# COGNITIVE MAPS BASED ON PLIANT LOGIC

József Dombi and József D. Dombi
Departure of Informatics
University of Szeged
6720, Szeged Árpád tér 2
E-mail: dombi@inf.u-szeged.hu

**KEYWORDS**
Aggregation operators, continuous valued logic, Cognitive map

## ABSTRACT

In this paper we present a tool for description, and for simulation of dynamic system. Our starting point is the aggregation concept, which was developed for multicriteria decision making. Using a continuous logic operator and proper transformation of sigmoid function we build positive and negative effects. From the input with the aggregation operator we calculate the output effect. This algorithm is comparable with the concept of fuzzy cognitive maps. We show this new technique, which could be much more efficient then the FCM.

## INTRODUCTION

Handling sophisticated systems we face serious difficulties, because we have to approach dynamic systems. Modeling a dynamic system can be hard in a computational sense. In addition formulating a system with mathematical model may be difficult, even impossible.
Developing the model requires effort and specialized knowledge. Usually the system involves complicated causal chains, which may be nonlinear. It should be mentioned to, that numerical data may be hard to get and even uncertain.
Our approach overcomes the above mentioned difficulties. It is qualitative approach, where enough to know rough description of the system and not necessary deep expert knowledge.
First type of this approach proposed by Kosko (Kosko 1986; Kosko 1992; Kosko 1994), and called Fuzzy Cognitive Map (FCM). FCM's are hybrid methods lie in some sense between fuzzy systems and neural networks. Knowledge is represented in symbolic manner, states, processes and events. All type of this information has numerical values.
FCM allows to performing qualitative simulations and experiment width the model. Compared FCM either expert system and neural networks it has good properties as it is relative easy to use for representing structured knowledge and the inference can be computed by numeric matrix operation instead of applying rules.

## BASIC CONCEPT OF PLIANT COGNITIVE MAP (PCM)

In this paper we make closer the FCM concept to the real world modeling. We will use the cognitive maps as a formal way of representing knowledge and modeling decision making, which was introduced by Axelrod (Axelrod 1976). Kosko used fuzzy values and matrix multiplication to calculate the next stage of the concepts representing by cognitive map. Instead of values we will use time dependent functions like an impulse function representing the positive and negative influences.
Another improvement is dropping the matrix multiplication concept, because on one hand it not works using continuous logic (or fuzzy logic), where the truth value has 1 and the false is 0 as usually used and the negative effect build by negation. The other hand more general operators could be more effective.
Logic and the cognitive map model correspond each other, and much more easy for the expert to build up and construct the system and from the identified system extract the knowledge.
Combining cognitive maps with logic can avoid many of knowledge extraction problems instead of rule based system. The classical knowledge representation in expert system is made through a decision tree. This form of knowledge presentation can not model the dynamic behavior of the world.
The cognitive map describes the whole system by a graph showing the cause effects along concepts. It is a directed graph with feedback, that the world as a collection of concepts and causal influences between the concepts. From logic point of view the causal concepts are unary operators of a continuous valued logic, which may negation operators in the case of inhibition effects.
The value of node reflects the degree of the activity of the system at a particular time. Concept values are expressed on a normal range denoting a degree of activation rather than an exact quantities value. The inverse of the normalization could express the values coming from the real world. 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, but this destroys getting the possibility of quantitative results. In pliant logic we use only continuously strict monotonously increasing functions, and the inverse function always corresponds the real world values with the logical values.

*Proceedings 16th European Simulation Symposium*
*György Lipovszki, István Molnár © SCS Press, 2004*
*ISBN 1-56555-286-5(book) / ISBN 1-84233-106-x(CD)*

In FCM the causal relationship are expressed by either positive or negative signs ordered by different weights. As we mentioned this will be replaced by unary operators.

