Constructive Feedforward ART Clustering Networks—Part I

Andrea Baraldi and Ethem Alpaydın

***Abstract*—Part I of this paper proposes a definition of the adaptive resonance theory (ART) class of constructive unsupervised on-line learning clustering networks. Class ART generalizes several well-known clustering models, e.g., ART 1, improved ART 1, adaptive Hamming net (AHN), and Fuzzy ART, which are optimized in terms of memory storage and/or computation time. Next, the symmetric Fuzzy ART (S-Fuzzy ART) network is presented as a possible improvement over Fuzzy ART. As a generalization of S-Fuzzy ART, the simplified adaptive resonance theory (SART) group of ART algorithms is defined. Gaussian ART (GART), which is found in the literature, is presented as one more instance of class SART. In Part II of this work, a novel SART network, called fully self-organizing SART (FOSART), is proposed and compared with Fuzzy ART, S-Fuzzy ART, GART and other well-known clustering algorithms. Results of our comparison may easily extend to the ARTMAP supervised learning framework.**

***Index Terms*—Absolute and relative membership function, adaptive resonance theory (ART), clustering, hard-and-soft competitive learning, pruning, reinforcement learning, unsupervised learning, Voronoi partition.**

I. Introduction

ALL NATURAL systems provided with cognitive capabilities feature feedback interaction with their external environment. Owing to this environmental feedback, natural systems weaken or reinforce their behaviors as a function of their success [1], [2]. Mimicking the real world, an artificial cognitive system employing reinforcement learning "is allowed to react to each training case; it is then told whether its reaction was good or bad" [3], "but no actual desired values are given" [4].

One example of artificial reinforcement learning can be found in the adaptive resonance theory (ART) class of clustering algorithms, e.g., ART 1 [5], Improved ART 1 (IART 1) [6], adaptive Hamming net (AHN) [7] and Fuzzy ART [8], whose origins go back to several 1976 pioneering papers in neural network history by Grossberg [9], [10]. In ART clustering networks, an orienting subsystem models some external evaluation of the pattern-matching reaction of the attentional subsystem to an input stimulus [11], [12]. The *a priori* knowledge exploited by the ART orienting subsystem consists of a user-defined vigilance threshold which is a relative number equivalent to a lower limit on the acceptable quality of the pattern recognition activity performed by the attentional subsystem. For example, in ART 1-based systems,[1] the vigilance threshold is equivalent to an upper limit on the size of the cluster region of support in input space.

In recent years, several ART 1-based models have been presented. It is well known that ART 1, which categorizes binary patterns, is sensitive to the order of presentation of the random sequence [5], [6]. This finding led to the development of IART 1, which also applies to binary patterns but is less dependent than ART 1 on the data set input presentation [6]. The AHN, which is a binary feedforward network that may employ a parallel implementation of its output stage (MAXNET), is functionally equivalent to ART 1 and optimizes ART 1 both in terms of computation time and memory storage [7]. ART 2, designed to detect regularities in analog random sequences, employs a computationally expensive architecture which presents difficulties in parameter selection [11]. To overcome these difficulties, the Fuzzy ART system was presented as a generalization of ART 1 to process binary as well as analog pattern distributions [8]. To deal with supervised learning tasks, the so-called Fuzzy ARTMAP classifier was developed around the combination of two Fuzzy ART modules. Fuzzy ARTMAP, which was shown to perform well in several benchmarks with respect to other supervised learning systems [12]–[14], is still widely employed in several application fields [12], [15], [16]. To reduce the sensitivity of Fuzzy ARTMAP to the order of training samples, output combinations of independently trained Fuzzy ARTMAP systems were proposed [12], [13].

Our conjecture is that the sensitivity of Fuzzy ARTMAP may be, at least in part, a legacy of Fuzzy ART. In other words, since Fuzzy ARTMAP employs two Fuzzy ART modules, we expect that ART 1 structural problems, if any, may affect Fuzzy ART and, as a consequence, Fuzzy ARTMAP. This conjecture is supported by the analysis of the potential weaknesses of Fuzzy ART conducted by Williamson [13], whose considerations led to the development of the supervised learning Gaussian ARTMAP (GAM), based on the unsupervised learning Gaussian ART (GART) module. In [13], [14], GAM was shown to be more accurate and less sensitive to the order of training samples than Fuzzy ARTMAP.

According to Backer and Jain, "in cluster analysis a group of objects is split up into a number of more or less homogeneous subgroups on the basis of an often subjectively chosen measure of similarity, such that the similarity between objects within a subgroup is larger than the similarity between objects belonging to different subgroups" [18]. And since the goal of clustering is to group the data at hand rather than provide an accurate char-

Manuscript received May 3, 1999; revised February 8, 2001.

A. Baraldi is with ICSI, Berkeley, CA, and ISAO-CNR, Bologna, Italy.

E. Alpaydın is with ICSI, Berkeley, CA, and Boğaziçi University, Istanbul, Turkey.

Publisher Item Identifier S 1045-9227(02)04451-X.

[1]For the sake of simplicity, we will further refer to ART 1, IART 1, AHN, and Fuzzy ART as ART 1-based algorithms.

acterization of unobserved (future) samples generated from the same probability distribution, the task of clustering may fall outside the framework of predictive (inductive) learning problems, such as vector quantization [19].

The subjective nature of the clustering problem precludes an absolute judgement as to the relative efficacy of all clustering techniques [18]. In line with this principle, the goal of this paper is not to choose the "best" clustering technique (such a task would be contrary to the very nature of clustering), but rather to emphasize those functional aspects most important to the user such as robustness to changes in input parameters, sensitivity to the order of the data set input presentation, accuracy, computation time, domain of applicability, etc., that may characterize the algorithms belonging to the class of ART clustering networks. Although focused on ART clustering networks, such as ART 1 and Fuzzy ART, our analysis may provide new insights into the understanding of the Fuzzy ARTMAP classifier.

Part I of this paper provides a definition of the class of ART clustering networks which is capable of generalizing the group of ART 1-based algorithms (see footnote 1). Based on this definition, ART 1, IART 1, and Fuzzy ART are optimized in terms of computation time and memory storage, while structural problems of ART 1-based algorithms are highlighted. Next, the symmetric Fuzzy ART (S-Fuzzy ART) network is proposed and discussed as a possible improvement over Fuzzy ART. As a generalization of S-Fuzzy ART, the group of simplified ART (SART) algorithms is defined. The GART clustering model, which is a Gaussian maximum-likelihood (ML) probability density function estimator found in the literature [13], is presented as one more instance of class SART.

In Part II of this paper, a constructive, on-line learning, topology-preserving, soft-to-hard competitive, minimum-distance-to-means clustering network, belonging to class SART and termed fully self-organizing SART (FOSART), is proposed as a new synthesis between properties of Fuzzy ART and other successful clustering algorithms such as the self-organizing map (SOM) [20], [21], and neural gas (NG) [22], to extend the capabilities of these separate approaches.

Part I of this paper is organized as follows: in Section II, the class of ART clustering networks is defined. In Section III, ART 1-based algorithms are interpreted in the light of the general ART framework proposed in Section II. Fuzzy ART is discussed in Section IV and S-Fuzzy ART is presented in Section V. Section VI presents an experimental comparison between Fuzzy ART and S-Fuzzy ART. In Section VII, generalization of the S-Fuzzy ART model leads to the definition of the class of SART clustering networks. Conclusions are reported in Section VIII.

II. The Class of ART Clustering Networks

Based on our interpretation and generalization of the AHN reformulation of the ART 1 clustering algorithm [7], this section proposes our definition of the class of ART clustering networks that generalizes and optimizes, in terms of memory storage, well-known clustering algorithms such as: 1) binary ART 1 [5]; 2) binary IART 1 [6]; 3) binary and feedforward AHN, shown in Fig. 1, which is functionally equivalent to ART 1 [7]; and 4) analog and feedforward Fuzzy ART [8]. In combination with the definition of class ART we also propose two versions of an efficient ART (EART) implementation scheme capable of optimizing ART networks in terms of computation time.

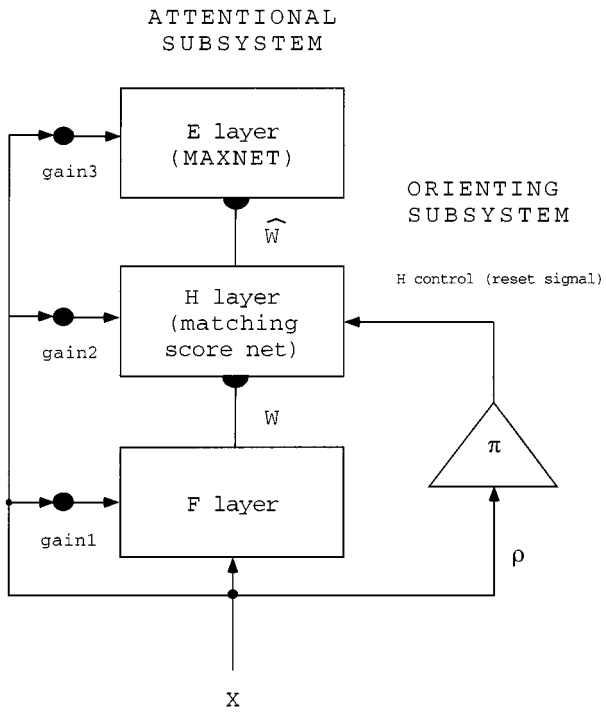

Fig. 1. AHN system. For details on the meaning of threshold π and weights $\mathbf{W}$ and $\hat{\mathbf{W}}$, refer to bibliography [7].

Class ART of clustering networks and class EART (version 1 and 2) of ART implementation schemes will be employed as general frameworks in this paper.

A. ART Optimization Problem

Let us consider, at presentation time t, an unlabeled input vector $\mathbf{X}^{(t)} \in \mathcal{D}^d$, where d is the dimensionality of input space, while domain $\mathcal{D} = \mathcal{R}$ in the analog case, or $\mathcal{D} = \{0, 1\}$ in the binary case. This input vector is processed by an unsupervised single-layer feedforward constructive clustering network, consisting of a layer of d input units F_k, $k = 1, \ldots, d$, fully connected to an output layer of processing elements (PEs, also termed output nodes, categories, components, or clusters) $E_j^{(t)}$, $j = 1, \ldots, c(t)$, where network size $c(t)$ may increase with time. Structural properties of the output node $E_j^{(t)}$ at time t are parameterized by a parameter (weight) vector learned from the data (also called cluster prototype or template) $\mathbf{W_j}^{(t)} \in \mathcal{D}^p$, $j = 1, \ldots, c(t)$, where $p \geq d$ is the dimensionality of parameter space.

In line with the AHN reformulation of the ART 1 clustering algorithm [7], we define an ART clustering scheme as an optimization problem where the best-matching unit at time t, $E_{w1(t)}^{(t)}$, is the solution, if any, that maximizes expression [7]

$$w1(t) = \arg \max_{j=1, \ldots, c(t)} \left\{ AF_{ART}\left(\mathbf{X}^{(t)}, \mathbf{W_j}^{(t)}\right)\right\} \quad (1)$$

where $AF_{ART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)})$, called activation function (AF), is a mapping

$$AF_{ART}(\mathbf{X}, \mathbf{W}): \mathcal{D}^d \times \mathcal{D}^p \rightarrow \mathcal{R}_0^+ \tag{2}$$

equivalent to a "compatibility" (i.e., typicality, membership) measure between data point $\mathbf{X}$ and cluster model $\mathbf{W}$, subject to constraint

$$MF_{ART}\left(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{w1(t)}}^{(t)}\right) \geq \rho, \qquad \rho \in [0, 1] \tag{3}$$

where vigilance threshold ρ, which is a user-defined relative number, *provides a model of top-down external expectations*, while $MF_{ART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{w1(t)}}^{(t)})$, called match function (MF), is a mapping

$$MF_{ART}(\mathbf{X}, \mathbf{W}): \mathcal{D}^d \times \mathcal{D}^p \rightarrow [0, 1] \tag{4}$$

equivalent to a normal "compatibility" measure between data point $\mathbf{X}$ and cluster model $\mathbf{W}$. The purpose of inequality (3) is to detect whether pattern $\mathbf{X}^{(t)}$ is an outlier, i.e., whether input data is very far from the ensemble of clusters at time t. In general, activation and match functions (2) and (4) may or may not be the same function.

In our view, the modular architecture of all ART clustering algorithms consists of (see Fig. 2):

i) a completely generic unsupervised single-layer feedforward (bottom-up) pattern recognition network, termed *attentional subsystem*, consisting of processing elements (PEs) which perform according to (1). This definition is not obvious if we consider that the "bidirectional" functional interpretation of ART 1, employing top-down as well as bottom-up adaptive weights [5], [6], still holds in recent papers [15], [16]. Exploitation of "unidirectional" rather than "bidirectional" adaptive weights guarantees an optimization, in terms of memory storage, of traditional "bidirectional" algorithms like ART 1 and IART 1 (see Appendix I) [17].

ii) An *orienting subsystem*, centered on inequality (3), equivalent to an interface between supervised and unsupervised knowledge, where the quality of unsupervised bottom-up pattern recognition is compared to top-down requirements (expectations, or prior knowledge) provided by the external environment (supervisor) in the form of an adimensional relative vigilance threshold $\rho \in [0, 1]$.

In the orienting subsystem, according to an example-driven mechanism [23], if (3) is satisfied, i.e., if unsupervised knowledge matches external expectations, then "resonance" occurs. In this case the unsupervised pattern recognition activity of the attentional module is reinforced (see Section I) by means of suitable prototype adaptation strategies.

