The thesis, EVOLVING ARTIFICIAL NEURAL NETWORKS USING PERMUTATION PROBLEM FREE MODIFIED CELLULAR ENCODING, submitted by MOHAMMAD MASUD HASAN, ROLL No. 040205035P, Session: April, 2002, Registration No. 95394 to the Department of Computer Science and Engineering of Bangladesh University of Engineering and Technology has been accepted as satisfactory for partial fulfillment of the requirements for the degree of M.Sc. Engg. in Computer Science and Engineering and approved as to its style and contents. Examination held on August 16, 2004.

BOARD OF EXAMINERS

1 ______________________________

Dr. Md. Monirul Islam
Assistant Professor
Department of CSE
BUET, Dhaka-1000

Chairman
(Supervisor)

2 ______________________________

Dr. Md. Shamsul Alam
Professor and Head
Department of CSE
BUET, Dhaka-1000

Member
(Ex-officio)

3 ______________________________

Dr. M. Kaykobad
Professor
Department of CSE
BUET, Dhaka-1000

Member

4 ______________________________

Dr. Md. Quamrul Ahsan
Professor
Department of EEE
BUET, Dhaka-1000

Member
(External)

Declaration

I, hereby, declare that the work presented in this thesis is done by me under the supervision of Dr. Md. Monirul Islam, Assistant Professor, Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology, Dhaka-1000. I also declare that neither this thesis nor any part thereof has been submitted elsewhere for the award of any degree or diploma.

Countersigned,

Dr. Md. Monirul Islam **(Supervisor)**	**Mohammad Masud Hasan**

Acknowledgements

All praises are for the Almighty ALLAH who is the most beneficent and the most merciful.

First of all, I would like to thank my supervisor Dr. Md. Monirul Islam, Assistant Professor, Department of CSE, BUET for teaching me how to carry out a research work. I express my heart-felt and most sincere gratitude to him for his constant supervision, valuable advice and continual encouragement, without which this thesis would have not been possible.

Special thanks should be given to Prof. Dr. Md. Shamsul Alam, Head of the CSE Department, for his continuous suggestions regarding the thesis.

I would like to thank the other members of my examination board Prof. Dr. M. Kaykobad and Prof. Dr. Md. Quamrul Ahsan for their valuable suggestions and of course for their unconditional encouragements.

I must have to acknowledge with sincere thanks the constant support and patience of my family members- my parents specially. The arduous task of having completed the thesis could not have been accomplished without their assistance.

Finally I would like to acknowledge the all-out cooperation and services rendered by the members of the Department of CSE, BUET.

Abstract

This thesis works with a new evolutionary system for feedforward artificial neural networks (ANNs). An indirect encoding scheme, to be particular, modified cellular encoding (MCE) is proposed to represent ANNs. The original cellular encoding is modified in such a way that it does not suffer from the well-known permutation problem or competing conventions problem of genetic algorithms for evolving ANNs. The functionality of some program symbols in cellular encoding is changed; new rules are added. As a consequence, it is possible to apply crossover operator in the genetic search. Radical change of architecture i.e. behaviour from parents to their children is stopped by keeping the application of crossover on genotypes within certain levels. It is shown in this work that addition / deletion of nodes / connections can evidently be done by crossover alone. Other attempts are also taken to minimize behavioural disruption between parents and their offspring. In the evolution system, the number of user specified parameters is also decreased.

The evolutionary system is also implemented and its performance is tested on some real world problems. The upshot of the genetic search is studied and assessed against the contemporary researches, although direct comparison with other evolutionary approaches to designing ANN is very difficult. It is shown in this thesis that the genetic search can find a reasonable ANN from the search space in considerably short period.

Contents

List of Figures

List of Tables

List of Notations

ANN	Artificial Neural Network
ART	Adaptive Resonance Theory
BP	Back Propagation
CE	Cellular Encoding
DE	Direct Encoding
EP	Evolutionary Programming
ES	Evolutionary Strategy
GA	Genetic Algorithm
GP	Genetic Programming
LMS	Least Mean Square
LVQ	Linear Vector Quantization
MCE	Modified Cellular Encoding
MLP	Multi-layer Perceptron
n	Number of nodes in the output layer
O_{max}	Maximum values of output coefficients
O_{min}	Minimum values of output coefficients
PST	Program Symbol Tree
RSG	Random Selection Group
SOM	Self Organizing Map
T	Total number of input pattern
$Y(i,t)$	Actual output for the i-th output neuron of the t-th input pattern
$Z(i,t)$	Desired output for the i-th output neuron of the t-th input pattern

