μARTMAP: Use of Mutual Information for Category Reduction in Fuzzy ARTMAP

Eduardo Gómez-Sánchez, *Member, IEEE*, Yannis A. Dimitriadis, *Member, IEEE*, José Manuel Cano-Izquierdo, *Associate Member, IEEE*, and Juan López-Coronado

***Abstract*—A new architecture, called μARTMAP, is proposed to impact a category proliferation problem present in Fuzzy ARTMAP. Under a probabilistic setting, it seeks a partition of the input space that optimizes the mutual information with the output space, but allowing some training error, thus avoiding overfitting. It implements an inter-ART reset mechanism that permits handling exceptions correctly, thus using few categories, especially in high dimensionality problems. It compares favorably to Fuzzy ARTMAP and Boosted ARTMAP in several synthetic benchmarks, being more robust to noise than Fuzzy ARTMAP and degrading less as dimensionality increases. Evaluated on a real-world task, the recognition of handwritten characters, it performs comparably to Fuzzy ARTMAP, while generating a much more compact rule set.**

***Index Terms*—Boosted ARTMAP, category proliferation, exceptions, Fuzzy ARTMAP, μARTMAP.**

I. Introduction

ARTIFICIAL neural networks have been successfully applied to a wide variety of real-world problems and are capable of outperforming some common symbolic learning algorithms [1]. However, they are not usually applied to problems in which comprehensibility of the acquired concepts is important [2]. This includes tasks where a human supervisor must have confidence in the way the network makes its predictions, or detection of salient features hidden in the data and previously unnoticed [3]. In addition, neural networks could be used for knowledge refinement if their concepts were easily interpretable [4]. Despite several advances achieved in multilayer perceptron (MLP) backpropagation-type neural networks [2], [5], IF-THEN rules can be derived more readily from a Fuzzy ARTMAP [6] architecture, besides other well-known advantages of adaptive resonance theory (ART) networks. In Fuzzy ARTMAP each category in the F_2^a field (Fig. 1) roughly corresponds to a rule. Each node is defined by a weight vector that can be directly translated into a verbal or algorithmic description of the antecedents of the corresponding rule [7].

Though Fuzzy ARTMAP inherently represents acquired knowledge in the form of IF-THEN rules, large or noisy datasets typically cause Fuzzy ARTMAP to generate too many rules [7]. This problem is known as category proliferation [8]. It is due to the application of the match tracking mechanism, that however is necessary to guarantee fast, accurate, on line learning. This mechanism is fired after a pattern has been presented, if the selected category in ART^a predicts a wrong label: vigilance is raised and a finer or new category is selected. Unnecessary categories will be committed to learn noisy patterns [9].

Category proliferation in Fuzzy ARTMAP has been handled in different ways in the literature. It can be overcome by a rule extraction process, after training has been completed, which proceeds by selecting a small set of highly predictive categories [7]. Other approaches propose modifications of the architecture or the training algorithm. Distributed ARTMAP (dARTMAP) [10] introduces distributed coding to avoid commitment of unnecessary categories, but category proliferation is only reduced for a particular type of problem [11]. Gaussian ARTMAP [9] defines the ART choice and match functions to be the discriminant function of a Gaussian classifier, achieving a reduced number of categories along with better performance than Fuzzy ARTMAP when trained on noisy data. However, geometric interpretation of categories changes in these architectures, and therefore dARTMAP and Gaussian ARTMAP are not useful for IF-THEN rule extraction.

Boosted ARTMAP [12] defines a probabilistic setting to evaluate the need for committing new categories, without modifying the architecture of unsupervised Fuzzy ART modules. The inter-ART reset mechanism is suppressed and thus an unsupervised on-line learning cycle is performed. An off-line evaluation of the training error will determine if a new cycle with higher vigilance is required to create finer categories. This approach optimizes the size of categories, so that a reduced set of them is generated. However, because of the lack of an inter-ART reset mechanism, Boosted ARTMAP cannot handle exceptions properly, as discussed in Section III.

In this paper, μARTMAP (read MicroARTMAP, use of Mutual Information for Category Reduction in fuzzy ARTMAP) architecture is proposed, which combines probabilistic information in order to reduce the number of categories by optimizing their sizes and the use of an inter-ART reset mechanism which will allow the correct treatment of exceptions.

The rest of this paper is organized as follows: for completeness, Section II briefly summarizes Fuzzy ARTMAP architecture and training algorithm, discussing the category proliferation problem. Section III reviews Boosted ARTMAP, as one relevant architecture to impact category proliferation

Manuscript received December 13, 2000; revised April 12, 2001. This work was supported in part by Spanish CICYT under Project TIC1999-0446-C02-01.

E. Gómez-Sánchez and Y. A. Dimitriadis are with the Department of Signal Theory, Communications and Telematics Engineering, University of Valladolid, Valladolid, Spain (e-mail: edugom@tel.uva.es).

J. M. Cano-Izquierdo and J. López-Coronado are with the Department of System Engineering and Automatic Control, Polytechnical University of Cartagena, Murcia, Spain.

Publisher Item Identifier S 1045-9227(02)00347-8.

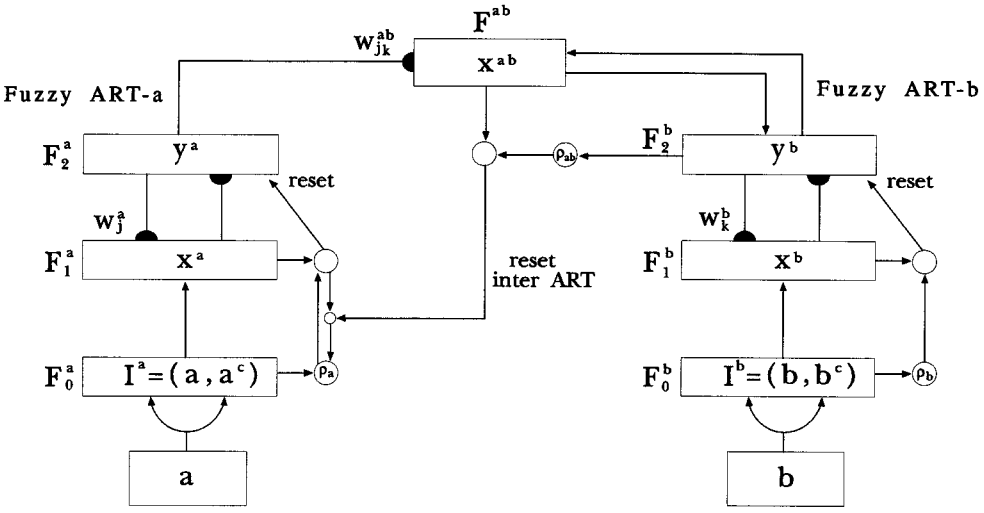

Fig. 1. Fuzzy ARTMAP architecture [6]. In ART^a module, input $\mathbf{a}$ is complemented to form vector I^a, that is transmitted to F_1^a through F_0^a. Category choice in ART^a reflects in F_2^a activity, $\mathbf{y}^a$. The same process is carried out in ART^b. If ART^a prediction is disconfirmed by ART^b, match tracking proceeds, raising ART^a vigilance, so that $\rho^a > |\mathbf{I}^a \wedge \mathbf{w}_J^a|/|\mathbf{I}^a|$ and a new ART^a category is searched, that correctly predicts $\mathbf{b}$.

while preserving original Fuzzy ART modules. The proposed μARTMAP architecture is presented in Section IV. Section V presents a comparative evaluation of μARTMAP with Fuzzy ARTMAP and Boosted ARTMAP, on variations of the well-known circle-in-square benchmark and in the difficult real-world task of handwriting recognition. Finally Section VI draws the main conclusions and outlines future research tasks.

II. Fuzzy ARTMAP

Fuzzy ARTMAP [6] is the most popular architecture derived from ART. It is capable of performing fast, stable learning in a supervised setting. In includes two unsupervised Fuzzy ART [8] modules, that partition the input and output spaces; however, fuzzy ARTMAP may suffer from category proliferation [8]–[10]. This section reviews the architecture and dynamics of Fuzzy ARTMAP and thus serves as a basis for Boosted ARTMAP [12] and μARTMAP, the proposed architecture. Emphasis will be placed on the causes of category proliferation.

A. Fuzzy ART

Fuzzy ART [8] is an extension of the original binary ART 1 system to the analog domain through the use of fuzzy AND operator ($\wedge$), instead of the logical intersection $\cap$. Fuzzy ART is a modular network (see Fig. 1) that includes an input field F_0 of nodes that store the current input vector; a choice field F_2 that contains the active categories; and a matching field F_1 that receives bottom-up input from F_0 and top-down input from F_2.

The F_0 activity vector is denoted by $\mathbf{I} = (I_1, \ldots, I_M)$, $I_i \in [0,1]$, $i = 1, \ldots, M$. The F_1 and F_2 activity vectors are $\mathbf{x} = (x_1, \ldots, x_M)$ and $\mathbf{y} = (y_1, \ldots, y_N)$, respectively. Each F_2 node is called a category and represents a prototype of the patterns selecting that category during the self-organizing activity of the Fuzzy ART module. Associated to each F_2 category node j $(j = 1, \ldots, N)$ there is a vector $\mathbf{w}_j = (w_{j1}, \ldots, w_{jM})$ of adaptive weights, or long-term memory (LTM) traces. This weight vector $\mathbf{w}_j$ subsumes both the bottom-up and top-down weight vectors of ART 1.

