

# 基于受控 Hadamard 门的量子神经网络模型及算法

李盼池 周红岩

(东北石油大学计算机与信息技术学院 黑龙江大庆 163318)  
(lipanchi@vip.sina.com)

## Model and Algorithm of Quantum Neural Network Based on the Controlled Hadamard Gates

Li Panchi and Zhou Hongyan

(School of Computer and Information Technology, Northeast Petroleum University, Daqing, Heilongjiang 163318)

**Abstract** To enhance the approximation capability of neural network, a quantum neural network model based on the controlled-Hadamard gates is proposed. This model takes a multi-dimensional discrete sequence as the input, which can be described by a matrix where the number of rows denotes the number of input nodes, and the number of columns denotes the length of discrete sequence. This model consists of three layers, the hidden layer consists of quantum neurons, and the output layer consists of common neurons. The quantum neuron consists of the quantum rotation gates and the multi-qubits controlled-Hadamard gates. Using the information feedback of target qubit from output to input in multi-qubits controlled-Hadamard gates, the overall memory of input sequence is realized. The output of quantum neuron is obtained from the controlled relationship between the control bits and target bit of controlled-Hadamard gates. The learning algorithm is designed in detail according to the basis principles of quantum computation. The characteristics of input sequence can be effectively obtained. The experimental results show that, when the input nodes and the length of the sequence satisfy certain relations, the proposed model is obviously superior to the common BP neural network.

**Key words** quantum computation; quantum rotation gate; controlled-Hadamard gate; quantum neuron; quantum neural network

**摘要** 为提高神经网络的逼近能力,提出一种基于受控 Hadamard 门设计的量子神经网络模型及算法。该模型输入为多维离散序列,可用矩阵描述,行数为输入节点数,列数为序列长度。模型为 3 层结构,隐层为量子神经元,输出层为普通神经元。量子神经元由量子旋转门和多位受控 Hadamard 门组成,利用多位受控 Hadamard 门中目标量子位的输出向输入端的反馈,实现对输入序列的整体记忆,利用受控 Hadamard 门中控制位和目标位之间的受控关系获得量子神经元的输出。基于量子计算理论设计了该模型的学习算法。该模型可高效地获取输入序列的特征。实验结果表明,当输入节点数和序列长度满足一定关系时,该模型明显优于普通 BP 神经网络。

**关键词** 量子计算;量子旋转门;受控 Hadamard 门;量子神经元;量子神经网络

中图法分类号 TP183

量子神经网络的概念出现于 20 世纪 90 年代,自诞生之日起,人们在这个崭新的领域进行了不同方向的探索,提出了很多想法和初步的模型,充分体

现了量子神经网络研究的巨大潜力。1995 年, Kak<sup>[1]</sup> 发表了“On Quantum Neural Computing”论文,首次提出量子神经计算的概念。1997 年, Gopathy

等人<sup>[2]</sup>借用量子理论中量子态叠加的思想,提出了基于多级激励函数的量子神经网络模型,在3层网络结构中,隐层量子神经元的激励函数采用多个传统激励函数的叠加,使网络有了一种固有的模糊性. 1998年, Ventura<sup>[3]</sup>提出了量子联想存储的算法,相对于传统存储量子联想存储具有指数级的存储容量. 2000年, Ajit 等人<sup>[4]</sup>从多宇宙的量子理论观点,提出了构建叠加的多宇宙量子神经网络模型的思想. 2001年,中国科学技术大学的解光军等人<sup>[5]</sup>首次在国内撰文阐述了量子神经计算的概念,对国内量子神经网络研究起到了一定的先导作用. 2004年,解光军等人<sup>[6-7]</sup>深入研究了以通用量子逻辑门组作为计算基函数,构造量子神经网络模型的新方法,研究结果表明该模型的性能优于传统神经网络. 2007年, Maeda 等人<sup>[8]</sup>根据量子线路结构提出一种求解异或问题的量子神经元学习算法. 同年,我们提出一种基于量子输入和量子权值的量子自组织特征映射网络模型及聚类算法,该算法可明显提高普通自组织网络的聚类性能<sup>[9]</sup>. 2008年,我们基于量子旋转门和量子受控非门的物理意义,提出一种基于通用量子门组演化的量子BP网络模型<sup>[10]</sup>. 2009年,我们提出一种基于量子权值及量子活性值的量子神经网络模型,基于梯度下降法构造了该模型的超线性收敛学习算法<sup>[11]</sup>. 2011年,我们基于量子线路的物理意义,提出一种量子门节点神经网络模型及算法<sup>[12]</sup>,借助三角函数的周期性,该模型含有大量的多重吸引子,可显著提高收敛概率. 目前量子计算与神经计算的融合正逐步成为一个崭新的研究方向.

在上述所有模型中,尽管量子计算的融入有效提高了神经计算的性能,但是,量子计算与神经计算的融合研究目前还远未成熟,有必要进一步深入研究二者新的融合方式,以期进一步提高神经计算的性能. 为此,本文提出一种基于受控 Hadamard 门的量子神经网络(controlled-Hadamard-based quantum neural network, CHQNN)模型,该模型每个样本的输入为多维离散序列,利用受控 Hadamard 门对目标位的循环控制,可以实现对输入序列的整体记忆,基于量子计算原理设计了学习算法. 该模型的核心优势在于信息的并行处理和网络参数的多维调整,以太阳光子数年度均值预测为例,实验结果表明,在一定条件下,本文提出的模型明显优于普通BP神经网络.

## 1 量子比特及量子门概述

### 1.1 量子比特

在量子计算中,量子比特有两个可能的状态: $|0\rangle$ 和 $|1\rangle$ ,与经典比特的区别在于量子比特的状态可以落在 $|0\rangle$ 和 $|1\rangle$ 之外,即可以是状态的线性叠加态:

$$|\varphi\rangle = \alpha|0\rangle + \beta|1\rangle, \quad (1)$$

其中, $\alpha, \beta$ 是一对复数,称为量子比特的概率幅,量子比特也可用概率幅表示为 $|\varphi\rangle = [\alpha, \beta]^T$ .

### 1.2 量子旋转门

量子门是物理实现量子计算的基础,它包含了量子计算的特点. 量子旋转门定义式为

$$\mathbf{R}(\theta) = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}. \quad (2)$$

$$\text{令 } |\varphi\rangle = \begin{bmatrix} \cos \theta_0 \\ \sin \theta_0 \end{bmatrix}, \text{ 由 } \mathbf{R}(\theta) |\varphi\rangle = \begin{bmatrix} \cos(\theta + \theta_0) \\ \sin(\theta + \theta_0) \end{bmatrix} \text{ 可}$$

知, $\mathbf{R}(\theta)$ 实现了对 $|\varphi\rangle$ 的相位旋转.

