Chapter 1

Introduction

The past few decades have seen a phenomenal growth in the field of digital computers in terms of speed and capacity of computation as well as size and variety of problems for which they are employed. It is however seen that the human brain outperforms even the fastest computer of today in many tasks which are considered mundane. Apart from such capabilities the brain has many more desirable features such as robustness, fault tolerance, flexibility, adaptability, high degree of parallelism, compactness and low power dissipation. Artificial neural networks, an alternative computing paradigm, derives its inspiration from the brain. Artificial neural networks have also been studied as models of brain structures and models of cognitive processes. This work is concerned with artificial neural networks as a model of computation.

The beginning of artificial neural networks as a computational model can be traced to the McCullogh-Pitts' nerve nets introduced in 1943 which modeled neurons as threshold functions. Almost parallelly, continuous models were developed in which differential equations were used for describing neural activity patterns. In the 1950s the McCullogh-Pitts' nerve nets were modeled as a spin system by physicists and a statistical theory of learning based on Hebb's rule was developed. The first learning machine was built by Rosenblatt [137] and coworkers in the late 50s and early 60s. Rosenblatt's discovery of the perceptron convergence procedure for learning the desired weights in such networks resulted in a lot of enthusiasm in the field. However, in 1969 Minsky and Papert [110]

pointed out that perceptrons can learn only what they can represent and that some simple operations like the exclusive-or function are outside its purview. This negative result led to a period of lull in the area of neural networks as an alternative computing model. During this period neural networks were developed as models of associative memory by a number of research groups (see references in [81]). A major contribution continuing from the late 60s is the comprehensive reformulation of neural modeling and learning, based on biological principles by Grossberg [50, 51, 52]. A major resurgence in this field which is continuing till today came about in the late 70s. This renewed widespread interest in the field resulted from a number of major developments such as the learning rule for multilayered perceptrons, concept of energy minimization, structured neural networks and the category learning and classification models. Discussions about these developments and their applications may be found in various books including [58, 87, 124, 106, 140]. A list of applications developed in the past few years may be found in [108]. However, though artificial neural networks is emerging as a powerful computing model for certain classes of problems, it has its share of critics and the topic of neural versus conventional computing is hotly debated.

The success of neural networks in providing efficient solutions to problems which were difficult to solve otherwise is pointed out as a distinct advantage of neural computation over conventional computation. While acknowledging the recent success of neural networks in various spheres of problem solving, Minsky [109] has questioned the capability of neural networks to perform high-level computations. Minsky points out that various tasks such as efficient systematic search, manipulation of complex objects, goal oriented reasoning etc. are beyond the capabilities of homogeneous networks. Michael Roth [139] has argued that the utility of systematic search should not be over emphasised and must be viewed within the overall perspective of the problem. Smolensky [151], while acknowledging the immense contribution of symbolic processing, holds the view that neural computing offers an escape from the brittleness of symbolic processing. Pinker and Price [132] and Latcher and Bever [90] holds the view that the success of neural networks in language processing arise from the network representation being motivated by explicit rules. Hecht-Neilsen [58] holds the view that "neural networks will be shown

to be a particularly good compromise that allows substantial information processing capability while at the same time providing sufficient structure for allowing efficient general purpose implementations". Fodor and Pylyshyn present arguments in favour of symbolic processing as a superior model in [39] and elsewhere. Randall Beer [12] presents a strong view against conventional symbolic processing. He has argued that the basis of intelligence is to be found in the simple day to day tasks of coping with the world rather than deliberate reasoning and number crunching. A comparative study of strong to moderate views put forth by both camps in this persistent debate is presented by Bringsjord [15].

The continuing debate seems to point more at our insufficient understanding of natural intelligence than the superiority of either paradigm. However, even as the debate remains unresolved, problem solving using neural networks is an area of major research interest. Several practical problems have been successfully solved using artificial neural networks.

The neural versus symbolic debate apart, various diverse views persist within the neural network community itself. Among the neural network researchers the debate is holographic versus structured representations [140, 35] and homogeneous versus heterogeneous networks [140]. The dominant view evolving is that structured connectionist networks with different components having different architectures and mechanisms constitute a plausible model for efficient problem solving. Neural networks is not a panacea but provides efficient solutions to certain classes of problems and certainly large systems will comprise of different subsystems each solving one or more of a class of problems. This work attempts to develop network models for solving certain classes of practical problems which are hard to solve using conventional computers.

