Blind Signal Separation through Cooperating ANNs

F. Bellas, R. J. Duro, F. López-Peña

Grupo Integrado de Ingeniería
Universidade da Coruña
fran@udc.es, richard@udc.es, flop@udc.es

Abstract. This paper is devoted to proposing and testing a strategy for decomposing compound signals obtained in remote sensing applications through the automatic generation of cooperating ANNs that model it. Each ANN will specialize in one of the primitives that make up the whole. The evolutionary based algorithm that is proposed for this purpose implies that the combination of networks takes place at a phenotypic operational level, this is, the architecture of the networks is not the issue, but rather function they implement. This way, a population of networks that are automatically classified into different species depending on the performance of their phenotype, and individuals of each species cooperate forming a group to obtain a complex output, in this case the signal that is required. The magnitude that reflects the difference between ANNs is their affinity vector, which must be automatically created and modified depending on the actuation of the phenotype of each individual. The main objective of this approach is to model complex functions such as multidimensional signals, which are typical of remote sensing application, providing a decomposition of them into primitive functions.

1 Introduction

Blind Signal Separation (BSS), that is, separating independent sources from mixed observed data is a fundamental problem that is difficult and challenging. It is an intrinsic problem present in most measurement systems and it becomes even more pronounced in remote sensing as there is a lot less control on the mixtures present in the signals acquired. In many practical situations, observations are usually modelled as linear mixtures of a number of source signals, ie. a linear multi-input multi-output system. However, reality is not usually that simple and more ambitious approaches are necessary.

Applications of BSS include fields such as telecommunication systems, sonar and radar systems, acoustics, and any other where multiple signal sources may be present. In the case of its application to remote sensing the main objective is usually to discriminate mixtures in the subpixel range [1][2]. Most authors have resorted to linear models and techniques such as Independent Component Analysis (ICA) whether in the time or in the frequency domain [3][4][5]. Others have considered neural based techniques with some success, especially in linear cases. These methods include the Hebbian learning algorithm [6] or robust adaptive algorithms [7]. In fact, the approaches even seek non linear discriminations, as is the case of nonlinear principal component analysis [8], or Recurrent Neural Networks as employed by Amari et al [9].

In this paper we propose an approach whose emphasis lies on the simultaneous modelling of the underlying signals that make up the sensed whole. A very powerful technique for the instantiation of the models when the learning process consists in modelling from input-output pairs is the use of artificial neural networks (ANNs). In the case of evolutionary learning, the models adapt to the task through evolution. Thus, the modularity of solutions can be obtained creating complex ANNs as a composition of simpler ones. In this line, a very common approach in the literature [10], [11], [12] is to apply an evolutionary algorithm over two different populations, one made up of simple neuronal units and the other of patterns that indicate how those neural units must be combined. These approaches have been successful, for example, in classification problems.

A different approach is proposed here, where the basic structures participating in the process are complete networks evolving in the same population and they are combined at the phenotypic level into groups that collaborate in the solution of the problem. The work of Xin Yao and Paul Darwen [13] must be mentioned here. They propose an evolutionary learning approach to design modular systems automatically. To do this, they use the concept of speciation introduced by Goldberg [14] through a fitness sharing technique. The main novelty of this work lies in the fact that within each species they use coevolutionary techniques where the concept of just one individual solution does not exist. The authors apply this technique to the prisioner's dilemma and the algorithm works successfully.

This kind of modular approaches imply the existence of collaboration between individuals of the population to achieve a successful result. This is called in the literature symbiotic evolution and is a background problem in multiobjective optimization [15]. In this field, the development of fitness sharing techniques, where the fitness of a given individual is scaled through some similitude measure with respect to the other individuals, is common. These approaches start from the knowledge of the Pareto front to be obtained, this is, they start from the knowledge of the desired objectives and search for individuals specialized in these objectives and that collaborate towards their achievement.

The problem we try to solve in this work is quite different; because we look for an automatic decomposition of the model into simple and basic networks (primitives) that combined provide the desired solution. No knowledge about how to do it (we don't know the best decomposition in primitives) is assumed. Each one of these simple networks is functionally different and, consequently, they belong to different species that evolve together.

Another algorithm that is based on coevolution and species formation is COVNET [16], where authors use a subnetwork population (nodules) that is divided into species that evolve independently and another network population that combine these nodules. The main difference with our approach lies at the species formation level. The number of species in COVNET is prefixed, whereas in our algorithm it arises automatically from the evolution itself.

In the next section we will introduce the main concepts used in this algorithm such as groups and affinity. The remaining sections provide an overview of the operation.

