HYBRID ARTMAP NEURAL NETWORKS (HART)

Ahmad Al-Daraiseh

creepymaster@yahoo.com
Electrical and Computer Engineering Technology
Farmingdale State College
Farmingdale, NY, 11735

Abstract—In order to reduce the effect of the category proliferation phenomenon in Fuzzy ARTMAP (FAM) and in ellipsoidal ARTMAP (EAM) architectures, The genetic algorithms were used to evolve networks of both architectures called GFAM and GEAM [3][4]. The results were very promising and the category proliferation (CP) phenomenon was minimized in most of the experiments, however, the author noticed that GEAM worked better on some problems while GFAM worked better on others, this triggered the idea of a hybrid Genetic ART architecture that uses categories from both EAM and FAM architectures, this architecture was then called HART. HART evolves networks that were designed to have hyper-rectangular and hyper-ellipsoidal categories. HART was tested on 24 different datasets, the results were compared against those collected from testing FAM, EAM, GFAM and GEAM, HART performed well against all other networks and gave a great balance between accuracy and network size.

Keywords: Neural Networks, ART, FAM, GFAM, GEAM, Hybrid, Genetic Algorithm

I. INTRODUCTION

THE Adaptive Resonance Theory (ART) architecture was developed by Grossberg (1976) [1]. In 1992 Mr. Carpenter developed a Neural Network (NN) called Fuzzy ARTMAP (FAM) [2]. FAM architectures became very popular and were used in the literature to successfully solve many classification problems. Researchers then developed other ART NN's such as EAM [5] and GAM [6] that used different category representations to attain better performance and to reduce the effect of a phenomenon called Category Proliferation (CP) (Creating extra categories for better performance especially when used with noisy data). The authors noted that FAM, GAM and EAM still created far more categories than needed when tried on many datasets; therefore, we came up with the idea of using Genetic Algorithms (GA) to evolve the different ART architectures, the newly created architectures were then called GFAM, GEAM and GGAM [3,4]. The Genetic architectures were tested on 24 different datasets, The results then were compared against those of the original architectures and showed the superiority of the Genetic architectures [3,4].

Although the method described above was very successful, the authors noticed that for some datasets one of the FAM, EAM, GAM architectures produced the best results, while for another dataset another architecture did better. Furthermore, it could also be the case that EAM is better at describing the input patterns in a portion of the input space, while FAM might be able to do a better job at another portion of the input space (and the same dataset). These observations gave birth to the idea of HART that does not a-priori determine which is the best category structure (hyper-rectangle, hyper-ellipsoid) that could best represent the data in various portions of the input space. From another point of view, it is reasonable to think that a problem space would not be suitable to be covered by only one of the geometrical shapes mentioned above, and this could explain the extra nodes created and the lack of accuracy attained by a specific ART architecture. Figure 1.a shows what might be the case when using an unsuitable architecture to cover the input space of a specific problem while 1.b shows the opposite.

The organization of this paper is as follows: In section 2 we present HART. In Section 3, we describe the experiments and the datasets used to assess the performance of HART, and we also compare HART to other ART and Genetic ART networks that attempted to resolve the category proliferation problem in ARTMAP. In Section 4, we summarize our work and draw some conclusions.

