan sucks in cold air from the environment (temperature Tai). After passing through the heat exchanger and the fan, the air is blown back out into the environment. The water temperature Two is controlled by manipulating the fan speed Sf. The control variable Two depends on the manipulated variable Sf and the measurable disturbances: the inlet water temperature Twi, air temperature Tai and water flow rate Fw. In most systems, the water flow rate is usually regulated by a PI-controlled pneumatic valve which strongly influences the behaviour of the heat exchanger $W2$ and it is a major challenge to design a temperature controller for Two when the flow rates vary over a wide range (Bittanti and Piroddi, 1997), (Ernst and Hecker, 1996). The operators of the heat exchanger gather experience that can be used to build a model. To construct an FCM system, we have to determine the concepts. Here, concepts stand for the input and output variables of the process. Earlier the thermal plant was described, and the concepts of the FCM were derived from a Stylios analysis (Stylios and Groumpos, 2004). Experts define five concepts for this situation:

- Concept1: The fan speed Sf, which is the manipulated variable.
- Concept2: The water flow rate Fw.
- Concept3: The water inlet temperature Twi.
- Concept4: The air inlet temperature Tai. The environmental temperature cannot be manipulated as it depends on the weather and season.
- Concept5: The water outlet temperature Two, which is the output of the model.

In the next step, the causal interconnections for the concepts have to be determined. Experts can describe the relation between any two concepts based on the system. The connections for the concepts are

- Linkage1: It connects concept1 (fan speed Sf) with concept5 (water outlet temperature Two). When the value of Sf increases, the value of Two decreases.
- Linkage2: It connects concept2 (flow rate Fw) with concept5 (water outlet temperature Two).

When the value of Fw increases, the value of *Two* increases.

- Linkage3: It connects concept2 (flow rate Fw) with concept1 (fan speed Sf). When the value of Fw increases, the value of Sf increases.
- Linkage4: It connects concept3 (water inlet temperature *Twi*) with concept5 (water outlet temperature *Two*). When the value of *Twi* increases, the value of *Two* increases.
- Linkage5: It connects concept3 (water inlet temperature *Twi*) with concept1 (fan speed Sf). When the value of *Twi* increases, the value of Sf increases.
- Linkage6: It connects concept3 (water inlet temperature *Twi*) with concept2 (flow rate Fw). When the value of *Twi* increases, the value of Fw decreases.
- Linkage7: It connects concept4 (air inlet temperature *Tai*) with concept5 (water outlet temperature *Two*). When the value of *Tai* increases, the value of *Two* increases.
- Linkage8: It connects concept4 (air inlet temperature *Tai*) with concept1 (fan speed Sf). When the value of *Tai* increases, the value of Sf decreases.
- Linkage9: It connects concept5 (water outlet temperature *Two*) with concept2 (flow rate Fw). When the value of *Two* increases, the value of Fw decreases.
- Linkage10: It connects concept5 (water outlet temperature *Two*) with concept1 (fan speed Sf). When the value of *Two* increases, the value of Sf increases.

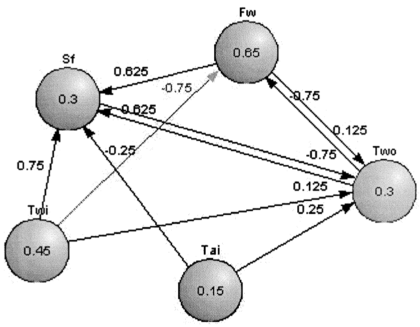

Figure 5: The FCM model for the heat exchanger system.

Figure 5 shows our system which describes, models and controls the heat exchanger system. The FCM model for the heat exchanger is in accordance with the models and experiments described in (M. Fischer and Isermann, 2000) (Ernst and Hecker, 1996). It is also possible to create an influence matrix for the system like that in Table 1.

Table 1: The weighted matrix of the model.

Sf	Fw	*Twi*	*Tai*	*Two*
0	0.625	0.75	-0.25	0.625
0	0	-0.75	0	-0.75
0	0	0	0	0
0	0	0	0	0
-0.75	0.125	0.25	-0.75	0

Table 2: Simulation results for the FCM.

