

基于多项式基函数的泛函网络构造方法与逼近理论^{*})

周永权¹ 刘晓冀¹ 张 明²

(广西民族大学数学与计算机科学学院 南宁 530006)¹ (大连水产学院理学院 大连 116023)²

摘 要 用构造性方法证明:对于给定的 r 阶多项式函数,可以具体地构造出一个三层泛函网络,以任意精度逼近该多项式,所构造的网络的中间神经元个数仅与多项式基函数的阶数 r 有关,并能用 r 表达。该文所得结果对于基于多项式基函数的泛函网络逼近任意函数类的网络具体构造和逼近具有理论指导意义。

关键词 多项式基函数,泛函网络,构造方法,逼近理论

Functional Network Construction Method and Approximation Theory Based on Polynomial Basis Functions

ZHOU Yong-quan¹ LIU Xiao-ji¹ ZHANG Ming²

(College of Computer and Information Science, Guangxi University for Nationalities, Nanning 530006, China)¹

(School of Science, Dalian Fisheries University, Dalian 116023, China)²

Abstract It is investigated in this paper that the polynomial basis functions are approximated by functional network with three-layer. It is shown that for a given polynomial function with r order a three-layer functional network can be constructed by a constructive method to approximate the polynomial to any degree of accuracy. The number of mid-unit of the constructed functional network only depends on the order of approximated basis functions. The obtained results are more important for constructing a three-layer functional network to approximate the class of polynomial basis functions and for realizing the approximation.

Keywords Polynomial basis functions, Functional network, Construction method, Approximation theory

1 引言

泛函网络是 Castillo 在 1998 年提出的一种网络模型^[1]。泛函网络处理的是一般泛函模型,它在各个神经元之间的连接无权重,并且神经元函数不固定,而是可学习的,常常是一个给定的函数簇(如多项式、三角函数、Fourier 展开级数等)的组合,在实际应用中,人们可根据特定的问题的需要来选取不同的基函数簇。泛函网络已经成功地被应用于非线性系统辨识,混沌时间序列的预测,微分、差分和泛函方程的求解, CAD, 线性、非线性回归^[2-6]等;泛函网络在解决上述问题中都表现出较好的性能。随着泛函网络理论的不断深入和应用范围的不断拓广,泛函网络在计算机代数^[7-15]、复杂系统建模^[16-19]、系统设计与优化^[20,21]等应用方面也取得了一些成功。

泛函网络虽然已经成功地被应用于很多方面,但由于它是近年来提出的一种新的对神经网络的推广,同神经网络一样,有些理论和应用方面的基础还不太健全,需要人们不断地提出更适合所要解决问题的新的网络结构,完善基础理论,提出一些新的系统模型及其逼近方法。基于这些想法,本文工作的重点是从理论上研究泛函网络的函数逼近性能,而函数的逼近是评价泛函网络计算能力的一个重要手段,泛函网络应用中的一个重要的基本问题是如何设计一个泛函网络结构与之有能力学习到给定的函数。近来较多的工作(如文献

[2])集中在针对泛函网络中一种特殊的模型,被称为可分离泛函网络^[5],从理论上反映了网络逼近性能与中间计算单元(相当于神经网络的隐层)结构之间的关系。然而,针对泛函网络一般模型结构,至今从理论上人们还没有具体给出实现函数逼近的具体方法,从而难应用到实际问题中去。因此限制了泛函网络的应用范围。而多项式函数是一种较为简单并且容易计算的函数,也是实际问题中最广泛使用的函数之一,它在表达一般的函数、构造样条及拟合曲面等方面也起到重要的作用,通过对多项式函数的研究与探讨,可以进一步研究一般函数的逼近问题,从而揭示泛函网络逼近一般函数的本质和实现逼近的机制。本文研究的目的是理论上给出基于多项式基函数神经元的三层泛函网络的构造过程和函数逼近方法,促进泛函网络的理论和应用研究。

