

# 一种多层前馈网参数可分离学习算法<sup>1)</sup>

章云

(广东工业大学自动化所 广州 510090)

毛宗源

(华南理工大学自动化系 广州 510641)

杨宜民

(广东工业大学自动化所 广州 510090)

**摘要** 目前大部分神经网络学习算法都是对网络所有的参数同时进行学习. 当网络规模较大时, 这种做法常常很耗时. 由于许多网络, 例如感知器、径向基函数网络、概率广义回归网络以及模糊神经网络, 都是一种多层前馈型网络, 它们的输入输出映射都可以表示为一组可变基的线性组合. 网络的参数也表现为二类: 可变基中的参数是非线性的, 组合系数是线性的. 为此, 提出了一个将这二类参数进行分离学习的算法. 仿真结果表明, 这个学习算法加快了学习过程, 提高了网络的逼近性能.

**关键词** 神经网络, 学习算法, 系统辨识, 参数解耦.

## 1 引言

自1982年Hopfield发表了自反馈神经网络的文章和1986年Rumelhart等人发表了专著PDP以来, 神经网络再度受到控制界的关注. 特别是随后许多学者的工作<sup>[1,2,3]</sup>都证实了多层前馈网所实现的映射可一致逼近紧集上的连续函数或以范数的意义逼近紧集上的平方可积函数, 神经网络这种任意逼近能力为它在非线性系统建模和控制中的应用提供了坚实的理论基础. 然而, 这种任意逼近能力的证明都是非构造性的, 这为神经网络的具体应用带来不便. 因而许多学者都致力于各种神经网络学习算法的研究, 以便得到满足实际应用的神经网络拓扑结构和参数.

大量的计算机仿真和实际应用表明, 当对象的非线性程度较高时, 为了达到一定的逼近精度, 所需要的神经网络结构往往较大, 相应的待确定的参数也就较多. 目前大部分学习算法都是对这些参数同时进行调整, 这也是学习过程较长的原因之一. 仔细分析当前使用的几种典型的神经网络可知, 这些网络中的参数可分离成两大部分: 一部分为线性参数, 另一部分为非线性参数. 本文就是在这一观察的基础上, 提出了一种参数可分离的学习算法, 从而加快了学习的进程.

## 2 几种典型神经网络的统一数学描述

### 2.1 多层感知器

一般认为, 大脑中实际的神经结构是前馈式的(不一定全部)和分层的. 依据这一物理

1)广东省自然科学基金资助项目.

收稿日期 1997-01-02

结构, Rosenblatt 等人给出了一种分层前馈网络结构的大脑模型, 称之为多层感知器 (PR). 这种多层感知器中最常用的是输入与输出层为线性层、隐层为非线性层的三层感知器, 它的输入输出映射关系可写为

$$y = f(\mathbf{x}) = \sum_{j=1}^q \theta_j \phi(a_j), \quad (1a)$$

$$a_j = \sum_{i=1}^d \omega_{ij} x_i - \omega_{0j}, j = 1, 2, \dots, q. \quad (1b)$$

式中  $a_j$  为第  $j$  个隐层神经元的输入激励;  $\phi(\cdot)$  为激励函数, 通常取为 Sigmoid 函数;  $q$  为隐层节点数;  $\omega_{ij}, \theta_j (i=0, 1, \dots, d; j=1, 2, \dots, q)$  为网络的权重与阈值;  $y \in R$  是网络输出;  $\mathbf{x} \in R^d$  是网络输入.

## 2.2 径向基函数网络

在大脑皮层和视网的研究领域里, 局部协调和重叠式接收域是一种常见的结构. Moody 和 Darken 基于这种生物接收域提出了一种网络结构<sup>[4]</sup>, 取名为径向基函数网络 (RBF). 这种网络利用局部接收域完成函数映射, 通常每个接收域单元的输出为高斯函数

$$p_j(\mathbf{x}) = \exp[-r_j \|\mathbf{x} - \mathbf{m}_j\|^2], \quad r_j > 0. \quad (2a)$$

径向基函数网络的输出为

$$y = f(\mathbf{x}) = \sum_{j=1}^q p_j(\mathbf{x}) \theta_j, \quad (2b)$$

式中  $\theta_j$  为接收域  $j$  的强度.

