

## 2013 Special Issue

## Single-hidden-layer feed-forward quantum neural network based on Grover learning

Cheng-Yi Liu<sup>a,b</sup>, Chein Chen<sup>a</sup>, Ching-Ter Chang<sup>c</sup>, Lun-Min Shih<sup>d,\*</sup><sup>a</sup> Department of Computer Science, National Chiao Tung University, 1001 University Road, Hsinchu 300, Taiwan<sup>b</sup> Department of Information Management, TransWorld University, 1221 Zhennan Road, Douliu, Yunlin 640, Taiwan<sup>c</sup> Department of Information Management, Chang Gung University, 259 Wen-Hwa 1st Road, Kwei-Shan, Taoyuan 333, Taiwan<sup>d</sup> Department of Accounting Information, DaYeh University, 168 University Road, Changhua 515, Taiwan

## HIGHLIGHTS

- A single-hidden-layer quantum neural network is proposed.
- Our model is proposed based on some concepts and principles in the quantum theory.
- Grover learning is presented for training the network.
- Some experiments are taken to compare the proposed method with other SLFNNs.

## ARTICLE INFO

## Keywords:

Neural network  
Quantum computing  
Grover algorithm

## ABSTRACT

In this paper, a novel single-hidden-layer feed-forward quantum neural network model is proposed based on some concepts and principles in the quantum theory. By combining the quantum mechanism with the feed-forward neural network, we defined quantum hidden neurons and connected quantum weights, and used them as the fundamental information processing unit in a single-hidden-layer feed-forward neural network. The quantum neurons make a wide range of nonlinear functions serve as the activation functions in the hidden layer of the network, and the Grover searching algorithm outstands the optimal parameter setting iteratively and thus makes very efficient neural network learning possible. The quantum neuron and weights, along with a Grover searching algorithm based learning, result in a novel and efficient neural network characteristic of reduced network, high efficient training and prospect application in future. Some simulations are taken to investigate the performance of the proposed quantum network and the result show that it can achieve accurate learning.

© 2013 Elsevier Ltd. All rights reserved.

## 1. Introduction

In the past seventy years, artificial neural networks (ANNs) have made rapid developments and been successfully applied into a great deal of practical scientific and engineering problems (Bishop, 1995; Hastie, Tibshirani, & Friedman, 2001; Haykin, 1994). Feed-forward neural networks (FNNs) and recurrent neural networks (RNNs) are two major types of popular artificial neural networks. The single-hidden-layer feed-forward neural network (SLFNN) is one of the most widely used ANNs, which have no lateral connections and/or cyclic connections and whose features resort to parameters of the weighted connections and hidden nodes (Erdogmus, Fontenla-Romero, Principe, Alonso-Betanzos, & Castillo, 2005; Huang, Chen, & Siew, 2006; Huang, Zhu, & Siew, 2006; Kim

& Adali, 2003; Liang, Huang, Saratchandran, & Sundararajan, 2006). Modification of these adjustable parameters in the ANNs allows the network to learn an arbitrary vector mapping from the space of inputs to the outputs. Finding an approximate set of parameters which can minimize the defined performance function is often realized by iterative learning. However, this process is very tedious and the optimization result relies heavily on the learning algorithm and the complexity of performance function. Moreover, the classical feed-forward neural networks are also facing many difficulties, including the dimensionality calamity, the determination of the best architecture, the limited memory capacity, time-consuming training, and so on.

In recent years there has been an explosion of interest in quantum computing. Quantum processing allows the solution of an optimization problem through the exhaustive search undertaken on all the possible solutions of the problem itself, and now it has been applied to several scientific fields such as physics, mathematics, and an extension to the entire field of computational

\* Corresponding author. Tel.: +886 937237596.

E-mail address: [lmshih@mail.dyu.edu.tw](mailto:lmshih@mail.dyu.edu.tw) (L.-M. Shih).

intelligence (Ezhov & Ventura, 2000; Fiasché, 2012; Friedman, Patel, Chen, Tolpygo, & Lukens, 2000; Gupta & Zia, 2001; Han, Gao, Su, & Nie, 2001; Han & Kim, 2002; Kak, 1995; Narayanan & Menneer, 2000; Panella & Martinelli, 2007; Perkowski, 2005; Platel, Schliebs, & Kasabov, 2009). Nowadays several essential ideas in the quantum computing have formed the basis for researches of quantum intelligent computing, including the linear superposition, entanglement and so on.

Linear superposition is closely related to the familiar mathematical linear combination. The basis of linear superposition state, and in general the superposition coefficient of the state is complex. Entangled quantum system is the potential to show the classical correlation. From a computing point of view, entanglement seems to be intuitive enough. In fact, the existence of quantum superposition makes the correlation exist. When the consistency lost, the communication correlation is probably between the qubits in some way.

Recently there have been growing interests in ANNs based on some concepts and principles in quantum theory (Ezhov & Ventura, 2000; Gupta & Zia, 2001; Kak, 1995; Narayanan & Menneer, 2000; Panella & Martinelli, 2007; Perkowski, 2005). The quantum system lays a foundation of the microcosmic systems for all the physical process, including the biologic process and mental process. Several works have combined the quantum computing with the traditional evolution algorithms that simulate the biological evolution (Fiasché, 2012; Han & Kim, 2002; Platel et al., 2009). So the quantum system is more suitable for the description of the complex biological evolution as well as the biological neurons.

