

Training of Multilayer Perceptron Neural Networks by Using Cellular Genetic Algorithms

M. Orozco-Montegudo, A. Taboada-Crispí, and A. Del Toro-Almenares

Center for Studies on Electronics and Information Technologies, Universidad Central de Las Villas, Carretera a Camajuaní, km 5 ½, Santa Clara, VC, CP 58430, Cuba
morozco@uclv.edu.cu, ataboada@uclv.edu.cu, anestodt@uclv.edu.cu

Abstract. This paper deals with a method for training neural networks by using cellular genetic algorithms (CGA). This method was implemented as software, CGANN-Trainer, which was used to generate binary classifiers for recognition of patterns associated with breast cancer images in a multi-objective optimization problem. The results reached by the CGA with the Wisconsin Breast Cancer Database, and the Wisconsin Diagnostic Breast Cancer Database, were compared with some other methods previously reported using the same databases, proving to be an interesting alternative.

Keywords: Neural networks, genetic algorithms, cellular automata, multi-objective classification.

1 Introduction

The trend of using Multilayer Perceptron Neural Networks (MLP) [1] for the solution of classification problems in pattern recognition applications is understandable due to their capacity to imitate the nature of the human brain (learning capacity), and the fact that their structure can be formulated mathematically. The functionality of the topology of the MLP is determined by a learning algorithm able to modify the parameters of the net. The algorithm of Backpropagation (BP), based on the method of steepest descent [1] in the process of upgrading the connection weights, is the most commonly used by the scientific community. The main limitations and problems that present the BP algorithm in training the MLP are exposed in [2]. Recently, numerous works have been reported trying to overcome their main limitations. However, the topology selection issue [3] for MLP still leaves margin for improvements. At the same time, it is not clear what algorithm, or combination of algorithms, is the most appropriate to achieve the objective of reducing complexity of the classifiers and, simultaneously, increasing their benefits in terms of classification effectiveness, for a particular application, and much less for one with a more general character.

On the other hand, Evolutionary Algorithms (EA) have demonstrated great effectiveness to solve problems of Multiobjective Optimization (MO) [4]. In addition, the use of Hierarchical Codification [5] (Fig. 1), combined with the EA, provides advantages in the determination of solutions for problems where the determination of a good structure (ignored *a priori*), is of vital importance. The enormous capacity of computation of the Cellular Automata (CA) [6] gives, in conjunction with EA, a very

desirable characteristic: the population, when having a certain space distribution, will be more representative of the search space and, therefore, the method will be less vulnerable to the phenomenon of premature convergence.

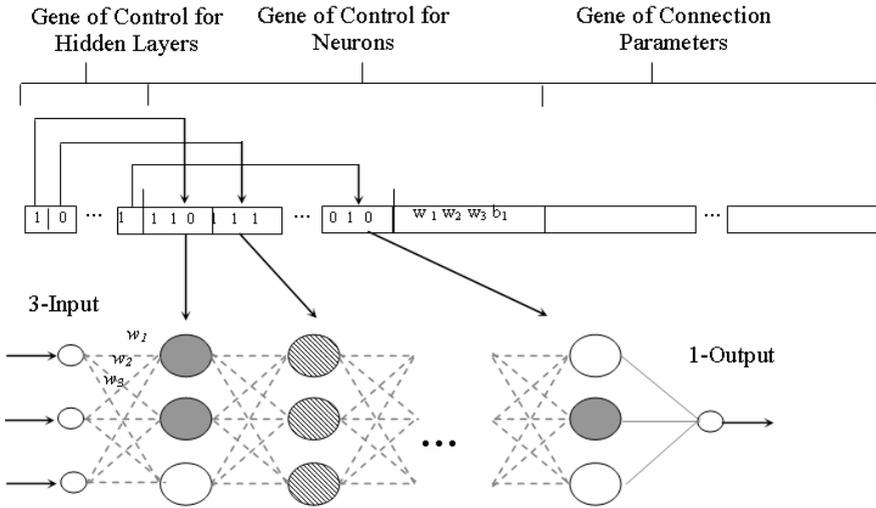


Fig. 1. MLP Hierarchical Codification

2 Materials and Methods

This work is conceived to build general purpose classifiers for pattern recognition, so its use is feasible for any classification problem where it is necessary to discern between two or more classes. This kind of problem is very common, therefore, the field of application of the developed method is very wide, and the definition of materials to be used depends on the problem to be solved.