## 2.3 概率广义回归网络

Specht 提出的概率广义回归网络是利用高斯型函数代替通常的 Sigmoid 函数, 来给出能逼近贝叶斯最优判决式的非线性决策边界<sup>[5]</sup>. 概率广义回归网络的输入输出映射关系可写为

$$y = f(\mathbf{x}) = E(\hat{y}|\mathbf{x}) = \frac{\sum_{i=1}^N \bar{y}_i \exp[-r \|\mathbf{x} - \bar{\mathbf{x}}_i\|^2]}{\sum_{i=1}^N \exp[-r \|\mathbf{x} - \bar{\mathbf{x}}_i\|^2]}, \quad r > 0. \quad (3)$$

式中  $\{(\bar{\mathbf{x}}_i, \bar{y}_i), i=1, 2, \dots, N\}$  为输入输出的样本值.

## 2.4 模糊神经网络

模糊神经网络 (FNN) 是以神经网络的结构实现模糊逻辑推理<sup>[6]</sup>, 如果采用如下形式的模糊规则

$$\text{if } x_1 \text{ is } F_1^j, x_2 \text{ is } F_2^j, \dots, x_d \text{ is } F_d^j, \text{ then } y \text{ is } G^j \quad (4)$$

以及乘积推理、中心去模糊化, 它的输入输出映射关系为

$$y = f(\mathbf{x}) = \frac{\sum_{j=1}^q \theta_j \prod_{i=1}^d \mu(x_i, m_{ij}, r)}{\sum_{j=1}^q \prod_{i=1}^d \mu(x_i, m_{ij}, r)} = \sum_{j=1}^q \theta_j \phi(\mathbf{x}, \mathbf{m}_j, r), \quad (5a)$$

$$\phi(\mathbf{x}, \mathbf{m}_j, r) = \frac{\prod_{i=1}^d \mu(x_i, m_{ij}, r)}{\sum_{j=1}^q \prod_{i=1}^d \mu(x_i, m_{ij}, r)}. \quad (5b)$$



式中  $m_j \in R^d$  是第  $j$  条规则中隶属函数的中心,  $0 < r \in R$  反映隶属函数的宽度.

若对式(4)的模糊规则进一步推广,即采用如下形式的模糊规则

$$\text{if } x_1 \text{ is } F_1^j, x_2 \text{ is } F_2^j, \dots, x_d \text{ is } F_d^j, \text{ then } y = f_j(\mathbf{x}) \quad (6)$$

这时的模糊神经网络实际上可实现分段建模的思想,即模糊规则的前提是对输入空间进行一种划分,模糊规则的结论反映相应的局部模型.而式(4)的模糊规则相当于采用的是常数型的局部模型.

采用何种形式的局部模型没有定论,由于多项式具有良好的性质常被选作局部模型.取

$$f_j(\mathbf{x}) = c + a_1 x_1 + \dots + a_d x_d + b_1 x_1 x_2 + \dots = \mathbf{g}_j^T(\mathbf{x}) \theta_j. \quad (7)$$

式中  $\mathbf{g}_j(\mathbf{x})$  是由  $\mathbf{x} \in R^d$  各分量及组合组成的列向量,参数  $\theta_j$  为对应的系数组成的列向量.则这时的模糊神经网络的输入输出映射关系可写为

$$y = f(\mathbf{x}) = \sum_{j=1}^q f_j(\mathbf{x}) \phi(\mathbf{x}, m_j, r) = \sum_{j=1}^q \mathbf{g}_j^T(\mathbf{x}) \phi(\mathbf{x}, m_j, r) \theta_j. \quad (8)$$

## 2.5 问题的描述

综上所述,上述网络模型无论是何种结构,无论是以什么样的思想进行的构造,它们给出的输入与输出的映射关系都是相似的(参见式(1)(2)(3)(5)(8)),都可看成是一种与模型无关的估计器.而且这种输入与输出的映射关系都可化为

$$y = f(\mathbf{x}, \omega, \theta) = \sum_{j=1}^q \phi(\mathbf{x}, \omega_j) \theta_j. \quad (9)$$

其中  $\omega_j$  是可调的基参数;  $\theta_j$  是投影系数;  $\phi(\mathbf{x}, \omega_j)$  为第  $j$  个可变基函数,它取决于神经元的激励函数或隶属函数;  $q$  与隐节点数或模糊规则数有关.基于此,这些网络的参数学习问题可描述如下.设期望的样本数据为  $\{y_{di}, \mathbf{x}_i, i=1, 2, \dots, N\}$ , 则

$$\begin{bmatrix} y_{d1} \\ \vdots \\ y_{dN} \end{bmatrix} = \begin{bmatrix} \phi(\mathbf{x}_1, \omega_1) & \dots & \phi(\mathbf{x}_1, \omega_q) \\ \vdots & & \vdots \\ \phi(\mathbf{x}_N, \omega_1) & \dots & \phi(\mathbf{x}_N, \omega_q) \end{bmatrix} \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_q \end{bmatrix} + \boldsymbol{\varepsilon}, \quad (10a)$$