1.1 Artificial Neural Networks – An Overview

Artificial neural networks consist of a large number of simple interconnected computing elements. The computing elements correspond to the neurons and the interconnections

correspond to the synaptic junctions. Real neurons and synapses are complex and our knowledge regarding the brain is very limited. There is large diversity of opinion among the research community regarding their detailed functioning. The main objective of artificial neural networks being to develop alternative computing models, rather than brain modeling, a simplified view of the brain is adopted. Artificial neural networks are, at best, very poor approximations of the better understood parts of the brain.

In artificial neural networks the neurons are modeled as implementing a function referred to as the *activation function* of the neuron. Several functions have been used as activtion functions. Sigmoidal functions and linear functions saturating at a maximum value and a minimum value are commonly employed activation functions. Taking a simplified view of the synapses as conduits for information from one neuron to another, the neurons are connected using *weighted interconnections* where the weight of a link stands for the synaptic conductance. The output of one neuron gets multiplied by the weight of the link connecting it to another and then feeds in to that neuron. Each neuron accumulates inputs from all other neurons connected to it and then modifies its output activation by applying its activation function. This process, by which the different neurons in a network change their activations, is specified by an *activation rule*. Another aspect of artificial neural networks is "learning" or the capability to adapt to new situations. Studies by Hebb have shown that the brain changes its response by changing the synaptic strengths. In the context of artificial neural networks, this corresponds to the changing of the weights associated to the links. The scheme for modifying the weights is referred to as the *learning rule*. It may be pointed out that the activation rule and/or learning rule may be probabilistic. Following the terminology of Bart Kosko [87], we shall refer to the process of change in neuronal activations as *neuronal dynamics* and the process of change in interconnection weights as *synaptic dynamics*. In some situations, as in the case of some recent feedforward models, learning may involve the modification of network structure by adding/deleting neurons and interconnections. Such operations may also be viewed as synaptic dynamics. It is a generally held belief that computations in the brain result from neuronal dynamics and the synaptic dynamics is a relatively slower process resulting in adaptation and long term memory. The other important fac-

tor influencing the operation of an artificial neural network is the external inputs and/or the reinforcement signals which it receives from the environment. Thus, we may identify the structural and operational aspects of a neural network as

- a set of processing units called neurons each of which implements an activation function.
- weighted interconnections between these neurons.
- neuronal dynamics determined by the interconnection weights, activation rule and the external inputs.
- synaptic dynamics determined by the learning rule and the reinforcement signals.

Different interconnection structures, activation rules and learning rules give different network models with different problem solving capabilities.

1.1.1 Some Common Network Models

A major aspect distinguishing different network models and their capabilities is the interconnection structure. The interconnection structure has a major influence on the neuronal and synaptic dynamics. Based on the interconnection structure we may classify the networks as *feedforward networks*, *layered networks with lateral interconnections* and *feedback networks.*

Feedforward Networks A Feedforward neural network is characterized by unidirectional acyclic connections between the neurons. The information flow is unidirectional and the network computes a transformation of the inputs, i.e., the neuronal dynamics specify a mapping from the input space to the output space. By applying a suitable learning rule, the interconnection weights and the interconnection structure of such networks may be modified in order to arrive at the desired mapping. Learning in such networks is typically based on the reinforcement signals received from the environment on the basis of actual and expected outputs and is termed *supervised learning.* Usually

the learning rule defines a gradient system where the gradient considered is that of the difference in actual and desired outputs. Feedforward neural networks find applications in supervised pattern classification, forecasting, prediction, generating control signals, learning evaluation functions etc.

Layered Networks with Lateral Interconnections These networks have a layered structure with unidirectional connections from one layer to the succeeding layer. However, there are lateral inhibitory interconnections within a layer. This leads to competition between neurons within a layer resulting in one or a few winners. The activation rule has to be such that the winners emerge clearly and oscillations between different sets of winners do not occur. The winners in a layer for a given activity pattern of the previous layer are learned by modifying the interconnection weights between the layers. Typically, these weights are modified by a process of regularity detection without using any external reinforcement signals and such learning is termed *unsupervised learning.* Usually a variant of the competitive weight learning rule is employed. It is imperative that the synaptic dynamics lead to a stable configuration of the weights where the neuronal dynamics lead to the desired winners. These networks are employed for category learning, classification, feature detection, vector quantization etc.

