博士学位论文

BP 神经网络样本分组计策

Pattern Grouping Strategy For BP Neural Networks

作　　者：吕 迎 阳
指导教授：吴 伯 僖 教授
　　　　　徐 慎 初 教授

厦门大学物理系

1999 年 10 月

学校代码：10384

厦门大学博士学位论文

BP 神经网络样本分组计策

Pattern Grouping Strategy For BP Neural Networks

系（所，室）： 物 理 系
专　　　业： 凝聚态物理
研 究 方 向： 人工神经网络
博士生 姓名： 吕 迎 阳
指 导 教 授： 吴 伯 僖 教授
徐 慎 初 教授

一九九九年十月

中文摘要

1986 年 Rumelhart 等人[18]利用三层 BP 神经网络成功解决了异或(XOR)问题，从而彻底结束了 Minsky 的“感知机”一书[7]对神经网络领域的负面影响。此后，BP 模型作为通用函数拟合器[28,70]以其学习算法简单易行等特点在多个领域被广泛应用于函数逼近、模式分类、模式匹配等工作，成为应用最广泛的人工神经网络模型。

但人们在应用及研究中也发现了 BP 网络的一些问题，最主要的就是其学习过程收敛速度慢且易陷入能量局部极小点而导致学习不成功。对此人们提出了对 BP 模型的多种改进方案，主要有引进动量项、噪声项，采用自适应学习步长，改变网络结构，采用一些动力学处理技术等方法。然而这些方法都没能使 BP 网络从实质上摆脱这些弱点。同时相关的研究更表明 BP 学习算法本身并不是一种有效率的算法[78]。

本文的工作试图找到新的方法以弥补 BP 模型的主要缺点，使之有可能在处理复杂度较高的问题中也能得到应用。

我们详细考察了 2-2-1 零阈值 BP 网络学习 XOR 问题时的能量曲面(第三章)，认为学习收敛速度慢及易陷入能量局部极小点是 BP 模型的内禀缺陷。BP 网络的学习过程等效于在能量曲面上寻找全局极小的过程，而 BP 学习算法本质上是梯度下降法，因而继承了梯度下降法受陷于局部极小点的缺陷。另一因素是 BP 模型采用 S 形函数作为神经元的激发函数，而 S 形函数除在原点附近有一定范围的坡度外其余的地方都是梯度很小的平缓区域，导致了能量曲面以平缓区域为主的“地貌特征”，因此网络学习过程会经常遇到这些使梯度下降法变得极为缓慢的平缓区域。

BP 模型的内禀缺陷使得该模型对复杂度较高的问题的学习能力有限。要想

利用 BP 模型简便易行等特点，必须能够充分利用 BP 模型的有限的学习能力。而从待学习的样本集中找到更多的信息用以辅助 BP 网络的学习则是一个值得探索的方向。

传统的 BP 算法将待学样本集看作一个整体进行学习，没有体现人脑对复杂事物总是通过分解为局部才完成对整体学习的过程。并且当样本集较复杂时因其有限的学习能力而常常失败。为此我们引用分而治之的思想提出了样本分组学习方案：将样本依照其特点预分成子组，然后用 BP 网对各个子组轮番学习。

模拟实验(第四章)表明，我们所提出的分组方案对对称性问题及 XOR 问题的学习在较大的学习步长范围内显著地提高了学习成功率，并改善了 BP 网学习能力敏感于步长的缺点。

在第五章，我们通过对隐层神经元在学习过程中的运动特性的分析，解释了分组学习方案提高学习成功率的原因，并探索了 BP 网络初始化的小值原则及其与神经生物学中对应原则的联系。

最后，我们利用分组学习方案训练了多个识别对称性的 BP 网络，并成功地构造了一个可自动识别不同位长二进制码的对称性的专家系统。

Abstract

Studies about the BP neural network indicate that vulnerable to local minima and to flat regions of the energy function is its inherent drawback which reduces it learning abilities when dealing with complex tasks. We think feeding additional information of the task patterns to the BP network to guide its learning is a method to improve its learning ability. We proposed the pattern grouping strategy(PGS) for training based on the idea of "divide and conquer": Divide the patterns into sub-groups according to their properties and train the network alternatively with each sub-groups. Simulations show that our PGS training is more efficient than normal BP training for identifying symmetrical patterns and XOR problems over a wide range of parameters' values. We explained the advantage of PGS by analyzing the movements of hidden neurons during training, and also explained the small value rule when initializing the connection weights. Finally we constructed a mixed expert system for symmetrical pattern identification by combining several BP networks trained successfully with PGS.