Combining quantum computing with training and implementation of neural networks has been studied by many researches (Ababneh & Qasaimeh, 2006; Ezhov & Ventura, 2000; Fiasché, 2012; Friedman et al., 2000; Gupta & Zia, 2001; Han et al., 2001; Han & Kim, 2002; Kak, 1995; Karayiannis et al., 2006; Kretzschmar, Bueler, Karayiannis, & Eggimann, 1996; Levy & McGill, 1993; Malossini, Blanzieri, & Calarco, 2008; Narayanan & Menneer, 2000; Narayana & Moore, 1996; Panella & Martinelli, 2007; Patel, 2001; Perkowski, 2005; Platel et al., 2009; Purushothaman & Karayiannis, 1997). In 1997, Quantum neural network with multi-level activation function is firstly proposed by Karayiannis, which uses quantum ideas superposition in the quantum theory (Bishop, 1995; Haykin, 1994). The multi-stage activation function in the network is a linear activation Sigmoid function in the hidden layer of QNN, which is the so-called superposition. Each Sigmoid function has a different quantum interval. By adjusting the Quantum interval, the data can be mapped to different spaces that are determined by the quantum level. Given appropriate training algorithm, the network can adaptively extract the inner rules even if class boundaries or regression functions are blurred. Therefore, in the classification task, if the feature vector in the boundary between the classes overlap, QNN will be assigned to all classes (Bandyopadhyay, Karahaliloglu, Balkir, & Pramanik, 2005; Barbosa, Vellasco, Pacheco, Bruno, & Camerini, 2000; Hou, 2011; Mukherjee, Chowdhury, Raka, & Bhat-tacharyya, 2011; Qiao & Ruda, 1999a, 1999b; Takahashi, Kurokawa, & Hashimoto, 2011; Xuan, 2011). Therefore, the multi-quantum level of activation is the key reason that QNN can solve fuzzy classification and regression effectively (Gao, Zhang, Liu, Chen, & Ni, 2010; Guowei, Ning, & Deyou, 2010; Huifang & Mo, 2010; Ji, Liu, Yu, & Wu, 2011; Li & Xu, 2009; Liu, Peng, & Yang, 2010; Sagheer & Metwally, 2010; Xianwen, Feng, Lingfeng, & Xianwen, 2010; Yan & Xia, 2010).

In this paper, we establish a quantum neuron and weights based feed-forward neural network model, single hidden layer feed-forward quantum neural network (SLFQNN), and propose a Grover learning algorithm based on the quantum parallelism and entanglement. The activation function of quantum neurons in the hidden layer is characteristic of quantum coherence and quantum

transition, whose type is not fixed. The coherent coefficient can be adjusted according to the problems to be solved, and a Grover based quantum learning algorithm is used to train the network for a fast and accurate learning. Some simple examples are also given to prove its superiority.

The rest of this paper is organized as follows. Section 2 introduces the single-hidden-layer feed-forward quantum neural network. In Section 3, the Grover based quantum learning is explained in detail. In Section 4, some experiments are taken to investigate the performance of our proposed method by comparing it with other related methods. The conclusions are finally summarized in Section 5.

## 2. Single-hidden-layer feed-forward quantum neural network (SLFQNN)

In a real human brain, there are many different kinds of neurons that process different information. In quantum computation, a quantum state can be looked as a superposition of many ground states. Inspired by it, we take quantum hidden neurons in SLFNN, which is in a state of quantum superposition that is a combination of many ground states. The method of superposition can be described by the amplitude of quantum probability and the position of quantum jump. For an explicit understanding of the quantum system, firstly we have a presentation on the quantum theory.

### 2.1. Quantum bits

Information is stored in the smallest unit in the two state quantum computer is called a quantum bit or qubit. A qubit is in the two dimensional Hilbert complex unit vector. For the purposes of Quantum computing, the ground state  $|0\rangle$  and  $|1\rangle$  represent the classical bit values 0 and 1. However, unlike classical bits, a qubit can in the superposition of state  $|0\rangle$  and  $|1\rangle$ .

In a quantum system, the quantum state is the carrier of information, and a quantum state  $|\Psi\rangle$  can be regarded as a superposition of many ground states (Kak, 1995; Narayanan & Menneer, 2000):

$$|\Psi\rangle = \sum_{n=1}^N C_n |n\rangle \quad \left( \sum_{n=1}^N |C_n|^2 = 1 \right), \quad (1)$$

with  $|n\rangle$  is the ground state with corresponding coefficient  $C_n$ , whose amplitude is the appear probability of the state  $|n\rangle$ . Given

an  $m$ -bit quantum system:  $\begin{bmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_m \\ \beta_1 & \beta_2 & \cdots & \beta_m \end{bmatrix}$ , where  $|\alpha_i|^2 + |\beta_i|^2 = 1$  ( $i = 1, 2, \dots, m$ ), it can represent the superposition of states

and a 3-bit quantum system is:  $\begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{\sqrt{3}}{2} & \frac{1}{2} \\ \frac{\sqrt{2}}{1} & \frac{1}{2} & \frac{\sqrt{3}}{2} \\ \frac{\sqrt{2}}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$ . So the states of

this system are:  $\frac{\sqrt{3}}{4\sqrt{2}}|000\rangle + \frac{3}{4\sqrt{2}}|001\rangle + \frac{1}{4\sqrt{2}}|010\rangle + \frac{\sqrt{3}}{4\sqrt{2}}|011\rangle + \frac{\sqrt{3}}{4\sqrt{2}}|100\rangle + \frac{3}{4\sqrt{2}}|101\rangle + \frac{1}{4\sqrt{2}}|110\rangle + \frac{\sqrt{3}}{4\sqrt{2}}|111\rangle$ . The result above means the appear probabilities of the states  $|100\rangle$ ,  $|101\rangle$ ,  $|110\rangle$ ,  $|111\rangle$ ,  $|000\rangle$ ,  $|001\rangle$ ,  $|010\rangle$ ,  $|011\rangle$  are  $\frac{3}{32}$ ,  $\frac{9}{32}$ ,  $\frac{1}{32}$ ,  $\frac{3}{32}$ ,  $\frac{3}{32}$ ,  $\frac{9}{32}$ ,  $\frac{1}{32}$ ,  $\frac{3}{32}$  respectively, i.e., we can get the information of 8 states from a 3-qubit system. Additionally, along with the convergence of the quantum chromosomes, the diversity fade away and the algorithm converges.