Vice versa, if resonance does not occur, an outlier is detected and the orienting subsystem allows the attentional module to dynamically increase its resources (processing elements) to meet external requirements. In particular, when (3) is not satisfied at time t, then, at time $(t + 1)$, parameter adaptation is restricted to a specific parameter subspace consisting of one single category specifically generated to satisfy (3). This network growing strategy: 1) aims at avoiding the "probabilistic (relative) membership problem" where, at time $(t + 1)$, template parameters existing at time t are affected by an outlier detected at time t [24], [25],[2] and 2) should be combined with a noise category removal mechanism, which is straightforward to add to ART architectures [13], [27].

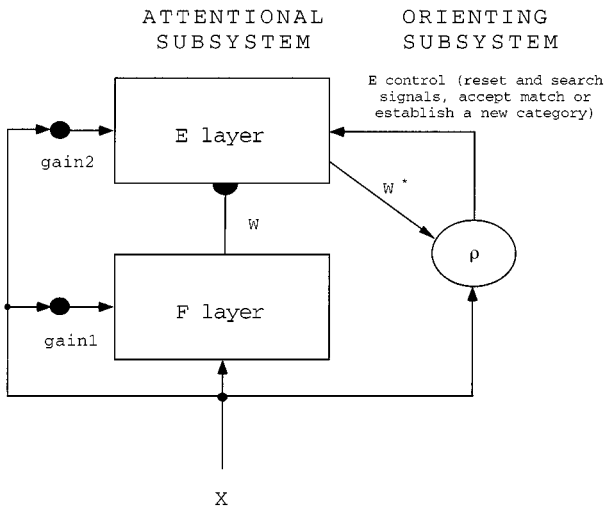

Fig. 2. ART and SART system architecture, where $\mathbf{W}$ identifies a matrix of bottom-up connections and $\mathbf{W}^*$ is the best-matching template. Unlike the common interpretation of ART 1-based systems, notice that no top-down connection is involved. For more details, refer to the text.

Supervision by the orienting subsystem over attentional activities is such that coarser partitions of input space are pursued when vigilance parameter ρ is lowered in (3). This means that the vigilance threshold is employed as a lower bound on a normal degree of "compatibility" (membership) between an input vector and a category structure pair.

Note that, in our ART attentional subsystem, provided with feedforward connections exclusively, the meaning of the term "resonance" is in contrast with that traditionally employed in the ART literature [5], [6], [8], [15], [16]. This term should no longer indicate "the basic feature of all ART systems, notably, pattern-matching between bottom-up input and top-down learned prototype vectors" [8, p. 760], just as the term "resonance" has never been applied to pattern matching activities performed by feedforward clustering networks, e.g., SOM or NG, where no top-down prototype vector does exist. In our view of ART systems, the term "resonance" means rather that if, in the orienting subsystem, unsupervised knowledge matches external (prior) expectations, then, in the attentional subsystem, successful pattern recognition activities are reinforced by means of prototype vector adaptation mechanisms.

B. Optimized Implementation of ART Clustering Networks

It is interesting to observe that the only structural difference between AHN and ART 1, IART 1 and Fuzzy ART is that AHN executes (3) first and (1) second, while the latter algorithms execute the same pair of operations in reverse order. This architectural difference allows AHN to detect, at time t, the

[2]In fuzzy set theory, an outlier tends to have small "possibilistic" (absolute) membership values with respect to all category structures, while its "probabilistic" (relative) membership values may be high [24]–[26].

TABLE I
PROPERTIES OF ART CLUSTERING NETWORKS

	ART 1	IART 1	AHN	Fuzzy ART	S-Fuzzy ART	GART	FOSART
EART version	1	1	1	1	2	1	2
class SART	No	No	No	No	Yes	Yes	Yes

best-matching unit, $E^{(t)}_{w1(t)}$, if any, by searching once through the $c(t)$ activation values whatever the input pattern may be, whereas the traditional implementation of ART 1, IART 1, and Fuzzy ART requires one up to $c(t)$ searches through activation values, managed by a so-called "mismatch reset condition and repeated search process" mechanism [5]–[8], [15], [16]. To summarize, binary AHN and ART 1 clustering models are functionally equivalent, but AHN, which employs feedforward connections exclusively, is more efficient in terms of computation time and memory storage [7].

Let us call the AHN-based optimal implementation of the ART maximization problem, defined by (1) and (2), the computationally efficient ART (EART) implementation scheme. Unlike traditional implementations of ART 1, IART 1, and Fuzzy ART, and in line with AHN, EART employs no time-consuming "mismatch reset condition and repeated search process." Depending on properties of activation and match functions, we propose two versions of the EART implementation scheme. Table I shows the relationship between versions 1 and 2 of the EART processing scheme with ART networks discussed in this paper.

1) Version 1 of the EART Implementation Scheme: EART version 1 is equivalent to the sequential version of the parallel AHN processing scheme. It holds when match function (3) does not increase monotonically with activation function (2), i.e., if $AF_{ART}(\mathbf{X}^{(t)}, \mathbf{W}^{(t)}_{\mathbf{j}}) > AF_{ART}(\mathbf{X}^{(t)}, \mathbf{W}^{(t)}_{\mathbf{h}})$ does not imply that $MF_{ART}(\mathbf{X}^{(t)}, \mathbf{W}^{(t)}_{\mathbf{j}}) > MF_{ART}(\mathbf{X}^{(t)}, \mathbf{W}^{(t)}_{\mathbf{h}}), \forall \mathbf{X}^{(t)} \in \mathcal{D}^d, \forall \mathbf{W}^{(t)}_{\mathbf{j}}, \mathbf{W}^{(t)}_{\mathbf{h}} \in \mathcal{D}^p$, and vice versa. Since this condition holds in ART 1, IART 1, and Fuzzy ART, besides AHN, then all ART 1-based networks may be implemented with version 1 of the EART implementation scheme, see Table I.

Step 0. Initialization: Presentation counter t and PE counter $c(t)$ are set to zero.

Step 1. Input Pattern Presentation: Presentation counter is increased by one as $t = t+1$, and a new pattern $\mathbf{X}^{(t)}$ is presented to input nodes.

Step 2. Detection of Processing Units Eligible for Resonance—Vigilance Testing (3): The orienting subsystem selects as candidates for resonance those processing units that match external requirements. To select these units, the orienting subsystem employs vigilance test (3). All PEs (generally, more than one) that satisfy this constraint constitute an ensemble passed to Step 3. If this ensemble is empty, goto Step 4b).

Step 3. Resonance Domain Detection—Activation Value Computation and Best-Matching Unit Selection (1): In line with (1), the largest activation among PEs that have passed disequality (3) in Step 2 is selected.

Step 4a). Resonance Condition—Reinforcement Learning: Prototype of the best-matching unit, $\mathbf{W}^{(t)}_{\mathbf{w1(t)}}$, is adjusted to input pattern $\mathbf{X}^{(t)}$ according to an ART model-dependent weight adaptation law. Other prototypes may also be considered suitable for adaptation if soft-competitive learning strategies are adopted.

Step 4b). Nonresonance Condition—New Processing Element Allocation: If there is no solution to the maximization problem described above, i.e., if the ensemble detected in Step 2 is an empty set, then "resonance" does not occur and one new processing unit is dynamically allocated to match external expectations. Thus, the PE counter is increased as $c(t+1) = c(t) + 1$ and a new output node $E^{(t+1)}_{c(t+1)}$ is allocated to match input pattern $\mathbf{X}^{(t)}$, e.g., when $p = d$, i.e., if dimensionalities of parameter and data spaces are the same, then $\mathbf{W}^{(\mathbf{t+1})}_{\mathbf{c(t+1)}} = \mathbf{X}^{(t)}$. As a consequence, ART algorithms require no randomization of initial templates since initial values are data-driven.

Step 5: Goto step 1.

2) Version 2 of the EART Implementation Scheme: If activation function (2) increases monotonically with match function (3), i.e., if $AF_{ART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}{}^{(t)}) > AF_{ART}(\mathbf{X}^{(t)}, \mathbf{W}^{(t)}_{\mathbf{h}})$ implies that $MF_{ART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}{}^{(t)}) > MF_{ART}(\mathbf{X}^{(t)}, \mathbf{W}^{(t)}_{\mathbf{h}})$, $\forall \mathbf{X}^{(t)} \in \mathcal{D}^d, \forall \mathbf{W}_{\mathbf{j}}{}^{(t)}, \mathbf{W}^{(t)}_{\mathbf{h}} \in \mathcal{D}^p$, and vice versa, then EART version 2 holds. This EART version employs the above condition to reduce computation steps required to cluster input patterns, i.e., EART version 2 is more efficient than version 1. ART 1, IART 1, AHN and Fuzzy ART do not satisfy the condition above and cannot employ EART version 2, see Table I. One obvious example in which the condition above is satisfied is when $AF_{ART}(\mathbf{X}^{(t)}, \mathbf{W}^{(t)}_{\mathbf{j}}) = MF_{ART}(\mathbf{X}^{(t)}, \mathbf{W}^{(t)}_{\mathbf{j}})$.

Step 0. Initialization: As in EART version 1.

Step 1. Input Pattern Presentation: As in EART version 1.

Step 2. Detection of Processing Units Eligible for Resonance—Activation Value Computation and Best-Matching Unit Selection (1): The largest activation is selected according to (1). The corresponding PE is the best-matching unit, which is the only processing unit passed to vigilance testing.

Step 3. Resonance Domain Detection—Vigilance Testing (3): Vigilance test (3) is applied to the best-matching unit exclusively. If the test is not satisfied, goto Step 4b).

Step 4a). Resonance Condition—Reinforcement Learning: As in EART version 1.

Step 4b). Non-Resonance Condition—New Processing Element Allocation: As in EART version 1.

Step 5: Goto step 1.

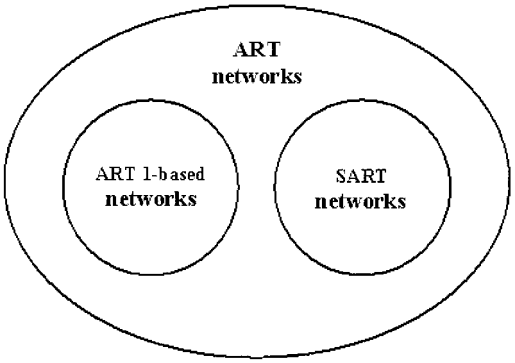

Fig. 3. Class ART, SART and ART 1-based group of networks.

III. The ART 1-Based Group of ART Clustering Networks

In this section, ART 1, IART 1, AHN, and Fuzzy ART are interpreted in the light of the general ART framework proposed in Section II. In other words, this section presents a special version of the ART clustering framework capable of modeling ART 1-based networks exclusively (see Fig. 3). This ART 1-based clustering framework optimizes ART 1-based networks in terms of memory storage (e.g., ART 1, IART 1) and/or computation time (e.g., ART 1, IART 1, Fuzzy ART) with respect to their traditional implementations [5], [6], [8], [15], [16].

In line with Section II, let us consider, at presentation time t, an input vector $\mathbf{X}^{(t)} \in \mathcal{D}^d$, where domain $\mathcal{D} = \mathcal{R}$ in the analog case (Fuzzy ART), or $\mathcal{D} = \{0, 1\}$ in the binary case (ART 1, IART 1, AHN). Cluster structures are parameterized as $\mathbf{W_j}^{(t)} \in \mathcal{D}^p$, $j = 1, \ldots, c(t)$, where $p = d$. In other words, in ART 1-based systems, parameter and data spaces have the same dimensionality, i.e., cluster parameter vectors are points in data space.

Definition 1: Let us define as unidirectional interpattern degree of match (UIDM) any (normal and not symmetric) mapping

$$UIDM(\mathbf{X}, \mathbf{W}): \mathcal{D}^d \times \mathcal{D}^d \rightarrow [0, 1] \tag{5}$$

where domain $\mathcal{D} = \mathcal{R}$ in the analog case, or $\mathcal{D} = \{0, 1\}$ in the binary case such that: 1) if $\mathbf{X} = \mathbf{W}$, then $UIDM(\mathbf{X}, \mathbf{X})$ takes on its maximum (equal to 1), but the contrary does not hold and 2) $UIDM(\mathbf{X}, \mathbf{W}) \neq UIDM(\mathbf{W}, \mathbf{X})$, i.e., the $UIDM$ function is not symmetric with respect to vectors $\mathbf{X}$ and $\mathbf{W}$.[3]

According to Section II and definition 1 above, we state that an ART 1-based clustering network is equivalent to an ART optimization problem, defined by (1)–(4), constrained as follows (see Fig. 3):

- activation function (2), $AF_{ART1\text{-}based}(\mathbf{X}^{(t)}, \mathbf{W_j}^{(t)})$, and match function (4), $MF_{ART1\text{-}based}(\mathbf{X}^{(t)}, \mathbf{W_j}^{(t)})$, both belong to class UIDM. In particular, activation function $AF_{ART1\text{-}based}(\mathbf{X}^{(t)}, \mathbf{W_j}^{(t)})$, employed in (1), measures the degree to which $\mathbf{X}^{(t)}$ matches $\mathbf{W_j}^{(t)}$, but it does not assess the reverse situation, i.e., the degree to which $\mathbf{W_j}^{(t)}$ matches $\mathbf{X}^{(t)}$. Vice versa, match function $MF_{ART1\text{-}based}(\mathbf{X}^{(t)}, \mathbf{W_j}^{(t)})$, employed in (3), measures the degree to which $\mathbf{W_j}^{(t)}$ matches $\mathbf{X}^{(t)}$, but it does not assess the degree to which $\mathbf{X}^{(t)}$ matches $\mathbf{W_j}^{(t)}$ (see Appendix I).
- When (3) is satisfied (i.e., resonance occurs), the parameter adaptation strategy is purely competitive (crisp, hard), i.e., only the best-matching prototype, detected by (1), is adapted.
- The network is implemented efficiently by version 1 of the EART implementation scheme (see Section II-B1).

[3]Term "unidirectional," i.e., not symmetric with respect to vectors $\mathbf{X}$ and $\mathbf{W}$, is introduced in line with term "bidirectional" adopted in [7, p. 609].