Contents

Acknowledgment

The author of this thesis would like to express deep appreciation to Professor Boxi Wu for his scientific guidance and patient encouragement throughout this research. The thesis would be impossible to finish without his guidance and encouragement.

The author would also like to express appreciation to Professor Shenchu Xu, who provided a lot of substantial advice and suggestions for this thesis. Appreciation is also expressed to Professor Zhenxiang Chen for his helps in many aspects during this research.

The author also thanks colleagues in our research group as well as all persons who have given the author helping, concerning, and encouraging.

Chapter I Introduction

1.1 Development of Artificial Neural Network

Studies of artificial neural networks (ANN), an international research frontier of modern science and technology, arises accompanying with the development of biology, physiology, psychology, electronic engineering, mathematics, and physics. The very great and highly efficient processing power inherent in biological neural systems has attracted the intensive studies on the structure and behavior of the brain, which uses a very complex network of interconnected processing elements--called neurons to process information, such as perception, organization, transferring, storing,...etc. The primary object of ANN is to mimic the neuron in terms of the structure and the processing. Because of the complex nature of ANN, the study is a very tough project. Development of ANN has experienced four historical periods[1].

(1) The Initial Period (1943--1969)

1943: McCulloch and Pitts published a well-known paper[3] "A Logical Calculus of the Ideas Immanent in Nervous Activity" showing the first mathematical model (called M-P model) of a neural cell, and proposed a general theory of information processing based on networks of M-P models.

1949: The book "The Organization of Behavior" by Hebb[4] gives the method(known as Hebb’s learning rule) of weight modulation. Today the Hebb's learning rule is still one of main learning rules for many types of artificial neural network models.

M-P model and Hebb’s rule formed the theoretic foundation for the artificial neural networks.

1958: Rosenblatt gave the first real ANN model--"Perceptron"[5] and introduced an iterative algorithm for constructing the connection weights. This two-layered system was actually a weight learning machine for simulation of sensory information processing. Rosenblatt’s group also proved their algorithm’s convergence[6]. Their success had indicated great potential of applications of ANN and triggered the first upsurge of ANN researches.

(2) The Ebb Period (1969--1982)

1969: After several years of intensive studies, Minsky and Papert took a pessimistic view point in their book "Perceptrons"[7] showing that the simple perceptron proposed by Rosenblatt could not be able to solve the exclusive OR(XOR) problem and believing that introducing additional layer(s) could enhance the processing ability but the studies of related learning algorithm may be very difficult. Under their influence coupling with the prosperous of artificial intelligence, some governments stopped projects in ANN and many researchers gave up their studies in this field. However, there were still some researchers continued their exploring and got very important results. Some of these works are:

1972: Concept of memory in ANN was proposed by Kohonen[8] and Anderson [9].

1973--1982: Grossberg and Carpenter proposed the adaptive resonance theory(ART)[10,11] in which they introduced the short term memory and long term memory.

1980--1982: Based upon the theories of biologic vision Fukushima set up a vast multi-layered network called "Neocognitron"[12], in which the concepts of competitive learning was utilized.

1982: Kohonen established self-organizing map model[85].

(3) The Resurrect Period (1982--1986)

1982: Physicist J.J.Hopfield published an important paper in which a model of whole-connected network(named Hopfield network) was proposed[13]. He also utilized theories of Ising model(a model of lattice of magnetic spins, which is quite similar to Hopfield network system) to study how such a network can store and retrieve information. Another feature of this model is that it can be easily realized with VLSI.

1984: Bell Lab of AT&T made the first ANN chip utilizing the Hopfield theory.

1984--1987: Hopfield set up the energy function for determination of the network stability and successfully solved the famous TSP problem which is of NP-complete, and thus started a new age of neural networks[14--17].

1986: Rumelhart D.E. and McClelland J.L. further developed the back-propagation algorithm for multi-layered feed-forward networks and solved the XOR problem[18].

