NERVES and ELENA : the Basic Research on Artificial Neural Networks in Europe

M. Verleysen
Université Catholique de Louvain
Microelectronics Laboratory
REGARDS
pl. du Levant 3, 1348 Louvain-la-Neuve (Belgium)
E-mail: verleysen@dice.ucl.ac.be

Abstract

Research on artificial neural networks goes beyond classical models and learning algorithms, and their application to adaptive tasks. The need for basic research exists, i.e. for research on learning algorithms, possibilities for specialized hardware, for interfaces between the world of conventional computers and neural networks, and especially for objective comparative studies between neural and classical methods.

These are roughly the aims of the Nerves project (July 1989 - June 1991), and of the Elena project (July 1992 - June 1995). The purpose of the Nerves project was to develop theoretical tools and technical means in order to design algorithms, machines and VLSI circuits for neurocomputing. The Elena project aims to investigate the relations between statistical methods of data classification (estimation of Bayes boundaries between classes,...), and the neural methods. It concerns theoretical studies on evolutive algorithms and relations with statistics, development of a graphical software environment and of test databases, and definition of VLSI architecture and analog/digital chips for evolutive neural networks. Conclusions of the NERVES project and preliminary results of the ELENA project are presented here.

1 Introduction

"Artificial neural networks": what does it mean? Since the pioneering works of MacCullogh and Pitts, a lot of literature was published in the various domains covered by neural networks. In fact, neural networks are not really a field. It is a collection of ideas, more or less related to the biological networks of neurons, and based on the same principles which will be detailed later. What concerns artificial neural networks, the study of models of course occupies the first place. Models are based on biological background, at least in the description of the behavior of neurons; some models also mimic biology at the structural level, i.e. in the way of how neurons are connected and organize themselves. Models can also be a way to study parallel structures, realizing simple, fast, and mostly non-linear operations, and whose only the global behavior is of importance. Models can also be a pretext to characterize a process, which needs to be described in simple terms to be efficiently analyzed. Finally, neural networks models can also be a means to describe methods already known in signal processing, in classification, in statistics, ..., but to describe them in terms more accessible to non-specialists, also providing tools which are most of the time not available in the original domain of the method.

By enumerating a non-exhaustive list of different types of models, we pointed out two important concepts which will govern the field. First, some properties make the definition of artificial neural networks: parallelism, simple computations, global behavior, asynchronism, fault-tolerance, learning, non-explanatory models,... The link between biology and neural systems is often not obvious, except that some properties are shared by the two domains. This fuzzy description of artificial neural networks, which is not, and does not have to be, a definition, leads to the second important concept: neural networks include a lot of very various fields, so much that it is impossible to find scientists aware of all developments in all branches of this research. The problem is even more complex: specialists of one of the application fields of neural networks, who want to introduce such new methods of computation and to compare them to results obtained with more "classical" methods, hardly find their way in the huge amount of papers, books, conference proceedings and lecture notes that are available. This is perhaps the point where most criticisms can be done against the neural networks field: too few studies compare the results obtained with neural networks to those obtained with classical methods of information or data processing, statistics, classification, signal processing,... The whole field has obviously much to win by setting links with other fields; we will come again later to this aspect.

But models are not alone in the different aspects of the field. There is also an abundance of application fields, and many researchers prefer to use top-down approaches from the application to the system than the opposite. Applications go from image compression to prediction of temporal series, through classification, OCR, speech processing, process control,... Again, the diversity of applications makes it difficult to have common views, common goals and common tools.

Finally, the way how to use neural networks are also varied: software simulation, parallel implementations on transputers or other specialized machines, dedicated stand-alone VLSI, dedicated accelerators, ..., all solutions providing of course advantages and drawbacks in various applications.

Considering these aspects, it should be more appropriate to speak about neural networks fields than field. Covering the whole domain is impossible, but interactions between the various aspects are beneficial for everybody. Sharing of knowledge between neural networks researchers and scientists from more classical fields is also necessary, and the domain of ANN can thus be viewed as a huge scientific field where differences and lack of understanding is the key for a healthy and beneficial future.

