

Dud.

137

114 706

IEEE – SOFA 2005

106

← Gyöngyi

419

← Sora

IEEE International Workshop on Soft Computing Applications

PROCEEDINGS

Organized by:

IEEE I&M Society TC 22 on Intelligent Measurement Systems
IEEE Hungary Section
IEEE Romania Section
EUROFUSE
Hungarian Fuzzy Association
Romanian Society of Control Engineering and Technical Informatics
The General Association of Engineers in Romania – Arad Section
Budapest University of Technology and Economics
“Aurel Vlaicu” University from Arad
Szent István University, Budapest
Budapest Tech

27–30 August, 2005
Szeged – Hungary and Arad – Romania

Pliant Cognitive Map using conjunctive operator

József Dombi, József D. Dombi

Department of Informatics

University of Szeged

Szeged, Hungary

e-mail: dombi@inf.u-szeged.hu

Abstract – We present a tool to describe and simulate dynamic systems based on positive and negative influences. Our starting point is aggregation. With proper transformations of the sigmoid function and using the conjunctive operator we build positive and negative effects. From the input we calculate the output effect with the help of the aggregation operator. This algorithm is comparable with the concept of fuzzy cognitive maps.

Keywords – Aggregation operator, Cognitive Map, Dombi Operator, Pliant concept

I. INTRODUCTION

we usually face serious difficulties when we handle sophisticated dynamic systems. Developing a model requires effort and specialized knowledge. Usually a system involves complicated causal chains, which might be nonlinear. It should also be mentioned, that numerical data may be hard to get, they can be even uncertain. Using a dynamic system model can be hard computationally. Fuzzy Cognitive Map (FCM) approach overcomes the above mentioned difficulties. FCM was proposed by Kosko [1], [2], and is a hybrid method that lies in some sense between fuzzy systems and neural networks [3]. Knowledge is represented in a symbolic manner using states, processes and events. All type of informations have numerical values. FCM allows us to perform qualitative simulations and experiment with a dynamic model. FCM has better properties than expert systems, since it is relatively easy to use them to represent structured knowledge and the inference can be computed by numeric matrix operation instead of applying rules. Our solution is similar to FCM. It is called Pliant Cognitive Map (PCM). It is based on a qualitative description of the influences, i.e. it is enough to know a rough description of the system. The main difference between PCM and FCM is the replacement of matrix multiplication by fuzzy operators.

II. SHORT DESCRIPTION OF FCM

In FCM the causal relationship is expressed by either positive or negative signs ordered by different weights. As we mentioned this will be replaced by unary operators in PCM. Let $\{C_1, \dots, C_m\}$ be concepts. Define a directed graph over

the concepts. We assign a weight $w_{ij} \in [0, 1]$ to the edge directed from concept C_i to concept C_j . The weight measures the influence of C_i on C_j . If the $w_{ij} = 1/2$ then w_{ij} is called neutral value. If $w_{ij} = 0$ then we say it is a maximum negative influence, if $w_{ij} = 1$ then we say it is a maximal positive influence or causality (In FCM $w_{ij} \in [-1, 0, 1]$):

- $w_{ij} > 1/2$ indicates direct (positive) causality between concepts C_i and C_j . That is the increase (decrease) in the value of C_i leads to increase (decrease) on the value of C_j .
- $w_{ij} < 1/2$ indicates inverse (negative) causality between concepts C_i and C_j . That is the increase (decrease) in the value of C_i leads to decrease (increase) on the value of C_j .
- $w_{ij} = 1/2$ indicates C_i and C_j are neutral to each other.

In the pliant case w_{ij} depends on time (t), i. e.

$$w(t) = (w_{ij}(t))_{n \times n} \quad (1)$$

The activation level a_i of concept C_i is calculated by an iteration process. In FCM

$$a_i^{(n)} = f \left(\sum_{i=1}^n w_{ij} a_i^{(n-1)} \right) \quad (2)$$

where $a_i^{(n)}$ is the new activation level of concept C_i at time $t+1$, a_i^0 is the activation level of concept C_i at time t and f is the threshold function. FCM has the advantage that we obtain the new state vector by multiplying the previous state vector by the edge matrix W that shows the effect of the change in the activation level of one concept to another concept.

III. BASIC CONCEPT OF PCM

In this paper we modify the concept of FCM. We use cognitive maps to represent knowledge and to model decision making, which was introduced by Axelrod [4].

