

基于多输入泛函网络的构造和学习策略^{*})

崔明义 张新祥 苏白云 张 瑞

(河南财经学院计算机科学系 郑州 450002)

摘 要 泛函网络是类似于人工神经网络的新型网络模型,是泛函方程的网络表达形式。本文针对复杂泛函网络构造和学习中存在的问题,提出了多输入泛函网络模型 MIOFN。通过对该模型的分析,提出了简化和学习的方法,并进行了仿真实验。结果表明,本文提出的 MIOFN 运行是可靠的,在工程应用中是有效可行的。

关键词 泛函网络, MIOFN, 泛函方程, 拓扑结构, 学习

Strategy of Structuring and Learning Based on Multi-Input and Single Output Functional Network

CUI Ming-Yi ZHANG Xin-Xiang SU Bai-Yun ZHANG Rui

(Dept. of Computer Science, Henan University of Finance & Economics, Zhengzhou 450002)

Abstract Functional network is new network model. It is similar to artificial neural network and is network expression of functional equation. In this paper, a multi-input and single output FN were presented for solving problems in structuring and learning of complex FN. Through the model was analyzed, a new simplifying and learning method was put forward by it. The simulating experiment was done by it. The result indicates that running of the model is reliable. It is applied to engineering availably and feasibly.

Keywords Functional network, MIOFN, Functional equation, Topology structure, Learning

1 引言

泛函网络(Functional Network, 即 FN)是 Castillo 在 1998 年提出的一种网络模型^[1],是人工神经网络(Artificial Neural Network, 即 ANN)的一种拓展,仅处理一般的泛函模型。它与 ANN 的主要区别在于它的泛函性质和无权连接。一个泛函网络的拓扑结构,由于其神经元函数的不固定和可学习性,往往能够描述一个函数簇(如 Fourier 展开级数、多项式、三角函数等)。近几年,泛函网络得到了较为广泛的应用,如局部动态建模^[2],非线性回归和混沌时间序列预测^[3],微分、差分和泛函方程求解^[4]等领域。这些成功的应用实例都显示了泛函网络的良好性能。

但是,泛函网络的局限性也是十分明显的。比如,一个泛函网络只能表示一个函数变换系统(尽管它可以描述一个函数簇),而一些工程问题的输出却不止一个。对于层次泛函网络^[5],其描述也是基于两种基本的泛函网络模型 SIOFN 和 DIOFN,而工程上所遇到的问题却相当复杂, SIOFN 和 DIOFN 的网络结构很难解决这类问题。本文仅对这一问题提出一种新的基于 MIOFN(Multi-Input and Single Output Functional Network)的泛函网络模型,以求为满足工程需要,用泛函网络解决工程问题提出一种新的思路。本文的理论分析和实验结果表明, MIOFN 用于解决工程问题是有效、可行的。

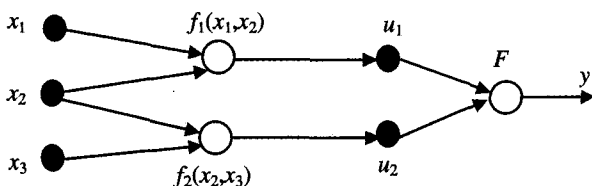


图 1 一种泛函网络的拓扑结构

2 泛函网络的拓扑结构

泛函网络对应的是泛函变换,它的拓扑结构描述的是一个函数变换系统。一般地,它主要由 5 部分组成。

- (1)输入层:该层包含有输入信息,如图 1 中的 x_1, x_2, x_3 。
- (2)存储层:该层的单元不具任何处理功能,只存储处理过程中的信息并使不同处理单元的输出相等。该层可以是一层,也可以是多层,如图 1 中的 u_1, u_2 。
- (3)处理层:该层对数据进行加工处理,可以是一层,也可以是多层,层间处理单元并非一定邻接,如图 1 中的 f_1, f_2 。
- (4)输出层:该层由含有输出信息的输出单元组成,如图 1 中的 y 。
- (5)一组连接:将输入层、存储层、处理层、输出层的神经元用箭头连接起来,箭头表示信息的流动方向。

