General Fuzzy Min-Max Neural Network for Clustering and Classification

Bogdan Gabrys and Andrzej Bargiela

***Abstract*—This paper describes a general fuzzy min-max (GFMM) neural network which is a generalization and extension of the fuzzy min-max clustering and classification algorithms developed by Simpson. The GFMM method combines the supervised and unsupervised learning within a single training algorithm. The fusion of clustering and classification resulted in an algorithm that can be used as pure clustering, pure classification, or hybrid clustering classification. This hybrid system exhibits an interesting property of finding decision boundaries between classes while clustering patterns that cannot be said to belong to any of existing classes. Similarly to the original algorithms, the hyperbox fuzzy sets are used as a representation of clusters and classes. Learning is usually completed in a few passes through the data and consists of placing and adjusting the hyperboxes in the pattern space which is referred to as an expansion–contraction process. The classification results can be crisp or fuzzy. New data can be included without the need for retraining. While retaining all the interesting features of the original algorithms, a number of modifications to their definition have been made in order to accommodate fuzzy input patterns in the form of lower and upper bounds, combine the supervised and unsupervised learning, and improve the effectiveness of operations.**

A detailed account of the GFMM neural network, its comparison with the Simpson's fuzzy min-max neural networks, a set of examples, and an application to the leakage detection and identification in water distribution systems are given.

***Index Terms*—Classification, clustering, fuzzy systems, fuzzy min-max neural networks, pattern recognition.**

I. Introduction

DESPITE significant progress, computer-based pattern recognition faces continuous challenge from human recognition. Humans seem to be more efficient in solving many complex classification tasks which still cannot be handled easily by computers.

One of the more promising approaches to computer-based pattern recognition is the use of artificial neural networks. They have been successfully used in many pattern recognition problems [7], [16], [29]. There are two main training strategies for pattern classification procedures: supervised and unsupervised learning. In supervised learning, often referred to as a pattern classification problem, class labels are provided with input patterns and the decision boundary between classes that minimizes misclassification is sought. In unsupervised learning, often referred to as a cluster analysis problem, the training pattern data is unlabeled and one has to deal with the task of splitting a set of patterns into a number of more or less homogenous clusters with respect to a suitable similarity measure. Patterns which are similar are allocated to the same cluster, while the patterns which differ significantly are put in different clusters. Regardless of the clustering method the final result is always a partition of patterns in disconnected or overlapped clusters.

Very often these two training strategies are treated separately since they have their specific environments and applications. However, it can be observed that one of the humans' strengths in solving classification problems is the ability to combine labeled and unlabeled data. When one does not know how to classify a new object (pattern), one is able to remember its characteristic features for later referencing or subsequent association with other objects.

Another important characteristic of human reasoning is the ease with of coping with uncertain or ambiguous data encountered in real life. The traditional (statistical) approaches to pattern classification have been found inadequate in such circumstances and this shortcoming has prompted the search for a more flexible labeling in classification problems. The theory of fuzzy sets was suggested as a way to remedy this difficulty. The seminal publication reporting the use of fuzzy sets in pattern recognition was that of Bellman *et al.* [4]. Since then the combination of fuzzy sets and pattern classification has been studied by many researchers [5], [6], [26]. The flexibility of fuzzy sets and the computational efficiency of neural networks with their proven record in pattern recognition problems has caused a great amount of interest in the combination of the two [2], [6], [9], [10], [17], [22], [25], [27], [30], [31].

The pattern recognition method reported in this paper originated from our investigation into uncertain information processing in the context of decision support (DS) for operational control of industrial processes [4], [12]. The essential requirements of such a system were: the ability to process input data in form of confidence limits; the ability to incorporate new information without need to completely retrain the network; and the ability to combine the supervised and unsupervised learning strategies within a single algorithm.

The fuzzy min-max (FMM) clustering and classification neural networks [30], [31], with their representation of classes as hyperboxes in n-dimensional pattern space and their conceptually simple but powerful learning process, provided a natural basis for our development. Interesting derivatives of the original FMM can also be found in [17] and [21].

The proposed generalized fuzzy min-max (GFMM) neural network incorporates significant modifications that improve the effectiveness of the original algorithms. In developing the

Manuscript received February 2, 1998; revised January 15, 1999 and June 22, 1999.

The authors are with the Real Time Telemetry Systems, Department of Computing, Nottingham Trent University, Nottingham NG1 4BU, U.K.

Publisher Item Identifier S 1045-9227(00)04929-8.

GFMM algorithm a number of issues have been addressed which have also drawn the attention of other researchers in recent years. These include the problems of distinguishing between ignorance and equal evidence or interpretation of degrees of membership as a measure of typicality or compatibility [18], incorporating labeled data into clustering algorithms [28], interval analysis [23], and combination of supervised and unsupervised learning.

An important development of the GFMM algorithm relates to the interpretation of the membership values, both during the training and the operation of the GFMM neural network, as the degree of belonging or compatibility as advocated by Krishnapuram and Keller [18]. By relaxing the probabilistic constraint that the memberships of a data point across classes have to add up to one (used in Bezdek's fuzzy C-means algorithm [5]) and suitably modifying an objective function to be minimized during the clustering process, the possibilistic C-means algorithm was proposed that not only provided the membership values corresponding more closely to the notion of typicality, but also proved to be more immune to noise.

Due to the fact that the fuzzy membership functions proposed by Simpson and used in FMM algorithms can assign a relatively high membership value to an input vector which is quite far from the cluster prototype, as illustrated in Figs. 3 and 4, it was necessary to propose a new membership function which monotonically decreases with a growing distance from a cluster prototype, as illustrated in Fig. 5, thus eliminating the likely confusion between cases of "equally likely" and "unknown" inputs. It will also be shown that after modification of expansion criteria the GFMM can create larger (in volumetric sense) hyperboxes with greater ability to identify outliers and reduce their influence on data partitioning.

In many applications uncertainty associated with input data is quantified and presented in form of confidence intervals or confidence limits [3], [14], [20]. The interval analysis [23], which underlies the fuzzy sets and fuzzy numbers theory, was initially used for accounting for a finite tolerance of elements and recognized that some physical entities can take any value from an interval rather than be described by a single crisp value. The input to GFMM neural network has been generalized from a point in n-dimensional pattern space to a hyperbox which is a multidimensional representation of a set of variables given in the form of lower and upper limits—intervals in R. This, combined with the modified membership function and an internal representation of data clusters as hyperboxes, provided a way of processing that type of uncertain inputs in a very efficient manner. The compatibility between the type of input and cluster representation has also been utilized in neural network structure optimization, i.e., reducing a number of hyperbox clusters encoded in the network without loss of recognition performance.

Another problem addressed in this paper concerns the combination of supervised and unsupervised learning within the framework of fuzzy min-max neural networks. While Simpson presented two separate approaches to neural network training: one for clustering problem and one for classification problem, the GFMM combines them in a single algorithm. Pedrycz and Waletzky, in their paper concerning clustering with partial supervision [28], have pointed out that quite often the real-world applications call for many intermediate modes of the structural search in the data set, the efficiency of which could be substantially enhanced by a prudent use of the available domain knowledge about the classification problem at hand. It was shown that even a small percentage of the labeled patterns substantially improved the results of clustering. While Pedrycz and Waletzky's algorithm is based on minimization of the suitably formulated objective function, the positive effect of labeled patterns on generated clusters have also been observed in GFMM as reported in this paper.

The problems of generalization, overfitting and reducing the number of hyperboxes created during the training were addressed by proposing an adaptive maximum size of hyperbox scheme. The maximum hyperbox size Θ is the most important user-specified parameter which decides how many hyperboxes will be created. Generally, the larger Θ, the fewer hyperboxes are created. This has an effect of increasing the generalization ability of the network but it decreases the ability to capture nonlinear boundaries between classes. On the other hand, small Θ may lead to data overfitting with the extreme case of individual inputs memorised as separate hyperboxes. The scheme attempting to find a compromise between these two conflicting options has been implemented in GFMM.

The essential characteristics of the proposed GFMM can be summarized in the following points.

1) Input patterns can be fuzzy hyperboxes in the pattern space, or crisp-points in the pattern space.
2) The fuzzy hyperbox membership function and basic hyperbox expansion constraint proposed in [30] and [31] have been modified.
3) The labeled and unlabeled input patterns can be processed at the same time which resulted in an algorithm that can be used as pure clustering, pure classification, or hybrid clustering/classification system.
4) The parameter regulating the maximum hyperbox size can be changed adaptively in the course of GFMM neural network training.

The remainder of this paper is organized as follows. Section II gives a short description of the original fuzzy min-max algorithm. In Section III, the detailed description of the GFMM neural network with emphasis on new features and reasons behind changes to the original fuzzy min-max neural networks is given. Section IV presents a set of examples demonstrating different aspects of the GFMM neural network operations. Its comparison with the original fuzzy min-max neural network for the IRIS data and some results of applying it to the leakage detection and identification in water distribution systems are also given. Finally, the conclusions are outlined in Section V.

II. The Original Fuzzy Min-Max Algorithms

The fuzzy min-max clustering and classification neural networks [30], [31] are built using hyperbox fuzzy sets. A hyperbox defines a region of the n-dimensional pattern space, and all patterns contained within the hyperbox have full cluster/class membership. A hyperbox is completely defined by its min point and its max point. The combination of the min-max points and the hyperbox membership function defines a fuzzy set (cluster). In the case of a classification, hyperbox fuzzy sets are aggregated to form a single fuzzy set class.

Learning in the fuzzy min-max clustering and classification neural networks consists of creating and expanding/contracting hyperboxes in a pattern space. The learning process begins by selecting an input pattern and finding the closest hyperbox to that pattern that can expand (if necessary) to include the pattern. If a hyperbox cannot be found that meets the expansion criteria, a new hyperbox is formed and added to the system. This growth process allows existing clusters/classes to be refined over time, and it allows new clusters/classes to be added without retraining. One of the undesirable effects of hyperbox expansion are overlapping hyperboxes. Because hyperbox overlap causes ambiguity and creates possibility of one pattern fully belonging to two or more different clusters/classes, a contraction process is utilized to eliminate any undesired hyperbox overlaps. In the case of a classification NN the overlap is eliminated only for hyperboxes that represent different classes.