2 Main concepts of the algorithm

The main objective of this algorithm is very simple: to develop model a complex function using ANNs. Because of this complexity, we assume it will be simpler to find the solution using more than one ANN. So our approach is based on providing a search algorithm with the capability of obtaining the solution by aggregation of simple ANNs.

Thus, the first important feature of our algorithm is that the solution is provided by a combination of ANNs (*a group*) that combine their outputs to provide a complete solution to the problem.

To conform the groups, we have developed a selection criteria based on a magnitude called *affinity* that classifies the ANNs depending on their phenotype. The main idea is that initially all the networks have a random affinity and through a self organization process these affinities become different automatically. Then, the groups that provide the solution to the problem are made up by combining ANNs with high affinity to each other.

To improve the networks that made up the groups, we propose the use of an evolutionary algorithm as search technique because it is based on a population (a set) of solutions and it is an established technique for adjusting the parameters of ANNs. Furthermore, evolutionary algorithms are very adequate for this approach because of their capabilities to create specialized individuals (*species*) in the population. This way, once some species appear in the population, we can make up the groups that conform the solutions using individuals from these species.

3 Details of the algorithm

As mentioned in the previous section, individuals will be evaluated (fitness calculation) as a part of a group and not individually. Evolution, on the other hand, takes place over the individuals as usual and the groups are formed in the fitness calculation stage of the algorithm.

The operation is simple and can be summarized in the following 6 steps:

1. Initial creation of a random population of ANNs, as usual in general evolutionary algorithms.
2. Initial creation of a random affinity value (represented by a vector in general) associated to each individual.
3. Fitness calculation of all the ANNs in population. To do this, we apply 5 basic steps:
    a. Each individual in the population creates its own group by choosing other individuals from a window of the population using its and their affinity values.
    b. Each group calculates a fitness value, resulting from the common application of its individuals to the modelling problem.
    c. Each individual creates a new group and the fitness in these new groups is calculated. The best groups remain.

   d. The fitness value obtained in step c) permits adjusting the affinity of the individuals depending on the increase or decrease of fitness.
   e. Steps c) and d) are repeated a given number of iterations. At the end, we assign each individual the fitness of its own group.
4. Selection over the population of ANNs (like in an ordinary evolutionary algorithm) including the affinity value of each individual as a part of the selection criteria favoring reproduction of individuals with similar functional characteristics (same species).
5. Crossover and mutation as usual but including the affinity vector that is transmitted from parents to offspring.
6. Steps 3, 4 and 5 are repeated a given number of generations.

As we can see, this is the operation schema of a simple evolutionary algorithm where the fitness calculation is more complex and where the reproduction is directed towards the formation of species by mixing operationally similar individuals.

Once a group is formed, its fitness is calculated in general as the fitness provided by a set of outputs that are the result of some combination of the individual outputs of its members. In the simplest case the combination is carried out through the addition of these outputs that must be clearly marked to discriminate which ANN affects which output. The group must provide values for all the outputs of the problem, otherwise it is penalized.

After calculating the fitness of all the groups, we carry out a self organization stage (steps from a) to e) in the algorithm), which allows each individual to create more affine groups. On each step of this stage, the individuals form new groups and their fitness is calculated. If the fitness of the new group is better than that of the old one, the group is maintained; otherwise the older group is preserved. After a set number of recombination steps, each individual has formed its most affine group using the technique explained in the next section.

Affinity is used in our algorithm as a label to classify the ANNs depending on the function they perform. It is represented by a numerical vector with fixed upper and lower limits, in which the dimension determines the complexity of the groups that may be formed.

Each element in the affinity vector is represented by a real number. When an individual forms a group it searches for other individuals in the population with *complementary* affinity vectors. Initially, the affinity vectors are random as no

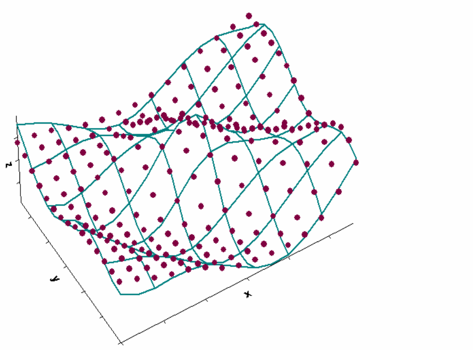

Fig 1. Modelling result (points) of a 3D function (surface) made up by the addition of 4 basic functions

knowledge of the characteristics of the individuals and/or their shortcomings is available. Every time a group is evaluated (fitness calculation), if the group improves the fitness of the previous group in which the individual participated, the distance of the affinity vector of the individual (in Euclidean terms) with respect to the affinity vectors of the rest of the individuals in the group is increased. In addition, from this instant, this will be the new group for the individual. On the other hand, if the fitness of the group is worse than that of the previous group, it is the distance with respect to the affinity of the members of the previous group that is increased. This method favours the formation of species, that is, of clusters of individuals with similar affinity vectors.