II. Evolving Hybrid FAM/EAM networks (HART)

In this article we assume that the reader is familiar with the Fuzzy ARTMAP (FAM) neural network architecture, its training phase, and its network parameters (for more info see [2]). For every classification problem (dataset) that we experimented with we assume that we have a training set, a validation set and a test set. HART can operate in three distinct phases: *the training phase*, the *geometry selection phase* (or Genetic Phase) and the *performance phase.* In the training phase of HART, the user defines the number of the networks the system should generate, referred to as Pop_{size}. The system then creates Pop_{size} trained ART networks, half of them FAM and the other half EAM networks. These FAM and EAM networks are generated by using different values of the baseline vigilance parameter and orders of training pattern presentations, as was done for GFAM and GEAM. For the training of this initial population of Pop_{size} ART networks a list of input patterns/output labels pairs, (i.e. $\{(\mathbf{I}^1, O(\mathbf{I}^1)),...,(\mathbf{I}^r, O(\mathbf{I}^r)),...,(\mathbf{I}^{PT}, O(\mathbf{I}^{PT}))\}$), is repeatedly presented to the FAM/EAM network until it learns the required mapping. Training is over when a user defined maximum list presentation number is reached. After creating a FAM or EAM trained network HART converts it into a chromosome and saves it in the FAM/EAM network container of the HART architecture. A pictorial illustration of the HART architecture in its training phase, consisting of two independently operating FAM and EAM architectures and the associated FAM/EAM chromosome container is shown in Figure 2. In the geometry selection phase, the goal is to find an ART network (HART network) which contains the best types and smallest number of ART categories (rectangles or ellipsoids, or a combination of the two) that achieves good generalization. This HART network is found by starting from an initial population of Pop_{size} HART networks and by applying the GA algorithm to this initial population, in the same way that it was used to produce GFAM and GEAM [3][4]. The distinct and important difference between the initial population of GFAMs, and GEAMs, that we started with then and the initial population of HARTs that we start with here is that the initial population of GFAMs and GEAMs consisted of chromosomes, each one of which contained categories of the same geometrical structure, such as rectangles or ellipsoids. Now, each chromosome in the initial population of HART starts from an initial population of ART networks, whose chromosome contains a mixture of rectangles and ellipsoids. Each member of this initial population chose the rectangles and categories contained in its chromosome randomly from the population of rectangles and categories included in the HART container from HART's training phase. It is important; to also mention that in the geometry selection phase (or genetic phase) HART is called upon to calculate its output for each input pattern in the validation data, i.e., HART is called upon to operate in the performance mode. The steps that HART is going through to produce an output label for an input pattern presented at its input during HART' performance mode are included below.

The HART architecture, in its performance phase, consists of three main layers. These are: the *input layer* (U_1^a), the *category representation layer* (U_2^a), and the *output layer* (U_2^b). The input layer of HART has $2M_a$ nodes, nodes numbered 1 through M_a are connected to all the nodes in U_2^a layer that represent a FAM category, while only the nodes 1 to M_a are connected to those categories in the U_2^a layer that represent an EAM category (remember that EAM does not require complement encoding).

During the performance phase, **a** is fed to U_1^a , so it occupies the first M_a nodes of U_1^a and its complement coded version, $\mathbf{I}$, occupies the $2M_a$ nodes of U_1^a. Layer U_2^a in HART represents all the categories that the HART network possesses, and hence the name *category representation layer*. This layer could have all the nodes as FAM categories, as EAM categories or as a mixture of both (these nodes represent categories that were randomly chosen from the mixture of FAM/EAM categories stored in the FAM/EAM container of HART's architecture, at the end of its training phase). The nodes in the category representation layer are connected to the nodes in the U_1^a layer as shown in Figure 3. Finally, the output layer (layer U_2^b) is the layer that produces the outputs of the network. Every node in the output layer of HART represents one of the labels of the pattern recognition task. The index k $(1 \le k \le N_b)$ designates a generic node in U_2^b; N_b represents the highest index needed to represent all the labels of the pattern classification task at hand.

A. *The Performance Phase of HART*

This phase is similar to those of a FAM or an EAM. The process can be summarized in the following steps:

1. Present an input pattern (from the validation or test set) to the HART network
2. Calculate the CCF function, corresponding to this input pattern, for all the nodes in the U_2^a layer according to the following equations:

   a. FAM Category: $$T(j\,|\,\mathbf{I}) = \frac{M_a - s(\mathbf{w}_j^a) - dis(\mathbf{I},\mathbf{w}_j^a)}{\beta_a + M_a - s(\mathbf{w}_j^a)}$$

   b. EAM Category: $$T(j\,|\,\mathbf{I}) = \frac{D - s(\mathbf{w}_j^a) - dis(\mathbf{I},\mathbf{w}_j^a)}{\beta_a + D - s(\mathbf{w}_j^a)}$$
3. Check the label of this node J. This will be the predicted label of the HART network for this input pattern.
4. If more patterns are still in the list (validation or test set) present the next input pattern to the HART network. Otherwise, the performance phase is completed.