IV. Fuzzy ART

Let us further specialize the ART 1-based clustering framework proposed in Section III to examine Fuzzy ART as the best-known representative of the ART 1-based network group (as a matter of fact, rather than as a standalone system, Fuzzy ART is known as the basic module of the Fuzzy ARTMAP classifier). For the sake of completeness, relationships between ART 1, IART 1, and Fuzzy ART equations are also highlighted.

A. Fuzzy ART Preprocessing

Fuzzy ART requires a preprocessing stage where input pattern normalization is used to prevent category proliferation. A possible normalization technique is [8]

$$\mathbf{X}^{(t)} = \left(X_1^{(t)}, \ldots, X_d^{(t)}\right) = \frac{\mathbf{Z}^{(t)}}{\|\mathbf{Z}^{(t)}\|} \tag{6}$$

where $\mathbf{Z}^{(t)} \in \mathcal{R}^d$ is the original input pattern and

$$\left\|\mathbf{Z}^{(t)}\right\| = \sqrt{\left(Z_1^{(t)}\right)^2 + \cdots + \left(Z_d^{(t)}\right)^2} \tag{7}$$

is the Euclidean length (modulus) or norm 2, such that $\|\mathbf{X}^{(t)}\| = 1$. Otherwise, in normalization by complement coding [8],

$$\begin{aligned}\mathbf{X}^{(t)} &= \left(X_1^{(t)}, \ldots, X_d^{(t)}\right) \\ &= \left(\mathbf{Z}^{(t)}, \mathbf{Z}_{\mathbf{comp}}^{(t)}\right) \\ &= \left(Z_1^{(t)}, \ldots, Z_q^{(t)}, Z_{1,comp}^{(t)}, \ldots, Z_{q,comp}^{(t)}\right)\end{aligned} \tag{8}$$

where $\mathbf{Z}^{(t)} \in \mathcal{R}^q$ such that $d = 2q$, $Z_{k,comp}^{(t)} = 1 - Z_k^{(t)}$, $k = 1, \ldots, q$, and $|\mathbf{X}^{(t)}| = q$, where the (non-Euclidean) norm 1 operator $|\cdot|$ is defined as [8]

$$\left|\mathbf{X}^{(t)}\right| = \sum_{k=1}^{d} X_k^{(t)}. \tag{9}$$

Normalization (6), by losing vector-length information, causes an unacceptable alteration of the informative content of non-normal data sets, which are typical in real-world applications. Unlike normalization (6), (8), by adding additional, complement-coded terms to the input vector, causes no loss or gain of information, although it doubles the number of connections (storage requirement) and network computation time [5], [12]. In other words, complement coding is just a way of formatting the input data so that the Fuzzy ART activation and match functions work correctly. Essentially, complement coding allows Fuzzy ART to store and evaluate the minimum and maximum values of inputs assigned to each cluster in each dimension, i.e., complement coding allows a geometric interpretation

of Fuzzy ART recognition categories as hyperbox-shaped regions of input space [8]. In this case, each hyperbox starts as an isolated point, then it can increase its size with time, up to a maximum size, which is determined by the vigilance threshold. This also implies that templates cannot "cycle," i.e., Fuzzy ART does have a type of stability. Without complement coding, Fuzzy ART would only store the minimum values, thus entailing a loss of half of its information. As a consequence, when Fuzzy ART or Fuzzy ARTMAP are employed in practical applications, normalization by complement coding (8) is adopted exclusively.

B. Fuzzy ART Equations

In the light of the EART version 1 implementation scheme described in Section II-B1, Fuzzy ART equations are presented. Whenever necessary, relationships with ART 1 and IART 1 are pointed out. In the general analog case, $\mathbf{X}^{(t)}$ belongs to $\mathcal{R}^d$.

Activation Function (2): In Fuzzy ART, it is defined as

$$AF_{FuzzyART}\left(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)}\right) = \frac{\sum_{k=1}^{d} \min\left\{X_k^{(t)}, W_{k,j}^{(t)}\right\}}{\alpha + \sum_{k=1}^{d} W_{k,j}^{(t)}} \in [0,1], \qquad X_k^{(t)}, W_{k,j}^{(t)} \in \mathcal{R} \quad j = 1, \ldots, c(t) \tag{10}$$

where parameter $\alpha > 0$ (e.g., $\alpha \in [0.001, 1)$, [12]), is included to break ties, i.e., to bias the function in favor of the longer of two template vectors. If $\alpha = 0$, (10) belongs to class $UIDM$ (see Section III). In this case, (10) provides the degree to which input vector $\mathbf{X}^{(t)}$ matches cluster prototype $\mathbf{W}_{\mathbf{j}}^{(t)}$, but it does not assess the reverse situation. In fact $AF_{FuzzyART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)}) \neq AF_{FuzzyART}(\mathbf{W}_{\mathbf{j}}^{(t)}, \mathbf{X}^{(t)})$, i.e., (10) is not symmetric with respect to vectors $\mathbf{X}^{(t)}$ and $\mathbf{W}_{\mathbf{j}}^{(t)}$. Note that parameters ρ in (3) and α in (10) are interrelated as illustrated in [28]. For example, if $\alpha \leq \rho/(1-\rho)$, then Fuzzy ART completes its learning in one list presentation when complement coding is employed for preprocessing.[4]

Match Function (4): In Fuzzy ART, (4) belongs to class $UIDM$ (see Section III) and is defined as

$$MF_{FuzzyART}\left(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{w1(t)}}^{(t)}\right) = \frac{\sum_{k=1}^{d} \min\left\{X_k^{(t)}, W_{k,w1(t)}^{(t)}\right\}}{\sum_{k=1}^{d} X_k^{(t)}} \in [0,1], \quad X_k^{(t)}, W_{k,w1(t)}^{(t)} \in \mathcal{R}. \tag{12}$$

In line with Section III, note that (12) computes the degree to which cluster prototype $\mathbf{W}_{\mathbf{w1(t)}}{}^{(t)}$ matches input vector $\mathbf{X}^{(t)}$, but it does not assess the reverse situation. In fact $MF_{FuzzyART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{w1(t)}}^{(t)}) \neq MF_{FuzzyART}(\mathbf{W}_{\mathbf{w1(t)}}^{(t)}, \mathbf{X}^{(t)})$, i.e., (12) is not symmetric with respect to vectors $\mathbf{X}^{(t)}$ and $\mathbf{W}_{\mathbf{w1(t)}}^{(t)}$.

In binary ART 1 and IART 1, match function (4) belongs to class $UIDM$ (see Section III) and applies to binary vector pairs. It is defined as [see also Appendix I, (A1.1)] [6]

$$MF_{ART1}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{w1(t)}}^{(t)}) = \frac{\sum_{k=1}^{d} X_k^{(t)} \cdot W_{k,w1(t)}^{(t)}}{\sum_{k=1}^{d} X_k^{(t)}} \in [0,1], \quad X_k^{(t)}, W_{k,w1(t)}^{(t)} \in \{0,1\}. \tag{13}$$

Note that (12) generalizes (13) by substituting the product and norm 1 operators with fuzzy-like operators (intersection and cardinality respectively [29]). Equation (13) has a geometrical meaning: it computes a normal measure of how many unit-valued (informative) components of binary vector $\mathbf{X}^{(t)}$ are matched by those of binary template $\mathbf{W}_{\mathbf{w1(t)}}^{(t)}$, i.e., it measures the degree to which $\mathbf{W}_{\mathbf{w1(t)}}^{(t)}$ matches $\mathbf{X}^{(t)}$. Since for a binary vector the Euclidean norm (vector length) is such that $\|\mathbf{X}^{(t)}\|^2 = |\mathbf{X}^{(t)}| = \sum_{k=1}^{d} X_k^{(t)}$, where $X_k^{(t)} \in \{0,1\}$, then (13) can be written as

$$MF_{ART1}\left(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{w1(t)}}^{(t)}\right) = \frac{\mathbf{X}^{(t)} \circ \mathbf{W}_{\mathbf{w1(t)}}^{(t)}}{\|\mathbf{X}^{(t)}\|^2} = \frac{\left\|\mathbf{X}^{(t)}\right\| \cdot \left\|\mathbf{W}_{\mathbf{w1(t)}}^{(t)}\right\| \cos\theta_{w1(t)}}{\|\mathbf{X}^{(t)}\|^2} = \frac{\left\|\mathbf{W}_{\mathbf{w1(t)}}^{(t)}\right\| \cdot \cos\theta_{w1(t)}}{\|\mathbf{X}^{(t)}\|} \in [0,1] \quad X_k^{(t)}, W_{k,w1(t)}^{(t)} \in \{0,1\} \tag{14}$$

where operator $\circ$ is the dot (scalar) product and $\theta_{w1(t)}$ is the angle between $\mathbf{X}^{(t)}$ and $\mathbf{W}_{\mathbf{w1(t)}}^{(t)}$. The geometrical interpretation of (14) is the projection of $\mathbf{W}_{\mathbf{w1(t)}}^{(t)}$ along the direction of $\mathbf{X}^{(t)}$ normalized by the length of $\mathbf{X}^{(t)}$. Besides (14), IART 1 employs another match function defined as [6]

$$MF_{IART1}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{w1(t)}}^{(t)}) = \frac{\sum_{k=1}^{d} X_k^{(t)} \cdot W_{k,w1(t)}^{(t)}}{\sum_{k=1}^{d} W_{k,w1(t)}^{(t)}} = \frac{\left\|\mathbf{X}^{(t)}\right\| \cdot \cos\theta_{w1(t)}}{\|\mathbf{W}_{\mathbf{w1(t)}}^{(t)}\|} \in [0,1], \quad X_k^{(t)}, W_{k,w1(t)}^{(t)} \in \{0,1\}. \tag{15}$$

[4] In binary ART 1 and IART 1, (2) applies to binary vector pairs and belongs to class $UIDM$ when $\alpha = 0$. According to (A1.6) in Appendix I, (2) is defined as

$$AF_{ART1}\left(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)}\right) = \frac{\sum_{k=1}^{d} W_{k,j}^{(t)} \cdot X_k^{(t)}}{\alpha + \sum_{k=1}^{d} W_{k,j}^{(t)}} \in [0,1], \qquad X_k^{(t)}, W_{k,j}^{(t)} \in \{0,1\} \quad \alpha > 0,\ j = 1, \ldots, c(t). \tag{11}$$

Equation (10) generalizes (11) by substituting operators product and norm 1 [see (9)] with operations that resemble those employed in fuzzy set theory (e.g., intersection and cardinality [29]). As Simpson observed [30, p. 37]: "for these operations to be correctly interpreted as fuzzy operations, they would have to be applied to membership values, not to the parameters of the activation function." This means that the "degree of fuzzification" of Fuzzy ART with respect to ART 1 is questionable.

In line with (13), (15) provides a normal measure of how many unit-valued components of $\mathbf{W}_{\mathbf{w1(t)}}^{(t)}$ are matched by those of $\mathbf{X}^{(t)}$, i.e., it measures the degree to which $\mathbf{X}^{(t)}$ matches $\mathbf{W}_{\mathbf{w1(t)}}^{(t)}$. In line with (14), the geometrical interpretation of (15) is the projection of $\mathbf{X}^{(t)}$ along the direction of $\mathbf{W}_{\mathbf{w1(t)}}^{(t)}$ normalized by the length of $\mathbf{W}_{\mathbf{w1(t)}}^{(t)}$.

The only difference between binary ART 1 and IART 1 networks is that the latter model sequentially applies vigilance test (3) twice, where the match function is implemented as either (13) or (15). In other words, IART 1 applies a pair of "unidirectional" vigilance tests, which is equivalent to stating that IART 1 adopts one "bidirectional" vigilance test (in line with terms adopted in [7]). It was proved that this functional difference is sufficient to make IART 1 more robust than ART 1 to changes in the order of presentation of the input sequence [6].

Resonance Condition: The hard-competitive weight adaptation law to be employed in Step 4a) of the EART version 1 implementation scheme (see Section II-B1) is

$$W_{k,\,w1(t)}^{(t+1)} = (1-\beta)\cdot W_{k,w1(t)}^{(t)} + \beta\cdot\min\left\{X_k^{(t)},\, W_{k,w1(t)}^{(t)}\right\} \qquad k=1,\ldots,d \tag{16}$$

with learning rate $\beta \in [0,\,1]$. In the fast-learning case, β is equal to one. Equation (16) stresses the fact that only the winner template $\mathbf{W}_{\mathbf{w1(t)}}^{(t)}$ is allowed to be attracted by input pattern $\mathbf{X}^{(t)}$, which makes the Fuzzy ART model hard-competitive. When normalization by complement coding (8) is adopted, (16) with $\beta = 1$ is such that each category is represented by perhaps the simplest statistics about its data: the minimum and maximum values in each dimension, this representation being best suited to data that are uniformly distributed within hyperrectangles [13].

C. Potential Weaknesses of Fuzzy ART

In line with the existing ART literature (e.g., see [13] and [30]), potential weaknesses and possible developments of Fuzzy ART are analyzed in this section. In [13], an analogous discussion led to the development of unsupervised learning GART as the basic module of the supervised learning Gaussian ARTMAP (GAM) network, which was shown to be more efficient and less sensitive to the order of training samples than Fuzzy ARTMAP [13], [14].

1) Sensitivity to Noise and Outliers: Fuzzy ART may be affected by overfitting, i.e., Fuzzy ART may fit the noise and not just the data [13]. The problem of category proliferation in noisy data is partly due to the fact that activation and match functions, i.e., (10) and (12) respectively, are flat within a category's hyperrectangle [13]. Thus, activation and match functions can be substituted with functions that, for example, monotonically increase toward the center of a category's region of support [13].