### 1.3 Hadamard 门

该门的定义式为 $\mathbf{H} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$ ,该门把计算基矢 $|0\rangle$ 和 $|1\rangle$ 变成新的基矢 $|+\rangle$ 和 $|-\rangle$ ,后者是计算基矢的叠加态: $\mathbf{H}|0\rangle = \sqrt{0.5}(|0\rangle + |1\rangle) \equiv |+\rangle$ ,  $\mathbf{H}|1\rangle = \sqrt{0.5}(|0\rangle - |1\rangle) \equiv |-\rangle$ . 由 $(\mathbf{H}^T)^* = \mathbf{H}$ 知, $\mathbf{H}$ 是酉算子(厄米算子)<sup>[13]</sup>.

### 1.4 受控 Hadamard 门

在真实的量子系统中,单个量子比特的状态往往受多位量子比特联合控制,多位受控 Hadamard 门 $\mathbf{C}^n(\mathbf{H})$ 是这种控制方式的一种模型. 在 $n+1$ 比特量子系统 $|x_1\rangle |x_2\rangle \cdots |x_n\rangle |\varphi\rangle$ 中,当目标比特 $|\varphi\rangle$ 同时受 $n$ 位输入比特 $|x_1\rangle |x_2\rangle \cdots |x_n\rangle$ 的联合控制时,其系统的动态特性可采用如下多比特量子受控 Hadamard 门描述,如图1所示.

在图1(a)中,若前 $n$ 个量子比特全为1,则 Hadamard 门作用到最后一位量子比特;否则不起任何作用. 运算关系可描述如下.

$$\mathbf{C}^n(\mathbf{H}) |x_1 x_2 \cdots x_n\rangle |\varphi\rangle = |x_1 x_2 \cdots x_n\rangle \mathbf{H}^{x_1 x_2 \cdots x_n} |\varphi\rangle. \quad (3)$$

令 $n$ 个控制比特为 $|x_i\rangle = a_i|0\rangle + b_i|1\rangle$ ,目标比特为 $|\varphi\rangle = c|0\rangle + d|1\rangle$ ,其中 $i=1, 2, \dots, n$ ,则式(3)可重写为

$$\begin{aligned}
& \mathbf{C}^n(\mathbf{H}) |x_1 x_2 \cdots x_n\rangle |\varphi\rangle = \\
& |x_1\rangle \otimes |x_2\rangle \otimes \cdots \otimes |x_n\rangle \otimes |\varphi\rangle - \\
& b_1 b_2 \cdots b_n c |11 \cdots 10\rangle - b_1 b_2 \cdots b_n d |11 \cdots 11\rangle + \\
& \sqrt{0.5} b_1 b_2 \cdots b_n (c+d) |11 \cdots 10\rangle + \\
& \sqrt{0.5} b_1 b_2 \cdots b_n (c-d) |11 \cdots 11\rangle. \quad (4)
\end{aligned}$$

由式(4)可知,  $\mathbf{C}^n(\mathbf{H})$  中目标位的输出端  $|\varphi'\rangle$  处于状态  $|1\rangle$  的概率为

$$P = 0.5(b_1 b_2 \cdots b_n)^2 (c^2 - d^2 - 2cd) + d^2. \quad (5)$$

在图 1(b) 中, 若前  $n$  个量子比特全为 0, 则 Hadamard 门作用到最后一位量子比特; 否则不起任何作用.  $\mathbf{C}^n(\mathbf{H})$  的运算关系可描述为

$$\begin{aligned}
& \mathbf{C}^n(\mathbf{H}) |x_1 x_2 \cdots x_n\rangle |\varphi\rangle = \\
& |x_1 x_2 \cdots x_n\rangle \mathbf{H}^{x_1 + x_2 + \cdots + x_n} |\varphi\rangle. \quad (6)
\end{aligned}$$

采用与图 1(a) 类似的分析方法可知,  $|\varphi'\rangle$  处于状态  $|1\rangle$  的概率为

$$P = 0.5(a_1 a_2 \cdots a_n)^2 (c^2 - d^2 - 2cd) + d^2. \quad (7)$$

此时, 经  $n$  个输入比特的联合控制后, 目标比特的输出可写为

$$|\varphi'\rangle = \sqrt{1-P} |0\rangle + \sqrt{P} |1\rangle. \quad (8)$$

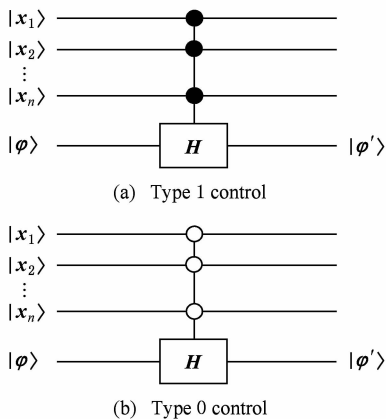


Fig. 1 The controlled-Hadamard gate of multi-qubits.

图 1 多比特受控 Hadamard 门

## 2 量子神经网络模型

### 2.1 量子神经元模型

本文提出的量子神经元模型由量子旋转门和多位受控 Hadamard 门组成, 如图 2 所示. 其输入为定义在时域区间  $[0, T]$  上的量子比特序列  $\{|x_i(t_r)\rangle\}$ , 其中  $t_r \in [0, T]$ ; 输出为在  $[0, T]$  上的聚合结果  $y$ ; 控制参数为量子旋转门的旋转角度  $\bar{\theta}_i(t_r)$ ,  $i = 1, 2, \dots, n, r = 1, 2, \dots, q, n$  为输入空间的维数,  $q$  为序列长度.

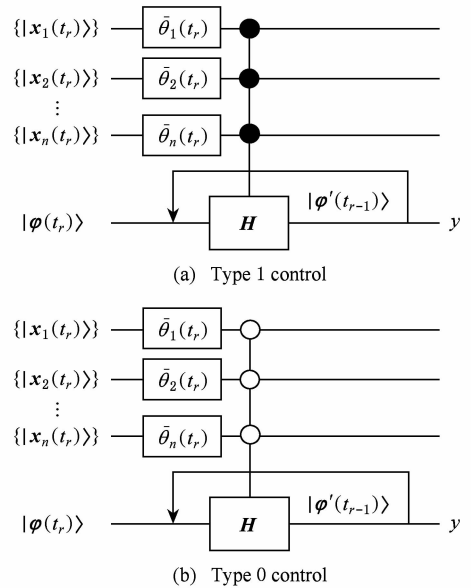


Fig. 2 The model of quantum neuron.