The cognitive map describes the whole system by a graph showing the cause effects along concepts. It is a directed graph with feedback, that describes the concepts of the world and the casual influences between the concepts. From logic point of view the causal concepts are unary operators of a continuous

valued logic containing negation operators in the case of inhibition effects. The value of a node reflects the degree of the activity of the system at a particular time. Concept values are expressed on a normal $[0, 1]$ range.

Kosko used fuzzy values and matrix multiplication to calculate the next state of the systems.

We drop the concept of matrix multiplication. On one hand matrix multiplication does not suit well in continuous logic (or fuzzy logic), where the truth value is 1 and the false is 0. On the other hand general operators are more effective.

Logic and the cognitive map model correspond to each other in the PCM. It is easier to build up a PCM. After we identified the PCM with the real world, extracting the knowledge is easy. Combination of cognitive maps with logic helps us extracting knowledge more efficiently opposed to the use of rule based systems. The classical knowledge representation in expert systems is made through a decision tree. This form of knowledge presentation in most cases cannot model the dynamic behavior of the world.

Instead of values, we use time dependent functions that are similar to impulse functions which represent positive and negative influences.

Values do not denote exact quantities, they denote the degree of activation. The inverse of normalization could express the values coming from the real world i.e. using sigmoid function. In spite of Fuzzy Cognitive Maps we do not use thresholds to force the values between 0 and 1. The mapping is a variation of the fuzzification process in fuzzy logic, furthermore it always destroys our desire to get quantitative results. In pliant logic we map the real world into logic. These maps are continuously strict monotonously increasing functions, and so the inverse of these functions yields the data of the real world.

In the pliant concept we aggregate the influences instead of summing up the values. The result always remains between 0 and 1, so we can avoid normalization as an additional step. The aggregation in pliant logic is a general operation, which contains conjunctive operators and disjunctive operators as well. Depending on the parameter called neutral value of the aggregation operator we build logical operators (Dombi operators). Using PCM (Pliant Cognitive Maps) we answer what if questions based on an initial scenario. Let a_0 be the initial state vector. The new state is calculated repeatedly with the aggregation operator until the system converges (i. e. $|a_i^0 - a_i^n| < \epsilon$). We obtain the resulting equilibrium vector, which provides the answer to our what-if questions. The PCM can be used in all areas covered by FCM.

IV. COMPONENTS OF PCM

A. Negation

The properties of negation are:

- defined on $(0, 1)$ and the values are also in $(0, 1)$
- $n(0) = 1$
- $n(1) = 0$
- continuous
- strictly decreasing function

- involutive, $n(n(x)) = x$

We can assign a fixed point ν_* to a negation so that $n(\nu_*) = \nu_*$ or we can define a value ν_0 called common threshold and neutral value so that $n(\nu) = \nu_0$. Usually $\nu_0 = 1/2$.

The corresponding negation function is

$$n_\nu(x) = \frac{1}{1 + \frac{1-\nu_0}{\nu_0} \frac{1-\nu}{\nu} \frac{x}{1-x}}, \tag{3}$$

$$n_{\nu_*}(x) = \frac{1}{1 + \left(\frac{1-\nu_*}{\nu_*}\right)^2 \frac{x}{1-x}}. \tag{4}$$