2 函数逼近问题的描述

一般函数逼近问题可以描述为:如何确定一个函数集合 Φ ,使得特定空间 A 上的任意函数 f 可由 Φ 中函数近似地构造出来。现存在的基本问题是:(1)如何选择 Φ ;(2)如何基于 Φ 中函数构造近似函数,使得与目标函数有尽可能小的偏差。对函数逼近论的研究,已经产生出许多成熟的函数逼近方法,如代数多项式逼近、三角多项式逼近、插值方法等^[22,23]。经典的方法通常用回归分析,即已知采样数据集 $\{(x_i, y_i) | x_i \in R^n, y_i \in R, i \in N\}$,构造满足采样数据点误差等约束条件的

^{*} 基金项目:国家自然科学基金(60461001);广西自然科学基金(0542048; 0832082);国家民委科研项目(08GX01)和广西民族大学重大科研项目资助课题。周永权 博士,教授,主要研究方向为神经网络、计算智能及应用;刘晓冀 博士,教授,主要研究方向为矩阵广义逆计算及应用;张 明 硕士,主要研究方向为计算智能及应用。

函数,但存在着相关却不同的其它问题,如控制等领域,通常要求解特定的模型,而该模型是否适当,只有当模拟执行该模型后依据其适合性才能评判。回归分析不适合求解该问题,当前的计算智能技术为解决该问题提供了新的逼近方法,本文仅从理论上致力于用基于多项式基函数的泛函网络逼近模型来解决该问题,使其泛函网络函数逼近技术有理论基础,使其泛函网络逼近技术的应用范围得以推广。

3 泛函网络的构造

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,它主要由以下5部分组成:(1)输入单元层:这是输入数据的一层单元,输入单元以带有相应的名字的空心圆表示;(2)输出单元层:这是最后一层单元,它输出网络的结果数据,输出单元也是以带有相应的名字的空心圆表示;(3)一层或多层神经元或计算单元:每一个神经元是一个计算单元,它计算一组来自前一层神经元或输入单元的输入值,并给下一层神经元或输出单元提供一组输入数据。计算单元相互连接,每一个神经元的输出可以作为另一个神经元或输出单元数据的一部分,一旦给定输入值,输出便由神经元的类型确定,它由一函数定义。例如,假定我们有一个神经元具有 s 个输入 $(x_1, x_2, \dots, x_s)$ 及 k 个函数 $F_j, j=1, 2, \dots, k$;使得 $y_j = F_j(x_1, x_2, \dots, x_s), j=1, 2, \dots, k$ 。函数 F_j 由网络的结构确定,神经元由带有相应的 F_j 函数名称圆圈来表示;(4)有向连接线:它们将输入层、第一层神经元、第二层神经元、最后一层神经元及输出单元连接起来,箭头表示信息流动的方向,所有这些一起形成泛函网络的结构,它确定了网络的泛函能力。泛函网络的结构是指神经元的组织及包含的连接。

图1所示的是一种较简单的泛函网络拓扑结构图,但具备泛函网络的所有功能组成。为了便于比较,图2给出的是对应的神经网络拓扑结构图。

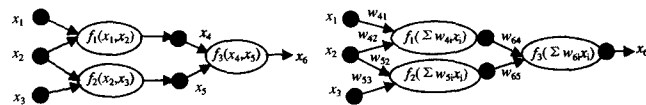


图1 一种泛函网络的拓扑结构

图2 与图1对应的神经网络拓扑结构

标准神经网络与泛函网络之间最明显的区别是:在标准神经网络模型中给定神经元的激活函数,而对激活的连接权值进行训练。在泛函网络模型中没有连接权值,泛函网络训练的是函数 F_j ,也即对激活函数进行训练。在标准神经网络模型中,输出单元是单个神经元的输出(最后一个单元),而在泛函网络模型中,输出单元可输出一个或多个神经元,这隐含了一种相容性,即连接到输出单元的神经元个数越多,输出神经元的自由度越少。因此,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。