2 Research funded by the CEC

2.1 ESPRIT projects

The interest from the Commission of the European Communities towards artificial neural networks is not new. Specific problems and issues have already been addressed in the BRAIN program, in the 80s. Phases II and III of the ESPRIT program in Information Technology went however a step further in the interest towards artificial neural networks.

Industrial projects have been lead in the domain: to mention only one of the most important, let us cite PYGMALION, and its second phase GALATEA. As stated in its synopsis [1], "the aim of PYGMALION was to create an independent European technological base for the applications, algorithms and software aspects of neurocomputing. The project led to the coordination of research on connectionist computing techniques and provided the means for developing the necessary software tools for productive research".

Aims of PYGMALION, besides the dissemination of information in the European research community, were to develop tools to stand as interface between applications and emulation architectures, both software and hardware, and to prove the usefulness and the potential of neural network approaches through chosen industrial applications (software and hardware). What concerns the results, the PYGMALION project led to the developments of specification languages for neural network programming, to first VLSI demonstrators which may be used for several neural models, and to positive results in some applications domains: low-level imaging (image compression, segmentation and texture analysis), high-level imaging, speech recognition and acoustic signal classification.

The objective of GALATEA obviously is to finalize the first results obtained in PYGMALION, and "to construct a general-purpose neural computing system for Europe" [1]. This system will encompass general-purpose neurocomputer hardware build to support a large class of neural networks, a programming environment for this hardware, for domain-specific processors and for ASICs, and a silicon compiler for the production of ASICs. The usefulness of the hardware and software developed will be demonstrated through three industrial applications: an Optical Character Recognition (OCR) system based on neural techniques, and an industrial vision package used in Surface Mounted Devices (SMD) techniques and in fruit video-grading systems.

PYGMALION and GALATEA are two industrial ESPRIT projects. Many industrial partners are involved in it, and a company MIMETICS was created to exploit the results of the two projects. Encouraging results obtained in these projects concern not only the development of hardware and software, but also the dissemination of information in industrial domains, by showing working demonstrators in specific applications. The coordinator of these two projects is THOMSON-CSF (France).

2.2 NERVES

The aims of the NERVES project, proposed to the Commission in 1988, were quite different from those of industrial projects PYGMALION and GALATEA. At the beginning, NERVES was intended to include 6 partners, with various experiences in the field of neural networks. The purpose of this proposal was clearly to coordinate the actions undertaken in some recognized European laboratories in this field, to avoid too important overlappings between the different works, and to exchange ideas necessary for the continuation of projects already started. The project was mainly oriented towards VLSI and machines, while some other aspects were included too.

In the 1989 call for proposals for ESPRIT projects, another proposal from Edinburgh, with 4 partners, had roughly the same objectives of sharing of knowledge. Moreover, some aspects on VLSI implementations of neural networks overlapped with the NERVES proposal. The EEC asked thus to merge the two proposals, even adding one supplementary laboratory. After a 3-months definition phase, financially supported by the Commission, a new proposal was sent to the Commission and accepted. This proposal included 11 partners from 7 countries :

Institution	Team coordinators
INPG Grenoble /LTIRF(coordinator) (F)	J. Hérault
EPFL Lausanne (CH)	J.D. Nicoud, F. Blayo
IMAG Grenobla (F)	T. Muntean, P. Bessière
IMS Stuttgart (D)	L. Spaanenburg
Politecnico di Torino (I)	D. Del Corso
St-Patrick's College Dublin (IR)	R. Reilly
UCL Louvain-la-Neuve (B)	P. Jespers, M. Verleysen
Universität Dortmund (D)	K. Goser, V. Tryba
University of Edinburgh (GB)	A. Murray
University of Oxford (GB)	L. Tarassenko
CSEM Neuchâtel (CH)	E. Vittoz

table 1: partners of the NERVES project

The purpose of the NERVES project was clearly more to create a network of abilities, to share knowledge and to benefit from the experience of some to conduct projects of others than to really work towards a common objective. A lot of aspects of the neural network field were covered by NERVES; the main results are explained here [2].