In summary, the fuzzy min-max neural network learning algorithm is a four-step process consisting of *Initialization*, *Expansion*, *Overlap Test*, and *Contraction* with the last three steps repeated for each training input pattern.

III. GFMM Algorithm

This section provides a description of the proposed GFMM algorithm based on the principle of expansion/contraction process. For the reference and comparison purposes, the notation used in the following section have been kept consistent, as far as possible, with the original papers introducing fuzzy min-max neural networks.

A. Basic Definitions

1) Input: The first extension introduced in the GFMM specification concerns the form of the input patterns that can be processed. The input is specified as the ordered pair

$$\{\boldsymbol{X}_h, d_h\} \tag{1}$$

where $\boldsymbol{X}_h = [\boldsymbol{X}_h^l \;\; \boldsymbol{X}_h^u]$ is the hth input pattern in a form of lower, $\boldsymbol{X}_h^l$, and upper, $\boldsymbol{X}_h^u$, limits vectors contained within the n-dimensional unit cube I^n; and $d_h \in \{0, 1, 2, \cdots, p\}$ is the index of one of the $p+1$ classes, where $d_h = 0$ means that the input vector is unlabeled.

In other words, instead of a point in n-dimensional space that has to be classified one has a hyperbox with the min point determined by the vector $\boldsymbol{X}_h^l$ and the max point determined by the vector $\boldsymbol{X}_h^u$. It can be observed, that when $\boldsymbol{X}_h^l$ and $\boldsymbol{X}_h^u$ are equal this hyperbox shrinks to a point. Therefore, the proposed representation of inputs is a generalization of a more conventional representation as points in n-dimensional space.

While for some simple classification problems the complete set of possible input vectors can be explicitly stated and learned, for the real, high dimensional problems typically there is a huge number of potential input vectors. Also, such data is often a mixture of labeled and unlabeled instances.

To reflect this input data complexity an additional index $(d_h = 0)$ for unlabeled data has been introduced. Using neural network phraseology, we attempt to define hybrid, supervised (labeled inputs—classification) and unsupervised (unlabeled inputs—clustering), NN.

2) Fuzzy Hyperbox Membership Function: The fuzzy hyperbox membership function plays a crucial role in the fuzzy min-max classification and clustering algorithms. The decision whether the presented input pattern belongs to a particular class or cluster, thus whether the corresponding hyperbox is to be expanded, depends mainly on the membership value describing the degree to which an input pattern fits within the hyperbox. Following Simpson [30], let the jth hyperbox fuzzy set, B_j, be defined by the ordered set

$$B_j = \{\boldsymbol{X}_h, .\boldsymbol{V}_j, \boldsymbol{W}_j, b_j(\boldsymbol{X}_h, \boldsymbol{V}_j, \boldsymbol{W}_j)\} \tag{2}$$

for all $h = 1, 2, \cdots, m$, where $\boldsymbol{X}_h = [\boldsymbol{X}_h^l \;\; \boldsymbol{X}_h^u]$ is the hth input pattern, $V_j = (v_{j1}, v_{j2}, \cdots, v_{jn})$ is the min point for the jth hyperbox, $\boldsymbol{W}_j = (w_{j1}, w_{j2}, \cdots, w_{jn})$ is the max point for the jth hyperbox, and the membership function for the ith hyperbox is $0 \leq b_j(\boldsymbol{X}_h, \boldsymbol{V}_j, \boldsymbol{W}_j) \leq 1$.

For a decision support system that intends to quantify the confidence limit on its advice, it is a natural assumption that the degree of membership of $\boldsymbol{X}_h$ for the hyperbox B_j is one if $\boldsymbol{X}_h$ is contained within the hyperbox B_j, and the degree of membership decreases as $\boldsymbol{X}_h$ moves away from the hyperbox B_j.

It should be noted that neither the membership function presented in [30] (shown in Fig. 3) nor the membership function presented in [31] (shown in Fig. 4) satisfies this assumption. The two-dimensional (2-D) example shows that even for patterns that are far from the hyperbox, the membership values are large. It can also be observed that the membership values do not decrease steadily with increasing distance from the hyperbox.

To meet the required criteria a new membership function (shown in Fig. 5) has been defined as

$$\begin{aligned} b_j(\boldsymbol{X}_h) = \min_{i=1\cdots n}(\min([1 - f(x_{hi}^u - w_{ji}, \gamma_i)],\\ [1 - f(v_{ji} - x_{hi}^l, \gamma_i)])) \end{aligned} \tag{3}$$

where $f(r, \gamma) = \begin{cases} 1 & \text{if } r\gamma > 1 \\ r\gamma & \text{if } 0 \leq r\gamma \leq 1 \\ 0 & \text{if } r\gamma < 0 \end{cases}$—two parameter ramp threshold function; $\gamma = [\gamma_1, \gamma_2, \cdots, \gamma_n]$—sensitivity parameters regulating how fast the membership values decrease.

The short interpretation of this function could be put in words as the minimum value of the maximum min-max hyperbox points violations for all dimensions. The 1-D membership function is shown in Fig. 1.

The 1-D illustration of membership value finding for an input in form of lower and upper bounds is shown in Fig. 2. The hyperbox min, v_{ji}, and max, w_{ji}, points can be violated by both lower, x_{hi}^l, and upper, x_{hi}^u, bound points at the same time. The smaller membership value is yield, however, by the upper bound value when the max hyperbox point is violated [Fig. 2(a)] and by the lower bound value when the min hyperbox point is violated [Fig. 2(c)]. On the basis of this observation, the upper bound of the input pattern is applied to max hyperbox points and the lower bound is applied to the min hyperbox points as shown in (3).

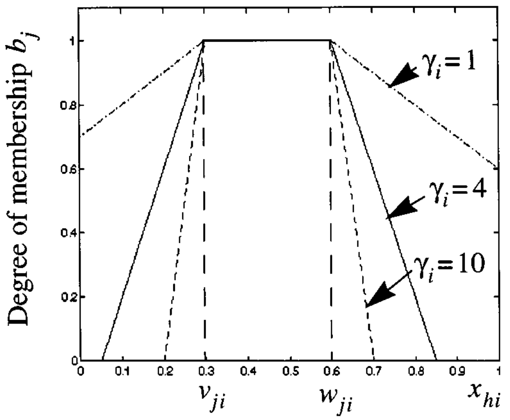


Fig. 1. One-dimensional (1-D) membership function for the hyperbox B_j and ith dimension. Examples for different γ.

B. GFMM Learning Algorithm

1) Initialization: When a new hyperbox needs to be created its min, $\boldsymbol{V}_j$, and max, $\boldsymbol{W}_j$, points are initialized in such a way that the hyperbox adjusting process used in the expansion part of the learning algorithm can be automatically used. The $\boldsymbol{V}_j$ and $\boldsymbol{W}_j$ are set initially to

$$\boldsymbol{V}_j = \boldsymbol{0} \quad \text{and} \quad \boldsymbol{W}_j = \boldsymbol{0}. \tag{4}$$

This initialization means that when the jth hyperbox is adjusted for the first time using the input pattern $\boldsymbol{X}_h = [\boldsymbol{X}_h^l \ \boldsymbol{X}_h^u]$ the min and max points of this hyperbox would be

$$\boldsymbol{V}_j = \boldsymbol{X}_h^l \quad \text{and} \quad \boldsymbol{W}_j = \boldsymbol{X}_h^u \tag{5}$$

identical to the input pattern.

2) Hyperbox Expansion: When the hth input pattern $\boldsymbol{X}_h$ is presented, the hyperbox B_j with the highest degree of membership and allowing expansion (if needed) is found. The expansion criterion, that has to be met before the hyperbox B_j can expand to include the input $\boldsymbol{X}_h$, consists of the following two parts:

$$\underset{i=1\cdots n}{\forall} \left(\max(w_{ji}, x_{hi}^u) - \min(v_{ji}, x_{hi}^l)\right) \le \Theta \tag{6}$$

and

$$\begin{aligned}
&\mathit{if} \quad d_h = 0 \ \mathit{then\ adjust}\ B_j \\
&\mathit{else} \\
&\qquad \mathit{if\ class}(B_j) = \begin{cases} 0 \Rightarrow & \begin{array}{l}\mathit{adjust}\ B_j \\ \mathit{class}(B_j) = d_h\end{array} \\ d_h \Rightarrow & \mathit{adjust}\ B_j \\ \mathit{else} \Rightarrow & \begin{array}{l}\mathit{take} \\ \mathit{another}\ B_j\end{array} \end{cases}
\end{aligned} \tag{7}$$

with the $\mathit{adjust}\ B_j$ operation defined as

$$\begin{aligned}
v_{ji}^{\text{new}} &= \min(v_{ji}^{\text{old}}, x_{hi}^l), \quad \text{for each } i = 1, \cdots, n \\
w_{ji}^{\text{new}} &= \max(w_{ji}^{\text{old}}, x_{hi}^u), \quad \text{for each } i = 1, \cdots, n.
\end{aligned}$$

If neither of the existing hyperboxes include or can expand to include the input $\boldsymbol{X}_h$, then a new hyperbox B_k is created (see *Initialization*), adjusted, and labeled by setting $\mathit{class}(B_k) = d_h$.