Chapter 1
Introduction

1.1 Introduction

For decades after Darwin laid down its basic principles, evolution was the domain of biologists and paleontologists. When the synthetic theory brought the successful union of Darwinian principles with Mendelian genetics at the turn of the nineteenth century, most biologists were confident that they had a solid conceptual basis for biology. The mathematical theory of evolution came to be dominated by population genetics, which was commonly thought to provide a sufficiently deep theoretical framework for analyzing the constituent mechanisms driving evolutionary processes. Over the same period that witnessed the flourishing of evolutionary science, starting in the mid- to late-nineteenth century, new concepts and methods were developed in mathematics and the natural sciences that now promise to remove several of the roadblocks to an integrative theory of evolutionary systems.

Evolutionary computation has provided an alternative to the more classical search and optimization methods in recent years. Classical methods tend to get stuck in local optima. One of the advantages of evolutionary computation is that the algorithms do not start from a local search point but explore different areas of the search space in parallel. Other advantages are that they have no presumptions with regard to the search space, that they are widely applicable, that they can be interpreted, that they provide several alternative solutions to the problem at hand and that they are easily combined with other methods.

The idea of using evolutionary computation as a problem solving technique exists since the 1950s. Since then, four major approaches have evolved [25]: Evolutionary Programming (EP), Evolution Strategies (ES), Genetic Algorithms (GA) and Genetic

Programming (GP). All these algorithms have been inspired by the notions of evolution and survival in nature.

Artificial neural networks (ANNs), also referred to as neuromorphic systems, artificial intelligence and parallel distributed processing, are an attempt at mimicking the patterns of the human mind. Many researches have concluded that understanding the human mind is probably the most difficult challenge left in science. Consequently, ANNs have seen an explosion of interest over the last few years, and are being successfully applied across an extraordinary range of problem domains, in areas as diverse as finance, medicine, engineering, geology and physics. Indeed, anywhere that there are problems of prediction, classification or control, ANNs are being introduced. This sweeping success can be attributed to a few key factors like power, easy of use and applicability.

The power of ANNs is that they are very sophisticated modeling techniques capable of modeling extremely complex functions. In particular, ANNs are nonlinear. For many years linear modeling has been the commonly used technique in most modeling domains since linear models have well-known optimization strategies. Where the linear approximation was not valid (which was frequently the case) the models suffered accordingly. ANNs keep in check the curse of dimensionality problem that bedevils attempts to model nonlinear functions with large numbers of variables.

Another key factor is the ease of use. ANNs learn by example. An ANN user gathers representative data, and then invokes training algorithms to automatically learn the structure of the data. Although the user does need to have some heuristic knowledge of how to select and prepare data, how to select an appropriate ANN, and how to interpret the results, the level of user knowledge needed to successfully apply ANNs is much lower than would be the case using (for example) some more traditional nonlinear statistical methods.

Also, ANNs are applicable in virtually every situation in which a relationship between the predictor variables (independents, inputs) and predicted variables (dependents, outputs) exists, even when that relationship is very complex and not easy to articulate in the usual terms of "correlations" or "differences between groups." The computing world has a lot to gain from ANNs. ANNs also contribute to other areas of research such as neurology and psychology.

1.2 Literature Review

The field of ANNs has a history of some five decades but has found solid application only in the past fifteen years, and the field is still developing rapidly. In the early 1940's scientists came up with the hypothesis that neurons, fundamental, active cells in all animal nervous systems might be regarded as devices for manipulating binary numbers. Thus spawning the use of computers as the traditional replicants of ANNs.
To be understood is that advancement has been slow. Early on it took a lot of computer power and consequently a lot of money to generate a few hundred neurons. In relation to that consider that an ant's nervous system is composed of over 20,000 neurons and furthermore a human being's nervous system is said to consist of over 100 billion neurons! To say the least replication of the human's neural networks seemed daunting. However, today ANNs are being applied to an increasing number of real- world problems of considerable complexity. The history of ANNs that was described above can be divided into several periods [46]:

a) First Attempts: There were some initial simulations using formal logic. During the decade of the first electronic computer, McCulloch and Pitts (1943) developed models of neural networks based on their understanding of neurology. These models made several assumptions about how neurons worked. Their networks were based on simple neurons which were considered to be binary devices with fixed thresholds. The results of their model were simple logic functions such as "a or b" and "a and b". Another attempt was by using computer simulations. There were two groups (Farley and Clark, 1954; Rochester, Holland, Haibit and Duda, 1956) [42]. The first group (IBM researchers) maintained close contact with neuroscientists at McGill University. So whenever their models did not work, they consulted the neuroscientists. This interaction established a multidisciplinary trend which continues to the present day.