Initially all weights are set to one, since all categories are uncommitted. When a category is first selected then it becomes *committed* [6] and as patterns are learned its associated weights decrease, but never increase. Thus each $\mathbf{w}_j$ converges to a limit and learning is stable.

1) Category Choice: The choice field nodes operate with winner-take-all dynamics, i.e., at most one F_2 node can become active at a given time, that is said to win the competition. To select this node for a given input $\mathbf{I}$ a *choice function* T_j is computed for each node j already committed in F_2, given by

$$T_j(\mathbf{I}) = \frac{|\mathbf{I} \wedge \mathbf{w}_j|}{\alpha + |\mathbf{w}_j|} \tag{1}$$

where $\wedge$ denotes the fuzzy intersection [13] defined by $(\mathbf{p} \wedge \mathbf{q})_i \equiv \min(p_i, q_i)$, $\alpha > 0$ is the choice parameter (typically $\alpha \cong 0$) and $|\cdot|$ denotes the L^1 norm defined by

$$|\mathbf{p}| = \sum_{i=1}^{N} |p_i|. \tag{2}$$

The Jth winner node in F_2 is selected by $T_J = \max_j\{T_j : j = 1, \ldots, N\}$. When a category J is chosen $y_J = 1$ and $y_j = 0$ for $j \neq J$.

$T_j(\mathbf{I})$ measures the degree of match between the current input $\mathbf{I}$ and the LTM weights of the jth node, $\mathbf{w}_j$. In particular, the ratio $|\mathbf{I} \wedge \mathbf{w}_j|/|\mathbf{w}_j|$ reflects the fuzzy subsethood of $\mathbf{w}_j$ with respect to $\mathbf{I}$. If there is any $\mathbf{w}_j$ that is a fuzzy subset of $\mathbf{I}$, then $|\mathbf{I} \wedge \mathbf{w}_j|/|\mathbf{w}_j| = 1$ and therefore $T_j(\mathbf{I}) \geq T_k(\mathbf{I})$ for $k \neq j$. The choice parameter α determines the winner category when both $\mathbf{w}_j$ and $\mathbf{w}_k$ are fuzzy subsets of $\mathbf{I}$, by selecting the node j such that $|\mathbf{w}_j| > |\mathbf{w}_k|$.

2) Resonance: The match field (F_1) activity vector $\mathbf{x}$ obeys

$$\mathbf{x} = \begin{cases} \mathbf{I} & \text{if } F_2 \text{ is inactive} \\ \mathbf{I} \wedge \mathbf{w}_J & \text{if the } J\text{th } F_2 \text{ node is active.} \end{cases} \tag{3}$$

Vector $\mathbf{w}_J$, that represents an expected template if node J is active, is fed down from F_2 and the input vector $\mathbf{I}$ comes from

F_0. They are combined to form $\mathbf{x}$, which must be sufficiently similar to $\mathbf{I}$ to meet the vigilance criterion

$$\frac{|\mathbf{I} \wedge \mathbf{w}_J|}{|\mathbf{I}|} \geq \rho \tag{4}$$

where $\rho \in [0,1]$ is the vigilance parameter.

When this happens, the network is said to enter a *resonance* state and the LTM weight vector $\mathbf{w}_J$ can be updated. Otherwise, if $|\mathbf{I} \wedge \mathbf{w}_J|/|\mathbf{I}| < 0$ *mismatch* happens, the system is reset and unit J is inhibited (i.e., $T_J = 0$) for the rest of this input presentation. If no node j is found to meet the vigilance criterion, a new node is committed.

3) Learning: When search is finished, the weight vector $\mathbf{w}_J$ is updated according to

$$\mathbf{w}_J^{(\text{new})} = \beta(\mathbf{I} \wedge \mathbf{w}_J^{(\text{old})}) + (1-\beta)\mathbf{w}_J^{(\text{old})} \tag{5}$$

where $\beta \in [0,1]$ is the learning rate parameter. If $\beta = 1$ then *fast learning* is carried out. Throughout this paper, fast learning will be assumed for all networks.

4) Complement Coding: Normalization of Fuzzy ART inputs prevents category proliferation to some extent [8]. Normalization is achieved if $|\mathbf{I}| \equiv \gamma$ for all inputs $\mathbf{I}$. One way to normalize the input and preserve amplitude information is complement coding. If $\mathbf{a} \in [0,1]^M$ denotes the original input, then take $\mathbf{I} = (\mathbf{a}, \mathbf{a}^c) \in [0,1]^{2M}$, where $\mathbf{a}^c = \{a_i^c\}$ and $a_i^c = 1 - a_i$. This vector is normalized since $|\mathbf{I}| = M$.

Thus, the new F_0 layer input vector $\mathbf{I}$ is complement coded and both $\mathbf{I}$ and $\mathbf{w}_j$ are of dimension $2M$.

B. Fuzzy ARTMAP

Fuzzy ARTMAP [6] is a supervised neural architecture that incorporates two Fuzzy ART modules, called ART^a and ART^b, linking them via an inter-ART module F^{ab} called the *map field*, as shown in Fig. 1. This field retains predictive associations between categories and implements the *match tracking mechanism*, i.e., the ART^a vigilance parameter ρ^a is increased in response to a predictive mistmach at ART^b. This process is necessary in order to guarantee that the category that resonates has the highest degree of matching to the input pattern.

The two Fuzzy ART modules accept inputs in complement code, denoted $\mathbf{I}^a = (\mathbf{a}, \mathbf{a}^c)$ and $\mathbf{I}^b = (\mathbf{b}, \mathbf{b}^c)$, where $\mathbf{a}$ is the stimulus and $\mathbf{b}$ is the response. For ART^a, $\mathbf{x}^a = (x_1^a, \ldots, x_{2M^a}^a)$ denotes the F_1^a output vector; $\mathbf{y}^a = (y_1^a, \ldots, y_{N^a}^a)$ denotes the F_2^a output vector; and $\mathbf{w}_j^a = (w_{j1}^a, \ldots, w_{j,2M^a}^a)$ is the jth ART^a weight vector. For ART^b, $\mathbf{x}^b = (x_1^b, \ldots, x_{2M^b}^b)$ and $\mathbf{y}^b = (y_1^b, \ldots, y_{N^b}^b)$ are the output vectors of fields F_1^b and F_2^b, respectively, while $\mathbf{w}_k^b = (w_{j1}^b, \ldots, w_{j,2M^b}^b)$ is the kth ART^b weight vector. For the map field, $\mathbf{x}^{ab} = (x_1^{ab}, \ldots, x_{N^b}^{ab})$ denotes the F^{ab} output vector and $\mathbf{w}_j^{ab} = (w_{j1}^{ab}, \ldots, w_{jN^b}^{ab})$ denotes the weight vector for the jth node to F^{ab}. All activity vectors are reset to zero between input presentations.

Map Field Activation: The map field F^{ab} receives input from either or both of the ART^a and ART^b category fields. Therefore, its activation is governed by both F_2^a and F_2^b activity as shown in (6) at the bottom of the page.

If the Jth F_2^a category is active, it sends input to the map field via the weights $\mathbf{w}_J^{ab}$, which represent the possible predictive classes. If F_2^b is also active, then F^{ab} remains active only if ART^a predicts the same category as ART^b, i.e., $\mathbf{x}^{ab} = 0$ if $\mathbf{y}^b$ fails to confirm the prediction made by $\mathbf{w}_J^{ab}$. In such a case the match tracking mechanism is triggered.

Match Tracking: When an input is first presented to ART^a, the vigilance parameter ρ^a is set to its baseline value, $\bar{\rho}^a$. The map field vigilance parameter ρ^{ab} governs matching between categories in ART^a and ART^b, i.e., if $|\mathbf{x}^{ab}| < \rho^{ab}|\mathbf{y}^b|$ a predictive error occurs. In this case match tracking raises ρ^a such that $\rho^a > |\mathbf{I}^a \wedge \mathbf{w}_J^a|/|\mathbf{I}^a|$ and search for a new F_2^a coding node is triggered. This process is performed until an ART^a category is selected that correctly predicts ART^b class, or a new category is committed in ART^a.

Map Field Learning: LTM traces associated with $F_2^a \rightarrow F_2^b$ paths are stored in the map field weight matrix. Initially $w_{jk}^{ab} = 1$, $j = 1, \ldots, N^a$ and $j = 1, \ldots, N^b$. When resonance occurs with the ART^a Jth category active, $\mathbf{w}_J^{ab}$ is set equal to $\mathbf{x}^{ab}$. The Jth category in ART^a always predict the same category in ART^b.

C. Category Proliferation in Fuzzy ARTMAP

Category proliferation may occur in any system, including ART networks, run with fast, on-line learning. Thus many works have been devoted to reducing this problem [7], [9], [10], [12]. This section will analyze how a inter-ART reset mechanism is required, but the match tracking process carried out in Fuzzy ARTMAP causes unnecessary category recruitment.

Fuzzy ART categories can be seen as hyperboxes, R_j, whose corners are defined by their associated weight vectors $\mathbf{w}_j$. Using fast learning and complement coding, w_{ji} and $1 - w_{j,M+i}$ equal the minimum and maximum values of the ith component among all the patterns $\mathbf{a}$ that selected category j. Therefore, we can define the jth *category size* $|R_j|$ by

$$|R_j| = M - |\mathbf{w}_j| = \sum_{i=1}^{M} l_{ji} \tag{7}$$

where $l_{ji} = (1 - w_{j,M+i}) - w_{ji}$ is the range along the ith component of the patterns learned by the jth category.

