

文章编号:1000-582X(2007)04-0091-06

基于改进遗传算法的BP神经网络自适应优化设计*

柴毅,尹宏鹏,李大杰,张可

(重庆大学自动化学院,重庆 400030)

摘要:BP(Back Propagation)神经网络在网络训练中存在着局部最优问题,其算法收敛过慢、局部收敛不理想,影响其工作性能.针对以上不足以及传统神经网络设计规模庞大等问题,提出了一种由EGA(改进的遗传算法)确定网络拓扑结构和训练网络的方法,该方法通过实数编码、自适应多点变异等操作有效地优化了网络拓扑结构和网络参数,从而有效缩小了网络规模 and 提高了BP网络训练的速度以及收敛的有效性.最后结合了番茄常见病害诊断的实例说明了此方法的可行性.

关键词:改进遗传算法;BP神经网络结构;多点自适应变异;病害诊断

中图分类号:TP391

文献标志码:A

BP型神经网络是一种多层前馈型网络,它是目前应用最广泛且使用最成熟的神经网络模型之一.通过深入研究发现BP算法中存在两个主要问题:一是从数学上看多层感知器算法是一个非线性优化问题,由于采用了传统的无约束最小化方法来实现误差函数的极小化,因此不可避免地存在有局部极小问题;二是神经网络的设计主要依据设计者的经验在大样本空间反复实验来进行选取,尚无理论上的指导.因此在神经网络的初始连接权以及网络结构的选择上具有很大的随机性,很难选取具有全局性的初始点,因而网络求得全局最优的可能性小.

遗传算法GA(Genetic Algorithm)是基于生物进化原理的一种具有鲁棒性的自适应优化方法^[1].遗传算法遵循通过基于问题样本适应度函数对初始群体选择、交叉和变异操作,来指导学习和确定搜索的方向.由于采用种群的方式组织搜索,所以它可以在全局解空间内的多个区域内寻求最优解,而且特别适合大规模并行处理.所以遗传算法和BP网络的结合正好优势互补.文献[2-4]表明,遗传算法和神经网络的结合对于寻求全局最优解其效果要优于单个的使用遗传算法或神经网络.但传统的遗传算法存在着收敛过慢,采用二进制编码导致的编码过长、计算量过大等问题.本文在传统的遗传算法的基础上通过引入实数编码、有

竞争的选择继承、多点交叉、多点自适应变异等操作,充分结合遗传算法和BP神经网络的长处提出了一种使用遗传算法在全局解空间内对BP神经网络拓扑结构和网络参数进行自适应的动态调整,从而获得网络的最优设计的新方法.文章最后以番茄病害诊断系统为仿真实例,验证了该方法的有效性.

1 问题的提出

由于遗传算法的搜索不依赖梯度信息,也不需要求解函数可微,只要求解适应度函数在约束条件下可解.并且遗传算法具有全局搜索的特性,用遗传算法优化神经网络连接权和网络结构,可以较好地克服BP神经网络的初始权值的随机性和网络结构确定过程中所带来的网络振荡,以及网络极易陷入局部解问题,并且有效提高神经网络的泛化能力.因此,利用遗传算法全局性搜索的特点,寻找最为合适的网络连接权和网络结构的方法来改变BP算法依赖梯度信息的指导,从而达到对网络结构和网络初始连接权值、阈值的最优配置.

在本文中引入了开关系数K,表示神经网络节点间的连接关系.其取值为1或0,分别表示两节点间有否连接.遗传-神经网络的优化问题数学描述如下:

* 收稿日期:2006-12-22

基金项目:国家863计划资助项目(2001AA113520;2002AA243031)

作者简介:柴毅(1962-),重庆大学教授,博士,研究方向为人工智能与专家系统,数字信号处理与模式识别.

