# Learning Algorithm and Application of Quantum Neural Networks with Quantum Weights

Dianbao Mu, Zunyou Guan, and Hong Zhang

***Abstract*—A novel neural networks model with quantum weights is present. Firstly, based on the information processing modes of biology neuron and quantum computing theory, a quantum neuron model is presented, which is composed of weighting, aggregating, activating, and inspiriting. Secondly the quantum neural networks model based on quantum neuron is constructed in which the input and the output are real vectors; the linked weight and the activation value are qubits. On the basic of the gradient descent algorithm, a learning algorithm of this model is proposed. It is shown that this algorithm is super-linearly convergent under certain conditions and can increase the probability of getting the global optimal solution. Finally, the availability of the model is illustrated in both convergence speed and convergence rate by two application examples of pattern recognition and function approximation.**

***Index Terms*—Quantum computing, quantum neuron, quantum neural network, super-linear convergence**

## I. Introduction

Artificial Neural Network (ANN) has widely been applied to every field since M-P model of neuron is proposed. ANN has gained the great development through decades of research, at the same time some problems is also appeared. Because it is based on the simplistic neuron model, ANN cannot satisfy the increasing demands of quantity and complexity of information. Therefore, it is one of the farther directions of development and research for the ANN to construct the perfect ANN theory and to endure with the more comprehensive mathematics basis, biology feature, and physical feature [1].

The combination of quantum computation and neural networks is a rising subject of ANN theory research, and the research of Quantum Neural Networks (QNN) has just begun. At present, many QNN research results have been acquired in international since the comparability comment on quantum theory and neural network theory is proposed by Perus. Menneer and Narayanan proposed a method in which introduce the multi-universe theory in quantum mechanics into neural network training, exists a neural network corresponding each sample in training set, and total networks is made up of superposition of these networks. Behrman et al. proposed a QNN model realized by an array of quantum dot molecules. In this model, a single quantum dot molecule is adopted for every input, the system is evolved in real time according to quantum mechanics, measures is performed by fixed time interval, and the different time pieces are regarded as the neurons in hidden layer. Therefore, the more time pieces the measures applies, the more hidden neurons the model has. As present, the QNN research direction approximately includes: the quantum associative memory [2], [3], the quantum competition learning [4], the neural network applied quantum dots [5], the common neural network applied quantum transform function, and the quantum Hopfield networks and so on. Ref. [6] proposed neural networks with quantum gated nodes and indicated that such quantum networks may contain some advantageous features of biological systems more efficiently than classical electronic devices in 2007. Ref. [7] has proposed a quantum BP neural networks model with learning algorithm based on single-qubit rotation gate and two-qubit controlled-NOT gates.

The research presented in this paper (a) constructs quantum neuron and quantum neural network model based on the feature of biology neuron and quantum mechanics, in which the linked weight and the activation value is expressed in qubits, (b) proposes an learning algorithm for this model, and prove this algorithm is super- linearly convergent under certain conditions, and can increase the convergence probability, (c) designs two experiments, the results show that this model and algorithm are superior to the common BP (CBP) neural network in both the convergence speed and the convergence rate.

## II. Quantum Neural Network Model

### A. Quantum Neuron Model

The biology research shows that, in biology neuron, the nerve impulse introduced from other neurons can induce the change of voltage differences between the voltage inside cell velum and that outside cell velum. This change can associate with the current activation value and can update it. When the activation value is greater than a certain threshold value, the neuron activation is inspired, the voltage inside cell velum sharp rises, and the electricity pulse (nerve impulse) is released by the neuraxon [7], [8]. Therefore, the information disposal process of the biology neuron includes four parts: weighting, aggregating, activating, and inspiriting. Some research results in the 1990s shows that the information disposal process of brain might be concerned with quantum phenomenon, the quantum mechanics effect, and the

Manuscript received November 24, 2012; revised January 25, 2013. This work was supported in part by the Information Center of Daqing Oilfield Company. Ltd.