When a category learns a pattern, either this pattern is already inside the hyperbox, or the hyperbox enlarges just enough to include it. The choice function (1) determines the winner category, showing preference for those whose hyperbox needs smaller

$$\mathbf{x}^{ab} = \begin{cases} \mathbf{y}^b \wedge \mathbf{w}_J^{ab} & \text{if the } J\text{th } F_2^a \text{ node is active and } F_2^b \text{ is active} \\ \mathbf{w}_J^{ab} & \text{if the } J\text{th } F_2^a \text{ node is active and } F_2^b \text{ is inactive} \\ \mathbf{y}^b & \text{if } F_2^a \text{ is inactive and } F_2^b \text{ is active} \\ 0 & \text{if } F_2^a \text{ is inactive and } F_2^b \text{ is inactive.} \end{cases} \tag{6}$$

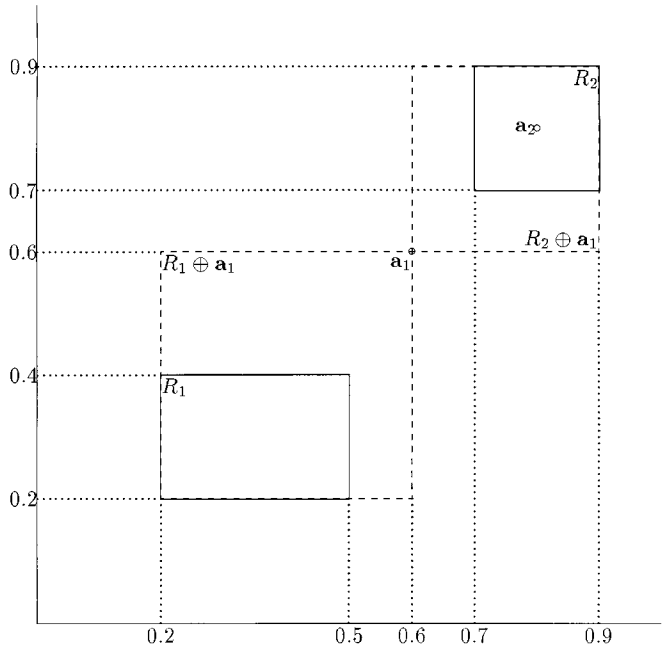

Fig. 2. Geometric representation of two hyperboxes associated to Fuzzy ART categories in a two-dimensional input space. If pattern $\mathbf{a}_1$ is presented, category R_2 will be selected, since it produces higher choice value. If its expanded size $|R_2 \oplus \mathbf{a}_1|$ would satisfy (8), it may definitely enlarge. In a supervised setting, if category 2 predicts the wrong class label, though category 1 may predict the correct one, a new hyperbox will be created of smaller size than $|R_2 \oplus \mathbf{a}_1|$, because of the match tracking mechanism. Pattern $\mathbf{a}_2$ will select category R_2, unless their predictions do not match.

changes to cover the pattern and whose size is smaller (larger $|\mathbf{w}_j|$). In addition, the vigilance condition (4) sets an upper limit on the hyperbox size, given by

$$R_j \leq M(1-\rho^a) \tag{8}$$

which applies for Fuzzy ART and also for Fuzzy ARTMAP considering $\bar{\rho}^a$, the baseline vigilance parameter. However, as match tracking can raise ρ^a during one pattern presentation, this bound may be very relaxed for Fuzzy ARTMAP. In fact, in the experiments shown in this paper $\bar{\rho}^a$ will be set to zero and thus this inequality is meaningless. However, it is important for other architectures discussed later in the paper.

These ideas are illustrated for a two-dimensional case in Fig. 2. First consider a Fuzzy ART architecture (i.e., unsupervised learning is performed), with two categories already existing, with associated weights $\mathbf{w}_1 = (0.2; 0.2; 0.5; 0.6)$ and $\mathbf{w}_2 = (0.7; 0.7; 0.1; 0.1)$ and sizes $|R_1| = 0.5$ and $|R_2| = 0.4$. If a new pattern $\mathbf{a}_1 = (0.6; 0.6)$ is presented, then the choice function is evaluated for each category, using (1), yielding $T_1 = 0.800$ and $T_2 = 0.875$ (with $\alpha \cong 0$). In this case, category R_2 wins the competition and its hyperbox could be eventually enlarged to cover pattern $\mathbf{a}_1$, yielding a hyperbox denoted by $R_2 \oplus \mathbf{a}_1$. However, if ρ is such that $|R_2 \oplus \mathbf{a}_1| > \rho$ then this unit is reset. If so, category R_1 would be selected and the vigilance criterion evaluated on it. If it could not be satisfied, a new unit with a hyperbox of null size at $\mathbf{a}$ would be created. In an unsupervised setting, pattern $\mathbf{a}_2 = (0.8; 0.8)$ will select category R_2 since it implies no changes to its hyperbox. Note that in Fuzzy ART training is unsupervised and thus the match tracking mechanism is not present.

Now consider the use of Fuzzy ARTMAP to carry out a supervised learning. While the ART^a module performs an unsupervised clustering of the patterns in the input space as described above, the match tracking mechanism will ensure that, for a given input sample $\mathbf{a}$, the category that resonates has a better match, so that if the pattern is presented again this category will be selected. Increasing ρ^a after Jth category has been reset implies that the next category selected, say J', verifies that $|R_{J'} \oplus \mathbf{a}| < |R_J \oplus \mathbf{a}|$. After learning, the new hyperbox $R_{J'} \oplus \mathbf{a}$ is the smallest containing the pattern and thus if pattern $\mathbf{a}$ is presented again it will select this category.

Now consider Fig. 2 and suppose that each category has a different associated class label through the inter-ART map. Consider that pattern $\mathbf{a}_1$ has the same class label as that predicted by category R_1. If this pattern is presented, category R_2 will be selected, since it offers higher choice value. However, since category R_2 predicts a wrong class, the match tracking mechanism is triggered raising ρ^a, by an amount sufficient to have $\rho^a > |\mathbf{I}^a \wedge \mathbf{w}_2|/|\mathbf{I}^a| = 0.7$. Also category R_2 is inhibited and then category R_1 is evaluated. However, since the match tracking mechanism raised ρ^a, this unit does not meet the vigilance criterion, i.e., $|\mathbf{I}^a \wedge \mathbf{w}_1|/|\mathbf{I}^a| = 0.6 < 0.7 < \rho^a$ and thus it is also reset. However, if baseline vigilance $\bar{\rho}^a = 0$ and category R_2 had not been already created, because all its patterns were to be presented later, pattern $\mathbf{a}_1$ could have been learned by category R_1. Thus, the match tracking mechanism, that is necessary to preserve predictive accuracy, can also cause category proliferation in some circumstances.

On the contrary, if pattern $\mathbf{a}_2$ is presented and category R_2 is selected, but their associated labels differ, the match tracking mechanism will create a new category. This category will be selected next time $\mathbf{a}_2$ is presented and the prediction would be correct. If hyperbox R_1 would have been let to grow to cover $\mathbf{a}_2$, then $|R_1 \oplus \mathbf{a}_2| \gg |R_2|$ and prediction would have been wrong next time $\mathbf{a}_2$ is presented. If additional patterns with the same class label are close to $\mathbf{a}_2$, they form what in this paper will be called *populated exceptions*, i.e., sets of patterns associated to one class label, with significant probability, surrounded by other patterns with different class label. However, if pattern $\mathbf{a}$ is noisy, then the newly created category will seldom be selected and therefore it could be obviated.

Thus, it can be said that the match tracking mechanism allows the correct treatment of populated exceptions, but may produce some category proliferation together with factors such as pattern presentation order, presence of noise in data or, class overlap.

III. BOOSTED ARTMAP

Boosted ARTMAP [12] attempts to reduce category proliferation by allowing some error on the training data and letting the underlying data distribution select the category size. It is a modification of Fuzzy ARTMAP for conducting boosted learning in a probabilistic setting. It is designed to improve generalization by optimizing category size and allowing a small training error. It is a modification of PROBART [14], which replaces the calculation of the F^{ab} activity (6) by (9) shown at the bottom of the next page, where the fuzzy AND operation ($\wedge$) is replaced by the addition ($+$). Thus, map field weights now contain information about the association frequencies between categories in F_2^a

and F_2^b, i.e., the jth ART^a node has been associated w_{jk}^{ab} times to the kth ART^b node, during the training. Initially $w_{jk}^{ab} = 0$, $j = 1,\ldots,N^a$, $k = 1,\ldots,N^b$.

In PROBART there is no match tracking and thus parameter ρ^{ab} does not exist. Therefore, the size of categories in ART^a is governed only by ρ^a. This ensures that a given input to ART^a will always select the same category and makes the network more robust to noise. Nevertheless, for a correct mapping ρ^a needs to be very high. Therefore the number of categories is also large, since very fine categories will be created averywhere in the input space.

