mass-grSOM: A flexible rule extraction for classification

S.E. Papadakis and V. G. Kaburlasos
Dept. of Industrial Informatics, Div. of Computing Systems,
Technological Educational Institute of Kavala,
GR-65404 Kavala, Greece.
{spap,vgkabs}@teikav.edu.gr

Abstract - *The objective of this work is an extension of a specific family of Self Organizing Maps, namely granular Self-Organizing Maps or grSOM for short, using a novel parameterized metric distance. The resulting Self-Organizing Map, namely mass-grSOM is applicable beyond the Euclidean space R^n to $\mathbb{F}^n$ where $\mathbb{F}$ denotes the metric set of fuzzy interval numbers (FINs). A FIN here represents a local probability distribution. The proposed mass-grSOM describes linguistically the input space of a system by a set of multi-dimensional FINs induced from input/output observation data. Learning by mass-grSOM takes place in two stages. Firstly, an optimal set of multidimensional FINs is calculated using Kohonen's self organization principles. Secondly, a genetic algorithm is employed to adjust the parameters of a tunable metric distance in order to improve the classification performance. Experimental results on a benchmark classification problem show that the mass-grSOM can produce very good classification results; moreover descriptive decision making knowledge (linguistic rules) is induced from the training data.*

Key words - Self organizing maps, lattice theory, metric distance, genetic algorithm, classification, rule extraction.

1 Introduction

The interest in building machine learning applications in a black-box fashion has been increased during the past years. Neuro-fuzzy systems and neural networks has been successfully used as universal model-free estimators [3],[16],[17],[18] for classification, modeling, and control of non-linear/dynamical systems. While advanced clustering techniques [1],[4],[13], [19], are widely used for identification [15], an interest grows in model definitions, for handling jointly and non-numeric data [8],[10],[14].

A novel self organizing map, based on mathematical lattice theory, namely grSOM, has been presented lately [11]. The grSOM is applicable in $\mathbb{F}^n$ where $\mathbb{F}$ denotes a metric set of fuzzy interval numbers [7]. Since each FIN represents a local probability distribution, its shape and location must be preserved. The later constraint imposes a changeover from conventional training rules [6],[17], [16] to a novel training approach, which permits the fine tuning of the model, without distorting the original local probability distributions.

In this work a new training rule is proposed based on a novel parameterized metric distance, defined on a lattice metric space. The proposed metric distance allows the formation of a

training rule, without violating the aforementioned constraint, applicable to disparate data types including (fuzzy) numbers and intervals.
This work is organized as follows. In section 2 the mathematical background is given. In section 3, the grSOM algorithm is summarized. The proposed mass-grSOM algorithm is presented in section 4. Comparative experimental results on Fisher-IRIS classification problem are realized in section 5. Finally, the contribution of this work is summarized in section 6.

2 Mathematical background and illustrations

The mathematical background required to define the novel training rule is summarized in this section.

2.1 Metric Lattices $\mathbb{M}^h$ of Generalized Intervals

Definition 1 *A positive generalized interval* $[x_1, x_2]^h$ *of height* h *is a mapping* $\mu^h_{x_1,x_2}(x) : R \to \{0, h\}$ *where* $x_1 \leq x_2$ *and* $\mu^h_{x_1,x_2}(x) = \begin{cases} h, & x_1 \leq x \leq x_2 \\ 0, & otherwise \end{cases}$
A negative generalized interval $[x_1, x_2]^h$ *of height* h *is a mapping* $\mu^h_{x_1,x_2}(x) : R \to \{0, -h\}$ *where* $x_1 > x_2$ *and* $\mu^h_{x_1,x_2}(x) = \begin{cases} -h, & x_1 \geq x \geq x_2 \\ 0, & otherwise \end{cases}$
where $h \in (0, 1]$.