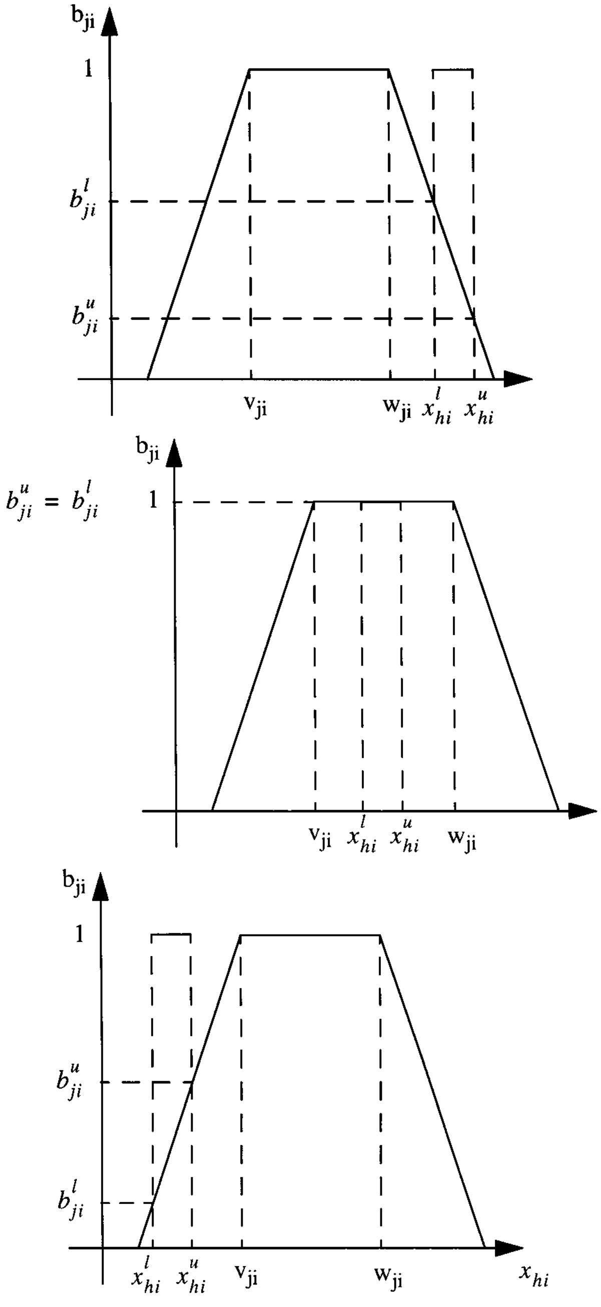


Fig. 2. The 1-D illustration of membership value finding for an input in form of lower and upper bounds.

The parameter Θ $(0 \le \Theta \le 1)$ is a user-defined value that imposes a bound on the maximum size of a hyperbox and its value significantly affects the effectiveness of the training algorithm.

In contrast to the FMM, the GFMM algorithm defines the constraint regulating the maximum size of the hyperbox (6) so as to control the size of the hyperbox for each dimension. We can thus ensure that the difference between max and min value for each dimension will not be greater than the user-specified value Θ. Example 3 in Section IV illustrates that this additional control can result in a more robust performance in presence of outliers.

$$b_j(A_h) = \frac{1}{2n}\sum_{i=1}^{n}\left[max(0, 1 - max(0, \gamma min(1, a_{hi} - w_{ji}))) + \right.$$
$$\left. + max(0, 1 - max(0, \gamma min(1, v_{ji} - a_{hi})))\right]$$

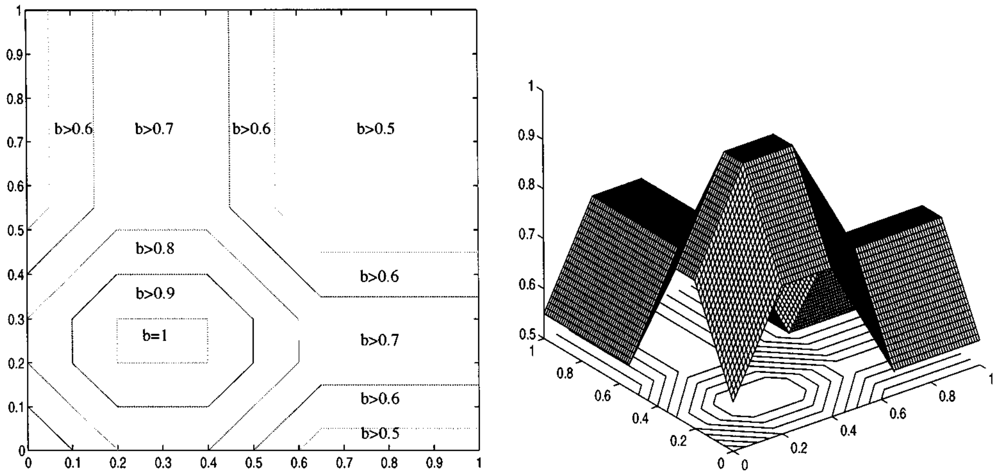


Fig. 3. The example of membership function b_j presented in [30] for the hyperbox defined by min point $V = [0.20.2]$ and max point $W = [0.30.4]$. Sensitivity parameter $\gamma = 4$.

$$b_j(A_h) = \frac{1}{n}\sum_{i=1}^{n}\left[1 - f(a_{hi} - w_{ji}, \gamma) - f(v_{ji} - a_{hi}, \gamma)\right]$$

$$\text{where } f(r, \gamma) = \begin{cases} 1 & if \quad r\gamma > 1 \\ r\gamma & if \quad 0 \le r\gamma \le 1 \\ 0 & if \quad r\gamma < 0 \end{cases}$$

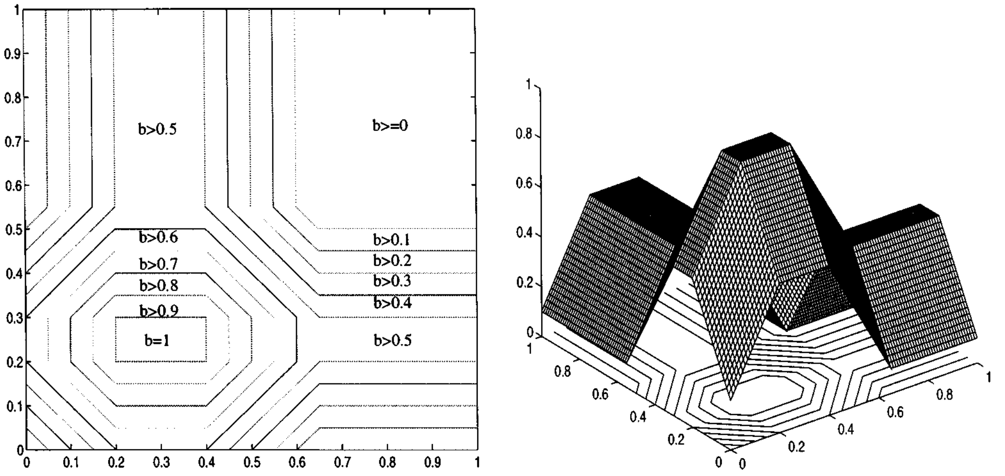


Fig. 4. The example of membership function b_j presented in [31] for the hyperbox defined by min point $V = [0.20.2]$ and max point $W = [0.30.4]$. Sensitivity parameter $\gamma = 4$.

The other differences in the expansion constraint result from admitting both labeled and unlabeled input patterns. While being a part of the expansion criterion, condition (7) describes an inference process that attempts to use all the available information carried by both labeled and unlabeled patterns to the full.

Assuming that the part of the hyperbox expansion constraint represented by (6) has been met, we have to consider the following possibilities represented by (7).

1) If the input pattern $\boldsymbol{X}_h$ is not labeled $(d_h = 0)$ then the hyperbox B_j can be adjusted to include this pattern. Since there is no information to the contrary, it is assumed that

$$b_j(X_h) = \min_{i=1..N}(\min([1-f(\overset{u}{x}_{hi} - w_{ji}, \gamma_i)], [1-f(v_{ji} - \overset{l}{x}_{hi}, \gamma_i)]))$$

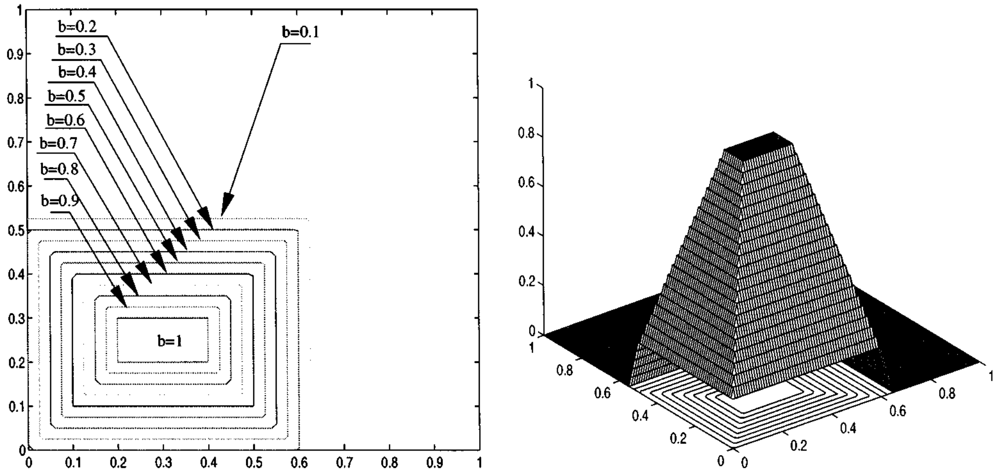


Fig. 5. The 2-D example of membership function b_j used in the GFMM classification/clustering algorithm. The hyperbox is defined by min point $V = [0.20.2]$ and max point $W = [0.30.4]$. Sensitivity parameter $\gamma = 4$.

such input could have originated from any class or cluster to which it is close enough.

2) If the input pattern is labeled $(d_h \neq 0)$—belongs to the particular class specified by d_h—the three additional cases have to be considered.

   a) If the hyperbox B_j is not a part of any of the existing classes $(class(B_j) = 0)$, then adjust the hyperbox B_j to include the input pattern $\boldsymbol{X}_h$ and since this input is labeled as belonging to the class specified by d_h, set $class(B_j) = d_h$. In this case, an association process takes place whereby a group of data represented by the hyperbox B_j is given a label on the basis of one labeled exemplar $\boldsymbol{X}_h$ that is similar (close) enough to be included into the B_j.

   b) If the hyperbox B_j is a part of the class specified by index d_h of the current input pattern $\boldsymbol{X}_h$ $(class(B_j) = d_h)$, then adjust the hyperbox B_j.

   c) If neither a nor b then take another hyperbox and test for possible expansion.

