

A Novel Multi-Objective Genetic Algorithm Approach to Artificial Neural Network Topology Optimisation: The Breast Cancer Classification Problem

V. Bevilacqua G. Mastronardi F. Menolascina P. Pannarale A. Pedone

DEE - Politecnico di Bari
Via Orabona, 4 - 70125 - Bari
Italy
bevilacqua@poliba.it

Abstract— This paper presents a novel approach to Artificial Neural Network (ANN) topology optimisation that uses Multi-Objective Genetic Algorithm in order to find the best network configuration for the Wisconsin Breast Cancer Database (WBCD) classification problem. The WBCD [1][2][3] is a publicly available database composed by 699 cases, each of which is defined by 11 parameters. The former first 10 values of each record account for geometrical features of cells extracted with FNA biopsy. The last parameter represents the nature of the tumour; two classes of tumour are considered in this database: benignant and malignant tumours. An Intelligent System, IDEST, was designed and implemented. At the core of this system there's an Artificial Neural Network that is able to classify cases. The design of such an ANN is a non trivial task and choices incoherent with the problem could lead to instability of the network. For these reasons a fixed topology Genetic Algorithm (GA) approach was used to find an optimal topology for the given problem. In a second step a Multi-Objective GA (MOGA) was developed and employed in order to refine the search in the "topology space". Results shown by the IDEST demonstrate the great potentialities of similar approaches.

I. INTRODUCTION

In the recent years the need of classification systems has become more and more pressing. The abundance of data to be analysed in fields like economy, meteorology and biomedicine has determined a considerable speed up in the research of novel data-mining techniques. In this context the interest of academic research groups has grown: great efforts were done mostly in the field of ANN and AI based classifiers. Potentialities of ANNs, in particular, have been exploited in order to find best clustering algorithm for a given problem. Similar approaches have proven to be successful in several real world problems and so they became a commonly used tool in many research fields. Among these fields the medical context is of course one of the most active. Healthcare is a priority and this makes all methodologies relating to population's health improvement an obliged research field. Biomedical databases started to appear in the '80s and proliferated during the last two decades of the XX century. Considering Moore's law, the amount of data per database and the time passed, it is easy to

imagine the dimensions of currently stored data of similar kind. Nevertheless ultimate frontiers of biotechnologies, like micro-array data analysis, provide larger and larger datasets. For all these reasons several computational intelligence based algorithms have been employed in this specific field. Intelligent systems based on Artificial Neural Networks are common in biomedical predictors but unfortunately the design of similar systems is not a trivial task. The main problem in the development of an Expert System based on ANN is, of course, in the design of the neural network. The choice of a particular topology is a quite important task and should be accomplished taking into account a complex set of variables. If incoherent choices are made in the design phase, the obtained result could consist in a system with limited performance or even in an instable system. In general the focus is on the optimisation of learning/generalisation capability rate of the network. This problem can obviously be approached as an optimisation task and so GAs can be used in order to find an optimal topology for a given classification problem. In this paper we address the need of finding similar solutions with a MOGA which will be described in the next sections. We employed and validated the results of this solution using IDEST[4]. IDEST is an Integrated Diagnosis Expert SysTem that classifies data from the WBCD. Its core is based on an ANN which was previously optimised using a fixed topology GA. In the next paragraphs we underline main differences between these two approaches giving some cues of results achieved. Performances and characteristics of the systems developed in these ways are given in order to highlight potentialities of similar approaches to the problem of ANN topology optimisation. A detailed analysis of the problem of cancer classification and some hints at modern data acquisition and analysis techniques are given in "The Breast Cancer Diagnosis problem: IDEST". In "Genetic Algorithm in ANN Design" is explained how Genetic Algorithms have been applied in the ANN topology optimisation. In the following paragraph "MOGA Approach in Topology Optimisation" more detailed information are given about the MOGA approach to the problem. "System Training and Validation" and "Results comparisons" paragraphs report relevant information about the Expert Systems set up on the basis of fixed topology GA and MOGA. Finally conclusions and further cues of research are collected in "Concluding remarks and further works" paragraph.