The set of positive (negative) generalized intervals of height h will be denoted by $\mathbb{M}^h_+$ ($\mathbb{M}^h_-$). The sets of positive (negative) generalized intervals for all $h \in (0, 1]$ will be denoted, respectively, by $\mathbb{M}_+$ ($\mathbb{M}_-$). An *ordering relation* can be defined in $\mathbb{M}^h, h \in (0, 1]$ as shown in the following.

$$\begin{array}{l} (R1)\, if\, [a,b]^h, [c,d]^h \in \mathbb{M}^h_+ \; then: \; [a,b]^h \leqslant [c,d]^h \Leftrightarrow [a,b] \subseteq [c,d] \\ (R2)\, if\, [a,b]^h, [c,d]^h \in \mathbb{M}^h_- \; then: \; [a,b]^h \leqslant [c,d]^h \Leftrightarrow [d,c] \subseteq [b,a], and \\ (R3)\, if\, [a,b]^h \in \mathbb{M}^h_-, [c,d]^h \in \mathbb{M}^h_+ \; then \; [a,b]^h \leqslant [c,d]^h \Leftrightarrow [b,a] \cap [c,d] \neq \varnothing \end{array} \quad (1)$$

The ordering relation given by equation (1) is a *partial ordering relation* [7]. Moreover the partially ordered set $\mathbb{M}^h$ is a mathematical lattice [7]. More specifically, $[a, b] \wedge [c, d] = [a \vee c, b \wedge d]$ and $[a, b] \vee [c, d] = [a \wedge c, b \vee d]$.

Definition 2 *A valuation in a lattice* $\mathbb{L}$ *is a real function* $V : \mathbb{L} \to R$ *which satisfies* $V(x) + V(y) = V(x \vee y) + V(x \wedge y), x, y \in \mathbb{L}$. *A valuation is called* positive *iff:* $x < y$ *in* $\mathbb{L} \Rightarrow V(x) < V(y) \forall x, y \in \mathbb{L}$.

A positive valuation function $V(.)$ in a lattice $\mathbb{L}$ implies [2] a metric distance function $d : \mathbb{L} \times \mathbb{L} \to R^+$ given by $d(x, y) = V(x \vee y) - V(x \wedge y)\, for\, x, y \in \mathbb{L}$.
For the totally-ordered lattice R of real numbers, in particular, any strictly increasing function $f(.)$ is a *positive valuation function*, the latter can be used for introducing a positive valuation function in $\mathbb{M}^h$.

Proposition 1 *Let $f : R \to R$ be a strictly increasing function, namely underlying positive valuation function. Then the function $V : \mathbb{M}^h \to R$ given by: $V([a,b]^h) = f(b) - f(a)$ is a positive valuation function in $\mathbb{M}^h$.*

An underlying positive valuation function $f : R \to R$ will be constructed here from an integrable *mass function* $m : R \to R^+$ using the formula: $f(x) = \int_0^x m(t)dt$
A mass function is used here for attaching a weight of significance to a real number.
For $x, y \in \mathbb{M}^h$ ($x = [a,b]^h, y = [c,d]^h$) there follows a metric distance between two generalized intervals as: $d_h([a,b]^h, [c,d]^h) = V([a,b]^h \vee [c,d]^h) - V([a,b]^h \wedge [c,d]^h)$, which leads to formula:

$$d_h([a,b]^h, [c,d]^h) = f(b \vee d) - f(b \wedge d) + f(a \vee c) - f(a \wedge c). \quad (2)$$

For example note that for a mass function $m(t) = h$ and $a \vee b = max(a,b)$, $a \wedge b = min(a,b) \, \forall \, a, b \in R$, it follows a metric distance between two generalized intervals $[a,b]^h$ and $[c,d]^h$ given by:

$$d_h([a,b]^h, [c,d]^h) = h(|a-c| + |b-d|) \quad (3)$$

In this work a non linear Positive Valuation Function (PVF) is introduced by the formula: $f(x) = \sum_{i=1}^{n} \{p_i \times \tanh(\frac{x-q_i}{r_i})\}$, where $p_i \in R^{*+}, q_i \in R, r_i \in R^*$ The integer n denotes the order of PVF. The function $f(x)$ is strictly increasing, thus suitable to be used as an *underlying positive valuation function.* An example of an underlined positive valuation function for $p_i = r_i = 1$ and $q_1 = 5, q_2 = 1, q_3 = -5$ and the corresponding *constructive* mass function is shown in Figure 1.