In ART 1-based systems, an additional cause of category proliferation is that template generation is example-driven [23], i.e., a single poorly mapped pattern (outlier) suffices to initiate the creation of a new unit. This outlier detection capability can be combined with a noise category removal mechanism, which is straightforward to add to ART 1-based architectures if necessary. Relying upon no *a priori* knowledge about the data, removal of noise categories may be based on enough accumulated evidence, i.e., it may employ a mini-batch learning framework to collect "robust" statistics averaged over the noise on subsets of the input sequence [4]. When these statistics show that one cluster is chosen infrequently, e.g., when the sum of the degrees of membership of data points with respect to a category structure (cardinality of a cluster) is below a user-defined threshold [31], then that cluster is considered a dead unit and is pruned. In the literature, pruning techniques have been applied to and recommended for ART constructive networks [13], [27], as well as other scalable unsupervised learning algorithms (e.g., refer to [31], [32]).

2) Inefficiency of Category Structures: If the dimensionality of the data space increases, a fixed quantity of data rapidly becomes sparse, providing a very poor representation of the input-output mapping to be estimated: this is the so-called "curse of dimensionality" [4]. In a high-dimensional space, e.g., when dimensionality is $\geq$10, most of the volume of a cube is concentrated in the large number of corners in which evidence tends to be sparse and predictions become unreliable [4], [19]. This implies that when the hypothesis that data are uniformly distributed within hyperboxes does not hold, Fuzzy ART may predict (infer) the existence of data in corners of rectangular regions of support where no evidence exists [13]. To reduce this problem, a natural choice is to model cluster structures as spheres, e.g., radial Gaussian functions, which is consistent with recommendations proposed in Section IV-C1. This has led to the development of GART, which employs spherical clusters [13]. Also in the case of spherical clusters, however, cluster parameter estimation may still rely upon sparse data. In fact, in a high dimensional space, most of the probability mass of a sphere is concentrated in a thin shell close to the surface while at the sphere's center, which must be estimated from the data, probability density is high, but there is only a small fraction of the data [4].

3) Dependence of Category Structures Upon Data Set Input Presentation: The goal of on-line learning methods is to avoid storage of a complete data set by discarding each data point once it has been used [4]. On-line learning methods are required when: 1) it is necessary to respond in real time and 2) the input data set is so huge that batch methods become impractical because of their numerical properties (e.g., in linear model regression, exact batch solutions may be affected by numerical problems with large data sets [4], [33]), or computation time, or memory requirement. On-line learning typically results in systems that become order-dependent during training, in line with complex biological systems [1].

It is well known that the number and position of clusters detected by ART 1-based clustering algorithms are very sensitive to the order of presentation of the input sequence [6], [30]. In [6], it was proved that binary IART 1 improves its robustness over ART 1 by sequentially applying match functions (13) and (15) within vigilance test (3). As IART 1 improves the ART 1 clustering accuracy and robustness to changes in the order of data set input presentation by replacing the "unidirectional" (asymmetrical) match function with a "bidirectional" (symmetrical) match function, we may expect that Fuzzy ART, too, may benefit from the replacement of (12) with a "bidirectional" match function. Extending this concept, we may expect further im-

provements to come by replacing the Fuzzy ART "unidirectional" activation function, (10), with a "bidirectional" activation function.

This simple extrapolation is supported by the following reasoning. ART 1-based clustering networks employ an inherently nonsymmetrical architecture (based on asymmetrical activation and match functions) to compute an intrinsically symmetrical degree of match between an input pattern, which belongs to input space, and a template vector, which belongs to weight space, but is equivalent to a point in data space. In other words, ART 1-based networks aims at computing a similarity measure, ranging in [0, 1], between two homogeneous objects in data space. Since it is computed between two homogeneous arguments, this compatibility (similarity) measure is intrinsically symmetrical. Despite this consideration, the ART 1-based vector pair similarity measure is split into two steps, where a pair of nonsymmetrical activation and match functions compute two "unidirectional" degrees of match. This match value pair is not treated as an information unit. Rather, the first degree of match (activation value) is employed to select the best-matching cluster while the second degree of match (match value) is exploited to check whether the input pattern falls within a bounded hypervolume of acceptance around the best-matching cluster. If the order of the match value pair is switched (i.e., if the input and template vectors in data space are switched), system behaviors may change. This sensitivity to the order of the match value pair reveals that ART 1-based networks feature an accidental dependence on the order of presentation of the input sequence not to be confused with the systematic dependence of on-line learning systems upon the order of data set input presentation.

V. S-Fuzzy ART

Let us introduce a modified version of Fuzzy ART, called symmetric Fuzzy ART (S-Fuzzy ART), whose activation and match functions are Fuzzy ART-based, but symmetric. According to Section IV-C3, S-Fuzzy ART should be more accurate and more stable with respect to changes in the order of data set input presentation than Fuzzy ART. If experimental results confirm these theoretical expectations, a new group of ART networks, called simplified ART (SART), may be developed, such that: 1) it is a generalization of S-Fuzzy ART and 2) it belongs to the general ART framework proposed in Section II (see Fig. 3).

Definition 2: Let us define as interpattern degree of match (IDM) any (normal and symmetric) mapping

$$IDM(\mathbf{X}, \mathbf{W}): \mathcal{D}^d \times \mathcal{D}^d \rightarrow [0, 1] \tag{17}$$

where domain $\mathcal{D} = \mathcal{R}$ in the analog case, or $\mathcal{D} = \{0, 1\}$ in the binary case, such that: 1) if $\mathbf{X} = \mathbf{W}$, then $IDM(\mathbf{X}, \mathbf{X})$ takes on its maximum, and vice versa and 2) $IDM(\mathbf{X}, \mathbf{W}) = IDM(\mathbf{W}, \mathbf{X})$, $\forall\, \mathbf{X}, \mathbf{W} \in \mathcal{D}^d$. One instance of the class of IDM functions, taken from the literature and capable of processing both analog and binary cases, is [24]

$$IDM_1(\mathbf{X}, \mathbf{W}) = \frac{1}{1 + ||\mathbf{X} - \mathbf{W}||^2} \in (0, 1] \qquad \forall \mathbf{X}, \mathbf{W} \in \mathcal{D}^d. \tag{18}$$

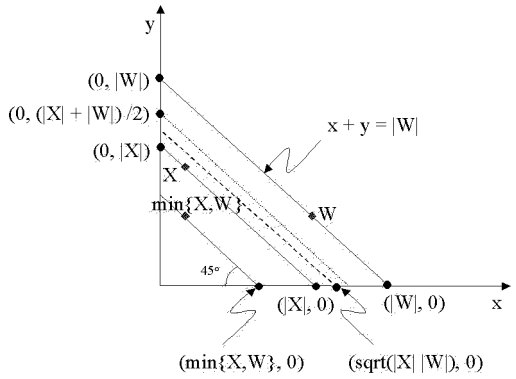

Fig. 4. Geometric interpretation of Fuzzy ART and S-Fuzzy ART activation and match functions.

One more instance of class IDM in both analog and binary cases is the combination of the two $UIDM$ equations (10) and (12). For example, when parameter α in (10) is omitted, the product between (10) and (12) gives

$$IDM_2(\mathbf{X}, \mathbf{W}) = \frac{\left(\sum_{k=1}^{d} \min\{X_k, W_k\}\right)^2}{\sum_{k=1}^{d} X_k \cdot \sum_{k=1}^{d} W_k} \propto \frac{\sum_{k=1}^{d} \min\{X_k, W_k\}}{\sqrt{\sum_{k=1}^{d} X_k \cdot \sum_{k=1}^{d} W_k}} \in [0, 1] \qquad \forall \mathbf{X}, \mathbf{W} \in \mathcal{D}^d. \tag{19}$$

In mathematical terms, the right side of (19) computes the ratio between the norm 1 [see (9)] of point $(\min\{X_k, W_k\},\ k = 1, \ldots, d)$, and the geometric mean[5] of norm 1 of points $\mathbf{X}$ and $\mathbf{W}$, see Fig. 4.[6]

We call S-Fuzzy ART a Fuzzy ART adaptation where activation and match functions are the same function, equivalent to the combination (e.g., sum or product) of (10) with (12). For example, when parameter α in (10) is omitted, we may choose

$$AF_{S\text{-}FuzzyART}\left(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)}\right) = MF_{S\text{-}FuzzyART}\left(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)}\right) = (19)$$

such that

$$AF_{S\text{-}FuzzyART}\left(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)}\right) = AF_{S\text{-}FuzzyART}\left(\mathbf{W}_{\mathbf{j}}^{(t)}, \mathbf{X}^{(t)}\right)$$

[5]Given two variables a and b in $\mathcal{R}_0^+$ the geometric and arithmetic means, defined as $\sqrt{ab}$ and $(a + b)/2$ respectively, satisfy disequality $\sqrt{ab} \leq (a + b)/2$.

[6]The binary version of (19), equivalent to the product between (11) and (13) when $\alpha = 0$, is provided with a simple geometric meaning. When parameter α in (11) is omitted, the product between (11) and (13) gives

$$IDM_3(\mathbf{X}, \mathbf{W}) = \frac{\left(\sum_{k=1}^{d} X_k \cdot W_k\right)^2}{\sum_{k=1}^{d} X_k \cdot \sum_{k=1}^{d} W_k} = \cos^2(\theta) \in [0, 1], \qquad \forall X_k, W_k \in \{0, 1\} \tag{20}$$

where θ is the angle between binary vectors $\mathbf{X}$ and $\mathbf{W}$. Equation (20) states that two binary vectors are the same vector iff their in-between angle θ is zero, regardless of their moduli (of course, this interpretation does not apply to nonbinary vector pairs).

i.e., both activation and match functions are symmetric with respect to vectors $\mathbf{X}^{(t)}$ and $\mathbf{W}_{\mathbf{j}}^{(t)}$. Exploitation of (19) in (1) and (3) allows S-Fuzzy ART to be implemented in line with version 2 of the EART implementation scheme (see Section II-B2 and Table I), which is more efficient than EART version 1 applied to Fuzzy ART (see Section III).

VI. Experimental Comparison Between Fuzzy ART and S-Fuzzy ART

In Appendixes II and III, a simple numerical example provides insights into how S-Fuzzy ART aims at improving Fuzzy ART accuracy and robustness with respect to changes in the order of presentation of the input data. An experimental comparison between Fuzzy ART and S-Fuzzy ART is provided below. These experiments show that S-Fuzzy ART improves Fuzzy ART in terms of accuracy and robustness. However, neither S-Fuzzy ART nor Fuzzy ART are: 1) competitive with several clustering algorithms found in the literature when the Iris data set is processed and 2) consistent with human perceptual grouping capabilities when the Simpson data set is processed. Although encouraging, these results reveal that, to improve its performances significantly, Fuzzy ART should be revised more in depth, e.g., according to the entire set of recommendations proposed in Section IV-C. To improve Fuzzy ART, first, we define a new group of ART networks, called SART (see Section VII), as a generalization of S-Fuzzy ART, and, second, we develop new instances of class SART that try to satisfy all constraints proposed in Section IV-C (e.g., see Appendix IV and Part II of this work).

Results of the comparison between Fuzzy ART and S-Fuzzy ART may easily extend to the ARTMAP supervised learning framework in general and, in particular, to the Fuzzy ARTMAP classifier (which employs two Fuzzy ART units as processing modules).

Iris Data Set: To provide a first assessment of Fuzzy ART and S-Fuzzy ART accuracy and robustness, let us consider 30 presentations of the standard four-dimensional Iris data set, consisting of 50 vectors for each of three classes [34]. In the first step of this comparison, prototypes are computed from the Iris data set without using vector labels. In the second step, a many-to-one class prediction function, i.e., a *multiple-prototype classifier* [35], is obtained by relating each cluster to the class having the largest number of representatives inside the cluster (majority vote, [4]).

In this experiment, for every input data presentation, vigilance threshold ρ is increased until the number of detected clusters is equal to the desired number of clusters c. Let us identify with number of ρs (no. ρs) the size of the set of discrete ρ values capable of detecting the desired number of clusters c in every input data presentation. Thus, for a given number of desired clusters c, both algorithms are run (30 $\times$ no. ρs) times, each run employing a different combination of an Iris input sequence with a vigilance threshold (in the same input sequence, if several ρs detect the same number of desired clusters c, then the best performance in terms of the classification error is selected).

Exploitation of the Iris data set allows comparison of Fuzzy ART and S-Fuzzy ART accuracy with those of other clustering models found in the literature. Typical error rates for unsupervised categorization of the Iris data set are 10–16 mistakes [34]–[36]. For example, when the number of clusters is three, then: 1) the Fuzzy Min–Max clustering model misclassifies 18 patterns (see [30, Fig. 10]); 2) the Fuzzy c-means algorithm is affected by 15 misclassifications [37]; 3) the Kohonen VQ algorithm is affected by 17 misclassifications [37]; 4) the class of on-line fuzzy algorithms for learning vector quantization is affected by 16 misclassifications [38]; and 5) the on-line GLVQ family of algorithms is affected by 16 misclassifications [39].

Results obtained with Fuzzy ART and S-Fuzzy ART are shown in Tables II and III, where the following symbols are used:

- c is the network size of interest, i.e., the desired number of clusters;
- $\overline{c}$ is the average number of detected clusters when input parameter ρ is set equal to ρ_m and ρ_M, respectively (see below);
- $\sigma(c)$, which is the standard deviation over detected c values, increases when system robustness with respect to changes in the presentation sequence decreases;
- no. ρs is the size of the set of discrete ρ values capable of detecting the desired number of clusters c in every input data presentation; the no. ρs value increases when the system robustness with respect to changes in the presentation sequence decreases;
- $\overline{\rho}$ is the average value of the user-defined input parameter ρ;
- $\sigma(\rho)$, which is the standard deviation over ρ values, increases when the system robustness with respect to changes in the presentation sequence decreases;
- ρ_m is the minimum ρ value;
- ρ_M is the maximum ρ value;
- $\overline{E}$ is the average classification error (i.e., the average number of misclassified patterns);
- $\sigma(E)$, which is the standard deviation of error E, increases when the system robustness with respect to changes in the presentation sequence decreases;
- E_m is the minimum value of E;
- E_M is the maximum value of E.