b) Promising and Emerging Technology: Not only was neuroscience influential in the development of neural networks, but psychologists and engineers also contributed to the progress of neural network simulations. Rosenblatt (1958) stirred considerable interest and activity in the field when he designed and developed the Perceptron. The Perceptron had three layers with the middle layer known as the association layer. This system could learn to connect or associate a given input to a random output unit. Rosenblatt also took part in constructing the first successful neurocomputer, the Mark I Perceptron. Another system was the ADALINE (ADAptive LInear Element) which was developed in 1960 by Widrow and Hoff (of Stanford University) [46]. The ADALINE was an analogue electronic device made from simple components. The method used for learning was different to that of the Perceptron, it employed the Least-Mean-Squares (LMS) learning rule.

c) Period of Frustration and Disrepute: In 1969 Minsky and Papert wrote a book in which they generalized the limitations of single layer Perceptrons to multilayered systems. In the book they said: "...our intuitive judgment that the extension (to multilayer systems) is sterile". The significant result of their book was to eliminate funding for research with neural network simulations. The conclusions supported the dis-enhancement of researchers in the field. As a result, considerable prejudice against this field was activated.

d) Innovation: Although public interest and available funding were minimal, several researchers continued working to develop neuromorphical based computational methods for problems such as pattern recognition. During this period several paradigms were generated which modern work continues to enhance. In 1988, Grossberg's influence founded a school of thought which explores resonating algorithms [42]. They developed the ART (Adaptive Resonance Theory) networks based on biologically plausible models. Anderson and Kohonen developed associative techniques independent of each other. Klopf in 1972 developed a basis for learning in artificial neurons based on a biological principle for neuronal learning called heterostasis.

In 1974, Werbos developed and used the back-propagation learning method, however several years passed before this approach was popularized. Back-propagation nets are probably the most well known and widely applied of the neural networks today. In essence, the back-propagation net is a Perceptron with multiple layers, a different threshold function in the artificial neuron, and a more robust and capable learning rule. Amari (A. Shun-Ichi 1967) was involved with theoretical developments: he published a paper which established a mathematical theory for a learning basis (error-correction method) dealing with adaptive pattern classification. While Fukushima developed a step wise trained multilayered neural network for interpretation of handwritten characters. The original network was published in 1975 and was called the Cognitron.

f) Re-Emergence: Progress during the late 1970s and early 1980s was important to the re-emergence on interest in the neural network field. Several factors influenced this movement. For example, comprehensive books and conferences provided a forum for people in diverse fields with specialized technical languages, and the response to conferences and publications was quite positive. The news media picked up on the increased activity and tutorials helped disseminate the technology. Academic programs appeared and courses were introduced at most major Universities (in US and Europe). Attention is now focused on funding levels throughout Europe, Japan and the US and as this funding becomes available, several new commercial with applications in industry and financial institutions are emerging.

A totally unique kind of network model is the Self-Organizing Map (SOM) introduced by Kohonen in 1982. SOM is a certain kind of topological map which organizes itself based on the input patterns that it is trained with. The SOM originated from the LVQ (Learning Vector Quantization) network the underlying idea of which was also Kohonen's in 1972.

Hopfield brought out his idea of a neural network in 1982. Unlike the neurons in Multilayered Perceptron (MLP), the Hopfield network consists of only one layer whose neurons are fully connected with each other. Since then, new versions of the Hopfield network have been developed. The Boltzmann machine has been influenced by both the Hopfield network and the MLP. Adaptive Resonance Theory (ART) was first introduced by Carpenter and Grossberg in 1983. The development of ART has continued and resulted in the more advanced ART II and ART III network models.

The application area of the MLP networks remained rather limited until the breakthrough in 1986 when a general backpropagation algorithm for a multi-layered perceptron was introduced by Rummelhart and Mclelland.