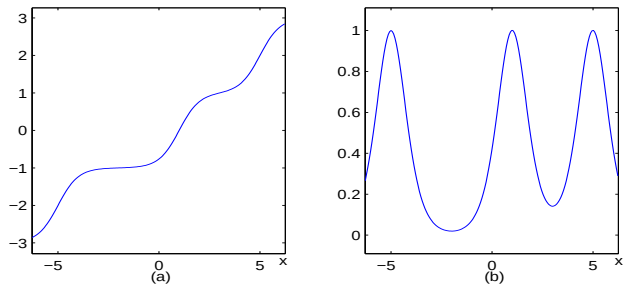

Figure 1: An underlying positive valuation function (a) and the corresponding mass function (b)

The parametric form of the proposed metric distance has certain advantages over alternative metric distances (i.e the conventional Euclidean distance). More specifically the conventional Euclidean metric distance between two numbers depends only on the relative difference between the two numbers regardless of their location on the universe of discourse; i.e. $d_E(x, x+c) = |c| \; \forall x \in R$ where as, here, the location matters. In terms of the proposed distance definition the aforementioned Euclidean distance corresponds, to a positive valuation function $f(x) = x \, \forall x \in R$ or mass function $m(t) = 1, \, \forall t \in [0, x]$.

2.2 The Metric Lattice $\mathbb{F}$ of FINs

Definition 3 *A Fuzzy Interval Number (FIN) is a function $F : (0,1] \to \mathbb{M}$ such that (1) $F(h) \in \mathbb{M}^h$, (2) either $F(h) \in \mathbb{M}^h_+$ (positive FIN), or $F(h) \in \mathbb{M}^h_-$ (negative FIN) for all $h \in (0,1]$, and (3) $h_1 \leq h_2 \Rightarrow \{x : F(h_1) \neq 0\} \supseteq \{x : F(h_2) \neq 0\}$, where $0 \leq h_1 \leq h_2 \leq 1$.*

The set of FINs is denoted by $\mathbb{F}$; more specifically the sets of positive (negative) FINs will be denoted, respectively, by $\mathbb{F}_+$ ($\mathbb{F}_-$). Note that a FIN is not a fuzzy set; rather a FIN is an abstract mathematical notion. The advantage of negative FINs is that convenient algebraic operations can be defined [7]. An ordering relation has been introduced in the set $\mathbb{F}$ of FINs as follows: $F_1 \leq F_2 \Leftrightarrow F_1(h) \leq F_2(h), \forall h \in (0,1]$. The following proposition introduces a metric distance in lattice $\mathbb{F}$.

Proposition 2 *Let F_1 and F_2 be FINs in lattice $\mathbb{F}$. A metric distance function $d_K : \mathbb{F} \times \mathbb{F} \rightarrow R$ is given by:*

$$d_K(F_1, F_2) = \int_0^1 d_h(F_1(h), F_2(h))dh \tag{4}$$

Where $d_h(F_1(h), F_2(h)) \in R_0^+$ is a metric distance between generalized intervals $F_1(h)$ and $F_2(h) \in \mathbb{F}$, and $\int(.)$ is the conventional integral operator.
In this work the metric distance given by (4) is extended by taking into account the parameterized positive valuation function as follows.
Supposing $F_1(h) = [a,b]^h = [a_h, b_h]$ and $F_2(h) = [c,d]^h = [c_h, d_h]$ (Note that the subscript notation is used here to highlight that a, b, c, d depend on h). Taking into account equation (2) the equation (4) becomes:

$$d_K(F_1, F_2) = \int_0^1 [f(b_h \vee d_h) - f(b_h \wedge d_h) + f(a_h \vee c_h) - f(a_h \wedge c_h)]dh \tag{5}$$

3 The grSOM algorithm

Consider a m-input single output (MISO) system and let $\underline{x} = [x_1, ..., x_j, ..., x_m]^T$ be the input vector, which consists of m-inputs. Suppose that y denotes the output of the system. Also, let $\Im_{m,q}$ denote the observed input/output data set comprising q m-input/output observed patterns: $\Im_q = \left\{ (\underline{x}^k, y^k), \ k = 1, ..., q \right\}$. Let $\Im_{k,\ell}$ denote a subset of $\Im_q$ which comprises k out of q data of the input x_ℓ. For each subset $\Im_{q,1}, \Im_{q,2}, \cdots, \Im_{q,\ell}$ a FIN is computed using algorithm *CALFIN* [7]. A Self Organizing Map, called grSOM, is proposed on two dimensional grid of units, each holding a m-dimensional vector of FINs; we compute a metric distance for $m(t) = 1$. Algorithm grSOM is shown in Algorithm 1.
The weights $W_{i,j}$ are initialized using *trivial FINs*, that is, $W_{i,j} \in R^m$ in the first epoch. After a user defined number N_{epochs} of epochs each unit stores a multidimensional FIN, which represents a local probability distribution function, that corresponds to a specific region of the input space. Each unit corresponds to a linguistic rule, where the antecedent part is the FIN stored in the unit and the consequent part is the respective unit's label. Note that, the linguistic rules represented by the units may not be *fuzzy rules* because a FIN represents a probability distribution rather than a possibility distribution [20]. Note that a probability distribution could be transformed to a possibility distribution [5]. Nevertheless, the aforementioned transformation is out of the scope of this paper.