图 1 所示的泛函网络是一种较简单的网络,但具备有泛函网络的所有功能组成。在实际应用中,根据问题的不同设计不同的泛函网络,完成相应的函数变换。函数变换系统越复杂,其泛函网络的拓扑结构也越复杂。但无论泛函网络如何复杂,都可将其拓扑结构看成是若干子泛函网络模型的“集成”。这样,我们就可将各种复杂泛函网络分解成若干个子泛函网络进行研究,分析各子网络的特性,研究其对整个网络的影响,探讨其处理层神经元学习算法,使泛函网络更趋近于各种工程实际,扩展泛函网络在工程中应用的范围,促进泛函网络的理论和应用研究。

3 MIOFN 模型

3.1 基本 MIOFN 模型

函数变换的类型很多,所以表示函数变换的泛函网络模

^{*}河南省自然科学基金(0411013800,0411014500)和河南省高校杰出科研人才创新工程项目(2004KYCX014)资助。崔明义 教授,博士,研究方向:计算智能、软件和网络。

型也很多,其拓扑结构也千姿百态,很难构造一个通用的泛函网络来表示所有的函数变换。为研究方便,本文从基本的泛函网络模型进行研究,而将各种泛函网络看成是基本泛函网络模型的组合。

定义1 设 n 维和 m 维实数空间存在函数变换 $F: R^n \rightarrow R^m$, 而 $X \in R^n$, 则泛函方程为

$$F = F(f_1, f_2, \dots, f_m) \quad (1)$$

其中 $X = (x_1, x_2, \dots, x_n)$, $f_1 = f_1(x_1, x_2, \dots, x_n)$, $f_2 = f_2(x_1, x_2, \dots, x_n)$, \dots , $f_m = f_m(x_1, x_2, \dots, x_n)$ 。泛函方程(1)所描述的泛函网络模型为基本 MIOFN 模型。

MIOFN 的拓扑结构如图 2 所示。显然, MIOFN 能够描述所有的一次函数变换。当 X 是 R^n 的线性向量, 且 $f_i = \sum_{j=1}^n a_{ij}x_j + c_i$ ($i=1, 2, \dots, m$, a_{ij}, c_i 分别为系数和常量), $F = (f_1, f_2, \dots, f_m)$ 时, 泛函就变换成了线性泛函。

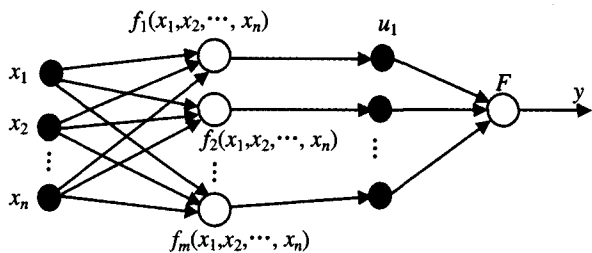


图 2 基本 MIOFN 模型的拓扑结构

特别地, 若 $n=m=1$, $f_i = a_i x$ ($i=1, 2, \dots, n$, a_i 为实数),

$F = \sum_{i=1}^n a_i x$, 该泛函就是最简单的一元函数变换。

但是, 在工程中所遇到的问题往往不是如此简单。有些问题若用泛函网络来描述, MIOFN 的神经元函数就比较复杂。有些复杂的泛函方程, 需要多次函数变换来完成。

3.2 MIOFN 的简化

我们根据函数变换构建的 MIOFN 还可进行简化, 以有利于问题的解决。初始 MIOFN 的简化依赖于泛函方程^[6]来完成。对一个给定的 MIOFN, 应分析判断是否存在对应于该 MIOFN 的输入都有相同输出的等价泛函网络。如果对任何一组给定输入, 两个 MIOFN 都有相同的输出, 则这两个 MIOFN 是等价的。对一个 MIOFN, 可以将与其等价的所有 MIOFN 作为一个等价类, 也就是与该 MIOFN 等价的 MIOFN 的集合。在实际应用时, 从该集合中找出最简单的 MIOFN 来使用, 以简化算法的运行。