D. B. Mu is with the visit scholar with University of Waterloo, Canada (e-mail: d2mu@uwaterloo.ca).

Z. Y. Guan is with the Information Center of Daqing Oilfield, China (e-mail: guanzy@petrochina.com.cn).

H. Zhang is with the Oil and gas well production data management system (A2) of Daqing Oilfield Company. China (e-mail: zhang_hong_a@petrochina.com.cn).

DOI: 10.7763/IJCTE.2013.V5.797

quantum system has the similar dynamics feature to the biology neural networks [9]. Therefore, the combination of ANN and quantum theory might perfectly simulate the information disposal process of brain.

Synthesizing above depiction, the quantum neuron model proposed in this paper includes four parts as follows: weighting, aggregating, activating, and inspiriting, where the linked weight value and activation value are expressed in qubits, the activation process is performed by inner product operator, and the inspiriting process adopts sigmoid function. The quantum neuron model is shown in Fig. 1.

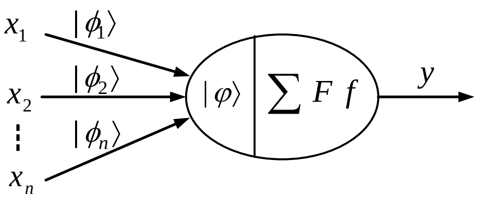

Fig. 1. The quantum neuron model

In the quantum neuron model, the linked weight value and activation value are expressed in qubits. The state of a qubit can be expressed as $|\varphi\rangle = \alpha|0\rangle + \beta|1\rangle$ ,where $\alpha$ and $\beta$ are complex numbers, $|\alpha|^2$and $|\beta|^2$ give the probabilities that the qubit will respectively be found in the $|0\rangle$ and $|1\rangle$ , and satisfy the normalization conditions as $|\alpha|^2 + |\beta|^2$ =1, The numbers $\alpha$ and $\beta$ satisfied above conditions are called the probability amplitudes of the qubit corresponding states. Therefore, qubit can also be described by the probability amplitudes as $[\alpha, \beta]^T$ .

Let $X = (x_1, \cdots, x_n)^T$ is the input vector, $y$ is the actual output, $|\varphi\rangle = (|\varphi_1\rangle, \cdots, |\varphi_n\rangle)^T = \left( \begin{pmatrix} \cos\theta_1 \\ \sin\theta_1 \end{pmatrix}, \cdots, \begin{pmatrix} \cos\theta_n \\ \sin\theta_n \end{pmatrix} \right)^T$ is the weight vector, and $|\phi\rangle = (\cos\xi, \sin\xi)^T$ is the activation value. The in-out relation of quantum neuron can be described as follows:

$$y = f(F(X^T|\varphi\rangle, |\phi\rangle)) = f(\sum_{i=1}^{n} x_i \langle\varphi_i|\phi\rangle) = f(\sum_{i=1}^{n} x_i \cos(\theta_i - \xi)) \tag{1}$$

where $F$ is the inner product operator and $f$ is the sigmoid function.

### B. Quantum Neural Network Model

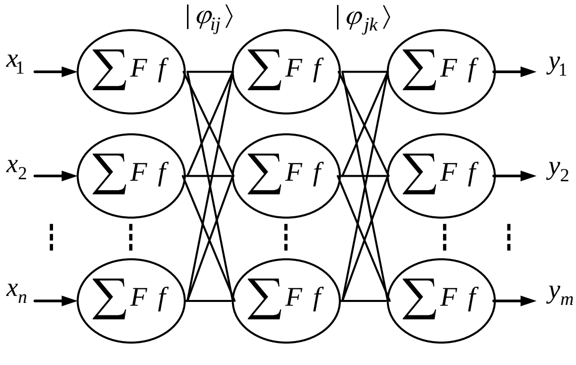

Fig. 2. Three layers quantum neural network model

The quantum neural network is made up of some quantum neuron according to definite linking rule. The three layers feed forward QNN model is presented in Fig. 2, where input layer, hidden layer, and output layer has $n$ , $p$ , and $m$ quantum neurons, respectively.