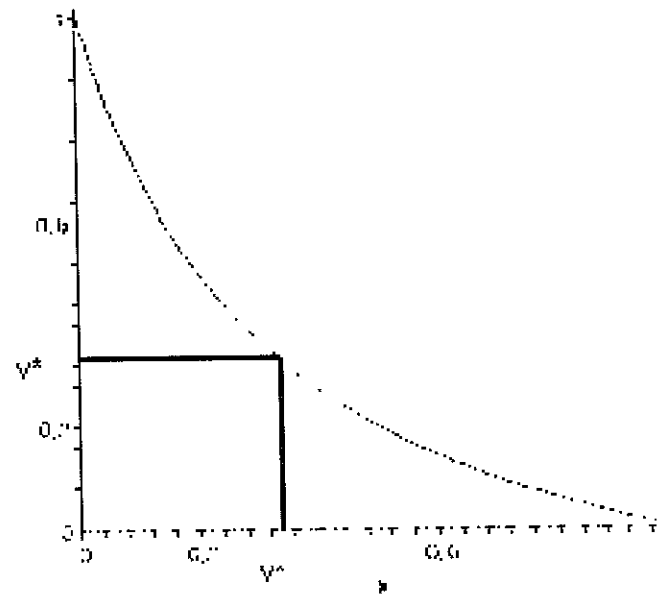


Fig. 1. Negative function with ν_* threshold

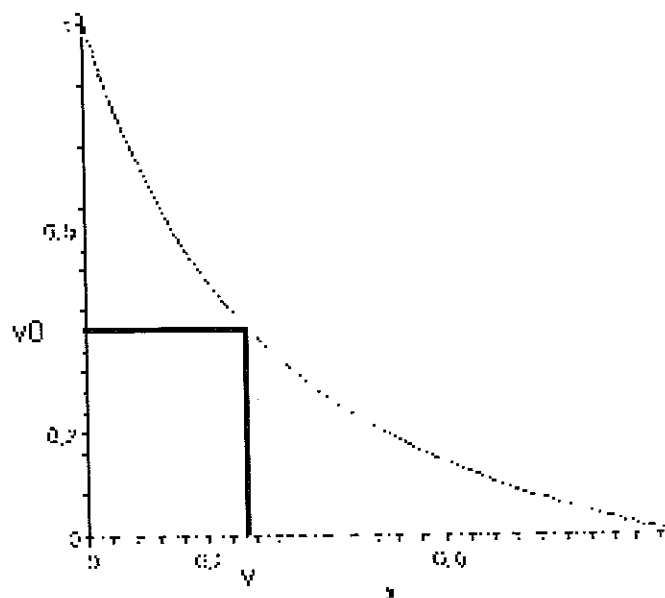


Fig. 2. Negative function with ν neutral value and ν_0 threshold

B. Conjunctive operator

Using an impulse function we can create positive and negative influences. Our task is to produce this function. The function is defined in time line and maps to [0, 1]. To construct this function we use a fuzzy operator and sigmoid function. It is important to consider that the fuzzy operator and sigmoid function fits well, accordingly we use Dombi operator. The associative function equation is the following:

$$c(x, y) = f^{-1}(f(x) + f(y)) \tag{5}$$

The function of Dombi operator:

$$f(x) = \frac{1-x}{x}, f^{-1}(x) = \frac{1}{1+x} \tag{6}$$

If we use Dombi operator in the conjunction function we get the following equation:

$$c(x, y) = \frac{1}{1 + \frac{1-x}{x} + \frac{1-y}{y}} \tag{7}$$

If we use λ power in the generator function we get another generator function:

$$c_\lambda(x, y) = f^{-1}\left(\left(f^\lambda(x) + f^\lambda(y)\right)^{\frac{1}{\lambda}}\right) \tag{8}$$

Using the Dombi operator we get the powered conjunction in the following form:

$$c(x, y) = \frac{1}{1 + \left(\left(\frac{1-x}{x}\right)^\lambda + \left(\frac{1-y}{y}\right)^\lambda\right)^{\frac{1}{\lambda}}} \tag{9}$$

If we pay attention to the general form of the weight we can generalize the formula. In this case we lose the associativity property, however we get other good properties, which are very useful (i.e. idempotency). Let u and v be the weights, with the following qualities:

$$0 \leq u \leq 1, 0 \leq v \leq 1 \quad u + v = 1 \tag{10}$$

With these restriction we get the generalized conjunction:

$$c_\lambda(u, x; v, y) = \frac{1}{1 + \left(u\left(\frac{1-x}{x}\right)^\lambda + v\left(\frac{1-y}{y}\right)^\lambda\right)^{\frac{1}{\lambda}}} \tag{11}$$

C. Aggregation and its properties

Beside the developed logical operators 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 [5]. A general class of the fuzzy operators is called t-norm and t-conorm (disjunctive case). Denoting by $c(x, y)$ the conjunctive operator and $d(x, y)$ the disjunctive operator, which belong to the t-norm class we have

$$c(x, y) \leq \min(x, y), d(x, y) \geq \max(x, y)$$

It often occurs in real world situation that the an aggregation value $a(x, y)$ is

$$\min(x, y) \leq a(x, y) \leq \max(x, y)$$

The rational form of an aggregation operator [6], [7] is:

$$a(x_1, \dots, x_n) = \frac{1}{1 + \frac{1-\nu_0}{\nu_0} \left(\frac{\nu}{1-\nu}\right)^n \prod_{i=1}^n \frac{1-x_i}{x_i}} \tag{12}$$

or

$$a(x_1, \dots, x_n) = \frac{1}{1 + \left(\frac{\nu_*}{1-\nu_*}\right)^{n-1} \prod_{i=1}^n \frac{1-x_i}{x_i}} \tag{13}$$

where ν is the neutral value or previous value of the node (see later), and ν_* is the eigenvalue of the negation.