Thus the Minsky's shadow on ANN was completely removed, and the renaissance period began.

(4) The Upsurge Period (1986--1990s)

1987: The First International Conference on Neural Networks was held in USA with more than one thousand persons in many disciplines such as biology, electronics, computer science, cybernetics, physics, to attend. Then the study of ANN extended to Europe, Japan, and China.

From than on, research works in the field of ANN including the models, algorithms, implementations, architectures, and applications had been growing up quickly. Many new models, algorithms, as well as theories such as cellular neural networks[19-22], genetic algorithms[23,24], EM algorithm[25,80,81], information geometry[25,86], etc. had been proposed.

Also, some difficult problems were found in the models, algorithms, nonlinear theories, approximate approach, etc.

There were many hundreds of papers published each year. References[26-41] are some of them published in 1989--1998. In China, studies in the field of ANN have attracted a great deal of attention. There are many ANN books and monographs[42--46], and one National Conference on ANN each year has been convened. Several hundreds of papers appear in several Chinese journals. References[47-66] are the selected papers in the period of 1998.

1.2 Problems of BP neural network model

As an answer to Minsky's XOR criticism, Backpropogation(BP) neural network model has a simple training procedure, and has shown high potential in near-term applications. After exhibiting surprisingly "intelligent" in the NETtalk system by Sejnowski[67], BP model has been widely implemented in enormous number of applications such as handwritten signature verification[68], environment data processing[69], etc. Been one of the most studied neural network models, BP networks with one hidden layer using arbitrary squashing functions has been proved theoretically to be an universal approximator which can approximate virtually any function of interest to any desired degree of accuracy[70]. Another advantage of BP was said to be its ability to store numbers of patterns far in excess of its built-in vector dimensionality[71].

However, implementations and studies also have shown out some drawbacks of the BP model[72,73,87,75,77]. Learning with BP algorithm is actually a process of searching global minima of an energy function with the gradient descent technique. When the energy function has local minimum(or usually minima), the searching process may be trapped and the learning fails. Even if the energy function has no local minimum[84], the searching process may also become extremely slow when it hits special part of the energy function where the gradient has very small or zero module.

Many works had been done in an attempt to cope with these blemishes and four kinds of methods had been proposed:

1) Basic modifications on the BP learning algorithm. In the original BP algorithm, modification of the weights vector in *i*th learning cycle can be expressed as: $\Delta \boldsymbol{W}_{(i)} = -\boldsymbol{h}\nabla E$, E is the energy function, $\boldsymbol{h}$ is a parameter named learning rate.

   i). Adding a momentum term: $\Delta \boldsymbol{W}_{(i)} = -\boldsymbol{h}\nabla E + \boldsymbol{k}\Delta \boldsymbol{W}_{(i-1)}$ so that the modification of the weights vector in a learning cycle will be partially kept in next cycle. Thus the learning process will act as a rolling ball on the energy surface: when hitting a spot where the gradient is zero it will not stop immediately but keeps moving for some distance.

   ii).Adding noises to the teacher's signal. Almost all the BP networks utilize supervised learning, i.e. in learning process, a pattern accompanied with a desired output is feed to the network, and the desired output is called teacher's signal. The teacher's signal is a main fact to determine the shape of the energy surface. When small noises are added to the teacher's signal, the energy surface will be slightly distorted, and this distortion may help the learning process escaping from local minima or area with small gradients. This can be thought as a slight earthquake making the rolling ball bouncing out off a small low-lying.

iii) Using variable learning rate and/or momentum factor. These techniques are helpful in increasing the speed of convergence of the learning process.

These above modifications on the BP learning algorithm can only overcome shallow local minima and small area of flat regions of the energy surface.

2) Adoption of techniques based on dynamic system theories. Two typical examples are the terminal attractor[75] and the terminal repeller[76].

When using the terminal attractor algorithm, the learning rate takes such a complicated form:

$\boldsymbol{h} = \Omega(E)/\|\nabla E\|^2$ $\Omega(E)$ is a non-negative continuous function of energy E.