B. *Geometry Selection Phase (Genetic Phase) of HART*

As it was the case for GFAM and GEAM we start with an initial population of Pop_{size} HARTs that we evolve. The GA parameters used for the evolution of FAM and EAM networks are also used for HART networks with some additions. This process follows the following steps:

Step 1: The process starts by creating Pop_{size} chromosomes from the FAM and EAM networks. To eliminate any advantage of a FAM over an EAM network, we extract all the categories from all networks into a group of categories. We then populate each chromosome in the generation with randomly selected categories up to a random length (# of categories). At this stage all one-point categories are eliminated. As shown in figure 4, a chromosome is composed of a number of categories. The categories can be either FAM's or EAM's. FAM categories encode vectors $\mathbf{u}$ and $\mathbf{v}$ (end points), an integer *l* for the label, and an integer *t* for the type of the category, while EAM categories encode vectors $\mathbf{m}$ (center) and $\mathbf{d}$ (direction), *l* represent the label, r encodes the radius, μ encodes the axis ratio and *t* type.

We denote the category of a trained HART network with index p $(1 \leq p \leq Pop_{size})$ by $\mathbf{w}_j^a(p)$, where $\mathbf{w}_j^a(p) = \mathbf{w}_j^{a,FAM}(p) = (\mathbf{u}_j(p), (\mathbf{v}_j(p)^c)$ or $\mathbf{w}_j^a(p) = \mathbf{w}_j^{a,EAM}(p) = (\mathbf{m}_j, \mathbf{d}_j, r_j)$ and the label of this category by $l_j(p)$ for $1 \leq j \leq N_a(p)$.

Step 2: Evolve the chromosomes of the current generation by executing the following sub-steps:

Sub-step 2a: Calculate fitness by converting each chromosome into a HART network, feeding into it the validation set and calculating the Percentage of Correct Classification $PCC(p)$ exhibited by each network *p*, if this network possesses $N_a(p)$ categories, the fitness is calculated according to equation (1).

$$Fit(p) = \frac{(Cat_{\max} - N_a(p)) \cdot PCC^2(p)}{\dfrac{100}{Cat_{\min}} - \dfrac{PCC(p)}{N_a(p)} + \varepsilon} \qquad (1)$$

where, $Cat_{\min}$ and $Cat_{\max}$ are the minimum and maximum number of categories that a HART network is allowed to have, and ε is a small positive number.

Sub-step 2b: Initialize an empty generation (referred to as *temporary generation*).

Sub-step 2c: Move the best NC_{best} (user defined) chromosomes from the current generation and copies them to the temporary generation.

Sub-step 2d: The remaining $Pop_{size} - NC_{best}$ chromosomes in the temporary generation are created by crossing over two parents from the current generation. Using the tournament selection method, two chromosomes are chosen say p, p', two random numbers n, n' are generated from the index sets $\{1, 2, ..., N_a(p)\}$ and $\{1, 2, ..., N_a(p')\}$, and the cross over is curried over as shown in figure 5.

Sub-step 2e: Apply the Cat_{add} operator to add a new category to every chromosome created in step 2d with a user defined probability $P(Cat_{add})$. With a 0.5 probability the new category is chosen to be a FAM or an EAM category. All the fields of the category are then filled randomly. If $Cat_{\max}$ is exceeded no addition is done.

Sub-step 2f: Apply the Cat_{del} operator to remove one of the categories of every chromosome created in step 2e with a user defined probability $P(Cat_{del})$. If $Cat_{\min}$ is crossed, no removal takes place.

Sub-Step 2g: Mutate every chromosome created by step 2f as follows: with a user defined probability $P(mut)$ every category is mutated. If a FAM category is chosen, with 50% chance either $\mathbf{u}$ or $\mathbf{v}$ is mutated, every element of this vector is mutated by adding to it a small number drawn from a Gaussian distribution with 0 mean and 0.01 standard deviation . On the other hand, an EAM category's center $\mathbf{m}$ or direction $\mathbf{d}$ is mutated in a similar fashion as above.

Step 3: If evolution has reached the maximum number $Gen_{\max}$ of iterations, then calculate the performance of the best-Fitness HART network on the test set and report classification accuracy and number of categories that this Best-Fitness HART network possesses. If the maximum number of iterations has not been reached yet, go to step 2 to evolve one more population.

III. HART PERFORMANCE

To examine the performance of HART we performed a number of experiments on real and simulated datasets. The collections of simulated and real datasets are depicted in Table 1. The legend of Table 1 explains briefly the simulated datasets, while the real datasets were extracted from the UCI repository.