II. THE BREAST CANCER DIAGNOSIS PROBLEM: IDEST EXPERIENCE

The Breast Cancer (BC) is the first cause of death in women living in the developed countries. Early diagnosis of this kind of tumour can save lives or improve life quality of patients. The diagnosis step then becomes a critical task in BC treatment. For these reasons several databases have been collected in order to improve the quality of diagnostic processes. Data mining techniques have been applied to similar sets and soft computing techniques were used in order to extract rules and information from these datasets. In [4] we described IDEST experience; with this system we were able to correctly classify tumour cases belonging to WBCD dataset. The WBCD is fundamentally based on the flattening principle. Composed by 699 cases, each defined by 11 fields, this dataset collects breast cancer cases observed by W.H. Wolberg in the late '80s [5]. Sixteen cases lack of one parameter. Database entries are characterised by the following structure:

(*ID, Parameters, Diagnosis*)

where ID is the primary key, parameters fields contain numerical values associated to the following 10 indicators:

- Radius;
- Area;
- Perimeter;
- Texture;
- Smoothness;
- Compactness;
- Concavity;
- Concave points;
- Symmetry;
- Fractal dimensions;

the last field in the entries contains medical diagnosis associated to the cases, it is a binary value representing malignant/benign tumour. The former first ten indicators are extracted analysing images obtained through Fine Needle Aspiration (FNA), a fast and easily repeatable breast biopsy exhaustively described in [6]. Microscope examination of the small drop of viscous fluid aspirated returns images similar to the one in figure 1.

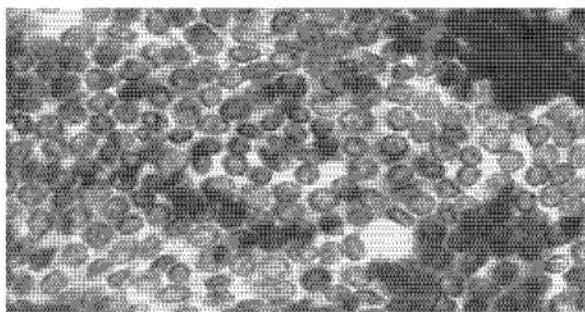


Figure 1. Microscope analysis of the image returned by the FNA.

An image segmentation algorithm looks for cells' nuclei and allows tracing contour lines, called "snakes", around cells. The resulting image would be similar to figure 2.

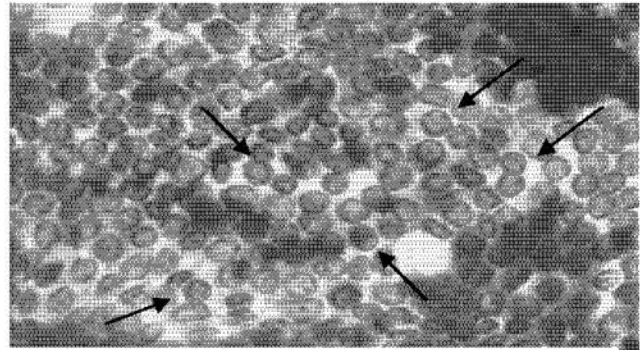


Figure 2. Cells and nuclei detected by image analysis algorithm

Main features are extracted from this image. An algorithm calculates Mean, Standard Error and Worst value for each of the ten parameters. Only relevant parameters are recorded [6] and through a simple scaling values are brought in the [0,10] range. Diagnosis values are obtained through cytological analysis of surgical biopsy and histological confirmation [6].

The investigation of data structure consists in the exploration of correlation levels among variables and between single parameters and the diagnosis value. Maintaining information consistency in data analysis is a priority in this context. Proposed data analysis method was engineered considering all these observations and trying to reach an optimal equilibrium between the amount of information retrieved and their coherence with the context.

In order to explore the data space defined by the 683 10-tuple linear and non linear analysis techniques were employed. Statistical tools like PCA and PFA were employed in order to analyse variance distribution among cases. A 20x20 Self Organising Map was set up and trained in order to observe classes returned by an unsupervised classification. The following step provided for the setup and training of an ANN based on the results of previous analyses. However, it is commonly known that the design of an ANN is not a trivial task and that mistakes in this phase could lead to incoherent results. For this reason we firstly decided to employ a *fixed topology* GA in order to optimise network topology. We will cover this aspect and related topics in the next paragraph.

Having obtained a valuable solution we proceeded to apply results to system the setup and training. A {25,12,1} ANN topology has been tested in order to acquire information about the accuracy of the observations. Competitive results were immediately reached by this system. The precision level registered in this phase seemed to be quite good and already comparable with state-of-the-art system described in literature [7][8][9][10].

III. GENETIC ALGORITHM IN ANN DESIGN

Decisions made in the network designing phase could turn out to be critical. Choices non coherent with the problem could influence negatively learning or generalisation ability of the Intelligent System.

In this field evolutionary techniques have proven to be a great support in exploring the complex spaces that characterise the designing process. The setup of a neural network can be thought of as an optimisation problem, indeed.