(Tel.):023-65106464;E-mail:Chaiyi@cqu.edu.cn.

$$\min E_1(w, v, \theta, r) = \frac{1}{2} \sum_{k=1}^{N_1} \sum_{t=1}^n [y_k(t) - \hat{y}_k(t)]^2 \quad (1)$$

s. t $w \in R^{m \times p}$, $v \in R^{p \times n}$, $\theta \in R^p$, $r \in R^n$

其中 E_1 为网络总的误差, $y_k(t)$ 为教师信号, $\hat{y}_k(t)$ 为网络实际输出

$$\hat{y}_k(t) = f \left\{ \sum_{j=1}^p v_{jt} \cdot f \left[\sum_{i=1}^m w_{ij} \cdot x_i(t) + \theta_j \right] \right\} + r_t \quad (2)$$

设 $E_2 = \frac{1}{N - N_1} \sum_{k=N_1+1}^N \sum_{t=1}^n [y_k(t) - \hat{y}_k(t)]^2$ 为检测样本平均均方误差. 表示对网络输出数据可靠性的估计. E_2 小, 网络输出的可靠性大一些, 反之, 网络输出就不大可靠.

在神经网络的应用过程中, 由于被逼近样本的性质不能精确知道, 因此即使在网络误差 E_1 为零的条件下, 也未必能保证 E_2 达到要求. 往往会出现 E_1 非常小, 而 E_2 却无法达到要求. 这就是所谓的“过拟合”现象, “过拟合”现象直接影响网络的泛化能力, 使得网络最终失去实用价值. 所以算法的设计应当使得 E_2 小于一个给定误差 ε , 满足网络输出的可靠性.

2 算法的设计

本算法通过使用 EGA 在由 BP 网络初步确定的基本解空间上(网络连接权和神经元阈值的取值范围), 通过对基因的选择, 交叉变异操作. 对样本个体不断择优进化, 直至进化 K (总的进化代数由初始时给定) 代后, 选取个体中适应度最大的个体来确定网络的结构和网络的初始权值和阈值.

1) 基因编码

基因编码正像遗传基因代表了 DNA 中的必要信息一样, 它在将问题参数表示成一串有意义的符号的过程中代表了有关这一问题的必要信息. 本文中编码生成的码串由开关系数编码、权重系数编码两部分组成. 开关系数编码表示隐含层神经元与输入输出神经元的连接状态, 开关系数编码串长 l_1 可由初始隐含层神经元的个数来确定. 而实际隐含层神经元的个数由开关系数编码中 1 的个数确定. 权重系数编码表示网络的连接权, 采用浮点数编码, 串长 $l_2 = m \times l_1 + l_1 \times nl_2$ (其中 m 为输入节点的个数, n 为输出节点个数). 编码按一定的顺序级联成一个长串, 每个串对应一组网络结构、连接权.

2) 基本解空间的确定

采用三层 BP 网络来初步确定基本解空间(网络连接权和神经元阈值的取值范围), 首先设定网络的训练次数以及网络的训练误差 ε_1 , 输入训练样本进行

训练, 之后再输入检测样本得到误差 ε_2 , 在误差 ε_1 和 ε_2 都比较满意时, 把连接权值中的最大值和最小值分别记为 u_{\max} 和 u_{\min} , 以该区间 $[u_{\min} - \delta_1, u_{\max} + \delta_2]$ (其中 $\delta_{1,2}$ 为调节参数) 作为连接权的基本解空间.

3) 初始化样本群体

该步骤的关键是设定群体规模, 即基因编码组合数目. 群体规模作为遗传算法的主要控制参数之一, 对于遗传算法的效能的发挥是有很大影响的. 在本文中初始群体由 L 个个体构成, 每个个体由两部分组成, 第一部分是串长为 l_1 的 0-1 串表示初始开关系数; 第二部分是区间 $[u_{\min} - \delta_1, u_{\max} + \delta_2]$ 上的 l_2 个均匀分布随机数表示初始权值系数. 设定群体规模, 即设定基因编码组合数目. 我们先随机生成一定数目的基因个体, 然后从中挑出最好的个体加到初始群体中. 这个规模不断迭代, 直至初始群体规模达到了网络预先设定的规模.