图 2 量子神经元模型

记  $t_1 < t_2 < \cdots < t_q$  为采样时刻,  $|x_i(t_r)\rangle = \cos \theta_i(t_r) |0\rangle + \sin \theta_i(t_r) |1\rangle$ . 设  $|\varphi(t_1)\rangle = |0\rangle$ , 由量子旋转门和多位受控 Hadamard 门的定义,  $|\varphi'(t_1)\rangle$  可写为

$$|\varphi'(t_1)\rangle = \cos \varphi(t_1) |0\rangle + \sin \varphi(t_1) |1\rangle. \quad (9)$$

对于图 2(a):

$$\varphi(t_1) = \arcsin\left(\sqrt{0.5} \prod_{i=1}^n \sin(\theta_i(t_1) + \bar{\theta}_i(t_1))\right),$$

对于图 2(b):

$$\varphi(t_1) = \arcsin\left(\sqrt{0.5} \prod_{i=1}^n \cos(\theta_i(t_1) + \bar{\theta}_i(t_1))\right).$$

当  $1 < r \leq q$  时, 由  $|\varphi(t_r)\rangle = |\varphi'(t_{r-1})\rangle$ ,  $|\varphi'(t_r)\rangle$  可写为

$$|\varphi'(t_r)\rangle = \cos \varphi(t_r) |0\rangle + \sin \varphi(t_r) |1\rangle, \quad (10)$$

其中:

$$\begin{aligned}
\varphi(t_r) = & \arcsin((S_r (\cos 2\varphi(t_{r-1}) - \\
& \sin 2\varphi(t_{r-1})) + \sin^2 \varphi(t_{r-1}))^{1/2}). \quad (11)
\end{aligned}$$

对于图 2(a):

$$S_r = 0.5 \prod_{i=1}^n \sin^2(\theta_i(t_r) + \bar{\theta}_i(t_r)),$$

对于图 2(b):

$$S_r = 0.5 \prod_{i=1}^n \cos^2(\theta_i(t_r) + \bar{\theta}_i(t_r)).$$

当  $r = q$  时, 式(10)即为量子神经元对  $[0, T]$  上序列的输出结果. 在本文中, 我们定义量子神经元的输出为目标比特处于状态  $|1\rangle$  的概率幅, 因此, 其输入输出关系可描述为

$$y = \sqrt{S_q (\cos 2\varphi(t_{q-1}) - \sin 2\varphi(t_{q-1})) + \sin^2 \varphi(t_{q-1})}. \quad (12)$$

### 2.2 量子神经网络模型

本文提出的基于受控 Hadamard 门的量子神经网络模型由 3 层组成, 其中隐层为量子神经元, 第 1, 3, 5, ... 个量子神经元采用 1 型控制, 第 2, 4, 6, ...

个量子神经元采用 0 型控制, 输出层为普通神经元, 如图 3 所示. 图 3 中,  $\{|\mathbf{x}_1(t_r)\rangle\}, \{|\mathbf{x}_2(t_r)\rangle\}, \dots, \{|\mathbf{x}_n(t_r)\rangle\}$  为输入序列;  $|\boldsymbol{\varphi}_1(t_r)\rangle, |\boldsymbol{\varphi}_2(t_r)\rangle, \dots, |\boldsymbol{\varphi}_p(t_r)\rangle$  为隐层量子神经元的目标位;  $h_1, h_2, \dots, h_p$  为隐层输出;  $w_{jk}$  为输出层连接权;  $y_1, y_2, \dots, y_m$  为网络输出; 输出层采用 Sigmoid 函数作为激励函数.

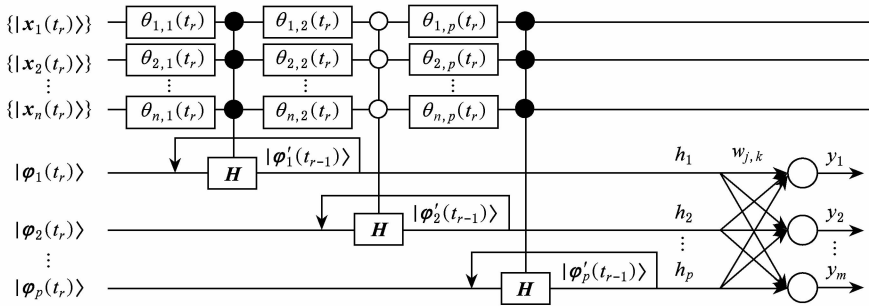


Fig. 3 The model of quantum neural networks.

图 3 量子神经网络模型

普通神经网络的单个样本用向量表示, 而 CHQNN 中, 单个样本用矩阵表示, 例如样本  $[\{|\mathbf{x}_1(t_r)\rangle\}, \{|\mathbf{x}_2(t_r)\rangle\}, \dots, \{|\mathbf{x}_n(t_r)\rangle\}]^T$  可表示为

$$\begin{bmatrix} \{|\mathbf{x}_1(t_r)\rangle\} \\ \{|\mathbf{x}_2(t_r)\rangle\} \\ \vdots \\ \{|\mathbf{x}_n(t_r)\rangle\} \end{bmatrix} = \begin{bmatrix} |\mathbf{x}_1(t_1)\rangle & |\mathbf{x}_1(t_2)\rangle & \dots & |\mathbf{x}_1(t_q)\rangle \\ |\mathbf{x}_2(t_1)\rangle & |\mathbf{x}_2(t_2)\rangle & \dots & |\mathbf{x}_2(t_q)\rangle \\ \vdots & \vdots & \ddots & \vdots \\ |\mathbf{x}_n(t_1)\rangle & |\mathbf{x}_n(t_2)\rangle & \dots & |\mathbf{x}_n(t_q)\rangle \end{bmatrix}. \quad (13)$$

设  $|\mathbf{x}_i(t_r)\rangle = \cos \theta_i(t_r) |0\rangle + \sin \theta_i(t_r) |1\rangle$ , 其中  $t_1 < t_2 < \dots < t_q$  为离散采样点. 令  $|\boldsymbol{\varphi}_j(t_1)\rangle = |0\rangle, j = 1, 2, \dots, p$ . 记:

$$S_{j,r} = \begin{cases} 0.5 \prod_{i=1}^n \sin^2(\theta_i(t_r) + \theta_{ij}(t_r)), & j = 1, 3, 5, \dots, \\ 0.5 \prod_{i=1}^n \cos^2(\theta_i(t_r) + \theta_{ij}(t_r)), & j = 2, 4, 6, \dots. \end{cases} \quad (14)$$

根据量子神经元的输入输出关系, 在时域区间  $[0, t_r]$  上隐层第  $j$  个量子神经元的阶段输出为

$$\begin{cases} h_j(t_1) = \sqrt{S_{j,1}}, \\ h_j(t_r) = \left[ S_{j,r} (1 - 2h_j^2(t_{r-1}) - 2h_j(t_{r-1})) \right]^{1/2} \\ \quad \sqrt{1 - h_j^2(t_{r-1})} + h_j^2(t_{r-1}) \end{cases}. \quad (15)$$

