

文章编号: 1000-8152(2005)06-0895-05

## 用过程神经网络和遗传算法实现系统逆向求解

李盼池

(大庆石油学院 计算机科学与工程学院, 黑龙江 大庆 163318)

**摘要:** 对于多输入多输出系统, 针对如何根据系统模型和期望输出反求系统输入的问题, 本文提出了一种基于过程神经网络和遗传算法相结合的方法. 首先根据实际系统的领域知识和学习样本集, 建立满足系统实际输入输出映射关系的正向过程神经网络. 然后按照系统在过程区间的某一期望输出, 用过程神经网络的输出误差构造适应度函数, 用遗传算法逆向确定系统的过程输入信号, 使该输入信号满足已建立的正向过程映射关系, 从而完成系统的逆向过程控制. 文中给出了具体的实现算法并给出了此方法的一个应用实例.

**关键词:** 过程神经网络; 遗传算法; 过程控制; 逆向求解

**中图分类号:** TP183      **文献标识码:** A

### Realization of system converse solution based on process neural networks and genetic algorithm

LI Pan-chi

(Department of Computer Science and Engineering, Daqing Petroleum Institute, Daqing Heilongjiang 163318, China)

**Abstract:** A optimization algorithm of process neural networks and genetic algorithm (PNN-GA) is proposed, it aims at determining MIMO system input by the system model and desired output. First, the process neural networks (PNN) that represent the mapping relation between input and output of the system is founded according to system field knowledge and training sample sets. Secondly, fitness function of genetic algorithm (GA) is constructed according to PNN output error. Based on the desired output of the system, we determined the process input signal which conforms to the PNN mapping relation that was found, thus the system converse process solution is accomplished. The general realization approach is presented in this paper. An application example is given to illustrate the applicability of the approach.

**Key words:** process neural networks; genetic algorithm; process control; converse solution

#### 1 引言 (Introduction)

在系统结构设计中, 通常遇到系统的逆向过程求解问题. 即事先给定系统的输出, 根据已建立的过程映射关系, 反求系统的输入, 其中系统的输入输出都可为依赖于过程或时间的函数. 随着系统输入输出维数的增加, 求解的过程趋于复杂. 近年来, 神经网络开始用于系统过程的逆向求解, 但网络的输入仅限于实值向量, 而对于系统的过程式输入是无能为力的. 文献[1]中提出了过程神经网络模型, 文献[2]给出了该模型基于权函数基展开的学习算法. 该模型的输入和权值都可是时变的, 较好的解决了系统的过程式输入输出映射问题. 但当系统的正向映射关系不可逆时, 逆向网络是不收敛的. 遗传算法具有很好的鲁棒性和全局搜索性, 目前在人工智能的许多领域有着很成功的应用. 针对系统过程的逆

向寻优问题, 本文提出了一种基于过程神经网络和遗传算法相结合的求解方案. 首先根据实际问题的领域知识和学习样本集, 用过程神经网络确定能够满足系统实际输入输出的正向映射关系, 然后根据系统的期望输出用遗传算法优化求解逆向时的网络权值. 文中给出了系统过程逆向求解的网络模型、算法原理及实施方案, 实际应用部分证明了方法的有效性.

#### 2 逆向求解网络模型 (Converse solution networks model)

##### 2.1 过程神经网络 (Process neural networks)

过程神经网络是由若干过程神经元和一般非时变神经元按一定拓扑结构组成的网络. 一个多输入多输出系统的前向过程神经网络拓扑结构如图 1 所示. 图中: 输入层有  $n$  个节点, 中间层 (过程神经元

隐层)有  $m$  个节点,激励函数为  $f$ ,输出层有  $p$  个节点,激励为线性函数.网络输入输出之间的映射关系为<sup>[1]</sup>

$$\begin{cases} y_k = \sum_{j=1}^m v_{jk} f\left(\sum_{i=1}^n \int_0^T w_{ij}(t) x_i(t) dt - \theta_j\right), \\ k = 1, 2, \dots, p. \end{cases} \quad (1)$$

式中:  $w_{ij}$  为输入层节点  $i$  到过程神经元隐层节点  $j$  的连接权函数,  $v_{jk}$  为隐层节点  $j$  到输出层节点  $k$  的连接权值,  $\theta_j$  为隐层节点  $j$  的阈值,  $y_k$  为网络实际输出.