Feedback Networks In feedback neural networks the interconnection pattern is cyclic leading to cyclic information flow. Such networks are employed in two modes - one where only the activation rule is applied and the other where both activation and learning rules are employed. The activation rule of a feedback network defines a nonlinear dynamical system where the variables are the neuronal activations. For productive computation, it is necessary to ensure that this system has convergent dynamics. Such networks are commonly employed for optimization, constraint satisfaction, pattern matching etc.

Learning rules are employed for modifying the interconnection weights so that a network having the desired neuronal dynamics evolves. The learning rule should be such that finally a stable state of the weights emerge and the resulting network has convergent neuronal dynamics. Such networks are used as associative memories, pattern classifiers

etc.

1.1.2 Computing Mechanism of Artificial Neural Networks

The computing mechanism of an artificial neural network based on neuronal and synaptic dynamics is significantly different from that of a conventional computer which is based on instruction sequences. In the case of artificial neural networks, the neuronal and/or synaptic dynamics result in a state of the network which represents the solution.

Consider a system where only the neuronal dynamics operate, i.e., the structure and interconnection weights are fixed. Since the activation functions may be nonlinear the network will, in general, constitute a nonlinear dynamical system where the variables are the neuronal activations. The interactions may be competitive, cooperative or mixed [52, 65]. In most cases the desired result of the computation is the state of the neuronal activations at which the dynamical system stabilizes. In some cases periodic limit sets of the system where the activations cycle through a set of states may also be of interest [41, 153]. In general, we may say that the computation performed as a result of neuronal dynamics is convergence to the limit set of the corresponding dynamical system. A chaotic dynamics will be of no use computationally, since no final solution results from it. Therefore, for productive computation using neural networks, it is essential that the neuronal dynamics be convergent. The limit points of a dynamical system can often be interpreted as a minimum energy state or a maximum harmony state where the process of computation settles. The limit sets of the dynamical system are determined by the interconnection structure, weights and the activation functions. It may be pointed out that in some network models, particularly, those with acyclic interconnections, the neuronal dynamics represent a time invariant function and the computation evaluates this function.

The synaptic dynamics of a network specified by the learning rule also constitute a dynamical system in which the variables involved are the interconnection weights. The objective of the synaptic dynamics specified by the learning rule is to arrive at a set of interconnection weights which would produce a desired neuronal dynamics. This

configuration of the weights constitute a distributed representation of the knowledge gathered by learning. The objective of learning rules where the interconnection pattern and the number of neurons get modified is also similar.

To summarize, the computational process of artificial neural networks is to arrive at a limit set or limit point of the relevant dynamical system. The network stores information in a distributed fashion in its interconnections and neurons. *Distributed representation* and *parallel relaxation* is the dominant computing mechanism in artificial neural networks. A major issue in designing a neural network is to *ensure convergence of the relevant dynamical system to a desired limit point or limit set.*

1.1.3 Problem Solving Domains of Neural Networks

Artificial neural networks have, in recent years, shown great promise as an efficient alternative computing model for solving various classes of problems. The problems solved using such networks include feature detection, pattern classification, pattern matching, vector quantization, category learning, speech processing, recognition, learning functional invariance in training samples, forecasting, prediction, learning evaluation functions, tracking, combinatorial optimization, constraint satisfaction, inferencing from partial information, associative storage and recall etc. Large neural systems may be viewed as consisting of different subsystems solving different classes of such problems. The problems solved by neural networks can be grouped into a few distinct classes. Iyengar and Kashyap [72] classify neural network problem solving as adaptive pattern recognition and classification, approximating mathematical mappings, combinatorial optimization and adaptive knowledge processing. Kohonen [84] identifies the classes of problems as parallel associative search and inferencing in knowledge data bases, pattern recognition, decision making and optimization. Hecht-Nielsen [57] identifies the information processing operations carried out by neural networks as mathematical mapping approximation, probability density function estimation, extraction of relational knowledge from binary databases, formation of a topologically contiguous and statistically conformal mapping, nearest neighbour pattern classification and categorization of data. Based on these ob-

servations neural network problem solving may be divided into the following four major categories.

- *Combinatorial Optimization.* Since Hopfield demonstrated the solution of travelling salesperson using neural networks, the neuronal dynamics of feedback neural networks has been successfully employed for solving various constraint satisfaction and optimization problems [130, 66, 61]. Some applications of neural optimization techniques include medical diagnosis by Peng and Reggia [127], rule matching in expert systems by Touretzky and Hinton [166] and weapons allotment by Tagliarini *et al.* [159]. A constraint satisfaction network has been employed for analogy retrieval [162].