例如, 有一泛函方程

$$f_4(x, y) = f_3(f_1(x, z), f_2(y, z)) \quad \forall x, y, z \quad (2)$$

其对应的泛函网络如图 3 所示。

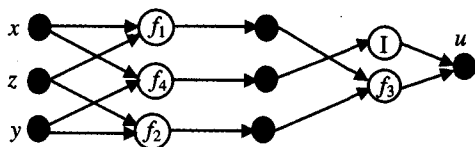


图 3 对应于式(2)的泛函网络

式(2)给出了函数 f_1, f_2, f_3 和 f_4 的约束条件, 可使其泛函结构简化。泛函方程(2)在实数矩形连续的一般解为

$$f_4(x, y) = F_3^{-1}(F_1(x) + F_2(y)), f_3(x, y) = F_3^{-1}(F_4(x) + F_5(y))$$

$$f_1(x, y) = F_4^{-1}(F_1(x) + F_6(y)), f_2(x, y) = F_5^{-1}(F_2(x)$$

$$- F_6(y)) \quad (3)$$

这里, F_1, \dots, F_6 是任意连续且严格单调函数。

将式(3)代入式(2), 得到泛函方程(2)的最简单表达式:

$$u = f_4(x, y) = f_3(f_1(x, z), f_2(y, z)) = F_3^{-1}(F_1(x) + F_2(y)) \quad (4)$$

简化后的泛函网络如图 4 所示。

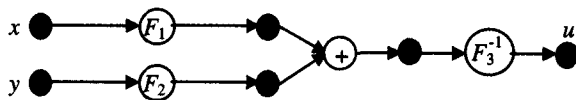


图 4 式(2)简化后的泛函网络

这样, 泛化网络的学习就可简化成用数据 (x_i, y_i, u_i) ($i=1, 2, \dots, n$) 评价神经元函数 F_1, F_2 和 F_3 。为此, F_j 可表示为给定基函数簇 $\varphi_j = \{\varphi_{j1}, \dots, \varphi_{jm_j}\}$ 的线性组合, 即

$$\hat{F}_j(x) = \sum_{i=1}^{m_j} a_{ji} \varphi_{ji}(x); j=1, 2, 3 \quad (5)$$

这里, 系数 a_{ji} 是泛函网络的参数(详见文[7]和[8])。

4 网络学习

同 ANN 一样, MIOFN 也要学习。所不同的是 MIOFN 不是权的学习, 而是结构和参数的学习。实际上, 前面所述的根据具体工程问题构建泛函网络和简化泛函网络也是学习。一旦 MIOFN 简化以后, 下面的学习就是神经元函数的学习, 对式(4)就是参数学习。学习的目的就是求出神经元函数的精确表达式或近似表达式。因而有两种学习方式: 精确学习和近似学习。

4.1 精确学习

精确学习就是确定 MIOFN 所表示的泛函方程的解函数。仍以 3.2 节简化后的泛函网络为例。我们考察该网络, 假设又有一个相关运算 $x \oplus y = h(x, y) = xy$, 图 4 表示了这个相关运算的网络拓扑结构。精确学习就是确定对应的神经元函数 F_1, F_2 和 F_3 。为此, 可加入其它一些泛函单元到泛函网络中。如图 5 所示, 这里 $h(x, y) = xy$ 。这时 F_3 满足泛函方程

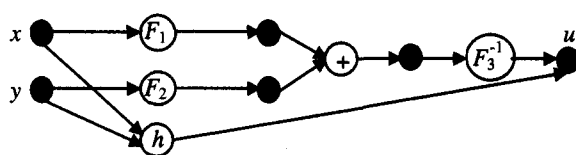


图 5 泛函网络的精确学习

$$F_3^{-1}[F_1(x) + F_2(y)] = xy \Leftrightarrow F_3(xy) = F_1(x) + F_2(y) \quad (6)$$