4) 适应度函数计算

在遗传算法中, 对适应度函数的唯一要求是针对输入可以计算出能加以比较的非负结果. 因此这里采用网络的误差函数作为适应度函数, 并认为误差大的个体其适应度小, 具体表示为

$$F(w, v, \theta, r) = 10 - \sqrt{\sum_{k=1}^{N_1} \sum_{t=1}^n [y_k(t) - \hat{y}_k(t)]^2} \quad (3)$$

计算群体中每个个体的适应度, 由开关系数编码得到网络的隐节点个数, 由权重系数码得到网络的连接权, 输入训练样本, 按照 (3) 式计算每个个体的适应度.

5) 引入竞争的选择继承

选择继承的目的是为了从当前群体中选出优良的个体, 使它们有机会作为父代产生后代的个体. 判断个体优良与否的准则就是各自的适应度. 计算完各个体的适应度后, 选择适应度大的个体遗传到下一代, 从而使问题的解越来越接近于最优解空间. 经典的选择继承是由下式确定的^[1]

$$n_i = (f_i' / \bar{f}_i') \lambda \quad (4)$$

式中, n_i 为 a_i 在 $(t+1)$ 代时的繁殖个数, f_i' 为 a_i 的适应度函数值, \bar{f}_i' 为 p 的适应度函数值的和. 但在研究中发现经典的选择继承的淘汰率过低, 特别是当解集中的成员数比较小时, 容易发生震荡. 因此为获得较好的继承个体, 在这里引入淘汰率 rd , 即每遗传一代, 则适应度函数最差的几个成员都被淘汰掉, 通过这一竞争机制能够比较好的解决这个问题, 选择继承的过程表示如下:

$$n_i' = (f_i' / \bar{f}_i') \lambda \quad (5)$$

式中 $\bar{f}_i' = \sum_{i=1}^{\lambda(1-rd)} f_i'$.

6)交叉算子和变异算子

交叉就是按较大的概率从群体中,随机选择2个个体,交换这2个个体的某些位,交换的目的在于产生新的基因组合,以限制遗传材料的丢失.变异是以较小的概率对群体中的某些个体的位进行改变,在二进制编码中,即“1”变“0”,“0”变“1”;在实数编码中是对某些个体的位在(0,9)产生一个随机数代替原来个体的位.变异的目的在于保护一些适应度低的个体中的优良基因,防止寻优过程中过早收敛于不成熟区.

(1)对开关系数编码的交叉和变异操作

对开关系数编码的交叉和变异采用经典遗传算法中的单点交叉、均匀变异的方法方法,在交叉和变异运算时,当某个神经元删除时,相应的有关权重系数编码和阈值系数编码被置为0,也就是说这个网络连接被删除;而运算增加某个神经元时,则随机初始化有关权重系数编码和神经元阈值编码.由于权重系数编码和神经元阈值编码采用浮点数编码,需要设计新的交叉算子和变异算子.

以 p_c 的概率对选择后的个体进行交叉.设在第 i 个体和第 $i+1$ 个体之间进行交叉,交叉算子如下:

$$\left. \begin{aligned} X_i^{t+1} &= c_i \cdot X_i^t + (1 - c_i) \cdot X_{i+1}^t \\ X_{i+1}^{t+1} &= (1 - c_i) \cdot X_i^t + c_i \cdot X_{i+1}^t \end{aligned} \right\} \quad (6)$$

式中 X_i^t, X_{i+1}^t 是一对交叉前的个体, X_i^{t+1}, X_{i+1}^{t+1} 是交叉后的个体, c_i 是区间 $[0, 1]$ 的均匀分布的随机数.

以 p_m 的概率对交叉后的个体进行变异,设对第 i 个体进行变异,变异算子如下:

$$X_i^{t+1} = X_i^t + c_i \quad (7)$$

式中 X_i^t 是变异前的个体, X_i^{t+1} 是变异后的个体,如变异点在权重系数编码上则 c_i 是区间 $[u_{\min} - \delta_1 - X_i^t, u_{\max} + \delta_2 + X_i^t]$ 上的均匀分布随机数;这样可以保证变异后的个体仍在搜索区间内.

(2)对权重系数编码和神经元阈值编码的交叉和变异操作

对权重系数编码和神经元阈值编码的交叉操作采用多点交叉遗传操作,而不是常规的单点交叉,这样可以进一步增加遗传搜索的分散性,使其可以更快地收敛于所需的精度.多点交叉的交叉点数根据所给定的概率随机生成,交叉点的位置也是随机生成.