With properly selected $\Omega(E)$, whenever the learning process approaches a local minimum $\boldsymbol{h}$ will become so large that the learning trajectory takes a great jump and the system is likely to escape from that local minimum, while near the global minimum points $\boldsymbol{h}$ will remain reasonably small to allow convergence. Unfortunately, the direction and extension of such jumps are not guaranteed to lead closer to the global minima. The trajectory may get stuck into the same local minimum basin, or jump into the basin of another minimum.

In the work of reference [76], a new energy function Œ(E, E^*,$\boldsymbol{W}$, $\boldsymbol{W}^*$) is defined, E is the original energy function, $\boldsymbol{W}$ is the connection weight vector and $\boldsymbol{W}^*$ is a vector close to the starting point of the current searching task, E^*=E($\boldsymbol{W}^*$). In any searching task, normal BP algorithm is used to find a minimum of Œ, and this minimum point is the $\boldsymbol{W}^*$ of next searching task. The new energy function Œ is constructed in a special way to bear two features:

i) for any part in the $\boldsymbol{W}$ space where $E \geq E^*$, the point $\boldsymbol{W}^*$ is the only repeller which has effects over the whole space;

ii) for any part in the $\boldsymbol{W}$ space where $E < E^*$, Œ has a shape very close to that of E and has the same critical points as E.

Therefore in each of the learning tasks, in regions of type i) the system will be pushed away from point $\boldsymbol{W}^*$ and tunnels through the current high E region. Once the system reaches a region of type ii), a nearly normal BP learning with energy function E begins. In one-dimensional cases, this technique can totally eliminate the risk of local minima and guarantee the convergence to the global minimum. However, for multi-dimensional cases such as neural network system, there exists new risk that the repellers push the system to a "death" direction along which $E \geq E^*$ always satisfied and the system drifts away to the infinity.

3) Adjusting the structure of the network. Changes of the network structure(number of neurons, neuron activation functions, etc) may significantly change the shape of the energy function and may help the learning process to over come local minima and saddle point areas. An example of changing the number of neurons is the work of Hirose[73]. He added a new neuron whenever the learning process got stucked in a local minimum or flat region, and restart the learning. New neurons kept being adding to the network until the learning process gave a reasonable result. Then he selected and removed a neuron and start the learning again in an attempt to cut off redundant neurons.

Yang et al[74]. implemented the techniques of homotopy in BP learning by using a series of functions as the neuron function: $f_l(z) = lz + (1-l)s(z)$, here $l \in [0,1]$, $s(z)$ is the desired final form of neuron function. The learning begins with $l_0 = 1$, and upon finishing, the learning restarts with $l_1 < l_0$. Such a procedure repeats, each with a new smaller l, and when l reaches 0, the final network comes out. At the beginning of the learning process, the energy function may have very few minimum points due to the simplicity of neuron function. The speed of the transformation of neuron function can be controlled by selecting suitable series of l. Thus to some extent this method can overcome the local minima. Fig.1-1 is a schematic diagram showing such a process. However, this method does not always work since, given the complexity of the energy function, the global minimum points of the energy function always form curves or even regions, and the learning procedure can not guarantee that currently reached

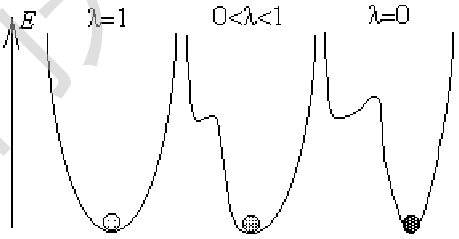

Fig.1-1

global minimum point locates near the global minimum regions related to next l. In other words, the global minimum regions related to next l may appear near the other end of the current global minimum region containing the currently reached global minimum point, as shown in Fig.1-2, and the local minima risk appears again.

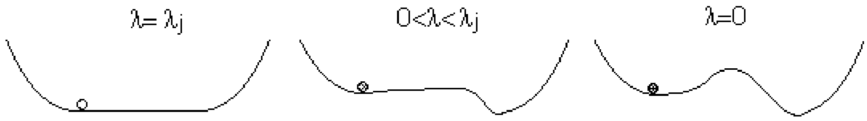

Fig.1-2

4) Methods that modidy the target pattern sets.

An example of this type is in reference [77]. This work actually utilizes the techniques of homotopy as in [74], but modifications are made to the patterns instead of neuron function. So the above analysis in 3) about [74] is also suitable to [77].