如果 $F_1 = F_2 = F_3 = f$, 这就是一个柯西方程, 其解为

$$f(x) = c \log x \Leftrightarrow f^{-1}(x) = \exp\left(\frac{x}{c}\right) \quad (7)$$

这里 c 是任意常数。对于任意 $c \neq 0$, 都产生一个乘法运算。

一旦 F_1, F_2 和 F_3 已知, 就可将泛函单元 h 从图 5 中移去, 恢复图 4 所示泛函网络, 恢复后的泛函网络产生 \oplus 运算的精确值。这种学习就是精确学习。

如果函数 h 未知, 或者不知道泛函方程的精确解, 则可用一些数据样本近似学习泛函函数。

4.2 近似学习

近似学习就是依据给定样本评价神经元函数。其基本方

法是线性组合基函数簇的函数,并且优化线性组合中的系数。有两种参数学习方法。

(1)线性方法:用相关优化函数产生一线性方程组,评价其系数。存在唯一一个最优值,解这个线性方程组,求出其最优参数。

(2)非线性方法:函数是非线性的。存在多个最佳参数,用梯度下降法迭代出最佳参数。例如,有一组数据样本 $D = \{(x_t, y_t, (x \oplus y)_t) | t \in T\}$, 可通过一组 k 函数 $\{g_1(x), g_2(x), \dots, g_k(x)\}$ 的线性组合 $\hat{f}(x) = \sum_{i=1}^k c_i g_i(x)$ 得到函数 f 近似值。并对近似值的误差进行度量,如用误差平方和等方法评价近似程度。以式(6)为例,数据点 t 的误差为

$$e_t = F_3((x \oplus y)_t) - F_1(x_t) - F_2(y_t) = \sum_{i=1}^k (c_i g_i((x \oplus y)_t) - c_i g_i(x_t) - c_i g_i(y_t))$$

然后,估计 $c_i (i=1, 2, \dots, k)$, 求相对于 c_i 的最小误差,误差平方和为

$$E = \sum_{t \in T} [\sum_{i=1}^k (c_i g_i((x \oplus y)_t) - c_i g_i(x_t) - c_i g_i(y_t))]^2 \quad (8)$$

这就是 MIOFN 的近似学习。

5 应用实例

5.1 建立解方程

仍以图 4 为例。假设对于输入 x, y 和输出 u , 构造的函数为

$$F_1(x) = \sum_{i=1}^m a_{1i} \phi_{1i}, F_2(y) = \sum_{i=1}^m a_{2i} \phi_{2i} \quad (9)$$

这里, m 是任意次数, ϕ 是多项式 $(1 x x^2 x^3 \dots)$ 表达式。则

$$F_3(u) = \sum_{i=1}^2 a_{3i} \phi_{3i} \quad (10)$$

由输入函数

$$\hat{F}_3(u) = F_1(x) + F_2(y) \quad (11)$$

得第 j 组数据的误差为

$$e_j = F_1(x_j) + F_2(y_j) - F_3(u_j) \quad (12)$$

写成矩阵形式

$$e_j = b_j a' \quad (13)$$

这里, $b_j = [1 x_j x_j^2 \dots 1 y_j y_j^2 \dots -1 u_j]$, $a' = [[a_{11} a_{12} \dots a_{21} a_{22} \dots a_{31} a_{32}]^T$ 。误差平方和为

$$E = \sum_{j=1}^n e_j^T e_j = a (\sum_{j=1}^n b_j' b_j) a' = a A a' \quad (14)$$

对于初始值 $\{x_0, y_0, u_0\}$, 均有唯一的解 $F_1(x_0) = \sum_{i=1}^{m_1} a_{1i} \phi_{1i}$

$$(x_0) = a_1, F_2(y_0) = \sum_{i=1}^{m_2} a_{2i} \phi_{2i}(y_0) = a_2 \text{ 和 } F_3(u_0) = \sum_{i=1}^{m_3} a_{3i} \phi_{3i}$$