Similar to the quantum state  $|\Psi\rangle$ , the neuron in human brain is just in such a quantum superposition and accordingly we call it quantum neuron. This quantum neuron has such an activation function  $\bar{f}(N, C, x)$  (Friedman et al., 2000):

$$\bar{f}(N, C, x) = \frac{1}{N} \sum_{n=1}^N |C_n|^2 f_n(x - \Delta_n). \quad (2)$$

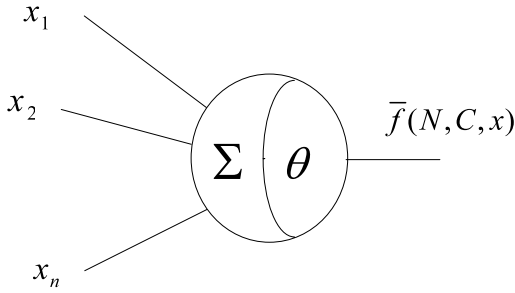


Fig. 1. The model of quantum neuron.

Here  $f_n(x)$  ( $n = 1, \dots, N$ ) is a nonlinear basis function. According to the quantum mechanics of biological brain neuron, the activation function of artificial neurons can be associated to the solution of basic such as Schrodinger's equation for various types of external potential. If the external potential is zero or constant then such basis functions will be Gaussian functions; if the external potential is harmonic such basis functions will be Hermite functions; moreover, a Morlet wavelet basis function can also be expanded by a spectral superposition of Gaussian basis functions. For other types of wavelet functions, the probability density functions associated to quantum mechanical equations can be approximated by wavelet expansions, however, these expansions do not coincide with the mathematical solutions of the associated partial differential equations. So in the formula (2) the admissible neuron activation function  $f(x)$  can be radial Gaussian function, or Wavelets functions,  $x$  is the input data. They are corresponding to the quantum ground states at different energy levels, and  $\Delta_n$  is the quantum transition position of these states. Accordingly, the output of quantum neuron  $\bar{f}(\cdot)$  will be in a superposition state that is formed by these ground states. The model of quantum neuron is shown in Fig. 1.

## 2.2. Quantum neural network

By adopting the quantum neurons as the hidden neurons of a SLFNN, we can establish a SLFQNN. Assume the connected weights in the output layer are represented by  $\bar{W}$ ,  $g$  is the activation function of the hidden layer (often a classical linear function is adopted), there is a hidden neuron in SLFQNN, so the output of the network is  $y = g(\bar{W}, \bar{f}(N, C, x))$  (Haykin, 1994). Here  $N$  is the number of ground states of quantum neurons;  $C$  is the amplitude of appear probability of states in quantum neuron. According to the result of Kreinovich in 1991, a nonlinear function in the hidden layer can make SLFNN approximate any continuous function on the condition of a weak limitation (Kreinovich, Vazquez, & Kosheleva, 1991), so SLFQNN can be proved to have generalized approximation ability. In the training, by adjusting  $|C_n|^2$ , we can get optimal activation functions of hidden layer in SLFQNN by the subsequent learning.

## 3. Grover based quantum learning

In comparison with the classical hidden neurons in ANNs, SLFQNN has two new parameters: the superposition coefficient and the jump position. Assume an SLFQNN with the structure  $L_i - 1 - L_o$ , if the number of the training samples is  $P$  and the number of the ground states is  $N$ . Firstly we define such a cost function  $J$  (Bishop, 1995):

$$J = \frac{1}{2} \sum_{i=1}^P e_i^2 = \frac{1}{2} \sum_{i=1}^P \sum_{j=1}^{L_o} (y_{i,j} - d_{i,j})^2 \quad (3)$$

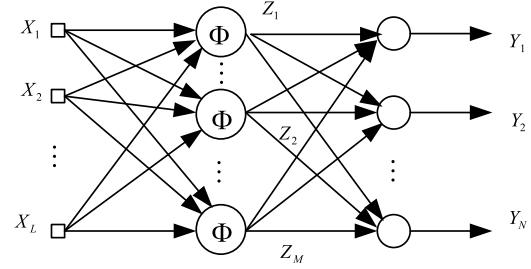


Fig. 2. The quantum neural network.

where  $y_{i,j}$  is the output of the  $j$ -th neuron for the  $i$ -th training sample:

$$\begin{aligned} y_{i,j} &= g(\bar{W}, \bar{f}(N, C, x)) \\ &= \frac{1}{N} \sum_{l=1}^M |D_l|^2 \bar{w}_{lj} \sum_{n=1}^N |C_n|^2 f_n(x_i - \Delta_n) \\ &\quad (i = 1, \dots, P, j = 1, \dots, L_o), \end{aligned} \quad (4)$$

and  $d_{i,j}$  is the corresponding expected output,  $\bar{w}_{lj}$  is the connected weights between the  $j$ -th output neuron and the hidden neuron. Searching for an approximate set of parameters including the weights  $W$  and the undetermined parameters in quantum neuron, including the quantum superposition state  $f_n$  and quantum transition  $\Delta_n$  in the possible solution space to minimize the cost function is the goal of training the network.