In all the experiments conducted with the aforementioned databases we had at our disposal a training set (used to design the trained ART network), a validation set (used to optimize the trained ART network), and a test set used to assess the performance of the optimized trained ART network.

A. Parameter Settings

We used the same default set of parameters used for GFAM to run all the experiments of HART with one modification, and the results were excellent. In HART‘s case we avoided the experimentation (applied to GFAM) to choose good default values for the GA parameters. Hence, HART is produced by first initializing a population of 20 trained FAM and EAM networks (they were trained with different values of the baseline vigilance parameter and different orders of training pattern presentations), and by evolving them for 500 generations. In particular, the GA parameters used for the creation of HART were: $\rho_a^{\min}$ = 0.1, $\rho_a^{\max}$ = 0.75, β_a =0.1, Pop_{size}= 20, $Gen_{\max}$ = 500, NC_{best} = 3, $Cat_{\min}$ = 1, $Cat_{\max}$ = 300, $P(Cat_{add})$ =0.1, $P(Cat_{del})$ =0.1, $P(mut)$ = 1/Na. HART is the network that attains the highest value of the fitness function at generation 500 of the evolutionary process.

B. Experimental Results and Comparisons of HART with other ART and Genetic ART Networks

After running HART on the datasets in Table 1, we produce the accuracy and size of the HART network that attained the highest value of the fitness function at the last generation of the evolutionary process. Table 2 lists the accuracy and the size of this HART network as well as the accuracy and the size of other ART and genetic ART architectures for the same dataset.

In Table 2 we are comparing HART’s performance with the performance of the following networks: ssFAM [2], ssEAM [5], GFAM [3] and GEAM [4]. We chose these networks for a reason. Each one of these ART networks at the time of their introduction into the literature emphasized that they were addressing the category proliferation problem in ART. More details about the specifics of each one of these networks can be found in their associated references. For the purposes of this paper it suffices to know that ssEAM covers the space of the input patterns with ellipsoids. Furthermore ssFAM, and ssEAM allow a category (hyper-rectangle or ellipsoid or hyper-dimensional) to encode patterns of different labels provided that the plurality label of a category exceeds a certain, user-specified, threshold. Finally, GFAM and GEAM use a genetic algorithm similar to the one used in this work to evolve FAM and EAM networks respectively.

In Table 2, the first column is the index of the dataset used in the figures. Column 2 is the name of the database that we are experimenting with, while columns 3-7 of Table 2 contain the performance of the designated networks. The HART performance reported corresponds to the accuracy on the test set and the number of categories created by the FAM/EAM network that attained the highest value of the fitness function at the last generation of the evolutionary process; this is the case for the GFAM and GEAM as well. For the other ART networks the reported performance is the performance of the ART network that achieves the highest value of the fitness function amongst the trained ART networks with different network parameter settings (e.g., in ssFAM the best network was determined after training more than 20000 different ssFAM networks for every single dataset each of which with different values of the choice parameter, vigilance parameter, order of pattern presentation, and amount of mixture of labels allowed within a category).

The performance of HART, as it is evidenced by the results in Table 2, is verified by some obvious observations. For instance, HART's performance on datasets 1-12 (Gaussian datasets of known amount of overlap) is nearly optimal; for example the best performance on the G6c-40 problem (6 class Gaussian dataset of 40% overlap) is a classifier with 6 categories and 60% correct classification, and HART is a classifier with 6 categories and 59.83% of correct classification. Similarly, in the CINS problem the optimal classifier would require 2 categories and attain a 100% correct classification; HART is a 2 category classifier exhibiting a 99.37% of correct classification. Finally, all of the real problems reported here, MOD-IRIS, ABALONE and PAGE, also gave very good results 94.88%, 64.1% and 95.73% of correct classification respectively, by creating 2,3 and 3 categories respectively only.

The performance of HART's was compared with the performance of the following networks: ssFAM, ssEAM, GFAM, GEAM, GGAM, ssGAM[6], and safe micro-ARTMAP (Table 2 shows the results of the first four networks because of the lack of space only, moreover GFAM outperformed those networks on the same datasets see [3]).