Boosted ARTMAP (BARTMAP) allows categories formed during training to define their own sizes. It has two unsupervised fuzzy ART modules, linked by a map whose activation is given by (9), as in PROBART. However, ART^a module is modified to associate a ρ_j^a vigilance parameter to each category j, instead of a single ρ^a. They are usually initialized with low values, which can result in poor generalization. To correct this, instead of using a match tracking mechanism, batch training is carried out. After one training epoch is complete the total training error, ε_T, is computed. Since the jth ART^a category predicts the K_jth class label that has been associated to j with highest frequency, i.e., $K_j = \arg\max_k \left\{ w_{jk}^{ab} : k = 1,\ldots,N^b \right\}$, ε_T is given by

$$\varepsilon_T = \frac{\sum_{j=1}^{N^a} (|\mathbf{w}_j^{ab}| - w_{jK_j}^{ab})}{\sum_{j=1}^{N^a} |\mathbf{w}_j^{ab}|} \tag{10}$$

which is the averaged sum of the error contribution of all categories in ART^a. This error is compared to a user parameter $\varepsilon_{\max}$. If $\varepsilon_T > \varepsilon_{\max}$ then the vigilance parameter of nodes with maximal error contribution is raised, by $\rho_j^{\text{new}} = \rho_j^{\text{old}} + \Delta\rho$, where $\Delta\rho$ is a user parameter, and another training epoch proceeds. During the training, the size of a category j, $|R_j|$, will be limited by its vigilance parameter ρ_j^a, as shown by (8).

Through this mechanism, BARTMAP allows some error on the training set, improving Fuzzy ARTMAP generalization and reducing the number of categories, when patterns from different classes overlap or data are noisy. In addition, category size can be determined by the underlying distribution rather than a vigilance parameter.

However, since no inter-ART reset is performed, a hyperbox cannot be created inside another hyperbox. This is important when many patterns with one class label are surrounded by many other patterns with a different class label, i.e., the so called *populated exceptions*, as Fig. 3(a). Since the size of the surrounding region increases with the dimensionality of the input space, this limitation of BARTMAP will become critical in problems with a large number of input features.

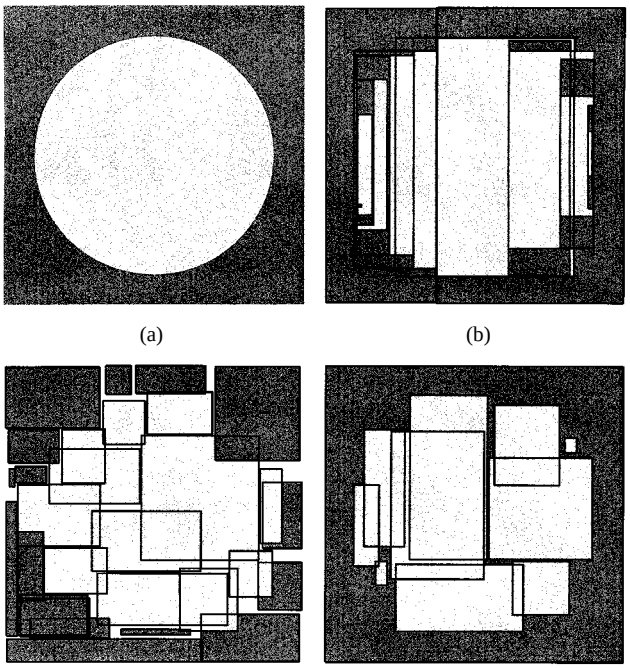

Fig. 3. The circle-in-the-square problem is depicted in (a), while (b), (c), and (d) show the hyperboxes created by Fuzzy ARTMAP, BARTMAP and μARTMAP, respectively, for the best category structure (i.e., the least categories) among those resulting from the ten training sets.

IV. μARTMAP

Boosted ARTMAP offers a means to solve the Fuzzy ARTMAP category proliferation problem, while preserving the association of each category to a hyperbox, which allows straight IF-THEN rule extraction from the learned weights. It suppresses the match tracking mechanism, that may cause category proliferation on noisy data, though it guarantees accuracy. Therefore, BARTMAP introduced an off-line evaluation mechanism in order to preserve predictive accuracy. However, BARTMAP lacks of an inter-ART reset mechanism that allows correct handling of *populated exceptions*. μARTMAP is proposed as a modification of Fuzzy ARTMAP that includes an inter-ART reset mechanism, that does not raise ART^a vigilance and thus does not cause category proliferation, while the predictive accuracy is guaranteed by an off-line learning stage.

The architecture of μARTMAP is similar to that of Fuzzy ARTMAP (Fig. 1): there are two unsupervised Fuzzy ART modules, that perform a clustering in the input and output spaces, linked by an associative map field governed by (9), i.e., one-to-many $F_2^a \rightarrow F_2^b$ relations are allowed and their probabilistic information stored in w_{jk}^{ab} weights, as in PROBART. By storing probabilistic information the need of committing a new category can be evaluated in terms of incrementing the correctness of

$$\mathbf{x}^{ab} = \begin{cases} \mathbf{y}^b + \mathbf{w}_J^{ab} & \text{if the } J\text{th } F_2^a \text{ node is active and } F_2^b \text{ is active} \\ \mathbf{w}_J^{ab} & \text{if the } J\text{th } F_2^a \text{ node is active and } F_2^b \text{ is inactive} \\ \mathbf{y}^b & \text{if } F_2^a \text{ is inactive and } F_2^b \text{ is active} \\ 0 & \text{if } F_2^a \text{ is inactive and } F_2^b \text{ is inactive} \end{cases} \tag{9}$$

the mapping. In addition, an off-line map field with weights v_{jk}^{ab} is introduced, which stores the probability of $F_2^a \rightarrow F_2^b$ relations when inter-ART reset is disabled, i.e., in prediction mode. Therefore these weights allow the system to evaluate the predictive entropy of the training set. Finally, a vigilance parameter is associated to each category node in ART^a, similarly to BARTMAP, so that category size can be determined by the underlying distribution.

A. Definitions

Given partitions of the input space A into N^a sets A_j and output space B into N^b sets B_k, the conditional entropy $H(B|A)$, here denoted simply by H, is given by

$$H = -\sum_{j=1}^{N^a} p_j \sum_{k=1}^{N^b} p_{jk} \log_2 p_{jk} \tag{11}$$

where p_j is the probability of occurrence of class A_j and p_{jk} is the conditional probability of B_k assuming A_j. Let us denote

$$h_j = -p_j \sum_{k=1}^{N^b} p_{jk} \log_2 p_{jk} \tag{12}$$

the contribution to H of set A_j.

It is important to remark that the mutual information of the partitions in A and B is given by $MI(B;A) = H(B) - H(B|A)$, where $H(B)$ is the entropy for the output space [15, Ch. 15]. Therefore, for a given $H(B)$ (as in classification tasks), minimizing the conditional entropy is equivalent to the maximization of the mutual information.

B. μARTMAP Training and Prediction

Before training all weights are initialized as in Fuzzy ARTMAP, but $w_{jk}^{ab} = 0, j = 1, \ldots, N^a, k = 1, \ldots, N^b$. A baseline ρ^a is set as a starting vigilance. This should be set to zero to minimize the number of categories, unless *a priori* knowledge of the problem indicates that fine categories will be required in all the input space. In addition, two user parameters $h_{\max}$ and $H_{\max}$ are defined to set upper bounds on h_j and H, as explained below.

Training proceeds by presenting input–output pairs, $(\mathbf{a}, \mathbf{b})$. When a pattern $\mathbf{a}$ is presented to ART^a, a category, say J, is selected according to (1) and if it is a newly committed category then $\rho_J = \rho^a$. The reset condition is evaluated using ρ_J in (4). If this condition is not satisfied, this node will be inhibited and a new search triggered. Pattern $\mathbf{b}$ is presented to ART^b, selecting the Kth category. Then the map field activity is calculated according to the PROBART equation (9).

1) Inter-ART Reset: After map field activity $\mathbf{x}^{ab}$ has been calculated, replacing p_j and p_{jk} in (12) by

$$\begin{aligned} p_{jk} &= \frac{x_k^{ab}}{|\mathbf{x}^{ab}|} \quad \text{if } j = J \\ p_j &= \frac{|\mathbf{x}^{ab}|}{|\mathbf{x}^{ab}| + \sum_{i=1, i\neq J}^{N^a} |\mathbf{w}_i^{ab}|} \\ p_{jk} &= \frac{w_{jk}^{ab}}{|\mathbf{w}_j^{ab}|} \quad \text{otherwise} \\ p_j &= \frac{|\mathbf{w}_j^{ab}|}{|\mathbf{x}^{ab}| + \sum_{i=1, i\neq J}^{N^a} |\mathbf{w}_i^{ab}|}. \end{aligned} \tag{13}$$

We can calculate h_J that represents the contribution to the total entropy of the Jth unit if it was allowed to learn this pattern. If $h_J > h_{\max}$ then this category is too entropic and thus the Jth node in ART^a is inhibited for the rest of this pattern presentation by setting $T_J(\mathbf{I}^a) = 0$, but its vigilance parameter is *not* raised. Other categories will be chosen in ART^a until the entropy contribution criterion is met. If a previously uncommitted category is selected, say J', then $p_{J'K} = 1$, while $p_{Jk} = 0$ for $k \neq K$ and therefore $h_J = 0$. Then weights in ART^a and ART^b are updated and also in the map field, by $\mathbf{w}_J^{ab} = \mathbf{x}^{ab}$.