Assume the type of quantum neuron is coded with  $l_1 = \log_2(N)$  qubits, the transition of quantum neuron is coded with  $l_2 = \log_2(Q)$  and quantum weights  $W$  are coded with  $l_3 = \log_2(M)$  quantum bits. Unlike classical bits, qubits are in the superposition of states  $|0\rangle$  and  $|1\rangle$ . At some time, the type of hidden quantum neuron is in the superposition state (Panella & Martinelli, 2007):

$$|\psi\rangle = \frac{1}{\sqrt{N}} \sum_{n=1}^N |f_n\rangle; \quad (5)$$

and the transition of the hidden neuron is also in such a superposition state:

$$|\varphi\rangle = \frac{1}{\sqrt{Q}} \sum_{n=1}^Q |\Delta_n\rangle. \quad (6)$$

The quantum weights are in the state:

$$|W\rangle = \frac{1}{\sqrt{M}} \sum_{k=1}^M |w_k\rangle \quad (7)$$

which is the superposition of all possible states  $w_k \in [0, 1, \dots, 2^l - 1]$ , and each state is corresponding to a determined weights. Consider a database of  $S$  elements exactly one of which is "marked" as satisfying some desirable characteristic, Grover's algorithm uses the parallelism of afforded by quantum superposition to accomplish the task with only  $O(S^{1/2})$  queries (Liu et al., 2010). In this paper, we introduce the quantum Grover algorithm to search for the optimal weights and neurons that can maximize the defined performance function. The network is shown in Fig. 2.

Defining  $1/J$  as the performance function of the network:

$$F(\bar{W}, f, \Delta) = 1/J. \quad (8)$$

Grover searching algorithm is to compute the performance function of different states in parallel and search the best parameter set with the highest performance function. Firstly define a  $|q\rangle$  with  $l_1 + l_2 + l_3$  quantum bits, which is in a combination state of  $|\psi\rangle$ ,  $|\varphi\rangle$  and  $|q\rangle$ . Then additional  $b$  quantum bits  $|c\rangle$  is

the control register to control the appearance of the optimal state and the obtained  $|q\rangle$  is of length  $n = l_1 + l_2 + l_3 + b$ :

$$|q\rangle : |\psi, \varphi, W, c\rangle.$$

The process of searching for the solution that maximizes the performance function can be described as the following steps. First, represent the performance function  $F$  as:

$$F(\bar{W}, f, \Delta) = F(q_1, q_2, \dots, q_n). \quad (9)$$

Define the unity quantum operation  $U$  as (Trugenberger, 2002):

$$U = \exp\left(i\frac{\pi}{2}F_{\text{nor}}(q_1, \dots, q_n)\right) \quad (10)$$

with

$$F_{\text{nor}}(q_1, \dots, q_n) = \frac{F(q_1, \dots, q_n) - F_{\text{min}}}{F_{\text{max}} - F_{\text{min}}}. \quad (11)$$

Here  $F_{\text{max}}, F_{\text{min}}$  represent the upper and lower bound of the performance function respectively. Perform the unity quantum operation  $U$  on the state  $|q\rangle$ :

$$U = \prod_{k=1}^{l_1+l_2+l_3} G^k \quad (12)$$

where  $G^k$  is the  $k$ -bit diagonals quantum gate as follows (Grover, 1997):

$$G^k = \text{diag}(e^{i\frac{\pi}{2}F_{\text{nor}}^k(0_1, \dots, 0_k)}, \dots, e^{i\frac{\pi}{2}F_{\text{nor}}^k(1_1, \dots, 1_k)}) \quad (13)$$

$$F_{\text{nor}}^k(q_1, \dots, q_k) = \frac{F(q_1, \dots, q_k) - \frac{1}{K}F_{\text{min}}}{F_{\text{max}} - F_{\text{min}}}$$

$$\left(K = \sum_{k=1}^{l_1+l_2+l_3} k! \binom{n}{k}\right). \quad (14)$$

Firstly the  $b$  control bits in  $|q\rangle$  are initialized as all zeros:

$$|q^0\rangle = \frac{1}{\sqrt{n}} \sum_{k=1}^n (q_k : 0_1, \dots, 0_b). \quad (15)$$

Perform the Hadamard gate  $H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$  on the first control bit, we can obtain:

$$|q^1\rangle = \frac{1}{\sqrt{2n}} \sum_{k=1}^n (q_k : 0_1, \dots, 0_b) + \frac{1}{\sqrt{2n}} \sum_{k=1}^n (q_k : 1_1, \dots, 0_b). \quad (16)$$

Introduce the control gate (Trugenberger, 2002):

$$U_{cq}^{\pm} = |1_c\rangle\langle 1_c| \otimes U_q + |0_c\rangle\langle 0_c| \otimes U_q^{-1} \quad (17)$$

which represent that if the control bit takes “1”, perform unity operation  $U$  on the state  $|q^i\rangle$ ; if the control bit takes “0”, then perform  $U^{-1}$  on the state  $|q^i\rangle$ , which can be realized by employing the control gate:

$$U_{cq}^{\pm} = \prod_{k=1}^m \prod (CG^{k-2})G^k. \quad (18)$$

Perform  $U_{c_1q}^{\pm}$  on  $|q^2\rangle$ , we can obtain,

$$|q^2\rangle = \frac{1}{\sqrt{2n}} \sum_{k=1}^n e^{-i\frac{\pi}{2}F_{\text{nor}}(q_k)} |q_k : 0_1, \dots, 0_b\rangle + \frac{1}{\sqrt{2n}} \sum_{k=1}^n e^{i\frac{\pi}{2}F_{\text{nor}}(q_k)} |q_k : 1_1, \dots, 0_b\rangle. \quad (19)$$