The aggregation operator is axiomatically based and it has several good properties as:

- defined on (0, 1) and the values are also in (0, 1)
- associativity
- strictly monotonously increasing
- continuous on [0, 1) interval
- $a(0, 0) = 0$ and $a(1, 1) = 1$

Aggregation is connected to negation operators. The negation function and the aggregation operator are closely related. It can be easily seen that:

- $n(a(x, y)) = a(n(x), n(y))$
- $a(x, n(x)) = \nu_0$
- $a(x, \nu_0) = x$

These properties of the aggregation are natural:

- Aggregating positive values and negating is the same as aggregating negative values
- By aggregating a positive and its negated value we get the neutral values back
- Aggregating x with the neutral value we get back x .

The property ensures that we can replace the sum function with the aggregation function of the pliant logic. The neutral value here is ν_0 instead of 0 which used in FCM. The aggregation connects the logical connectives with the help neutral value. It can be proved that

- 1), if $x, y \leq \nu$, then $a(x, y) \leq \min(x, y)$
- 2) if $x, y \geq \nu$, then $a(x, y) \geq \max(x, y)$
- 3) if $x \leq \nu \leq y$, then $\min(x, y) \leq a(x, y) \leq \max(x, y)$

1) means that if the values are less than the neutral value then the aggregation is a conjunction.

2) means that if the values are larger than the neutral value then the aggregation is a disjunction.

3) means that the aggregation of positive and negative values always lies between the two values.

We can model conjunctive and disjunctive operators with the aggregation operator. If ν is close to 0 then the operation has a disjunctive characteristic and if ν 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 using the sum function in FCM. By changing the neutral values at the nodes different operations can be carried out.

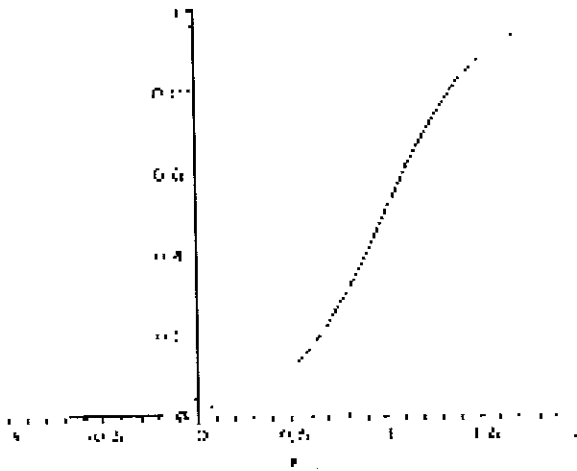


Fig. 3. Sigmoid function

D. Sigmoid function

Our starting point is the sigmoid function:

$$\sigma_a^{(\lambda)}(t) = \frac{1}{1 + e^{-\lambda(t-a)}} \tag{14}$$

It is easy to see that:

- $\sigma_a^{(\lambda)}(a) = 1/2 (= \nu_0)$
- $\sigma_a^{(\lambda)'}(a) = \frac{\lambda}{4}$
- $\sigma_a^{(-\lambda)}(t) = 1 - \sigma_a^{(\lambda)}(t)$.

E. Value transformation

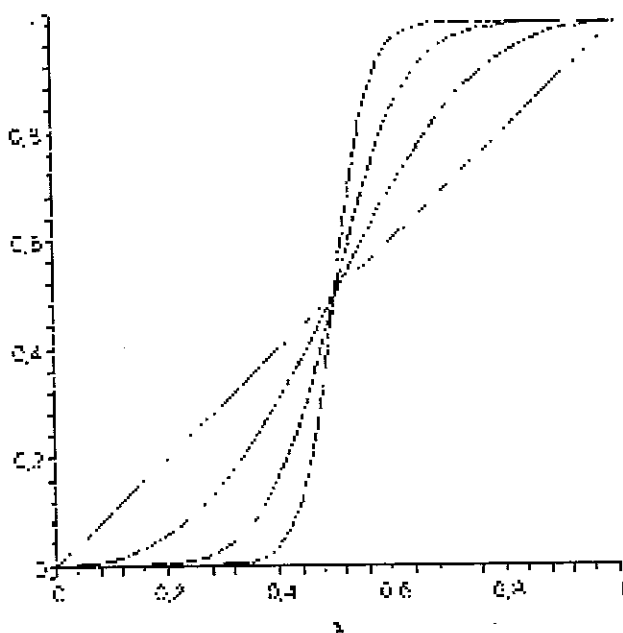


Fig. 4. Modification operator with the parameter $\nu = 1/2$, $\lambda = 1, 2, 4, 8$