2) Off-Line Evaluation: After all patterns have been processed, the off-line map field is initialized by $v_{jk}^{ab} = 0$, $j = 1, \ldots, N^a$, $k = 1, \ldots, N^b$ and the data are presented again to update these weights. However, this time the entropy contribution criterion is not evaluated, so that units are selected in ART^a in an unsupervised manner and weights in ART^a and ART^b are not updated. In fact, this is equivalent to making a test on the training data and storing the results in weights $\mathbf{v}_j^{ab}$. Replacing p_j and p_{jk} in (11) by

$$\begin{aligned} p_{jk} &= \frac{v_{jk}^{ab}}{|\mathbf{v}_j^{ab}|} \\ p_j &= \frac{|\mathbf{v}_j^{ab}|}{\sum_{i=1}^{N^a} |\mathbf{v}_i^{ab}|} \end{aligned} \tag{14}$$

the entropy H, is computed and compared to $H_{\max}$. If $H > H_{\max}$ then the mapping defined by μARTMAP between the input and output partitions is too entropic and thus a finer partitioning of the input space is necessary to improve predictive relations. To achieve this, the ART^a node J that has maximal contribution to the total entropy, $J = \arg\max_j h_j : j = 1, \ldots, N^a$, is searched. This node is removed (which means $\mathbf{w}_J^a = \mathbf{1}$ and $\mathbf{w}_J^{ab} = \mathbf{0}$), after the baseline vigilance is set to

$$\rho^a = \frac{|\mathbf{w}_J|}{M^a} = 1 - \frac{R_J}{M^a} + \Delta\rho \tag{15}$$

so that newly created categories will have smaller size than $|R_J|$, since the category size is bounded as shown in (8). All the patterns that previously selected the Jth ART^a category are presented again in a new training epoch, while the rest of the patterns are not. This will make a finer partition of the input space previously covered by the removed category, while the rest of the categories remain the same. The process carries on until $H < H_{\max}$.

μARTMAP Prediction : As in BARTMAP, μARTMAP prediction is carried out by selecting the Jth ART^a category node that has highest $T_J(\mathbf{I})$ value and then predicting the class label corresponding to the K_Jth ART^b category node, where $K_J = \arg\max_k \{w_{Jk}^{ab} : k = 1, \ldots, N^b\}$, i.e., K_J is the most frequent association to node J.

C. Discussion

If $\rho^a = 0$, $h_{\max} = 0$ and fast learning is assumed, the first training epoch of μARTMAP will generate as many ART^a categories as existing class labels, i.e., as ART^b categories. This means that all patterns associated to a given class label will lie inside the same ART^a hyperbox, which can be arbitrarily large.

The off-line evaluation will measure the probabilistic overlapping of the created hyperboxes. This is related to the number of patterns that select a different category when inter-ART reset is enabled and when it is disabled, which occurs because the inter-ART reset does not raise ART^a vigilance.

If patterns with different class labels lie apart in the input space, i.e., there is no overlapping, $H = 0$ and learning can be stopped. However, this overlapping will often be large, i.e., $H > H_{\max}$ and some of the categories must be refined. To refine a hyperbox, it is deleted and all patterns that previously selected it are presented again, but smaller hyperboxes are forced to cover the same region. Through this batch learning process, large hyperboxes are placed in regions where all patterns have the same class label, while small categories are placed in the boundaries between classes. In addition, *populated exceptions* can be handled with one large hyperbox, which is a general rule and one smaller hyperbox, which represent a specific rule.

Parameter $h_{\max}$ is intended to avoid that nonpopulated exceptions, i.e., outliers, create new single-point categories. Though most of the patterns that select one category will predict the same class label, by setting $h_{\max} > 0$ a few patterns with a different one can be allowed. In addition, gaussian noise can be controlled by setting $h_{\max} > 0$ and then tuning $H_{\max}$ so that it partitions again regions where noise is strong, as in the problems shown in Section V-C. In the limit, if $h_{\max} = \log_2 N^b$ μARTMAP suppresses the inter-ART reset and then behaves similarly to BARTMAP and if $H_{\max} = \log_2 N^b$ too, the off-line stage is not necessary and μARTMAP reduces to a PROBART network.

As in Fuzzy ARTMAP, μARTMAP rules can be extracted from the weights in the form

$$\texttt{IF } \mathbf{a} \texttt{ is } C_j \texttt{ THEN } \textit{output} \texttt{ is } L_k \quad (\texttt{priority } P_i) \tag{16}$$

where "$\mathbf{a}$ is C_j" means "pattern $\mathbf{a}$ selects the jth category" and L_k is the predicted label. The priority of the rule is the choice function (1), that reduces to an inverse proportionality to the hyperbox size, if patterns are inside hyperboxes. Considering this, μARTMAP algorithm is related to the way ID3 [16] constructs decision trees, if categories are the *attributes* on which rules are evaluated, as in (16). Initially, the most general rule (category with largest hyperbox) is evaluated. If the first rule is impure, ID3 adds an attribute that partitions the patterns in order to increment the information gain, while μARTMAP dynamically finds some category (another attribute) that augments the mutual information between input and output partitions. When entropy has been sufficiently reduced, both ID3 and μARTMAP training algorithms stop. Though μARTMAP does not generate a decision tree, its rules are constructed to be as general as possible, adding others with increasing specificity to refine the general rules.

V. Experimental Work

A comparative study of Fuzzy ARTMAP, μARTMAP and BARTMAP performance will be conduced on several benchmarks. Performance will be evaluated by the error rate on a test data set and by the number of categories generated, i.e., the number of rules that could be extracted. Therefore, the objective will be to test the capabilities of each architecture to reduce category proliferation, while preserving generalization. The first set of benchmarks will consist of variations of the well-known circle-in-the-square problem [17] that has been widely used in ARTMAP literature [6], [9], [10]. It will serve to illustrate the concept of *populated exception* and its effect on the training of the evaluated networks. In addition, the influence of the dimensionality of the input space will be assessed on a variation of this problem.

TABLE I
Committed Categories and Generalization Error for the Circle-in-the-Square Problem

classifier	rules	error
Fuzzy ARTMAP	25.2	5.69%
BARTMAP	39.1	6.83%
μARTMAP	9.9	5.24%

Another benchmark, with patterns generated by Gaussian sources, will test the performance when there is class overlap. As a particular cause for overlapping, the impact of additive noise will also be evaluated on the circle-in-the-square benchmark.

In addition, all networks will be evaluated in the difficult real-world task of on-line handwriting recognition, on UNIPEN [18] uppercase letters. In this problem, there is a definite need for a reduced set of comprehensible rules, that can be used for syntactic recognition, or for handwriting reconstruction [19].

In order to achieve maximal generalization, in all the experiments $\rho^a = 0$ and $\alpha = 0.001$ for the three networks, which will favor the creation of a smaller number of categories [20]. Fuzzy ARTMAP is trained until category stability is achieved, i.e., no more categories are created even if training continues for more epochs.

A. Circle in the Square

The circle-in-the-square problem [Fig. 3(a)] requires a system to decide whether points are inside or outside a circle lying within a square of twice its area [8]. This problem illustrates the concept of *populated exceptions* and there is not an optimum number of categories since decision boundaries cannot be described with a finite number of hyperboxes. Thus, the performance of Fuzzy ARTMAP, BARTMAP and μARTMAP was evaluated comparing both the number of committed categories, or generated rules and the generalization performance. For the experiments, data were generated randomly from an uniform source, to form ten 1000-point training sets and one single 10 000-point test set. Results are averaged in Table I, for BARTMAP trained with $\varepsilon_{\max} = 0.04$ and $\Delta\rho = 0.02$ and μARTMAP trained with $h_{\max} = 0.0$, $H_{\max} = 0.15$ and $\Delta\rho = 0.02$.

As shown in Fig. 3(c), BARTMAP must create a number of categories to cover the region surrounding the circle, since it cannot create hyperboxes inside others, due to the lack of an inter-ART reset mechanism. Though Fuzzy ARTMAP has an inter-ART reset mechanism, because the match tracking process always raises ART^a vigilance, smaller categories are

TABLE II
COMMITTED CATEGORIES AND GENERALIZATION ERROR FOR THE OVERLAPPING GAUSSIANS PROBLEM

classifier	rules	error
Fuzzy ARTMAP	27.0	6.44%
BARTMAP	16.1	6.29%
μARTMAP	11.6	5.45%

created. In addition, because Fuzzy ARTMAP must learn to correctly classify all training patterns, several categories are created along the circle boundary [see Fig. 3(b)], which improve very slightly generalization performance. Fig. 3(d) also shows how μARTMAP dedicates only one ART^a category to predict the class *outside*, while several categories are dedicated to describe the class *circle*, resulting in better generalization performance, while a reduced set of rules is generated.

In [10], dARTMAP is proposed to impact category proliferation and evaluated on the circle-in-the-square problem. When distributed learning is enabled, a pattern can be learned by several categories simultaneously, so that the input space need not be covered thoroughly. However, when the winning ART^a category node predicts the wrong class label, distributed learning is disabled and the network behaves like Fuzzy ARTMAP. This implies that ART^a vigilance can be raised, creating categories that are necessary but possibly of small relevance to the generalization error. In [10], dARTMAP is reported to use 10.8 categories to produce 7.9% generalization error on the circle-in-the-square problem. As it can be seen, μARTMAP uses a similar number of rules achieving higher test accuracy, by adequately positioning the hyperboxes and allowing some errors near class boundaries.

B. Overlapping Gaussians

In the previous experiment there is no overlap between classes. However, class overlap is a major cause of category proliferation in Fuzzy ARTMAP, since match tracking is often triggered and small categories are required to cover exceptions that are statistically unimportant. Consider the problem where points are generated from five Gaussian sources with means $\mu_1 = (0.5, 0.5)$, $\mu_2 = (0.2, 0.2)$, $\mu_3 = (0.2, 0.8)$, $\mu_4 = (0.8, 0.2)$, $\mu_5 = (0.8, 0.8)$ and deviation $\sigma_1 = \sigma_2 = \sigma_3 = \sigma_4 = \sigma_5 = (0.1, 0.1)$. Each source $N(\mu_2, \sigma_2)$, $N(\mu_3, \sigma_3)$, $N(\mu_4, \sigma_4)$ and $N(\mu_5, \sigma_5)$, has probability 1/8 and is associated to the same class label, while source $N(\mu_1, \sigma_1)$ has probability 1/2 and is associated to a different output class. Therefore, both classes have the same total probability. The geometry of this problem resembles the circle-in-the-square problem, but in this case no zero error decision boundary exists. For performance comparison, ten 1000-point datasets were generated and one single 10 000-point test set and all input patterns were normalized to the unit square. The results are shown in Table II, for BARTMAP trained with $\varepsilon_{\max} = 0.04$ and $\Delta\rho = 0.02$ and μARTMAP trained with $h_{\max} = 0.0$, $H_{\max} = 0.1$ and $\Delta\rho = 0.02$.