According to the results in Table 2, in all instances (except minor exceptions) the accuracy of HART (generalization performance) is higher than the accuracy of the other ART network (where ART is ssEAM, ssEAM). According to the results in Table 2, in all instances (with no exceptions) the size of GGAM is either equal or smaller than the size of the other ART networks (where ART is ssEAM, ssEAM), sometimes even by a factor of 15. For example, the PCC of HART is 12% better than that of ssFAM for dataset 21 while its size is almost half.

From a quick look at table 2, it is clear that in the Gaussian databases all of the genetic ART modules performed very well with minor differences. On the structure within structure databases however, the performance was different. Table 2 shows that GFAM gave better accuracy on databases 14, 15, 18 and 21 while GEAM gave better accuracy on databases 13, 16, 17, 19 and 20. If we investigate further, we find that GFAM gave better accuracy when the problem didn't have a circle in it, on the other hand, GEAM gave better results on those problems that have circles in them, *this was the reason behind the development of HART*.

HART performance compares very well to the best of the two on all problems, for example, HART gave 99.37% and 99.3% for databases 16 and 17 respectively with only 2 categories in each case, this performance is very close to that of GEAM on the same dataset of 99.99% with 2 categories, on the other hand, HART gave 98.5% accuracy on database 15 with only 7 categories, while GFAM gave 97.2% with 7 categories. The bottom line is HART performed very well on all the datasets described above.

It is worth pointing out that the better performance of HART is attained with similar time to GFAM or GEAM and reduced computations compared with the computations needed by the alternate networks (ssEAM, ssEAM, ssGAM, safe micro-ARTMAP[7]) as shown in [4].

IV. SUMMARY AND CONCLUSIONS

HART is a novel approach of mixing different types of ART categories to obtain better coverage of the input space. HART used two types of categories, namely: FAM categories and EAM categories to enhance its performance.

HART used genetic algorithms to solve the category proliferation problem in ART. This method relies on evolving a population of trained ART networks, and more specifically Ellipsoidal ARTMAP (EAM) and Fuzzy ARTMAP (FAM) neural networks. The evolution of trained FAM's and EAM's creates an ART network, referred to as HART.

In [3,4] we defined a methodology of evolving trained FAM networks, resulting in GFAM. This methodology was also applied successfully for the evolution of FAM/EAM networks, resulting in HART. In [3,4] we experimented with a number of datasets that helped us identify good default parameter settings for the evolution of FAM The same parameters and settings used for the evolution of FAM networks (GFAM) were also used for the evolution of FAM and EAM networks (HART).

Experiments with HART indicate that HART is superior to a number of other ART techniques (ssEAM, ssEAM, ssGAM, safe micro-ARTMAP) that have been introduced into the literature to address the category proliferation problem in ART. More specifically, HART gave a better generalization performance (in almost all problems tested) and a smaller size network (in all problems tested), compared to these other ART techniques. What is also worth mentioning is that HART outperformed those other ART techniques by requiring only a fraction of the computations needed by these other networks.

Based on a simple comparison, HART also outperformed the other genetic ART modules we introduced in [3,4] on many databases and gave almost as good results on the rest of the databases.

REFERENCES

[1] Grossberg, S. (1976). Adaptive pattern recognition and universal recoding II: Feedback, expectation, olfaction, and illusions, *Biological Cybernetics*, 23, 187-202.

[2] Carpenter, G.A., Grossberg, S., Markuzon, N., & Reynolds, J.H. (1992). Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multi-dimensional maps, IEEE Trans. Neural Networks, 3 (5), 698-713.

[3] Al-Daraiseh, A., M. Georgiopoulos, G. Anagnostopoulos, A. S. Wu, M. Mollaghasemi.(2007), **"***GFAM: Evolving Fuzzy ARTMAP Neural Networks*", *Neural Networks* , Volume 20, Number 8,pp. 874-892.

[4] Al-Daraiseh, A. (2006). Genetically engineered adaptive resonance theory (art) neural network architectures, Doctoral Thesis, University of Central Florida, Orlando, Florida.

[5] Anagnostopoulos, G.C. & Georgiopoulos, M. (2001). Ellipsoid ART and ARTMAP for incremental clustering and classification, in Proc. of the IEEE-INNS International Joint Conference on Neural Networks, pp. 1221-1226, Washington, DC, July 14-19.