该量子神经元的最终(即时刻  $t = t_q$ )输出为

$$h_j = h_j(t_q). \quad (16)$$

输出层第  $k(k = 1, 2, \dots, m)$  个普通神经元的输出为

$$y_k = \frac{1}{1 + e^{-\sum_{j=1}^p w_{jk} h_j}}. \quad (17)$$

### 3 量子神经网络算法

#### 3.1 样本量子态描述

对于  $n$  维实序列样本  $\{\bar{\mathbf{X}}(t_r)\} = [\{\bar{x}_1(t_r)\}, \{\bar{x}_2(t_r)\}, \dots, \{\bar{x}_n(t_r)\}]^T, r = 1, 2, \dots, q$ , 记:

$$\begin{cases} Max_i = \max(\bar{x}_i(t_1), \bar{x}_i(t_2), \dots, \bar{x}_i(t_q)), \\ Min_i = \min(\bar{x}_i(t_1), \bar{x}_i(t_2), \dots, \bar{x}_i(t_q)), \end{cases} \quad (18)$$

$$\theta_{ir} = \frac{\bar{x}_i(t_r) - Min_i \pi}{Max_i - Min_i} \frac{\pi}{2}. \quad (19)$$

这些样本可用量子态形式描述如下:

$$\begin{aligned} \{|\mathbf{X}(t_r)\rangle\} &= [\{|\mathbf{x}_1(t_r)\rangle\}, \\ &\{|\mathbf{x}_2(t_r)\rangle\}, \dots, \{|\mathbf{x}_n(t_r)\rangle\}]^T, \end{aligned} \quad (20)$$

其中,  $|\mathbf{x}_i(t_r)\rangle = \cos(\theta_{ir}) |0\rangle + \sin(\theta_{ir}) |1\rangle$ .

#### 3.2 网络参数调整方法

网络参数包括隐层量子旋转门旋转角度、输出层连接权. 设归一化后的期望输出为  $\bar{y}_1, \bar{y}_2, \dots, \bar{y}_m$ , 对应的实际输出为  $y_1, y_2, \dots, y_m$ , 误差函数定义为

$$E = \frac{1}{2} \sum_{k=0}^m (e_k)^2 = \frac{1}{2} \sum_{k=0}^m (\bar{y}_k - y_k)^2. \quad (21)$$

根据梯度下降法, 隐层旋转角的梯度计算式为

$$\frac{\partial e_k}{\partial h_j(t_q)} = - \sum_{k=1}^m y_k (1 - y_k) w_{jk}, \quad (22)$$

$$\frac{\partial h_j(t_r)}{\partial h_j(t_{r-1})} = ((1 - 2S_{j,r})h_j(t_{r-1})\sqrt{1 - h_j^2(t_{r-1})} + S_{j,r}(2h_j^2(t_{r-1}) - 1))/(h_j(t_r)\sqrt{1 - h_j^2(t_{r-1})}). \quad (23)$$

对于  $j=1, 3, 5, \dots$ :

$$\frac{\partial h_j(t_r)}{\partial \theta_{ij}(t_r)} = (S_{j,r}(1 - 2h_j^2(t_{r-1}) - 2h_j(t_{r-1})\sqrt{1 - h_j^2(t_{r-1})}) \times \cot(\theta_i(t_r) + \theta_{ij}(t_r)))/h_j(t_r). \quad (24)$$

对于  $j=2, 4, 6, \dots$ :

$$\frac{\partial h_j(t_r)}{\partial \theta_{ij}(t_r)} = (S_{j,r}(1 - 2h_j^2(t_{r-1}) - 2h_j(t_{r-1})\sqrt{1 - h_j^2(t_{r-1})}) \times \tan(\theta_i(t_r) + \theta_{ij}(t_r)))/h_j(t_r). \quad (25)$$

由式(22)~(25)可得:

$$\frac{\partial e_k}{\partial \theta_{ij}(t_r)} = \frac{\partial e_k}{\partial h_j(t_q)} \prod_{s=r+1}^q \frac{\partial h_j(t_s)}{\partial h_j(t_{s-1})} \frac{\partial h_j(t_r)}{\partial \theta_{ij}(t_r)}. \quad (26)$$

输出层连接权的梯度计算式为

$$\frac{\partial e_k}{\partial \omega_{jk}} = -y_k(1 - y_k)h_j(t_q). \quad (27)$$

因梯度计算式和误差曲面较为复杂,故梯度下降法(最速下降法)收敛相对较慢或不易收敛;虽然牛顿法的收敛速度通常比最速下降法快,但需要计算目标函数的二阶导数,且其表现很复杂,除了存在容易陷入到鞍点的问题外,算法还可能出现震荡和发散. L-M(levenberg-marquardt)算法<sup>[14]</sup>是牛顿法的变形,用以最小化作为其他非线性函数平方和的那些函数,这非常适合于性能指数是均方误差的神经网络训练. 因此本文采用 L-M 算法调整网络参数.

记  $\mathbf{X}$  表示参数向量,  $\mathbf{V}$  表示误差向量,  $\mathbf{J}$  表示雅可比矩阵,它们分别定义如下:

$$\mathbf{X}^T = [\theta_{11}(t_1), \dots, \theta_{np}(t_q), \omega_{11}, \dots, \omega_{pm}], \quad (28)$$

$$\mathbf{V}^T(\mathbf{X}) = [e_1, e_2, \dots, e_m], \quad (29)$$

$$\mathbf{J}(\mathbf{X}) = \begin{bmatrix} \frac{\partial e_1}{\partial \theta_{11}(t_1)} & \dots & \frac{\partial e_1}{\partial \theta_{np}(t_q)} & \frac{\partial e_1}{\partial \omega_{11}} & \dots & \frac{\partial e_1}{\partial \omega_{pm}} \\ \frac{\partial e_2}{\partial \theta_{11}(t_1)} & \dots & \frac{\partial e_2}{\partial \theta_{np}(t_q)} & \frac{\partial e_2}{\partial \omega_{11}} & \dots & \frac{\partial e_2}{\partial \omega_{pm}} \\ \vdots & & \vdots & \vdots & & \vdots \\ \frac{\partial e_m}{\partial \theta_{11}(t_1)} & \dots & \frac{\partial e_m}{\partial \theta_{np}(t_q)} & \frac{\partial e_m}{\partial \omega_{11}} & \dots & \frac{\partial e_m}{\partial \omega_{pm}} \end{bmatrix}. \quad (30)$$

根据 L-M 算法, CHQNN 参数调整的迭代方程为

$$\mathbf{X}_{t+1} = \mathbf{X}_t - [\mathbf{J}^T(\mathbf{X}_t)\mathbf{J}(\mathbf{X}_t) + \mu_t \mathbf{I}]^{-1} \mathbf{J}^T(\mathbf{X}_t)\mathbf{V}(\mathbf{X}_t), \quad (31)$$

其中  $t$  为迭代步数,  $\mathbf{I}$  为单位矩阵,  $\mu_t$  为一小正数, 以使  $\mathbf{J}^T(\mathbf{X}_t)\mathbf{J}(\mathbf{X}_t) + \mu_t \mathbf{I}$  可逆.