Iteration	Sf	Fw	*Two*	Improvement:
1	0,77	0,52	0,56	0,8577
2	0,85	0,44	0,54	0,1906
3	0,85	0,43	0,51	0,0419
4	0,85	0,43	0,5	0,0107
5	0,85	0,43	0,5	0,0036

To simulate a real environment in the FCM, the values of concepts correspond to real measurements that have been transformed to the interval [0,1]. A corresponding mechanism is needed that will transform the measures of the system to their representative values of concepts in the FCM model and vice versa. The initial measurements of the heat exchanger system were transformed to concept values and the initial vector of the FCM is $A_0 = [0.3\ 0.65\ 0.45\ 0.15\ 0.3]$. In Figure 5, it also shows the initial values of each concept and the interconnections with their weights. With these initial values of concepts, the FCM starts to simulate the behaviour of the process. For the FCM area, a running step is defined as the time step during which the values of the concepts are calculated. The value of each concept is defined by taking all the causal linkage weights pointing to this concept and multiplying each weight by the value of the concept that causes the linkage, and adding the last value of each concept. Then the sigmoid function with $\lambda = 1$ is applied and the result falls into the range [0,1]. The results of the simulation are listed below.

This table does not contain input node values where the value is the same all time. Evaluating the results, we see that the value of fan speed Sf has increased, the value of flow rate Fw has decreased, and after the third step, the water outlet temperature *Two* falls below the value of 0.50. We also see that the value between any two simulation steps decreases, but this decrease is not uniform, which is not as good as we first thought. Because concepts control physical devices, we should vary the values as smoothly as possible.

6 EVALUATE WITH THE PCM

Our method works with real data measurements, which means that we do not need to transform real values between [0,1] in order to run the simulation. In our model we use the same concepts and connections, and the initial values of the concepts are the same as before. So first of all to evaluate our method we need to identify the range values of the concepts. Because previous articles do not mention these values, we will choose the following values:

Table 3: The range values of the concepts.

Concept	Minimum	Maximum	Default
Sf	100	500	250
Fw	2	20	6
Twi	20	50	30
Tai	18	35	24
Two	20	40	30

The default value is used to specify the real values of the first step. Using the above-mentioned ranges we define a sigmoid function that will be used for the calculations. For example, the initial value of the Sf concept is 0.3, and we apply the following sigmoid function:

$$S_f(x) = \frac{1}{1+e^{-\frac{4}{500}(x-250)}}$$

In this case, the real value of Sf should be 144.08. Based on these calculations, we can compute the initial values of the concepts:

Table 4: The initial values of the concepts.

Sf	Fw	Twi	Tai	Two
144.08	3.7	27.5	8.8	21.52

This method also shows that with each simulation step it is easy to recover the real value. In the classical FCM method, the influence does not change during the simulation. In order to compare it with our method, we will also define a constant influence. Hence in a simulation step we calculate the new concept value in the following way. For each node in the set we create a set that contains all the incoming nodes. Also, we will use the following expression to calculate the strength of the incoming node:

$$\left(\frac{1-x_i}{x_i}\right)^{w_{ij}},$$

where x_i is the actual value of the node and w_{ij} is the value of the influence between concept C_i and C_j. After, we calculate this method for each node in the set, then we use the aggregation operator to calculate the new value of the concept. For example, the new value of Fw is obtained by computing

$$\frac{1}{1+\left(\frac{1-0.45}{0.45}\right)^{-0.75}\left(\frac{1-0.3}{0.3}\right)^{-0.75}}$$

Now we can run the simulation until it exceeds a limit. The following table shows the results of our simulation.

Table 5: Simulation results for the PCM.

Iteration	Sf	Fw	Two	Improvement:
1	0.53	0.68	0.75	0.7262
2	0.81	0.33	0.59	0.7870
3	0.52	0.46	0.31	0.7024
4	0.42	0.67	0.57	0.5787
5	0.71	0.47	0.67	0.5881
6	0.66	0.40	0.42	0.3810
7	0.46	0.59	0.46	0.4327
8	0.60	0.56	0.63	0.3492
9	0.69	0.43	0.53	0.3310
10	0.54	0.51	0.44	0.3176
11	0.54	0.58	0.56	0.1951
12	0.65	0.48	0.57	0.2203
13	0.60	0.47	0.47	0.1590
14	0.54	0.55	0.51	0.1829
15	0.61	0.52	0.57	0.1602
16	0.63	0.47	0.51	0.1224
17	0.56	0.52	0.49	0.1266
18	0.58	0.53	0.55	0.0833
19	0.62	0.49	0.54	0.0908
20	0.59	0.50	0.50	0.0734
21	0.57	0.53	0.52	0.0712
22	0.60	0.51	0.54	0.0675
23	0.60	0.50	0.52	0.0405
24	0.58	0.52	0.51	0.0462
25	0.59	0.52	0.54	0.0337
26	0.60	0.50	0.53	0.0380
27	0.59	0.51	0.51	0.0336
28	0.59	0.52	0.53	0.0256
29	0.60	0.51	0.53	0.0264
30	0.60	0.51	0.52	0.0174
31	0.59	0.51	0.52	0.0191
32	0.59	0.51	0.53	0.0155
33	0.60	0.51	0.52	0.0148
34	0.59	0.51	0.52	0.0141
35	0.59	0.51	0.53	0.0084

Evaluating the results we notice that they are different from those got by using the FCM method. In Figure 6, we see how the values vary.