Apply the Hadamard gate  $H$  on the first control bit  $c_1$ , we can obtain,

$$|q^3\rangle = \frac{1}{\sqrt{n}} \sum_{k=1}^n \sin\left(\frac{\pi}{2}F_{\text{nor}}(q_k)\right) |q_k : 0_1, \dots, 0_b\rangle + \frac{1}{\sqrt{n}} \sum_{k=1}^n \cos\left(\frac{\pi}{2}F_{\text{nor}}(q_k)\right) |q_k : 1_1, \dots, 0_b\rangle. \quad (20)$$

So performing the transform  $H_c U_{cq}^{\pm} H_c$  can fulfill the performance function estimation of  $|q\rangle$ , and consequently we perform such a transform on each control bit  $C_1, \dots, C_b$ , and finally we get the result (Trugenberger, 2002):

$$|q^{pf}\rangle = \frac{1}{\sqrt{n}} \sum_{k=1}^n \sum_{i=0}^b \cos^{b-i}\left(\frac{\pi}{2}F_{\text{nor}}(q_k)\right) \times \sin^i\left(\frac{\pi}{2}F_{\text{nor}}(q_k)\right) \sum_{\{J^i\}} |q^k : J^i\rangle \quad (21)$$

where  $\{J^i\}$  represent the length of binary string with  $i$  “zero” and  $b - i$  “1”. In this procedure,  $b$  times of operation  $H_c U_{cq}^{\pm} H_c$  is used to amplify the probability amplitude of the state with higher performance function. The state with highest performance function should have a lot of zero control bit. When all the bits in the control register being in zero, the amplitude of the optimal state reaches the maximum. Commonly speaking, the required state can be obtained by repeatedly performing the above determined quantum transformation and random quantum measurements. The expected iterative times needed to obtain the optimal state is  $1/P_b^0$ , and

$$P_b^0 = \frac{1}{n} \sum_{k=1}^n \sin^{2b}\left(\frac{\pi}{2}F_{\text{nor}}(q_k)\right) \quad (22)$$

is the probability of  $|c_1, \dots, c_b\rangle = |0_1, \dots, 0_b\rangle$ . Once the control register detects the desired state, the measurement can be performed on  $|q\rangle$ , and then we can obtain the state  $|q_k\rangle$  with the probability  $P_b(q_k)$ . We can see from it that with the  $F_{\text{nor}}(q_k)$  approaches 1,  $P_b(q_k)$  becomes bigger and bigger. Finally, as we have expected,  $P_b$  achieves the maximum at the state with the highest performance function.

So the Grover based quantum learning algorithm can be described as:

---

*The procedure of the Grover based learning algorithm:*

---

*Step 1:* Initialize the parameters of the network:

$$|q\rangle : |q_1, q_2, \dots, q_{l_1}, q_1, q_2, \dots, q_{l_2}, q_1, q_2, \dots, q_{l_3}, c_1, c_2, \dots, c_b\rangle$$

where  $|q_1, q_2, \dots, q_{l_1}\rangle$  code the type of quantum neuron;

$|q_1, q_2, \dots, q_{l_2}\rangle$  code the transition of quantum neuron;

$|q_1, q_2, \dots, q_{l_3}\rangle$  code the connection weights of output layer

and  $|c_1, c_2, \dots, c_b\rangle$  are the control bits. The  $c_1, \dots, c_b$  control bits in  $|q\rangle$  are initialized as all zeros:

*Step 2:* Defining  $1/J$  as the performance function of the network, and perform the unity quantum operation  $U$  on the state  $|q\rangle$ .

*Step 3:* Perform the quantum transform  $H_c U_{cq}^{\pm} H_c$  on the coding bits and control bits to estimate the performance function of the states. Repeat this process for  $P_b^0$  times.

*Step 4:* Perform the observation on the quantum neuron and weights, which will collapse into one state, and it is the best state having the highest performance function.

---

**Table 1**  
The probability of falling into a local optimal solution.

Length of $b$	Length of $l_1, l_2, l_3$	$R = F_{\text{local max}}/F_{\text{global max}}$			
		$R = 0.2000$	$R = 0.500$	$R = 0.800$	$R = 1.000$
$b = 1$	2	0.500	0.452	0.250	0.048
	3	0.354	0.320	0.177	0.034
	4	0.125	0.226	0.063	0.024
$b = 2$	2	0.500	0.409	0.125	0.005
	3	0.354	0.289	0.088	0.003
	4	0.125	0.205	0.063	0.002
$b = 3$	2	0.500	0.370	0.063	4.354e−004
	3	0.354	0.262	0.044	3.078e−004
	4	0.125	0.185	0.031	2.177e−004

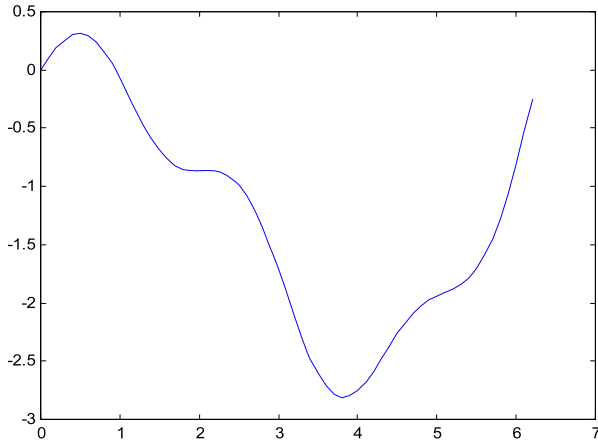


Fig. 3. The function to be approx.