$$\mathbf{Y} = \Phi(\mathbf{X}, \omega) \boldsymbol{\theta} + \boldsymbol{\varepsilon}, \mathbf{Y} \in R^N, \mathbf{X} \in R^{N \times d}, \omega \in R^{p \times q}, \boldsymbol{\theta} \in R^q, \boldsymbol{\varepsilon} \in R^N, \quad (10b)$$

$$J = \|\mathbf{Y} - \Phi(\mathbf{X}, \omega) \boldsymbol{\theta}\| = J(\omega, \boldsymbol{\theta}), \quad (10c)$$

$$\{\omega^*, \boldsymbol{\theta}^*\} = \arg \min J(\omega, \boldsymbol{\theta}). \quad (10d)$$

式(10)中  $\boldsymbol{\varepsilon}$  是模型残差.下面的问题是如何优化指标(式 10c),得到网络的最佳参数.

## 3 参数可分离学习算法(PSL)

不失一般性,设  $\text{rank}(\Phi(\mathbf{X}, \omega)) = q \leq N$ , 则存在正交矩阵  $U, V$ , 使得

$$U \Phi(\mathbf{X}, \omega) V = \begin{bmatrix} \Delta \\ 0 \end{bmatrix}, \text{rank}(\Delta) = q, \Delta \in R^{q \times q}, \quad (11)$$

$$\Phi^+(\mathbf{X}, \omega) = V^T [\Delta^{-1} \quad 0] U^T. \quad (12)$$

由于正交矩阵的保范性,从式(10c)得

$$J = \|U(\mathbf{Y} - \Phi(\mathbf{X}, \omega) \boldsymbol{\theta})\| = \left\| \begin{bmatrix} U_1 \mathbf{Y} - \Delta V^T \boldsymbol{\theta} \\ U_2 \mathbf{Y} \end{bmatrix} \right\| = \|U_1 \mathbf{Y} - \Delta V^T \boldsymbol{\theta}\| + \|U_2 \mathbf{Y}\|. \quad (13)$$

取

$$\theta = V\Delta^{-1}U_1Y = \Phi^+(X, \omega)Y, \quad (14)$$

则由式(13)得到另一个优化指标

$$\hat{J} = \|U_2Y\| = \|(I - P_\phi)Y\| = Y^T(I - \Phi(X, \omega)\Phi^+(X, \omega))Y, \quad (15)$$

$$P_\phi = \Phi(X, \omega)\Phi^+(X, \omega). \quad (16)$$

式中  $P_\phi$  为投影算子. 可见,  $\hat{J}$  只依赖于参数  $\omega$ , 并且有如下定理.

定理 1. 若  $\omega \in \Omega$ ,  $\text{rank}(\Phi(X, \omega)) = q \leq N$ ,  $\omega^*$  是  $\hat{J}(\omega)$  的全局极小点,  $\theta^*$  按照式(14)计算, 那么  $\{\omega^*, \theta^*\}$  一定是  $J(\omega, \theta)$  的全局极小点, 且  $J(\omega^*, \theta^*) = \hat{J}(\omega^*)$ .

这个定理从理论上保证了式(10)的参数学习问题可分两步进行, 首先优化指标  $\hat{J}(\omega)$  得到  $\omega^*$ , 然后由式(14)求出  $\theta^*$ . 为了优化指标  $\hat{J}(\omega)$ , 需要得到  $\hat{J}(\omega)$  对  $\omega$  的导数, 下面给出它的求解公式.

记  $\dot{\Phi}(\omega)$  为  $\Phi(\omega)$  对  $\omega$  的导数, 并将  $Y$  分解为

$$Y = Y_1 + Y_2, \quad Y_1 = \Phi(X, \omega)\theta \in R(\Phi), \quad Y_2 \in N(\Phi^T). \quad (17)$$

式中  $R(\Phi)$ ,  $N(\Phi)$  分别记  $\Phi$  的像空间、零空间.