3) Hyperbox Overlap Test: Because of the admittance of labeled and unlabeled input patterns in the GFMM the problem of overlapping hyperboxes includes the consideration of not only different classes (like in the FMM classification) but also of all these hyperboxes that are not labeled.

The resulting scheme can be described as follows.

1) If the hyperbox B_j, expanded in the last expansion step, is not labeled then test for overlapping with all the other hyperboxes. This ensures that all unlabeled hyperboxes do not overlap with any of the other existing ones.

2) If the hyperbox B_j, expanded in the last expansion step, belongs to one of the existing classes then test for the overlap only with the hyperboxes not being part of the same class as B_j. Notice that this allows to overlap the hyperboxes belonging to the same class.

The full hyperbox overlap test can be therefore summarized as follows.

Assuming that hyperbox B_j was expanded in the previous step, test for overlapping with B_k if

$$class(B_j) = \begin{cases} 0 \Rightarrow & \text{test for overlapping with all the other hyperboxes} \\ else \Rightarrow & \text{test for overlapping only if } class(B_j) \neq class(B_k). \end{cases} \tag{8}$$

The principle of minimal adjustment, where only the smallest overlap for one dimension is adjusted to resolve the overlap, is used. Consequently, the smallest overlap along any dimension, the index of the dimension and the index of the case are saved.

The four cases are being considered (where initially $\delta^{\text{old}} = 1$).

Case 1: $v_{ji} < v_{ki} < w_{ji} < w_{ki}$

$$\delta^{\text{new}} = \min(w_{ji} - v_{ki}, \delta^{\text{old}}).$$

Case 2: $v_{ki} < v_{ji} < w_{ki} < w_{ji}$

$$\delta^{\text{new}} = \min(w_{ki} - v_{ji}, \delta^{\text{old}}).$$

Case 3: $v_{ji} < v_{ki} \leq w_{ki} < w_{ji}$

$$\delta^{\text{new}} = \min(\min(w_{ki} - v_{ji}, w_{ji} - v_{ki}), \delta^{\text{old}}).$$

Case 4: $v_{ki} < v_{ji} \leq w_{ji} < w_{ki}$

$$\delta^{\text{new}} = \min(\min(w_{ki} - v_{ji}, w_{ji} - v_{ki}), \delta^{\text{old}}).$$

If overlap for the ith dimension has been detected (one of the above four cases is valid) and $\delta^{\text{old}} - \delta^{\text{new}} > 0$, then $\Delta = i$,

$\delta^{\text{old}} = \delta^{\text{new}}$ and $case = l$ $(l = \{1, 2, 3, 4\}$—the case for which the smallest overlap was found).

If overlap for the ith dimension has not been detected, set $\Delta = -1$ signifying that the contraction step is not necessary.

4) Hyperbox Contraction: If $\Delta > 0$ then only the Δth dimensions of the two hyperboxes are adjusted.

Having saved the index of overlapping case in the previous step we can go to adjusting stage without examining the cases again.

The adjusting process is the same as in min-max classification algorithm although alternative formulas for cases 1 and 2 have been proposed and tested. The differences between the two contraction procedures for these two cases will be discussed in more detail later when presenting examples of the GFMM operation.

Case 1: $v_{j\Delta} < v_{k\Delta} < w_{j\Delta} < w_{k\Delta}$

$$v_{k\Delta}^{\text{new}} = w_{j\Delta}^{\text{new}} = \frac{v_{k\Delta}^{\text{old}} + w_{j\Delta}^{\text{old}}}{2} \text{ or alternatively } (w_{j\Delta}^{\text{new}} = v_{k\Delta}^{\text{old}}).$$

Case 2: $v_{k\Delta} < v_{j\Delta} < w_{k\Delta} < w_{j\Delta}$

$$v_{j\Delta}^{\text{new}} = w_{k\Delta}^{\text{new}} = \frac{v_{j\Delta}^{\text{old}} + w_{k\Delta}^{\text{old}}}{2} \text{ or alternatively } (v_{j\Delta}^{\text{new}} = w_{k\Delta}^{\text{old}}).$$

Case 3: $v_{j\Delta} < v_{k\Delta} \leq w_{k\Delta} < w_{j\Delta}$

if $w_{k\Delta} - v_{j\Delta} < w_{j\Delta} - v_{k\Delta}$ then $v_{j\Delta}^{\text{new}} = w_{k\Delta}^{\text{old}}$
otherwise $w_{j\Delta}^{\text{new}} = v_{k\Delta}^{\text{old}}$.

Case 4: $v_{k\Delta} < v_{j\Delta} \leq w_{j\Delta} < w_{k\Delta}$

if $w_{k\Delta} - v_{j\Delta} < w_{j\Delta} - v_{k\Delta}$ then $w_{k\Delta}^{\text{new}} = v_{j\Delta}^{\text{old}}$
otherwise $v_{k\Delta}^{\text{new}} = w_{j\Delta}^{\text{old}}$.

5) An Adaptive Maximum Size of the Hyperbox: In the original FMM NN's the user defined parameter Θ controlling the maximum size of created hyperboxes is set up at the beginning of the learning process and stays the same all the time. To find the best value of this parameter the network has to be trained for several different Θs and verified by checking the number of misclassifications.

After having tested the algorithm for different types of data we can say that fixing parameter Θ during the training of the network can have undesired effects on performance and the number of created hyperboxes. A large value of Θ can cause too many misclassifications, especially when there are complex, overlapping classes. On the other hand, when Θ is small, many unnecessary hyperboxes may be created, especially for concentrated, stand-alone groups of data which normally would form one class. But of course small Θ helps to resolve overlapping classes.

These problems have been addressed by introducing an adaptive maximum size of the hyperbox.

The idea is to start training the network with large Θ and decrease it (if necessary) in subsequent presentations of the data. In original versions of the algorithm the training stops after presenting the data once. Using the adaptive maximum size of the hyperbox requires defining the stopping condition. In other words when the training should be assumed to be completed.

Let us first consider the simplified case where input patterns are points in n-dimensional space. Assuming that there are no two identical points in the data which are labeled as belonging to two different classes, we can say that the training is completed when after presentation of all input patterns there have been no misclassifications for the training data. This, however, may lead to memorization of individual data patterns (overfitting) and deterioration in recognition performance for an independent testing data set.

In the case of input patterns being represented by lower and upper bound values for each dimension it is a reasonable assumption that two patterns labeled as belonging to two different classes can have overlapping regions. In such a case, an attempt to resolve the overlap over the subsequent presentations of the data, might lead to infinite adjusting of hyperboxes. Consequently, the stopping condition has to be augmented in order to ensure a finite training time.

Both goals have been achieved by specifying the minimum value that the parameter Θ can take. On one hand, it prevents the memorization of the individual input patterns while on the other hand it ensures that the inputs in form of confidence limits can be accommodated and the training stops when this minimum value is reached.

Reflecting the above the stopping condition is defined as follows.

The training is completed when:

a) after presentation of all training patterns there have been no misclassifications for the training data;
b) or the minimum user-specified value of the parameter Θ has been reached.

The value of Θ is modified after each presentation of the training data as follows:

$$\Theta^{\text{new}} = \varphi \Theta^{\text{old}} \tag{9}$$

where φ—the coefficient responsible for the speed of decrease of Θ $(0 < j < 1)$.

C. Implementation of the Neural Network

The neural network that implements the GFMM clustering/classification algorithm is shown in Fig. 6. It is a three-layer feedforward neural network that grows adaptively to meet the demands of the problem. The input layer has $2 * n$ processing elements, two for each of the n dimensions of the input pattern $\boldsymbol{X}_h - [\boldsymbol{X}_h^l \;\; \boldsymbol{X}_h^u]$. Each second-layer node represents a hyperbox fuzzy set where the connections of the first and second layers are the min-max points and the transfer function is the hyperbox membership function. The connections are adjusted using the algorithm described in Section III-B. The min points matrix $\boldsymbol{V}$ is applied to the first n input nodes representing the vector of lower bounds $\boldsymbol{X}_h^l$ of the input pattern and the max points matrix $\boldsymbol{W}$ is applied to the other n input nodes representing the vector of upper bounds $\boldsymbol{X}_h^u$ of the input pattern. The connections between the second-

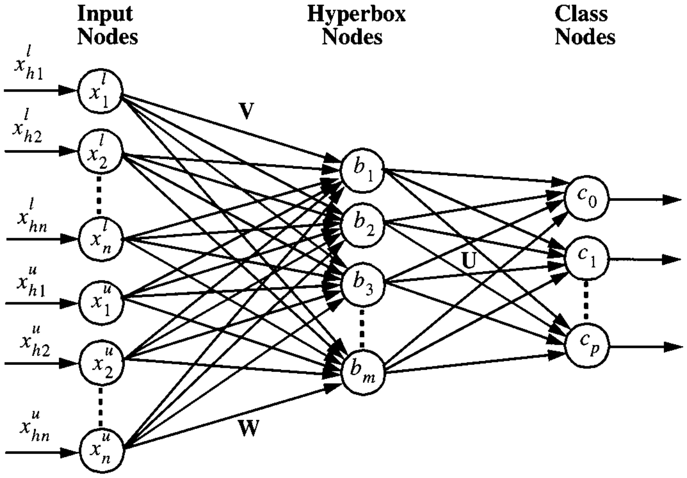


Fig. 6. The three-layer neural network that implements the GFMM clustering/classification algorithm.

and third-layer nodes are binary values. They are stored in the matrix $\boldsymbol{U}$. The equation for assigning the values of $\boldsymbol{U}$ is

$$u_{jk} = \begin{cases} 1, & \text{if } b_j \text{ is a hyperbox for class } c_k \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

where b_j is the jth second-layer node and c_k is the kth third-layer node. Each third-layer node represents a class. The output of the third-layer node represents the degree to which the input pattern $\boldsymbol{X}_h$ fits within the class k. The transfer function for each of the third-layer nodes is defined as

$$c_k = \max_{j=1}^{m} b_j u_{jk} \tag{11}$$

for each of the $p+1$ third-layer nodes. Node c_0 represents all unlabeled hyperboxes from the second layer. The outputs of the class layer nodes can be fuzzy when calculated using (11) directly, or crisp when a value of one is assigned to the node with the largest c_k and zero to the other nodes.