The employment of such techniques appeared to be the optimal method in order to find a competitive solution. In a previous work [4] a *fixed topology* mono-objective GA was set up: its chromosome codified all those characteristics of the net that have a high impact on the precision level of the system. A binary chromosome was then assembled considering the following constraints:

Maximum number of hidden layers: 4;
Maximum number of neurons per layer: 32, 16, 16, 16, 8;
Number of activation functions: 3 (pure linear, tansig, logsig).

The actual maximum number of layers is 6 (I/O layers not included). Such a contained number of layers excludes *a priori* too complex solutions (limiting the space of search) and maintains fixed and limited the number of bits needed for codification. The maximum number of neurons in each layer has been chosen so that layers near to inputs could potentially have a large number of neurons; neurons number limit decreases till the last layer in which a unique neuron is allowed. For each of the 6 potential layers a 2 bit long field is reserved in the chromosome. Total chromosome length is, hence, 34 bit. Operators employed in the GA set up were:

Crossover, with a probability of 30%;

Mutation, with probability of 5% (dependent on generation number, and active for inter-chromosomal variance smaller than 0.02);

Selection system: elitism;

$$\text{Fitness function} = \frac{1}{SSE^2}$$

In less than 25000 generations the GA returned a solution that proved to be the “optimum sub condicione” (considering the limitations to the research space imposed); after a few generations, individuals showed very low inter-chromosomal variance. Decoding phase carried out on the chromosome which showed the best fitness function returned the following solution to the topology optimisation problem:

Number of hidden layers: 1;
Number of neurons per layer: 25, 12, 1;
Activation functions: logsig, logsig, purelin;

The result highlighted by GA runs appears to be coherent with the complexity of the problem: the relatively high number of neurons in the first layer is supported by the dimensions of data. The number of hidden layers is congruent with expectations. Details, like pure linear activation function of the last layer result consistent with documentation in literature [11][12][13].

Although the results achieved were quite satisfactory, a multi-objective GA approach was used in order to optimise the search for the optimal topology. Details about this approach are given in the following paragraph.

IV. MOGA IN TOPOLOGY OPTIMISATION

The results obtained with the first implementation of IDEST demonstrated to be already competitive. However, given the dimensions of the ANN returned by the *fixed topology* approach, the system training showed computational intensive behaviours. For this reason the possibility of further optimisations were explored. A Multi-Objective GA approach was selected in order to provide necessary optimisations [17][18]. The MOGA engineered follows a Pareto-Based Approach [19][20][21]. Several experiences of application of EAs to engineering problems [22] are reported in literature, the most representative approaches to these kinds of problems have been analysed [30]. The strength points of Vector Evaluated Genetic Algorithm (VEGA)[23], Weighting-based Genetic Algorithm (HLGA)[24], Multi-Objective Genetic Algorithm (MOGA)[25], Niched Pareto Genetic Algorithm (NPGA) [26], Non-dominated Sorting Genetic Algorithm (NSGA)[27] and Strength Pareto Evolutionary Algorithm (SPEA) [28] solutions were evaluated; considerations about the possible impacts on system performances were made. Taking into account the potentialities of all these approaches a novel MOGA has been developed. In this solution information relating to a single layer is concentrated in the genotype following spatial-neighbourhood criterions. This choice reflects principles of the “*Schema Theory*”. The chromosome is encoded in 38 bits. Reflecting the biological paradigm, chromosomes are grouped in genes and sub-genes. On the basis of similar characteristics the implementation proposed can be said to belong to the “structured GA” class of evolutionary algorithms. In addition, *over-specification* has been adopted. As demonstrated by Holland the binary codification employed can lead to better results [29]. Through the decoding of the genotype it is possible to obtain optimal or sub-optimal solution to the neural network topology optimisation problem. The number of neurons per

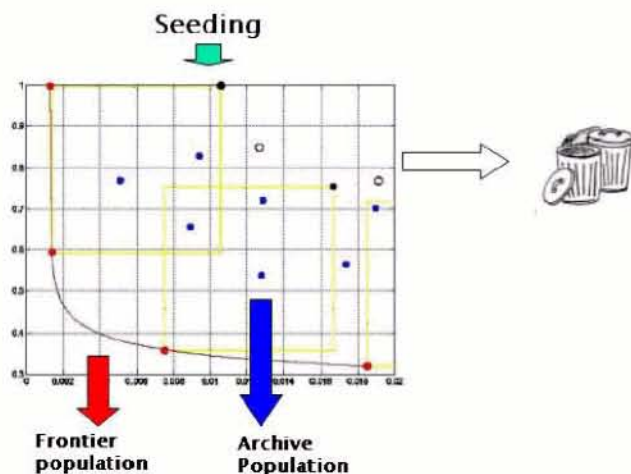


Figure 4. On the Pareto curve the “Frontier” individuals, “Archive” individuals in the rectangles. Other solutions are excluded from further evolution. On the x/y axis the values of the fitness functions