定理 2. 若  $Y$  按式(17)分解, 则

$$\frac{\partial \hat{J}}{\partial \omega_{kj}} = -2Y_2^T \dot{\Phi}(\omega_{kj})\theta, \quad k = 1, 2, \dots, p; j = 1, 2, \dots, q, \quad (18)$$

$$\hat{J} = Y_2^T Y_2. \quad (19)$$

证明. 为简明起见, 在不引起混淆的情况下忽略符号  $\omega$ .

$$\because \Phi = P_\phi \Phi,$$

$$\therefore \dot{\Phi} = \dot{P}_\phi \Phi + P_\phi \dot{\Phi}, \quad \dot{P}_\phi \Phi = (I - P_\phi) \dot{\Phi}.$$

$$\because P_\phi = P_\phi P_\phi,$$

$$\therefore \dot{P}_\phi = \dot{P}_\phi P_\phi + P_\phi \dot{P}_\phi.$$

由式(15)得

$$\begin{aligned} \frac{\partial \hat{J}}{\partial \omega_{kj}} &= -2Y^T(I - P_\phi)\dot{P}_\phi Y = -2Y^T(I - P_\phi)(\dot{P}_\phi P_\phi + P_\phi \dot{P}_\phi)Y = \\ &= -2Y^T(I - P_\phi)\dot{P}_\phi P_\phi Y = -2Y^T(I - P_\phi)\dot{P}_\phi \Phi \Phi^+ Y = \\ &= -2Y^T(I - P_\phi)\dot{\Phi} \Phi^+ Y. \end{aligned} \quad (20)$$

将式(17)代入式(20)便得到式(18). 另外将式(17)代入式(15)可推出式(19), 定理证毕.

式(18)给出了  $\hat{J}(\omega)$  对  $\omega$  的导数计算公式, 该公式不需要计算  $\Phi(\omega)$  的广义逆阵, 这为基于梯度信息的各种学习算法带来了方便. 下面给出一种基于梯度法的参数分离学习算法:

步骤 1. 随机选取  $\omega(0) \in R^{p \times q}$ , 确定步长  $h$ ;

步骤 2. 假定已得到  $\omega(k)$ , 计算  $\theta(k), Y_2$ ;

步骤 3. 计算并判断  $\hat{J} = Y_2^T Y_2 < \epsilon$ , 若成立,  $\omega(k), \theta(k)$  为最优点, 否则转下一步;

步骤 4. 按式(18)计算  $\frac{\partial \hat{J}}{\partial \omega_{ij}}$  依下式进行迭代并转第二步,

$$\omega_{ij}(k+1) = \omega_{ij}(k) - \frac{\partial \hat{J}}{\partial \omega_{ij}} h, \quad i = 1, 2, \dots, p; j = 1, 2, \dots, q. \quad (21)$$



关于该算法的几点说明:

1) 算法第一步的初始值选取, 可以不用随机的方式. 对于模糊神经网络可以根据先验知识予以确定.

2) 算法第二步求取  $\theta(k)$  有多种方法<sup>[7]</sup>: 一是可以用式(14)来做, 但这需要计算一个广义逆阵, 计算广义逆阵现也有许多成熟的算法; 二是利用式(17)的分解得到, 像零空间的分解也有许多算法; 三是直接求解下面的  $q \times q$  方程组,

$$\Phi^T(X, \omega)\Phi(X, \omega)\theta = \Phi^T(X, \omega)Y. \quad (22)$$

3) 计算步长可用自适应的方法调整. 在学习的起始阶段取大一些, 进入到极小点附近时取小一些.

4) 式(21)的参数调整是按梯度法进行的, 此外还可采用共轭梯度法, 变尺度法等.

## 4 仿真实例

给定非线性系统

$$y = 0.1e^{(2x_1 - 0.5)^2} + 0.1e^{(2x_2 - 0.5)^2} + \sin(2\pi x_1) - \cos(2\pi x_2) - (x_1 - 0.75)^2 - (x_2 - 0.25)^2, \quad x_1, x_2 \in [0, 1] \quad (23)$$

下面以五种网络模型来逼近式(23)这个非线性系统.

1) 三层感知器(PR): 输入与输出层为线性层, 隐层为非线性层, 激励函数为 Sigmoid 函数, 参数初始值随机选取.

2) 模糊神经网络(FNN  $i, i=0, 1, 2, 3$ ): 局部模型取为多项式函数,  $i$  为多项式的阶次,  $i=0$  即为通常的模糊神经网络. 采取单值模糊化、乘积推理、中心平均去模糊, 高斯型隶属函数,  $r=q$ . 参数初始值随机选取.

参数学习算法选取两种: 运用较多的反向传播算法<sup>[8]</sup>(BP)和本章的参数可分离学习算法(PSL). 每次学习均随机选取 100 个样本, 然后等间隔选取 100 个样本进行检验. 经过多次学习得到一组典型的仿真数据, 如表 1 所示, 表中  $MEAN(e)$  ( $MEAN(\bar{e})$ ),  $STD(e)$  ( $STD(\bar{e})$ ) 分别表示训练拟合误差(检验拟合误差)的均值、标准差,  $K$  为迭代次数,  $T$  为迭代过程的计算机运行时间. 图 1 至图 4 为 4 组典型的仿真结果图. 从上述结果可看出, 采用参数可分离学习算法大大加快了学习进程, 而且得到的网络具有较好的泛化能力.