The topology of the network depicted in Fig. 6 is almost identical to the original fuzzy min-max neural network topology except for two changes. First, the number of input nodes has been extended from n to $2{*}n$. This has eliminated the need for double connections from input nodes to second-layer nodes. Second, an additional node representing all the unlabeled hyperboxes from the second layer has been introduced in the output layer.

IV. Simulation Results and Examples

A. Example 1—Classification of Patterns in Form of Lower and Upper Boundaries Vectors

The data set used in this example was constructed in order to show the performance of GFMM algorithm on fuzzy input patterns and, at the same time, to present the potential advantages of the adaptive maximum size of a hyperbox scheme. This data set consists of 42 input patterns representing three classes. The first and second classes have been constructed in such a way that finding the boundaries between them is nontrivial while the third class is a set of patterns standing alone and not overlapping with the other two. Two slightly different contraction procedures have been used in this experiment. The difference is reported in the table given at the bottom of the page and regards only Cases 1 and 2 of the contraction part of the algorithmy.

To illustrate the superior performance of the algorithm with the adaptive maximum size of a hyperbox, compared to the algorithms with parameter Θ preset and kept constant during the training, the training of the network for a few different constant Θ's have been carried out. The results are shown in Fig. 7.

The table alongside Fig. 7 gives a statistical information on a number of hyperboxes created and a number of misclassifications for values of Θ ranging between 0.08 and 0.033.

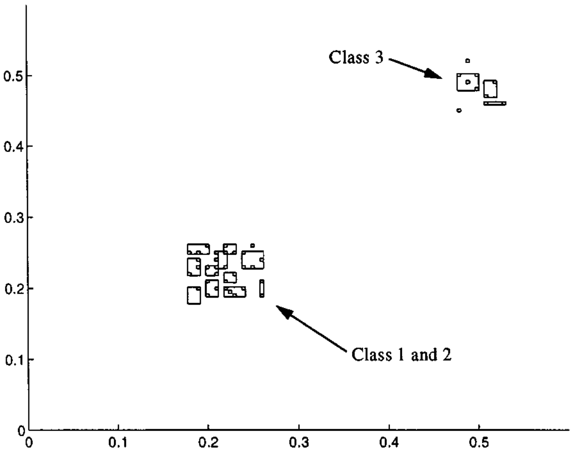
To obtain the perfect recall (zero misclassifications) for the training data the growth parameter had to be set to $\Theta = 0.033$ resulting in the formation of 17 hyperboxes as compared to 9 (for Procedure 1) or 8 (for Procedure 2) hyperboxes formed while using the adaptive scheme. Notice that five hyperboxes (Fig. 7) have been formed for class 3 while one would be sufficient.

The training of the GFMM for both contraction procedures was performed in 11 passes through the data set and testing produced a perfect recall. The starting growth parameter was $\Theta = 0.1$ and the coefficient responsible for the decrease of Θ was $\varphi = 0.9$. In this example, using the second contraction procedure, giving preference to the new inputs, resulted in all training patterns having full memberships in appropriate hyperboxes (classes).

B. Example 2—The Example of Clustering/Classification of Labeled and Unlabeled Fuzzy Input Patterns

This example was constructed to show the ability of the algorithm to process fuzzy labeled and unlabeled input patterns. The data set consists of 26 patterns from which 15 are labeled

Procedure 1	Procedure 2
Case 1: $v_{j\Delta} < v_{k\Delta} < w_{j\Delta} < w_{k\Delta}$ $v^{\text{new}}_{k\Delta} = w^{\text{new}}_{j\Delta} = \dfrac{v^{\text{old}}_{k\Delta} + w^{\text{old}}_{j\Delta}}{2}$ Case 2: $v_{k\Delta} < v_{j\Delta} < w_{k\Delta} < w_{j\Delta}$ $v^{\text{new}}_{j\Delta} = w^{\text{new}}_{k\Delta} = \dfrac{v^{\text{old}}_{j\Delta} + w^{\text{old}}_{k\Delta}}{2}$	Case 1: $v_{j\Delta} < v_{k\Delta} < w_{j\Delta} < w_{k\Delta}$ $w^{\text{new}}_{j\Delta} = v^{\text{old}}_{k\Delta}$ Case 2: $v_{k\Delta} < v_{j\Delta} < w_{k\Delta} < w_{j\Delta}$ $v^{\text{new}}_{j\Delta} = w^{\text{old}}_{k\Delta}$



Θ	Number of hyperboxes	Misclassifi- cations
0.08	4	10
0.07	6	6
0.06	7	3
0.05	9	6
0.04	12	2
0.033	17	0

Fig. 7. The result of NN training for the 42 input pattern data set (three classes). Left: the hyperboxes created for $\Theta = 0.033$—the biggest Θ for which there have been no misclassifications for the training data. Right: the table showing the number of created hyperboxes and number of misclassifications for various Θ (Θ was constant during training).

as belonging to one of four classes and the remaining 11 are unlabeled. The starting growth parameter $\Theta = 0.1$ and $\varphi = 0.9$. The training has been completed in three passes through the data set and four hyperboxes have been formed. The GFMM algorithm performed well and dealt successfully with both labeled and unlabeled patterns as shown in Fig. 8. GFMM allowed all unlabeled patterns to be included into the labeled hyperboxes while resolving all overlappings between hyperboxes from different classes.

C. Example 3—Clustering in Presence of Outliers

A simple 2-D example has been included to illustrate the enhanced error-rejection characteristics of the GFMM which derives from the modified expansion criteria and membership function. The data consisted of 41 patterns, 40 of which were randomly generated around point [0.25 0.25] within ±0.05 range, and the remaining one represented an outlier. Fig. 9 and Table I illustrate the advantage of GFMM expansion criteria, where the maximum size along individual dimensions can be controlled, over the original expansion criteria where only an average value from all the dimensions is restricted by parameter Θ. This difference means that GFMM can produce larger hyperboxes (in volumetric sense) and still avoid the undesired effect of including outliers into created hyperboxes. The advantage of the GFMM is even more pronounced for high dimensional cases.

This is to be expected since in the original FMM expansion criteria, the maximum hyperbox size for an individual dimension (in the worst case) is restricted by $n * \Theta$. The growing dimension of the input vectors may mean that a practical value for Θ, for which the outliers could be identified (not included into other hyperboxes), is unobtainable.

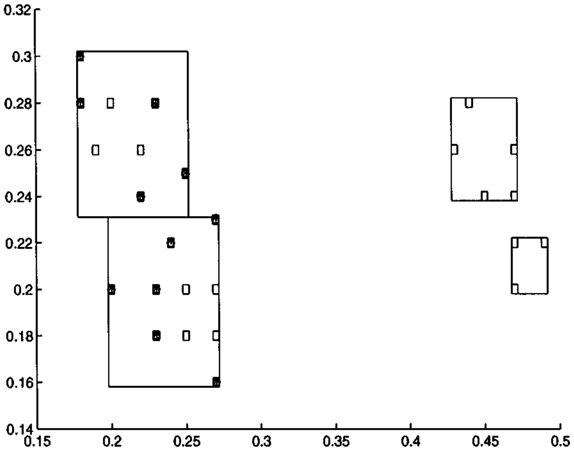


Fig. 8. The example of clustering/classification of the fuzzy labeled and unlabeled patterns. The unlabeled patterns are the rectangles with an "*" mark inside.

Table I also shows that using modified contraction procedure (Procedure 2 from Example 1) significantly improves the number of runs required for cluster stabilization.

D. Standard Data Sets

The performance of the GFMM have also been tested on standard data sets used in various clustering and classification studies. The IRIS, wine, and ionosphere data sets have been obtained from the machine learning repository of the University of California at Irvine [8]. The repository also contains the details of these data sets with some statistics and experimental results. In our study the experimental results have been restricted to the direct comparison between original FMM algorithms and GFMM. For comparative results for various statistical, fuzzy, and neural clustering and classification techniques (including Simpson's FMM), the interested readers may refer to [17].

In order to provide a meaningful comparison the algorithms have been tested in the same environment for the same splits of data sets for training and testing, the same orders of input pattern presentations and a full range of parameters.

The results shown in Table II have been obtained for a fixed parameter Θ ranging from zero (when FMM and GFMM collapse to a nearest neighbor method with a distance measure provided by respective membership functions) to one with step 0.01. The only difference between FMM and GFMM in this experiment was the membership functions and expansion criterion. High recognition rates were obtained in both cases with GFMM having a slight edge for IRIS and ionosphere data sets. It has been observed that the number of hyperboxes created decreases with the increase of parameter Θ and reaches its minimum equal to the number of classes represented in the data set for large enough Θ to encapsulate all the data from each of the classes.