1) task B1: visual processing of text
The aim of this task was ambitious, and could obviously not be reached in the lifetime of the project. It consisted in the design and the implementation of a connectionnist software system for the visual processing of text that would take as input a pixel-based text image and produce as output a semantic-level representation. Optical text

recognition techniques are generally not based on cognitive modeling; the purpose here was to model as much as possible the retinal architecture and eye-movement control mechanism of human readers, and to build a vertical connectionnist model with different levels of representation (from the visual to the linguistic). Having several representation levels in the same model is of most importance to have the possibility to include contextual information, like limited number of possible characters in the lowest levels and linguistic context in the highest levels [3].

2) task B2: silicon implementation constraints and implications
While implementing neural algorithms on VLSI, several supplementary parameters have to be taken into consideration. The main one is the accuracy that is needed to efficiently perform the operations in VLSI neural networks. By using analog techniques, accuracy in circuits is limited to the matching of components (transistors, capacitors,...), which generally does not exceed 6-7 bits unless special area-consuming design is used. Digital networks can reach any accuracy provided all cells (memory points, multipliers, adders,...) cope with this accuracy; increasing the number of necessary bits automatically increases either the silicon area, either the computation times, either a compromise between the two. Specific techniques, like pulse-stream circuits developed by some partners of the NERVES project [4], are an interesting compromise between the accuracy reached, the silicon area used and the performances of the circuit.

Investigations of task B2 showed that Hamming networks are the most insensitive to reduction of accuracy. Binary-valued networks are ideal for implementation, even if simple analog cells are used. Nearest-neighbor classifiers show almost the same insensitivity. It was also proved that the pulse-stream technique and others developed in the project could lead to accuracies around 6-7 bits, which is sufficient for such types of networks, but not for Multi-Layer Perceptrons (MLP) learning. It was however shown on the text-to-speech problem that the use of a MLP network after learning can be achieved with such accuracy.

3) task B3: high-level specification language
Implementing neural algorithms either on standard computers either on specialized hardware can be done in various ways, going from standard C programming with adequate interfaces to design of hardware implementing a particular model. In this task, it was chosen to develop several layers of software between the description of artificial neural networks models and their implementation on standard or specialized computers [5]. A high-level description language (MENTAL) was specified, together with a virtual machine layer aimed to describe the execution of the algorithms. This constitutes a top-down approach, where the compilers between these two layers and between the virtual machines and the hardware were also considered.

A bottom-up axis was also developed, starting from the target machines considered in the project (SMART and SuperNode), and adding several software layers around their kernel language. The last of these layers consisted in Vectorial-C, an extension of the C language with abilities to manipulate matrices and vectors for the SMART machine, and NEURAL, an extension of OCCAM, for SuperNode. Both Vectorial-C and NEURAL are linked to the virtual machine layer through ...

Finally, several applications were implemented, mostly in low-level languages, to prove the feasibility of this task. Specifications of languages were finalized before the end of the project, but not all the implementations.

4) task D1: architectures for neurocomputers
To implement neural networks on hardware, three main issues must be considered: the flexibility for implementing diverse ANN models, the huge number of interconnected cells, and the interfacing with existing machines. During the NERVES project, several partners studied several possibilities to consider these three issues. The problem of interfacing was studied both between two "intelligent" systems (a workstation and a neural architecture based accelerator board for example), and for a master-slave configuration (a workstation and a chip), leading to two definitions of interfaces.

Three architectures were also studied and developed. GENES [6], developed at EPFL, is based on systolic computations and is very efficient if the algorithms are adequately transformed in a systolic representation; this work has been done for single-layer and Hopfield networks, implementing Hebb, perceptron and Kohonen rules. Secondly, an architecture specially adapted to sparse matrix computations, SMART [7], was developed at INPG. It consists in a loose pipe-line with FIFO buffering, connected to a SparcStation. Finally, IMAG focused on the possibility to run concurrent processes on different processors to accelerate the computations; the main problems which have been addressed concern the communications between chips.

