

# 前向神经网络拓扑结构和权值的进化

李彦民<sup>1</sup>, 白本督<sup>2</sup>, 李映<sup>2</sup>

(1. 空军工程大学 导弹学院, 陕西 三原 713800; 2. 西安电子科技大学, 陕西 西安 710071)

**摘要:**基于进化规则(Bvolutionary Programming)的方法,提出一种进化前向神经网络的新算法。该算法能同时进化网络的拓扑结构和连接权值(包括阈值),产生非常紧凑的网络结构,并且由于其全局搜索能力能够避免结构的局部最小。仿真结果证明了该算法的有效性。

**关键词:**进化规则;前向神经网络;进化

**中图分类号:**TT391.4 **文献标识码:**A **文章编号:**1009-3516(2001)04-74-76

多层前向网络是目前应用最为广泛的神经网络模型之一,一直受到人们的普遍重视,在模式识别、回归分析等许多方面获得了很大的成功。

用多层前向网络解决问题的过程也即设计适当的网络结构和构造相应的网络权值的过程。目前人们在设计网络结构多采用探测性的删除算法或构造算法,权值是通过采用BP算法或其它一些基于梯度的算法得到的。但是这样的探测方法既不能避免BP算法的固有不足,即对于特定问题的网络结构难以确定及网络训练慢且经常陷于局部收敛,也必将使训练出的结果网络的优劣依赖于具体的探测过程,并且一旦当前探测过程失败,目前得到的中间结果也不能继续沿用。近几年来人们提出用进化算法来设计神经网络<sup>[1]</sup>。进化算法是从自然进化的思想和理论发展而来的一类基于群体的随机搜索算法。包括遗传算法、进化规划和进化策略。所有这些算法的一个重要特征是基于群体的搜索策略,它们不依赖梯度信息,比传统的基于梯度的搜索方法更不易陷入局部最小。

本文在Fogel的进化规划基础上提出一种新的算法来进化前向网络。采用进化规划而不是遗传算法来进化网络有两个优点:一是进化规划采用实数编码方案,这就省略了二进制数编码方案复杂的网络编码表示与解码计算,更适合表示和操作网络;二是进化规划只用变异算子,避免了遗传算法中的交叉算子在操作时很容易破坏已进化出的网络结构。

## 1 算法及网络模型

### 1.1 网络模型

在算法的实现中,对可进化的神经网络的类型约束很小,网络不必严格分层或全连接,并且隐结点可以具有不同的变换函数。本文所进化的网络结构如图1所示。

### 1.2 算法实现

算法框图如图2所示。

#### 1.2.1 编码模式

在算法中,网络的结构和连接权值用一个 $(m+N+n+1) \times (m+N+n+1)$ 的矩阵表示,一个矩阵对应着一个网络。这里讨论的是前向神经网络,因此只有矩阵的下三角有非零元素,用这样的表示模式可以很容易地进行结构变异。

图1中X和Y分别为输入和输出,m和n为输入结点和输出结点,N为隐结点。

$$x_i = X_i, 1 \leq i \leq m$$

$$\begin{aligned}
 net_i &= \sum w_{ij}x_j, m < j \leq m + N + n \\
 x_j &= f(net_j), m < j \leq m + N + n \\
 Y_i &= x_{i+m+N}, 1 \leq i \leq n
 \end{aligned}$$

其中  $f$  为 sigmoid 函数,  $f(z) = \frac{1}{1 + e^{-z}}$

1.2.2 适应度评估和选择机制

在算法中,每个个体的适当度用  $f=1/E$  来表示,  $E$  为网络的能量函数:

$$E = \frac{1}{T \times n} \sum_{i=1}^T \sum_{i=1}^n (Y_i(t) - Z_i(t))^2 \quad (1)$$

其中  $Y_i(t)$  和  $Z_i(t)$  是模式  $t$  的第  $i$  个结点的实际输出和期望输出,  $n$  和  $T$  分别是输出结点数目和输入模式数目。