2.1 CGA for the Construction of Binary Classifiers Based on MLP

A cellular genetic algorithm (CGA) [7] is represented by a probabilistic cellular automaton where the state of each cell is represented by a chromosome.

Each cell should reproduce with some other of its neighborhood, being the descending chromosomes those that substitute the previous generation, according to the replacement technique that is used. The differences between CGA and conventional genetic algorithms reside in three fundamental aspects:

1. There is a space distribution that determines the reproduction processes.
2. The selection process is for every element of the population (better or worse adapted) and the candidates to mate will be only those that are in the neighborhood inside the cellular automata.

3. The replacement will be for the population's individuals and it will be in:
 - Generational Genetic Algorithm, where the offspring always replaces the progenitor.
 - Elitist replacement, where the replacement occurs only if the offspring has more adaptation than its father.

These characteristics guarantee a better representation of the population for the search space, for being the reproduction mechanism less elitist and the population's individual compete only with its neighborhood.

2.2 Punctuation by Means of Successive Refinement of the Pareto Optimal Set

From the multiobjective formulation of the problem of construction of binary classifiers, and the description of the cellular genetic algorithms, the combination of both is presented to create the CGA that solves the outlined problem.

Fig. 2 shows the complete process that governs the developed CGA. The relationships among the main stages that compose this process are illustrated here. Next, every stage is described, as well as the interactions among them.

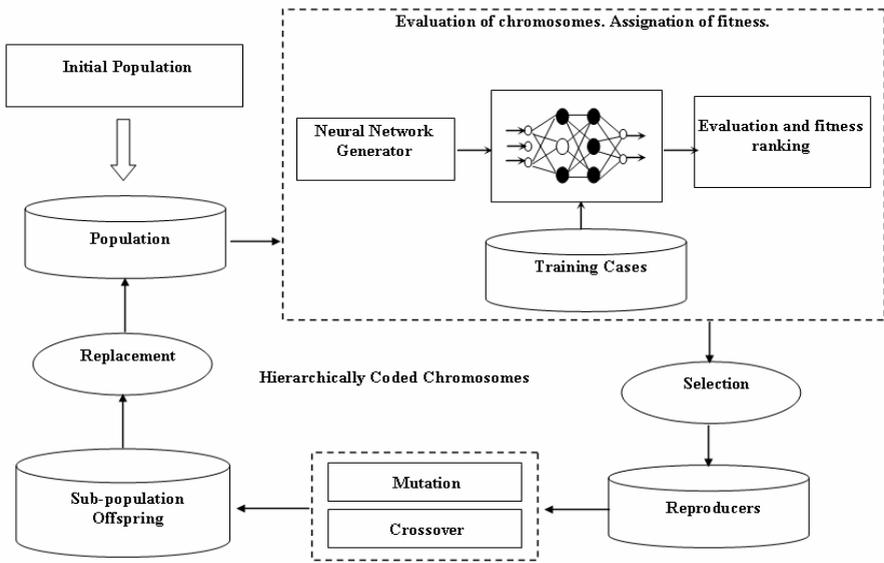


Fig. 2. Scheme for construction of classifiers based on MLP by using CGA

Initial population: A population of chromosomes where each chromosome codes a MLP (topology and connection weight randomly assigned) that constitutes a potential solution of the problem of MO to solve.

Evaluation and punctuation of chromosomes: The assignment of fitness is carried out by using the punctuation outline developed in [8]. The ranking of an individual is calculated as the weighting sum of the dominances of an individual in each one of the corresponding schemes, obtained by suppressing objective functions corresponding to

the problem. This method of fitness assignment reduces the effect introduced by not dominated solutions in a local way, the specialization phenomenon, and it is capable, also, of sampling uniformly regions of the space of solutions.

Selection: A stochastic scheme of selection is used to allow that the individuals with more fitness have bigger probability of being selected and, as a result, they can give place to a bigger number of descending.

Mutation operators and crossover: The characteristics of the structure used in the Hierarchical Code of the chromosomes, presuppose the employment of specific genetic operations, developed for levels, by using any variant of the traditional methods: one-point crossover, multipoint crossover, or uniform crossover [2].

Subpopulation of offspring: This stage comprises those descendants that will participate in the replacement process. This offspring subpopulation will be the initial population's image and each individual has an offspring that occupies, in the subpopulation, the same place that the father in the initial population.

Replacement: A strategy of generational or elitist replacement is used. The process of optimization finishes when a specific number of iterations has been reached.