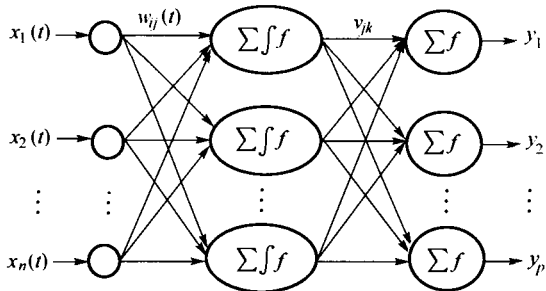


图1 含一个隐层的过程神经网络

Fig. 1 Process neural networks with one hidden layer

## 2.2 逆向过程神经网络模型 (Converse solution process neural networks model)

图1所示模型实际为系统的正向模型.由文献[3]知,逆向过程神经网络模型(图2)可在正向模型前端加一新隐层,新隐层的输入为一维函数向量  $X_0(t) \equiv 1$ ,并且在根据误差信号调整网络的权函数时,只需调整新隐层权函数  $W_0(t) = [w_{01}(t), w_{02}(t), \dots, w_{0n}(t)]^T$ ,而原正向模型的网络权值保持不变.由图2知,新隐层的输入为  $S_1(t) = [s_1(t), s_2(t), \dots, s_n(t)]^T$ ,新隐层采用线性转移函数  $f(s(t)) = s(t)$ ,输出为  $U_1(t) = [u_1(t), u_2(t), \dots, u_n(t)]^T = f(S_1(t)) = [w_{01}(t), w_{02}(t), \dots, w_{0n}(t)]^T$ .这表明,新隐层权函数向量等于原模型的输入函数向量.注意到原模型的输入为逆向求解的目标,当调整权函数  $W_0(t)$  且网络训练收敛时,相应的权函数  $W_0(t)$  便是系统期望输出对应的逆向求解目标,从而实现了系统逆向过程的求解.

如上所述,图2中第一隐层权值其实对应着逆向求解的目标,而由于求解的目标仍然需要满足系统的正向映射关系,所以,求解过程中原系统的正向网络权值是不能改变的.因此若用梯度下降法将导致收敛困难.对此现象的解释是:用梯度下降算法(BP算法)来调整网络权值时,需要同时调整所有各

层的权值;而该模型只调整新隐层权值,原模型的权值保持不变,只能获得总体误差信号趋于极小的部分贡献,所以很难使这种调整策略获得最终收敛.因此,对于新隐层权值的调整,本文采用遗传算法.在遗传算法中染色体适应度函数的构造仍需要依赖于系统正向过程的误差函数,需要用到过程神经网络确定的正向映射关系,所以该算法也可称为“过程神经网络-遗传”算法,简称 PNN-GA (Process Neural Network-Genetic Algorithm) 算法.近年来,遗传算法的良好特性使它在一些连续及离散变量的优化设计领域显示出良好的应用前景<sup>[4]</sup>,其主要优点是具有很强的通用优化能力,不需梯度信息,不需函数连续,且优化结果具有全局性.

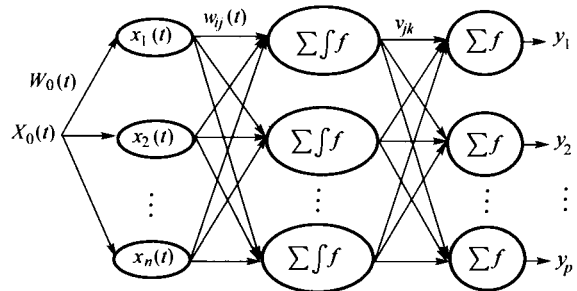


图2 逆向求解过程神经网络模型

Fig. 2 Converse solution process neural networks model

## 3 逆向求解的 PNN-GA 算法 (PNN-GA algorithm of converse solution)

### 3.1 建立正向映射关系的 PNN 算法 (PNN algorithm for finding forward mapping relation)

设系统输入函数空间为  $(C[0, T])^n$ , 输出为反映系统阶段累积效应的一组有序离散值.  $b_1(t), b_2(t), \dots, b_n(t)$  为  $C[0, T]$  中一组标准正交基函数, 则输入函数和网络权函数在该组基函数下的展开式可表示为 ( $L$  为满足展开精度要求的自然数)

$$x_i(t) = \sum_{l=1}^L a_{il} b_l(t) \quad (i = 1, 2, \dots, n), \quad (2)$$

$$w_{ij}(t) = \sum_{l=1}^L w_{ij}^{(l)} b_l(t) \quad (i = 1, 2, \dots, n; j = 1, 2, \dots, m). \quad (3)$$