- *Associative Memory.* Researchers have evinced keen interest in implementing associative memory using neural networks [35, 82]. Applications of neural associative memory include inferencing from partial knowledge, tracking, pattern classification etc.

- *Adaptive Pattern Recognition.* Various pattern recognition tasks have been successfully solved using neural networks. These include feature detection, pattern classification, category learning, clustering, vector quantization etc. The major models used for this purpose are the competitive learning models [82, 20, 141]. One of the earliest complete recognition systems built, was the cognitron models by Fukushima [44, 46, 45].

- *Approximating Mathematical Mappings.* This aspect of neural networks has been widely applied since the backpropagation rule [140] was popularized. Such systems have been employed for learning various control tasks such as driving a car, prediction tasks such as load forecasting, speech processing etc.

Artificial neural network, which is an alternative computing paradigm derived from an oversimplified view of the brain, does provide efficient and fast methods for solving various problems. However, there are several classes of problems in the above categories for which fast and efficient neural network solution techniques are not available. It

appears that, for obtaining efficient solutions for practical problems, new network models and new analysis and mapping tools will have to be explored.

1.2 The Thesis

In this work, we intend to explore ways and means of solving some classes of practical problems using artificial neural networks. In spite of the numerous problems which have been efficiently solved using neural networks, there are several problems of practical importance for which efficient network models are not available. Commonly used network models employing neuronal dynamics are limited to uniform/symmetric networks only. One way of achieving more versatile problem solving capabilities could be to harness the far richer computational power of asymmetric networks or networks with functional interconnections. Unfortunately these networks are notoriously difficult to analyze and no general methodology is known for analyzing arbitrary asymmetric networks. Analysis is an important factor in determining the general utility of a network. In this situation, two options present themselves. One option is to take special classes of networks, analyze them and find their problem solving capabilities. Alternatively one can take specific classes of problems and try to design and analyze networks for solving them. In this case the choice of the problem should be complex enough to warrant non-trivial neural networks, simple enough so that tractable networks are sufficient and general enough so as to cover a large number of applications. The present work takes this approach. Specifically, the following problems have been identified.

Combinatorial Optimization 0-1 integer programming problems (0-1 IPP) with inequality constraints, encountered in a large number of practical applications, cannot be efficiently solved using symmetric neural networks. In general, techniques such as those of Tagliarini *et al.* [159] and Mjolsness *et al.* [112, 113] may not be practical. Boltzmann machines used for solving 0-1 IPP with inequality constraints [60] are slow and their performance is dependent on the temperature schedule. Convergent asymmetric networks could provide an efficient alternative for approximately solving such problems. Such

networks would be very useful in several diverse applications such as packing, covering, scheduling, assignment, matching, system design etc.

Adaptive Pattern Recognition Various tasks in pattern recognition need to estimate the characteristics of appropriate distributions. However, existing neural networks suitable for this purpose employ synaptic dynamics and face several difficulties [19, 20, 82, 124, 141]. Networks employing only neuronal dynamics would provide efficient solutions to such problems. However, for this, nonlinear interconnections would be required. Such networks would benefit several applications in computer vision and image processing such as feature detection and image data clustering.

Approximating Mathematical Mappings A major difficulty, in learning an input-output mapping using feedforward neural networks, is the choice of a good network structure. Existing network growing algorithms are specially suited for binary valued functions and grow uniform networks [7, 61, 107]. An algorithm for growing nonuniform networks for continuous (real valued) mappings could result in smaller and/or sparser networks for such mappings and would be useful in applications such as learning utility functions, generating control signals etc.

The work presented in the thesis is an attempt to address the above problems. The broad aims are:

- Develop convergent asymmetric network models for solving 0-1 Integer Programming problems which cover a large number of combinatorial optimization problems of great practical utility.

- Develop networks with nonlinear interconnections (which can also be looked upon as a class of asymmetric networks) for estimating characteristics of distributions.

- Develop a method for growing a nonuniform feedforward network for approximating continuous mathematical mappings.

The other major application area of neural networks, namely associative memory, has not been addressed in this work. The interconnection weights of the network models developed in this work are obtained from the problem specification. The issue of learning the weights have not been addressed in this work.