2.3 Evaluation of the Method to Train Multilayer Perceptron by Using CGA

In this work, specifically, the model developed for the problem of diagnostic of breast cancer will be used. Two standard databases, the Wisconsin Breast Cancer Database (WBC) and the Wisconsin Diagnostic Breast Cancer Database (WDBC) are used for evaluation. Next, these databases are briefly described, along with the works where they are used for evaluation of algorithms of classification.

Wisconsin Breast Cancer Database (WBC). The Wisconsin Breast Cancer Database (WBC) was obtained in the Hospitals of the University of Wisconsin, USA. This contains 699 examples belonging to two different classes: malign and benign. Sixteen of these cases present incomplete information, so only 683 were used, 444 (65%) of the Benign Class and 239 (35%) of Malign class. Each case is described by nine attributes, corresponding to subjective evaluations (Clump Thickness, Uniformity of Cell Size, Uniformity of Cell Shape, Marginal Adhesion, Single Epithelial Cell Size, Bare Nuclei, Bland Chromatin, Normal Nucleoli, and Mitoses), with values between 1 and 10, corresponding to descriptions of the cells obtained by means of microscopic examination. An additional attribute indicates the class associated to this description.

The methods reported with the WBC and used to compare with the results of the cellular genetic algorithms (CGA) here implemented are:

- M1: Uses the multi-surface method [9], [10], [11].
- M2: Uses instance-based learning [12].
- M3: Uses linear programming to train neural networks [13].
- M4: Uses evolutionary algorithms to build classifiers based on MLP, by means of elimination of connections among the hidden neurons [3].
- M5: Uses neural networks after a previous process of selection of features [14], [15].
- M6: Uses a group of extracted rules of an MLP [16].
- M7: Uses a fuzzy system built by means of evolutionary algorithms [17].

- M8: Uses evolutionary algorithms with hierarchical code to build classifiers based on MLP [2].
- M9: Uses classification rules by means of a scheme of weighting of the nearest neighborhood method [18].

Wisconsin Diagnostic Breast Cancer Database (WDBC). The Wisconsin Diagnostic Breast Cancer (WDBC), similarly, was obtained in the Hospitals of the University of Wisconsin. This database contains a total of 569 cases, 357 Benign (63%) and 212 Malign (37%). Observations are described by 30 attributes corresponding to three statistical values that are calculated from ten geometric measurements of the cells (Radius, Texture, Perimeter, Area, Softness, Compactness, Concavity, Number of Points of Concavity, Symmetry, and Fractal Dimension), plus one corresponding to its classification: benign or malign. The observations are based on descriptions of the cells, obtained by means of the microscopic observation.

The methods reported with the WDBC and used to compare with the results of the cellular genetic algorithms (CGA) here implemented are:

- D1: Uses Machine Learning with linear programming [19].
- D2: Uses classification rules by means of an outline of weighting of the nearest neighborhood method [18].

During the processing stage, in this work (CGA), the attributes of the WDBC database were normalized between 0 and 1 by using a linear transformation.

Software CGANN-Trainer. To validate the proposed method, the software CGANN-Trainer was implemented. This software allows changing the input parameters of the algorithm such as: MLP topology (the widest), transfer functions, objective functions to optimize with their respective grade of importance, dimensionality, and type of neighborhood of the CA, crossover and mutation probabilities of the genes of every layer in which the chromosome is divided. The process of training can be supervised by monitoring the evaluations of every objective function. When finishing, the nets with the best benefits can be saved in a text file, which can be used by other applications later.

Formulation of optimal binary classifiers, based on MLP, as a problem of multiobjective optimization. The formulation of the problem for the construction of classifiers based on MLP as a problem of MO tries to find a binary classifier, based on a MLP that minimizes some of the objective functions ($f_1, f_2, f_3, f_4, f_5, f_6$) related with the acting of the classifier and, simultaneously, with the complexity of the associated topology.

Five of these objective functions (f_1, f_2, f_3, f_4, f_5) are calculated by using well-known indexes: Sensibility (Se), Specificity (Sp), Positive Predictivity (PP), Negative Predictivity (NP) and Rate of Classification (RC). In turn, these approaches are calculated by using four basic measures recommended by the AAMI (American for the Advancement of Medical Instrumentation): True Positive (TP), False Positive (FP), True Negative (TN) and False Negative (FN).