4 逼近定理及其构造性证明

由于函数变换的类型很多,因此表示函数变换的泛函网络模型类型也很多,其拓扑结构的几何形状也各异,很难构造出一通用的泛函网络来表示所有的函数变换。为了研究方便,本文从一元函数变换入手,即考虑单输入单输出泛函网络结构的函数逼近问题,根据泛函网络结构的构造方法,一般的

单输入单输出泛函网络结构如图3所示。

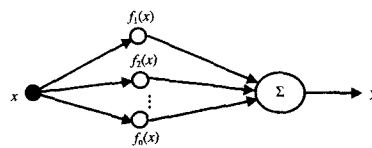


图3 单输入单输出三层泛函网络

其中 $f_i (i=1, 2, \dots, n)$ 是中间计算单元。此时,该泛函网络输出

$$y = \sum_{i=1}^n f_i(x) \quad (1)$$

是一元多项式函数表达式。根据Castillo的作法,我们可以将(1)式中的 $f_i(x)$ 表示成一个已知函数簇的线性组合的形式

$$f_i(x) = \sum_{j=1}^m a_{ij} \varphi_j(x) \quad (2)$$

式中,基函数集 $\Phi = \{\varphi_j(x), i=1, 2, \dots, n, j=1, 2, \dots, m\}$ 是给定的,适合于逼近 f_i 到期望精度的、线性独立的函数集,系数 a_{ij} 是泛函网络的参数。将(2)式代入(1)式有,该泛函网络的最终输出

$$y = \sum_{i=1}^n \sum_{j=1}^m a_{ij} \varphi_j(x) \quad (3)$$

引理 设 $\Phi = \{\varphi_j(x), i=1, 2, \dots, n, j=1, 2, \dots, m\}$ 是定义在 R 上具有 $r+1$ 次连续有界导数的线性独立的函数集,则当 $m=r+1$ 时,必存在实数 α ,使得矩阵

$$A = \begin{bmatrix} \varphi_1(\alpha) & \varphi_2(\alpha) & \cdots & \varphi_m(\alpha) \\ \varphi_1^{(1)}(\alpha) & \varphi_2^{(1)}(\alpha) & \cdots & \varphi_m^{(1)}(\alpha) \\ \vdots & \vdots & \cdots & \vdots \\ \varphi_1^{(r-1)}(\alpha) & \varphi_2^{(r-1)}(\alpha) & \cdots & \varphi_m^{(r-1)}(\alpha) \\ \varphi_1^{(r)}(\alpha) & \varphi_2^{(r)}(\alpha) & \cdots & \varphi_m^{(r)}(\alpha) \end{bmatrix}, 1 \leq i \leq n$$

是非奇异的,即 A^{-1} 存在。

证明:(用反证法)倘若对任意的实数 α ,均有

$$\det A = \det \begin{bmatrix} \varphi_1(\alpha) & \varphi_2(\alpha) & \cdots & \varphi_m(\alpha) \\ \varphi_1^{(1)}(\alpha) & \varphi_2^{(1)}(\alpha) & \cdots & \varphi_m^{(1)}(\alpha) \\ \vdots & \vdots & \cdots & \vdots \\ \varphi_1^{(r-1)}(\alpha) & \varphi_2^{(r-1)}(\alpha) & \cdots & \varphi_m^{(r-1)}(\alpha) \\ \varphi_1^{(r)}(\alpha) & \varphi_2^{(r)}(\alpha) & \cdots & \varphi_m^{(r)}(\alpha) \end{bmatrix} = 0$$

$$1 \leq i \leq n$$

即当 $m=r+1$ 时,

$$\det \begin{bmatrix} \varphi_1(x) & \varphi_2(x) & \cdots & \varphi_{r+1}(x) \\ \varphi_1^{(1)}(x) & \varphi_2^{(1)}(x) & \cdots & \varphi_{r+1}^{(1)}(x) \\ \vdots & \vdots & \cdots & \vdots \\ \varphi_1^{(r-1)}(x) & \varphi_2^{(r-1)}(x) & \cdots & \varphi_{r+1}^{(r-1)}(x) \\ \varphi_1^{(r)}(x) & \varphi_2^{(r)}(x) & \cdots & \varphi_{r+1}^{(r)}(x) \end{bmatrix} \equiv 0 \quad (x \in R) \quad (4)$$