Let $x_i$ is the input of network, $h_j$ is the hidden layer output, and $y_k$ is the output of network, $|\varphi_{ij}\rangle$ is the weight value in hidden layer, $|\varphi_{jk}\rangle$ is the weight value in output layer, $|\phi_j\rangle$ is the activation value in hidden layer, and $|\phi_k\rangle$ is the activation value in output layer. The in-out relation of the QNN can be described as follows:

$$\begin{gathered} y_k = f(\sum_{j=1}^{p} h_j \langle\varphi_{jk}|\phi_k\rangle) = f(\sum_{j=1}^{p} f(\sum_{i=1}^{n} x_i \langle\varphi_{ij}|\phi_j\rangle)\langle\varphi_{jk}|\phi_k\rangle) = \\ f(\sum_{j=1}^{p} f(\sum_{i=1}^{n} x_i \cos(\theta_{ij} - \xi_j))\cos(\theta_{jk} - \xi_k)) \end{gathered} \tag{2}$$

where $i = 1, \cdots, n$ , $j = 1, \cdots, p$ , $k = 1, \cdots, m$ .

## III. Quantum Neural Network Learning Algorithm

In Fig. 2, for the qubits of the weight value and the activation value, the updating process can be performed by one-bit quantum rotation gate. The updating formulas are defined as follows:

$$|\varphi(t+1)\rangle = \begin{bmatrix} \cos\Delta\theta_t & -\sin\Delta\theta_t \\ \sin\Delta\theta_t & \cos\Delta\theta_t \end{bmatrix} |\varphi(t)\rangle \tag{3}$$

$$|\phi(t+1)\rangle = \begin{bmatrix} \cos\Delta\xi_t & -\sin\Delta\xi_t \\ \sin\Delta\xi_t & \cos\Delta\xi_t \end{bmatrix} |\phi(t)\rangle \tag{4}$$

namely: $\begin{bmatrix} \cos(\theta_{t+1}) \\ \sin(\theta_{t+1}) \end{bmatrix} = \begin{bmatrix} \cos(\theta_t + \Delta\theta_t) \\ \sin(\theta_t + \Delta\theta_t) \end{bmatrix}$ and

$$\begin{bmatrix} \cos(\xi_{t+1}) \\ \sin(\xi_{t+1}) \end{bmatrix} = \begin{bmatrix} \cos(\xi_t + \Delta\xi_t) \\ \sin(\xi_t + \Delta\xi_t) \end{bmatrix}$$

Therefore, the key matter of algorithm design is how to ascertain the rotation angle $\Delta\theta_t$ and $\Delta\xi_t$ that make the iteration sequences defined by Eqs.(5-6) rapidly converge.

$$\theta(t+1) = \theta(t) + \Delta\theta_t \tag{5}$$

$$\xi(t+1) = \xi(t) + \Delta\xi_t \tag{6}$$

The network error function is defined as follows:

$$E = \frac{1}{2}(\tilde{Y} - Y)^2 \tag{7}$$

where $\tilde{Y}$ and $Y$ is the normalized desired output and the real output, respectively. It is obvious that $E$ is the coherent continuous function of the $\theta$ and the $\xi$ .

**Lemma**. Suppose $\theta(t)\to\theta^*$ , let $\rho(t)=\theta(t)-\theta^*$ , $\delta(t)=\theta(t+1)-\theta(t)=\rho(t+1)-\rho(t)$ . Then that the sequence $\{\theta(t)\}$ is super-linearly convergent is equal to $\left\|\rho(t+1)\right\|=o\left(\left\|\delta(t)\right\|\right)$ [10].

There exists the following theorem about the computation of rotation angle $\Delta\theta$ and $\Delta\xi$ of quantum gate.