After this initial testing, we have decided to concentrate on the IRIS data set and in the following we examine the issues concerning the adaptive maximum size of hyperbox scheme,

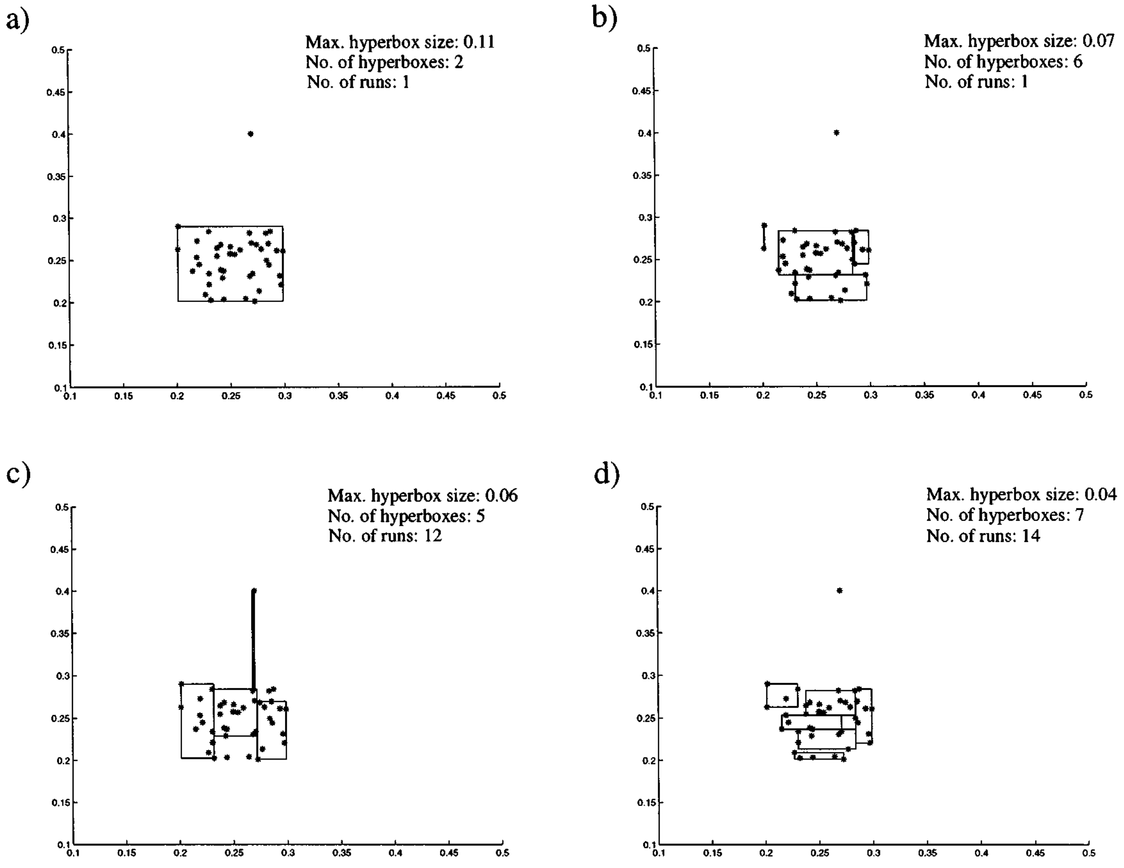


Fig. 9. The clustering in presence of outlier. (a) and (b) GFMM with modified membership function, expansion criterion, and contraction procedure. (c) and (d) Original FMM algorithm.

TABLE I
The Results Illustrating the Number of Created Hyperboxes and Number of Runs Required for Stabilization of Clusters for Simple 2-D Clustering in Presence of Outlier

Method	Maximum size of hyperbox	Contraction procedure 1		Contraction procedure 2	
		No. of hyperboxes	No. of runs	No. of hyperboxes	No. of runs
Simpson's FMM	0.03	9	12	10	2
	0.04	7	14	8	2
	0.05	5	12	5	1
	0.06	5	12	4	1
GFMM	0.05	7	7	7	2
	0.07	6	13	6	1
	0.09	3	1	3	1
	0.11	2	1	2	1

classification with superimposed noise and potential advantages of representing the input patterns in form of confidence intervals, clustering/classification with partial supervision and pure clustering.

The Fisher IRIS data set was selected because of the huge number of published results for a wide range of classification techniques that can provide a measure of relative performance. The IRIS data consists of 150 four-dimensional (4-D) feature vectors (patterns) in three separate classes, 50 for each class. In a way this example is very similar to Example 1. In Example 1, we considered the case of three classes where two of them were overlapping and the third was easily distinguishable from the others. In the case of IRIS data we have two species of flowers that can be confused (similar features—classes 2 and 3) and the third one with characteristic features allowing to distinguish it from the other two (class 1). Several test data sets have been used to determine the performance of the GFMM algorithm in different conditions.

The results presented here concern the following test data sets:

1) 25 randomly selected patterns from each class have been used for training and the remaining 75 for testing;

TABLE II
RECOGNITION RATES FOR GFMM AND FMM NEURAL NETWORKS FOR THREE REAL WORLD DATA SETS. RESULTS OBTAINED FOR FIXED MAXIMUM HYPERBOX SIZES RANGING FROM ZERO TO ONE WITH STEP 0.01

Method	IRIS Best (Range)	Wine Best (Range)	Ionosphere Best (Range)
Simpson's FMM	97.33% (97.33-92%)	100% (100-94.32%)	98.01% (98.01-84.77%)
GFMM	100% (100-92%)	100% (100-88.64%)	98.68% (98.68-90.07%)

TABLE III
THE RESULTS OF CLASSIFICATION OF THE FISHER IRIS DATA BY THE PROPOSED GENERAL FUZZY CLASSIFICATION-CLUSTERING NEURAL NETWORKS

Test data set	Starting Θ	Contraction procedure 1				Contraction procedure 2			
		No of hyperboxes	Passes through data	Final Θ	Misclassifications	No of hyperboxes	Passes through data	Final Θ	Misclassifications
1	0.3	6(1/2/3)	7	0.1594	1	5(1/2/2)	3	0.2430	1
1	0.06	29(7/10/12)	1	0.06	2	29(7/10/12)	1	0.06	2
1	0.03	49(15/17/17)	1	0.03	0	49(15/17/17)	1	0.03	0
2	0.3	10(1/5/4)	16	0.0618	0	7(1/3/3)	7	0.1594	0
2	0.06	43(10/14/19)	1	0.06	0	43(10/14/19)	1	0.06	0

2) all available data patterns have been used for training and testing.

1) Comparison of FMM with GFMM Including the Adaptive Hyperbox Size Scheme: For the first test data set (as defined above), the results presented in [30] are as follows. The growth parameter was $\Theta = 0.0175$ and the number of hyperboxes built was 48. Training was performed in a single pass through the data set. The number of misclassifications was two. This has been consistent with our implementation and testing of Simpson's algorithm.

In comparison our algorithm produced five hyperboxes for starting parameter $\Theta = 0.3$ and $\varphi = 0.9$. Training was completed in three passes through the data set. The number of misclassifications was one.

The algorithm has been tested for various starting parameters Θ ranging from zero to one with the minimum $\Theta_{\min}$ changing each time from zero to starting Θ, two different contraction procedures (presented in Example 2), and using both test data sets. A representative set of the results is shown in Table III.

It can be observed that the GFMM method produced, in general, considerably fewer hyperboxes then the fuzzy min-max classification NN and also it resulted in fewer misclassifications.

The influence of the changes of parameter φ on classification performance has been tested for values ranging from 0.98 to 0.5. No significant changes in recognition rates were noticed.

The comparison of the performance of the proposed algorithm with several other neural, fuzzy, and traditional classifiers on the same data set is presented in Table IV. This comparative performance test demonstrates that the proposed algorithm performed better then other listed classifiers.

2) Addition of Noise to Data: A noise with amplitude of 0.04 has been superimposed on training and testing data sets from point 1. The point representations have been used in Simpson's FMM. The same level of noise has been represented as hyperboxes where $\boldsymbol{X}^u$ was generated by adding 0.04 to $\boldsymbol{X}$ and $\boldsymbol{X}^l$ was generated by subtracting 0.04 from $\boldsymbol{X}$. The testing was carried for the full range of Θ and the best recognition level for FMM was recorded at 96% with a significant deterioration in recognition rates for a vast majority of tested examples. The best recognition level for GFMM, trained and tested using data in form of upper and lower limits, was recorded at 98.67%. It has also been observed that the uncertainty associated with high level of noise was directly reflected in the decrease of the membership values when the input data is given in form of hyperboxes. The level of information uncertainty (if available) cannot be represented in FMM and although the data is very noisy each point is treated as if it was absolutely accurate and with the same level of confidence irrespective of presence or absence of noise.

TABLE IV
A COMPARISON OF THE CLASSIFICATION PERFORMANCE OF VARIOUS TRADITIONAL, FUZZY, AND NEURAL CLASSIFIERS

Technique	Misclassifications
Bayes classifier [1]	2
k-nearest neighbour [1]	4
Fuzzy k-NN [2]	4
Fisher ratios [1]	3
Ho-Kashyap [1]	2
Perceptron [3]	3
Fuzzy perceptron [3]	2
Fuzzy min-max NN [1]	2
GFMM algorithm [1]	1/0
GFMM algorithm [3]	0

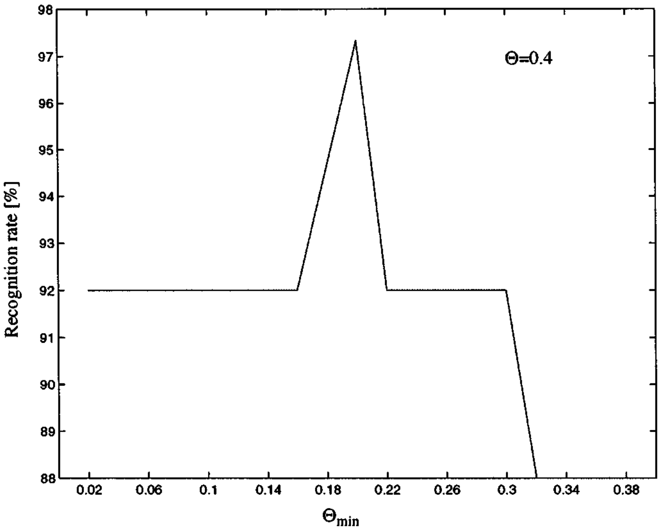


Fig. 10. Classification results for IRIS data with superimposed noise using GFMM with adaptive maximum size of hyperbox. Starting $\Theta = 0.4$.