式中  $a_{il}, w_{ij}^{(l)}$  分别为  $x_i(t)$  和  $w_{ij}(t)$  相对于  $b_l(t)$  的展开式系数. 将式(2), (3)带入式(1)中, 则网络输入输出关系可表述为

$$y_k = \sum_{j=1}^m v_{jk} f\left(\sum_{i=1}^n \int_0^T \left(\sum_{l=1}^L w_{ij}^{(l)} b_l(t)\right) \left(\sum_{s=1}^L a_{is} b_s(t)\right) dt - \theta_j\right). \quad (4)$$

由基函数的正交性,式(4)可化简为

$$y_k = \sum_{j=1}^m v_{jk} f\left(\sum_{i=1}^n \sum_{l=1}^L a_{il} w_{ij}^{(l)} - \theta_j\right) (k = 1, 2, \dots, p). \tag{5}$$

给定学习样本集,按照梯度下降算法,可确定出正向网络权值,即可确定满足系统输入输出正向映射关系的过程神经网络.

### 3.2 用于逆向求解的 GA 算法 (GA algorithm for converse solution)

将  $W_0(t)$  用与式(3)相同的正交基展开得

$$w_{0i}(t) = \sum_{l=1}^L w_{0i}^{(l)} b_l(t) (i = 1, 2, \dots, n).$$

由基函数的确定性易知,  $w_{0i}(t)$  与一组基函数展开式的系数  $w_{0i}^{(l)} (l = 1, 2, \dots, L)$  存在一一对应关系,即可由一组  $w_{0i}^{(l)}$  唯一确定. 因此只需优化求解  $w_{0i}^{(l)}$  即可. 一种基于实数编码的遗传算法结构见图 3, 具体可描述为

1) 种群染色体的初始化. 可直接使用问题域空间中待优化的解向量作为染色体, 即采用十进制实数直接对  $W = [w_{01}^{(1)}, w_{01}^{(2)}, \dots, w_{0n}^{(L)}]$  编码, 每条染色体上基因数为  $nL$  个. 分别赋予  $[-1, 1]$  间随机数, 完成种群染色体的初始化.

2) 染色体适应度函数的构造. 染色体适应度函数可结合图 2 过程神经网络的输出误差函数构造, 考虑到适应度的实际意义具体可取为误差函数的倒数<sup>[5]</sup>.

3) 种群选择. 可采用转轮规则选择染色体种群<sup>[6]</sup>, 其基本思想是每个染色体的选择概率正比于它的适应度值. 对于适应度值为  $g_k$  的染色体  $k$ , 其选择概率按如下方式计算:  $p_k = g_k / \sum_{j=1}^{P_s} g_j$ . 其中,  $P_s$  为种群规模. 根据这些概率值构造一个转轮, 旋转转轮  $P_s$  次, 每次选择一个染色体加入新的种群.

4) 算术交叉. 算术交叉定义为两条染色体的如下组合方式:  $W'_1 = \lambda W_1 + (\bar{1} - \lambda) W_2, W'_2 = \lambda W_2 + (\bar{1} - \lambda) W_1$ . 这里:  $\lambda$  为  $nL$  维实值向量,  $\lambda_i \in (0, 1) (i = 1, 2, \dots, nL)$  为随机数.

5) 非均匀变异. 非均匀变异定义为: 对于给定的父染色体  $W$ , 若它的元素  $w_i$  被选来变异, 则生成的后代为  $W' = (w_{01}^{(1)}, \dots, w'_i, \dots, w_{0n}^{(L)})$ . 其中  $w'_i$  随机地按如下两种可能的机会变化:

$$w'_i = w_i + \Delta(t, w_i^U - w_i) \text{ 或 } w'_i = w_i - \Delta(t, w_i - w_i^L),$$

其中:  $t$  为遗传代数,  $w_i^U$  和  $w_i^L$  分别是  $w_i$  的上下界. 函数  $\Delta(t, z)$  随  $t$  的增加而趋于 0. 如可取  $\Delta(t, z)$  为  $\Delta(t, z) = z \cdot r(1 - t/\text{epoch})^d$ .