$(u_0) = a_3$ 。写成矩阵形式:

$$a \begin{bmatrix} \{\phi_{10}\} & \{0\} & \{0\} \\ \{0\} & \{\phi_{20}\} & \{0\} \\ \{0\} & \{0\} & \{\phi_{30}\} \end{bmatrix} - [a_1 \ a_2 \ a_3] = 0 \quad (15)$$

即 $a \Phi_0 - a = 0$ 。用拉格朗日乘数定义扩展函数 R , 并微分求极小值, 可得到对于任意 $\{x_i, y_i\}$ 求 u_i 的公式

$$F_3(u_i) = F_1(x_i) + F_2(y_i) = a_{31} + a_{32} u_i \quad (16)$$

即 $u_i = [F_3(u_i) - a_{31}] / a_{32}$ 。

5.2 仿真实验

使用文[9]中有限元分析的数据进行仿真实验。有模口角度和冲击速度两个输入, 分别用 x, y 表示, 输出锻造负载用 u 表示。所有数据都相对于其最大值 100、1400、400 进行了归一化处理。初始值取 $x_0 = y_0 = u_0 = 0.3, a_1 = a_2 = a_3 = 0.2$ 。输入用二次函数, 输出用一次函数。MIOFN 简化后的拓扑结构如图 4 所示。归一化数据和 MIOFN 的仿真数据见表 1。

MIOFN 仿真效果如图 6 所示。

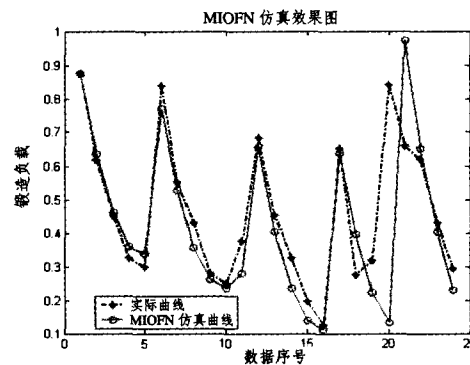


图 6 有限元的 MIOFN 仿真效果

从表 1 和图 6 可知, MIOFN 仿真刚开始与实际数据拟合较好, 后来偏差稍大, 这是由于数据噪声影响了 MIOFN 的稳定性。限于篇幅, 这里不对 MIOFN 的稳定性做进一步的分析。

从实验结果不难看出, MIOFN 是完全满足工程需求的, 这也说明了 MIOFN 作为泛函网络的基本模型, 有着较大的应用空间。

表 1 MIOFN 仿真数据表

NO.	x	y	u	MIOFN	NO.	x	y	u	MIOFN
1	0.900	0.286	0.875	0.876	13	0.450	0.429	0.455	0.405
2	0.900	0.429	0.618	0.635	14	0.450	0.571	0.325	0.236
3	0.900	0.571	0.450	0.462	15	0.450	0.714	0.195	0.138
4	0.900	0.714	0.325	0.360	16	0.450	0.857	0.125	0.112
5	0.900	0.857	0.300	0.339	17	0.300	0.286	0.650	0.638
6	0.750	0.286	0.838	0.771	18	0.300	0.429	0.275	0.396
7	0.750	0.429	0.550	0.528	19	0.300	0.571	0.318	0.223
8	0.750	0.571	0.430	0.356	20	0.300	0.714	0.840	0.132
9	0.750	0.714	0.280	0.262	21	0.300	0.857	0.660	0.973
10	0.750	0.857	0.250	0.236	22	0.150	0.286	0.620	0.650
11	0.600	0.571	0.375	0.281	23	0.150	0.429	0.430	0.403
12	0.450	0.286	0.683	0.656	24	0.150	0.571	0.293	0.231

结论 泛函网络是人工神经网络的一种拓展, 它又与人工神经网络有着完全不同的训练方法。它的实质是泛函方程的网络表达。本文通过对泛函网络的分析, 提出了 MIOFN

的泛函网络模型, 对 MIOFN 进行了理论分析, 阐述了构造和简化 MIOFN 的方法, 研究了神经元函数的学习和参数优化 (下转第 192 页)