layer, activation function for each layer and the learning rate are all encoded in the chromosome (see figure 3, App. A). The mutation operators employed are 4 and are applied sequentially, they act on groups of semantically similar bits; each operator has a characteristic probability of employment. The crossover operator is a simple uniform single point crossover operator. The main goals of all these choices are the *precision* and *generalisation* shown by the ANN defined by each genotype, after the training phase. The training of ANNs was accomplished using the Error Back Propagation (EBP); a brief overview of the EBP implementation is given in the next paragraph.

The main principles at the basis of the algorithm provided for a first step in which a population is initialised through partial-seeding, fitness values for these first individuals are then computed. Pareto curve is drawn and individuals standing on this curve are included in a population called “Frontier” (see figure 4). On the “Pareto individuals” rectangles of predefined dimensions are built. The dimensions of these figures are determined considering the “Distances”, individuals in the rectangles are included in the “Archive”, the others are eliminated (see figure 5).

The iterations of the algorithm then start; for each step, two individuals from the “Frontier” are selected and included in a population referred to as “Laboratory”. Two individuals from the “Archive” are picked up and the crossover operator is applied. The resulting offspring constitute the third and the fourth individuals in the “Laboratory” population. These two genotypes are mutated and the resulting individuals are placed in the last two positions of the “Laboratory” population. Fitness function evaluation for each individual are accomplished training ANN based on the decoded genotypes. Stop criterions for the training of the ANNs are set in terms of epochs and SSE. Considering the “Frontier” and “Archive” individuals of the i th step, “Frontier” and

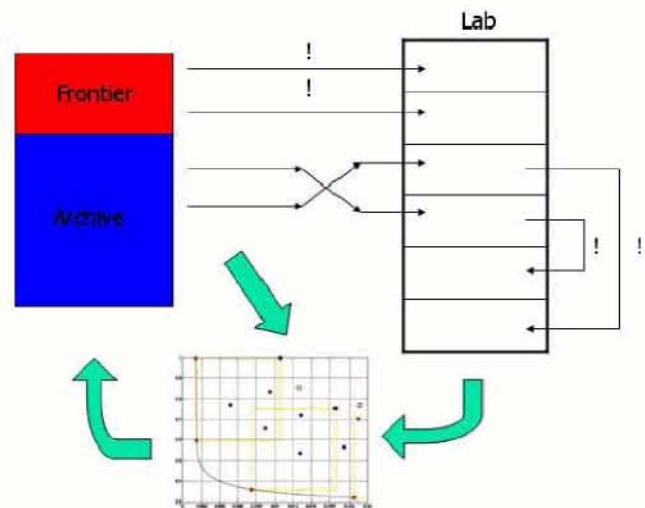


Figure 5. Frontier/Archive/Laboratory populations dynamics.

“Archive” individuals of the $i+1$ step are obtained.

It is worth noting that “Distances” and the maximum number of training epochs constitute two important aspects of this particular implementation. These values are periodically updated. Epochs are increased and “Distances” are decreased. Periodically these two variables are set to a minimum value for the epochs and a maximum for the “Distances”. This choice allows for an acceptable compromise between exploration and exploitation. The “Distances” are computed considering the average value of the fitness functions in the population; training epochs are linearly increased so that all the individuals characterised by one or more objective values above the average threshold (calculated over all the individuals and multiplied by a proper coefficient) are eliminated. The set up of this coefficient and of the maximum predetermined values the distances are cyclically re-set to, allow to control at run-time the dimension of the population.

When the number of epochs is increased, the training of “Frontier” and “Archive” individuals that didn’t reach a precision threshold is continued restarting from the weights previously saved. In a first stage an approximated fitness function is employed in order to speed up the search, this can be obtained thanks to the high robustness of GA to the noise. The training of individuals that don’t show, in a certain number of epochs, a precision level above a threshold is stopped; these individuals, in fact, will be unlikely to converge and then computational times are saved, even given the highly computational intensive behaviour of the EBP algorithm. The results returned by the described approach provided for a {13,8,1} ANN topology with 0.4 learning rate.