When category structures are detected in the Iris data set, S-Fuzzy ART improves accuracy and robustness of Fuzzy ART, e.g., see $\overline{E}$, $\sigma(c)$, no. ρs, $\sigma(\rho)$ and $\sigma(E)$ values in Tables II and III. Finally, when $c = 3$, observe that neither S-Fuzzy ART nor Fuzzy ART are competitive with several clustering algorithms found in the literature (see the list of typical error rates reported earlier in this section).

Simpson Data Set: Despite its simplicity, the unsupervised Simpson data set [30], consisting of 24 patterns, is sufficient to highlight some functional differences between Fuzzy ART and S-Fuzzy ART. The Simpson data set is shown in Fig. 5, where we provide data points with five different labels reflecting our global impression of Fig. 5 (see perceptual grouping problems in vision, which deal with the detection of the "right" partition of an image into subsets [40]). At lower spatial resolution, a

TABLE II
FUZZY ART. EPOCHS = 10. THIRTY PRESENTATIONS OF THE IRIS DATA SET

c	$\overline{c}$	$\sigma(c)$	**no. ρs**	$\overline{\rho}$	$\sigma(\rho)$	ρ_m	ρ_M	$\overline{E}$	$\sigma(E)$	E_m	E_M
3	2.81	0.49	2	0.594	0.0198	0.530	0.600	46.417	3.878	41	51
5	5.29	1.09	5	0.716	0.0195	0.700	0.750	11.826	4.579	12	36
8	7.79	1.05	4	0.794	0.0067	0.770	0.800	12.071	4.891	14	44
12	11.87	0.70	3	0.828	0.0030	0.823	0.830	6.773	2.045	4	11

TABLE III
S-FUZZY ART. EPOCHS = 10. THIRTY PRESENTATIONS OF THE IRIS DATA SET

c	$\overline{c}$	$\sigma(c)$	**no. ρs**	$\overline{\rho}$	$\sigma(\rho)$	ρ_m	ρ_M	$\overline{E}$	$\sigma(E)$	E_m	E_M
3	3.00	0.00	1	0.600	0.0000	0.600	0.600	21.458	6.413	15	32
5	5.17	0.38	3	0.736	0.0072	0.730	0.750	12.708	3.556	8	18
8	8.25	0.60	3	0.784	0.0072	0.770	0.790	7.869	3.307	3	13
12	12.31	0.46	2	0.826	0.0017	0.825	0.830	6.500	2.554	3	11

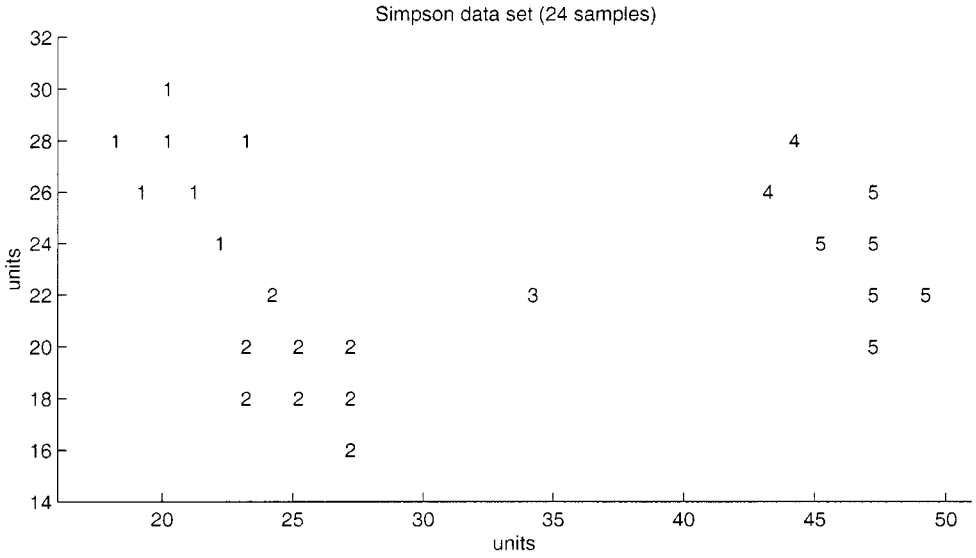

Fig. 5. Simpson data set, consisting of 24 patterns. We provide data points with five different labels reflecting our global impression of the image (this is a perceptual grouping problem in vision, dealing with the detection of the "right" partition of an image into subsets).

three-cluster partition may be perceived, where label 1 is joined with label 2, label 4 with label 5 while label 3 stays isolated.

Fuzzy ART and S-Fuzzy ART are input with six different presentations of the Simpson data set, while vigilance parameter ρ is adapted until the two algorithms detect either three or five clusters in every input sequence. Corresponding confusion matrices averaged over six runs are shown in Tables IV–VII.

Tables IV and V show that when the number of detected clusters is three, robustness of S-Fuzzy ART is superior to that of Fuzzy ART (see values of standard deviation per cell). Separation of labels one and two also appears to be more consistent in S-Fuzzy ART, while neither one of the two algorithms is capable of detecting label 3 as an isolated cluster.

Tables VI and VII show that when the number of detected clusters is five, both algorithms are insensitive to the order of presentation of the input sequence. When the unsupervised first stage is combined with a supervised second stage employing a majority vote mechanism, then S-Fuzzy ART is superior to Fuzzy ART in terms of misclassification points (zero versus two, respectively).

In line with conclusions drawn from the Iris data clustering, this experiment shows that S-Fuzzy ART seems superior to

TABLE IV
FUZZY ART. $\rho = 0.75$. EPOCHS = 10. SIX PRESENTATIONS OF THE SIMPSON DATA SET. NO. OF CLUSTERS = 3. AVERAGE CONFUSION MATRIX REPORTING POINT ALLOCATIONS AND, IN PARENTHESES, STANDARD DEVIATION PER CELL

	Cluster 1	Cluster 2	Cluster 3
Sup. Label 1	7 (0)	0 (0)	0 (0)
Sup. Label 2	2.5 (2.73)	5.5 (2.73)	0 (0)
Sup. Label 3	0 (0)	1 (0)	0 (0)
Sup. Label 4	0 (0)	0 (0)	2 (0)
Sup. Label 5	0 (0)	0 (0)	6 (0)

TABLE V
S-FUZZY ART. $\rho = 0.685$. EPOCHS = 10. SIX PRESENTATIONS OF THE SIMPSON DATA SET. NO. OF CLUSTERS = 3. AVERAGE CONFUSION MATRIX REPORTING POINT ALLOCATIONS AND, IN PARENTHESES, STANDARD DEVIATION PER CELL

	Cluster 1	Cluster 2	Cluster 3
Sup. Label 1	7 (0)	0 (0)	0 (0)
Sup. Label 2	0.5 (0.54)	7.5 (0.54)	0 (0)
Sup. Label 3	0 (0)	1 (0)	0 (0)
Sup. Label 4	0 (0)	0 (0)	2 (0)
Sup. Label 5	0 (0)	0 (0)	6 (0)

Fuzzy ART in terms of accuracy and robustness, although both algorithms seem incapable of solving even simple clustering problems consistently with human perceptual grouping (e.g., when the Simpson data set is partitioned with three clusters).

VII. THE SART GROUP OF ART CLUSTERING NETWORKS

Owing to its Fuzzy ART-based symmetric intern-pattern similarity measure, S-Fuzzy ART is superior to Fuzzy ART, in terms of clustering accuracy and robustness, when the Iris and Simpson data sets are processed (see Section VI). Our strategy is to generalize S-Fuzzy ART to generate, within the general ART framework proposed in Section II a new class of algorithms, called SART, whose aim is to perform better than the ART 1-based group of networks defined in Section III. To avoid ART 1-based potential weaknesses discussed in Section IV-C, class SART does not overlap the ART 1-based group of networks, see Fig. 3. In synthesis:

1) SART networks must be capable of processing analog as well as binary input patterns.
2) The SART optimization problem is a specialization of the general ART framework, proposed in Section II, where match and activation functions satisfy a set of constraints different from those required by ART 1-based networks in Section III.

A. Absolute and Relative Fuzzy Memberships

In the terminology adopted in fuzzy set theory [24]–[26]: 1) the "possibilistic" (absolute) membership value of a point with respect to a cluster (equivalent to a vague concept or fuzzy set) does not depend on its membership values in other clusters and 2) the "probabilistic" (relative) membership value of a point with respect to a cluster is a relative number, and it depends on the absolute membership of the point in all clusters, thus indirectly on the total number of clusters itself.

Let us recall that, in the ART as well as SART processing frameworks, any analog input vector $\mathbf{X}^{(t)}$ belongs to analog data space $\mathcal{R}^d$, where d is the dimensionality of the input space, while any cluster structure $\mathbf{W}_\mathbf{j}^{(t)}$, $j = 1, \ldots, c(t)$, belongs to parameter space $\mathcal{R}^p$, with $p \geq d$.

Definition 3: We define as absolute (or possibilistic) membership (AM) of pattern $\mathbf{X} \in \mathcal{R}^d$ with respect to (the vague concept of) cluster structure $\mathbf{W} \in \mathcal{R}^p$, $p \geq d$, a mapping

$$AM(\mathbf{X}, \mathbf{W}): \mathcal{R}^d \times \mathcal{R}^p \rightarrow \mathcal{R}_0^+ \tag{21}$$

equivalent to a "compatibility" (i.e., typicality, membership) measure between data point $\mathbf{X}$ and cluster model $\mathbf{W}$. In the case of $p = d$, i.e., when vectors of cluster parameters are points in data space, 1) if $\mathbf{X} = \mathbf{W}$, then $AM(\mathbf{X}, \mathbf{X})$ takes on its maximum, and vice versa and 2) $AM(\mathbf{X}, \mathbf{W}) = AM(\mathbf{W}, \mathbf{X})$, $\forall \mathbf{X}, \mathbf{W} \in \mathcal{R}^d$.

Definition 4: If the least upper bound of the range of values of an AM function is unity, then we call this mapping normal absolute (or possibilistic) membership (NAM), i.e.,

$$NAM(\mathbf{X}, \mathbf{W}): \mathcal{R}^d \times \mathcal{R}^p \rightarrow [0, 1] \tag{22}$$

equivalent to a "compatibility" (i.e., typicality, membership) measure between data point $\mathbf{X}$ and cluster model $\mathbf{W}$. In the case of $p = d$, i.e., when vectors of cluster parameters are points in data space, 1) if $\mathbf{X} = \mathbf{W}$, then $NAM(\mathbf{X}, \mathbf{X})$ takes on its maximum, and vice versa and 2) $NAM(\mathbf{X}, \mathbf{W}) = NAM(\mathbf{W}, \mathbf{X})$, $\forall \mathbf{X}, \mathbf{W} \in \mathcal{R}^d$. This implies that in the case of $p = d$, a NAM function is equivalent to an IDM function, see (17).

One instance of the class of NAM functions, where $p > d$, is the unit-height Gaussian distribution [13]

$$\begin{aligned} NAM_1(\mathbf{X}, \mathbf{W}) &= NAM_1(\mathbf{X}, (\mu, \sigma)) \\ &= \exp\left[-\frac{1}{2}\sum_{k=1}^{d}\left(\frac{d(X_k, \mu_k)}{\sigma_k}\right)^2\right] \\ &= \exp\left[-\frac{1}{2}\sum_{k=1}^{d}\left(\frac{X_k - \mu_k}{\sigma_k}\right)^2\right] \in [0, 1] \\ &\quad \forall \mathbf{X} \in \mathcal{R}^d, \forall \mathbf{W} \in \mathcal{R}^p \end{aligned} \tag{23}$$

where, in this case, distance $d(X_k, \mu_k)$ identifies the Euclidean distance.

Definition 5: Let us define as relative (or probabilistic) membership (RM) of pattern $\mathbf{X}^{(t)} \in \mathcal{R}^d$ with respect to (the vague concept of) cluster structure $\mathbf{W}_\mathbf{j}^{(t)} \in \mathcal{R}^p$, $j = 1, \ldots, c(t)$, belonging to codebook $\hat{\mathbf{W}}^{(t)} = \{\mathbf{W}_\mathbf{1}^{(t)}, \ldots, \mathbf{W}_\mathbf{c(t)}^{(t)}\}$, any normal mapping [24], [25]

$$RM_j\left(\mathbf{X}^{(t)}, \hat{\mathbf{W}}^{(t)}\right): \mathcal{R}_1^p \times \cdots \times \mathcal{R}_{c(t)}^p \times \mathcal{R}^d \rightarrow [0, 1] \tag{24}$$

TABLE VI
FUZZY ART. $\rho = 0.85$. EPOCHS = 10. SIX PRESENTATIONS OF THE SIMPSON DATA SET. NO. OF CLUSTERS = 5. AVERAGE CONFUSION MATRIX REPORTING POINT ALLOCATIONS AND, IN PARENTHESES, STANDARD DEVIATION PER CELL

	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
Sup. Label 1	7 (0)	0 (0)	0 (0)	0 (0)	0 (0)
Sup. Label 2	0 (0)	8 (0)	0 (0)	0 (0)	0 (0)
Sup. Label 3	0 (0)	0 (0)	1 (0)	0 (0)	0 (0)
Sup. Label 4	0 (0)	0 (0)	0 (0)	2 (0)	0 (0)
Sup. Label 5	0 (0)	0 (0)	0 (0)	3 (0)	3 (0)

TABLE VII
S-FUZZY ART. $\rho = 0.8$. EPOCHS = 10. SIX PRESENTATIONS OF THE SIMPSON DATA SET. NO. OF CLUSTERS = 5. AVERAGE CONFUSION MATRIX REPORTING POINT ALLOCATIONS AND, IN PARENTHESES, STANDARD DEVIATION PER CELL

	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
Sup. Label 1	7 (0)	0 (0)	0 (0)	0 (0)	0 (0)
Sup. Label 2	0 (0)	8 (0)	0 (0)	0 (0)	0 (0)
Sup. Label 3	0 (0)	0 (0)	1 (0)	0 (0)	0 (0)
Sup. Label 4	0 (0)	0 (0)	0 (0)	2 (0)	0 (0)
Sup. Label 5	0 (0)	0 (0)	0 (0)	0 (0)	6 (0)

such that [24]

$$\sum_{j=1}^{c(t)} RM_j\left(\mathbf{X}^{(t)}, \hat{\mathbf{W}}^{(t)}\right) = 1 \tag{25}$$

where function $RM_j(\mathbf{X}^{(t)}, \hat{\mathbf{W}}^{(t)})$ is monotonically nonincreasing with a generic distance $d(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)})$ and monotonically nondecreasing with distances $d(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{h}}^{(t)})$, $h = 1, \ldots, c(t)$, $h \neq j$. If $p = d$, then $RM_j(\mathbf{X}^{(t)}, \hat{\mathbf{W}}^{(t)}) = RM_j(\hat{\mathbf{W}}^{(t)}, \mathbf{X}^{(t)})$ must hold, i.e., if cluster parameter vectors are points in data space then function RM_j must be symmetric with respect to vectors $\mathbf{X}^{(t)}$ and $\mathbf{W}_{\mathbf{j}}^{(t)}$, $j = 1, \ldots, c(t)$.