其中:  $r \in [0, 1]$  为随机数,  $\text{epoch}$  为最大遗传代数,  $d$  是确定非均匀度的参数.

6) 循环条件的控制. 按给定的适应度函数计算每个染色体的适应度; 循环条件可选择遗传代数控制或目标函数极小化精度控制或适应度函数目标值控制.

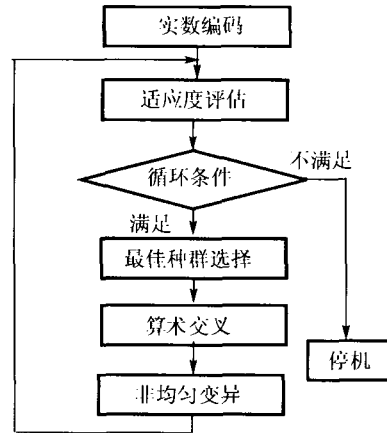


图 3 基于实数编码的遗传算法  
Fig. 3 Genetic algorithm based on real number coding

## 4 实际应用 (Application example)

实验室渗流研究是油田开发采收率预测中的一项重要工作. 在渗流实验中各参量的变化可反映油藏的采收率. 现在已知油藏区块采收率与各参量变化情况的 38 组对应数据(表 1 只给出一组), 要求以这些数据建立过程映射关系, 然后根据相同环境下测得的一组新数据的期望输出(表 2 末行)和已经建立好的映射关系, 反求各参量的一组变化趋势. 这是石油开采中经常遇到的重要的实际问题.

表 1 渗流实验记录的一组样本数据(训练用)

Table 1 Sample data of seepage experiment (for training)

$t/h$	$V_w/ml$	$V_o/ml$	$P_v$	$C/(%)$	$P/Pa$
1	6.30	0.40	0.6481	97.08	0.51
2	6.30	0.40	0.7058	97.08	0.52
3	6.30	0.35	0.7632	97.08	0.51
4	6.50	0.35	0.8227	97.70	0.51
5	6.60	0.35	0.8821	97.73	0.50
...	...	...	...	...	...
20	7.50	0.20	1.5283	98.05	0.49

$$V = 585.6985 \quad V_p = 123.5 \quad R = 21.38\%$$

表2 渗流实验记录的一组样本数据(反求用)

Table 2 Sample data of seepage experiment (for converse solution)

$t/h$	$V_w/ml$	$V_o/ml$	$P_v$	$C/(%)$	$P/Pa$
1	6.18	0.36	0.6238	97.15	0.51
2	6.18	0.36	0.6316	97.18	0.51
3	6.20	0.35	0.6895	97.30	0.51
4	6.30	0.33	0.7693	97.55	0.50
5	6.35	0.31	0.8365	97.68	0.50
...	...	...	...	...	...
20	7.38	0.15	1.5016	98.13	0.49

$V = 579.3298$   $V_p = 128.9$   $R = 21.85\%$

表1、表2中参数说明:

$V_w$ :产水量;  $V_o$ :产油量;  $P_v$ :孔隙体积倍数;  $C$ :含水率;  $P$ :压强;  $V$ :岩样体积;  $V_p$ :空隙体积;  $R$ :采收率.

表1中第一列为以小时记的时间,其他几列表示在一段测试时间段中几个参数的变化情况.最后一行给出了作为过程神经网络输出的3个测试参数.

#### 4.1 应用 PNN 建立系统正向映射关系 (Finding forward mapping relation by PNN)

应用上述38组数据可以得到过程神经网络的38个学习样本,用来构建正向过程的映射关系.网络输入及权函数采用walsh基函数展开.本例中过程神经网络的训练参数见表3.