Let $\{C_1,\ldots C_m\}$ be concepts. Let define over the concepts a directed graph. A directed edge $w_{ij}$ from concept $C_i$ to concept $C_j$ measures how much $C_i$ causes $C_j$. $w_{ij} \in [0,1]$ where $\frac{1}{2}$ is the neutral value, 0 is maximum negative and 1 is maximal positive influence or causality (In FCM $w_{ij} \in [-1,0,1]$):

- $w_{ij} > \frac{1}{2}$ 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 $C_j$.
- $w_{ij} < \frac{1}{2}$ 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 $C_j$.
- $w_{ij} = \frac{1}{2}$ indicates no relationship between $C_i$ and $C_j$.

In pliant case $w_{ij}$ depends on time(t) i.e. $w(t) = (w_{ij}(t))_{n\times n}$ The activation level $a_i$ of concept $C_i$ calculated by an iteration process. In FCM is $a_i^n = f\left(\sum_{i=1}^{n} w_{ij} a_i^o\right)$ where $a_i^n$ is the new activation level of concept $C_i$ at time t+1, $a_i^o$ is the activation level of concept $C_i$ at time t and f is threshold function. FCM has the advantage that we get the new state vector a by multiplying the previous state vector a by the edge matrix W showing the effect of the change in the activation level of one concept to another concept. In the pliant concept we aggregate the influences instead of summing up the values. The result is always remaining between 0 and 1, so we can avoid to normalization as an artificial step. The aggregation is pliant logic is general operation, which contain the conjunctive operators and disjunctive operators as well. Depends on the parameter – called neutral value – of aggregation operator we can build logical operators (Dombi operators). Using PCM (Pliant Cognitive Maps) can be used to answer "what if" question based on an initial scenario. Let $a_0$ the initial state vector. Repeatedly calculate with the aggregation operator the new state until the system convergence (i.e. $\left|a_i^o - a_i^n\right| < \varepsilon$). We get the resulting equilibrium vector, which provides the answer to the "what-if" question. The PCM can be used all the areas covered by FCM.

**AGGREGATION AND ITS PROPERTIES**

Beside the developed logical operators in fuzzy theory appears a non logical operator. The reason was insufficiency of using either conjunctive or disjunction operators for real world situation [Zimmermann]. 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 then

$$c(x,y) \le \min(x,y)$$
$$d(x,y) \ge \max(x,y)$$

In the real world situation often occur that an aggregation value $a(x,y)$ is

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

The rational form of an aggregation operator is (Dombi 1982a):

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

or

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

Where $\nu$ is the neutral value or previous value of the node (see later).

The corresponding negation function is

$$n_\nu(x) = \frac{1}{1+\frac{1-\nu_0}{\nu_0}\cdot\frac{1-\nu}{\nu}\cdot\frac{x}{1-x}}$$

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

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

1. defined on (0,1) and the values are also in (0,1)
2. associativity

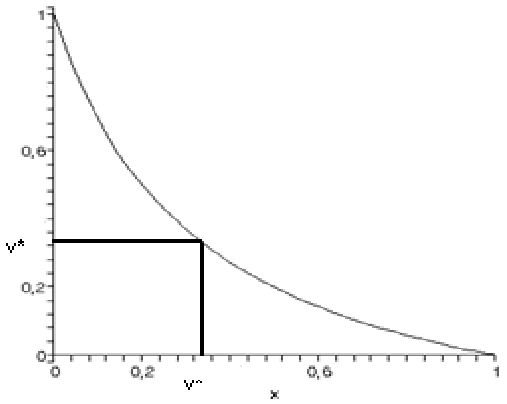
3. continuously
4. strictly monotonously increasing
5. continuous on [0,1) interval
6. a(0,0)=0 and a(1,1)=1

Aggregation is connected to negation operators. The proportions of that negation are

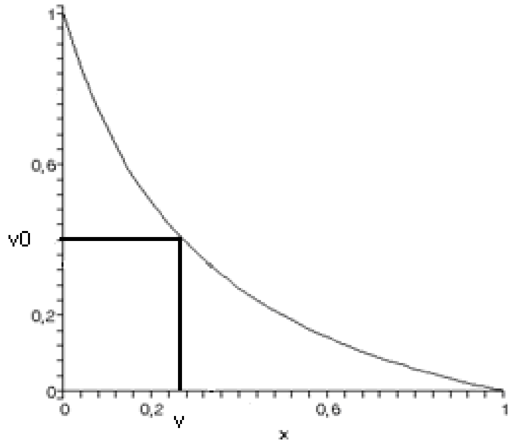
1. defined on (0,1) and the values are also in (0,1)
2. n(0)=1
3. n(1)=0
4. continuous
5. strictly decreasing function
6. involutiv (the double negation is the identity) n(n(x)=x