The work of reference [45] tried to improve the efficiency of the BP algorithm in a way similar to human's habit: grasp the main feature of an object first, and then pay attention to details. In this work all the *n* patterns are divided into *m* groups, then create *m* new patterns by averaging each group's patterns and let the network learn these new patterns. On completion let m=min(mk, n) and repeat the above grouping-averaging-learning program, here k>1. Finally m=n and all the original patterns are learned. Although such a program lacks mathematical proves, simulations had indicated its obvious improvement on the learning speed and success rate. This method deserves further investigations.

1.3 Works of This Thesis

Carefully reading of related books and papers as well as our own studies had help us form the opinion that vulnerable to local minima and to flat regions are inherent drawbacks of the BP neural network model and can not be totally over come with simple methods. To make it worse, some other researchers have proven theoretically that training a three-node BP network is NP-hard. This implies that BP algorithm itself may not be an efficient training method[78]. Hence the up-limit of the BP model's learning ability may be under suspecting, especially when the task is of certain complexity. In fact some researchers had had to use a combination of quite a number of BP networks, each of which took a small part of the whole task, to undertake the desired work[89].

In order to use BP network of reasonably size to undertake a task of relatively high complexity , other works besides modifications of the BP algorithm should be done, such as feeding additional helpful information about the target patterns to the network to guide its learning process, as done in reference [45].

Consider the fact that when the task is relatively small those mentioned above blemishes of BP model do not overwhelm the model's advantages such as easiness and relatively fast, and that BP network tries to learn the whole set of patterns in a single session. It is nature to resort to what has long been adopted by human: divide and conquer.

Besides the work of reference [88], another example of divide and conquer is the work of R.A. Jacobs *et al*[79,80,81], They set up the system of mixtures of local experts, which was composed of several neural networks to treat different subtasks, and gained some inspiring results.

Based upon background mentioned above, the first part of this thesis is to study the implementation of "divide and conquer" idea in BP neural network. We name our method of grouping patterns as pattern grouping strategy (PGS).

We made comparison on the success rates of PGS learning and that of normal BP learning in four systems. We had also examined the influence of learning rate on the success rate of PGS learning so as to find training conditions favored to PGS learning.

In the second part of this thesis we analyzed the activities of the hidden neurons of the BP model during learning process and used related reasoning to explain the PGS learning. We also explained the small value rule in initializing the BP network.

In the third part we combined most of our works to construct a mixed expert system which can recognize symmetrical binary patterns of 7--14 bits.

The difference between our work and those of reference [88] and R.A. Jacobs is: they use several sub-networks to treat different subsets of the whole crew of patterns; we use those subsets of patterns to train a single BP network, and finally make this single network competent to the whole task.

Chapter II Related Basic Knowledge of Neural Networks

Section 1. Feed Forward Neural Network

2.1 Artificial Neural Network:

To acquire the ability of intellectualized information processing, researchers had opened up, based on the hints of biological nervous systems, the research area of artificial neural network(or neural network for brevity).

A neural network is a network system formed by connecting together a collection of processing units. Each processing unit is called a neuron, and the connection way between two neurons is called a synapsis or just a connection. The strength of a connection between two neurons is named the connection weight or weight.

1. Neurons: A neuron receives signals from other neuron(s) via the connections, and then performs some kind of computation and passes the results as its output to other neurons. A type of commonly used neuron has the following mathematical model:

$$O_j = f(\sum_i w_{ji} x_i - q_j)$$

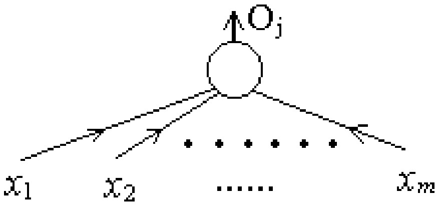

Fig.2-1

Here x_i is the output of the ith neuron, O_j and q_j are the output and threshold of the jth neuron, w_{ji} denotes the connection weight from the ith neuron to the jth neuron. Function $f(\ )$, which can be chosen from wide variety of types, determines the output(or named the state) of the neuron and is called the activation function.

2. Weights: The weights used on the connections between neurons have much significance in the working of the neural network and the characterization of a network. All the information memorized by a neural network are actually stored among the