[6] Williamson J.R. Gaussian ARTMAP: A Neural Network for Fast Incremental Learning of Noisy Multidimensional Maps, Neural Networks, Volume 9, Number 5, July 1996 , pp. 881-897

[7] Gomez-Sanchez, E., Dimitriadis, Y.A., Cano-Izquierdo, J.M., Lopez-Coronado, J.: Safe-μARTMAP: a new solution for reducing category proliferation in Fuzzy ARTMAP, in *Proc. of the IEEE International Joint Conference on Neural Networks*, Vol. 2, pp. 1197-1202, July 15-19, 2001.

[8] Yao, X., "Evolving Artificial Neural Networks," Proceedings IEEE, Vol. 87, No. 9, pp. 1423-1447, 1997.

Figures

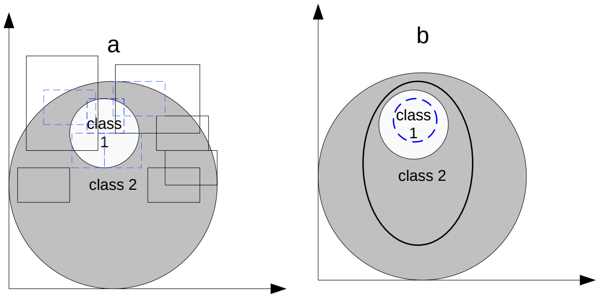

Figure 1: A hypothetical diagram shows what might be the difference between using a non-suitable module (a) and using a suitable one (b).