Sensibility (Se): Is the fraction of elements of the main class (in this case, Malign) correctly classified and it is calculated as,

$$Se = \frac{TP}{TP + FN} = 1 - \frac{FN}{TP + FN} . \tag{1}$$

Specificity (*Sp*): Is the fraction of elements of the complementary class (in this case, Benign) correctly classified and it is calculated as,

$$Sp = \frac{TN}{TN + FP} = 1 - \frac{FP}{TN + FP} . \tag{2}$$

Positive Predictivity (*PP*): Is the fraction of elements of the main class (in this case, Malign) correctly classified with regard to the positive classifications and it is calculated as,

$$PP = \frac{TP}{TP + FP} = 1 - \frac{FP}{TP + FP} . \tag{3}$$

Negative Predictivity (*NP*): Is the fraction of elements of the complementary class (in this case, Benign) correctly classified with regard to the negative classifications and it is calculated as,

$$NP = \frac{TN}{TN + FN} = 1 - \frac{FN}{TN + FN} . \tag{4}$$

Rate of Classification (*RC*): Is the total fraction of correct classifications and it is calculated as,

$$RC = \frac{TP + TN}{TP + TN + FP + FN} = 1 - \frac{FP + FN}{TP + TN + FP + FN} . \tag{5}$$

In this context, *TP* is the number of elements of the main class (Malign) correctly classified; *FP* and *TN* correspond to the number of elements of the complementary class (Benign) incorrectly and correctly classified, respectively, and *FN* is the number of elements of the main class incorrectly classified.

From these indexes, some of the following objective functions were considered to minimize, during the process of optimization:

$$f_1 = 1 - Se = \frac{FN}{TP + FN} , \tag{6}$$

$$f_2 = 1 - Sp = \frac{FP}{TN + FP} , \tag{7}$$

$$f_3 = 1 - PP = \frac{FP}{TP + FP} , \tag{8}$$

$$f_4 = 1 - NP = \frac{FN}{TN + FN} , \text{ and} \tag{9}$$

$$f_5 = 1 - RC = \frac{FP + FN}{TP + TN + FP + FN} . \tag{10}$$

In general, it is desirable that these objective functions, simultaneously, reach values near to zero. However, solutions of compromise are usually selected. Among the used indexes, sensibility is generally considered the most critical one. This is given by

the importance that has to classify correctly an element of the main class (Malign), in comparison with the false alarms that can be emitted.

The sixth objective function (f_6) is related to the complexity of the topology associated to the classifier and it is calculated as the number of active neurons (N_A) of the MLP divided by the maximum number of neurons that admits the topology (N_{Ad}):

$$f_6 = \frac{N_A}{N_{Ad}} \cdot \tag{11}$$

The specification of the grade of importance, that is, the associated weight, of every objective function is determined by the specific characteristics of the problem to solve and corresponds to the specialist its assignment.

By means of the optimization of these four objective functions simultaneously, it is guaranteed that the classifiers have a drop error rate, as well as a topology of reduced complexity, increasing the benefits during their exploitation.

During the process of optimization were used the functions f_1 , f_5 and f_6 . f_1 depends directly on the FN, which is the most sensitive variable of a problem of classification of cancerous cells. The function f_5 involves all the other ones. f_6 guarantees to find the simplest classifier.

Experiment 1: WBC database classification

Training Cases: 333 cases from WBC (49 %), randomly selected. The rest (51 %) was used as the test set.

Topology of the MLP:

9 input neurons; 2 hidden layers; 6 neurons per layer; and one output neuron.

Hidden layer transfer function: $lsigm(n) = \frac{1}{1 + e^{-n}}$ (12)

Output layer transfer function: $hardl(n) = \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases}$ (13)

Population Size: 121

Population Topology: Square with Moore (8-element) neighborhood.

Objective functions (to minimize): f_1, f_5, f_6

Weights associated: 0.45, 0.45, 0.10, for f_1, f_5, f_6 , respectively.

Replacement: Elitist.

Experiment 2: WDBC database classification

Training Cases: 350 cases from WDBC (61 %), randomly selected. The rest (39 %) was used as the test set.

Topology of the MLP:

30 input neurons; 2 hidden layers; 8 neurons per layers; and one output neuron.

Hidden layer transfer function: $\tanh(n) = \frac{e^n - e^{-n}}{e^n + e^{-n}}$ (14)

Output layer transfer function: $hardl(n) = \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases}$

Population Size: 121

Population Topology: Square with Moore (8-element) neighborhood.