当把(4)式左边的行列式展开后,我们可把(4)看作是是关于 $\varphi_{r+1}(x)$ 为未知函数的 r 阶微分方程,由于 $\varphi_1(x), \varphi_2(x), \dots, \varphi_r(x)$ 在 R 上线性独立,所以,易知它的通解为:

$$\varphi_{r+1}(x) = c_1 \varphi_1(x) + c_2 \varphi_2(x) + \cdots + c_r \varphi_r(x) \quad (5)$$

这里, $c_1, c_2, \dots, c_r$ 是 r 个独立的任意常数。但(5)式说明 $\varphi_1(x), \varphi_2(x), \dots, \varphi_{r+1}(x)$ 在 R 上线性相关,与已知条件矛盾。所以必存在实数 α ,使得矩阵 A 非奇异,即 A^{-1} 存在。证毕。

定理 设 $\Phi = \{\varphi_j(x), i=1, 2, \dots, n, j=1, 2, \dots, m\}$ 是定义在 R 上具有 $r+1$ 次连续有界导数的线性独立的函数集,则对任意给定的 r 阶多项式 $P_r(x)$ 及 $\varepsilon > 0$,可构造出一个输入、

一个输出及中间计算单元个数为 $r+1$ 的三层泛函网络(见图 3):

$$T_n(x) = \sum_{i=1}^n \sum_{j=1}^m a_{ij} \varphi_{ij}(x), a_{ij} \in R, m=r+1$$

使得

$$|T_n(x) - P_r(x)| < \varepsilon$$

证明: 设 $P_r(x) = b_0 + b_1x + \dots + b_r x^r$ 是给定的 r 阶多项式, 令

$$T_n(x) = \sum_{i=1}^n \sum_{j=1}^m a_{ij} \varphi_{ij}(x), a_{ij} \in R, m=r+1 \quad (6)$$

则我们适当地选取参数 $a_{ij}, i=1, 2, \dots, n, j=1, 2, \dots, m$; 使得

$$T_n(0) + T_n^{(1)}(0)x + \dots + \frac{1}{r!} T_n^{(r)}(0)x^r = b_0 + b_1x + \dots + b_r x^r = P_r(x) \quad (7)$$

事实上, 由于 $\{\varphi_{ij}(x), i=1, 2, \dots, n, j=1, 2, \dots, m\}$ 是定义在 R 上具有 $r+1$ 次连续有界导数的线性独立的函数集, 因此由(6)式可得

$$T_n^{(s)}(x) = \sum_{i=1}^n \sum_{j=1}^m a_{ij} \varphi_{ij}^{(s)}(x), s=0, 1, \dots, r \quad (8)$$

并可得到以下矩阵方程组:

$$\begin{bmatrix} \varphi_{11}(0) & \varphi_{12}(0) & \dots & \varphi_{1m}(0) \\ \varphi_{11}^{(1)}(0) & \varphi_{12}^{(1)}(0) & \dots & \varphi_{1m}^{(1)}(0) \\ \vdots & \vdots & \dots & \vdots \\ \varphi_{11}^{(r-1)}(0) & \varphi_{12}^{(r-1)}(0) & \dots & \varphi_{1m}^{(r-1)}(0) \\ \varphi_{11}^{(r)}(0) & \varphi_{12}^{(r)}(0) & \dots & \varphi_{1m}^{(r)}(0) \end{bmatrix} \begin{bmatrix} \sum_{i=1}^n a_{i1} \\ \vdots \\ \sum_{i=1}^n a_{im} \end{bmatrix} = \begin{bmatrix} b_0 \\ 1! b_1 \\ \vdots \\ (r-1)! b_{r-1} \\ r! b_r \end{bmatrix}$$