As seen in Fig. 4(c), BARTMAP can roughly describe source $N(\mu_1, \sigma_1)$ with a few hyperboxes, dedicating several more to the other sources, since it cannot represent source $N(\mu_1, \sigma_1)$ as a *populated exception*. Because of this, it generates more rules than μARTMAP. However, since both BARTMAP and μARTMAP allow some error in the training set, they do not commit categories to describe the multiple points of overlapping between classes and therefore generate more compact rule sets than Fuzzy ARTMAP and have superior generalization performance.

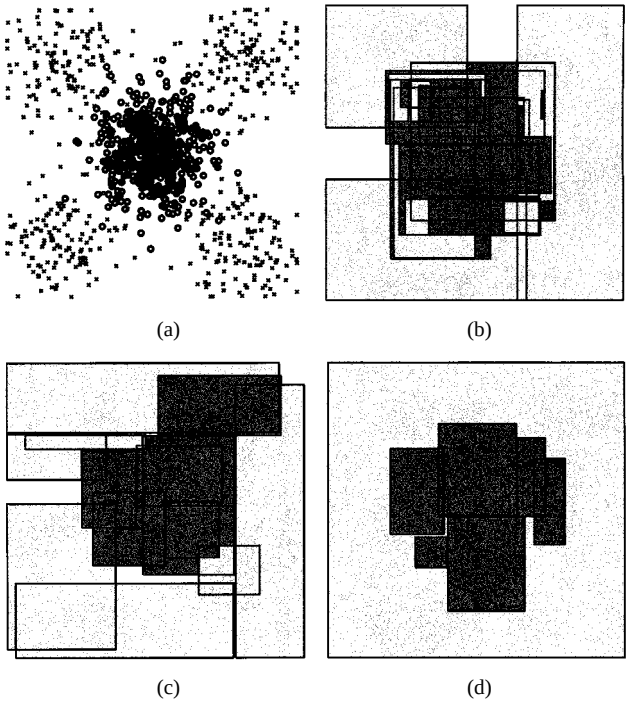

Fig. 4. (a) Patterns from five Gaussian sources, the four outermost associated to one class label and the inner to a different class label. (b), (c) and (d) show the hyperboxes created by Fuzzy ARTMAP, BARTMAP, and μARTMAP, respectively, for the simplest network structure among those resulting from the ten training sets.

C. Robustness to Noise

The presence of noise in the training data is one major cause of category proliferation in a fast-learning on-line system [9]. However, if there are just a few outliers, several single-point categories will be created, with little influence on the prediction error. If additive noise corrupts all data, decision boundaries are more vague and prediction will degrade. In this situation, class overlapping occurs and, as shown in the previous experiment, BARTMAP and μARTMAP can allow some error on the training set and thus it can be expected that they degrade less than Fuzzy ARTMAP due to additive noise.

To evaluate the impact of noise experimentally, the same data sets generated for the circle-in-the-square problem (Section V-A) were used and additive Gaussian noise added to the input patterns, i.e., $\mathbf{a}_n = \mathbf{a} + N(\mathbf{0}, \sigma)$. Different levels of noise were used, given by $\sigma_x = \sigma_y = k\dot{1}0^{-2}$, $k = 0, 1, \ldots, 10$. Parameters $\varepsilon_{\max}$ in BARTMAP and $h_{\max}$ and $H_{\max}$ in μARTMAP, were progressively relaxed as the level of noise increased, in order to avoid overfitting to noisy data.

Fig. 5 jointly plots the number of categories (abscissa) and the generalization error (ordinate). The lower left of this graph

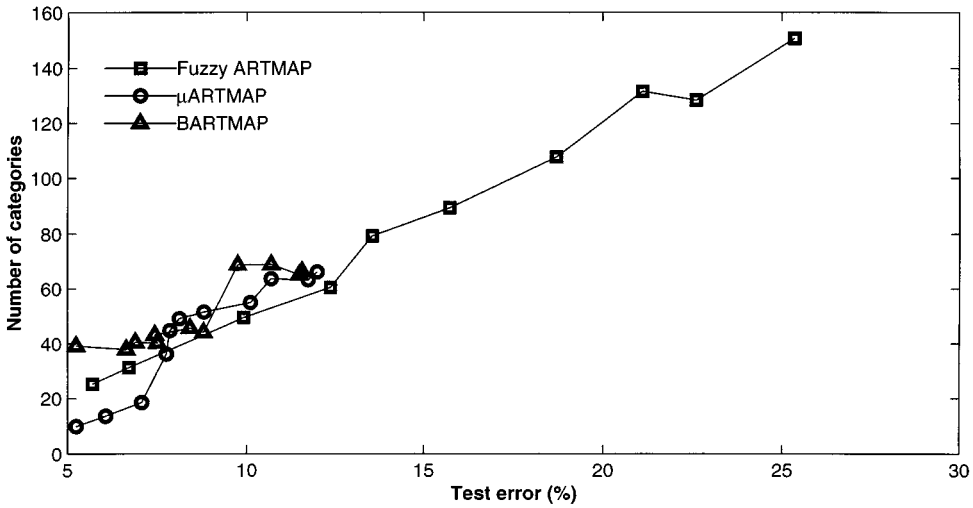

Fig. 5. From left to right along each curve, marks represent the number of categories versus the generalization error, for Gaussian noise added to the original data, of deviation $\sigma = k10^{-2}$, $k = 0, 1, \ldots, 10$.

is the desired performance region, where low error is achieved with few categories. All networks offer their best performance in the absence of noise and degrade as its level increases. This is especially noticeable for Fuzzy ARTMAP, that suffers strong category proliferation and accuracy losses. BARTMAP and μARTMAP are clearly more robust than Fuzzy ARTMAP, but μARTMAP degrades more with strong noise. When noise is low, one single category can be used to describe the *outside* class. However, if noise increases, categories with associated *inside* class label are placed outside the circle. To correct this effect, more categories predicting *outside* are generated. This is achieved by increasing $h_{\max}$. In fact, the last two simulations, $\sigma = 0.09$ and $\sigma = 0.1$, were carried out with $h_{\max} = 1$, i.e., without inter-ART reset mechanism and thus μARTMAP behaves similarly to BARTMAP.

D. Influence of Dimensionality

Performance of many statistical and machine learning algorithms degrades in problems with high dimensionality [21]. This is due to the fact that, as the number of dimensions increases, the input space will be sampled more sparsely. In addition, because Fuzzy ART categories are associated to hyperboxes, they can be inefficient for high dimensionality [9], since the hyperbox is defined by the minimum and maximum of its data and not by a tighter curve bound. Therefore, if sampling is sparse, the category infers the existence of data where no evidence exists. This may cause the recruitment of smaller categories at the corners associated to a different ART^b class label, resulting to poor generalization on new data.

Though it is convenient for rule interpretation to represent templates by hyperboxes, it must be assumed that performance degradation will occur for high dimensionality. This degradation can be evaluated by defining a series of problems of increasing dimensionality, M^a, but with similar geometry. Here we propose a generalization of the circle-in-the-square, named the hypersphere-centered-in-the-hypercube, i.e., it must be decided if points within the unit hypercube also lie or not inside a hypersphere cocentered with the hypercube. The radius of the hypersphere is selected so that its intersection with the hypercube has volume 1/2, while the hypercube itself has volume 1. For $M^a = 1, 2, 3$ the hypersphere is contained in the hypercube, while for larger M^a it is not. This implies that for $M^a > 3$ the "outside" class will not be connected. Its patterns distribute along the corners of the cube, which are smaller but many more as dimension increases. This problem maintains the main features through the different dimensions (equal probability to each class and an inner class surrounded by an outer class) and therefore can be used for this study. Experimentally, ten 1000-point training sets and one single 10 000-point test set were generated for each problem in the series, from $M^a = 1$ through $M^a = 10$. Note that the number of training samples is independent of M^a. Training parameters are those indicated above for the circle-in-the-square problem.

In Fig. 6, from left to right along each curve the number of categories (abscissa) and the generalization error (ordinate) are jointly plot, for $M^a = 1$ though $M^a = 10$. This graph clearly shows that performance degrades for all three networks as M^a increases, though μARTMAP always offers a better solution, achieving a lower error rate using fewer categories.

It is remarkable that, while relative degradation for μARTMAP and Fuzzy ARTMAP is similar, BARTMAP is severely affected. This is due to the lack of an inter-ART reset mechanism to allow placing hyperboxes inside others. Thus, many categories must be placed in the boundaries of the hypersphere [see Fig. 3(c)]. Since increasing dimensionality means a wider boundary, a larger number of categories need to be recruited. This example shows that handling *populated exceptions* correctly is important in concept learning problems defined on a high dimensional input space.