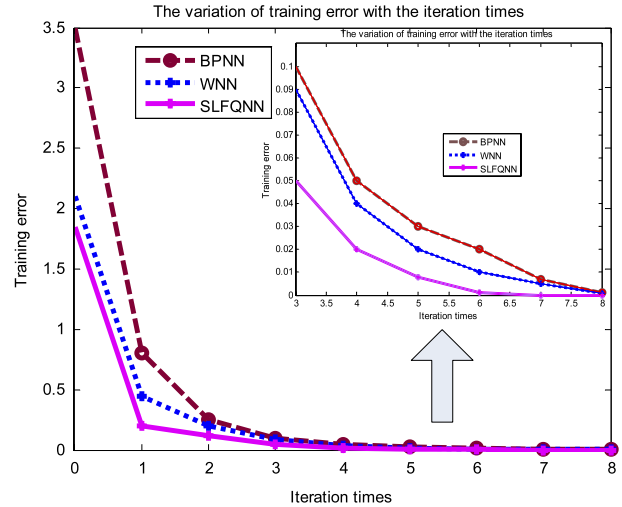


Fig. 4. The convergence curves of BPNN, WNN and SLFQNN.

**4. Experimental results**

In this section, some experiments are taken to investigate the performance of the proposed method.

**Experiment 1.** In the following, a simple example is used to illustrate the efficiency of the proposed Grover quantum learning algorithm. Firstly we investigate the probability of Grover algorithm based quantum learning falling into a local optimal solution. The result is shown in Table 1. From it we can see that when the number of control bits reaches 3 and the information bits reaches 4, the possibility of falling into a local optimal solution becomes very small. So as long as the bits are defined in  $|q\rangle$ , Grover based quantum learning is feasible.

**Experiment 2.** Considering the problem of the function approximation, the function to be approximated is

$$y(x) = \sin x + \frac{\sin 3x}{3} - 2 \sin \frac{x}{2}. \tag{23}$$

The function is shown in Fig. 3.

In SLFQNN Gaussian function is adopted as the fundamental activation function of the network. The difference among different  $f_n$  relies on the width of Gaussian function, and the transition  $\Delta_n$  reflects the centers of Gaussian function. Let  $l_1 = l_2 = l_3 = 4$  and  $b = 3$ . BP-NN model is a feed forward neural network with the learning rule of error back propagation. It has a three-layer (or single-hidden-layer) network structure: the input layer, hidden layer and the output layer. In the network, the parameters of the network are tuned using the traditional gradient descent algorithm. It provides a possible way of finding an approximated optimal solution to the network parameters and is very popular in the

practical application of ANNs. It is also a representative neural network model that has the activation functions with global support. WNN is an improved version of the BP-NN, which replace the Sigmoid function in the hidden layer by a local and multiscale wavelet function, followed by a linear output layer. In the training process of the network, the parameters are also adjusted by the traditional gradient descent algorithm, including the connected weights of the network, the scale and position of the hidden neurons. Because the local property of wavelet functions in time and frequency, WNN is more efficient than BP-NN in approximating and classification tasks. It proves to present good results in many engineering fields. Therefore we compare these two networks with our proposed SLFQNN. In this experiment, BP-NN, WNN and SLFQNN are used to approximate the function in (21). In the training, the initial weight and neuron parameters are selected randomly. The SLFQNN network is trained according to the algorithm above.

The result shows that BPNN (Backpropagation neural network), SLFQNN and WNN (wavelet neural network) all can converge within eight epochs to reach the error below 0.01. The SLFQNN converges rapider than BPNN and WNN, as shown in Fig. 4. However, for SLFQNN, it only requires four iterations to reach the error  $1.3741e-3$ . Additionally the classical neural network needs more hidden neurons to obtain a satisfactory solution, however, only one hidden neuron is employed in our proposed SLFQNN.

**Experiment 3.** In this test, we applied it to the task of classification, and the data is from <http://www.research.att.com/~yann/ocr/mnist> in AT&T Bell laboratory in America. The number of test samples is 1000, and  $N = 3$ . Moreover, two binary classification datasets came from UCI dataset: Bld (Bupa Liver Disorders), Ttt

**Table 2**  
Result of classification.

Dataset	# training samples	RBF NN		SLFQNN	
		Error	$L_h$	Error	$L_h$
AT&T Bell	100	2.4	72	2.0	1
	196	1.8	127	1.8	4
	289	1.1	206	0.9	10
Bld (Bupa Liver Disorders)	400	5.2	143	4.1	7
Ttt (tic-tac-toe endgame)	300	5.9	221	4.4	8

(tic-tac-toe endgame) are also used to test the proposed method. RBFNN is a popular single-hidden-layer feed-forward neural network that takes the radial-basis-function as the activation function in the hidden layer. The activation function has the multiscale characteristics and it is considered as a special case of the WNN. Different with WNN, RBFNN calculates the distance between the input pattern and the centers in the hidden layer. Because the centers of the hidden neurons can be determined by some supervised clustering algorithm, the training of RBFNN is rapid than that of WNN. In the experiment, we compare the proposed SLFQNN with RBFNN, which uses the Gaussian function as the activation function in the hidden layer and works well in function fitting, and the result is shown in Table 2. Here  $L_h$  represents the number of hidden layer. From it we can see that the classical neural network needs a great deal of hidden neurons to obtain a satisfactory solution, however, only a small number of hidden neurons are needed to reach the same precision in SLFQNN.