Algorithm 1 The grSOM algorithm

Step-1: Define the dimensions I and J of a two-dimensional grid of $I \times J$ units (neurons). Each unit can store both a m-dimensional vector (weight) $W_{i,j}, i = 1, ..., I, j = 1, ..., J$ of FINs and a category label.

Step-2: Initialize randomly the weight of each unit by a trivial m-dimensional FIN. Repeat steps 3 and 4 below for a user-defined integer number N_{epochs} of epochs.

Step-3: For each input datum $\underline{x}_k \in \mathbb{F}^m, k = 1, ..., m$, do

3.1 Calculate the distance $d(\underline{x}_k, W_{i,j})$ between x_k and $W_{i,j}, i = 1, ..., I, j = 1, ..., J$.

3.2 Competition among the $I \times J$ units in the grid: Winner is the unit whose weight is the nearest to $\underline{x}_k$.

3.3 Assign input $\underline{x}_k$ to both the winner unit and to all the units in the neighborhood of the winner.

Step-4: Re-compute the weight $W_{ij}, i = 1, ..., I, j = 1, ..., J$ based on the data assigned to the corresponding unit in Step-3 of the current epoch.

Step-5: To each unit in the grid assign the label of the category, which provided the majority of the input data to the unit in question during all epochs.

4 A novel training algorithm (mass-grSOM)

Initially, the grSOM [11] is used for performing the structure identification process (input space partition). Then, a novel training rule is applied for fine tuning the parameters of the proposed metric distance in a supervised manner as follows:
A Positive Valuation Function (PVF) $f_j(x) \,|\, j = 1, 2, ..., m$ of order $n = 3$ is defined for each input x_j, as folows:

$$f_j(x_j) = p_{j,1} \times \tanh(\frac{x_j - q_{j,1}}{r_{j,1}}) + p_{j,2} \times \tanh(\frac{x_j - q_{j,2}}{r_{j,2}}) + p_{j,3} \times \tanh(\frac{x_j - q_{j,3}}{r_{j,3}}) \qquad (6)$$

Accordingly, the metric distance between two one dimensional FINs is defined by:

$$d_j(F_1, F_2) = \int_0^1 [f_j(b_h \vee d_h) - f_j(b_h \wedge d_h) + f_j(a_h \vee c_h) - f_j(a_h \wedge c_h)]dh. \qquad (7)$$

In the case that F_2 is a trivial FIN, that is $F_2(h) = c \,\forall\, h \in (0, 1]$ it holds: $[c, d]^h = c$, $c \vee d = c$, $c \wedge d = c \forall\, h \in (0, 1]$. For any number z a trivial FIN could be assigned as follows: $F(h) = z$, $\forall h \in (0, 1]$. Selecting the operators $\vee, \wedge$ as: $e \vee f = max(e, f)$ and $e \wedge f = min(e, f)$ where $e, f \in R$ equation the metric distance between a FIN and a real number becomes:

$$d_j(F, z) = \int_0^1 [f_j(\max(b_h, z)) - f_j(\min(b_h, z)) + f_j(\max(a_h, z)) - f_j(\min(a_h, z))]dh. \qquad (8)$$

A m-dimensional FIN $\underline{F} \in \mathbb{M}^m$ with $m \in N^*$ coordinates is considered as the cartesian-product of m one-dimensional FIN, that is $\underline{F} = \{F_1, .., F_j, .., F_m\}$ where $F_j \in \mathbb{M}_+$. The

parameterized metric distance between the m-dimensional FINs $\underline{F} \in \mathbb{M}^m$ and the vector $\underline{z} \in R^m$, $\underline{z} = \{z_1, ..., z_j, ..., z_m\}$ is given by formula:

$$D(\underline{F}, \underline{z}) = \frac{1}{m} \times \sum_{j=0}^{m} d_j(F_j, z_j)^2 \tag{9}$$

Equation (9) could be used to calculate the distance between an input vector $\underline{x}_k \in \Im_{m,q}$ and a unit of the $I \times J$ grid of grSOM. For each datum $\underline{x}_k \in D_{m,q}$, the distance from all units of the grid is calculated using (9). The datum is classified into the category indicated by the label of the unit, which corresponds to the minimum (metric) distance. For a given structure of grSOM, the success classification rate, that is the % percent of input/output patterns which are successfully classified, depends on the value of the adjustable parameters $p_{i,j}, q_{i,j}, r_{i,j} | i = 1, ..., m, j = 1, 2, 3$ in (8), (9). The novel training rule could be summarized in Algorithm 2.