对权重系数编码和神经元阈值编码的变异操作采用多点自适应变异.就是使适应度大的个体在较小范围变异,而使适应度小的个体在较大的范围内变异^[5-6].基于这种考虑,引入变异温度的概念,这一概念类似于模拟退火算法中温度的概念,解的变异温度定义如下:

$$T = 1 - \frac{f(s)}{f_{\max}} \quad (8)$$

其中 $f(s)$ 表示个体 s 的适应度, f_{\max} 是待求解问题的最大适应度值.对于很多问题, f_{\max} 是难以确定的.这里只要给一个粗略的上限即可,也可以使用当前群体中的最大适应度值作为 f_{\max} .引入变异温度的概念以后,可以采用这样的变异方法.在个体 v 中随机地选择一个个体 v_k ,对变异后的个体 v_k 服从 $N(v_k, \delta^2(T))$ 正态分布.

7)生成新一代群体.

8)反复进行5~7,每进行一次,群体就进化一代,连续进化到 K 代(总的进化代数).

9)把第 K 代选适应度最高的个体解码得到相应的网络连接权和隐节点个数,输入检测样本检验模型的泛化能力.

3 番茄常见病害诊断应用实例及其结果分析

在本例中,选取7个病害特征参数的数据(经过必要的编码)来作为所要构建的神经网络的输入,网络的输出将作为诊断所得到的病害.7个作为网络输入的参数是: X_1 为发病部位; X_2 为病斑颜色; X_3 为病斑形状; X_4 为霉层颜色; X_5 为霉层形状; X_6 为生长特征; X_7 为其它特征.4个网络输出 Y_1, Y_2, Y_3, Y_4 为二值输出,其输出的16种输出状态分别代表16种番茄常见病害(番茄茎基腐病、番茄白绢病、番茄斑枯病等).在建立基于遗传算法BP神经网络模型时,首先采用三层BP神经网络估计网络连接权的基本解空间,输入训练样本(表1)和检测样本(表2),按照步骤1求得 $u_{\min} = 10.7012, u_{\max} = 14.9523, \theta_{\min} = -9.595, \theta_{\max} = 10.7953$ 则连接权的解空间初步设为 $[-12, 16]$ 阈值的解空间初步设定为 $[-11, 13]$,隐节点个数搜索范围为3~16,遗传算法进化过程中初始种群 $L = 30$,总的进化代数 $K = 100$,交叉概率 $p_c = 0.85$,淘汰率 $r_d = 0.15$,当前群体中的最大适应度 $f_{\max} = 10$,变异概率 $p_m = 0.006$.得到遗传算法误差平方和曲线、适应度曲线如图1.