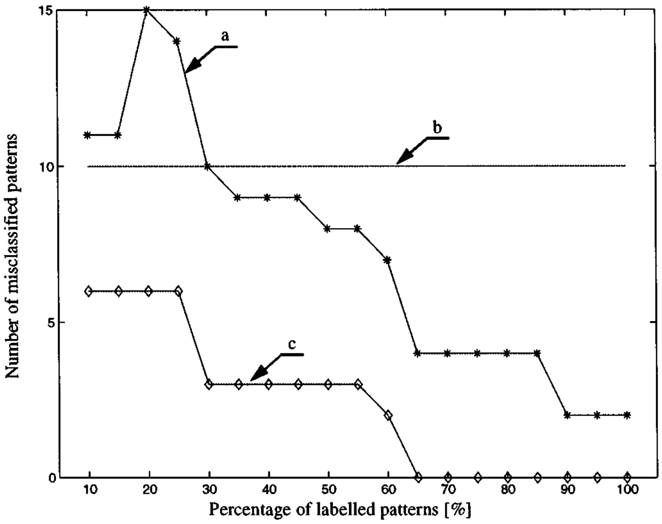


Fig. 11. Comparison of the recognition performance for the IRIS data using the training sets with varying % of labeled and unlabeled data (0%—pure clustering; 100%—pure classification). (a) FMM using only labeled data—pure classification. (b) FMM using all available data but discarding the labels—pure clustering. (c) GFMM using all available data—hybrid clustering/classification approach.

Fig. 10 illustrates how, for noisy data, a suitable choice of $\Theta_{\min}$ can prevent overfitting, which occurs for small $\Theta_{\min}$, and at the same time provides a mechanism for resolving legitimate nonlinearities when algorithm starts with relatively large value of Θ. If $\Theta_{\min}$ is too large the recognition rate decreases because of too general representation of the encoded data (too small number of hyperboxes).

3) Combination of Labeled and Unlabeled Data (Partial Supervision): In order to show the potential benefits of combing labeled and unlabeled data a number of experiments involving various proportions of labeled and unlabeled input patterns in the training data set have been carried out. The training data set from point 1 was used with the percentage of labeled patterns ranging from 100% (all training data labeled—pure classification problem) to 10% (a case of clustering with partial supervision). All the training data were used in the GFMM algorithm while the results for FMM were obtained by either applying the pure classification using only the labeled patterns or pure clustering discarding the available labels. A clear advantage in using hybrid approach is illustrated in Fig. 11 where the results are presented for all three approaches.

As rightly observed in [28], for the benefits of partial supervision to be noticed the labeled patterns have to be representative of the data set to be clustered.

4) Clustering Performance: All 150 data points of IRIS data set were used to determine the performance of GFMM in a pure clustering task. Similarly to the FMM algorithm the clustering has been performed for a fixed Θ and only after the clusters (hyperboxes) were formed a class information was used to determine how well the underlying data structure was identified. The experiments were carried out for Θ ranging from 0.03 to 0.3 with step 0.01 for GFMM and Θ ranging from 0.01 to 0.2 with step 0.01 for FMM algorithm. Representative results for both algorithms are shown in form of confusion matrices in Fig. 12. It has been observed that generally the GFMM required a smaller number of hyperboxes to obtain the same level of recognition performance as FMM, which corroborates the results presented in Example 3.

E. Leakage Detection and Identification in Water Distribution Systems

The GFMM neural network has been also applied to a complex decision support task of classification of the states of a water distribution system. Due to the space limitation, only a general description of the training and testing data sets and the performance of the neural recognition system applied to leakage detection and identification will be given. A more detailed analysis can be found in [4] and [15].

While for the well-maintained water distribution systems the normal operating state data can be found in abundance the instances of abnormal events are not that readily available. In order to observe the effects of abnormal events in the physical system, one sometimes is forced to resort to deliberate closing of valves to simulate a blocked pipe or opening of hydrants to simulate leakages. Although such experiments can be very useful to confirm the agreement between the behavior of the physical system and the mathematical model, it is not feasible to carry out such experiments for all pipes and valves in the system for an extended period of time as might be required in order to obtain a representative set of labeled data.

It is an accepted practice that, for processes where the physical interference is not recommended or even dangerous, mathematical models and computer simulations are used to predict the consequences of some emergencies so that one might be prepared for quick response. In our case, the computer simulations were used to generate data covering 24-h period of the water distribution network operations.

The simulated water distribution network was the Doncaster Eastern Zone of the Yorkshire Water Authority and consisted of 29 nodes and 38 pipes. By systematically working through the network, ten levels of leakages were introduced, one at a time, in every single pipe for every hour of the 24-h period. By applying the confidence limit analysis [3], [12], [14], the possible variations of individual input patterns have been quantified and

GFMM

a) Hyperbox size: 0.03
No. of hyperboxes: 87
Overall confusion: 0%

	1	2	3
1	100%		
2		100%	
3			100%

b) Hyperbox size: 0.06
No. of hyperboxes: 44
Overall confusion: 2.0%

	1	2	3
1	100%		
2		96%	4%
3		2%	98%

c) Hyperbox size: 0.12
No. of hyperboxes: 13
Overall confusion: 4.67%

	1	2	3
1	100%		
2		86%	14%
3			100%

Simpson's FMM

d) Hyperbox size: 0.01
No. of hyperboxes: 117
Overall confusion: 0%

	1	2	3
1	100%		
2		100%	
3			100%

e) Hyperbox size: 0.04
No. of hyperboxes: 47
Overall confusion: 3.87%

	1	2	3
1	100%		
2		96.2%	3.8%
3		7.8%	92.2%

f) Hyperbox size: 0.12
No. of hyperboxes: 12
Overall confusion: 4.9%

	1	2	3
1	100%		
2		87%	13%
3		1.7%	98.3%

Fig. 12. Representative results, in form of confusion matrices, of comparison between FMM and GFMM for pure clustering problem.

TABLE V
MISCLASSIFICATION RATES FOR THE TESTING SET CONSISTING OF 91 440 EXAMPLES

Membership values	Overall misclassification	Split of misclassifications according to different levels of leaks									
		2 [l/s]	5 [l/s]	8 [l/s]	11 [l/s]	14 [l/s]	17 [l/s]	20 [l/s]	23 [l/s]	26 [l/s]	29 [l/s]
Highest	16.8274%	33.22%	18.33%	11.02%	8.18%	7.12%	6.01%	5.02%	4.41%	3.59%	3.12%
Top2	6.1144%	57.23%	21.41%	8.80%	4.69%	2.77%	2.06%	1.06%	0.88%	0.61%	0.50%
Top3	3.1868%	73.61%	17.64%	5.42%	1.99%	0.75%	0.24%	0.14%	0.10%	0.10%	0%
Top5	1.4129%	86.22%	11.07%	2.09%	0.62%	0%	0%	0%	0%	0%	0%

stored in form of lower and upper limits. In other words, the data used in training stage were hyperboxes rather than points in the pattern space.

As a result, a training data set comprising of 9144 examples of 35 dimensional input patterns and representing 39 categories has been compiled. These categories stood for normal operating state and leakages in 38 pipes of the network.

For testing purposes an independent large testing set consisting of 91 440 patterns have been generated. But this time, the patterns to be classified were the best state estimates (points in the pattern space) obtained for measurements with superimposed random errors.

A two-level recognition system has been used. The first level consisted of one neural network and its purpose was to distinguish different typical behaviors of the water system (i.e., night load, peak load, etc.) by selecting one of the second-level neural networks. These neural networks can be viewed as "experts" since each of them was trained using only a part of the training set and covered a distinctive part of 24 hour operational period. The "experts" were responsible for detection of anomalies for some characteristic load patterns.

The training of all six second-level neural networks has been completed in a single pass through the data. Parameter Θ was determined separately for each dimension of each of the six subsets of the training set and was set to the value of the largest input hyperbox for each of these six subsets. There were no misclassifications for the training data set.

The classification results for the testing data set are shown in Table V. The percentage of misclassified input patterns for the class with the highest membership value, top two, three, and five alternatives, have been used as a means of assessing the ability to correctly detect and locate leakages. Additionally, the share of patterns representing different levels of leakages in the overall misclassification rate is presented.

The first row in Table V illustrates the overall rate of misclassified patterns for the class with the highest membership value. This is equivalent to the hard decision classifiers that are specifically designed to choose only one class which is closest to the input pattern. The rate of almost 17% of misclassified testing patterns leaves some room for improvement although over 62% of all those misclassifications were recorded for patterns representing small leakages of magnitude less or equal to 8 [l/s]. It is interesting to note that as much as 56% of all 2 [l/s] leakages from the testing set were misclassified. Let us, however, emphasize that the variation of some of the consumptions can be as much as 14 [l/s] which can easily hide the 2 [l/s] leakage. Nevertheless, it is clear that the hard classifier is not the best option in this case. The subsequent rows of the Table V illustrate the flexibility of the recognition system based on the GFMM neural networks. In contrast to the hard decision classifiers, a number of alternatives can be easily obtained and sorted with respect to the membership values. Utilizing this property the tests for the top two, three, and five alternatives have been carried out and misclassification rates calculated. Looking at the top two alternatives the overall misclassification rate has been dramatically improved to average 6.11%. When the top five alternatives have been considered the overall misclassification fell to 1.51% and practically there were no misclassifications for leakages larger or equal to 11 [l/s].

As it is very difficult to detect and pinpoint the actual location of small leakages the fuzzy outputs of the classification system have proved to be extremely useful. In this particular application when an input pattern is not distinctive enough to be classified, with a reasonable level of confidence, as belonging to only one class, the system can return a number of viable alternatives. In terms of a leakage detection problem the algorithm facilitates the identification of a problematic area if there is not enough evidence to pinpoint the leaking pipe.

V. Conclusions

A neural algorithm for clustering and classification called the General Fuzzy Min-Max has been presented. The development of this neural network resulted from a number of extensions and modifications made to the fuzzy min-max neural networks developed by Simpson. Similarly to the original methods the GFMM utilizes min-max hyperboxes as fuzzy sets. The advantages of the GFMM clustering/classification neural network over the fuzzy min-max neural networks discussed in [30] and [31] can be summarized as follows.