**Theorem 1**. When $\Delta\theta$ and $\Delta\xi$ is respectively computed according to Eq.(8) and Eq.(9) and $r>2$ , the iteration sequences $\{\theta(t)\}$ and $\{\xi(t)\}$ are super-linearly convergent.

$$\Delta\theta(t)=-\eta(\frac{\partial E(\theta(t),\xi(t))}{\partial\theta(t)})(E(\theta(t),\xi(t)))^{\frac{1}{r}}\tag{8}$$

$$\Delta\xi(t)=-\eta(\frac{\partial E(\theta(t),\xi(t))}{\partial\xi(t)})(E(\theta(t),\xi(t)))^{\frac{1}{r}}\tag{9}$$

where $\eta$ is the learning rate.

**Proof**. Firstly, prove the convergence of two sequences. According to Thaler formula

$$\begin{aligned}&E(\theta(t+1),\xi(t+1))-E(\theta(t),\xi(t))\\&=\begin{pmatrix}\Delta\theta(t)\\\Delta\xi(t)\end{pmatrix}^T\begin{pmatrix}\frac{\partial E(\theta(t),\xi(t))}{\partial\theta(t)}\\\frac{\partial E(\theta(t),\xi(t))}{\partial\xi(t)}\end{pmatrix}+O\left(\left\|\begin{pmatrix}\Delta\theta(t)\\\Delta\xi(t)\end{pmatrix}^T\right\|\right)\\&\approx-\eta\left(\left(\frac{\partial E(\theta(t),\xi(t))}{\partial\theta(t)}\right)^2+\left(\frac{\partial E(\theta(t),\xi(t))}{\partial\xi(t)}\right)^2\right)(E(\theta(t),\xi(t)))^{\frac{1}{r}}\end{aligned}$$

Therefore $E(\theta(t+1),\xi(t+1))-E(\theta(t),\xi(t))\le 0$ , namely, the iteration sequences $\{\theta(t)\}$ and $\{\xi(t)\}$ is monotone reductive. On the other hand, $E(\theta(t),\xi(t))$ has the lower boundary, therefore, there certainly exists the limit of sequence. Because $E(\theta(t),\xi(t))$ is the coherent continuous function of $\theta$ and $\xi$ , therefore, exist equation as follows: $\lim\limits_{t\to\infty}(\theta(t),\xi(t))=(\theta^*,\xi^*)$ , namely, the iteration sequences $\{\theta(t)\}$ and $\{\xi(t)\}$ are convergent.

Secondly, applying the method of Ref. [10], prove the super- linearity of the sequence convergence.

Let $A=\left(\frac{\partial E(\theta(t),\xi(t))}{\partial\theta(t)}\right)_{\max}$ , then

$$\begin{aligned}\frac{\left\|\rho(t+1)\right\|}{\left\|\delta(t)\right\|}&=\frac{\left\|\theta(t+1)-\theta^*\right\|}{\left\|-\eta(\frac{\partial E(\theta(t),\xi(t))}{\partial\theta(t)})(E(\theta(t),\xi(t)))^{\frac{1}{r}}\right\|}\\&\ge\frac{\left\|\theta(t+1)-\theta^*\right\|}{\eta A(E(\theta(t),\xi(t)))^{\frac{1}{r}}}\end{aligned}$$

Because $E(\theta(t),\xi(t))$ is the quadratic equation of $\theta(t)$ and $\xi(t)$ , when $r>2$ , $\left\|\theta(t+1)-\theta^*\right\|=O\left((E(\theta(t),\xi(t)))^{\frac{1}{r}}\right)$ . Therefore $\left\|\rho(t+1)\right\|=O\left(\left\|\delta(t)\right\|\right)$ . According to the lemma, the iteration sequence $\{\theta(t)\}$ is super-linearly convergent. With the same reasoning, the iteration sequence $\{\xi(t)\}$ is also super-linearly convergent.