表 1 仿真数据

No.	Network	$K$	$MEAN(e)$	$STD(e)$	$MEAN(\bar{e})$	$STD(\bar{e})$	$T(s)$
1	PR, PSL, $q=15$	300	0.0000	0.0341	0.0261	0.1104	73.82
2	PR, BP, $q=15$	1800	0.0000	0.2859	0.1822	0.8704	402.61
3	FNN0, PSL, $q=15$	300	0.0000	0.0179	0.0279	0.1087	54.65
4	FNN0, BP, $q=15$	1800	0.0000	0.1000	0.0203	0.2104	280.07
5	FNN3, PSL, $q=5$	300	0.0000	0.0101	-0.0001	0.0262	101.06
6	FNN3, BP, $q=5$	1800	-0.0021	0.1000	0.0195	0.1570	210.64
7	FNN2, PSL, $q=5$	300	-0.0000	0.0183	0.0108	0.0678	47.46
8	FNN2, BP, $q=5$	1800	-0.0012	0.1056	0.0414	0.1684	164.12
9	FNN1, PSL, $q=5$	300	-0.0000	0.0557	0.0116	0.1770	25.54
10	FNN1, BP, $q=5$	1800	-0.1677	0.2031	-0.1091	0.2840	120.12
11	FNN0, PSL, $q=5$	300	-0.0000	0.2117	0.0734	0.3717	16.43
12	FNN0, BP, $q=5$	1800	0.1229	0.3010	0.1541	0.3477	91.39