To complete the Pliant Cognitive Map we have to give a unary transformation on the values produced by the aggregation. Using pliant logic, the general form of the modification operator is:

$$\kappa_{v_{ij}}^{(\lambda_{ij})}(x) = \frac{1}{1 + \left(\frac{v_{ij} - x}{1 - v_{ij}}\right)^{\lambda_{ij}}} \tag{15}$$

The sign of λ_{ij} means that the influence is positive or negative, the value of λ_{ij} is the sharpness and v_{ij} is the actual value of C_j .

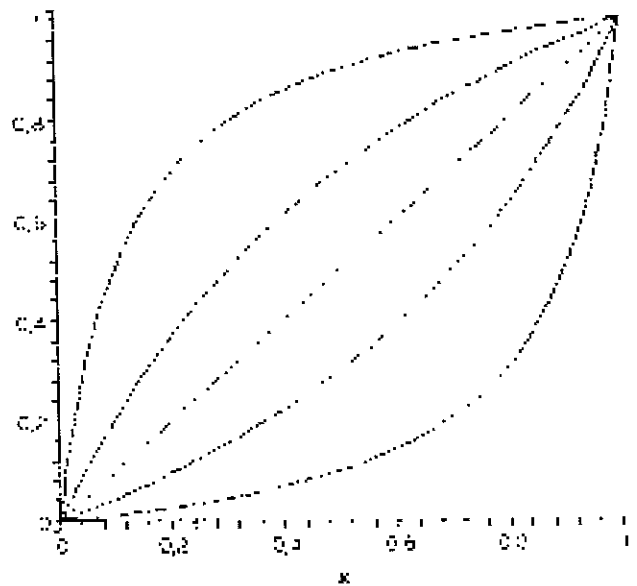


Fig. 5. Modification operator with the parameter $\lambda = 1, \nu = 1/10, 3/10, 5/10, 7/10, 8/10$

V. PRODUCING INFLUENCES

The sigmoid function naturally maps the values to the (0,1) interval. Positive (negative) influences can be built with the help of $\sigma_a^{\lambda_1}(x)$, $\sigma_b^{\lambda_2}(x)$ and the conjunctive operator, where $\lambda_1 > 0$, $\lambda_2 < 0$ and $a < b$. It is worth seeing that if we use $\frac{1-x}{x}$ and $x = \sigma_a^{(\lambda)}(t)$ what we get:

$$\begin{aligned} \frac{1-x}{x} &= \frac{1 - \sigma_a^{(\lambda)}(t)}{\sigma_a^{(\lambda)}(t)} = \\ &= \frac{1 - \frac{1}{1 + e^{-\lambda(t-a)}}}{\frac{1}{1 + e^{-\lambda(t-a)}}} = \frac{1 + e^{-\lambda(t-a)} - 1}{e^{-\lambda(t-a)}} = \frac{e^{-\lambda(t-a)}}{e^{-\lambda(t-a)}} = e^{-\lambda(t-a)} \end{aligned}$$

$$\begin{aligned} c_m(u, x; v, y) &= c_m(u, \sigma_a^{(\lambda_1)}(t); v, \sigma_b^{(\lambda_2)}(t)) = \\ &= \frac{1}{1 + e^{-\lambda_1(t-a)} + e^{-\lambda_2(t-b)}} \end{aligned}$$

Using weighted conjunction, it is worth to choose the following values $u = \frac{\lambda_2}{\lambda_1 + \lambda_2}$, $v = \frac{\lambda_1}{\lambda_1 + \lambda_2}$. So we get the generalized positive impulse function:

$$c_m(u, x; v, y) = \frac{1}{1 + \frac{\lambda_2}{\lambda_1 + \lambda_2} e^{-\lambda_1(x-a)} + \frac{\lambda_1}{\lambda_1 + \lambda_2} e^{-\lambda_2(x-b)}} \quad (16)$$