### 3.3 网络的收敛性

1989 年, 关于两层前馈误差反向传播神经网络的收敛性, Hecht-Nielsen 证明了如下引理<sup>[15]</sup>.

**引理 1.** 具有两层连接权值的误差反向传播神经网络, 在隐层节点足够多的情况下能够逼近  $[0, 1]^n$  上的任意 L2 函数.

关于本文提出的基于受控 Hadamard 门的量子神经网络的收敛性, 我们给出如下结论.

**定理 1.** 当隐层神经元足够多时, CHQNN 能够逼近  $[0, 1]^n$  上的任意 L2 函数.

证明. CHQNN 隐层量子神经元输出可重写为

$$h_j = h_j(t_q) = \sqrt{S_{j,1} + \sum_{r=1}^{q-1} S_{j,r+1} \bar{S}_{jr}}, \quad (32)$$

其中,  $\bar{S}_{jr} = 1 - 2h_j^2(t_r) - 2h_j(t_r)\sqrt{1 - h_j^2(t_r)}$ .

由式(14)知,  $S_{j,r}$  为旋转角度的正余弦函数; 由式(15)知,  $h_j(t_r)$  为  $S_{j,r}$  的复合函数; 由式(32)知,  $h_j = h_j(t_q)$  为  $S_{j,r}$  和  $h_j(t_r)$  的复合函数; 因此  $h_j = h_j(t_q)$  也为旋转角度的正余弦函数. 由于正余弦函数及其复合函数均连续可微, 所以  $h_j = f(\cdot)$  连续可微; 由于  $h_j$  为目标比特处于状态  $|1\rangle$  的概率幅, 所以  $0 \leq h_j \leq 1$ , 即  $h_j = f(\cdot)$  有界. 因此量子神经元可视作为传统神经元在激励函数、聚合方式两方面的推广, 即将传统的 Sigmoid 函数改为  $f(\cdot)$ , 而将对输入的加权聚合改为相位移动. CHQNN 的输出层就是普通神经元. 因此 CHQNN 可视为一种特殊形式的传统两层神经网络. 当  $\mu_t$  较大时, L-M 算法接近于如下有小的学习速度的最速下降算法:

$$\mathbf{X}_{t+1} \cong \mathbf{X}_t - \frac{1}{\mu_t} \mathbf{J}^T(\mathbf{X}_t)\mathbf{V}(\mathbf{X}_t) = \mathbf{X}_t - \frac{1}{\mu_t} \nabla E(\mathbf{X}_t). \quad (33)$$

而当  $\mu_t$  较小时, L-M 算法接近于高斯牛顿算法, 该算法较标准牛顿算法的优点是不需要计算二阶导数. 根据最优化理论, 最速下降法和牛顿法均可用于神经网络训练, 实质上是误差反向传播算法中参数调整量两种具体的计算形式, 区别在于牛顿法比最速下降法收敛速度快, 但梯度计算较为复杂. 因此, L-M 算法兼有最速下降法和牛顿法的优点, 从而采用 L-M 算法的 CHQNN 本质上也是一种误差反向传播神经网络. 证毕.

### 3.4 序列样本的构造方法

对于适合传统神经网络的向量式样本, 必须首先

转化为矩阵式的序列样本才能提交 CHQNN 训练. 我们给出的转化方法如下. 对于传统神经网络的一个  $n$  维向量样本  $\mathbf{X}:1$  如果  $n$  是素数, 令  $m=n+1$ , 并通过使  $\mathbf{X}(m)=\mathbf{X}(n)$  将该样本扩展到  $m$  维, 否则, 令  $m=n$ ; 2) 将  $m$  分解为两因子  $m_1$  和  $m_2$  的乘积, 并使  $m_1$  和  $m_2$  尽可能接近. 此时, 一个  $n$  维向量样本即可被转化为  $m_1$  维序列样本, 每维序列长度为  $m_2$ , 或  $m_2$  维序列样本, 每维序列长度为  $m_1$ , 进而可以提交 CHQNN 进行训练.

## 4 仿真对比

本节通过两个实验, 并通过与采用 L-M 算法的 BP 神经网络(BP neural network, BPNN)对比, 验证 CHQNN 的有效性.

### 4.1 时间序列逼近

以文献中广泛研究的无序 Mackey-Glass 时间序列为例, 说明 CHQNN 在时间序列逼近中的具体应用, 并验证其有效性. Mackey-Glass 时间序列可由式(34)产生:

$$x(t+1) = \frac{ax(t-\tau)}{1+x^{10}(t-\tau)} + (1-b)x(t), \quad (34)$$

其中,  $t$  和  $\tau$  是整数,  $a=0.2, b=0.1, \tau=17, x(0) \in$

$(0,1)$ .

从迭代方程(34)我们可以得到时间序列  $\{x(t)\}_{t=1}^{1000}$ , 部分数据如表 1 所示:

Table 1 Some Samples of Mackey-Glass Sequence

表 1 Mackey-Glass 序列的部分样本示例

| $t$ | $x(t)$ | $t$ | $x(t)$ | $t$ | $x(t)$ |
|-----|--------|-----|--------|-----|--------|
| 1   | 0.8678 | 5   | 0.6504 | 9   | 0.4817 |
| 2   | 0.8086 | 6   | 0.6036 | 10  | 0.4480 |
| 3   | 0.7528 | 7   | 0.5599 | 11  | 0.4188 |
| 4   | 0.7001 | 8   | 0.5192 | 12  | 0.3957 |

为充分获取序列特征, 实验方案为采用前面 54 个值预测其后相邻的一个值. 因此, 每个样本的输入为长度 54 的离散序列, 输出为一个单值. 设归一化后的最大绝对误差为 0.15, 迭代步数为 100, 两种模型的控制参数  $\mu_t=0.05$ . CHQNN 隐层旋转角度均初始化为  $(-\pi/2, \pi/2)$  中随机数, 输出层连接权初始化为  $(-1, 1)$  中随机数. 对于 BPNN, 所有权值均初始化为  $(-1, 1)$  中随机数, 隐层和输出层激励均采用 Sigmoid 函数. 根据实验方案, CHQNN 和 BPNN 的输出层均只有一个节点. BPNN 有 54 个输入节点, 关于 CHQNN 的输入节点, 我们采取表 2 中给出的 8 种情形:

Table 2 The Input Nodes and The Sequence Length Setting of Two Models

表 2 两种模型输入节点和序列长度设置

| CHQNN     |             |                 | BPNN   |             |                 |
|-----------|-------------|-----------------|--------|-------------|-----------------|
| Model     | Input Nodes | Sequence Length | Model  | Input Nodes | Sequence Length |
| CHQNN1_54 | 1           | 54              | BPNN54 | 54          | 1               |
| CHQNN2_27 | 2           | 27              | BPNN54 | 54          | 1               |
| CHQNN3_18 | 3           | 18              | BPNN54 | 54          | 1               |
| CHQNN6_9  | 6           | 9               | BPNN54 | 54          | 1               |
| CHQNN9_6  | 9           | 6               | BPNN54 | 54          | 1               |
| CHQNN18_3 | 18          | 3               | BPNN54 | 54          | 1               |
| CHQNN27_2 | 27          | 2               | BPNN54 | 54          | 1               |
| CHQNN54_1 | 54          | 1               | BPNN54 | 54          | 1               |

为充分对比两种模型的逼近能力, 两种模型的隐层分别取 5, 6, ..., 20 个节点. 对于输入节点和隐层节点的每种组合, 两种模型分别训练 10 次. 以 10 次最大逼近误差的平均值为评价指标, 两种模型的逼近能力对比如图 4 所示.