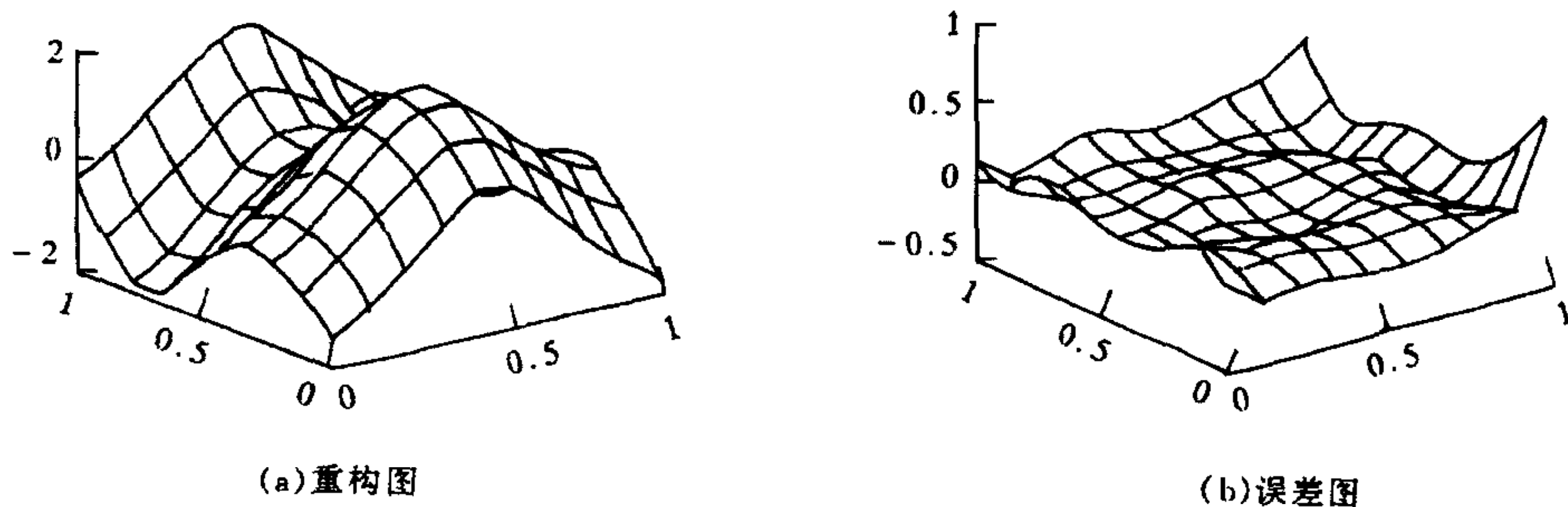


图 1 PR, PSL,  $q=15$  时的仿真结果

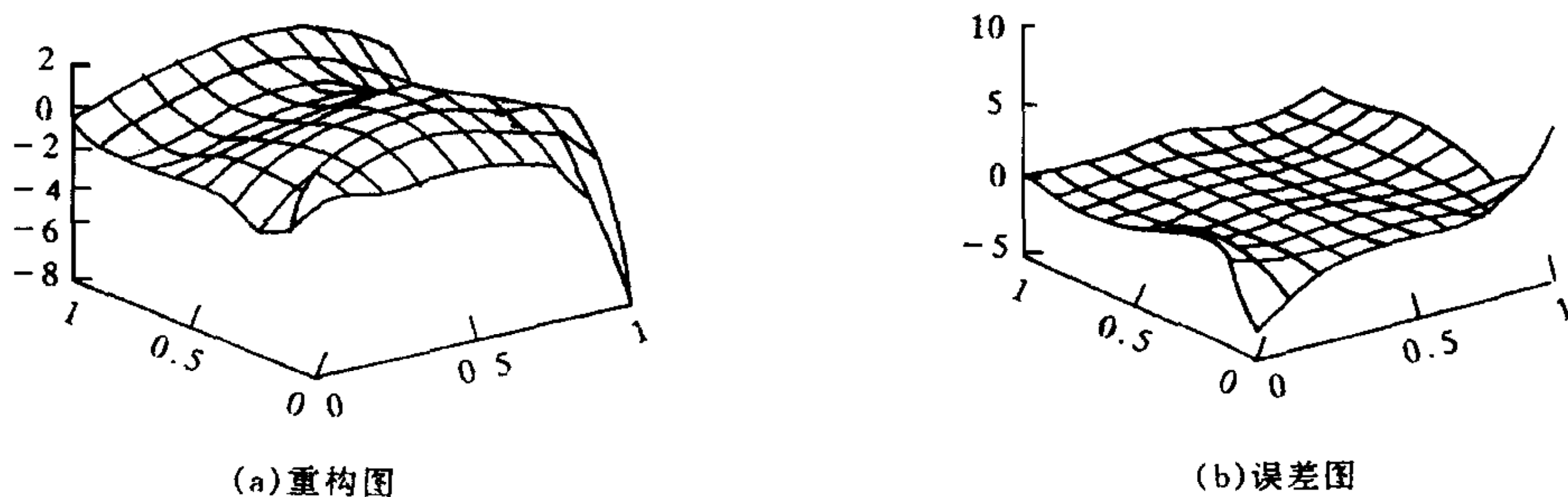


图 2 PR, BP,  $q=15$  时的仿真结果

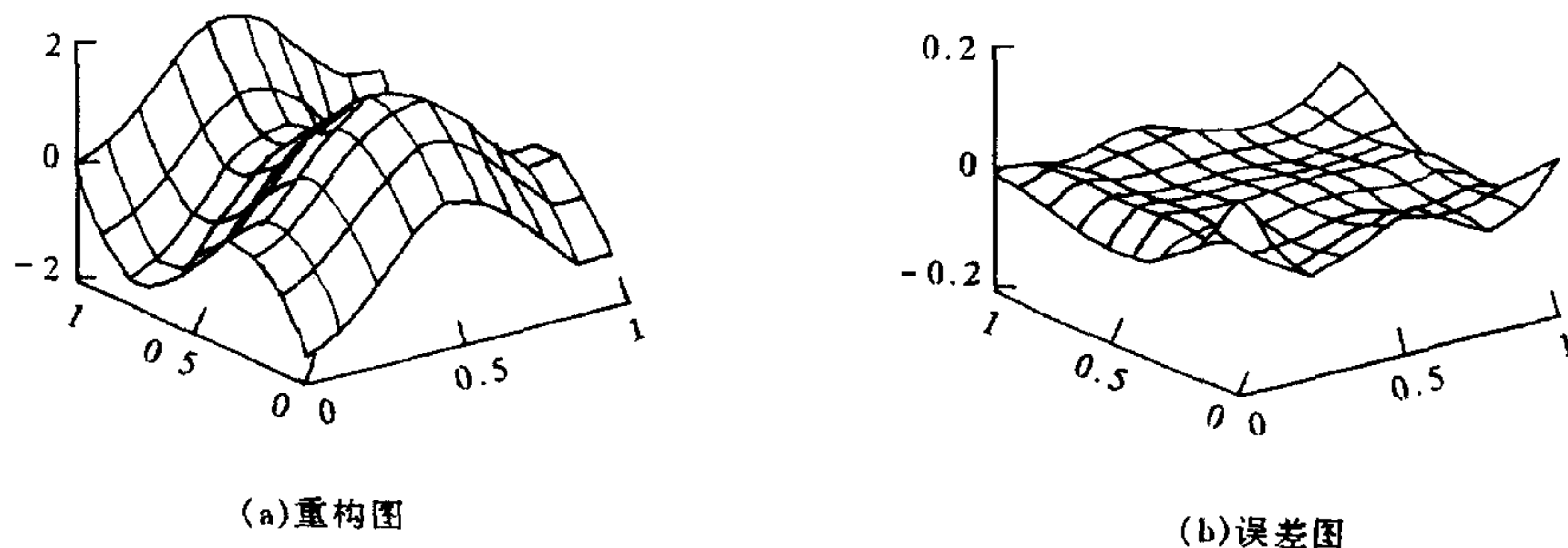


图 3 FNN3, PSL,  $q=5$  时的仿真结果

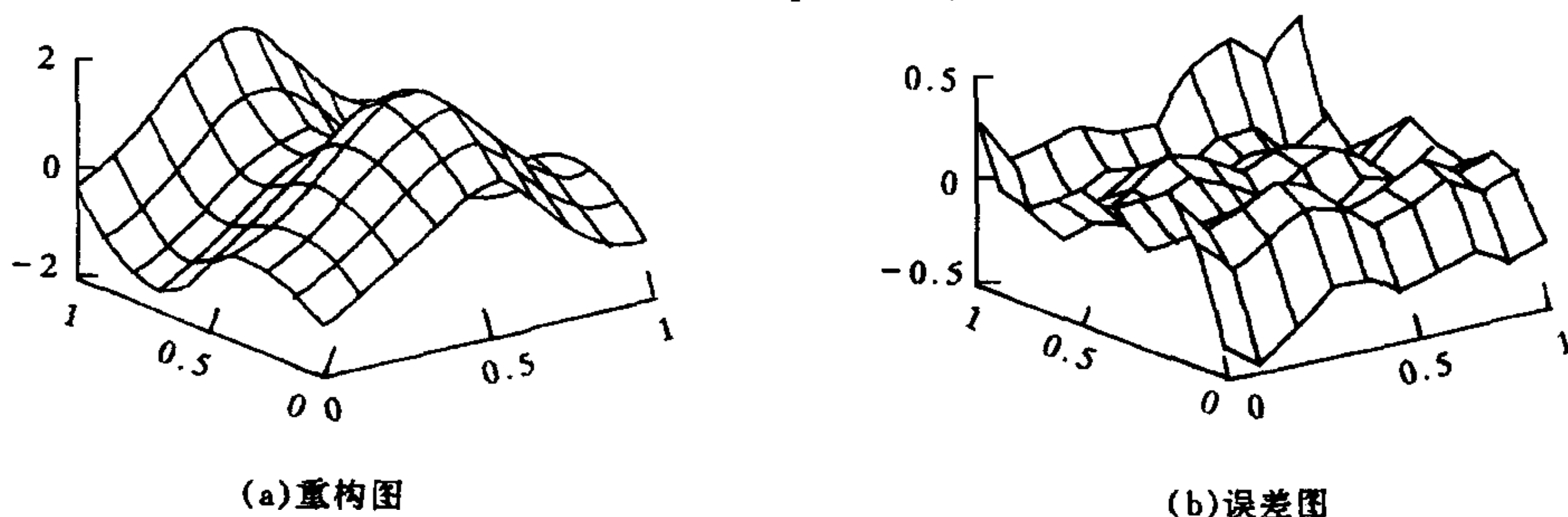


图 4 FNN3, BP,  $q=5$  时的仿真结果

### 5 结束语

本文在几种典型的神经网络模型统一数学描述的基础上,提出了一种参数可分离学习算法.由于只需要一部分参数参与迭代学习,从而加快了学习的进程;另一方面,由于参



与迭代的过程是沿着一个较优的超曲面进行的,因而用于迭代的步长可以加大,进而加快了学习的速度,这一点在仿真过程中可明显的感知.

### 参 考 文 献

- 1 Horinik K, Stinchcombe M, White H. Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks. *Neural Networks*, 1990, **3**:551—560
- 2 Williamson R C, Helmke U. Existence and uniqueness results for neural network approximation. *IEEE Trans. Neural Networks*, 1995, **6**(1):2—13
- 3 Chen T, Chen H. Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems. *IEEE Trans. Neural Networks*, 1995, **6**(4):911—917
- 4 Moody J, Darken C. Learning with localized receptive fields. in: Proc. 1988 connectionist models summer school, Carnegie Mellon Univ. ,1988
- 5 Specht D F. A general regression neural network. *IEEE Trans. on Neural Network*, 1991, **2**(6):568—576
- 6 Wang L X. Adaptive fuzzy systems and control——design and stability analysis. Prentice-Hall, Inc. ,USA, 1994
- 7 黄琳. 系统与控制理论中的线性代数. 科学出版社, 1984
- 8 Lin C T. Neural fuzzy control systems with structure and parameter learning. World Scientific, Singapore, 1994