E. On-Line Handwriting Recognition

On-line handwriting recognition has been in the focus of research for many years [22]. Currently, it is a key issue in the development of wireless computing that requires small, easy to use

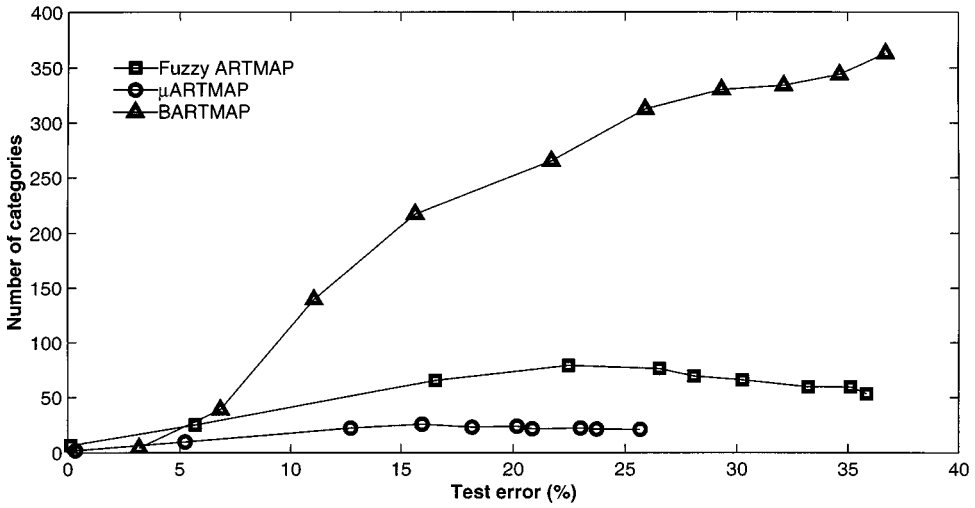

Fig. 6. From left to right along each curve, marks represent the number of categories vs. the generalization error, for the hypersphere-in-the-hypercube problem, an M^a-dimensional generalization of the circle-in-the-square problem, for $M^a = 1$ through $M^a = 10$.

devices [23]. Nevertheless, it presents intrinsic difficulties due to the variability existing among writers, languages, or and digitizing pads. Additionally, recognition of on-line written characters normally involves several tasks, including segmentation of sentences into words, words into characters and characters into strokes. This last step is motivated by biological models of handwriting generation. According to [24], a stroke is a piece of handwriting generated by a simple motor impulse to the hand and a component (handwriting between pen lifts) is made of a series of overlapping strokes. Besides segmentation, discriminant features must be extracted for constructing the input to the classifier.

Once handwriting data have been reduced to vectors of features, machine learning approaches can be taken to build a classifier [25]. However, in order to better understand the human capability for both recognition and generation tasks, it is useful to build a syntactic recognizer with a reduced number of rules [19], as noted by the many different research approaches made to this problem (e.g., [26]). For this purpose, Fuzzy ARTMAP and especially μARTMAP, can be used.

For the experiment shown here, data were taken from the *train_r01_v02* UNIPEN data release. The UNIPEN project [18] has collected more than 5 000 000 characters, from many writers, languages, and pads, so that conclusions can be general enough. Here 2106 samples were selected to build the training set, while 2092 different samples form the test set, provided that all writers contribute to both sets and samples are restricted to be upper case letters, i.e., there are 26 class labels, though similar conclusions can be extracted from the recognition of digits or isolated lower case letters. Characters were segmented using velocity minima, as inspired by biological models [24] and 11 features were extracted for each stroke: length, three angles that describe the curvature of the stroke (each angle is represented by its sine and cosine and therefore six features are required), last y coordinate, mean x and mean y values of the strokes coordinates and a discrete feature indicating if the stroke starts and/or ends a component. The feature vector corresponding to a character is made by the sum of the features of its strokes, plus one additional feature, the ratio between the sides of the box containing the whole character. For more details see [25].

TABLE III
TOTAL NUMBER OF RULES AND AVERAGE ERROR RATE FOR THE RECOGNITION OF ON-LINE HANDWRITTEN UPPER CASE LETTERS

classifier	rules	error
Fuzzy ARTMAP	254	6.02%
BARTMAP	489	6.74%
μARTMAP	105	7.03%

Since training samples have different numbers of strokes, six different networks are trained, with network n trained only on samples with n strokes, $n = 1, \ldots, 6$. Therefore, the dimension of input vectors is different for each network, namely $11n+1$. If a character has more than six strokes it is considered badly segmented and counted as a wrong prediction. All networks were trained, with $h_{\max} = 0.0$, $H_{\max} = 0.2$ and $\Delta\rho = 0.02$ for μARTMAP and $\varepsilon_{\max} = 0.04$, $\Delta\rho = 0.02$ for BARTMAP.

In this difficult task, given a test pattern each network will provide a ranked list of all possible class labels. This information can be used by a postprocessing algorithm using contextual information, like [27], where a syllabic dictionary is employed. Therefore, in this work a prediction will be considered correct if the expected class label is among the first two predicted. Table III shows total number of rules, comprising the six networks (each devoted to characters of a given number of strokes) and the average rate of the expected class label not being among the first two ranked by the classifier.

Fuzzy ARTMAP achieves a high accuracy, but it commits a high number of categories, i.e., it generates a large rule set. On the contrary, μARTMAP achieves slightly lower recognition rates with a much simpler set of rules. Considering that there are 26 output class labels, an average of four rules for class label is generated, while Fuzzy ARTMAP dedicates an average of ten.

This can be explained considering that, due to the high dimensionality of the problems and the variability of handwriting, pat-

terns with the same class label are distributed in several "clouds" in the input space, which can be seen as case of multiple *populated exceptions*. In addition, *isolated exceptions* appear if one writer contributes with very few samples, or he is unstable or uncomfortable writing on the digitizing pad, or some characters are badly labeled. By allowing hyperboxes be as large as necessary, but accepting small a training error, μARTMAP generates such a compact rule set. In addition, since Fuzzy ARTMAP distributes training samples among several categories, μARTMAP is a better estimator of the underlying distribution. Thus, it will be simpler to apply rule pruning by usage frequency [7] to their rules than to those generated by Fuzzy ARTMAP.

BARTMAP accuracy lies between that of Fuzzy ARTMAP and μARTMAP, but at the expense of a large number of categories. This is due to the appearance of many *populated exceptions*, as already mentioned. In these high-dimensionality input spaces, many categories are devoted to describe the surrounding of these *populated exceptions*. In fact, BARTMAP performance degrades as the number of strokes, and thus the dimensionality of the problem, increases, pointing out the utility of some kind of inter-ART reset.

VI. Conclusion

A new neural architecture called μARTMAP has been introduced as a solution to the category proliferation problem sometimes present in Fuzzy ARTMAP-based architectures. It then reduces the number of committed categories, while preserving generalization performance, without changing the geometry of category representation. Therefore, a compact set of IF-THEN rules can be easily extracted. This is important for favoring the use of neural networks in problems where comprehensibility of decisions is required, or where it is important to gain insight into the problem through the data.

To achieve this category reduction, μARTMAP intelligently positions hyperboxes in the input space and optimizes their size. For this purpose, two different learning stages are considered: in the first stage an inter-ART reset mechanism is fired if selected ART^a category has an entropic prediction. However, ART^a vigilance is not raised. In the second stage, total prediction entropy is evaluated and, if required, some patterns are presented again with increased ART^a vigilance values. This way, μARTMAP allows some training error, avoiding committing categories with small relevance for generalization and also permits placing hyperboxes inside other hyperboxes, to describe efficiently *populated exceptions*, i.e., problems where many patterns associated to one class label are surrounded by many others associated to a different one.

Experimental results obtained on synthetic benchmarks show that an inter-ART reset mechanism is necessary for treating correctly these *populated exceptions*. In μARTMAP, vigilance in ART^a is not raised after inter-ART reset and therefore this mechanism does not cause category proliferation, while the predictive accuracy can be guaranteed by the second learning stage. Furthermore, some kind of inter-ART reset mechanism turns out to be more significant in higher dimensionalities, since otherwise an increasingly large number of categories will be devoted to describe *populated exceptions*. Thus μARTMAP has been shown to outperform BARTMAP, another ARTMAP-based approach to reduce category proliferation that suppresses the inter-ART reset.

In addition, because μARTMAP, as BARTMAP, allows a small error on the training set, it finds more compact rule sets when there is overlap between concept classes and therefore no exact solution. This results generalizes in μARTMAP and BARTMAP being more robust to noise than Fuzzy ARTMAP.

Furthermore, μARTMAP has been tested in a difficult real-world task, i.e., recognizing upper-case letters written on-line on a digitizing pad, where the extraction of a reduced set of rules is very important. Because of the high variability of the data, patterns are organized as many "clouds" in an input space of high dimensionality, where many of these clouds are surrounded by patterns with other labels, i.e., *populated exceptions*. In this situation, μARTMAP significantly reduces the number of generated rules, to achieve similar performance. In addition, these rules reflect more reliably the underlying distribution of the data and thus postprocessing methods could be more efficient. On the contrary, BARTMAP fails to produce a reduced number of rules because the lack of an inter-ART reset mechanism becomes critical in this high-dimensional problem.

Current research pursues modifying μARTMAP to control category growth on each input feature independently. This is interesting because the vigilance criterion (4) limits the total size of the hyperbox, while *a priori* knowledge, or the underlying distribution, may determine that restriction should be applied only in some particular direction. By doing this, a smaller number of categories would be recruited in some problems, while gaining independence of the order of pattern presentation and an indirect measure of feature importance could be derived.