在算法中我们采用退火选择机制<sup>[2]</sup>,即在当前的群体中以下列概率选择个体  $x_i$ :

$$P(x_i) = \frac{e^{f(x_i)/T_k}}{\sum_{i=1}^N e^{f(x_i)/T_k}} \quad (2)$$

其中  $f(x_i)$  为个体  $x_i$  的适应度,  $N$  是群体规模,  $\{T_k\}$  是趋近于 0 的温度控制序列。

1.2.3 权值训练

对权值的训练我们采用改进的 Aploex 算法<sup>[3]</sup>。这个算法的搜索过程把随机性和确定性结合起来,既能避免局部最小,又能加快收敛速度。算法描述如下:

若第  $k$  次迭代:  $\Delta E(k) = E(k - 1) - E(k - 2) \quad (3)$

若  $\Delta E(k) < 0$ , 权值按原相同方向前进一个  $\delta$  长度,若结果不理想,再回退半个  $\delta$ ;

若  $\Delta E(k) > 0$ , 则  $W_{ij}(k) = W_{ij}(k - 1) + \delta_{ij}(k) \quad (4)$

其中  $\delta_{ij}(k) = \begin{cases} -\delta, & \text{以概率 } P_{ij}(k) \\ +\delta, & \text{以概率 } 1 - P_{ij}(k) \end{cases}$

概率  $P_{ij}(k)$  为波尔兹曼分布:

$$P_{ij}(k) = \frac{1}{1 + e^{-\frac{C_{ij}(k)}{T(k)}}} \quad (5)$$

其中  $C_{i,j}(k)$  表示权值改变与误差改变的相关量,  $C_{ij}(k) = \Delta w_{ij}(k) \cdot \Delta E(k)$ ,  $T(k)$  为温度。

1.2.4 结构变异

在算法中,结构变异算子为结点删除和结点增加,并且结点删除总是先于结点增加,只有当删除结点不能产生一个好的子代时,才增加结点。这样顺序变异的目的是为了保证网络规模尽可能的小。而结点增加是通过“细胞分裂”<sup>[4]</sup>来实现的,即在所有现存的隐结点中均匀地随机选取一个结点进行分裂,分裂后的结点具有下列连接权值:

$$\begin{aligned}
 W_{ij}^1 &= W_{ij}^2 = W_{ij}, i \geq j \\
 W_{ki}^1 &= (1 + \alpha) W_{ki}, W_{ki}^2 = -\alpha \cdot W_{ki}, i < k
 \end{aligned} \quad (6)$$

其中,  $W$  是原来结点  $i$  的权矢量,  $W^1$  和  $W^2$  是分裂后的两个结点的权矢量。

2 模拟结果

采用本文提出的算法对  $N$  位奇偶性问题来进化前向网络,  $N$  取 4 ~ 8, 训练样本数目为  $2^N$ , 表 1 给出了实验参数。对每一个  $N$  ( $N$  取 4 ~ 8) 位的奇偶性问题分别运行 10 次, 实验结果如表 2 所示。