实验结果表明, 当输入节点取 1, 2, 27, 54 时, CHQNN 的逼近能力劣于 BPNN, 而当输入节点取 3, 6, 9, 18 时, CHQNN 的逼近能力优于 BPNN.

从以上实验结果我们可作如下猜想: 当输入节点和序列长度比较接近时, CHQNN 呈现优良的性能. 为验证这一猜测, 我们将两种模型的输入节点和序列长度按表 3 重新设置. 此时, 逼近能力对比如图 5 所示.

实验结果验证了猜想的正确性. 即输入节点和序列长度比较接近时, CHQNN 的逼近能力明显优于 BPNN.

Table 3 The Input Nodes and The Sequence Length Setting of Two Models

表 3 两种模型输入节点和序列长度设置

| CHQNN     |             |                 | BPNN   |             |                 |
|-----------|-------------|-----------------|--------|-------------|-----------------|
| Model     | Input Nodes | Sequence Length | Model  | Input Nodes | Sequence Length |
| CHQNN5_10 | 5           | 10              | BPNN50 | 50          | 1               |
| CHQNN6_9  | 6           | 9               | BPNN54 | 54          | 1               |
| CHQNN6_8  | 6           | 8               | BPNN48 | 48          | 1               |
| CHQNN7_7  | 7           | 7               | BPNN49 | 49          | 1               |
| CHQNN8_6  | 8           | 6               | BPNN48 | 48          | 1               |
| CHQNN9_6  | 9           | 6               | BPNN54 | 54          | 1               |
| CHQNN10_5 | 10          | 5               | BPNN50 | 50          | 1               |

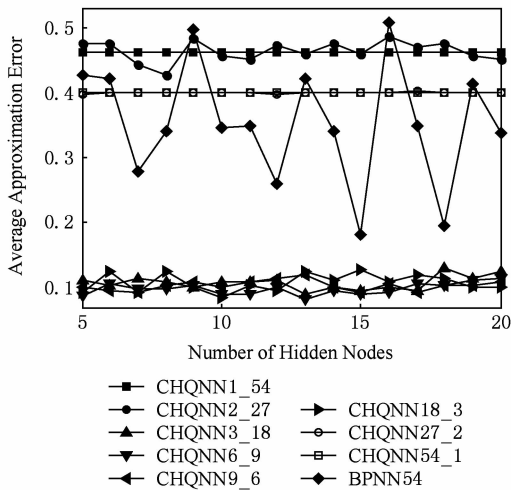


Fig. 4 The average approximation error contrasts.  
图 4 平均逼近误差对比

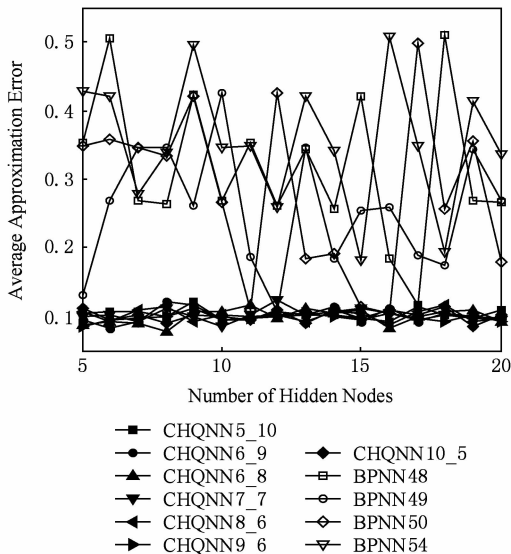


Fig. 5 The average approximation error contrasts.  
图 5 平均逼近误差对比

4.2 太阳黑子数时间序列预测

太阳黑子是在太阳的光球层上发生的一种太阳活动,是太阳活动中最基本、最明显的活动现象。太

阳黑子产生的带电离子可以破坏地球高空的电离层,使大气发生异常,还会干扰地球磁场,从而使电讯中断,因此研究太阳黑子的变换规律有着重要的现实意义。人类在观测太阳的漫长过程中已经积累了很多关于太阳黑子的观测数据,显然,它们是一个典型时变过程的记录。本节利用这些观测数据,采用 CHQNN 建立太阳黑子的预测模型。

4.2.1 构造样本数据

本实验采用太阳黑子的年度平均值序列作为仿真对象,从 1749—2007 年,共计 259 个数据。该数据呈现高度非线性,致使常规预测模型很难凑效。其分布特征如图 6 所示:

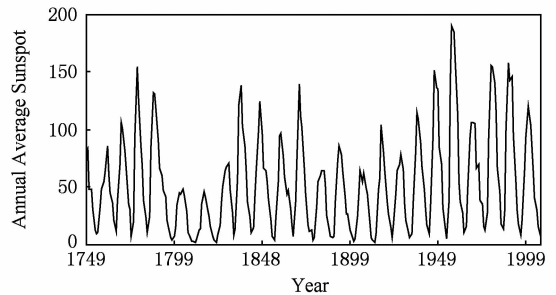


Fig. 6 The measured data of annual average of sunspot (1749—2007).

图 6 1749—2007 年太阳黑子数年平均值数据

样本数据的构造方法是用连续  $n$  年的数据预测其后 1 年的太阳黑子数。例如用 1749—1792 年数据预测 1793 年的太阳黑子数,此时  $n=54$ ,以此类推。用 1749—1948 共 200 年的数据构造训练样本集,完成网络训练。用余下的 59 个样本作为测试集,以检验网络的预测能力。

4.2.2 模型参数设置

为体现 CHQNN 预测模型的优越性,本仿真仍然与采用 L-M 算法的 BPNN 模型对比。由样本特征可知,两种预测模型的输出层只有 1 个节点。为增强对比的客观性,两种模型的隐层均分别取 5,6,...