或可写成

$$AC = B \quad (9)$$

其中

$$A = \begin{bmatrix} \varphi_{11}(0) & \varphi_{12}(0) & \dots & \varphi_{1m}(0) \\ \varphi_{11}^{(1)}(0) & \varphi_{12}^{(1)}(0) & \dots & \varphi_{1m}^{(1)}(0) \\ \vdots & \vdots & \dots & \vdots \\ \varphi_{11}^{(r-1)}(0) & \varphi_{12}^{(r-1)}(0) & \dots & \varphi_{1m}^{(r-1)}(0) \\ \varphi_{11}^{(r)}(0) & \varphi_{12}^{(r)}(0) & \dots & \varphi_{1m}^{(r)}(0) \end{bmatrix},$$

$$C = (\sum_{i=1}^n a_{i1}, \dots, \sum_{i=1}^n a_{im})^T$$

$$B = (b_0, b_1, \dots, (r-1)! b_{r-1}, r! b_r)^T.$$

当 $m=r+1$ 时, A 是方阵, 由引理可知, 存在实数 α , 使得 A 非奇异。为了简单起见, 本文我们选取 $\alpha=0$ 。因此, 方程(9)有唯一解

$$C = A^{-1}B \quad (10)$$

令 $c_j = \sum_{i=1}^n a_{ij}, j=1, 2, \dots, m$, 则

$$C = (\sum_{i=1}^n a_{i1}, \sum_{i=1}^n a_{i2}, \dots, \sum_{i=1}^n a_{im})^T = (c_1, c_2, \dots, c_m)^T,$$

并且

$$\begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_m \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \vdots & \vdots & \dots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \quad (11)$$

$$\text{记 } D = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \vdots & \vdots & \dots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{bmatrix}, E = (1, 1, \dots, 1)^T. \text{ 则 } C =$$

DE , 结合(10)式, 我们有

$$DE = A^{-1}B \quad (12)$$

由(12)可得

$$D = (A^{-1}B)E^+ + Y(I - EE^+) \quad (13)$$

其中 Y 是任意矩阵, E^+ 是 E 的 M-P 广义逆。

假定矩阵 D 已由公式(13)所确定, 那么, 根据 Taylor 公式, 可得

$$T_n(x) = T_n(0) + T_n^{(1)}(0)x + \dots + \frac{1}{r!} T_n^{(r)}(0)x^r + R_r(x) \quad (14)$$

其中

$$R_r(x) = \frac{1}{(r+1)!} T_n^{(r+1)}(\xi)x^{r+1}, \xi \in S \quad (15)$$

又 $T_n^{(r+1)}(x) = \sum_{i=1}^n \sum_{j=1}^m a_{ij} \varphi_{ij}^{(r+1)}(x)$, 所以, $|T_n^{(r+1)}(x)| = |\sum_{i=1}^n \sum_{j=1}^m a_{ij} \varphi_{ij}^{(r+1)}(x)| \leq M$, 其中

$$M = \max_{x \in R} T_n^{(r+1)}(x)$$

所以, 由(15)式得

$$|R_r(x)| \leq \frac{M}{(r+1)!} |x|^{r+1},$$

于是, $\forall \varepsilon > 0, \exists \delta > 0$, 当 $x \in (-\delta, \delta)$ 时,

$$|T_n(x) - P_r(x)| < \varepsilon,$$

这里 $\delta = (r+1)! (\frac{\varepsilon}{M})^{\frac{1}{r+1}}$ 。证毕。

对于多元函数变换, 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个子泛函网络模型的“集成”。这样, 我们可将复杂泛函网络分解成若干个子泛函网络进行研究, 分析其子泛函网络的特性, 研究其对整个泛函网络的影响。根据这个思想, 针对多元多项式函数变换, 可用一元多项式函数变换经过多次函数变换来完成。对任意一元多项式函数可写成

$$F = F(x_1, x_2, \dots, x_n) = f_n(x_2, \dots, x_n)x_1^n + f_{n-1}(x_2, \dots, x_n)x_1^{n-1} + \dots + f_0(x_2, \dots, x_n) \quad (16)$$