Given any AM or NAM function, a possible RM_j function is

$$RM_j\left(\mathbf{X}^{(t)}, \hat{\mathbf{W}}^{(t)}\right) = \frac{AM\left(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)}\right)}{\sum_{h=1}^{c(t)} AM\left(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{h}}^{(t)}\right)}, \qquad j = 1, \ldots, c(t). \tag{26}$$

Because of condition (26), where any relative (probabilistic) membership depends on the absolute (possibilistic) membership of the point in all clusters, any jth processing element (PE), with $j = 1, \ldots, c(t)$, is context-sensitive, i.e., relative membership computation provides a tool for modeling "network-wide internode communication by subsuming that processing elements are coupled through feed-sideways (lateral) connections" [41].

In the literature, probabilistic (relative) and possibilistic (absolute) fuzzy clustering algorithms are both affected by some well-known drawbacks. On the one hand, in probabilistic fuzzy clustering, noise points and outliers, featuring low possibilistic typicalities with respect to all templates, may have significantly high probabilistic membership values which may severely affect the prototype parameter estimate [24], [25]. On the other hand, in possibilistic fuzzy clustering, learning rates computed from absolute typicalities tend to produce coincident clusters [25], [42]. This poor behavior can be explained by the fact that cluster prototypes are uncoupled in possibilistic clustering, i.e., possibilistic clustering algorithms try to minimize an objective function by operating on each cluster independently. This leads to an increase in the number of local minima.

B. SART Optimization Problem

In contrast with the ART 1-based group of networks proposed in Section III, the SART clustering framework is defined as an ART optimization problem, consisting of (1)–(4), constrained as follows (see Fig. 3):

- activation function (2), $AF_{SART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)})$, employed in (1), is either an RM_j function, see (24) and (25), or a function monotonically increasing with an RM_j function [e.g., an AM or NAM function, see (21) and (22), implicitly related to function RM_j through (26)].
- Match function (4), $MF_{SART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{w1(t)}}^{(t)})$, employed in (3), belongs to the class of NAM functions, see (22).

- When (3) is satisfied (i.e., resonance occurs), the parameter adaptation strategy may be either hard- or soft-competitive [in the latter case, the best-matching prototype, detected by (1), may not be the only prototype to be adapted].
- The network is implemented efficiently by versions 1 or 2 of the EART implementation scheme (see Section II-B).

Based on the definitions of classes of functions NAM and RM_j (see Section VII-A), the above conditions imply that: 1) in the case of $p = d$, then $AF_{SART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)})$ and $MF_{SART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)})$ are symmetric and 2) $AF_{SART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)})$ and $MF_{SART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)})$ may be the same NAM function.

C. Examples of SART Networks

One instance of the class of SART networks, where $AF_{SART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)}) = MF_{SART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)})$ with $p = d$, is the S-Fuzzy ART system proposed in Section V. In this case, it is obviously true that the match function increases monotonically with the activation function and vice versa. Thus, S-Fuzzy ART can be implemented efficiently according to version 2 of EART, see Section II-B2 and Table I.

A second instance of class SART, where $AF_{SART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)}) = MF_{SART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)})$ while $p > d$, is a system where

$$AF_{SART}\left(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)}\right) = MF_{SART}\left(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)}\right) = (23).$$

A third instance of class SART, where $AF_{SART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)}) \neq MF_{SART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)})$ and $p > d$, is the GART probability density function (pdf) estimator, briefly described by Williamson in [13] (for more details about GART, refer to Appendix IV). Since $AF_{GART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)}) =$ (A4.2) (see Appendix IV) does not monotonically increase with $MF_{GART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)}) =$ (23), GART can be implemented efficiently according to version 1 of EART, see Section II-B1 and Table I.

Note that the only difference between this second instance of class SART and GART is that the former model ignores, in its activation equation (23), prior probability terms (i.e., it considers prior terms equiprobable) that are, instead, explicitly considered in activation equation (A4.2) of GART.

VIII. Conclusion

Class ART is defined as a generalization of several well-known clustering models, e.g., ART 1, Improved ART 1, AHN, and Fuzzy ART, which are optimized in terms of memory storage and computation time. S-Fuzzy ART, whose symmetric activation and match functions are Fuzzy ART-based, is proposed as a possible alternative to Fuzzy ART. Simple numerical examples and experimental evidence reveal that S-Fuzzy ART tends to be more robust, accurate and computationally efficient than Fuzzy ART. Generalization of the S-Fuzzy ART network leads to the definition of a specific group of networks, termed SART, which belongs to class ART. Besides S-Fuzzy ART, one more instance of class SART is the GART algorithm, which is sketchily described in the literature. GART is interpreted as an on-line constructive clustering network equivalent to a maximum-likelihood probability density function estimator for Gaussian mixtures. In Part II of this paper another clustering network, called FOSART, which belongs to class SART and, unlike GART, tries to minimize a distortion (quantization) error, is discussed and compared with Fuzzy ART, S-Fuzzy ART, GART, and other well-known clustering algorithms.

Appendix I
Simplification of Bidirectional Connections in ART 1

In binary ART 1, where bottom-up and top-down vectors, $\mathbf{B}_{\mathbf{j}}^{(t)}$ and $\mathbf{W}_{\mathbf{j}}^{(t)}$, $j = 1, \ldots, c(t)$, respectively, are adopted, the following equations hold [5], [6].

- Match function (4), to be employed in Step 2 of the EART version 1 implementation scheme (see Section II-B1):

$$MF_{ART1}\left(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)}\right) = \frac{\sum_{k=1}^{d} X_k^{(t)} \cdot W_{k,j}^{(t)}}{\sum_{k=1}^{d} X_k^{(t)}} \in [0, 1], \qquad X_k^{(t)}, W_{k,j}^{(t)} \in \{0, 1\} \quad j = 1, \ldots, c(t). \tag{A1.1}$$

- Activation function (2), to be employed in Step 3 of the EART version 1 implementation scheme (see Section II-B1):

$$AF_{ART1}\left(\mathbf{X}^{(t)}, \mathbf{B}_{\mathbf{j}}^{(t)}\right) = \sum_{k=1}^{d} B_{k,j}^{(t)} \cdot X_k^{(t)} \in [0, 1], \qquad X_k^{(t)} \in \{0, 1\} \quad B_{k,j}^{(t)} \in [0, 1],\ j = 1, \ldots, c(t). \tag{A1.2}$$

- Hard-competitive top-down weight adaptation law, to be employed in Step 4a) of the EART version 1 implementation scheme (see Section II-B1):

$$W_{k,w1(t)}^{(t+1)} = W_{k,w1(t)}^{(t)} \cdot X_k^{(t)} \in \{0, 1\} \quad X_k^{(t)}, W_{k,w1(t)}^{(t)} \in \{0, 1\} \quad w1(t) \in \{1, c(t)\},\ k = 1, \ldots, d. \tag{A1.3}$$

- Hard-competitive bottom-up weight adaptation law, to be employed in Step 4a) of the EART version 1 implementation scheme (see Section II-B1):

$$B_{k,w1(t)}^{(t+1)} = \frac{W_{k,w1(t)}^{(t)} \cdot X_k^{(t)}}{\alpha + \sum_{h=1}^{d} W_{h,w1(t)}^{(t)} \cdot X_h^{(t)}} \in [0, 1] \quad X_k^{(t)}, W_{k,w1(t)}^{(t)} \in \{0, 1\},\ w1(t) \in \{1, c(t)\} \quad \alpha > 0,\ k = 1, \ldots, d. \tag{A1.4}$$

Substituting (A1.3) into (A1.4) we obtain

$$B_{k,w1(t)}^{(t+1)} = \frac{W_{k,w1(t)}^{(t+1)}}{\alpha + \sum_{h=1}^{d} W_{h,w1(t)}^{(t+1)}} \in [0, 1], \quad W_{k,w1(t)}^{(t+1)} \in \{0, 1\} \quad w1(t) \in \{1, c(t)\},\ \alpha > 0,\ k = 1, \ldots, d. \tag{A1.5}$$

Thus, substituting (A1.5) into (A1.2) we get, for every template vector $\mathbf{W}_{\mathbf{j}}^{(t)}$, $j = 1, \ldots, c(t)$,

$$AF_{ART1}\left(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)}\right) = \frac{\sum_{k=1}^{d} W_{k,j}^{(t)} \cdot X_k^{(t)}}{\alpha + \sum_{k=1}^{d} W_{k,j}^{(t)}} \in [0, 1], \qquad X_k^{(t)}, W_{k,j}^{(t)} \in \{0, 1\} \quad j = 1, \ldots, c(t). \tag{A1.6}$$

Equations (A1.1) and (A1.6) show that: 1) ART 1 activation and match functions at time t depend exclusively on unidirectional weight vectors $\mathbf{W}_{\mathbf{j}}^{(t)}$, $j = 1, \ldots, c(t)$ and 2) these weight vectors are bottom-up (feedforward) [17]. To summarize, the attentional subsystem of ART 1 is single-layer and feedforward, in line with the general ART clustering framework proposed in Section II.

Appendix II
Numerical Example of Fuzzy ART Clustering

As ART 1 was found to be sensitive to changes in the order of presentation of the input sequence [6], we expect Fuzzy ART, which is ART 1-based, to be sensitive to this type of perturbation as well. Fuzzy ART, see Section IV-B, is implemented according to the EART processing scheme version 1, see Section II-B1 and Table I.

Let us consider the following example. The input parameters are $\rho = 0.55$, $\alpha = 0.0$, $\beta = 1$ (see Section IV-B). The presentation list is $\mathbf{X^{(1)}} = (0, 0, 1, 0, 0)$, $\mathbf{X^{(2)}} = (0, 1, 1, 1, 1)$, and $\mathbf{X^{(3)}} = (1, 1, 1, 0, 0)$. For simplicity's sake, we submit this input sequence to the Fuzzy ART preprocessing normalization step (6) rather than (8) (our conclusions will not depend on the adopted normalization strategy). The presentation list becomes $\mathbf{X^{(1)}} = (0, 0, 1, 0, 0)$, $\mathbf{X^{(2)}} = (0, 0.5, 0.5, 0.5, 0.5)$, and $\mathbf{X^{(3)}} = (1/\sqrt{3}, 1/\sqrt{3}, 1/\sqrt{3}, 0, 0) = (0.57, 0.57, 0.57, 0, 0)$.

Patterns $\mathbf{X^{(1)}}$ and $\mathbf{X^{(2)}}$ generate two categories $\mathbf{W_1^{(3)}} = \mathbf{W_1^{(2)}} = \mathbf{X^{(1)}}$ and $\mathbf{W_2^{(3)}} = \mathbf{X^{(2)}}$, respectively [since vigilance test (3), which employs match function (12), is such that $MF_{FuzzyART}(\mathbf{X^{(2)}}, \mathbf{W_1^{(2)}}) = 0.5/(4 \cdot 0.5) = 0.25 < \rho$]. The winner template for pattern $\mathbf{X^{(3)}}$ is chosen according to Steps 2) and 3) in the EART processing scheme version 1, where vigilance test (3) and match function (12) are applied before computing activation function (10). Equation (12) is such that $MF_{FuzzyART}(\mathbf{X^{(3)}}, \mathbf{W_1^{(3)}}) = (1/\sqrt{3})/(3 \cdot (1/\sqrt{3})) = 0.33 < \rho$, i.e., template $\mathbf{W_1^{(3)}}$ is not eligible for resonance. Since $MF_{FuzzyART}(\mathbf{X^{(3)}}, \mathbf{W_2^{(3)}}) = (2 \cdot 0.5)/(3 \cdot (1/\sqrt{3})) = 0.57 > \rho$, template $\mathbf{W_2^{(3)}}$ satisfies the vigilance test and is the winner template. Then, fast category adaptation (12) (where $\beta = 1$) leads to $\mathbf{W_2^{(4)}} = (0, 0.5, 0.5, 0, 0)$. Thus, final templates are $\mathbf{W_1^{(4)}} = \mathbf{X^{(1)}}$ while $\mathbf{W_2^{(4)}} = (0, 0.5, 0.5, 0, 0)$.