## 5. Conclusions

In this paper, a single hidden layer feedforward quantum neural network is proposed based on the combination of quantum theory and neural network. In the network, the hidden neurons are quantum neurons that are characteristic of the superposition and transition of multiple states, which can produce a wide range of nonlinear activation functions in the hidden layer of the network. Moreover, the Grover searching algorithm is proposed to outstand the optimal parameter setting iteratively and thus makes very efficient neural network learning possible. The quantum neuron and weights, along with a Grover searching algorithm based learning, result in a novel and efficient neural network characteristic of reduced network, high efficient training and prospect application in future.

## References

- Ababneh, J. I., & Qasaimeh, O. (2006). Simple model for quantum-dot semiconductor optical amplifiers using artificial neural networks. *IEEE Transactions on Electron Devices*, 53(7), 1543–1550.
- Bandyopadhyay, S., Karahaliloglu, K., Balkir, S., & Pramanik, S. (2005). Computational paradigm for nanoelectronics: self-assembled quantum dot cellular neural networks. *IEE Proceedings—Circuits, Devices and Systems*, 152(2), 85–92.
- Barbosa, C., Hall, V., Vellasco, M., Pacheco, M. A., Bruno, A. C., & Camerini, C. S. (2000). Nondestructive evaluation of steel structures using a superconducting quantum interference device magnetometer and a neural network system. *Review of Scientific Instruments*, 71(10), 3806–3815.
- Bishop, C. (1995). *Neural networks for pattern recognition*. Oxford: Oxford Univ. Press.
- Erdogmus, D., Fontenla-Romero, O., Principe, J. C., Alonso-Betanzos, A., & Castillo, E. (2005). Linear-least-squares initialization of multilayer perceptrons through backpropagation of the desired response. *IEEE Transactions on Neural Networks*, 16(2), 325–337.
- Ezhov, A. A., & Ventura, D. (2000). Quantum neural networks. In N. Kasabov (Ed.), *Future directions for intelligent systems and information sciences* (pp. 213–234). Berlin, Germany: Springer-Verlag.
- Fiasché, M. (2012). A quantum-inspired evolutionary algorithm for optimization numerical problems. In *LNCS: Vol. 7665. ICONIP 2012, Part III* (pp. 686–693). (Part 3).
- Friedman, J. R., Patel, V. V., Chen, W., Tolpygo, S. K., & Lukens, J. E. (2000). Quantum superposition of distinct macroscopic states. *Nature*, 406(6791), 43–46.
- Gao, Kim, Zhang, Yan, Liu, Ying-hui, Chen, Xiao-mei, & Ni, Guo-qiang (2010). PSF estimation for Gaussian image blur using back-propagation quantum neural network. In *2010 IEEE 10th international conference on signal processing, ICSP*. (pp. 1068–1073). 24–28 October.
- Grover, L. K. (1997). Grover searching algorithm. *Physical Review Letters*, 79, 325.
- Guowei, Cai, Ning, Liu, & Deyou, Yang (2010). The transformer fault diagnosis based on quantum neural network. In *T2010 International conference on computer, mechatronics, control and electronic engineering, Vol. 4, CMCE*. (pp. 396–400). 24–26 August.
- Gupta, S., & Zia, R. K. P. (2001). Quantum neural networks. *Journal of Computer and System Sciences*, 63, 355–383.
- Han, M., Gao, X., Su, J. Z., & Nie, S. (2001). Quantum-dot-tagged microbeads for multiplexed optical coding of biomolecules. *Nature Biotechnology*, 19(7), 631–635.
- Han, K. H., & Kim, J. H. (2002). Quantum-inspired evolutionary algorithm for a class of combinatorial optimization. *IEEE Transactions on Evolutionary Computation*, 6(6), 580–593.
- Hastie, T., Tibshirani, R., & Friedman, J. (2001). *The elements of statistical learning: data mining, inference and prediction*. Springer.
- Haykin, S. (1994). *Neural networks: a comprehensive foundation*. New York: Macmillan.
- Hou, Xuan (2011). Research of model of quantum learning vector quantization neural network. In *2011 International conference on electronic and mechanical engineering and information technology, Vol. 8, EMEIT*. (pp. 3893–3896). 12–14 August.
- Huang, G.-B., Chen, L., & Siew, C.-K. (2006). Universal approximation using incremental constructive feed-forward networks with random hidden nodes. *IEEE Transactions on Neural Networks*, 17(4), 879–892.
- Huang, G.-B., Zhu, Q.-Y., & Siew, C.-K. (2006). Real-time learning capability of neural networks. *IEEE Transactions on Neural Networks*, 17(4), 863–878.
- Huifang, Li, & Mo, Li (2010). A new method of image compression based on quantum neural network. In *2010 International conference of information science and management engineering, Vol. 1, ISME*. (pp. 567–570). 7–8 August.
- Ji, Xiao, Liu, Yawen, Yu, Xiaowei, & Wu, Jiangbin (2011). Alternative combination of quantum immune algorithm and back propagation neural network. In *2011 Seventh international conference on natural computation, Vol. 1, ICNC*. (pp. 66–70). 26–28 July.
- Kak, S. C. (1995). Quantum neural computing. *Advances in Imaging and Electron Physics*, 94, 259–314.
- Karayiannis, N. B., Mukherjee, A., Glover, J. R., Frost, J. D., Hrachovy, R. A., Jr., & Mizrahi, E. M. (2006). An evaluation of quantum neural networks in the detection of epileptic seizures in the neonatal electroencephalogram. *Soft Computing*, 10(4), 382–396.
- Kim, T., & Adali, T. (2003). Approximation by fully complex multilayer perceptrons. *Neural Computation*, 15, 1641–1666.
- Kreinovich, V., Vazquez, A., & Kosheleva, O. (1991). Prediction problem in quantum mechanics is intractable (NP-hard). *International Journal of Theoretical*, 30(2), 113–122.
- Kretzschmar, R., Bueler, R., Karayiannis, N.B., & Eggimann, F. (2000). Quantum neural networks versus conventional feedforward neural networks: an experimental study. In *Neural Networks for Signal Processing X. Proceedings of the 2000 Signal Processing Society Workshop: Vol. 1* (pp. 328–337).
- Levy, H. J., & McGill, T. C. (1993). A feed-forward artificial neural network based on quantum effect vector-matrix multipliers. *IEEE Transactions on Neural Networks*, 4(3), 427–433.
- Liang, N.-Y., Huang, G.-B., Saratchandran, P., & Sundararajan, N. (2006). A fast and accurate online sequential learning algorithm for feed-forward networks. *IEEE Transactions on Neural Networks*, 17(6), 1411–1423.
- Li, Fei, & Xu, Guobiao (2009). Quantum BP neural network for speech enhancement. In *Asia-Pacific conference on computational intelligence and industrial applications, 2009, Vol. 2, PACIIA 2009*. (pp. 389–392). 28–29 November.
- Liu, Kai, Peng, Li, & Yang, Qin (2010). The algorithm and application of quantum wavelet neural networks. In *2010 Chinese control and decision conference, CCDC*. (pp. 2941–2945). 26–28 May.
- Malossini, A., Blanzieri, E., & Calarco, T. (2008). Quantum genetic optimization. *IEEE Transactions on Evolutionary Computation*, 12(2), 231–241.
- Mukherjee, Sankha Subhra, Chowdhury, Raka, & Bhattacharyya, Siddhartha (2011). Image restoration using a multilayered quantum backpropagation neural network. In *2011 International conference on computational intelligence and communication networks, CICN*. (pp. 426–430). 7–9 October.
- Narayanan, A., & Menneer, T. (2000). Quantum artificial neural network architectures and components. *Information Sciences*, 128, 231–255.
- Narayana, A., & Moore, M. (1996). Quantum-inspired genetic algorithms. In *Proceedings of IEEE international conference on evolutionary evolution* (pp. 61–66).
- Panella, M., & Martinelli, G. (2007). Binary neuro-fuzzy classifiers trained by nonlinear quantum circuits. In F. Masulli, S. Mitra, & G. Pasi (Eds.), *Lecture notes in artificial intelligence: Vol. 4578. Applications of fuzzy sets theory* (pp. 237–244). Berlin, Germany: Springer-Verlag.
- Patel, Apoorva (2001). Why genetic information processing could have a quantum basis. [arXiv:quant-ph/0105001v2](https://arxiv.org/abs/quant-ph/0105001v2).