After each self organization stage, all the individuals have automatically adjusted their affinity vector depending on the function they carry out in the group.

4 Operation example

Just to display the results the mechanism can produce and the way it works, we consider here a simple problem consisting of an objective function that was constructed by adding 4 different functions (sinusoidal and Gaussian 3D) limited to a range from [-1.5, 1.5] in x and y. In this experiment, the affinity vector considered is two dimensional and the values of each dimension can vary between [-1, 1]. Thus, it seems that an optimal solution for this problem would be the generation of four species, each one specialized in modelling each one of the known functions or any other combination of functions that lead to the perfect modelling of the global function.

The networks that will be evolved in this case are standard multilayer perceptrons which have two inputs (the x and y coordinates) and one output (the z coordinate). The objective is obviously to obtain the network or combination of networks that best approximate the objective function.

The results of modelling obtained with the best group are presented in Fig. 1, where we have represented the prediction (points) and a sampling of the objective function (surface). The results are very satisfactory taking into account the complexity of the expected signal. We are going to take a closer look at what have happened with the affinity values for the individuals and what types of species arose.

Fig. 2 displays the evolution of the affinity vectors for the whole population (1000 individuals in this case). In the left graph we have represented the initial random affinities distributed from -1 to 1 in both axes. In the middle graph we show the affinity vectors in generation 2000. As we can see, the values are still not clustered, and some individuals do not belong clearly to a species, although four affinity areas are starting to be delimited. Finally, in the right graph we show the affinity vectors in generation 3600

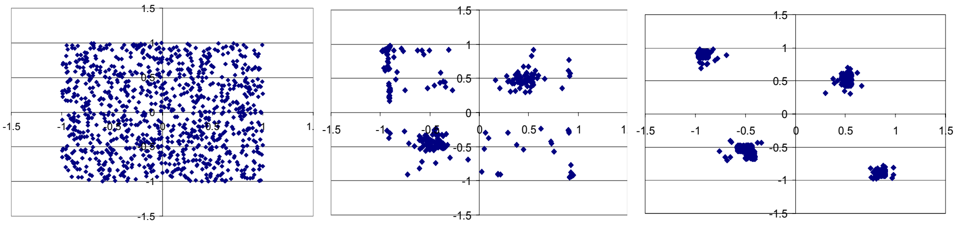

Fig. 2. Evolution of the affinity vectors for the whole population. Left graph represents the initial distribution, middle graph the distribution after 2000 generations and right graph the final distribution

(final) where 4 clear species have been formed with affinity vectors (-1, 1), (1, -1), (0.5, 0.5) and (-0.5, -0.5).

The final group taken as solution of the modelling problem (that provides prediction shown in Fig. 1) is made up of 4 individuals, one from each species. In Fig. 3 we present the functions provided by each one of these individuals in the final group. What is important here is that these primitives have been obtained automatically and combined provide a very successful prediction of the original complex function.

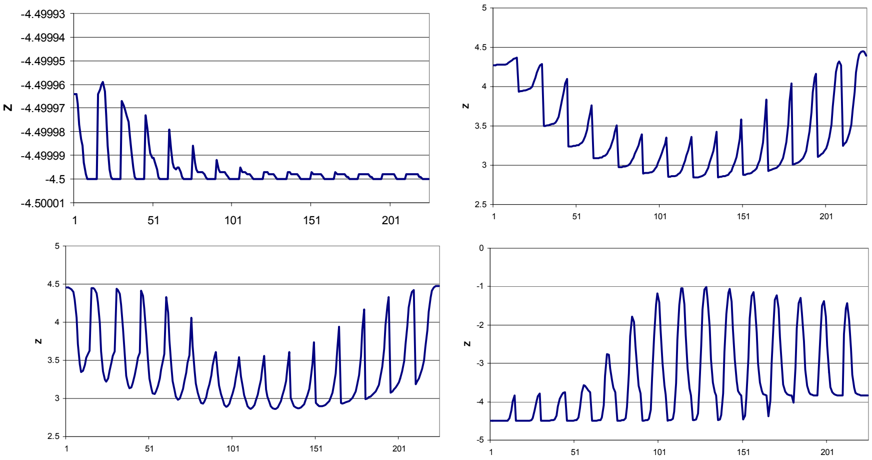

Fig. 3. Functions provided by each individual in the best group (phenotype corresponding to each specie) that combined provide the output shown in Fig 1.