表3 建立正向映射关系的过程神经网络参数

Table 3 Process neural networks parameter for finding forward mapping relation

输入节点	基函数个数	隐层节点	输出节点	学习速率	惯性系数	误差精度	限定次数
5	8	50	3	0.5	0.3	0.01	5000

按梯度下降法训练网络,图4给出了网络误差的动态收敛曲线( $N$ 为迭代次数, $E$ 为均方根误差).收敛后的网络将正向过程的映射关系存储在网络权值中.此时,网络对全部38个样本数据均能较好的逼近.图5给出了逼近结果曲线( $X$ 为样本序号, $Y$ 为样本输出值).

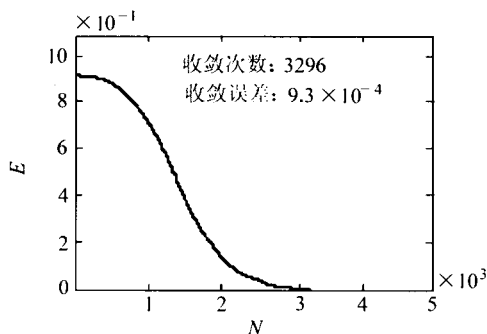


图4 网络误差动态收敛曲线

Fig. 4 Networks error convergence curve

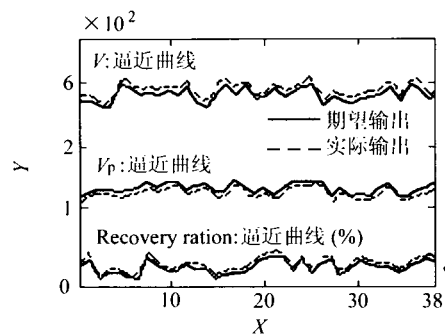


图5 收敛后的网络输出逼近曲线

Fig. 5 Networks output approach curve after convergence

#### 4.2 应用 PNN-GA 逆向反求系统输入 (Converse solution system input by PNN-GA)

采用表2中最后一行数据作为期望输出,现在用遗传算法根据前面建立的正向映射关系反求各参数的一组输入.实验中用到的各参数为:种群规模  $P_s = 50$ , 基因值上界  $w^U = 15$ , 基因值下界  $w^L = -5$ , 交叉率  $P_c = 0.7$ , 变异率  $P_m = 0.01$ , 最大遗传代数  $epoch = 4000$ , 限定适应度值  $g = 3.5 \times 10^3$ . 根据逆向求解网络模型(图2), 遗传算法的优化目标为与  $W_0(t)$  对应的一组基展开系数:  $w_{0i}^{(l)}$  ( $i = 1, 2, \dots, n, l = 1, 2, \dots, L$ ). 本例中,  $n = 5, L = 8$ . 将全部  $w_{0i}^{(l)}$  按序排列成40维向量  $W = [w_{01}^{(1)}, w_{01}^{(2)}, \dots, w_{05}^{(8)}]$  完成染色体的构造. 将种群中全部染色体各基因位赋值  $[-1, 1]$  间十进制随机数, 完成基因编码及种群初始化. 适应度函数采用网络误差函数的倒数

$$g(W) = \left( \sum_{k=1}^3 (d_k - y_k)^2 \right)^{-1}. \quad (6)$$

其中:  $d_k$  为表2中样本数据第  $k$  维的期望输出;  $y_k$  为与  $d_k$  对应的实际输出, 具体如式(5).

按3.2节所述步骤, 首先对种群染色体实施遗传操作(选择、交叉、变异)产生新一代种群; 将新种群各染色体解码还原为网络权值  $w_{0i}^{(l)}$  ( $i = 1, 2, \dots, 5, l = 1, 2, \dots, 8$ ), 应用已建立的正向映射关系求得网络输出, 按式(6)分别计算各染色体的适应度, 并依此建立各染色体的选择概率. 若当代种群中最高染色体适应度值达到或超过系统给定的限定适应度值  $g$ , 则对该染色体解码可得最终反向寻优的结果; 否则, 继续遗传操作, 直到寻优成功或遗传代数达到限定值. 图6给出了两次求解的适应度曲线( $N$ 为遗传代数,  $F$ 为适应值). 图7以输入参数  $w/ml$  为例给出了两次求解得到的输入曲线与真实输入曲线对比.