表2 实验1中不同学习模型在不同模型参数下的分类误差

学习模型	模型参数	训练误差	测试误差
KBNNE	$g=1, C=100$ 隐层节点数 20	0.0%	1.0%
	$g=0.5, C=100$ 隐层节点数 20	2.73%	4.18%
SVMs	$g=1, C=100$	1.82%	1.91%
	$g=0.5, C=100$	5.45%	5.91%
Neural Network Ensembles	隐层节点数 20	10.91%	12.45%

训练样本个数(20,30,40,50)来检测模型的泛化能力。测试样本个数都是50。我们将KBNNE与基于Shepard插值的核函数方法、SVM方法和神经网络集成方法进行了对比测试。实验结果见图4。

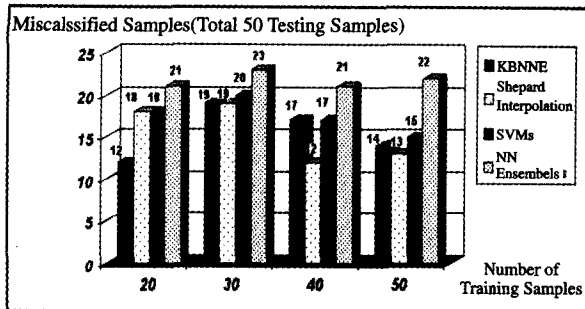


图4 KBNNE模型与其它学习算法的对比

可以看出,基于知识发现的神经网络集成模型KBNNE的泛化性能显得很稳定。尽管有时候在所有方法中它的判别准确率不是最高的,但也远远高于一般的神经网络集成的方法,并且与最佳值比较接近。

结论 本文从知识发现的角度出发,以构造神经网络集成的方式将知识发现技术与神经网络训练算法进行结合,实现了归纳学习和演绎学习的有效互补。与传统的EBNN模型不同,本文提出的KBNNE模型以并行操作的方式建立了知识的描述方式和网络训练之间内部的联系,通过对训练数据的动态变换,不仅增强了模型的适用范围,而且对于解决神经网络以何种方式应用知识,如何理解网络计算的“可靠性”、“透明度”等有关神经网络本质的问题都是有益的探索。

参考文献

- Haykin S. Neural Networks: A Comprehensive Foundation (2nd edition). Prentice-Hall, New Jersey, 1999
- Duda R O, Hart P E, Stork D. Pattern Classification (2nd edition). New York: John Wiley, 2001
- Fayyad U M, Shapiro G P, Smyth P, Uthurusamy R. Advances