It is evident that, just like in NSGA, individuals closest to the Pareto curve have higher probability of being selected for reproduction. A fixed number of individuals is selected among non dominant individuals (belonging to the “Frontier”

population, two individuals are selected in this set). Furthermore “Distances” decrease with the evolution of the generations and are cyclically re-increased. For these reasons, in the same cycle, the individuals nearest to the *frontier* have higher probability to be picked up because they remain in the population for a longer time. Then this higher probability is not generated by a direct effect on the fitness values; general dynamics of the algorithm are at the basis of this positive behaviour. The correlation with SPEA approach is then evident. In both these approaches, in fact, non dominant individuals are placed in a separated population, with all the implications this choice involves.

V. SYSTEM TRAINING AND VALIDATION

The ANN setup based on the MOGA results obtained in previous steps was trained using Back-propagation algorithm in the variation that updates weights and biases according to gradient descent momentum.

The application developed for this work is able to optimise network topologies with at most five layers (including input and output). All connections among neurons are characterised by numeric values (*weights*) that are updated during the training. In the training phase the network processes all the pairs of input-output presented by the user (502 cases in this work), learning how to associate a particular input to a specific output trying to extend the information acquired also for cases that don't belong to the *training set spectrum*. Any pair of data in the training set is presented to the system a quantity of time determined by the user *a priori*. The learning step is based on the Error Back Propagation (EBP) algorithm. The weights of the network are updated sequentially, from the output layer back to the input layer, by propagating an error signal backward along the neural connections (hence the name “back-propagation”) according to the gradient-descent learning rule:

$$\Delta\omega_{ij} = -\eta \frac{\partial E}{\partial W_{ij}} \quad 0 \leq \eta \leq 1 \quad (1)$$

where η is the *learning rate*, E is the error and ω is the weight associated to the link that connects the neuron i with the neuron j . Any neuron processes the sum of the signals received from the other linked neurons, using one of the following activation functions: *logsig*, *tansig*, *purelin*. The error is calculated at the end of an epoch, when all elements of the training set have been processed, and computed as Sum Squared Error (*SSE*):

$$SSE = \frac{1}{2} \sum_j (Y_j - D_j)^2 \quad (2)$$

SSE is employed also in order to estimate the error in the *validation set*. The *validation set* is a dataset used to test the network after the training; it accounts for 181 cases. For the

initialisation of the weights the application allows three choices for any layer: *file*, *random*, *Nguyen-Widrow*. The “file” option loads layer weights from a file, “random” generates random values in the range $(-1,1)$, “Nguyen-Widrow” initialises the weights following the algorithm of Nguyen-Widrow[14][16]. This is an efficient initialisation method commonly used to significantly speed up the convergence phase, distributing even over the input space the weights values. Finally, in order to modify weights, we introduced a term called *momentum*. The basic idea using momentum is to ignore small features in the error surface, without momentum a network may stop at a local minimum. The momentum considers previous adjustment to modify weights. The equation used to update the weights is the following:

$$W_{ij}(t+1) = W_{ij}(t) + \Delta W_{ij}(t) + \beta \cdot \Delta W_{ij}(t-1) \quad (3)$$

$$0 \leq \beta \leq 1$$

where β is the momentum.

The starting learning rate chosen was 0.3. This choice avoided the occurrence of step-back phenomena in learning phase and it gave the network sufficient energy to exit from “local minima”: all this resulted in ANN's good aptitude for convergence. Stop criterion was set to SSE equal to 7×10^{-3} or to the limit of 50000 training epochs.

As shown in figure 6, the system was able to complete the training phase in about 13000 epochs reaching, therefore, the SSE target. The relatively contained number of epochs needed to accomplish the training step confirms the correctness of results obtained via linear and non linear analysis and, in particular, the accuracy of GA search.

System validation was carried out submitting to the network the 228 cases of the validation set and calculating the misclassified ones. The results returned by this analysis proved to be quite good: no misleading prediction was made on the 228 cases analysed.

The highest obtainable system accuracy was then reached: it is evidently an indicative result but potentialities of similar systems seem, now, to be supported by more concrete elements. In an *a posteriori* analysis we have tried to estimate the impact of most significant choices on the accuracy of ANN's predictions. In this phase we observed how particular decisions contributed to the achievement of such a result.