The value of the fan speed Sf has decreased, the value of flow rate Fw has decreased and the water outlet temperature Two is below the value of 0.50. We also see that the differences between two simulation steps decreases (see Figure 7), but this decrease

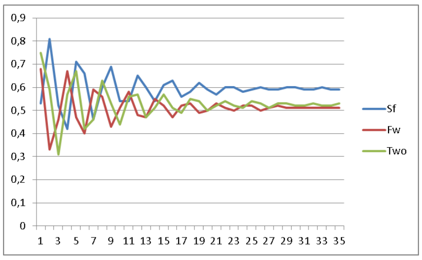

Figure 6: Results for the PCM model.

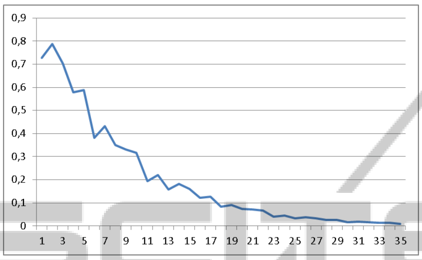

Figure 7: Sum of value change for each step.

is smooth, and this is why it requires more simulation steps.

7 CONCLUSIONS

In this paper, we made use of numerical methods to model complex systems based on positive and negative influences. This concept is similar to the FCM concept, but the functions and the aggregation procedures are quite different. It is based on a continuous-valued logic and all the parameters have a semantic meaning. Here, we showed that we can apply this method in a real environment. The values of the simulation steps smoothly decrease, but it requires more simulation steps. In this example, we used the same influence for each concept all the time, but it is also possible to vary the strength of the influence, and then we can model a real-world situation better.

REFERENCES

Axelrod, R. (1976). *Structured of Decision: the cognitive maps of political elites*. Princeton University press, New Jersey.

Bittanti, S. and Piroddi, L. (1997). Nonlinear identification and control of a heat exchanger: a neural network approach. *J. Franklin Inst.*, page 135153.

Dombi, J. (1982). A general class of fuzzy operators, the De Morgan class of fuzzy operators and fuzziness measures induced by fuzzy operators. *Fuzzy Sets and Systems*, 8:149–163.

Ernst, E.-J. and Hecker, O. (1996). Predictive control of a heat exchanger. *Proc. Seminar Theory Applications Model Based Predictive Control*, pages 1–18.

Glykas, M. (2010). *Fuzzy Cognitive Maps Advances in Theory, Methodologies, Tools and Applications*. Springer-Verlang.

J. Dombi, J. D. D. (2005). Pliant cognitive map using conjunctive operator. *Proceedings of IEEE International Workshop on Soft Computing Applications*, page 11.

Jozsef Dombi, J. D. D. (2005). Cognitive maps based on pliant logic. *16TH European Simulation Symposium*, page 63.

Kosko, B. (1986). Fuzzy entropy and conditioning. *Information Sciences*, 40:165–174.

Kosko, B. (1992). *Neural Networks and Fuzzy Systems*. Prentice Hall, Englewood Cliffs.

Kosko, B. (1994). *Fuzzy Thinking*. Flamingo. hardcopy.

M. Fischer, O. N. and Isermann, R. (2000). Knowledge-based adaptation of neurofuzzy models in predictive control of a heat exchanger. *Soft Computing and Intelligent Systems*, page 469489.

Maio, C. D., Fenza, G., Gaeta, M., Loia, V., and Orciuoli, F. (2011). A knowledge-based framework for emergency DSS. *Knowl.-Based Syst*, 24(8):1372–1379.

Salmeron, J. L. and Lopez, C. (2012). Forecasting risk impact on ERP maintenance with augmented fuzzy cognitive maps. *IEEE Trans. Software Eng*, 38(2):439–452.

Salmeron, J. L., Vidal, R., and Mena, A. (2012). Ranking fuzzy cognitive map based scenarios with TOPSIS. *Expert Syst. Appl*, 39(3):2443–2450.

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

Yaman, D. and Polat, S. (2009). A fuzzy cognitive map approach for effect-based operations: An illustrative case. *Inf. Sci*, 179(4):382–403.

Zimmermann, H. J. and Zysno, P. (1980). Latent connectives in human decision making. *Fuzzy Sets and Systems*, 4(1):37–51.