Algorithm 2 The novel training rule by using the proposed parameterized metric distance

Step-1: Apply the grSOM algorithm (Algorithm 1) to locate the linguistic rules that describes the input/output patterns using the metric distance given by equation (3).

Step-2: Initialize the parameters p, q, r in (6) such that $f_j(x_j) = x_j \forall j = 1, ..., m$.

Step-3: Optimize genetically the values of p, q, r, subject to maximize the classification success rate.

In order to prevent convergence to local minima, a genetic algorithm is employed to optimize the values of the adjustable parameters p, q, r so as to maximize the success classification rate. For each input variable x_j, the respective positive valuation function f_j involves nine adjustable parameters. For m input variables (attributes in classification literature) $9m$ adjustable parameters are binary encoded, each uses 16 bits, to constitute the chromosome of each individual. The fitness function, is the success classification rate, achieved by grSOM, when the parameters obtain the values encoded into the chromosome of the individual. The genetic algorithm includes elitism, roulette wheel selection for reproduction, multi-point crossover-mutation and adaptive crossover-mutation rates. Moreover, the genetic optimization scheme is enhanced by specific operators such as hill-climbing, the ASER/RSCS [12] and the microgenetic algorithm [9]. The evolution ends when the fitness of the elite individual is not further improved for a consecutive number of 20 generations.

5 Experimental results

We carried out experiments on the Fisher IRIS benchmark. This benchmark data set comprises four inputs that represent measured attributes of a crinum family such as x_1:*sepal-length*, x_2:*sepal-width*, x_3:*petal-length*, x_4:*petal-width*. The flowers are classified into three classes (Iris versicolor(class-1), Iris sestosa (class-2), Iris virginica (class-3)) represented by an integer number from 1 to 3. The data set consists of 150 input-output data points. Each attribute is normalized in the range $[0, 1]$.
Ten data sets were created, by reshuffling randomly the initial data set. For each data set the first 80% =120 data points were used, first, for linguistic rule extraction using algorithm grSOM and second using algorithm mass-grSOM. The rest 20%=30 data points were used for

testing the classification performance of any algorithm. For comparative reasons Kohonen's SOM namely Crisp SOM, was also implemented. In all cases a 4×4 grid of units was used for i) $N_{ephocs} = 10,000$ for crisp SOM and ii) $N_{ephocs} = 100$ for grSOM. In each case the average success classification rate over the ten data sets, are summarized in Table 1.
Further experiments were carried out using a different partition of input/output patterns: 60%=90 data points were used for training and 40%=60 data points for testing. The average success classification rate over the ten data sets, are summarized in Table 2.

Table 1: Iris classification benchmark 80% for training - 20% for testing

Algorithm	Training Set		Testing Set		Rules	
	Average	std	Average	std	Average	std
Crisp SOM	92.1	4.1	92.3	4.7	16	0.0
grSOM	93.3	1.5	93.8	2.1	10	1.3
mass-grSOM	95.7	0.8	95.5	1.3	10	1.3

Table 2: Iris classification benchmark 60% for training - 40% for testing

Algorithm	Training Set		Testing Set		Rules	
	Average	std	Average	std	Average	std
Crisp SOM	92.4	4.5	91.1	5.2	16	0.0
grSOM	94.3	1.5	94.1	2.3	8	1.5
mass-grSOM	96.6	1.2	96.3	1.2	8	1.5

From the experimental results it follows that the grSOM is better than Crisp SOM, while mass-grSOM is better than grSOM. Hence, genetically optimized mass functions improve classification accuracy. Moreover, the Crisp SOM requires more rules/nodes in the grid. Finally note that the difference between Training/Testing set is not statistically significant.