If the influence is neutral, we represent it by 1/2 (= ν_0) value. If there are no influences, then we continuously order 1/2 value to the system. If we want to model positive influences, we order a value, which is larger than 1/2, and maximal value is 1. The negative influence is the negation of the positive

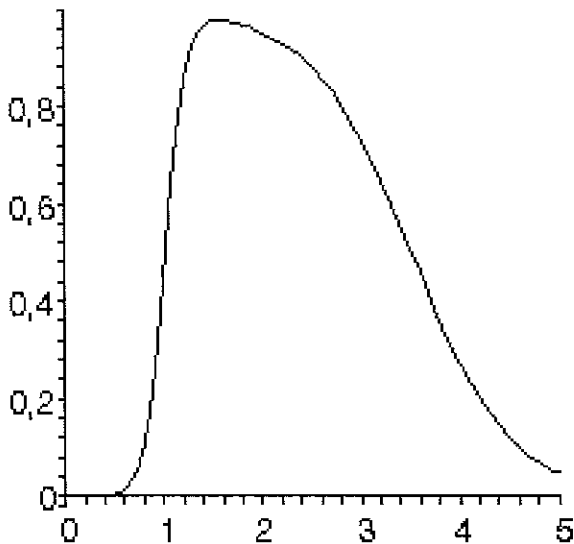


Fig. 6. Asymmetrical positive influence on [0, 1] $a = 1, b = 3.5, \lambda_1 = 10, \lambda_2 = 2$

influence. To create these influences we use the following transformations:

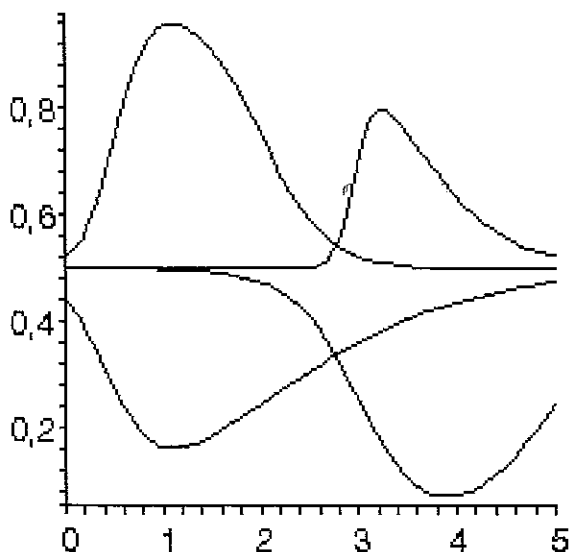


Fig. 7. Transformation of 2 Positive and 2 Negative influences

$$P(t) = \frac{1}{2}(1 + \sigma_{a_1, a_2}^{\lambda_1, \lambda_2}(t)), \quad (17)$$

$$N(t) = \frac{1}{2}(1 - \sigma_{a_1, a_2}^{\lambda_1, \lambda_2}(t)). \quad (18)$$

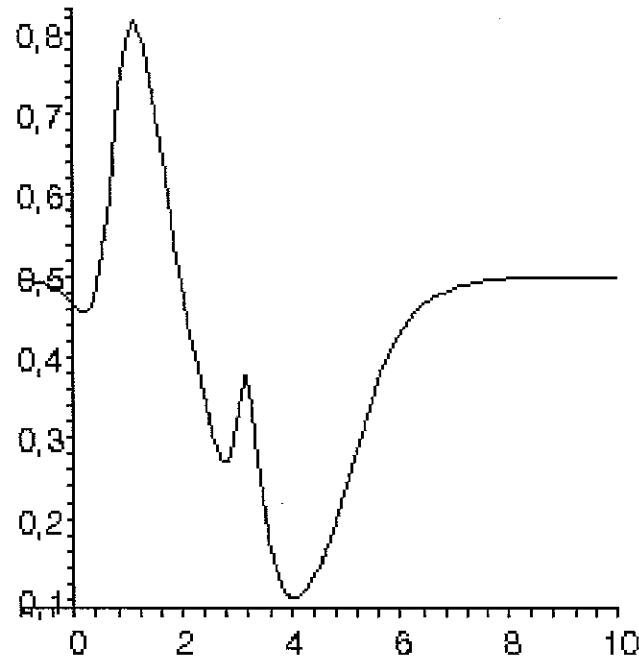


Fig. 8. Aggregation of the fig. 7 influences

The positive influences have four parameters $a_1, a_2, \lambda_1, \lambda_2$. These parameters have semantical meanings: a_1 is the starting point of the influence and a_2 is the end point of the influence, λ means the speed or intensity of the influence. It is also possible to set the maximal value of the influence (called the amplitude). This means that we have to multiply the value of the function with the amplitude.