5) tasks E1/E2: design of associative memories
Associative memories may be used in many classification problems (speech and image recognition,...). Many techniques are however grouped under the term "associative memories": single-layer networks (Hamming nets in the case of binary weights), Hopfield and related nets, Kohonen maps, vector quantizers,... The purpose of these tasks were to investigate different possibilities to implement either analog or digital associative memories. Digital Kohonen maps [8] and analog single-layer networks (Hopfield nets) [9] were realized to prove the feasibility of the architectures and to evaluate the performances in what concerns accuracy and speed. The use of analog, digital, stochastic and pulse-stream computations was considered.

6) task E3: design of source separation circuits
Independent sources separation certainly is one of the most impressive examples of applications of neural networks. Considering two independent sources which are not observable, and two different observable linear mixings of these two sources, the problem is to retrieve the two original signals, the mixing matrix being of course unknown. The solution of this problem has many applications in signal processing, data transmission,... INPG has shown how to solve the independent sources separation problem for linear instantaneous mixings [10]; the work is still under progress for convolutive mixings. During the NERVES project, the problem has been addressed both theoretically by the analyses of convergence and stability of the proposed neural network solution, and on the implementation level by proposing a fully analog realization of the algorithm [11].

6) task F1: design of pulse-stream synapses
Several VLSI architectures for neural network implementations were developed during the NERVES project. These architectures need of course, to be implemented, the design of VLSI cells, paying special attention to some parameters such as accuracy and size. Tasks F1 to F3 concern the design of analog and mixed analog-digital building blocks for these implementations.

In task F1, cells for pulse-stream architectures were developed [4]. Two cells have been implemented, one based on time division of pulses, the other one on pseudo-switched-capacitor techniques, and test chips were realized. Coupled to digital weight memory and pulse-width input-weight multiplication, a complete chip with 4 blocks of 16 neurons with 16 synapses each has been designed. Four-transistors transconductance multiplier-based synapses with weight leakage compensation, new neuron oscillators, and several cells aimed to reduce the area needed for pulse-stream computations were also designed.

7) task F2: design of building blocks for analog implementations
This task is more or less the equivalent to task F1 but this time for analog realizations instead of pulse-stream techniques. The problem of analog sum-of-products has been examined through the realization of test chips and through theoretical studies about matching. Reconfigurable neurons and cascadable matrices of synapses were also studied. The storage of analog weights is also of great importance in the analog realizations of neural networks; this problem has been studied by CSEM and UCL, who propose two solutions [12]: the first one relies on local refreshment of weights through the comparison of the stored value to an external ramp, the second one on a global sequential refreshment of all weights stored on the chip by comparing each stored value to a set of predefined levels. In both cases, once the comparison is realized, the weight is refreshed to the next upper discretized level, allowing thus a subsequent decrease during a defined period because of the leakage currents. By these methods, weights will be kept into the range between two defined levels, corresponding to the accuracy achieved in the memory.

8) task F3: design of technology for analog synapse memory
The same problem of analog synapse memories has been addressed in task F3, but here by proposing new VLSI technologies or the use of specialized ones to realize the memory points. Several solutions have been proposed: analog EEPROMs where the floating gate is accurately controlled by special programming techniques [12], use of UV-light to modify the conductance of a special technological layer used to connect the storage capacitor with the rest of the circuit, use of non-volatile 2-terminal α-SI (amorphous silicon) metal/p^+ junctions, and use of special technological layers to reduce the programming voltages of EEPROMs. Some studies have also been carried out on circuitry for the compensation of leakage currents.

As it can be seen in this description, many ways have been explored under the NERVES project. The expertise areas of the different partners were very different one from another, and this project had the merit to open new perspectives to all researchers who contributed in it by the sharing of their knowledge and their new results. The good results obtained in the NERVES project are not only due to common works, but also to the frequent meetings (three per year), to the participation of members of the consortium to various international conferences, to the dissemination of information realized by many of them through courses and lectures in their institutions and abroad, to the participation in exhibits (Neuro-Nîmes),... The state-of-the-art in the neural network field, between 1989 and 1991, period during which the NERVES contract was running, certainly needed such opening of new perspectives and dissemination of information.