Leaving unchanged the phases of process described until now but employing training and validation sets assembled ignoring results of linear and non linear analysis (i.e. using sets obtained simply dividing the original dataset in two

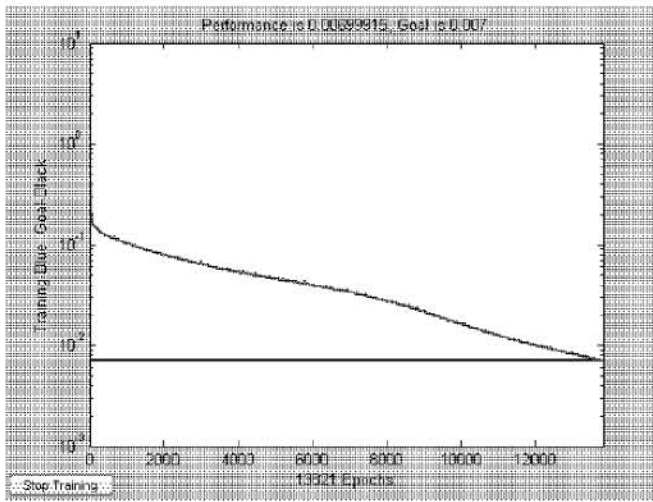


Figure 6: SSE graph. On the x axis training epochs, on the y axis one the SSE.

subsets), it is possible to observe an error on the validation set equal to 4 cases on the 228. This corresponds to an accuracy of 98.6%, a competitive value indeed, which shows, the importance of observation mostly on intrinsic variance of cases.

Another interesting observation can be made leaving unchanged the process described but eliminating the hybrid ranking inspired by elitist method typical of “evolutionary programming” contexts. Suppressing this step we can incur an error, on the validation set, approximately equal to 0.9—0.4%.

Results obtained highlight the contribution to the accuracy of learning that choices in data pre-processing and GA phases have generated. Adopted devices allowed obtaining a system characterised by a more compact architecture and capable of taking full advantage of peculiar characteristics of the datasets and of the distribution of information in it.

VI. RESULTS COMPARISONS

Vast documentation of experiences relating to diagnostic system based on WBCD is collected in literature. Starting from 1991 many teams tried one’s hand at developing Intelligent Systems able to correctly classify breast cancer cases.

The series of similar projects was started in 1991 by *Xcyl*: this was the name that O. Mangasarian, N. Street and W. Wolberg gave to their diagnostic system. *Xcyl* was fundamentally based on linear programming techniques through which it tried to find one or more hyper planes able to correctly separate cases belonging to different classes. Although it is interesting, performances registered by *Xcyl* proved to be improvable (Tab. 1); unfortunately the use of more hyper planes did not bring the expected amelioration and instead generated over-fitting-like phenomena, as shown reported in [3].

Other authors have reported experiences of hybrid systems: in [7] Reyes and Sipper describe a Fuzzy Co-evolutionary Cooperative system, i.e. an approach to fuzzy modelling in

which two coevolving species are defined: a database (membership functions) and a set of rules.

Similar approach, but based on the framework defined by Potter and De Jong [15], registered quite competitive results. Reyes and Sipper’s research then, focused on the extraction of rules from the trained ANN. This was an objective in common with Taha and Ghosh [8] who designed a Neuro-Fuzzy system able to extract rules to be employed in genetic-fuzzy inference systems.

Results of each of these projects are collected in Table 2.

	Correctness Training Set	Correctness Test Set
Visually Extracted Features	100%	91%
Digitally Extracted Features	95%	90%

Table 1. Correctness levels registered by *Xcyl* using 1e Leave One Out validation method

Larger set of rules allows to achieve better results than those characterising system with fewer rules; for this reason for each project best results were indicated (the note field contains information about the kind of system the result is related to).

Even if some teams focused on priorities slightly different from the development of an accurate system (e.g. the extraction of rules from the ANN in Setiono and Liu [9] and Taha and Ghosh [8]), it is evident that the availability of an accurate ANN is a necessary condition for the retrieving of precise rules for the inference systems.

Systems here considered have proven to be globally competitive: however, in the performance comparison, IDEST, even if characterised by more compact architecture, is able to reach the highest result obtainable.

	Mangasarian, Wolberg et Street [1]	Setiono et Liu [9]	Taha et Ghosh [8]	Pena et Reyes [7]	This work
Best results	91%*	97.21%*	96.19%*	97.80%*	100%
Notes	*3 hyper planes	*3 rules per system	*5 rules per system	*3 rules per system	

Table 2: System accuracy comparison. In “Notes” the characteristics of system which showed the best behaviour

VII. CONCLUDING REMARKS

In this paper we have presented a novel approach to ANN topology optimisation using evolutionary algorithms. A Multi-Objective Genetic Algorithm was developed in order to optimise the architecture of the ANN which IDEST is based on. The proposed solution proved to be able to reach a good level of optimisation pruning the original architecture ({25,12,1}, returned by the *fixed topology optimisation*

approach), to a {13,8,1} solution, leaving unchanged the accuracy level of the system. We would like to focus on the peculiar aspects of the developed MOGA. In particular, it could be argued that the employment of "Frontier", "Laboratory" and "Archive" *populations* can reveal good potentialities in terms of convergence rate. The capability of the proposed EA to extract optimal solutions could even be improved by this particular aspect. In general terms, it could be said that the obtained system show interesting results.