Conclusions

This work deals with an affinity based strategy for obtaining groups of cooperating artificial neural networks as compound models of complex signals obtained in measuring processes, in particular in remote sensing applications. This strategy may be applied in general to any type decomposition process where the objective is to obtain a group of solution points which, through their combination and cooperation in a phenotypic level produce a modular or decomposable solution to the problem so that parts of these solutions may be later reused in other problems. The main idea behind the phenotypic affinity based strategy is that during the evaluation phase of the evolutionary algorithm the individuals can try out how fit they are through collaboration with other individuals by forming groups. The results of these trials result in modifications of each individual's affinity vector. This process leads to the formation of species of individuals specialized in different aspects or primitives of the global signal, thus allowing for the modularity we desire. The affinity of each individual also participates in the evolution stages as it is assumed that there is a certain correspondence between families or species and affinities for other species in terms of collaboration towards the common task. The results applying this strategy have been very fruitful. Here we have presented a quite complex case with the arising of four species into the population.

Acknowledgements

This work was funded by the MCYT of Spain thorugh project VEM2003-20088-C04-01 and Xunta de Galicia through project PGIDIT03TIC16601PR.

References

[1] C. I. Chang, "Target Signature-Constrained Mixed Pixel Classification for Hyperspectral Imagery" IEEE Trans On Geoscience and remote sensing, Vol 40-5, pp 1065-1081, 2002

[2] L. Parra, K. Mueller, C. Spence, A. Ziehe, and P. Sajda, "Unmixing Hyperspectral Data," Advances in Neural Information Processing Systems 12, MIT Press, pp 942-948, 2000.

[3] T. Nishikawa, H. Saruwatari, and K. Shikano, "Comparison of time-domain ICA, frequency-domain ICA and multistage ICA for blind source separation," in Proc. Eur. Signal Processing Conf. (EUSIPCO), vol. 2, Sep. 2002, pp. 15–18.

[4] Marc Castella, Jean-Christophe Pesquet, Athina P. Petropulu "A Family of Frequency- and Time-Domain Contrasts for Blind Separation of Convolutive Mixtures of Temporally Dependent Signals" IEEE Trans. on Signal Processing, vol. 53, n. 1, pp. 107-120, 2005

[5] Chia-Yun Kuan, Glenn Healey "Using independent component analysis for material estimation in hyperspectral images" JOSA A, Volume 21, Issue 6, 1026-1034 June 2004

[6] C. Jutten and J. Herault, "Blind separation of sources, Part I: An adaptive algorithm based on neurominic architecture", Signal Processing, Vol. 24, pp.1-10, 1991.

[7] A. Rummert Cichocki, R. Unbehauen, "Robust Neural Networks with on-line learning for blind identification and blind separation of sources", IEEE Trans Circuits and Systems I: Fundamentals theory and Applications, Vol. 43, No.11, pp. 894-906, 1996.

[8] E. Oja and J. Karhunen, "Signal separation by non-linear Hebbian learning, In M. Palaniswami, et al (editors), Computational Intelligence – A Dynamic System Perspective, IEEE Press, New York, NY, pp.83-97, 1995.

[9]. S. Amari, A. Cichochi and H. H Yang, Recurrent Neural Networks For Blind Separation of Sources", 1995 International Symposium on Nonlinear Theory and its Applications, NOLTA'95, Las Vegas, December, pp.37-42, 1995.

[10] Smalz, R., Conrad, M., "Combining evolution with credit apportionment: A new learning algorithm for neural nets", Neural Networks, vol 7, num 2, 1994.

[11] Opitz, D.W., Shavlik, J.W., "Actively searching for an effective neural network ensemble", Connection Sci, vol 8, num 3, 1996.

[12] Moriarty, D.E., Miikkulainen, R., "Forming neural networks through efficient and adaptive coevolution", Evolutionary Computation, vol 4, num 5, 1998.

[13] Darwen, P., Yao, X., "Speciation as automatic categorical modularization", IEEE Transactions on Evolutionary Computation, 1(2), 1997.

[14] Goldberg, D., Richardson, J., "Genetic algorithms with sharing for multimodal function optimization", Proceedings of 2nd International Conf. on Genetic Algorithms, 1987.

[15] Coello, C., Van Veldhuizen, D., Lamont, G., "Evolutionary algorithms for Solving Multi-Objective Problems", Kluwer Academic Publishers, 2002.

[16] García-Pedrajas, N., Hervás-Martínez, C., Muñoz-Pérez, J., "COVNET: A Cooperative Coevolutionary Model for Evolving Artificial Neural Networks", IEEE Transactions on Neural Networks, vol 14, num 3, 2003.

[17] F. Bellas, R.J. Duro, "Multimodule ANNs Through evolution using an Affinity Based Operator", Proceedings book JCIS 2003, pp 311-314