We can ordered to a negation a $\nu_*$ fixed point such $n(\nu_*) = \nu_*$ or defined $\nu_0$ the so called common threshold and neutral value than $n(\nu) = \nu_0$.

Usually $\nu_0 = \frac{1}{2}$




Figures 1: Negative function width $\nu_*$ threshold




Figures 2: Negative function width $\nu$ neutral value and $\nu_0$ threshold

The negative function and aggregation operator is closely related. It can be seen easily that

1. $n(a(x, y)) = a(n(x), n(y))$
2. $a(x, n(x)) = \nu_0$
3. $a(x, \nu_0) = x$

The properties of the aggregation are natural: 1, aggregating positive values and negating it is the same if we aggregating negative values; 2, aggregating positive and negative values we get the neutral values back; 3, 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 $\nu_0$ instead of 0 using in FCM. The neutral value corresponds the aggregation width the logical connectives to.
It can be proved that

1. if $x, y \le \nu$ than $a(x, y) \le \min(x, y)$
2. if $x, y \ge \nu$ than $a(x, y) \ge \max(x, y)$
3. if $x \le \nu \le y$ than

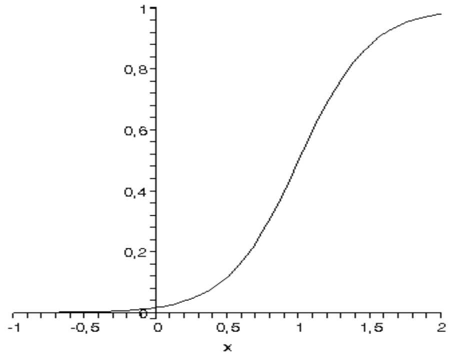
$$\min(x, y) = x \le a(x, y) \le y = \max(x, y)$$

First means that if the values are less then neutral value then the aggregation is conjunction.
Second means that if the values are larger then neutral value then the aggregation is disjunction.
Third means that the aggregation of positive and negative values is always between the two values.
We can model conjunctive and disjunctive operator with the aggregation operator. If $\nu$ is close to 0 then operation has disjunctive character and if $\nu$ is close to 1 then the operation has conjunctive character. From this property it can be seen, that using aggregation we have much more possibilities instead of using the sum function in FCM. Changing in the nodes the neutral value different operation can be carried out.

**PRODUCING INPUT INFLUENCES**

Our starting point is the sigmoid function.

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




Figures 3: Sigmoid function

It is easy to see:

1. $\delta_a^{(\lambda)}(a) = \frac{1}{2}(= \nu_0)$
2. $\delta_a^{'}(a) = \lambda$
3. $\delta_a^{(-\lambda)}(t) = 1 - \delta_a^{(\lambda)}(t)$

The sigmoid function natural way maps the values to the (0,1) interval. Positive (Negative) influences can be build with $\delta_a^{\lambda_1}(t)$, $\delta_b^{\lambda_2}(t)$ and conjunctive operator where $\lambda_1 > 0$, $\lambda_2 < 0$ and $a < b$ ($\lambda_1 > 0, \lambda_2 < 0$ and $b < a$).
Using the Dombi operator (Dombi 1982b) and sigmoid