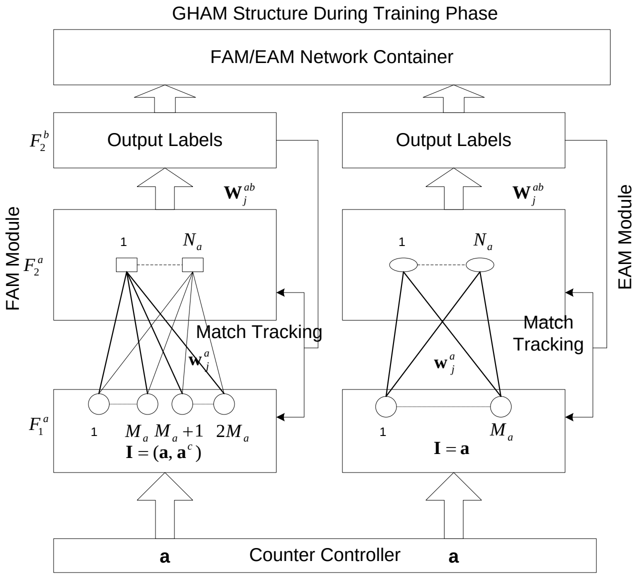

Figure 2: HART During the training phase

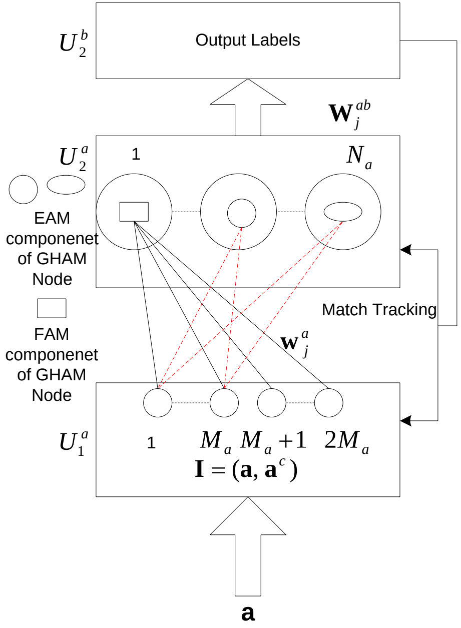

Figure 3: HART During the performance phase

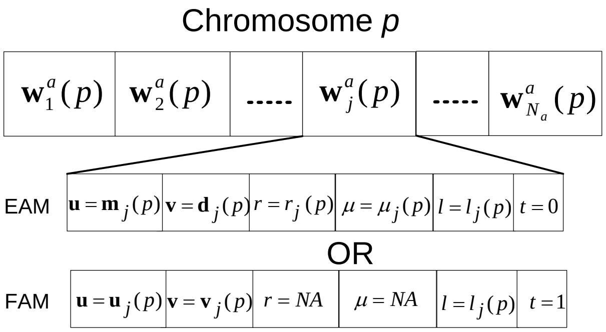

Figure 4: HART Chromosome structure

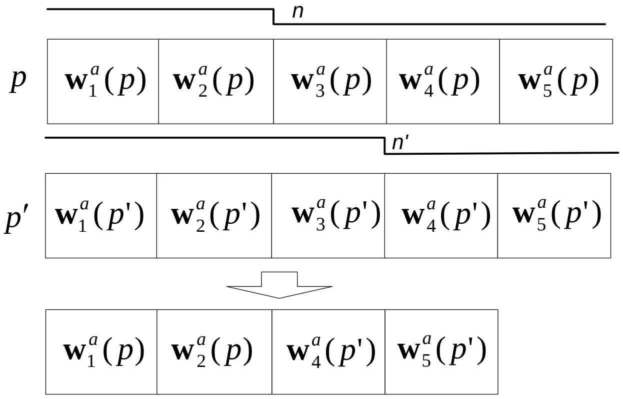

Figure 5: Crossover implementation

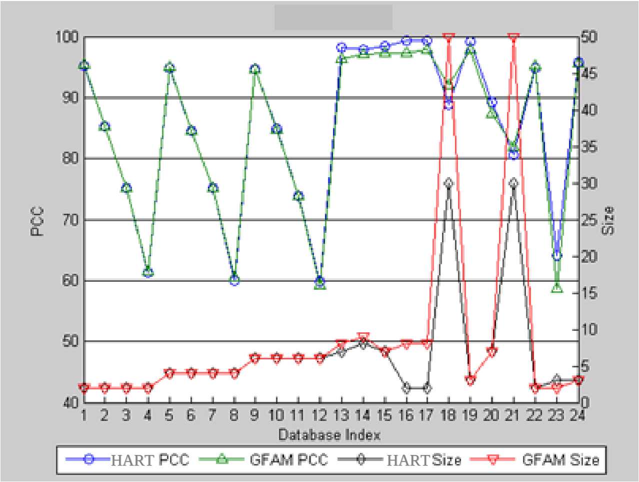

Figure 6: Performance and Size comparison of HART vs GFAM

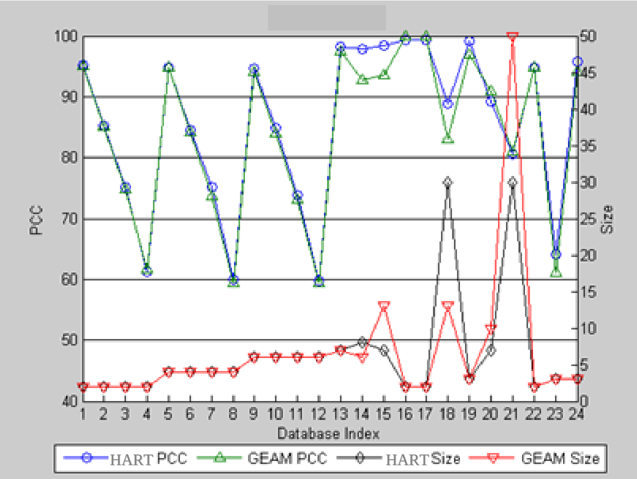

Figure 7:Performance and Size comparison of HART vs GEAM

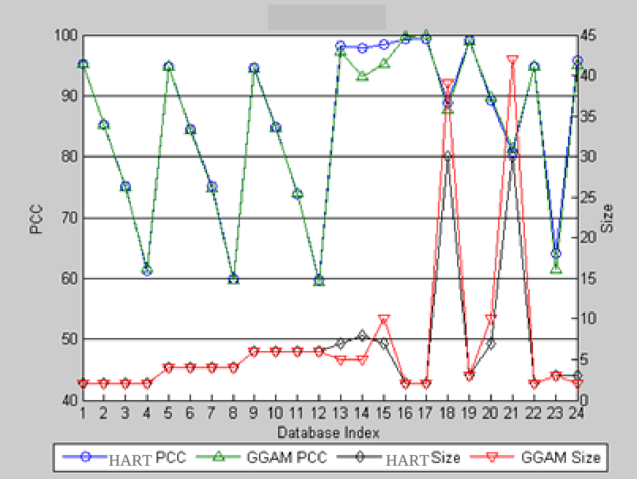

Figure 8: Performance and Size comparison of HART vs GGAM

Tables

Table 1: Databases used in the GFAM experiments, where G*c_** represent a Gaussian dataset with * classes and ** overlap, 16-24 represent a shape within a shape dataset where Ci is a circle and Sq is a square; in the last two datasets WN means with noise (10%)