Let us consider a different order of the input sequence where the input vectors described above are presented as follows: $\mathbf{X^{(1)}} = (0, 0, 1, 0, 0)$, $\mathbf{X^{(2)}} = (1, 1, 1, 0, 0)$, and $\mathbf{X^{(3)}} = (0, 1, 1, 1, 1)$. Due to input pattern normalization, the presentation list becomes $\mathbf{X^{(1)}} = (0, 0, 1, 0, 0)$, $\mathbf{X^{(2)}} = (1/\sqrt{3}, 1/\sqrt{3}, 1/\sqrt{3}, 0, 0) = (0.57, 0.57, 0.57, 0, 0)$, and $\mathbf{X^{(3)}} = (0, 0.5, 0.5, 0.5, 0.5)$. Patterns $\mathbf{X^{(1)}}$ and $\mathbf{X^{(2)}}$ generate two categories $\mathbf{W_1^{(3)}} = \mathbf{W_1^{(2)}} = \mathbf{X^{(1)}}$ and $\mathbf{W_2^{(3)}} = \mathbf{X^{(2)}}$ respectively [since vigilance test (3) employing match function (12) is such that $MF_{FuzzyART}(\mathbf{X^{(2)}}, \mathbf{W_1^{(2)}}) = (1/\sqrt{3})/(3/\sqrt{3}) = 0.33 < \rho$]. Thus, according to (3) and (12) in Step 2) of the EART implementation scheme version 1, $MF_{FuzzyART}(\mathbf{X^{(3)}}, \mathbf{W_2^{(3)}}) = (2 \cdot 0.5)/(4 \cdot 0.5) = 0.5 < \rho$, while $MF_{FuzzyART}(\mathbf{X^{(3)}}, \mathbf{W_1^{(3)}}) = 0.5/(4 \cdot 0.5) = 0.25 < \rho$, i.e., neither of the two templates satisfies the vigilance test. Then, a new category is dynamically allocated so that the final templates are $\mathbf{W_1^{(4)}} = \mathbf{X^{(1)}}$, $\mathbf{W_2^{(4)}} = \mathbf{X^{(2)}}$, and $\mathbf{W_3^{(4)}} = \mathbf{X^{(3)}}$.

In this example the number and position of clusters detected by Fuzzy ART is sensitive to the order of the presentation sequence.

Appendix III
Numerical Example of S-Fuzzy ART Clustering

To test the S-Fuzzy ART model proposed in Section V, where $AF_{S\text{-}FuzzyART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)}) = MF_{S\text{-}FuzzyART}(\mathbf{X}^{(t)}, \mathbf{W}_{\mathbf{j}}^{(t)}) = (19)$, implemented according to version 2 of the EART implementation scheme (see Section II-B2 and Table I), let us consider the same example employed to test Fuzzy ART in Appendix II.

The input parameters are $\rho = 0.55$, $\beta = 1$ (see Section IV-B). The normalized presentation list is $\mathbf{X^{(1)}} = (0, 0, 1, 0, 0)$, $\mathbf{X^{(2)}} = (0, 0.5, 0.5, 0.5, 0.5)$, and $\mathbf{X^{(3)}} = (1/\sqrt{3}, 1/\sqrt{3}, 1/\sqrt{3}, 0, 0) = (0.57, 0.57, 0.57, 0, 0)$.

Patterns $\mathbf{X^{(1)}}$ and $\mathbf{X^{(2)}}$ generate two categories $\mathbf{W_1^{(3)}} = \mathbf{W_1^{(2)}} = \mathbf{X^{(1)}}$ and $\mathbf{W_2^{(3)}} = \mathbf{X^{(2)}}$, respectively [since vigilance test (3), employing match function (19), is such that $MF_{S\text{-}FuzzyART}(\mathbf{X^{(2)}}, \mathbf{W_1^{(2)}}) = (0.5)^2/(4 \cdot 0.5) = 0.125 < \rho$]. The winner template for pattern $\mathbf{X^{(3)}}$ is chosen according to Steps 2) and 3) in the EART processing scheme version 2, where vigilance testing is applied after detecting the largest value of activation function (19). Equation (19) is such that $AF_{S\text{-}FuzzyART}(\mathbf{X^{(3)}}, \mathbf{W_1^{(3)}}) = (1/\sqrt{3})^2/(3 \cdot (1/\sqrt{3})) = 0.19 < AF_{S\text{-}FuzzyART}(\mathbf{X^{(3)}}, \mathbf{W_2^{(3)}}) = (2 \cdot 0.5)^2/(2 \cdot 3 \cdot (1/\sqrt{3})) = 0.28 < \rho$, i.e., template $\mathbf{W_2^{(3)}}$, which is the best-matching template, does not satisfy the vigilance test. Thus, a new category is dynamically allocated so that the final templates are $\mathbf{W_1^{(4)}} = \mathbf{X^{(1)}}$, $\mathbf{W_2^{(4)}} = \mathbf{X^{(2)}}$, and $\mathbf{W_3^{(4)}} = \mathbf{X^{(3)}}$.

In line with Appendix II, the second presentation of the normalized input sequence to be considered is $\mathbf{X^{(1)}} = (0, 0, 1, 0, 0)$, $\mathbf{X^{(2)}} = (1/\sqrt{3}, 1/\sqrt{3}, 1/\sqrt{3}, 0, 0) = (0.57, 0.57, 0.57, 0, 0)$, and $\mathbf{X^{(3)}} = (0, 0.5, 0.5, 0.5, 0.5)$. Patterns $\mathbf{X^{(1)}}$ and $\mathbf{X^{(2)}}$ generate two categories $\mathbf{W_1^{(3)}} = \mathbf{W_1^{(2)}} = \mathbf{X^{(1)}}$ and $\mathbf{W_2^{(3)}} = \mathbf{X^{(2)}}$, respectively [since vigilance test (3), employing match function (19), is such that $MF_{S\text{-}FuzzyART}(\mathbf{X^{(2)}}, \mathbf{W_1^{(2)}}) = (1/\sqrt{3})^2/(3/\sqrt{3}) = 0.19 < \rho$]. Thus, according to (1), (3) and (19) in Steps 2) and 3) of the EART processing scheme version 2, $AF_{S\text{-}FuzzyART}(\mathbf{X^{(3)}}, \mathbf{W_1^{(3)}}) = (0.5)^2/(4 \cdot 0.5) = 0.125 < AF_{S\text{-}FuzzyART}(\mathbf{X^{(3)}}, \mathbf{W_2^{(3)}}) = (2 \cdot 0.5)^2/(3 \cdot 1/\sqrt{3} \cdot 4 \cdot 0.5) = 0.28 < \rho$, i.e., template $\mathbf{W_2^{(3)}}$, which is the best-matching

template, does not satisfy the vigilance test. Thus, a new category is dynamically allocated so that the final templates are $\mathbf{W_1^{(4)}} = \mathbf{X^{(1)}}$, $\mathbf{W_2^{(4)}} = \mathbf{X^{(2)}}$, and $\mathbf{W_3^{(4)}} = \mathbf{X^{(3)}}$.

In this example the number and position of clusters detected by S-Fuzzy ART is insensitive to the order of the presentation sequence.

APPENDIX IV
GART AS AN INSTANCE OF CLASS SART

GART is an on-line constructive clustering ART network sketchily proposed in [13]. To the best of our knowledge, GART has never been employed as a standalone system. On the contrary, GART was conceived as part of the GAM classifier [13], [14]. According to its author, "the GART module plays the same role within the ARTMAP architecture as does an ART 1 module, or a Fuzzy ART module" [13].

This Appendix shows that GART: 1) belongs to the class of ML pdf estimators for Gaussian mixtures; 2) belongs to the SART class of networks (see Section VII); and 3) can be efficiently implemented with version 1 of the EART implementation scheme (see Section II-B1 and Table I).

In GART, the problem of clustering is (implicitly) defined as that of minimizing the negative log-likelihood (NLL) for data set $\mathcal{X} = \{\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(m)}\}$, where m is the size of the data set, under the hypothesis that data vectors are mutually independent and identically distributed, i.e.,

$$\begin{aligned} E_{ML} &= NLL = -\log p(\mathcal{X}) \\ &= -\log \prod_{t=1}^{m} p\left(\mathbf{X}^{(t)}\right) \\ &= -\sum_{t=1}^{m} \log p\left(\mathbf{X}^{(t)}\right) \\ &= -\sum_{t=1}^{m} \log \left[\sum_{j=1}^{c(t)} p\left(\mathbf{X}^{(t)}\middle| C_j\right) p(C_j)^{(t)}\right] \\ &= -\sum_{t=1}^{m} \log \left[\sum_{j=1}^{c(t)} Q_j\left(\mathbf{X}^{(t)}\right)\right] \end{aligned} \tag{A4.1}$$

where network size $c(t)$ increases with time, density function $p(\mathbf{X}^{(t)})$ is treated as a mixture (linear combination) of components $p(\mathbf{X}^{(t)}|C_j)$ modeled as Gaussian densities, and

$$\begin{aligned} AF_{GART}\left(\mathbf{X}^{(t)}, \mathbf{W_j^{(t)}}\right) &= Q_j\left(\mathbf{X}^{(t)}\right) \\ &= p\left(\mathbf{X}^{(t)}\middle| C_j\right) \cdot p(C_j)^{(t)} \in [0, 1] \\ & \quad j = 1, \ldots, c(t) \end{aligned} \tag{A4.2}$$

where GART recognition categories are parameterized by weight vectors $\mathbf{W_j^{(t)}} = (\boldsymbol{\mu}_j^{(t)}, \boldsymbol{\sigma}_j^{(t)})$, $j = 1, \ldots, c(t)$, where $\boldsymbol{\mu}_j^{(t)}$ is the mean and $\boldsymbol{\sigma}_j^{(t)}$ is the standard deviation. The GART activation function, (A4.2), is substituted into (1) of the ART optimization framework (see Section II-A). On identifying the $c(t)$ mixture components as C_j, $j = 1, \ldots, c(t)$, let $p(C_j)$ be the *a priori* probability that a pattern belongs to mixture component C_j, and $p(\mathbf{X}^{(t)}|C_j)$ be the class conditional probability that the pattern is $\mathbf{X}^{(t)}$, given that the pattern's state is C_j. In (A4.2), let us consider

$$p\left(\mathbf{X}^{(t)}\middle| C_j\right) = \frac{G_j\left(\mathbf{X}^{(t)}\right)}{(2\pi)^{d/2} \prod_{k=1}^{d} \sigma_{k,j}^{(t)}} \in (0, 1], \quad j = 1, \ldots, c(t) \tag{A4.3}$$

where $G_j(\mathbf{X}^{(t)})$ is the jth category's unit-height Gaussian distribution such that (see Section VII-A)

$$\begin{aligned} &MF_{GART}\left(\mathbf{X}^{(t)}, \mathbf{W_j^{(t)}}\right) \\ &\quad = G_j\left(\mathbf{X}^{(t)}\right) = (23) \in (0, 1], \quad j = 1, \ldots, c(t). \end{aligned} \tag{A4.4}$$

To detect outliers, the GART match function, (A4.4), is substituted into vigilance test (3) of the ART optimization framework (see Section II-A).

In (A4.2), owing to a hard (crisp) competitive learning strategy adopted by GART, priors are computed as

$$p(C_j)^{(t+1)} = \frac{n_j^{(t)}}{\sum_{h=1}^{c(t)} n_h^{(t)}} \in [0, 1], \quad j = 1, \ldots, c(t),\ t \in \{1, m\} \tag{A4.5}$$

where $n_j^{(t)}$ is the number of patterns assigned to the jth category, such that constraint

$$\sum_{j=1}^{c(t)} p(C_j)^{(t+1)} = 1, \qquad t \in \{1, m\} \tag{A4.6}$$

holds true. Observe that, first,. (A4.2) and (A4.4) belong to the class of NAM (see Section VII-A) functions. Second, (A4.2) increases monotonically with the posterior probability, which is a RM (see Section VII-A) defined as

$$p\left(C_j\middle|\mathbf{X}^{(t)}\right) = \frac{Q_j\left(\mathbf{X}^{(t)}\right)}{\sum_{h=1}^{c(t)} Q_h(\mathbf{X}^{(t)})} \in [0, 1], \quad j = 1, \ldots, c(t) \tag{A4.7}$$

such that

$$\sum_{j=1}^{c(t)} p\left(C_j\middle|\mathbf{X}^{(t)}\right) = 1, \qquad t \in \{1, m\}. \tag{A4.8}$$

Thus, maximization of (A4.2), which minimizes cost function (A4.1), is equivalent to maximization of (A4.7).

Overall, the properties of (A4.2), (A4.4), and (A4.7) satisfy the constraints required by the SART optimization framework (see Section VII-B).

To summarize, GART belongs to the SART class of clustering algorithms. Moreover, since $MF_{GART}(\mathbf{X}^{(t)}, \mathbf{W_j^{(t)}})$ does not monotonically increase with $AF_{GART}(\mathbf{X}^{(t)}, \mathbf{W_j^{(t)}})$, and vice versa, GART can be implemented according to version 1 of the EART implementation scheme (see Section II-B1 and Table I).

Exploitation of match function (A4.4) allows GART not to be subjected to the so-called "probabilistic membership problem" in which an outlier affects all category parameters during training [25] (on the contrary, in GART, outlier detection leads

to generation of a noise category so that pruning mechanisms should be adopted to avoid category proliferation).

Given (A4.1)–(A4.5), GART update equations are a stochastic (on-line) and hard-competitive version of the standard "batch" solution to maximize the likelihood of parameters for a Gaussian mixture model of the input data (see [4, pp. 46 and 65]).