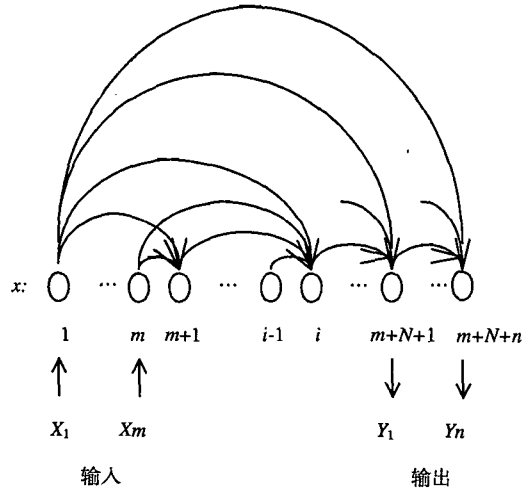


图 1 全连接的前向神经网络

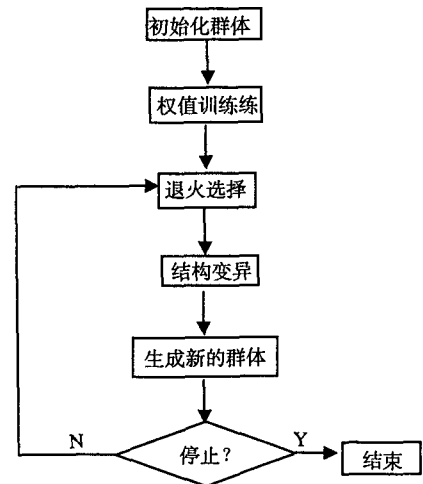


图 2 算法框图

表1  $N$ 位奇偶性问题的实验参数

群体规模	20	权值训练步长	0.005
初始隐结点数	$2 \sim N$	权值训练迭代次数	100
最大结点数	$2m + n + 1$	Aploex 中的温度设置	100
		温度调整参数	10

表2  $N$ 位奇偶性问题的实验结果统计

$N$ 位奇偶性问题		4位	5位	6位	7位	8位
隐结点数	最小值	2	2	3	3	3
	最大值	3	4	4	5	6
	平均值	2.5	2.7	3.3	3.6	4.8
网络误差	最小值	$5.4 \times 10^{-5}$	$3.9 \times 10^{-4}$	$3.1 \times 10^{-4}$	$4.7 \times 10^{-4}$	$6.2 \times 10^{-3}$
	最大值	$6.3 \times 10^{-3}$	$7.5 \times 10^{-2}$	$8.2 \times 10^{-2}$	$8.2 \times 10^{-2}$	$9.4 \times 10^{-2}$
	平均值	$7.1 \times 10^{-4}$	$3.2 \times 10^{-3}$	$4.7 \times 10^{-3}$	$2.9 \times 10^{-2}$	$1.5 \times 10^{-2}$

从计算机模拟的结果中可以看到,采用本文提出的算法能进化出非常紧凑的网络结构,并且随着问题规模(即  $N$  值)的增加,由算法进化出来的网络规模增加得却很缓慢,对于 8 位奇偶性问题可以找到只有 3 个隐结点的网络结构。但由于其搜索空间很大,对位数较大的奇偶性问题需要较长的计算时间。

### 3 结论

本文在进化规划的基础上提出一种进化前向网络拓扑结构和权值分布的新算法。该算法采用结点删除先于结点增加的结构变异顺序和结点分裂等方法,以保证网络规模尽可能的小而泛化能力尽可能的强。同时该算法对于可行的网络结构限制条件很少,因而对不同网络的搜索空间很大,并且由于其全局搜索能力,可以避免结构的局部最小。实验结果验证了该算法的有效性。

#### 参考文献:

- [1] Maniezzo V. Genetic Evolution of the Topology and Weight Distribution of Neural Networks[J]. IEEE Trans. Neural Networks, 1994, 5(1): 39-53.
- [2] 张讲社. 整体退火遗传算法及其收敛充要条件[J]. 中国科学(E辑), 1997, 27(2): 154-164.
- [3] 徐立新. 神经网络训练的一种随机方法[J]. 哈尔滨工业大学学报, 1996, 29(3): 85-87.
- [4] Werbos P J. The Roots of Backpropagation: From Ordered Derivatives to Neural Networks and Political Forecasting[M]. New York: 1994.

## Evolution of Multilayer Feedforward Networks

LI Yan-min<sup>1</sup>, BAI Ben-du<sup>2</sup>, LI Ying<sup>2</sup>

(1. The Missile Institute of the Air Force Engineering University, Sanyuan 713800, China; 2. University of Electronic Science and Technology of Xi'an, Xi'an 710071, China)

**Abstract:** Based on evolutionary programming, a novel algorithm for designing the topology and weight distributions of feedforward networks is proposed. This algorithm can evolve network architectures and connect weights (including biases) simultaneously and produce very compact networks. It can escape from structure local minima due to its global search capability. The effectiveness of this algorithm has been confirmed by the simulation results.

**Key words:** evolutionary programming; feedforward network; evolution