其中, 我们把变元 x_1 看作主变元, 进一步作以下变换可得到

$$F = F(x_1, x_2, \dots, x_n) \rightarrow F'(x_1, x_2, \dots, x_n) = f_n^{n-1} F(x/f_n, x_2, \dots, x_n) \quad (17)$$

这样, 可对任意一元多项式函数都可归结为首项系数为 1 的一元多项式函数变换的情形, 同样可设计出式(16)所对应的泛函网络拓扑结构, 参照以上证明过程, 可得到类似的结论。

结束语 本文利用线性代数相关的知识, 用构造性方法证明: 对于给定的 r 阶多项式函数, 可以具体地构造出一个三层泛函网络, 以任意精度逼近该多项式, 所构造的网络的中间神经元个数仅与多项式基函数的阶数 r 有关, 并能用 r 可表达。该文所得结果对于基于多项式基函数的泛函网络逼近任意函数类的网络具体构造和逼近方法具有理论指导意义。

参考文献

- [1] Castillo E. Functional Networks [J]. Neural Processing Letters, 1998, 7, 151-159

(下转第 164 页)

- the Se ntlantic Grid: A Future e— Science Infrastructure <http://www.Semantic.grid.Org/html/semgrid.html>. 2004-12-8
- [3] Buyya R, Chapin S, Dinucci D. Architectural Models for Resource Management in the Grid[A]// The 1st IEEE/ACM International Workshop on Grid Computing[C]. London: Springer Verlag, 2000; 18-35
- [4] The Grid: Blueprint for a New Computing Infrastructure// Ian Foster and Carl Kesselman, eds. Morgan-Kaufman, 1998
- [5] Fox G, Pierce M, Gannon D, et al. An Overview of Grid Computing Environments, Global Grid Forum GFD-I. 9. <http://www.gridforum.org/documents/GFD>, 2003
- [6] Fox G, Berman F, Hey T. Grid Computing: Making the Global Infrastructure a Reality// Fox, G, ed. Wiley, 2003
- [7] Wachob C. What are personalization and customization? <http://www.workz.com/content/>
- [8] Li M, Santen P V, Walker D W, et al. PortalLab: A Web Services Toolkit for Building Semantic Grid Portals// CCGrid Symposium. Tokyo, Japan, 2003
- [9] Network for Earthquake Engineering Simulation Grid (NEES-grid). <http://www.neesgrid.org>
- [10] NCSA Alliance Scientific Portal Project. <http://www.extreme.indiana.edu/alliance>
- [11] Barbera R. The GENIUS Grid Portal. Computing in High Energy and Nuclear Physics, La Jolla, California, March 2003
- [12] Allan R. Integrated e-Science Environment for CLRC. <http://esc.dl.ac.uk/IeSE>
- [13] NLNR Grid Portal Development Kit. <http://dast.nlanr.net/Features/GridPortal/>
-
- (上接第 121 页)
- [2] Castillo E, Cobo A, Gutiérrez J M, et al. Functional Networks with Applications [M]. Kluwer Academic Publishers, 1999
- [3] Castillo E, Gutierrez J M, Cobo A, et al. A minimax method for learning functional networks [J]. Neural Processing Letters, 2000, 11(1): 39-49
- [4] Iglesias A, Arcay B, Cotos J M, et al. A comparison between functional networks and artificial neural networks for the prediction of fishing catches [J]. Neural compute & applies, 2004, 13: 24-31
- [5] Castillo E, Gutierrez J M. Nonlinear time series modeling and prediction using functional networks. Extracting information masked by chaos [J]. Physics Letter A, 1998, 244: 71-84