20 个节点. 由 4.1 节实验可知, 当输入节点和序列长度比较接近时, CHQNN 呈现出优良的性能, 因此, 输入节点仍然按表 3 设置. 对于训练过程, 归一化后的最大绝对误差取为 0.05, 限定步数取为 100. 网络参数的初始化方法同前.

### 4.2.3 仿真结果对比

对于隐层节点的每种取值, 分别用两种模型训练 10 次, 并统计 10 次训练的平均逼近误差、平均迭代步数、收敛次数作为评价指标. 两种模型的训练结果对比如图 7~9 所示:

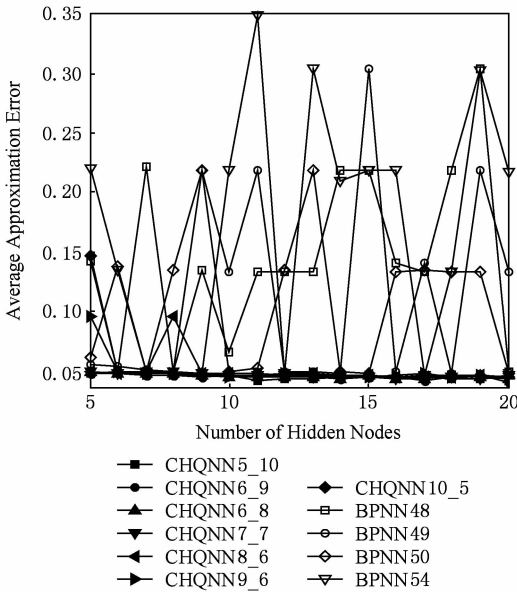


Fig. 7 The average approximation error contrasts.

图 7 平均逼近误差对比

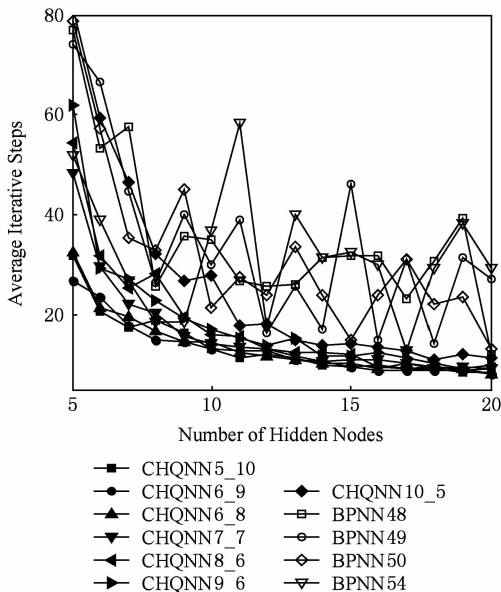


Fig. 8 The average iterative steps contrasts.

图 8 平均迭代步数对比

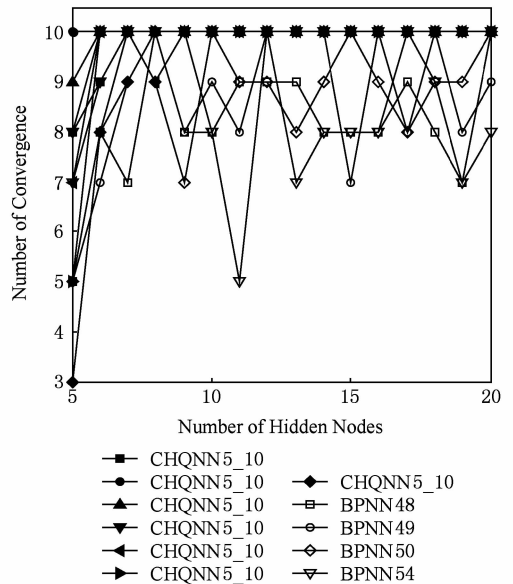


Fig. 9 The number of convergence contrasts.

图 9 收敛次数对比

从图 7~9 可知, 在输入节点和隐层节点的各种取值下, CHQNN 的逼近能力均明显好于 BPNN. 从而进一步证实了 CHQNN 的适用条件.

下面我们考察 CHQNN 的预测能力. 以 6 个输入节点、9 序列长度、10 个隐层节点为例, 此时, BPNN 有 54 个输入节点. 两种模型对预测集的预测结果对比如图 10 所示. 其中 CHQNN 预测结果的最大绝对误差为 16.8175, 误差均值为 0.4650, 方差为 33.3187. 而对于 BPNN, 以上 3 个指标值分别为 28.2763, 1.9836, 147.1196.

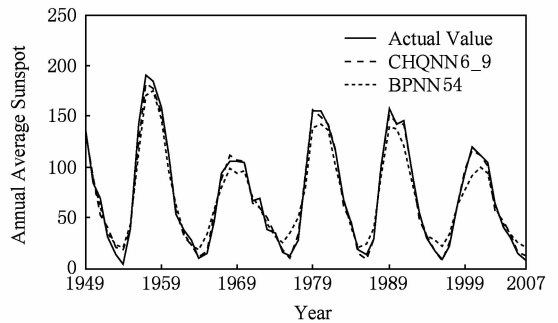


Fig. 10 The prediction result contrasts.

图 10 预测结果对比

综合以上 2 个实验结果, 当样本输入满足一定条件时 CHQNN 的逼近及预测能力明显优于 BPNN. 这个条件就是输入节点和序列长度比较接近. 对于这种结果可作如下分析.

1) 与 BPNN 相比, CHQNN 的核心优势在于信息的并行处理和网络参数的多维调整. 信息的并行



处理可以使 CHQNN 同时接收多维空间中序列式的矩阵信息, 而 BPNN 只能同时接收多维空间中几何点式的向量信息, 因此 CHQNN 具有更强的信息获取能力. 网络参数的多维调整可以使 CHQNN 更为准确地把握网络训练的收敛方向、更为精细地计算网络参数的搜索步长, 从而可以加速收敛的进程.

2) CHQNN 和 BPNN 对输入信息采取了完全不同的处理方式, CHQNN 直接接收离散序列, 利用量子信息处理机制, 将输入序列中的数据循环地映射为隐层量子受控 Hadamard 门的输出, 由于多位受控 Hadamard 门中目标位的受控关系远比 BPNN 的 Sigmoid 函数复杂, 所以这种映射具有高度的非线性, 并使 CHQNN 有更强的逼近能力.