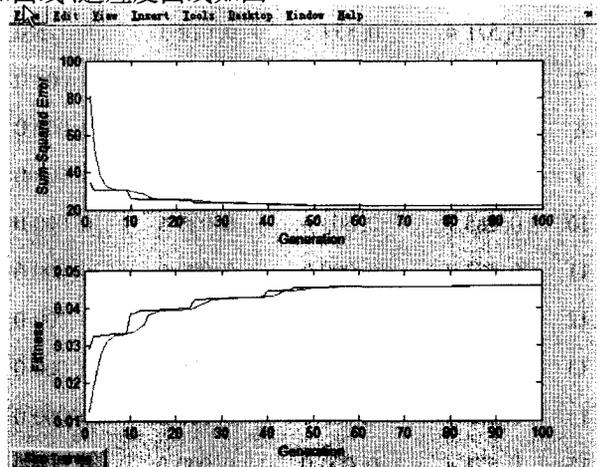


图1 其遗传算法误差平方和曲线、适应度曲线

表1 训练样本

No.	Input							Expect output			
	x_1	x_2	x_3	x_4	x_5	x_6	x_7	y_1	y_2	y_3	y_4
1	0.000 0	0.857 1	0.000 4	0.000 0	0.000 0	0.285 7	0.177 8	0	0	0	0
2	0.000 0	0.285 7	0.024 4	0.125 0	0.500 0	1.000 0	0.711 1	0	0	0	1
3	0.412 9	0.857 1	0.013	0.000 0	0.000 0	0.142 9	0.211 1	0	0	1	0
4	0.000 0	0.285 7	0.012 6	0.000 0	0.000 0	0.428 6	0.033 3	0	0	1	1
5	0.585 7	0.857 1	0.006 5	0.125 0	0.003 9	0.857 1	0.755 6	0	1	0	0
6	0.571 4	0.857 1	0.018 3	0.125 0	1.000 0	0.857 1	0.188 9	0	1	0	1
7	0.285 7	1.000 0	0.195 1	0.000 0	0.000 0	0.000 0	0.266 7	0	1	1	0
8	0.857 1	0.285 7	0.404	0.000 0	0.000 0	0.142 9	0.000 0	0	1	1	1
9	0.142 9	0.285 7	0.024 4	0.000 0	0.000 0	0.142 9	0.322 2	1	0	0	0
10	0.285 7	0.303 6	0.025 5	0.000 0	0.000 0	0.000 0	0.544 4	1	0	0	1
11	0.571 4	0.000 0	0.000 0	0.000 0	0.000 0	0.285 7	0.422 2	1	0	1	0
12	0.285 7	0.285 7	0.195 5	0.000 0	0.000 0	0.000 0	0.855 6	1	0	1	1
13	0.714 3	0.000 0	0.000 0	0.000 0	0.000 0	0.000 0	0.866 7	1	1	0	0
14	0.285 7	0.000 0	0.000 0	0.000 0	0.000 0	0.571 4	0.877 8	1	1	0	1
15	0.571 4	0.000 0	0.000 0	0.000 0	0.000 0	0.000 0	0.977 8	1	1	1	0
16	1.000 0	0.000 0	0.000 0	0.000 0	0.000 0	0.000 0	0.366 7	1	1	1	1

表2 测试样本

No.	Input							Expect output			
	x_1	x_2	x_3	x_4	x_5	x_6	x_7	y_1	y_2	y_3	y_4
1	0.000 0	0.857 1	0.000 1	0.000 0	0.000 0	0.285 7	0.177 8	0	0	0	0
2	0.000 0	0.285 7	0.024 4	0.125 0	0.500 0	1.000 0	0.711 1	0	0	0	1
3	0.142 9	0.857 1	0.013	0.000 0	0.000 0	0.142 9	0.211 1	0	0	1	0
4	0.000 0	0.285 7	0.012 6	0.000 0	0.000 0	0.428 6	0.033 3	0	0	1	1
5	0.585 7	0.857 1	0.006 5	0.125 0	0.003 9	0.857 1	0.755 6	0	1	0	0
6	0.571 4	0.857 1	0.018 3	0.125 0	1.000 0	0.857 1	0.188 9	0	1	0	1
7	0.285 7	1.000 0	0.195 1	0.000 0	0.000 0	0.000 0	0.266 7	0	1	1	0
8	0.857 1	0.285 7	0.404	0.000 0	0.000 0	0.142 9	0.000 0	0	1	1	1
9	0.142 9	0.285 7	0.024 4	0.000 0	0.000 0	0.142 9	0.322 2	1	0	0	0
10	0.285 7	0.303 6	0.025 5	0.000 0	0.000 0	0.000 0	0.544 4	1	0	0	1
11	0.571 4	0.000 0	0.000 0	0.000 0	0.000 0	0.285 7	0.422 2	1	0	1	0
12	0.285 7	0.285 7	0.195 5	0.000 0	0.000 0	0.000 0	0.855 6	1	0	1	1
13	0.714 3	0.000 0	0.000 0	0.000 0	0.000 0	0.000 0	0.866 7	1	1	0	0
14	0.285 7	0.000 0	0.000 0	0.000 0	0.000 0	0.571 4	0.877 8	1	1	0	1
15	0.571 4	0.000 0	0.000 0	0.000 0	0.000 0	0.000 0	0.977 8	1	1	1	0
16	1.000 0	0.000 0	0.000 0	0.000 0	0.000 0	0.000 0	0.366 7	1	1	1	1