- Perkowski, M. A. (2005). Multiple-valued quantum circuits and research challenges for logic design and computational intelligence communities. *IEEE Connections*, 3(4), 6–12.
- Platel, M. D., Schliebs, S., & Kasabov, N. (2009). Quantum-inspired evolutionary algorithm: a multimodel EDA. *IEEE Transactions on Evolutionary Computation*, 13(6), 1218–1232.
- Purushothaman, G., & Karayiannis, N. B. (1997). Quantum neural networks (QNNs): inherently fuzzy feed-forward neural networks. *IEEE Transactions on Neural Networks*, 8(3), 679–693.
- Qiao, Bi, & Ruda, Harry E. (1999a). Evolution of a two-dimensional quantum cellular neural network driven by an external field. *Journal of Applied Physics*, 85(5), 2952–2961.
- Qiao, Bi, & Ruda, Harry E. (1999b). Jordan block structure of two-dimensional quantum neural cellular network. *Journal of Applied Physics*, 86(1), 634–639.
- Sagheer, A., & Metwally, N. (2010). Communication via quantum neural networks. In *2010 Second world congress on nature and biologically inspired computing, NaBIC*. (pp. 418–422). 15–17 December.
- Takahashi, Kazuhiko, Kurokawa, Motoki, & Hashimoto, Masafumi (2011). Controller application of a multi-layer quantum neural network trained by a conjugate gradient algorithm. In *IECON 2011-37th annual conference on IEEE industrial electronics society* (pp. 2353–2358). 7–10 November.
- Trugenberger, C. A. (2002). Quantum optimization for combinatorial searches. *New Journal of Physics*, 4, 26.1–26.2.
- Xianwen, Ren, Feng, Zhang, Lingfeng, Zheng, & Xianwen, Men (2010). Application of quantum neural network based on rough set in transformer fault diagnosis. In *2010 Asia-Pacific power and energy engineering conference, APPEEC*. (pp. 1–4). 28–31 March.
- Xuan, Hou (2011). Research on quantum adaptive resonance theory neural network. In *2011 International conference on electronic and mechanical engineering and information technology, Vol. 8, EMEIT*. (pp. 3885–3888). 12–14 August.
- Yan, Zhou, & Xia, Wu (2010). Quantum neural network algorithm based on multi-agent in target fusion recognition system. In *2010 International conference on computational and information sciences, ICCIS*. (pp. 537–540). 17–19 December.