Objective functions (to minimize): f_1, f_5, f_6

Weights associated: 0.45, 0.45, 0.10, for f_1, f_5, f_6 , respectively.

Replacement: Elitist.

3 Results and Discussion

Fig. 3 and Fig. 4 show how the sensibility and the rate of classification (actually f_1 and f_5) behaved during the first 200 iterations of the training process in the experiments 1 and 2, respectively.

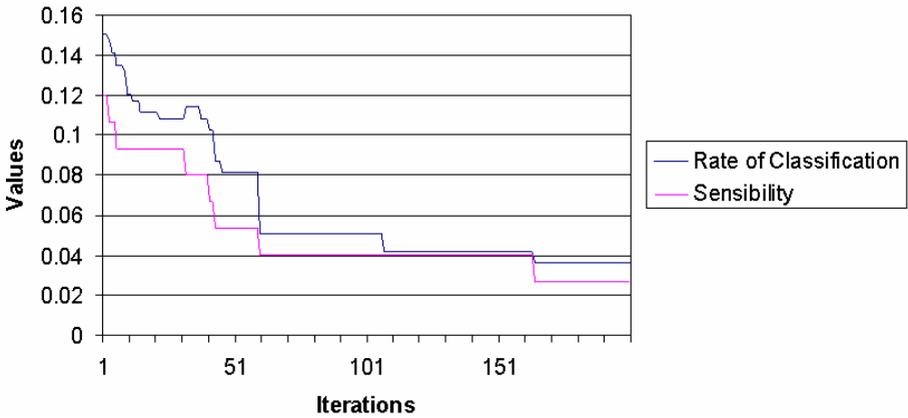


Fig. 3. Behavior of sensibility and rate of classification (f_1 and f_5) during the first 200 iterations of the training process in experiment 1

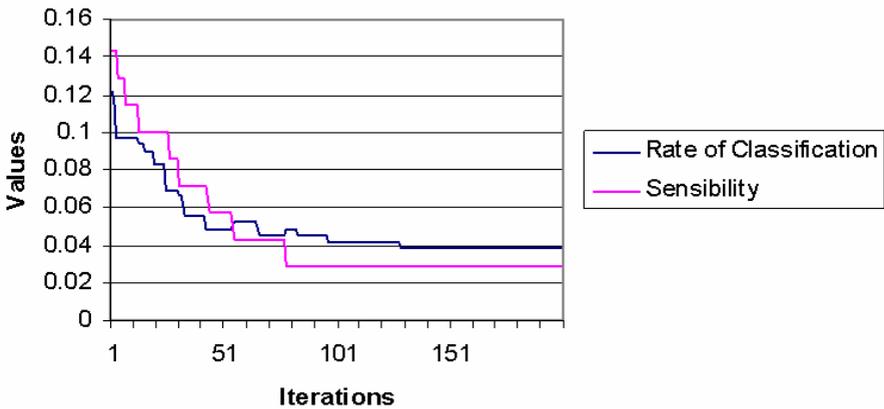


Fig. 4. Behavior of sensibility and rate of classification (f_1 and f_5) during the first 200 iterations of the training process in experiment 2

Table 1 presents the results reached by the two classifiers, based on MLP, built by means of the experiments 1 and 2, respectively. Its selection was motivated from its behavior in all indexes under optimization, with regard to the rest of the solutions corresponding to the set of best individuals of each experiment. Every index was calculated for the training cases (TC1), the test cases (TC2) and for the complete database (TC1+TC2), in each experiment. The information on the topology of the MLP, corresponding to the opposing classifiers is also shown in Table 1.

Table 1. Indexes obtained with the CGA for experiments 1 and 2

Criteria	Experiment 1			Experiment 2		
	TC1	TC2	TC1+TC2	TC1	TC2	TC1+TC2
<i>Se (%)</i>	98.67	96.95	97.49	98.57	95.77	96.70
<i>Sp (%)</i>	98.06	95.70	97.07	100	99.27	99.72
<i>PP (%)</i>	93.67	95.21	94.71	100	99.27	99.51
<i>NP (%)</i>	99.60	97.27	98.63	99.54	95.80	98.07
<i>RC (%)</i>	98.19	96.28	97.22	99.65	97.50	98.59
Topol.	1 hidden layer, 3 neurons			1 hidden layer, 4 neurons		

The results obtained for both experiments were compared with those reported by different methods in the specialized literature, for the same databases.