The situation is now different. While sharing of knowledge and dissemination of information is still needed in the neural network field, as in any scientific domain, people now begin to be more aware of the progress in their domain of interest, and, which is very important, begin to make the difference between real innovations and rediscoveries of known methods. The projects proposed to the Commission in 1991 reflected such tendency: they were more specialized, gathered less partners, and generally not concerned only neural networks but more often relations between neural networks and other fields.

2.3 ELENA

ELENA may be considered as a continuation of the NERVES project, as it gathers several of its partners (through some of the researchers moved from one institution to another; this is also a result of the European collaboration...). ELENA is a project focused on classification and evolutive neural algorithms.

Classification concerns a lot of applications where data (often high-dimensional) must be associated to a finite number of classes. Most often however, neural networks are used in classification tasks without any study of comparisons with more classical statistical methods. In particular, the Bayes theory shows where to fix limits between classes when their distributions are overlapping; most neural networks even don't try to approximate the Bayes boundaries, and give thus results much worst than statistical methods...

Neural networks however can find advantages in the evolutive character of some structures; many statistical methods indeed only give asymptotic proves of convergence, with no indication on the sizes (learning set, number of kernels in case of kernel estimators,...) needed to obtain a defined accuracy in the estimation of classes boundaries.

The ELENA project tries to make the link between evolutive architectures for neural networks, i.e. networks whose size evolves with the number, the nature and the complexity of the data to handle, and statistical methods of classification, based on kernel estimators of probability densities. The project includes three axes, detailed below: theory, software and benchmarks, and hardware. Six partners are involved in the ELENA project, which runs since July 1992.

Institution	Team coordinators
INPG Grenoble /LTIRF (coordinator) (F)	C. Jutten
EPFL Lausanne (CH)	J.D. Nicoud, Y. Cheneval
UCL Louvain-la-Neuve (B)	M. Verleysen
EERIE Nîmes (France)	F. Blayo
Universidad Politecnica de Catalunya (SP)	J. Cabestany
Thomson-Sintra Sophia-Antipolis (F)	P. Comon

table 2: partners of the ELENA project

1) Axis A: theory

Axis A basically covers the theory of kernel estimators of probability densities and radial-basis functions. Once probability densities of classes are known, the Bayes criterion gives the best limits between classes to use to decrease the number of misclassifications, a priori probabilities of classes being given. The first results of this axis [13] concern the estimation of probability densities by kernel estimators, including their choice and the number of samples required in large dimensions, the probability of errors and the computation of the confusion matrix, the relations between these methods and radial-basis functions (RBF), and practical issues concerning the use of RCE and LVQ procedures to approximated Bayes boundaries [14-15-16].

2) Axis B: simulations and benchmarks

Studies about classification cannot be performed without comparisons between different techniques. For objective comparisons, it was decided to collect two types of databases to use as standards, at least inside the project: artificially-generated databases for preliminary tests on algorithms, and real databases collected in the literature of this field and among often tested problems of classification (Fisher's Iris database, ATT Bell labs characters database,...).

Testing new algorithms is also fastidious to implement in low-level programming languages such as C, especially when graphical input-outputs are needed. Based on the experience of the NERVES project where some graphical simulators were developed, it was decided to create a graphical environment for neural networks simulation, called PACKLIB, which includes abilities for easy visualization of neural networks, graphical description of neural algorithms based on modules developed in C,... Many algorithms have already been implemented in this environment; its main advantage is that changes in the use of basic functional blocks to describe an algorithm may be achieved without any programming, giving thus an incredible flexibility to test new models. Finally, since the development of basic modules must be done in C, matrix computations and graphical libraries were develop to help the programmer in his task.

3) Axis C: VLSI implementations

Different problems are encountered when implementing evolutive neural algorithms on VLSI chips rather than other neural networks. Problems of precision must be reconsidered, and the cascadability is of course important too. The first work achieved in this axis concerns the influence of the limitations of accuracy in evolutive algorithms, and investigations to evaluate the attainable accuracy in analog VLSI chips. In particular, it is studied how the SOI (Silicon-On-Insulator) technology could be used to exploit the reduced leakage currents obtained with such process.