6 Conclusions and future work

An novel self organizing map, namely mass-grSOM, was presented in this work using a novel parameterized metric distance function. The mass-grSOM operates using SOM principles extended to the metric space of Fuzzy Interval Numbers (FINs). A FIN has been interpreted here statistically as a local probability density function. The mass-grSOM algorithm allows fine tuning, without distorting the statistical interpretation of the input space. The efficiency of mass-grSOM algorithm was demonstrated experimentally in a benchmark classification problem. Future work includes the application of mass-grSOM on larger real-world classification problems in the industry of phosphoric fertilizers.

Acknowledgment

This work has been supported in part by the third European framework programme: Operational Programme in Education and Initial Vocational Training II, under project Archimedes contract no. 04-3-001/1.

References

[1] J. Bezdek (1981), *Pattern Recognition with Fuzzy Objective Function Algorithms*, New York, Plenum Press.

[2] G. Birkhoff (1967),*Lattice Theory*, American Math. Society, Colloquium Publications

[3] S. Chen and S.A. Billings (1989), Representation of non-linear systems: The narmax model, *Int. J. control*, **vol. 49** p. 1013-1032.

[4] S. Chen and S.A. Billings (1994), Fuzzy model identification based on cluster estimation, *Journal of Intelligent and Fuzzy Systems*, **vol. 2** p. 267-278..

[5] D. Dubois and H. Prade (1986), Fuzzy sets and statistical data, *European Journal of operational research*, **vol. 25** p. 345-356.

[6] J.-S. R. Jang (1993), Anfis: Adaptive-network-based fuzzy inference system, *IEEE Trans. on Systems, Man & Cybernetics*, **vol. 23** p. 665-685.

[7] V.G. Kaburlasos (2004), Fins: Lattice theoretic tools for improving prediction of sugar production from populations of measurements, *IEEE Transactions on Systems, Man and Cybernetics -Part B*, **vol. 34** p. 1017-1030.

[8] V.G. Kaburlasos and V. Petridis (2000), Fuzzy lattice neurocomputing (fln) models, *Neural Networks*, **vol. 13** p. 1145-1170.

[9] S.A. Kazarlis, S.E. Papadakis, J. B. Theocharis, and V. Petridis (2001), Micro-genetic algorithms as generalized hill-climbing operators for ga optimazation, *IEEE Trans. on Evolutionary Computation*, **vol. 5** p. 204-217.

[10] T. Kohonen and P. Somervuo (2002), How to make large self-organizing maps for nonvectorial data, *Neural Networks*, **vol. 15** p. 945-952.

[11] S.E. Papadakis, C.C. Marinagi, V.G. Kaburlasos, and M.K. Theodorides (2004), Estimation of industrial production using the granular self-organizing map (grsom), *Proc. of the 12th Mediterranean Conf. on Control and Automation (MED'04), Kusadasi, Turkey, session TuM2-D.*

[12] S.E. Papadakis and J.B. Theocharis (1996), A novel genetic training algorithm with application to fuzzy neural networks, *IASTED Int. Conf. On Syst. Modeling Identification and Control,Austria, Insburg*, , p. 125-128.

[13] W. Pedrycz and A. Bargiela (2002), Granular clustering: A granular signature of data, *IEEE Trans. on Systems, Man and Cybernetics - Part B*, **vol. 32** p. 212-224.

[14] Seo S. and Obermayer K (2004), Self-organizing maps and clustering methods for matrix data, *Neural Networks Special Issue*, **vol. 17** p. 1211-1229.

[15] M. Sugeno and T. Yasukawa (1993), A fuzzy-logic-based approach to qualitative modeling, *IEEE Trans. Fuzzy Syst.*, **vol. 1** p. 7-31.

[16] L. Wang and R. Langari (1995), Building sugeno-type models using fuzzy discretization and orthogonal parameter estimation techniques, *IEEE Trans. Fuzzy Syst.*, **vol. 3** p. 454-458.

[17] L. X. Wang and J. M. Mendel (1992), Back-propagation fuzzy system as non-linear dynamic system identifiers, *IEEE Conf. on Fuzzy Systems*, p. 1409-1416.

[18] L.X. Wang (1992), Fuzzy systems are universal approximators, *Proc. IEEE Int. Conf. Fuzzy Systems (San Diego, CA)*, p. 1163-1170.

[19] R.R. Yager and D.P. Filev (1992), Approximating clustering via mountain method, *Tech. Report # MII 1305, Machine Intelligent Institute, Iona College. New Rochelle. N.Y.*

[20] L.A. Zadeh (1978), Fuzzy sets as a basis for a theory of possibility, *Fuzzy Sets Syst.*, **vol. 1** p. 3-28.