- in Knowledge Discovery and Data Mining. American Association for Artificial Intelligence, Menlo Park, CA, USA, 1996
- 郭萌,王珏. 数据挖掘与数据库知识发现:综述. 模式识别与人工智能, 1998, 11(3): 292~299
- Fayyad U M, Shapiro G P, Smyth P. The KDD Process for Extracting Useful Knowledge from Volumes of Data. Communications of the ACM, 1996, 39(11): 27~34
- Fayyad U M. Data Mining and Knowledge Discovery: Making Sense Out of Data. IEEE Expert, 1996, 11(5): 20~25
- Mitra S, Pal S K, Mitra P. Data Mining in Soft Computing Framework: A Survey. IEEE Trans. on Neural Networks, 2002, 13(1): 3~14
- 员巧云,程刚. 近年来我国数据挖掘研究综述. 情报学报, 2005, 24(2): 250~256
- Lee M L, Hsu W. Improving data quality: eliminating dupes & I-D-ing those spurious links. IEEE Potentials, 2005, 24(2): 35~38
- Alimonte D D, Zibordi G. Statistical Assessment of Radiometric Measurements From Autonomous Systems. IEEE Trans. on Geoscience and Remote Sensing, 2006, 44(3): 719~728
- Cervellera C, Muselli M. Deterministic Design for Neural Network Learning: An Approach Based on Discrepancy. IEEE Trans. on Neural Networks, 2004, 15(3): 533~544
- Tenenbaum J B, Silva V, Langford J C. A Global Geometric Framework for Nonlinear Dimensionality Reduction. Science, 2000, 290(5500): 2319~2323
- Roweis S T, Saul L K. Nonlinear Dimensionality Reduction by Local Linear Embedding. Science, 2000, 290(5500): 2323~2326
- Arabie P, Hubert L J, Soete G D. Clustering and Classification. World Scientific, 1996
- Kaufman L, Rousseeuw P J. Finding Groups in Data: An Introduction to Cluster Analysis. Wiley Series In Probability And Mathematical Statistics, New York: John Wiley, 2005
- Cao L J, Tay F E H. Support Vector Machine with Adaptive Parameters in Financial Time Series Forecasting. IEEE Trans. on Neural Networks, 2003, 14(6): 1506~1518
- Awate S P, Whitaker R T. Unsupervised, Information-Theoretic, Adaptive Image Filtering for Image Restoration. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2006, 28(3): 364~376
- Tholudur A, Ramirez W F, McMillan J D. Interpolated Parameter Functions for Neural Network Models. Computers and Chemical Engineering, 2000, 24: 2545~2553
- 孙晓军,丁群,朱勇. 乘窗信号谱分解及其数据修复. 电子学报, 2005, 33(3): 538~540
- Little R J A, Rubin D B. Statistical Analysis with Missing Data (2nd edition). Wiley Series In Probability And Statistics, New York: John Wiley, 2002
- 周志华,陈世福. 神经网络规则抽取. 计算机研究与发展, 2002, 39(4): 398~405
- Jordan M I, Sejnowski T J. Graphical Models: Foundations of Neural Computation. MIT Press, 2001
- Garson G D. Interpreting neural-network connection weights. AI Expert, 1991, 6(4): 47~51
- Keller T, Gerjets P, Scheiter K, et al. Information visualizations for knowledge acquisition: The impact of dimensionality and color coding. Computers in Human Behavior, 2006, 22: 43~65
- Bergadano F, Giordana A, Saitta L. Machine Learning: An Integrated Framework and its Application. Ellis Horwood Series in Artificial Intelligence, Ellis Horwood, 1991
- Thrun S. Explanation-Based Neural Network Learning: A Lifelong Learning Approach. Kluwer Academic, Boston, 1996
- Opitz D, Shavlik J. Actively searching for an effective neural network ensemble. Connection Science, 1996, 8(3-4): 337~353
- 吴涛,贺函根,贺明科. 基于插值的核函数构造. 计算机学报, 2003, 26(8): 990~996

(上接第171页)

的算法。文中的分析和仿真实验的结果表明,本文提出的MIOFN运行是可靠的,在工程应用中是有效、可行的。

参考文献

- Castillo E. Functional networks[J]. Neural Processing Letters, 1998, 7: 151~159
- Noelia S M, Oscar F R, Amparo A B. Self-Organizing Maps and Functional Networks for Local Dynamic Modeling [C]. In: European Symposium on Artificial Neural Networks Bruges (Belgium). April 2003. 39~44
- Castillo E, Gutiérrez J M. Some Applications of Functional Networks in Statistics and Engineering [J]. Technometrics, 2001, 43(1): 10~24

- Castillo E, Cobo A, Gutiérrez J M. Working with Differential, Functional and Difference Equations Using Functional Networks [J]. Applied Mathematical Modeling, 1999, 23: 89~107
- 周永权,焦李成. 层次泛函网络整体学习算法. 计算机学报[J], 2005, 28(8): 1277~1286
- Castillo E, Ruiz-Cobo R. Functional Equations in Science and Engineering [M], New York: Marcel Dekker, 1992
- Castillo E. Functional networks [J]. Neural Processing Letters, 1998, 7: 151~159
- Castillo E, Gutiérrez J M. A Minimax Method for Learning Functional Networks [J]. Neural Processing Letters, 2000, 11: 39~49
- Zienkiewicz O C. Flow Formulation for Numerical Solution of Forming Processes [C]. Numerical Analysis of Forming Processes. New York: Wiley, 1984. 1~44