得到网络的输入层与隐含层、隐含层与输出层最优连接权值矩阵分别如下

$$\begin{bmatrix} -1.05080 & 9.67470 & -6.42860 & 5.53350 & 1.12450 & -5.52900 & 5.12340 \\ 5.92690 & -3.85980 & -5.96270 & -3.25320 & 8.45170 & 1.19850 & -5.26690 \\ 4.79580 & -7.18620 & -1.56390 & -4.52870 & -5.75880 & -2.21090 & 0.65814 \\ -6.47800 & -1.48560 & 4.29620 & 1.62720 & -6.81050 & 3.07570 & 2.68700 \\ -7.80730 & -7.48030 & 0.61827 & 7.51680 & 1.71780 & 5.69070 & 0.10178 \\ 1.76540 & 3.00220 & 4.47860 & 4.61360 & 5.82050 & 4.02280 & -1.84780 \\ -4.78900 & 3.75510 & 2.22540 & 7.07340 & 4.71490 & -1.53840 & 3.27220 \\ 0.91777 & 4.83660 & -4.45570 & -1.14130 & 13.95230 & 1.00310 & 4.67720 \\ 4.39130 & -1.79530 & -4.03650 & -1.06200 & 1.28890 & 4.63960 & -0.93392 \\ 3.47560 & -3.23350 & -3.67760 & -4.48450 & -4.36950 & -1.69970 & -1.00240 \end{bmatrix}$$

图2 输入层与隐含连接权值矩阵

$$\begin{bmatrix} 0.24016 & 11.1457 & 5.9192 & -10.0173 & -2.7089 & 5.3735 & -1.79730 & 0.36120 & 0.86271 & 0.94549 \\ -4.0781 & 1.9140 & 12.1116 & 8.0289 & 1.0495 & -12.637 & -2.0565 & -4.79300 & -0.524 & 5.9125 \\ 4.9134 & -9.3880 & 7.4591 & -3.0249 & 3.6317 & -5.7222 & -10.7012 & 17.6237 & 0.52656 & -4.7313 \\ -14.7947 & 7.6973 & -1.4562 & -0.19772 & 15.6988 & 2.7717 & 2.1596 & 6.2001 & -3.7745 & -9.8506 \end{bmatrix}$$

图3 隐含层与输出层连接权值矩阵

根据上述网络参数设置网络得到的网络仿真如图4,与传统设计的BP网络仿真结果图5相比较.基于改进遗传算法的BP神经网络自适应优化设计的网络性能明显克服传统BP设计网络局部最优问题.

训练样本的实际输出和测试样本的实际输出如表3.

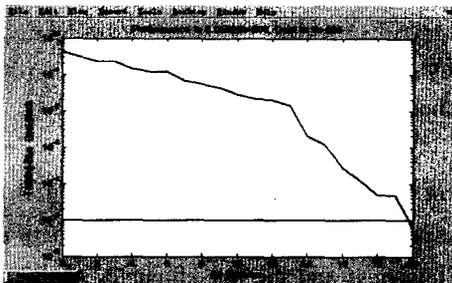


图4 基于改进遗传算法的BP神经网络自适应优化设计的网络仿真

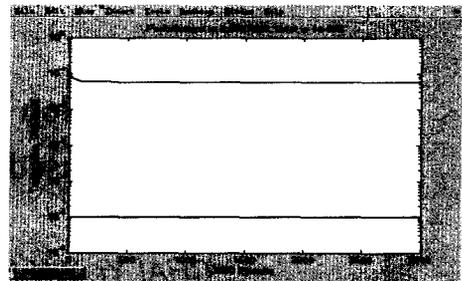


图5 传统BP设计,陷入局部最优的仿真结果

从表3我们可以得知对于已训练样本的诊断的正确率100%.对于从未训练过的测试样本的正确率也达到68.75%,证明基于改进遗传算法的BP神经网络自适应优化设计的网络具有良好的网络泛化性.