**Theorem 2**. If $\theta^*$ and $\xi^*$ is the convergent solutions of sequence $\{\theta(t)\}$ and $\{\xi(t)\}$ , respectively, then any $\tilde{\theta}^*$ and $\tilde{\xi}^*$ defined by $\tilde{\theta}^*-\tilde{\xi}^*=\pm(\theta^*-\xi^*)$ is also the convergent solutions of sequence $\{\theta(t)\}$ and $\{\xi(t)\}$ , respectively.

**Proof**. Because $\theta^*$ and $\xi^*$ is the convergent solution of sequence $\{\theta(t)\}$ and $\{\xi(t)\}$ , respectively, according to Eq.(2), the network output is described as follows:

$$\begin{aligned}y_k&=f(\sum_{j=1}^{p}f(\sum_{i=1}^{n}x_i\cos(\theta_{ij}^*-\xi_j^*))\cos(\theta_{jk}^*-\xi_k^*))\\&=f(\sum_{j=1}^{p}f(\sum_{i=1}^{n}x_i\cos(\pm(\theta_{ij}^*-\xi_j^*)))\cos(\pm(\theta_{jk}^*-\xi_k^*)))\\&=f(\sum_{j=1}^{p}f(\sum_{i=1}^{n}x_i\cos(\tilde{\theta}_{ij}^*-\tilde{\xi}_j^*))\cos(\tilde{\theta}_{jk}^*-\tilde{\xi}_k^*))=\tilde{y}_k\end{aligned}$$

Therefore $\tilde{\theta}^*$ and $\tilde{\xi}^*$ is also the convergent solutions of sequence $\{\theta(t)\}$ and $\{\xi(t)\}$ , respectively. Where $\tilde{y}_k$ is desired output corresponding $y_k$ .

Theorem 1 assures the speediness of network convergence. Theorem 2 assures the diversity of convergent solutions, which only demands the phase difference between $|\varphi\rangle$ and $|\phi\rangle$ satisfy a certain condition, and ignore the actual phase values of $|\varphi\rangle$ and $|\phi\rangle$ . Therefore, this algorithm can evidently increase the convergence probability.

Synthesizing above depiction, the algorithm proposed in this paper is described as follows:

**Step1:** Initialize the network parameters. Including: the node number of each layer, restriction error $\varepsilon$ , the maximum of iterative steps Max , set the current iterative step $t=0$ .

**Step2:** Initialize the weights value and the activation values.

Hidden layer:

$$\theta_{ij}(0)=2\pi\times\text{Rnd}\;;\xi_j(0)=2\pi\times\text{Rnd}\;;\left|\varphi_{ij}\right.>=\begin{bmatrix}\cos(\theta_{ij}(0))\\\sin(\theta_{ij}(0))\end{bmatrix};$$

$$\left|\phi_j\right.>=\begin{bmatrix}\cos(\xi_j(0))\\\sin(\xi_j(0))\end{bmatrix}$$

Output layer:

$$\theta_{jk}(0)=2\pi\times\text{Rnd}\;;\xi_k(0)=2\pi\times\text{Rnd}\;;\left|\varphi_{jk}\right.>=\begin{bmatrix}\cos(\theta_{jk}(0))\\\sin(\theta_{jk}(0))\end{bmatrix};$$

$$\left|\phi_k\right.>=\begin{bmatrix}\cos(\xi_k(0))\\\sin(\xi_k(0))\end{bmatrix}$$

where $i=1,\cdots,n$ , $j=1,\cdots,p$ , $k=1,\cdots,m$ , Rnd is a random number in [0,1].

**Step3:** Compute network output according to Eq.(2), and update the weight value and activation value of each layer according to Eqs.(3-4).

**Step4:** Compute the output error according to Eq.(7), if $E < \varepsilon$ or $t > \text{Max}$, then go to step5,else $t = t + 1$, go back Step3.