In addition, an interesting topic of ongoing research to reduce category proliferation concerns the assessment of modified architectures, such as dARTMAP, BARTMAP or the proposed μARTMAP, as compared to rule pruning or extraction methods. In some cases some of the rules generated by Fuzzy ARTMAP may contribute little to the predictive accuracy and thus could be removed, yielding a network with a compact set of rules, but preserving the on-line feature. In [28] we partially address the study of the computational implications and effectiveness to reduce category proliferation of rule pruning methods, while more extended research is an important issue for future works.

Acknowledgment

The authors would like to thank M. Araúzo-Bravo, E. Parrado-Hernández, M. Martín Marino-Acera and M. Bote-Lorenzo, for their suggestions during the preparation of this paper. We would also like to thank the comments of the reviewers on the first draft and also those of Dr. G. Heileman, that significantly helped to improve the paper.

References

[1] J. W. Shavlik, R. J. Mooney, and G. G. Towell, "Symbolic and neural learning algorithms: An experimental comparison," *Machine Learning*, vol. 6, pp. 111–143, 1991.

[2] M. W. Craven, "Extracting Comprehensible Models From Trained Neural Networks," Ph.D. dissertation, Dept. Comput. Sci., Univ. Wisconsin, Madison, 1996.

[3] K. J. Cherkauer and J. W. Shavlik, "Selecting salient features for machine learning from large candidate pools through parallel decision-tree construction," in *Massively Parallel Artificial Intelligence*, H. Kitano, Ed. Menlo Park, CA: AAAI Press/MIT Press, 1993, pp. 102–136.
[4] C. G. Towell, J. W. Shavlik, and M. O. Noordewier, "Refinement of approximately correct domain theories by knowledge-based neural networks," in *Proc. 8th Nat. Conf. Artificial Intell.*, Boston, MA, 1990, pp. 861–866.
[5] M. W. Craven and J. W. Shavlik, "Learning symbolic rules using artificial neural networks," in *Proc. 10th Int. Joint Conf. Machine Learning*, M. Kaufmann, Ed., Amherst, MA, 1993, pp. 73–80.
[6] G. A. Carpenter, S. Grossberg, N. Markuzon, J. 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, Sept. 1992.
[7] A. Carpenter and H. A. Tan, "Rule extraction: From neural architecture to symbolic representation," *Connection Sci.*, vol. 7, no. 1, pp. 3–27, 1995.
[8] G. A. Carpenter, S. Grossberg, and D. B. Rosen, "Fuzzy ART: Fast stable learning and categorization of analog patterns by an adaptive resonance system," *Neural Networks*, vol. 4, no. 1, pp. 759–771, 1991.
[9] J. Williamson, "Gaussian ARTMAP: A neural network for fast incremental learning of noisy multidimensional maps," *Neural Networks*, vol. 9, no. 5, pp. 881–897, 1996.
[10] G. A. Carpenter, B. L. Milenova, and B. W. Noeske, "Distributed ARTMAP: A neural network for fast distributed supervised learning," *Neural Networks*, vol. 11, no. 4, pp. 793–813, 1998.
[11] E. Parrado-Hernández, E. Gómez-Sánchez, Y. A. Dimitriadis, and J. L. Coronado, "A neuro-fuzzy sytem that uses distributed learning for compact rule set generation," in *Proc. IEEE Conf. Syst., Man, Cybern., SMC99*, vol. 3, Tokyo, Japan, Oct. 1999, pp. 441–446.
[12] S. J. Verzi, G. L. Heileman, M. Georgiopoulos, and M. J. Healy, "Boosted ARTMAP," in *Proc. IEEE World Congr. Comput. Intell., WCCI'98*, Anchorage, AK, May 1998, pp. 396–400.
[13] L. A. Zadeh, "Fuzzy sets," *Inform. Contr.*, vol. 8, pp. 338–353, June 1965.
[14] S. Marriott and R. Harrison, "A modified Fuzzy ARTMAP architecture for the approximation of noisy mappings," *Neural Networks*, vol. 8, no. 4, pp. 619–641, 1995.
[15] A. Papoulis, *Probability, Random Variables and Stochastic Processes*, 3rd ed: McGraw-Hill, 1991.
[16] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, 1986.
[17] K. J. Lang and M. J. Witbrock, "Learning to tell two spirals apart," in *Proc. 1988 Connectionist Models Summer School*, D. Touretzky, G. Hinton, and T. Sejnowski, Eds., San Mateo, CA, 1989, pp. 52–59.
[18] I. Guyon, L. Schomaker, R. Plamondon, M. Liberman, and S. Janet, "UNIPEN: Project of on-line data exchange and recognizer benchmarks," in *Proc. 12th Int. Conf. Pattern Recognition*, Jerusalem, Israel, October 1994, pp. 9–13.
[19] M. Parizeau and R. Plamondon, "A handwriting model for syntactic recognition of cursive script," in *Proc. 11th Int. Conf. Pattern Recognition*, vol. 2, The Hague, Netherlands, 1992, pp. 308–312.
[20] M. Georgiopoulos, H. Fernlund, G. Bebis, and G. Heileman, "Order of search in fuzzy art and fuzzy ARTMAP: Effect of the choice parameter," *Neural Networks*, vol. 9, no. 9, pp. 1541–1559, 1996.
[21] A. K. Jain, R. P. W. Duin, and J. Mao, "Statistical pattern recognition: A review," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 22, pp. 4–37, Jan. 2000.
[22] R. Plamondon and S. N. Srihari, "On-line and off-line handwritting recognition: a comprehensive survery," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 22, pp. 63–84, d, Jan. 2000.
[23] R. Want and G. Borriello, "Survey on information appliances," *IEEE Comput. Graphics Applicat.*, vol. 20, no. 3, pp. 24–31, May 2000.
[24] R. Plamondon, "A kinematic theory of rapid human movements. Part 1: Movement representation and generation," *Biol. Cybern.*, no. 72, pp. 295–307, 1995.
[25] E. Gómez-Sánchez, J. A. Gago-González, Y. A. Dimitriadis, J. M. Cano-Izquierdo, and J. L. Coronado, "Experimental study of a novel neuro-fuzzy system for on-line handwritten recognition," *Pattern Recognition Lett.*, vol. 19, no. 3, pp. 357–364, Mar. 1998.
[26] L. Duneau and B. Dorizzi, "Incremental building of an allograph lexicon," in *Advances in Handwriting an Drawing: A Multidisciplinary Aproach*, C. Fanre, P. Kenss, G. Lorette, and A. Vinter, Eds., Europia, 1994, pp. 39–54.
[27] F. P. Merino, Y. A. Dimitriadis, R. G. García, and J. C. López, "A dictionary-based neural network scheme for on-line handwriting recognition," in *Handwriting and Drawing Research: Basic and Applied Issues*, M. Simner, C. Leedham, and A. Thomassen, Eds. Amsterdam, The Netherlands: IOS Press, 1996, pp. 343–358.
[28] E. Parrado-Hernández, E. Gomez-Sánchez, and Y. A. Dimitriadis, "Study of distributed learning as a solution to category proliferation in fuzzy ARTMAP based neural systems," *Neural Networks*, 2001.

Eduardo Gómez-Sánchez (S'96–M'02) received the M.S. and Ph.D. degrees in telecommunications engineering from the University of Valladolid, Valladolid, Spain, in 1996 and 2001, respectively.

He is currently an Assistant Professor at the Department of Signal Theory, Communications and Telematics Engineering, University of Valladolid. His research interests include ART neural networks and their application to pattern recognition and function approximation, as well as the use of neural networks for knowledge extractions.

Dr. Gómez-Sánchez is a member of the ENNS.

Yannis A. Dimitriadis (M'93) received the engineering degree from the National Technical University of Athens, Greece, in 1981, the M.S. from the University of Virginia, in 1983, and the doctoral degrees from the University of Valladolid, Spain, 1992 and 1995, all of them in telecommunications engineering.

He is currently an Associate Professor at the Department of Signal Theory, Communications and Telematics Engineering, University of Valladolid. His research interests include neuro-fuzzy systems, pattern recognition, and computer supported collaborative learning.

Dr. Dimitriadis is member of the ACM and of the editorial board of Applied Intelligence, Kluwer Academic.

José Manuel Cano-Izquierdo (S'96–A'97) received the M.S. and Ph.D. degrees in industrial engineering from the University of Valladolid, Spain, in 1993 and 1997, respectively.

He is currently an Associate Professor at the Department of System Engineering and Automatic Control, Polytechnical University of Cartagena, Spain. His research interests include ART neural networks and fuzzy systems and their application to system identification and control.

Dr. Cano-Izquierdo is member of the CEA-IFAC and EUSFLAT.

Juan López-Coronado received the engineering degree from the Polytechnical University of Barcelona, Barcelona, Spain, in 1974, the doctoral degrees from the Aerospace High National Engineering School (E.N.S.A.E), Toulouse, France, in systems engineering and advanced automation in 1981 and from the University of Valladolid, Spain, in industrial engineering in 1985.

He was at the University of Valladolid as an Associate Professor leading a research group on computer vision and applied artificial intelligence. Since 1998, he is a Full Professor at the Politechnical University of Cartagena, Spain, where he is the chair of the Department of System Engineering and Automatic Control and the chair of the Neurotechnology, Control and Robotics Laboratory (NEUROCOR). His research interests include neuro-fuzzy systems and their application to pattern recognition, control, and identification of nonlinear systems as well as neurobiologically inspired models for vision and motor control and their application to robotics and general industrial problems.

Dr. López-Coronado is member of the CEA-IFAC.