## A PARAMETER-SEPARABLE LEARNING ALGORITHM FOR MULTILAYER FEEDFORWARD NEURAL NETWORKS

ZHANG YUN

(Research Institute of Automation, Guangdong University of Technology, Guangzhou 510090)

MAO ZONGYUAN

(Department of Automation, South China University of Technology, Guangzhou 510641)

YANG YIMIN

(Research Institute of Automation, Guangdong University of Technology, Guangzhou 510090)

**Abstract** Most learning algorithms simultaneously process all the parameters of the neural network. This often needs a large amount of time if the neural network is large. Many neural networks, such as perceptrons, radial basis function networks, general regression neural networks and fuzzy neural networks, may be regarded as a kind of multilayer feedforward neural networks. Their input-output mapping can be expressed as a linear combination of variable basis functions. Parameters of these neural networks can also be divided into two kinds: nonlinear parameters of the variable basis function and the coefficients of the linear combination. Based on this a parameter-separable learning algorithm is proposed. Simulation results show that the algorithm can accelerate the learning process and improve the approximating quality of the neural network.

**Key words** Neural network, learning algorithm, identification, parameter-separable.

**章 云** 1963年生. 副教授. 1982年、1986年先后在湖南大学获学士、硕士学位, 随后留校任教, 1989年调入广东工业大学, 现在华南理工大学攻读在职博士. 研究领域: 智能控制、机器人、离散事件动态系统.

**毛宗源** 1936年生. 1962年7月毕业于大连工学院. 现为华南理工大学自动化系教授, 博士生导师. 研究兴趣包括工业自动化, 智能控制. 已出版著作6本, 发表论文70篇.

**杨宜民** 1945年生. 教授, 博导. 1969年毕业于广东工学院, 毕业后留校任教, 先后在国内东北大学、北方工业大学、中山大学研习四年. 在日本筑波大学、东京大学从事“微驱动器、微机器人”研究二年半. 现为广东工业大学自动化研究所所长. 研究领域: 自动控制、机器人.

### 中国自动化学会、香港城市大学 CIDAM 研究中心联合召开 自动化科学技术应用学术会议征文通知

为交流和推进自动化应用技术的发展, 中国自动化学会与香港城市大学智能设计、自动化与制造研究中心(CIDAM)拟于1998年4季在香港城市大学联合召开自动化科学技术应用学术会议. 欢迎广大自动化专业科技人员踊跃投稿, 论文录取后即编入论文集(将按所占版面酌收部份印刷费).

#### 一、征文范围

1. 自动化应用技术、理论、方法与装备及在国民经济各产业部门控制和管理中的应用;
2. 各种检测、驱动装置、工业控制系统、现场总线、工作站及其网络系统, 能取得显著技术经济效益的各种控制和管理系统;
3. 先进控制与优化理论方法和系统在工业自动化中的应用.

#### 二、征文要求

1. 论文未公开发表过.
2. 论文内容应是已投入运行并显示出先进性或有显著经济效益的.
3. 凡投寄来的稿件, 要求一律用计算机按照下列要求打印出一式两份寄来.
  - 1) 用 B5 纸打印, 版芯尺寸: 即纸面排文字、图表、页码部分大小, 高 220mm×147mm, 按 5 号宋体字排, 每页 39 行, 每行 40 字.
  - 2) 文字、图的打印要求: 标题排在篇首, 用 3 号黑体字居中排, 作者姓名、单位, 排在标题下面, 姓名用小 4 号楷体, 单位用 6 号宋体; 论文正文全用 5 号宋体; 分节标题用小 4 号黑体, 居中排. 图和表格: 随文排入, 但都要在版芯尺寸内, 复印的图纸要求清晰、反差大、无底灰, 必须标有图名和表名.
  4. 希望论文不超过 6 页.
  5. 录取后的论文需按时提交用 WPS、WORD 排版后录入的 3"软盘和两份激光清稿.
  6. 来稿请注明详细地址及联系电话等.
  7. 凡被录取的论文作者, 应能出席本次学术会议, 因本次会议在香港召开, 要由中国科学技术协会下达出关任务书等手续, 由中国自动化学会通知作者所在单位办理出关手续. 费用由作者所在单位支付.

#### 三、论文截止日期: 1998 年 8 月 10 日, 并发出录取通知.

来稿请寄: 100071 北京 919 信箱 中国自动化学会应用专业委员会.

联系电话: (010)63812255—3382 传真: (010)63816990

中国自动化学会  
香港城市大学 CIDAM 研究中心

1998. 3