	Database Name	# Numerical Attributes	# Classes	% Major Class	Expected Accuracy
1	G2c-05	2	2	1/2	0.95
2	G2c-15	2	2	1/2	0.85
3	G2c-25	2	2	1/2	0.75
4	G2c-40	2	2	1/2	0.6
5	G4c-05	2	4	1/4	0.95
6	G4c-15	2	4	1/4	0.85
7	G4c-25	2	4	1/4	0.75
8	G4c-40	2	4	1/4	0.6
9	G6c-05	2	6	1/6	0.95
10	G6c-15	2	6	1/6	0.85
11	G6c-25	2	6	1/6	0.75
12	G6c-40	2	6	1/6	0.6
13	MOD-IRIS	2	2	1/2	0.95
14	ABALONE	7	3	1/3	0.6
15	PAGE	10	5	0.832	0.95
16	4Ci/Sq	2	5	0.2	1
17	4Sq/Sq	2	5	0.2	1

18	7Sq	2	7	1/7	1
19	1Ci/Sq	2	2	0.5	1
20	1Ci/Sq/0.3:0.7	2	2	0.7	1
20	5Ci/Sq	2	6	1/6	1
21	2Ci/Sq/5:25:70	2	3	0.7	1
22	2Ci/Sq/20:30:50	2	3	0.5	1
23	7SqWN	2	6	1/7	0.9
24	5Ci/SqWN	2	6	1//6	0.9

Table 2: HART performance and size compared to other genetic ART architectures (ss is the Semi-Supervised version)

#	Database Name	HART		ssFAM		ssEAM		GFAM		GEAM	
1	G2c-05	95.2	2	94.90	2	94.94	2	95.36	2	95	2
2	G2c-15	85.22	2	84.80	3	85.20	2	85.30	2	85.12	2
3	G2c-25	75.16	2	74.60	2	74.50	2	75.08	2	74.74	2
4	G2c-40	61.24	2	61.34	3	60.98	2	61.38	2	61.6	2
5	G4c-05	94.9	4	94.10	7	94.14	4	95.02	4	94.8	4
6	G4c-15	84.6	4	81.40	11	83.20	4	84.46	4	84.22	4
7	G4c-25	75.18	4	70.80	9	72.72	4	75.20	4	73.7	4
8	G4c-40	59.96	4	58.48	14	55.62	13	60.60	4	59.5	4
9	G6c-05	94.7	6	91.42	11	93.80	7	94.68	6	94.12	6
10	G6c-15	84.85	6	81.11	7	81.80	6	84.71	6	84.03	6
11	G6c-25	73.90	6	69.62	15	71.10	7	73.90	6	73.04	6
12	G6c-40	59.83	6	56.35	17	54.21	17	59.19	6	59.35	6
13	4Ci/Sq	98.16	7	87.23	18	94.68	5	96.32	8	97.4	7
14	4Sq/Sq	97.9	8	97.24	13	88.89	5	97.12	9	92.76	6
15	7Sq	**98.5**	**7**	97.26	16	88.5	19	**97.2**	**7**	**93.46**	**13**
16	1Ci/Sq	**99.37**	**2**	92.97	8	97.02	8	**97.2**	**8**	**99.9**	**2**
17	1Ci/Sq/ 0.3:0.7	**99.3**	**2**	93.21	8	97.13	8	**97.8**	**8**	**99.9**	**2**
18	5Ci/Sq	**88.87**	30	81.95	52	78.68	87	**92**	**50**	**83.03**	**13**
19	2Ci/Sq/ 20:30:50	99.2	3	90.24	12	97.01	3	97.87	3	96.86	3
20	7SqWN	89.3	7	80.15	24	75.23	32	87.3	7	90.82	10
21	5Ci/SqWN	80.67	30	68.39	57	69.2	136	81.97	50	81	50
22	MOD-IRIS	94.88	2	93.41	8	94.54	2	95.31	2	94.81	2
23	ABALONE	64.1	3	59.52	6	56.80	7	58.73	2	61.00	3
24	PAGE	95.73	3	90.63	3	89.54	3	95.59	3	94.12	3