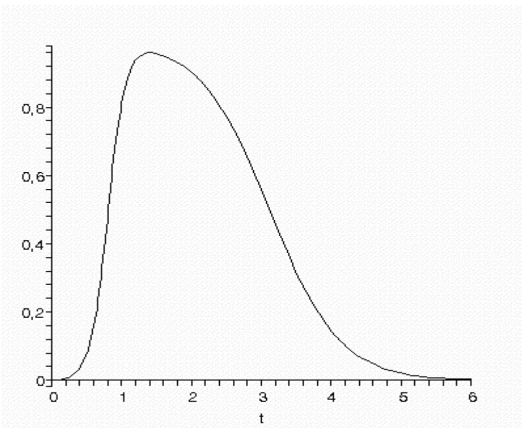
$$c(x_1, w_1, \ldots, x_n, w_n) = \frac{1}{1 + \left( \sum_{i=1}^{n} w_i \left( \frac{1 - x_i}{x_i} \right)^{\alpha} \right)^{\frac{1}{\alpha}}}$$

function with proper weights we can get

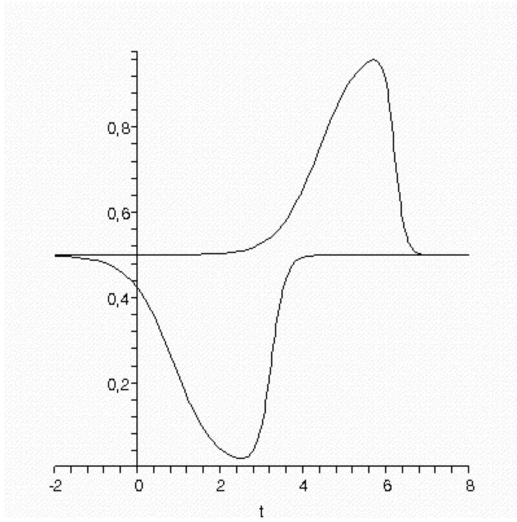
$$\delta_{a,b}^{\lambda_1,\lambda_2}(t) = \frac{1}{1 + \frac{\lambda_2}{\lambda_1 + \lambda_2} \cdot e^{-\lambda_1(t-a)} + \frac{\lambda_1}{\lambda_1 + \lambda_2} \cdot e^{-\lambda_1(t-b)}}$$

where $\lambda_i > 0$ and $a < b$.
For the aggregation we have to transform $P(t)$ the positive influence into $\left[\frac{1}{2}, 1\right]$ interval and $N(t)$ the negative influence into $\left[\frac{1}{2}, 0\right]$ interval:

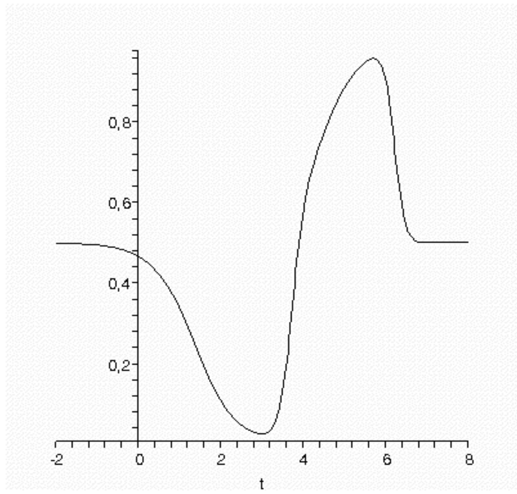
- $P(t) = \frac{1}{2}\left(1 + \delta_{a,b}^{\lambda_1,\lambda_2}(t)\right)$
- $N(t) = \frac{1}{2}\left(1 - \delta_{a,b}^{\lambda_1,\lambda_2}(t)\right)$



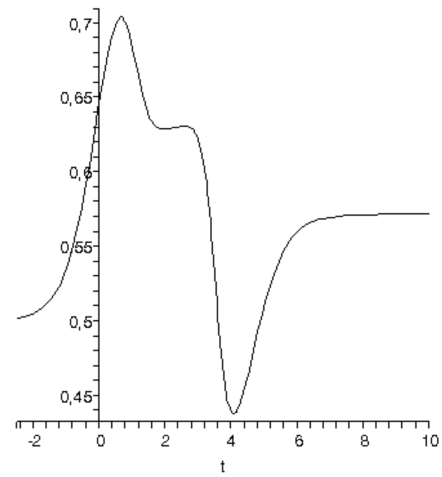

Figures 4: Asymmetrical influence on [0,1]
$a = 1, b = 3.5, \lambda_1 = 10, \lambda_2 = 2$