A second task in this axis concerns the definition of VLSI architectures aimed to implement PLS (Piecewise Linear Separation) and ROI (Region-Of-Influence) algorithms; two digital architectures have been up to now proposed [17].

Finally, a last task will include the realization of VLSI building blocks for analog and digital chips.

The work of the ELENA project is currently under progress. The aim for the remaining of the project is to clearly set methods and evaluate their condition of use and their limitations for the classification of data, and to set up both software and hardware tools which could be used for classification applications. Evolutive neural algorithms with be mainly studied, but without forgetting their links with the statistical Bayesian methods.

Acknowledgments

This paper (too) briefly summarizes the work realized by more than 70 researchers during several years. It would be too long to cite them here, but I would like to thank them for the wonderful job they did and they are still doing. My special thanks go to Jeanny Hérault and Christian Jutten, respectively coordinators of the NERVES and ELENA project, who make the European collaboration described here possible.

References

[1] Esprit Synopses: Information processing Systems and Software. Commission of the European Communities, Publication office, Luxemburg, 1992.
[2] NERVES, an ESPRIT Basic Research Action : final Report. J. Hérault, INPG Grenoble, 1991.
[3] Connectionist approaches to sentence processing. R. Reilly, C. Doherty, proceedings of Neuro-Nîmes 1990 (Nîmes, France, November 1990), EC2, Paris.
[4] Pulse Stream VLSI neural networks - mixing analog and digital techniques. A. Murray, D. Del Corso, L. Tarassenko, IEEE Transactions on Neural Networks, 1991, pp. 193-204.
[5] A virtual machine model for artificial neural networks programming. P. Bessière, A. Chams, T. Muntean, proceedings of INNC90, Paris, 1990.
[6] Digital VLSI generic elements for neuro-emulators using systolization. C. Lehmann, F. Blayo, proceedings of the International Workshop on Algorithm and Parallel VLSI Architectures, Pont-à-Mousson (France), June 1990.

[7] SMART: how to simulate huge networks. J.C. Lawson, A. Chams, J. Hérault, ITG-IEEE Workshop on Microelectronics for Neural Networks, Dortmund (Germany), June 1990.
[8] VLSI implementation of an associative memory based on distributed storage of information. U. Rückert, K. Goser, ITG-IEEE Workshop on Microelectronics for Neural Networks, Dortmund (Germany), June 1990.
[9] Analog VLSI implementations of Hopfield's neural network. M. Verleysen, P. Jespers, IEEE Micro, December 1989.
[10] Separation of sources, part 1: an adaptive algorithm based on neuromimetic architecture. C. Jutten, J. Hérault, Signal Processing, July 1991.
[11] CMOS integration of Hérault-Jutten cells for separation of sources. E. Vittoz, X. Arreguit, in: "Analog Implementations of Neural Systems", C. Mead and M. Ismail eds., Kluwer Academic Publishers, Norwell (MA), 1990.
[12] Analog storage of adjustable synaptic weights. E. Vittoz et al., ITG-IEEE Workshop on Microelectronics for Neural Networks, Dortmund (Germany), June 1990.
[13] Neural Bayesian Classifier. C. Jutten, P. Comon, proceedings of the International Workshop on Artificial Neural Networks, Sitges (Barcelona, Spain), June 1993.
[14] Setting initial conditions for the RCE model. F. Blayo, M. Verleysen, proceedings of the First IFIP Working Group 10.6 Workshop, Grenoble (France), March 1992.
[15] Optimal decision surfaces in LVQ1 classification of patterns. M. Verleysen, P. Thissen, J.D. Legat, proceedings of the European Symposium on Artificial Neural Networks 1993, Brussels (Belgium), May 1993.
[16] Learning Vector Classification: an improvement on LVQ algorithms to create classes of patterns. M. Verleysen, P. Thissen, J.D. Legat, proceedings of the International Workshop on Artificial Neural Networks, Sitges (Barcelona, Spain), June 1993.
[17] Hardware implementation of piecewise linear separation incremental algorithms, J.M. Moreno, F. Castillo, J. Cabestany, proceedings of Neuro-Nîmes93 (Nîmes, France, October 1993), EC2, Paris.