Radial Basis Function (RBF) networks were first introduced by Broomhead & Lowe in 1988. Although the basic idea of RBF was developed 30 years ago under the name method of potential function, the work by Broomhead & Lowe opened a new frontier in the neural network community [42]. The development of ANNs has proceeded as described in Figure 1.

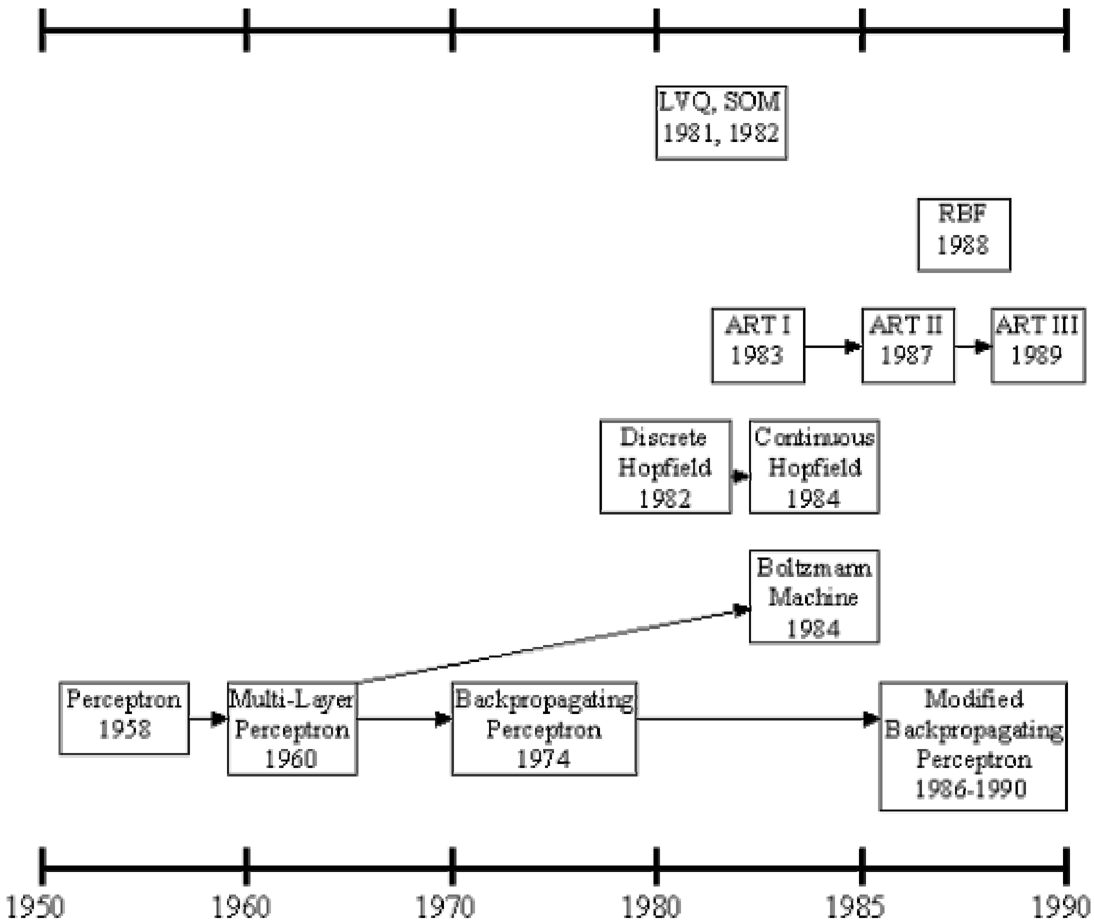

Figure 1.1: The evolution of the most popular artificial neural networks.

g) Today: Significant progress has been made in the field of neural networks-enough to attract a great deal of attention and fund for further research. Advancement beyond current commercial applications appears to be possible, and research is advancing the field on many fronts. Neural based chips are emerging and applications to complex problems developing. Clearly, today is a period of transition for neural network technology.

Most applications of ANNs use feedforward networks and variants of the classical backpropagation (BP) algorithm. All these training algorithms assume a fixed ANN architecture. They only train weights in the fixed architecture that includes both connectivity and node transfer functions. Many attempts have also been made in designing ANN architectures automatically, such as various constructive and pruning algorithms [29], [38], [48], [49], [50].