表3 实际输出和测试样本

Traning sample output				Testing sample output				Traning sample output				Testing sample output			
y_1	y_2	y_3	y_4												
0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.00	0.00	0.00	0.99	0.00	0.00	0.99
0.00	0.00	0.00	0.99	0.00	0.00	0.00	1.00	0.99	0.00	0.00	0.99	1.00	0.00	0.00	0.00
0.00	0.00	0.99	0.00	0.00	0.00	1.00	0.00	0.99	0.00	0.99	0.00	0.99	0.00	0.99	0.00
0.00	0.00	0.99	1.00	0.00	0.00	1.00	1.00	1.00	0.00	0.99	0.99	1.00	0.00	0.99	0.99
0.00	0.99	0.00	0.00	1.00	0.97	0.00	0.99	1.00	0.99	0.00	0.00	1.00	0.99	0.00	0.00
0.00	1.00	0.00	1.00	0.00	0.99	1.00	0.04	0.99	0.99	0.00	0.99	0.99	0.99	0.00	0.99
0.00	0.99	1.00	0.00	0.00	1.00	1.00	0.25	1.00	0.99	0.98	0.00	1.00	0.99	0.98	0.00
0.00	0.99	1.00	0.99	0.99	1.00	0.99	0.99	0.99	1.00	0.99	0.99	0.99	1.00	0.99	0.99

4 结束语

长期以来,由于对神经网络的拓扑结构设计和初始权值的设置,缺乏理论支持,使得所设计的网络往往存在网络规模庞大,设计效率低下与经典 BP 算法收敛过慢、局部收敛等问题。这常常是神经网络得不到有效应用的因素之一。本文使用遗传算法在全局解空间内对网络拓扑结构和网络参数进行自适应的动态调整,从而获得网络的最优设计。克服了由于神经网络初始权值的随机性和网络结构确定过程中所带来的网络振荡,以及网络极易陷入局部解问题并且有效提高神经网络的泛化能力,这种方法避免了一般神经网络依靠经验确定网络结构的困难。并将该方法应用到番茄常见病害预测的实例中,收到了比较好结果。

参考文献

[1] GOLDBERG D Genetic Algorithms in Search, Optimization and Machine Learn [M]. London: Addison-Wesley Publish-

ing Company Inc, 1990.

- [2] DJ MONTANA, L Da Training Feedforward Neural Networks Using Genetic Algorithms [C] // Proceedings 11th International Joint Conference on Artificial Intelligence. Detroit MI: Morgan Kaufman, San Mateo CA: 1989. 762-767.
- [3] H KITANO. Empirical Studies on the Speed of Convergence of Neural Network Training using Genetic Algorithms [C] // Proceeding 8th JMJT National Conference in Artificial Intelligence, 1990, Boston MASS, 1990: 789-796.
- [4] H KITANO. Empirical Studies on the Speed of Convergence of Neural Network Training using Genetic Algorithms [C] // Proceedings 8th JMIT National Conference in Artificial Intelligence. Boston MASS, 1990: 789-796.
- [5] 虞和济. 基于神经网络的智能诊断 [M]. 北京: 冶金工业出版社, 2000.
- [6] 阎平凡. 人工神经网络与模拟进化计算 [M]. 北京: 清华大学出版社, 2005.

Self-adaptation Optimize BP Neural Network Design Based on the Genetic Algorithms

CHAI Yi, YIN Hong-peng, LI Da-jie, ZHANG Ke

(College of Automation, Chongqing University, Chongqing 400030, China)

Abstract: BP (Back Propagation) Neural networks is in the presence of the local optimization in the Neural networks training. The algorithm have slow convergence and the local convergence problem which impact the neural networks work performance. In order to cover these shortcomings and solves the size's hugeness and the low efficiency of the net problem in the traditional NN designing, the action principles of BP-Neural network's structure are analyzed, and a new method is formed which is confirmed from the Enhance genetic algorithms (EGA). The method can identify network configuration and network training methods. By adopting the number coding, self-adaptable multi-point variations operation, this method can effectively reduce the network size and the network convergence time, increase the network training speed. Tomatoes disease diagnosis examples illustrate the feasibility of this approach.

Key words: improved genetic arithmetic; EGA; BP arithmetic; multi-layer sensor; NN Structure

(编辑 张小强)