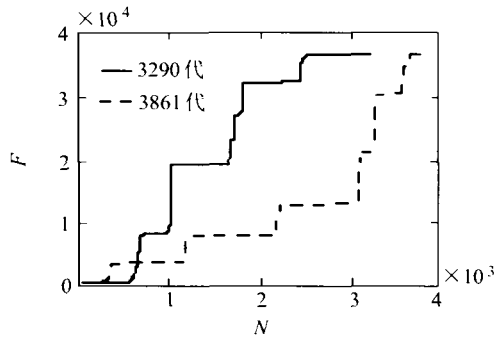


图6 遗传算法逆向求解的适应度曲线

Fig. 6 Fitness curve of genetic algorithm for converse solution

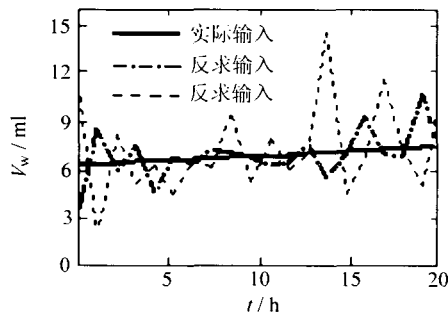


图7 求解结果曲线与真实曲线对比

Fig. 7 Comparison between result curve and real curve

实验中发现:应用本文提出的 PNN-GA 算法求得系统输入,虽然能够保证系统在该输入下,使实际输出较好的逼近期望输出,但在一般情况下却不能保证这样求得的输入也能够较好的逼近真实输入,并且求得的结果也不是唯一的.图7示意了一个输入特征,其他4个特征与之类似.这反映出在给定期望输出的情况下,满足系统正向映射关系的输入具有多样性;也反映了 PNN-GA 算法解空间的多样性.因此,该算法具有较好的柔韧性和可行性.关于如何保证该算法求解的唯一性,融入样本先验知识可能是一条可行的途径.目前作者正在进行这方面的研究.

## 5 结束语(Conclusion)

对于输入为过程而输出为实值向量的系统逆向求解问题,本文提出了一种基于过程神经网络和遗

传算法相结合的 PNN-GA 方法,该方法融合了过程神经网络能够处理过程式输入而方便对时变系统建模和遗传算法能够进行全局搜索的优点.文中的应用实例表明,该模型和算法对过程控制问题具有较好的适应性.

## 参考文献(References):

- [1] 何新贵,梁久祯.过程神经网络的若干理论问题[J].中国工程科学,2000,2(12):40-41.  
(HE Xingui, LIANG Jiuzhen. Some theoretical issues on procedure neural networks [J]. *Engineering Science*, 2000, 2(12): 40-41.)
- [2] 何新贵,梁久祯,许少华.过程神经网络的训练及其应用[J].中国工程科学,2001,3(4):31-32.  
(HE Xingui, LIANG Jiuzhen, XU Shaohua. Learning and application of procedure neural networks [J]. *Engineering Science*, 2001, 3(4): 31-32.)
- [3] 谢庆生,尹健,罗延科.机械工程中的神经网络方法[M].北京:机械工业出版社,2003:153-155.  
(XIE Qingsheng, YIN Jian, LUO Yanke. *Neural Networks Method in Machine Engineering* [M]. Beijing: China Machine Press, 2003: 153-155.)
- [4] 黄卫忠,高国琴.基于遗传算法的最优控制加权阵的设计[J].计算机测量与控制,2003,11(10):761-762.  
(HUANG Weizhong, GAO Guoqin. Design of weighting matrix for optimal controller based on genetic algorithm [J]. *Computer Measurement & Control*, 2003, 11(10): 761-762.)
- [5] 王小平,曹立明.遗传算法——理论应用与软件实现[M].西安:西安交通大学出版社,2002:25-50.  
(WANG Xiaoming, CAO Liming. *Genetic Algorithm—Theoretical Application & Programme Realization* [M]. Xi'an: Xi'an Jiaotong University Press, 2002: 25-50.)
- [6] 陈国良,王煦法,庄镇泉,等.遗传算法及其应用[M].北京:人民邮电出版社,1999:75-88.  
(CHEN Guoliang, WANG Xufa, ZHUANG Zhengquan, et al. *Genetic Algorithm & Application* [M]. Beijing: Posts & Telecom Press, 1999: 75-88.)

## 作者简介:

李盼池 (1969—),男,教师,研究方向为神经网络、模糊控制、进化计算等, E-mail: lipanchi@vip.sina.com.