The use of data structure research system based on linear and non linear analysis techniques and data filtering strategies allowed acquiring a set of information that permitted to bring out all the potentialities of ANN. These phases could prove to be a good starting point for any system architecture optimisation. In this way, a system characterised by quite linear data flows is able to reach good results, if supported by an adequate data pre-processing phase.

Relevant aspect of this project is the application of a novel MOGA to the development and optimisation of critical systems, like the diagnostic ones. Topological optimisation determined by the proposed MOGA allowed obtaining a stable system able to set an optimal compromise between training time and learning/generalisation capabilities.

The first priority in IDEST development, the precision in classification, can be said to be reached. IDEST shows competitive performance even in those cases where the diagnosis formulation of a human expert could become a problematic task.

The principal objective was reached: the system show necessary characteristics to be tested in real world. However, in the future, even though IDEST seems to be a good tool, a prognostic module, able to estimate the Disease Free Survival, would be a valid complement to the engineered system. In this case, intrinsic characteristics of prognostic process, could impose a careful estimate of tools to be used: it is common knowledge those cases characterised by similar biomedical indicators can evolve in substantially different manners. A careful consideration, then, appears to be needed on which information has to be used in order to train the Intelligent System. An interesting support in this problem could be represented by the new frontiers in biotechnology: gene expression analysis through micro-arrays can help in the investigation of the main causes of development of the examined tumours. The research in the field is in progress and the employment of Intelligent Systems in this context could become a fundamental factor for the analysis of genetic data and of information in them. The study of gene expression levels, furthermore, could be used as base for the engineering of a parallel diagnostic system able to support the one described in this paper. The support of genetic information could constitute further confirmation of ANN classification. Micro-array data analysis, in addition, could allow gaining more information about correlation between genotype and phenomenology of tumours with easily conceivable effects on the preventive process.

Information retrieved integrating these systems could be, in a second stage, merged in a Bayesian network which can

further improve the abilities of the decision-making support system.

Undoubtedly the maximum advantage for this kind of systems would derive from a continuous training; abandoning the "stand-alone" paradigm we could think to a telemedicine system able to interconnect more diagnostic centres that continuously submit cases to the network. Similar model, however, could force to modify the architecture of ANN: convergence control and, mainly, the necessity to frequently alternate training phases to operative ones proved to be quite incompatible with the ties on the ANN topology imposed previously.

From the computational point of view the research could move in the direction of investigation of performance impact of some software optimisation like SIMD SSE2-SSE3 instructions.

The employment of IDEST in distributed systems can give an estimate of the possibility of compressing computational time related to the execution of generated code (e.g. great benefits could be registered in image analysis systems based on GA). The exploitation of most recent processor architecture like x86-64 and EM64T, in single or dual core configuration could represent a valid support to the study of possible code optimisation.