Figures 5: Transformation of two influences
$a^p = 4, b^p = 6, \lambda_1^p = 2, \lambda_2^p = 8$
$a^n = 1, b^n = 3, \lambda_1^n = 2, \lambda_2^n = 6$




Figures 6: Aggregation of transformed positive and negative input influences



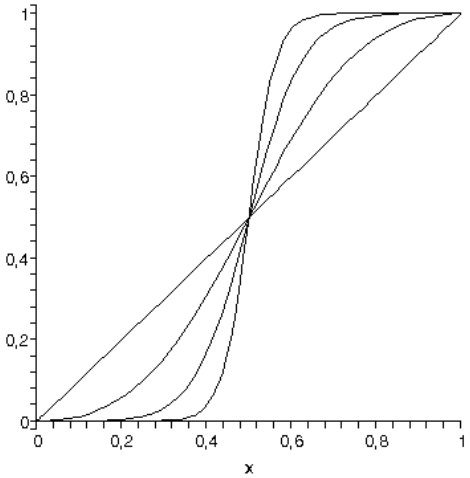

Figures 7: Aggregation influences

## VALUE TRANSFORMATION

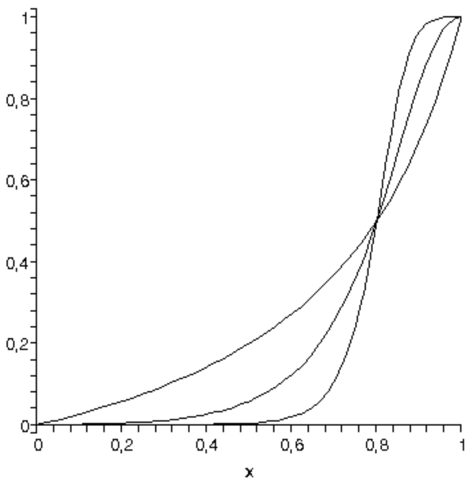
To complete the Pliant Cognitive Map we have to give unary transformation on the values produced by the aggregation. In the pliant logic the general form of the modification operators are:

$$\kappa_{\nu_{ij}}^{\lambda_{ij}}(x) = \frac{1}{1+\left(\frac{1-\nu_{ij}}{\nu_{ij}}\cdot\frac{1-x}{x}\right)^{\lambda_{ij}}}$$

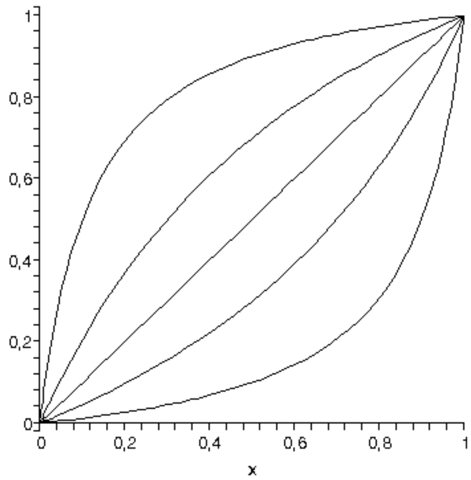
The sign of $\lambda_{ij}$ means that it is a positive or negative influence, the value of $\lambda$ mean the sharpness and $\nu_{ij}$ is the actual value of $C_j$.