To simulate the system the only thing we have to do is to aggregate the influences. The aggregation operator is a guarantee, that we use influences in the right manner.

If we know the real process, then our task is to divide the function values into positive and negative influences. It is an optimization problem. Starting the optimization we need initial values. This is the critical point the success of optimization. If we set it wrong, then the optimization method finds only the local optimum, but the shape of the real process can help us. We can find the starting and the end points of the influences, even we can estimate the λ parameters. These initial values guarantee that the optimization method with a high probability finds global optimum. This gives us a new method to analyze real process.

VI. APPLICATION

Our model consists of four interacting sector: population, food consumption, sorghum production and animal production. As the model contains approximately 200 variables, it is

impossible to discuss it in detail. Figure 9 presents a feedback diagram from a part of the model structure. We generate

meaning. We are working on a real world application and on an effective learning of the parameters of the system.

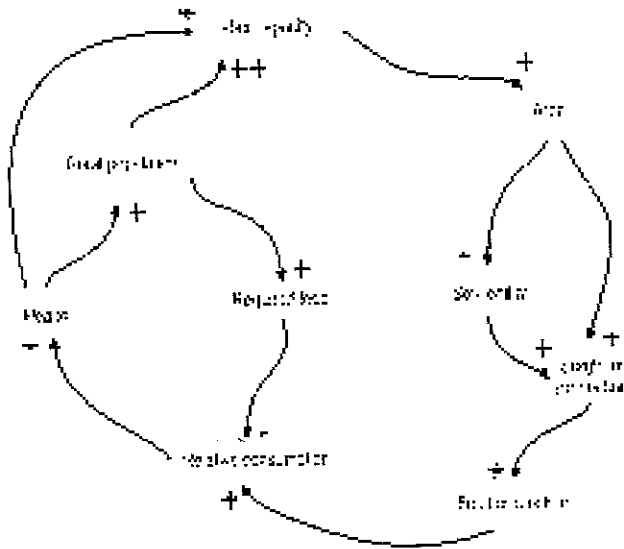


Fig. 9. Feedback diagram of the model structure

several input functions and calculate its effect on the nodes. The model works as it was expected. In the future we will work on an identification process based on historical data. We would like to find the proper parameters of the modification operators.

VII. SUMMARY OF CONSTRUCTION PCM

With the above mentioned procedure, it is easy to build PCM. The following steps should be carried out:

- 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 between the concepts).
- 4) Define the influences (i.e. are they positive or negative)

The iterative method:

- i) Use the proper function or give a timetable for the input nodes.
- ii) Calculate the positive and negative influences using step 4.
- iii) Aggregate the positive and negative influences, where the ν_* value is the previous value of C_j .

VIII. CONCLUSION

We propose a new type of numerical calculus to model complex systems based on positive and negative influences. This concept is similar to FCM, but the functions and the aggregation procedures are quite different. It is based on a continuous valued logic and all the parameters have semantic

REFERENCES

- [1] Kosko B., Ed., *A dynamic system approach to machine intelligence*, Neural networks and fuzzy Systems, 1992.
- [2] Kosko B., "Fuzzy cognitive maps," *Int. Journal of Man Machine Studies*, vol. 24, no. 1, pp. 65-75, Jan. 1986.
- [3] Dickerson J. Kosko B., Ed., *Fuzzy virtual worlds*, AI Expert, 1994.
- [4] Axelrod R., Ed., *Structured of Decision: the cognitive maps of political elites*, Princeton University press, 1976.
- [5] Zysno P. Zimmermann, H.-J., "Latent connectives in human decision making," *Fuzzy Sets and Systems*, vol. 4, no. 1, pp. 37-51, Jan. 1980.
- [6] Dombi J., "Basic concept for a theory of evaluation: the aggregation operator," *European Journal of Operations Research*, vol. 10, no. 1, pp. 282-293, Jan. 1982.
- [7] Dombi J., "A general class of fuzzy operators, the de morgan class of fuzzy operators and fuzziness measures induced by fuzzy operators," *Fuzzy Sets and Systems*, vol. 8, no. 1, pp. 149-163, Jan. 1982.