As would be evident from above, the present work does not constitute a single project. The field of neural networks is evolving and general methodologies for neural systems are not yet well understood. As an observer of the history of science might suspect, general methodologies for a system based on something as complex as the brain may have to evolve from simpler special methodologies. It is the belief of the author that incremental extension of the domain of problems for which fast and compact neural solutions are available, is an important component of developing general neural network strategies. The work presented in this thesis is an endeavour in that direction.

1.2.1 Overview of the Work Done

Combinatorial Optimization

1. *The feasibility of mapping inequality constraints onto symmetric neural networks has been investigated.* It has been shown that, in general, mapping inequality constraints onto symmetric neural networks will require exponential number of extra neurons.

2. *The convergence of asymmetric networks at finite as well as infinite gain of the sigmoidal activation function has been investigated.* The global stability of the neuronal dynamics of asymmetric networks has been investigated with the help of a Lyapunov function. Based on this study a class of networks have been identified such that almost every trajectory converges to a limit point.

3. *A neural network model, called Resultant Projection Neural Networks, has been developed for 0-1 IPP with inequality constraints and linear objective functions.* The network is derived based on the principle of orthogonal displacements and projections. The effect of network parameters on the optimization process has been

studied and a probabilistic analysis of the network has been done with respect to the 0-1 knapsack problem.

4. *A uniform scheme has been developed for constructing a network from the mathematical specification of a 0-1 IPP.* The mapping scheme covers 0-1 IPP with linear and quadratic objective functions and equality as well as inequality constraints.

5. *A number of 0-1 IP problems arising in various application domains have been solved using the network.* The network has been employed for solving 0-1 knapsack, multidimensional knapsack, weighted set covering, job processing with deadlines, point pattern matching and redundancy allocation for reliability optimization.

Adaptive Pattern Recognition

1. *A two layered synchronous network for detecting multiple local peaks in a discretized distribution has been developed.* Each layer of the network has one neuron corresponding to a cell of the discretized distribution. One layer of the network accumulates evidence in favour of a peak from its neighbourhood. The second layer of neurons induces a competition among potential peaks so that distinct and reliable peaks emerge as winners.

2. *The peak detection network has been analyzed.* The dynamic nature of the interactions in the network has been studied. A method for computing the weights, such that the network converges with the desired winners (peaks), has been derived. It has been shown that the network employs adaptive smoothing and adaptive thresholding.

3. *The peak detection network has been applied for various image processing applications.* It has been employed for selecting prototypes from multidimensional histograms, detecting parameters of straight lines from the Hough space and multithresholding gray level images.

4. *A classification network has been developed for delineating clusters around the prototypes selected by the peak detection network.* The network architecture is similar

to that of the peak detection network. Classification is done taking into account the distribution in addition to the distances between the pattern and prototypes. This network together with the peak detection network has been employed for various clustering problems such as color segmentation, remotely sensed image segmentation, gray level image reduction and X-ray image segmentation.

Growing Non Uniform Feedforward Networks

1. *A greedy algorithm has been developed for incrementally growing nonuniform networks for approximating continuous mappings.* The network for realizing a given example set is grown by adding one neuron at a time to a (given) partial network. The addition is done based on a greedy estimate of the resulting decrease in error, estimated from the gradient of the error surface.

2. *The convergence of the algorithm has been investigated.* It has been proved that after almost every addition of a neuron, the error of the new network would be less than that of the previous network.

3. *The algorithm has been employed for growing networks for several examples sets.* The example sets have been constructed from known mathematical mappings including a two dimensional surface generated using sine functions and the transient capacitor charge in R-L-C circuit.

1.2.2 Layout

The thesis is organized as follows. A brief survey of neural network strategies and applications in the areas of combinatorial optimization, associative memory, approximating mathematical mappings and adaptive pattern recognition is presented in *chapter 2.* In *chapter 3,* the issue of mapping 0-1 IPP onto symmetric neural networks and the convergence of asymmetric networks are investigated. The method for constructing a neural network for 0-1 IPP programming with inequality constraints and the analysis of the resulting network are presented in *chapter 4.* A number of optimization problems solved

using the network are reported in *chapter 5*. The design and analysis of a network for detecting multiple local peaks and its applications constitute the contents of *chapter 6*. In *chapter 7* the classification network and its applications are presented. In *chapter 8* we present a greedy algorithm growing nonuniform feedforward neural networks for continuous mappings. *Chapter 9* constitutes the conclusion where we present a summary of the work covered in this thesis.