In the first case, when, at time $t \in \mathcal{N}^+$, vigilance test (3) employing (A4.4) fails, then a new processing unit is allocated and the following update equations are applied:

$$c(t+1) = c(t) + 1, \qquad t \in \mathcal{N}^+ \tag{A4.9}$$

$$\begin{aligned} n_j^{(t+1)} &= n_j^{(t)}, \qquad j = 1, \ldots, c(t) \\ n_{c(t+1)}^{(t+1)} &= 1, \qquad t \in \mathcal{N}^+ \end{aligned} \tag{A4.10}$$

$$p(C_j)^{(t+1)} = \frac{n_j^{(t+1)}}{\sum_{h=1}^{c(t+1)} n_h^{(t+1)}}, \qquad j = 1, \ldots, c(t+1)$$

$$t \in \mathcal{N}^+ \tag{A4.11}$$

$$\begin{aligned} \boldsymbol{\mu}_j^{(t+1)} &= \boldsymbol{\mu}_j^{(t)}, \qquad j = 1, \ldots, c(t) \\ \boldsymbol{\mu}_{c(t+1)}^{(t+1)} &= \mathbf{X}^{(t)}, \qquad t \in \mathcal{N}^+ \end{aligned} \tag{A4.12}$$

$$\begin{aligned} \boldsymbol{\sigma}_j^{(t+1)} &= \boldsymbol{\sigma}_j^{(t)}, \qquad j = 1, \ldots, c(t) \\ \boldsymbol{\sigma}_{c(t+1)}^{(t+1)} &= \gamma, \qquad t \in \mathcal{N}^+ \end{aligned} \tag{A4.13}$$

where γ, equivalent to the initial standard deviation of categories, should be a "large" user-defined scale parameter (e.g., $\gamma = 0.5$ [13]), generally larger than the final σ value computed when convergence is reached, such that a newly generated category has a small *a priori* probability and a large standard deviation, and thus a weak but ubiquitous activation function (A4.2) [14]. When γ increases, then the number of detected clusters decreases. In terms of classification rate, an optimal γ exists for each data set and a given ρ value [13]. In terms of user interaction, there are two user-defined parameters, ρ and γ, capable of controlling the number of clusters detected by GART.

In the second case, having identified the best-matching category $w1(t) \in \{1, c(t)\}$ based on (1) combined with (A4.2), if vigilance test (3) employing match function (A4.4) is satisfied, then [13]

$$c(t+1) = c(t), \qquad t \in \mathcal{N}^+ \tag{A4.14}$$

$$\begin{aligned} n_j^{(t+1)} &= n_j^{(t)}, \qquad j = 1, \ldots, c(t),\ j \neq w1(t) \\ n_{w1(t)}^{(t+1)} &= n_{w1(t)}^{(t)} + 1, \qquad t \in \mathcal{N}^+ \end{aligned} \tag{A4.15}$$

$$p(C_j)^{(t+1)} = \frac{n_j^{(t+1)}}{\sum_{h=1}^{c(t)} n_h^{(t+1)}}, \qquad j = 1, \ldots, c(t),\ t \in \mathcal{N}^+ \tag{A4.16}$$

$$\begin{aligned} \boldsymbol{\mu}_j^{(t+1)} &= \boldsymbol{\mu}_j^{(t)}, \qquad j = 1, \ldots, c(t),\ j \neq w1(t) \\ \boldsymbol{\mu}_{w1(t)}^{(t+1)} &= \boldsymbol{\mu}_{w1(t)}^{(t)} + \frac{1}{n_{w1(t)}^{(t+1)}} \cdot \left(\mathbf{X}^{(t)} - \boldsymbol{\mu}_{w1(t)}^{(t)} \right), \quad t \in \mathcal{N}^+ \end{aligned} \tag{A4.17}$$

$$\begin{aligned} \boldsymbol{\sigma}_j^{(t+1)} &= \boldsymbol{\sigma}_j^{(t)}, \qquad j = 1, \ldots, c(t),\ j \neq w1(t) \\ \mathrm{var}_{k,w1(t)}^{(t+1)} &= \frac{\left\| X_k^{(t)} - \mu_{k,w1(t)}^{(t+1)} \right\|^2}{n_{w1(t)}^{(t+1)}} + \left(1 - \frac{1}{n_{w1(t)}^{(t+1)}} \right) \mathrm{var}_{k,w1(t)}^{(t)} \\ & k = 1, \ldots, d,\ t \in \mathcal{N}^+ \end{aligned} \tag{A4.18}$$

where $\boldsymbol{\sigma}_{k,\,w1(t)}^{(t+1)} = \sqrt{\mathrm{var}_{k,\,w1(t)}^{(t+1)}}$.

Within the supervised learning ARTMAP architecture, GART is adopted as part of the Gaussian ARTMAP (GAM) classifier, which is hard-competitive in its first incarnation [13], and soft-competitive in a later, more successful, implementation [14]. In several benchmarks, GAM has been seen to perform better than other supervised learning systems, such as Fuzzy ARTMAP and the EM approach to mixture modeling [13], [14].

REFERENCES

[1] R. Serra and G. Zanarini, *Complex Systems and Cognitive Processes*. Berlin, Germany: Springer-Verlag, 1990.

[2] D. Parisi, "La scienza cognitiva tra intelligenza artificiale e vita artificiale," in *Neuroscienze e scienze dell'artificiale: Dal neurone all'intelligenza*, E. Biondi, P. Morasso, and V. Tagliasco, Eds. Bologna, Italy: Patron, 1991, pp. 321–341.

[3] T. Masters, *Signal and Image Processing With Neural Networks—A C++ Sourcebook*. New York: Wiley, 1994.

[4] C. Bishop, *Neural Networks for Pattern Recognition*. Oxford, U.K.: Oxford Univ. Press, 1995.

[5] G. A. Carpenter and S. Grossberg, "A massively parallel architecture for a self-organizing neural pattern recognition machine," *Comput. Vision, Graphics, Image Processing*, vol. 37, pp. 54–115, 1987.

[6] F. Y. Shih, J. Moh, and F. Chang, "A new ART-based neural architecture for pattern classification and image enhancement without prior knowledge," *Pattern Recognition*, vol. 25, no. 5, pp. 533–542, 1992.

[7] C. Hung and S. Lin, "Adaptive Hamming Net: A fast-learning ART 1 model without searching," *Neural Networks*, vol. 8, no. 4, pp. 605–618, 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, pp. 759–771, 1991.

[9] S. Grossberg, "Adaptive pattern classification and universal recoding, I: Parallel development and coding of neural feature detectors," *Biol. Cybern.*, vol. 23, pp. 121–134, 1976.

[10] ——, "Adaptive pattern classification and universal recoding, II: Feedback, expectation, olfaction, and illusions," *Biol. Cybern.*, vol. 23, pp. 187–202, 1976.

[11] G. A. Carpenter and S. Grossberg, "ART2: Self-organization of stable category recognition codes for analog input patterns," *Appl. Opt.*, vol. 26, no. 21, pp. 4919–4930, 1987.

[12] 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, Sept. 1992.

[13] J. R. Williamson, "Gaussian ARTMAP: A neural network for fast incremental learning of noisy multidimensional maps," *Neural Networks*, vol. 9, no. 5, pp. 881–897, 1996.

[14] ——, "A constructive, incremental-learning network for mixture modeling and classification," *Neural Comput.*, vol. 9, pp. 1517–1543, 1997.

[15] G. A. Carpenter, M. N. Gjaja, S. Gopal, and C. E. Woodcock, "ART neural networks for remote sensing: Vegetation classification from Landsat TM and terrain data," *IEEE Trans. Geosci. Remote Sensing*, vol. 35, pp. 308–325, 1997.

[16] G. A. Carpenter, S. Gopal, and C. E. Woodcock, "A neural network method for efficient vegetation mapping," *Remote Sensing Environment*, vol. 70, no. 3, pp. 326–338, Dec. 1999.

[17] A. Baraldi and F. Parmiggiani, "A neural network for unsupervised categorization of multivalued input patterns: An application to satellite image clustering," *IEEE Trans. Geosci. Remote Sensing*, vol. 33, pp. 305–316, 1995.

[18] E. Backer and A. K. Jain, "A clustering performance measure based on fuzzy set decomposition," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-3, pp. 66–75, Jan. 1981.
[19] V. Cherkassky and F. Mulier, *Learning From Data: Concepts, Theory, and Methods*. New York: Wiley, 1998.
[20] T. Kohonen, "The self-organizing map," *Proc. IEEE*, vol. 78, pp. 1464–1480, 1990.
[21] ——, *Self-Organizing Maps*. Berlin, Germany: Springer-Verlag, 1995.
[22] T. Martinetz, G. Berkovich, and K. Schulten, "Neural-gas network for quantization and its application to time-series predictions," *IEEE Trans. Neural Networks*, vol. 4, pp. 558–569, July 1993.
[23] T. Mitchell, *Machine Learning*. New York: McGraw-Hill, 1997.
[24] R. Krishnapuram and J. M. Keller, "A possibilistic approach to clustering," *IEEE Trans. Fuzzy Syst.*, vol. 1, pp. 98–110, 1993.
[25] R. N. Davè and R. Krishnapuram, "Robust clustering method: A unified view," *IEEE Trans. Fuzzy Syst.*, vol. 5, pp. 270–293, 1997.
[26] A. Baraldi and E. Alpaydın, "Simplified ART: A new class of ART algorithms," Int. Comput. Sci. Inst., Berkeley, CA, Tech. Rep. TR-98-004.
[27] G. A. Carpenter and A. H. Tan, "Rule extraction, from neural architecture to symbolic representation," *Connection Sci.*, vol. 7, pp. 3–27, 1977.
[28] J. Huang, M. Georgiopoulos, and G. L. Heileman, "Fuzzy ART properties," *Neural Networks*, vol. 8, no. 2, pp. 203–213, 1995.
[29] Y. Pao, *Adaptive Pattern Recognition and Neural Networks*. Reading, MA: Addison-Wesley, 1989.
[30] P. K. Simpson, "Fuzzy min–max neural networks—Part 2: Clustering," *IEEE Trans. Fuzzy Syst.*, vol. 1, pp. 32–45, 1993.
[31] H. Frigui and R. Krishnapuram, "A robust competitive clustering algorithm with applications in computer vision," *IEEE Trans. Neural Networks*, vol. 21, pp. 450–465, Sept. 1999.
[32] B. Fritzke. (1997) Some competitive learning methods. Draft document. [Online]. Available: http://www.neuroinformatik.ruhr-uni-bochum.de/ini/VIDM/research/gsn/DemoGNG
[33] M. J. Jordan and C. M. Bishop, *An Introduction to Graphical Models and Machine Learning*: draft document, 1998.
[34] J. C. Bezdek and N. R. Pal, "Two soft relatives of learning vector quantization," *Neural Networks*, vol. 8, no. 5, pp. 729–743, 1995.
[35] J. C. Bezdek, T. Reichherzer, G. S. Lim, and Y. Attikiouzel, "Multiple-prototype classifier design," *IEEE Trans. Syst., Man, Cybern. C*, vol. 28, pp. 67–79, 1998.
[36] E. C. Tsao, J. C. Bezdek, and N. R. Pal, "Fuzzy Kohonen clustering network," *Pattern Recognition*, vol. 27, no. 5, pp. 757–764, 1994.
[37] Y. S. Kim and S. Mitra, "Integrated Adaptive Fuzzy Clustering (IAFC) algorithm," in *Proc. 2nd IEEE Int. Conf. Fuzzy Syst.*, vol. 2, 1993, pp. 1264–1268.
[38] N. B. Karayiannis and P. Pai, "Fuzzy algorithms for learning vector quantization," *IEEE Trans. Neural Networks*, vol. 7, pp. 1196–1211, Sept. 1996.
[39] N. B. Karayiannis, J. C. Bezdek, N. R. Pal, R. J. Hathaway, and P. Pai, "Repair to GLVQ: A new family of competitive learning schemes," *IEEE Trans. Neural Networks*, vol. 7, pp. 1062–1071, Sept. 1996.
[40] J. Shi and J. Malik, "Normalized cuts and image segmentation," in *Proc. IEEE Conf. Comp. Vision Pattern Recognition*, Puerto Rico, June 1997.
[41] F. Ancona, S. Ridella, S. Rovetta, and R. Zunino, "On the importance of sorting in "Neural Gas" training of vector quantizers," in *Proc. Int. Conf. Neural Networks '97*, vol. 3, Houston, TX, 1997, pp. 1804–1808.
[42] M. Barni, V. Cappellini, and A. Mecocci, "Comments on 'A possibilistic approach to clustering'," *IEEE Trans. Fuzzy Syst.*, vol. 4, pp. 393–396, 1996.

Andrea Baraldi was born in Modena, Italy, in 1963. He graduated in electronic engineering from the University of Bologna, Bologna, Italy, in 1989. His Master's thesis focused on the development of unsupervised clustering algorithms for optical satellite imagery.

From 1989 to 1990, he was a Research Associate at CIOC-CNR, an Institute of the National Research Council (CNR) in Bologna, and served in the army at the Istituto Geografico Militare in Florence, working on satellite image classifiers and GIS. As a Consultant at ESA-ESRIN in Frascati, Italy, he worked on object-oriented applications for GIS from 1991 to 1993. From December 1997 to June 1999, he joined the International Computer Science Institute, Berkeley, CA, with a postdoctoral fellowship in Artificial Intelligence. Since his master thesis, he has continued his collaboration with ISAO-CNR in Bologna. As a postdoctoral researcher, he currently works at the European Commission Joint Research Centre, Ispra, Italy, in the development and validation of classification algorithms applied to wide area radar mosaics of forest ecosystems. His main interests center on image segmentation and classification, with special emphasis on texture analysis and neural-network applications in computer vision.

Ethem Alpaydın was born on June 23, 1966. He received the B.Sc. degree in 1987 from Boğaziçi University, Istanbul, Turkey, and the Ph.D. degree in 1990 from Ecole Polytechnique Fédérale, Lausanne, Switzerland.

In 1991, he was a Postdoctoral Researcher with the International Computer Science Institute (ICSI), Berkeley, CA. Since October 1991, he has been teaching with the Department of Computer Engineering, Boğaziçi University, where he is now Associate Professor. He held visiting research positions at Massachusetts Institute of Technology, Cambridge, in 1994, ICSI (as a Fulbright scholar) in 1997, and IDIAP in 1998. His research interests are artificial neural networks and machine learning.