3) 设  $n$  表示输入节点数,  $q$  表示序列长度, CHQNN 的每个样本可以描述为一个  $n$  行  $q$  列的矩阵, 从 CHQNN 算法可以看出, 对于  $n$  和  $q$  的不同组合, 隐层量子神经元的计算结果也是不同的. 事实上, 输入节点个数  $n$  可以视为模式记忆的维数, 主要体现在提取信息的多路性; 而离散序列长度  $q$  可以视为对每一维模式记忆的强化次数, 主要体现在通过反馈不断对各路信息实施强化记忆. 当维数和强化次数适当匹配时, CHQNN 呈现出明显优于 BPNN 的性能. 当  $q=1$  时输入节点是最多的, 隐层量子神经元输出的计算式也是最简单的, 但此时网络只能以多维方式提取模式信息, 而不能对每一维信息实施强化记忆, 因此, 直接导致了逼近能力的降低; 当  $n=1$  时, 隐层量子神经元输出的计算是最复杂的, 虽然此时可以提高网络的映射能力, 然而此时网络参数调整的计算式也是最复杂的, 过长的链式求导计算会使梯度趋于零或无穷, 从而阻碍收敛进程, 并导致逼近能力的降低. 因此, 当  $q=1$  或  $n=1$  时, CHQNN 的逼近能力弱于 BPNN, 而当  $q>1$  和  $n>1$  时, CHQNN 的逼近能力趋于改进, 并且在一定条件下必然会超越 BPNN. 关于  $q$  和  $n$  究竟满足怎样的精确关系 CHQNN 的逼近能力才是最强的, 这个问题尚需进一步研究, 并且通常依赖于具体问题. 我们通过实验得到的结论是: 当  $q/2 \leq n \leq 2q$  时, 具有  $n$  个输入节点和  $q$  个序列长度的 CHQNN 明显优于具有  $nq$  个输入节点的 BPNN.

4) 对于 BPNN, 由于不能直接处理序列式输入, 因此每个样本只能表述为一个  $n$  维向量, 即只能以多维方式获取样本信息而不能对每一维实施强化记忆, 因此在 BPNN 的信息处理过程中, 其信息提

取能力不可避免地受到影响, 从而降低了网络的逼近及预测能力.

## 5 结束语

本文基于量子计算原理提出了一种新型量子神经网络模型, 不同于普通量子神经网络, 该模型的多维输入为量子比特序列. 实验结果表明, 利用量子受控 Hadamard 门的信息处理机制, CHQNN 可以从多度和对每一维实施强化记忆两方面高效地获取样本信息, 当输入节点和序列长度比较接近时可明显提高其逼近和泛化能力, 从而揭示出量子计算与神经计算的融合切实能够改进神经网络的性能. 关于 CHQNN 的连续性、计算复杂度等是我们下一步将要深入研究的问题.

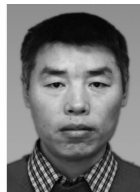
## 参 考 文 献

- [1] Kak S. On quantum neural computing [J]. Information Sciences, 1995, 83(3): 143-160
- [2] Gopathy P, Nicolaos B. Quantum neural networks (QNNs) inherently fuzzy feedforward neural networks [J]. IEEE Trans on Neural Networks, 1997, 8(3): 679-693
- [3] Ventura D, Tony M. Quantum associative memory with exponential capacity [C] // Proc of the 1st IEEE Int Joint Conf on Computational Intelligence. Piscataway, NJ: IEEE, 1998: 509-513
- [4] Ajit N, Tammy M. Quantum artificial neural network architectures and components [J]. Information Sciences, 2000, 128(3): 231-255
- [5] Xie Guangjun, Zhuang Zhenquan. Quantum neural networks [J]. Computer Science, 2001, 28(7): 1-6 (in Chinese)  
(解光军, 庄镇泉. 量子神经网络[J]. 计算机科学, 2001, 28(7): 1-6)
- [6] Xie Guangjun, Fan Haiqiu, Cao Licheng. A quantum neural computational network model [J]. Journal of Fudan University: Natural Science, 2004, 43(5): 700-703 (in Chinese)  
(解光军, 范海秋, 操礼程. 一种量子神经计算网络模型[J]. 复旦大学学报: 自然科学版, 2004, 43(5): 700-703)
- [7] Xie Guangjun, Zhou Dian, Fan Haiqiu. A neural network model based on quantum gates cell and its Applications [J]. Systems Engineering Theory & Practice, 2005, 25(5): 113-117 (in Chinese)  
(解光军, 周典, 范海秋. 基于量子门组单元的神经网络及其应用[J]. 系统工程理论与实践, 2005, 25(5): 113-117)
- [8] Maeda M, Suenaga M, Miyajima H. Qubit neuron according to quantum circuit for XOR problem [J]. Applied Mathematics and Computation, 2007, 185(2): 1015-1025

- [9] Li Panchi, Li Shiyong. A quantum self-organization feature mapping networks and clustering algorithm [J]. Chinese Journal of Quantum Electronics, 2007, 24(4): 463-468 (in Chinese)  
(李盼池, 李士勇. 一种量子自组织特征映射网络模型及聚类算法[J]. 量子电子学报, 2007, 24(4): 463-468)
- [10] Li P C, Li S Y. Learning algorithm and application of quantum BP neural networks based on universal quantum gates [J]. Journal of Systems Engineering and Electronics, 2008, 19(1): 167-174
- [11] Li Panchi. A learning algorithm and its applications to the quantum neural network model [J]. Control Theory & Applications, 2009, 26(5): 531-534 (in Chinese)  
(李盼池. 一种量子神经网络模型学习算法及应用[J]. 控制理论与应用, 2009, 26(5): 531-534)
- [12] Li P C, Song K P, Yang E L. Model and algorithm of neural networks with quantum gated nodes [J]. Neural Network World, 2010, 20(2): 189-206
- [13] Giuliano B, Giulio C, Giuliano S. Principles of Quantum Computation and Information, Volume I: Basic Concepts [M]. Translated by Wang Wenge, Li Baowen. Beijing: Science Press, 2011 (in Chinese)  
(Giuliano B, Giulio C, Giuliano S. 量子计算与量子信息原理[M]. 王文阁, 李保文, 译. 北京: 科学出版社, 2011)
- [14] Martin T H, Howard B D, Mark H B. Neural Networks Design [M]. Translated by Dai Kui. Beijing: China Machine Press, 1996 (in Chinese)

(Martin T H, Howard B D, Mark H B. 神经网络设计[M]. 戴葵, 译. 北京: 机械工业出版社, 1996)

- [15] Hecht-Nielsen R. Theory of the back propagation neural network [C] //Proc of the 1st Int Joint Conf on Neural Networks. New York: IEEE Communications Society, 1989: 593-605



**Li Panchi**, born in 1969. Received his BSc and MSc degrees from Northeast Petroleum University, China, in 1998 and 2004, respectively, and PhD degree from Harbin Institute of Technology, China, in 2009. Currently professor in the School of Computer and Information Technology, Northeast Petroleum University, China. His current research interests include quantum neural networks and quantum optimization algorithms.



**Zhou Hongyan**, born in 1990. Received her BSc degree from Northeast Petroleum University, China, in 2013. Currently MSc candidate in the School of Computer and Information Technology, Northeast Petroleum University, China. Her current research interests include neural networks and optimization algorithms.