- [6] Castillo E, Cobo A, Gutiérrez J M. Working with differential, functional and difference equations using functional networks [J]. Appl. Math. Model, 1999, 23: 89-107
- [7] Zhou Yongquan, Jiao Licheng. Interpolation Mechanism of Functional Networks [J]. Lecture Notes in Computer Science 3697, 2005, 45-51
- [8] Zhou Yongquan, Jiao Licheng. Approximate Factorization Learning Algorithm of Multivariate Polynomials Based on Functional Networks [J]. Journal of Information and Computational Science, 2005, 2(1): 205-210
- [9] Zhou Yongquan, He Dengxu, Nong Zheng. Application of Functional Network to Solving Classification Problems [J]. ENFOR-MATIKA, 2005, 7: 390-393
- [10] Zhou Yongquan, He Dexu, Jiao Licheng. A Neural Computation Structure for Solving Functional Equations with Functional Networks [J]. Journal of Information and Computational Science, 2005, 2(4): 835-842
- [11] He Dexu, Nong Zheng, Zhou Yongquan. Approximate Factorization of Univariate Polynomials Using Functional Networks [J]. Journal of Information and Computational Science, 2005, 2(3): 473-479
- [12] Zhou Yongquan, Jiao Licheng. Application of functional networks to solving functional equations [C]// Proceedings 2005 International Conference on Neural Networks & Brain (ICNN&B'05). Beijing, China. IEEE Press, Oct. 2005; 1378-1383
- [13] Zhou Yongquan, Jiao Licheng. One-variable interpolation function based on functional networks. International Journal of Information Technology [J], 2006, 12(2): 120-129
- [14] Zhou Yongquan, Lin Daozhu, Yang Yindong. Functional networks and Tunable Activation Function Neural Networks [C]// Proceedings of the 6th World Congress on Control and Automation. IEEE Press, IEEE Catalog Number: 06EX1358; 2757-2762. Dalian, China, June 2006; 21-23
- [15] Zhou Yongquan, Wang Dongdong, Zhang Min. Designing Functional networks Through Evolutionary Programming [C]// Proceedings of the 6th World Congress on Control and Automation. IEEE Press, IEEE Catalog Number: 06EX1358; 3250-3254. Dalian, China, June 2006; 21-23
- [16] 周永权, 焦李成. 层次泛函网络整体学习算法 [J]. 计算机学报, 2005, 28(8): 1277-1286
- [17] 周永权, 焦李成, 李陶深. 一种基于泛函网络的多项式 Euclidean 算法 [J]. 计算机科学, 2006, 33(9): 131-134
- [18] 周永权, 赵斌, 焦李成. 一种复值泛函网络逼近理论及学习算法 [J]. 系统工程与电子技术, 2006, 28(8): 1244-1248
- [19] 周永权, 焦李成. 基于泛函网络的多维函数逼近理论及学习算法 [J]. 系统工程与电子技术, 2005, 27(5): 906-909
- [20] 周永权, 焦李成. 基于遗传规划实现泛函网络神经元的函数类型优化 [J]. 计算机科学, 2007, 34(2): 7-9
- [21] 周永权, 焦李成, 李陶深. Fuzzy 插值及其 Fuzzy 泛函网络构造理论 [J]. 计算机科学, 2007, 34(7): 5-9
- [22] 徐利治, 王仁宏, 周蕴时. 函数逼近的理论与方法 [M]. 上海: 上海科学技术出版社, 1983
- [23] 葛建琦, 陈火旺, 王兵山. 多维函数的进化逼近 [J]. 计算机学报, 2000, 23(6): 593-601