REFERENCES

- [1] O.L. Mangasarian, W.N. Street et W.H. Wolberg, Breast cancer diagnosis and prognosis via linear programming. *Operations Research*, 43(4), pages 570-577, July-August 1995.
- [2] O.L. Mangasarian, W.N. Street, Breast-cancer-wisconsin.names.
- [3] W.H. Wolberg, W.N. Street and O.L. Mangasarian, Image analysis and machine learning applied to breast cancer diagnosis and prognosis. *Analytical and Quantitative Cytology and Histology*, Vol. 17 No. 2, pages 77-87, April 1995.
- [4] Bevilacqua, V., Mastronardi, G., Menolascina, F., *Hybrid data analysis methods and Artificial Neural Network design breast cancer diagnosis: IDEST experience, CIMCA 2005*.
- [5] W.H. Wolberg, W.N. Street, et O.L. Mangasarian, Breast cytology diagnosis via digital image analysis. *Analytical and Quantitative Cytology and Histology*, Vol. 15 No. 6, pages 396-404, December 1993.
- [6] Wolberg WH, Street WN, Heisey DM, Mangasarian OL, Computer-derived nuclear features distinguish malignant from benign breast cytology. *Cancer Cytopathology* 81:172-179, 1997.
- [7] Pena Reyes, Sipper, A Fuzzy-Genetic approach to breast cancer diagnosis, (1999).
- [8] Taha, I. et Ghosh, J. Evolvable fuzzy rules for breast cancer diagnosis, in proceedings of NOLTA'98, Lousanne, vol. 2 pp369-372., (1998).
- [9] Setiono, R. et Liu, H., Symbolic representation of neural networks, *Computer*, vol. 29, no. 3, pp. 71-77, (1996)
- [10] Carpenter, G. et Grossberg, S., The ART of Adaptive Pattern Recognition by a Self-Organising Neural Network, *Computer*, v.21 n.3, p.77-88, March (1988).
- [11] Shortliffe, E.H., *Computer Based Medical Consultations: WCIN*. New York: Elsevier, 1976.
- [12] Brause, R.W. (2001). Medical Analysis and Diagnosis by Neural networks. Computer Science Department, Frankfurt a.M., Germany.
- [13] Yao, X. and Liu, Y. (1999), Neural Network for Breast Cancer Diagnosis. Birmingham, UK.
- [14] A.Pavelka and A. Procházka. Algorithms for Initialization of neural networks weights.
- [15] Potter, M.A. et De Jong, K.A. The Design and Analysis of a Computational Model of Cooperative Co-evolution. PhD thesis, George Mason University, (1997).
- [16] Nguyen, D. Widrow B. Improving The Learning Speed of Two Layer

Neural Networks by Choosing Initial Values of Adaptive Weights, Proc. IJCNN'90, vol. 3, pp21-26, 1990

[17] Fonseca, C. M. and Fleming, P. J., 1995, Multiobjective Genetic Algorithms Made Easy: Selection, Sharing and Mating Restriction, Proceedings of GALEZIA'95, 45-52.

[18] Yaochu Jin, Tatsuya Okabe, Bernhard Sendhoff. Neural Networks Regularisation and Ensembling Using Multi-objective Evolutionary Algorithms.

[19] Jeffrey Horn, Nicholas Nafpliotis. Multiobjective Optimization Using the Niche Pareto Genetic Algorithm. (1993)

[20] Eckart Zitzler. Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications (1999)

[21] Joshua Knowles. The Pareto Archived Evolution Strategy : A New Baseline Algorithm for Pareto Multiobjective Optimisation. Proceedings of the Congress on Evolutionary Computation (1999)

[22] Dipankar Dasgupta and Zbigniew Michalewicz, Evolutionary Algorithms in Engineering Applications, Springer-Verlag (1997)

[23] J.D. Schaeffer. Multiple objective optimisation with vector evaluated genetic algorithms. Proceedings of the First International Conference on Genetic Algorithms and Their Applications. Lawrence Erlbaum Associates, Publishers, pp. 160-168 (1985)

[24] P. Hajela and C.-Y. Lin. "Genetic search strategies in multicriterion optimal design". Structural Optimization, 4, pp.99-107, 1992.

[25] Fonseca, C. M. and Fleming, P. J., 1995, Multiobjective Genetic Algorithms Made Easy: Selection, Sharing and Mating Restriction, Proceedings of GALEZIA'95, 45-52.

[26] Jeffrey Horn, Nicholas Nafpliotis, David E. Goldberg. A Niche Pareto Genetic Algorithm for Multiobjective Optimization. Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence

[27] N. Srinivas and K. Deb. "Multi-Objective function optimization using non-dominated sorting genetic algorithms". Evolutionary Computation, Vol. 2, pp. 221-248, 1995.

[28] E. Zitzler, K. Deb and L. Thiele. "Comparison of multiobjective evolutionary algorithms: Empirical results". Evolutionary Computation, Vol. 8, pp. 173-195, 2000.

[29] John Holland, "Genetic Algorithms", Scientific American, July 1992

[30] V. Maniezzo, "Genetic evolution of the topology and weight distribution of neural networks," IEEE Trans. Neural Networks, vol. 5, pp. 39-53, 1994.

APPENDIX A

1st Layer	2nd Layer			3rd Layer			4th Layer	Learning rate
0	8	9	10	18	19	20	28	30
XXXXXXXX	x	x	XXXXXXXX	x	x	XXXXXXXX	xx	XXXXXXXX
Neurons (1-256)	I/0	logsig	Neurons (1-256)	I/0	logsig	Neurons (1-256)	logsig/tansig/ purelin/-	Variable
		tansig			tansig			

Figure 3. Chromosome scheme