This comparison is shown in the Table 2, for the experiments 1 and 2. The CGA proved to be an interesting alternative for pattern recognition applications, obtaining results comparable and sometimes better than those previously reported. The software CGANN-Trainer was very useful to generate the neural network and to evaluate its performance.

Table 2. RC (%) reported for different algorithms used to classify cases from the WBC and WDBC databases

WBC									
M1	M2	M3	M4	M 5	M6	M7	M8	M9	CGA
95.9	93.7	97.4	97.5	97.2	97.9	97.8	95.5	97.0	97.2
WDBC									
D1			D2			CGA			
97.1			98.1			98.6			

Acknowledgement

This work was partially supported by the Canadian International Development Agency Project Tier II-394-TT02-00 and by the Flemish VLIR-UOS Programme for Institutional University Co-operation (IUC).

References

1. H. Demuh, *Neural Network Design*, PWS Publishing Company, 1996.
2. K.F. Man, K.S. Tang, S. Kwong, *Genetic Algorithms: Concepts and Designs*, Springer, 1999. Chapter 7, Hierarchical Genetic Algorithm in Computational Intelligence, 155-174.
3. G. Bebis, K. Takis, Coupling Weight Elimination and Genetic Algorithms, Proc. of Int. Conf. on Neural Networks, Washington D.C., USA, 2 (1996) 1115-1121.
4. F. Menczer, R. Belew, Local Selection, *Evolutionary Programming*, VII, LNCS 1447, Berlin, Springer, (1999).
5. K.F. Man, K.S. Tang, S. Kwong, *Genetic Algorithms: Concepts and Designs*, Springer, 1999. Chapter 4, Hierarchical Genetic Algorithm, 65-74.
6. O. Lafe, "Cellular Automata Transforms" KLUWER ACADEMIC PUBLISHERS, 2000.
7. D. Whitley, "An overview of Evolutionary Algorithms: Practical Issues and Common Pitfalls" *Information and Software Technology*, 43(14):817-831, 2001.
8. A. del Toro, M. Orozco, C. A. Ferrer, M. Mendoza, A. Taboada, "A Novel Pareto-Based Fitness Assignment Method and its Evaluation in Genetic Programming", I Congreso Nacional de Reconocimiento de Patrones, Havana, 2003, ISBN 959-7056-19-4.
9. O.L. Mangasarian, R. Setiono, W.H. Wolberg, *Pattern Recognition for Linear Programming, Theory and Applications in Medical Diagnosis*, (1990) 22-30.
10. O.L. Mangasarian, W.H. Wolberg, *Cancer Diagnosis Via Linear Programming*, SIAM News, 23 (5) (1990) 1-18.
11. W.H. Wolberg, O.L. Mangasarian, Multi-Surface Method of Pattern Separation for Medical Diagnosis Applied to Breast Cytology, Proc. of the National Academy of Sciences, U.S.A., 87 (1990) 9193-9196.
12. J. Zhang, Selecting Typical Instances in Instance-Based Learning, In Proc. of the Ninth International Machine Learning Conference, Aberdeen, Scotland: Morgan Kaufmann, (1992) 470-479.
13. K.P. Bennett, O.L. Mangasarian, *Neural Network Training via Linear Programming*, Advances in Optimization and Parallel Computing, (1992) 56-67.
14. R. Setiono, Extracting Rules from Pruned Neural Networks for Breast Cancer Diagnosis, *Artificial Intelligence in Medicine*, (1996) 37-51.
15. R. Setiono, R.H. Liu, Symbolic Representation of Neural Networks, *Computer*, (1996) 71-77.
16. L. Taha, J. Ghosh, Evaluation and Ordering of Rules Extracted from Feed-Forward Networks, Proc. IEEE International Conference on Neural Networks, (1997) 221-226.
17. C.A. Peña-Reyes, M. Sipper, A Fuzzy-Genetic Approach to Breast Cancer Diagnosis, *Artificial Intelligence in Medicine*, (1999) 131-155.
18. S.C. Bagui, S. Bagui, K. Pal, N.R. Pal, Breast Cancer Detection using Rank Nearest Neighbor Classification Rules, *Pattern Recognition*, 36 (2003) 25-34.
19. N. Street, *Cancer Diagnosis and Prognosis via Linear-Programming-Based Machine Learning*, Ph.D. Dissertation, University of Wisconsin-Madison, Available as UW Mathematical Programming Technical Report 94-14, (1994).