Associative memories (AMs) can be implemented using networks with or without feedback. In [63], a two layer feedforward ANN is utilized and proposed a new learning algorithm that efficiently implements the association rule of AM. In order to find an appropriate architecture for a large scale real world application automatically and effieciently, a natural method is to divide the original problem into a set of subproblems. In [51], a simple ANN task decomposition method based on output parallelism is presented. Hsin et. al. [26] suggest divide and conquer learning (DCL) schemes for the design of modular ANNs. When a training process in a multilayer perceptron falls into a local minimum or stalls in a flat region, the proposed DCL scheme is applied to divide the current training data region into two easier to be learned regions. In [37], a constructive algorithm for training cooperative neural network ensembles (CNNEs) is presented which have good generalization ability. Paul et. al. [40] show the use of parallel self scaling quasi-Newton (QN) optimization techniques to improve the rate of convergence of the training process for ANNs.

Angeline et. al. [43] indicate two problems of constructive and pruning hill climbing methods: they may be trapped at local optima and they do not investigate complete class of network architectures. That is why researchers [35], [62] argued on behalf of evolutionary algorithms for finding a near optimal system in the ANN architecture search space.

The central task in evolving ANNs is finding a genetic representation, also called chromosome, genotype or encoding, for an ANN [35]. It dictates how the search landscape is structured, and how scalable the method is [41]. Importance has to be given on the optimal representable structures, excluding meaningless structures, yielding valid offspring by the genetic operators etc. Since the first attempts to combine genetic algorithm and neural network started in the late 1980s, other researchers have joined the research and created a flood of papers. A variety of different encoding methods does outcome. Two main directions of ANN encoding are direct encoding and indirect encoding. As characterized by Whitely in 1992, low level or direct encoding techniques mostly specify directly parameters such as connectivity or weight values in the genome. Researchers proposed different types of direct encoding based on connections, nodes, layers, pathway etc.

On the other hand, indirect encoding techniques specify not the parameter themselves but production rules that define how to generate these parameters are encoded. This is biologically motivated by the fact that in case of the human brain, there is much more neurons than nucleotides in the genome. So, there has to be a more efficient way of description. Probably, the first indirect encoding scheme was proposed during 1990 by Kitano [23]. Boers and Kuiper [16] proposed another indirect encoding system which was based on Lindemayer's [2] biological model. But may be the most sophisticated encoding method is developed by Frederic Gruau [18] in 1994 in his PhD thesis, which is called cellular encoding. Yet, cellular encoding is not a fully precise representation method. As argued by Talib Hussain [55] in 1997, the scope of improvement defining cellular encoding would be the components of a representation, possible properties of a cell and limit of a cell has on its navigation of the program symbol tree (PST).

In this thesis work, an attempt is taken to find out an enhanced cellular encoding technique so that it can be applied in ANN search space through evolutionary operators.

1.3 Objectives of this Thesis

This thesis work focuses on interactions between ANN's indirect encoding techniques with the evolutionary algorithms. It tries to remove a well known problem of evolutionary neural network encoding called permutation problem and develop a fast evolutionary search scheme with this genome. In summary, the targets are:

- ✓ Representing ANNs using a new indirect encoding scheme i.e. modified CE scheme that does not suffer from permutation problem.
- ✓ Introducing a new evolutionary system for feedforward ANNs.

- ✓ Applying crossover operator in the genetic search with the intention to reduce the number of user specified parameters.
- ✓ Examine the effects of crossover on cellular encoded genotype.
- ✓ Finding out attempts to reduce the noise in fitness evaluation and minimize behavioural disruption between parents and their offspring.
- ✓ Presenting an algorithm to convert genotype from CE to DE.
- ✓ Applications of the new approach with some real world problems and analysis the result.

1.4 Thesis Organization

The organization of the rest of this thesis is as follows: Chapter 2 represents preliminaries of ANN and its evolution. The biological motivation of artificial neural network, its structure, types, encoding and applications, the evolutionary operators, permutation problem, ways of training and selecting good network etc are covered here.

Chapter 3 starts with the basics of the original cellular encoding. It presents how to develop an ANN from the cellular encoding. Then some modifications are suggested and its new properties are described.

Chapter 4 introduces different types of evolutionary methodology. Along with the new approach, the effects of the genetic operator crossover upon the MCE encoded ANNs are discussed. The algorithm to realize the PST is also presented.

The experimental setup, dataset used and the experimental outcome are given in Chapter 5. An analytical review of the result and the comparison with other works are also given.

Chapter 6 concludes with a summary of the thesis and a few additional remarks about future research directions.