Figures 8: Modification operator with the parameters $\nu = 0.5, \lambda = 1,2,4,8$




Figures 9: Modification operator with the parameters $\nu = 0.2, \lambda = 1,2,4,8$




Figures 10: Modification operator with the parameters $\lambda = 1, \nu = 0.1,0.3,0.5,0.7,0.8$

## CONSTRUCTION PCM

It is easy to build PCM. The following steps should be carries 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)

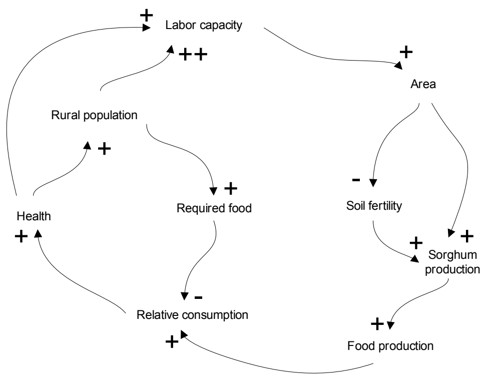
The iterative method:

1. Use the proper function to the input nodes $\delta_{a,b}^{\lambda_1,\lambda_2}$
2. Build the $P(t)$ and $N(t)$ for every nodes
3. Aggregating in the nodes the positive and negative influences, where the $\nu_*$ value is the previous value of $C_j$

The system now ready to make simulation test. We are developing a program to test the system. First we are studying artificial situation, and this shows that the system is very flexible, and easy to adapt to various situation. For the real world application we invent learning process finding the best parameter. This lead to a nonlinear problem.

## 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 11 presents a feedback diagram of the part of the model structure.




Figures 11: Feedback diagram of model structure

We generate several input function, and calculate it's effects of the nodes. The model works as it was expected. In the future we will work on an identification process i.e. based on historical data. We would like to find the proper parameters of the modification operators.

## CONCLUSION

We propose a new type of numerical calculus modeling 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 meaning.
We are working a real world application and on an effective learning of the parameters of the system.

## REFERENCES

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

Bazara, M.S.; Sherali, H.D.; Shetti, C.M.; 1993. "Nonlinear programming: Theory and Algorithms", John Willey&Sons, New York, 325p

Dombi J., 1982a, "Basic Concept for a theory of evaluation: the aggregation operator", Europian Journal of Operations Research, 10:282-293

Dombi J., 1982b. "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

Dombi J. 1987. *Membership function as an evaluation*, ITRC report, Bristol,

Fodor J., (1996). "A new look at fuzzy connectives ", Fuzzy Sets and Systems, 57:141-148

Kosko B. 1986. "Fuzzy Cognitive Maps", Int. Journal of Man-Machine Studies, Vol. 24, pp. 65-75

Kosko B. 1992. *Neural Networks and Fuzzy systems*, *A dynamic system approach to machine intelligence*, Prentice Hall, New Jersey

Kosko B., Dickerson J. 1994. "Fuzzy virtual worlds". AI Expert, pp. 25-31

Kosko B. 1997. *Fuzzy Engineering*, Prince-Hall, New Jersey

Kosko B., 1998. "Global Stability of Generalized Additive Fuzzy Systems", IEEE Transactions on Systems, Man and Cybernetics- Part C: Aplications and Reviews, Vol. 28, No 3

Tjark. E. Struif Bontkes 1993. "Dynamics of rural development in southern Sudan", System Dynamics Review, Vol. 9, No 1, Winter, 1-21p

## AUTHOR BIOGRAPHIES

**JÓZSEF DOMBI** was born in Zalaegerszeg, Hungary, and went to the University of Szeged, where he studied mathematics. His scientific degree: M. Sc. degree mathematician (1972), Ph.D. degree (1977, Summa cum laude on Hydrogen transfer reaction), CSC degree Candidate of the mathematical science (1994, The structure of the operators of fuzzy sets in the respect of multiple criteria decisions).
His research area is Intelligence, Operation Research, Optimization of logistic process. He has 70 publications. He won sum scientific prizes: including European Information Technology prize in Brussels (1997), the best software prize on the COMDEX in Las Vegas (1999) and Kalmár László prize (1998).
His e-mail address is: dombi@inf.u-szeged.hu and his Web-page can be found at http://www.inf.u-szeged.hu/~dombi.

**JÓZSEF D. DOMBI** was born in Szeged, Hungary, and went to the University of Szeged, where he studied informatics. He is a student. He several times successfully attends on the Hungarian Scientific Conference. He's research interest is modeling dynamic systems and artificial intelligence. His e-mail address is: d.j.dombi@dopti.hu.