1) GFMM allows processing both fuzzy (hyperboxes in pattern space) and crisp (points in pattern space) input patterns. This means that the uncertainty associated with input patterns, represented by confidence limits, can be processed explicitly.
2) The fusion of clustering and classification in GFMM allows the algorithm to be used for pure clustering, pure classification or hybrid clustering/classification. As it was also advocated in [28], where incorporation of labeled data into clustering algorithm was investigated, a prudent use of all available information in pattern recognition problems can significantly improve the recognition performance and improve the process of finding the underlying structure of the data at hand.
3) The adaptation of the size of hyperboxes in the GFMM algorithm tends to result in larger hyperboxes without sacrificing the recognition rate. As it has been shown in case of the Fisher IRIS data, GFMM produced considerably fewer hyperboxes (compared to FMM) with fewer misclassifications.
4) Modifications to the membership function ensured consistent interpretation of membership values which distinguishes between the cases of "equal evidence" (class membership values high enough and equal for a number of alternatives) and "ignorance" (all class membership values equal or very close to zero).

The training of the GFMM neural network is very fast and, as long as there are no identical data belonging to two different classes, the recognition rate for training data is 100%. Since all the manipulations of the hyperboxes involve only simple compare, add, and subtract operations, the resulting algorithm is extremely efficient.

Since the GFMM forms the decision boundaries by covering the pattern space with hyperboxes, its performance will deteriorate when the characteristics of the training and test data will be very different. Therefore, it is important to provide as representative training data for the problem as possible. However, even when a large representative data set is available, the use of hyperboxes may lead to inefficient representation when one has to deal with elongated and rotated clusters of hyperelipsoidal data. In a similar manner where hyperboxes were preferred in this paper as a representation of clusters because of the specific nature of data to be processed (inputs in form of confidence limits), a suitable cluster representation should be used in problems where evidence suggests that it could be more efficient from the point of view of encoding or recognition performance.

Acknowledgment

The authors would like to acknowledge the helpful comments by the anonymous referees and the editorial comments of Dr. J. Zurada which contributed to the improvement of the final version of the paper.

References

[1] S. Abe and R. Thawonmas, "A fuzzy classifier with ellipsoidal regions," *IEEE Trans. Fuzzy Syst.*, vol. 5, pp. 358–368, Aug. 1997.

[2] Y. R. Asfour, G. A. Carpenter, S. Grossberg, and G. W. Lesher, "Fusion ARTMAP: An adaptive fuzzy network for multi-channel classification," in *Proc. World Congress on Neural Networks (WCNN-93)*, 1993, pp. 210–215.

[3] A. Bargiela and G. D. Hainsworth, "Pressure and flow uncertainty in water systems," *J. Water Resources Plan. Manage.*, vol. 115, no. 2, pp. 212–229, Mar. 1989.

[4] A. Bargiela, "Operational decision support through confidence limits analysis and pattern classification," in *Plenary Lecture, 5th Int. Conf. Computer Simulation and AI,* Mexico, Feb. 2000.

[5] R. E. Bellman, R. Kalaba, and L. A. Zadeh, "Abstraction and pattern classification," *J. Math. Anal. Appl.*, vol. 13, pp. 1–7, 1966.

[6] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Algorithms*. New York: Plenum, 1981.

[7] ——, "Computing with uncertainty," *IEEE Commun. Mag.*, pp. 24–36, Sept. 1992.

[8] C. M. Bishop, *Neural Networks for Pattern Recognition*. Oxford, U.K.: Clarendon, 1995.

[9] C. Blake, E. Keogh, and C. J. Merz. (1998) UCI repository of machine learning databases. Univ. Calif., Irvine. [Online] http://www.ics.uci.edu/\~mlearn/MLRepository.html

[10] G. A. Carpenter, S. Grossberg, N. Markuzon, J. H. Reynolds, and D. B. Rosen, "Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps," *IEEE Trans. Neural Networks*, vol. 3, pp. 698–713, 1992.

[11] G. A. Carpenter and S. Grossberg, "Fuzzy ARTMAP: A synthesis of neural networks and fuzzy logic for supervised categorization and nonstationary prediction," in *Fuzzy Neural Networks and Soft Computing*, R. R. Hager and L. A. Zadeh, Eds., 1994, pp. 126–166.

[12] A. Cichocki and A. Bargiela, "Neural networks for solving linear inequality systems," *Parallel Computing*, vol. 22, no. 11, pp. 1455–1475.

[13] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. New York: Wiley, 1973.

[14] B. Gabrys, "Neural network based decision support: Modeling and simulation of water distribution networks," Ph.D. dissertation, Nottingham Trent Univ., Nottingham, U.K., 1997.

[15] G. Gabrys and A. Bargiela, "Neural networks based decision support in presence of uncertainties," *J. Water Resources Plan. Manage.*, vol. 125, no. 5, pp. 272–280.

[16] M. H. Hassoun, *Fundamentals of Artificial Neural Networks*. Cambridge, MA: MIT Press, 1995.

[17] A. Joshi, N. Ramakrishman, E. N. Houstis, and J. R. Rice, "On neurobiological, neurofuzzy, machine learning, and statistical pattern recognition techniques," *IEEE Trans. Neural Networks*, vol. 8, Jan. 1997.

[18] R. Krishnapuram and J. M. Keller, "A possible approach to clustering," *IEEE Trans. Fuzzy Syst.*, vol. 1, pp. 98–110, May 1993.

[19] R. Krishnapuram, "Generation of membership functions via possibilistic clustering," in *Proc. 1994 IEEE 3rd Int. Fuzzy Systems Conf.*, vol. 2, June 1994, pp. 902–908.

[20] D. Lowe and K. Zapart. (1998) Point-wise confidence interval estimation by neural networks: A comparative study based on automotive engine calibration. Tech. Rep. NCRG/98/007. [Online] http://www.ncrg.aston.ac.uk/

[21] M. Meneganti, F. S. Saviello, and R. Tagliaferri, "Fuzzy neural networks for classification and detection of anomalies," *IEEE Trans. Neural Networks*, vol. 9, Sept. 1998.

[22] S. Mitra and S. K. Pal, "Self-organizing neural network as a fuzzy classifier," *IEEE Trans. Syst., Man, Cybern.*, vol. 24, Mar. 1994.

[23] R. Moore, *Interval Analysis*. Englewood Cliffs, NJ: Prentice-Hall, 1966.

[24] O. Nasaroui and R. Krishnapuram, "An improved possibilistic C-means algorithm with finite rejection and robust scale estimation," in *Proc. North American Fuzzy Information Processing*, June 1996, pp. 395–399.

[25] S. C. Newton, S. Pemmaraju, and S. Mitra, "Adaptive fuzzy leader clustering of complex data sets in pattern recognition," *IEEE Trans. Neural Networks*, vol. 3, pp. 794–800, Sept. 1992.

[26] W. Pedrycz, "Fuzzy sets in pattern recognition: Methodology and methods," *Pattern Recognit.*, vol. 23, no. 1/2, pp. 121–146, 1990.

[27] ——, "Fuzzy neural networks with reference neurons as pattern classifiers," *IEEE Trans. Neural Networks*, vol. 3, Sept. 1992.

[28] W. Pedrycz and J. Waletzky, "Fuzzy clustering with partial supervision," *IEEE Trans. Syst., Man, Cybern.*, vol. 27, pp. 787–795, Oct. 1997.

[29] P. K. Simpson, *Artificial Neural Systems: Foundations, Paradigms, Applications, and Implementations*. New York: Pergamon, 1990.

[30] ——, "Fuzzy min-max neural networks—Part 1: Classification," *IEEE Trans. Neural Networks*, vol. 3, pp. 776–786, Sept. 1992.

[31] ——, "Fuzzy min-max neural networks—Part 2: Clustering," *IEEE Trans. Fuzzy Syst.*, vol. 1, pp. 32–45, Feb. 1993.

[32] R. R. Yager and L. A. Zadeh, *Fuzzy Sets, Neural Networks, and Soft Computing*. New York: Van Nostrand Reinhold, 1994.

Bogdan Gabrys received the M.Sc. degree in electronics and telecomunication (specialization: computer control systems) from the Silesian Technical University, Poland, in 1994 and the Ph.D. degree in computer science from Nottingham Trent University, Nottingham, U.K., in 1998.

He is a Research Fellow at the Department of Computing, Nottingham Trent University. His research interests span the domains of mathematical modeling, simulation, and control with a particular emphasis on the theory and applications of artificial neural networks, fuzzy logic, genetic algorithms, and evolutionary programming and their combinations. Current research projects include the application of the abovementioned techniques for state estimation, confidence limit analysis, pattern recognition, fault detection, and decision support in water distribution networks and traffic systems.

Andrzej Bargiela received the M.Sc. degree in 1978 and the Ph.D. degree in 1984.

He is a Professor of Simulation Modeling at Nottingham Trent University, Nottingham, U.K., and a Leader of the Real Time Telemetry Systems—Simulation and Modeling Group. He is the Research Coordinator in the Department of Computing, Nottingham Trent University, and represents the Nottingham Trent University on the U.K. Engineering Professors Council. His research interest is focused on processing of uncertainty in the context of modeling and simulation of various physical and engineering systems. The research involves development of algorithms for processing uncertain information, investigation of computer architectures for such processing, and the study of information reduction through visualization. His research has been published widely, and he has been invited to lecture on simulation and modeling at a number of universities.

Dr. Bargiela has been involved in the activities of the Society for Computer Simulation (SCS) Europe since 1994 and has been a member of the International Program Committee and/or Track Chair at several SCS conferences including the recent World Simulation Congress, Singapore, in 1997. He was the General Conference and Program Chair of the 10th European Simulation Symposium, hosted by Nottingham Trent University in 1998, and is Program Chair of the forthcoming European Simulation Multiconference ESM'2004. He chaired the Traffic and Transportation Telematics track at the IEEE-sponsored conference AFRICON'99 and was a member of the Program Committee for the Harbour, Maritime and Logistics Modeling and Simulation Conference. In 1999, he was elected to serve as Chairman of the SCS European Conference Board and to be an Associate Vice President of the SCS (USA) Conference Board. He is the Editor of the SCS book series *Frontiers in Modeling* and a Member of the Editorial Board of the *Encyclopaedia of Life Support Systems*.