**Step5:** Save the weight values and activation values of each layer.

## IV. Comparison Experiment

In the classical computer, the information disposal process is performed by logic gate. The quantum states in quantum register evolve through the quantum gate. In Hilbert space, as its linear restriction, the quantum gate would simultaneously have an effect on all quantum ground states, which is equal to computing $2^n$ numbers at one time, however, any classical computer need to repeat $2^n$ computations or need to $2^n$ parallel processor to the same task. This is the parallel quantum computation. In the other words, the entanglement and the superposition of quantum information in the quantum computer presents the great advantage, which is proved by Shor algorithm. However, the hardware realization of quantum circuit and quantum algorithm has been proved to be an austere challenge, especially, the applied quantum computers are being researching [11].

Therefore, the algorithm proposed in this paper cannot be realized in computer equipped with the quantum hardware, and cannot test the great advantage brought by the parallel quantum computation. But the quantum theory makes the QNN differ from the common neural network in information process mode. In QNN, the weight value qubits are rotated in a unit circle by the quantum rotation gate. According to theorem 1, these qubits can soon be matched to the optimum place by setting the proper quantum gates rotation angle. According to theorem 2, the number of the optimum solutions in the QNN is far more than that in the common neural network, and can evidently increase the convergence probability, which can be tested in the common computer.

To testify the validity of QNN, two kinds of experiments are designed and the QNN is compared with the common BP network (CBP) in this part. To make comparison equitable, the QNN adopts the same structure and parameters as CBP in the experiments.

### A. Pattern Recognition

For Nine sample points in Fig. 3, the samples data are shown in Table I. Determine the pattern of each point by QNN. This is a typical two-pattern classification problem, which is regard as the generalization of XOR problem.

TABLE I: The Sample Data of Nine-Sample Patterns Classification Problem

| Input | 0, 0 | 0, 1 | 1, 0 | 1, 1 | 0, 2 | 2, 0 | 1, 2 | 2, 1 | 2, 2 |
|---|---|---|---|---|---|---|---|---|---|
| Output | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 |

The network construct is 2-10-1, and the learning rate is 0.8. According to three different restriction errors, The QNN and the CBP is learned 50 times, restively, then is computed the average of convergence steps. Learning results are shown in Table II

TABLE II: The Learning Result Comparison Between QNN and CBP for Nine-Sample Patterns Problem

| Restriction error | Restriction steps | QNN | | CBP | | |
|---|---|---|---|---|---|---|
| | | Iteration steps | Convergence rate | Iteration steps | Convergence rate | Necessary steps for convergence |
| 0.10 | 100 | 65.78 | 98.36% | 100 | Non | 605.36 |
| 0.05 | 500 | 221.56 | 94.51% | 500 | Non | 1693.21 |
| 0.01 | 5000 | 3735.64 | 86.67% | 5000 | Non | 14238.53 |

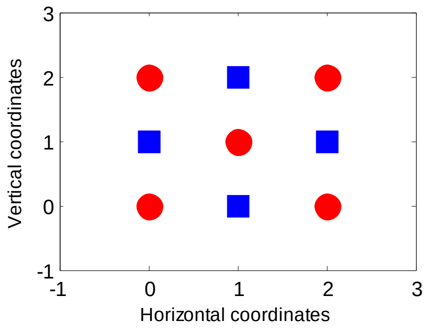

Fig. 3. Nine-sample patterns

### B. Function Approximation

Take example for the trigonometric function holding singular value, compute 11 discrete value of the following function $f(x) = \cos(1/x)$ in range [1/( $2\pi$ +1),1]. These discrete values are shown in Table III.

TABLE III: 11 Discrete Values of $f(x) = \cos(1/x)$

| $x$ | $f(x) = \cos(1/x)$ |
|---|---|
| 0.1373 | 0.5403 |
| 0.1503 | 0.9317 |
| 0.1659 | 0.9672 |
| 0.1852 | 0.6333 |
| 0.2096 | 0.0575 |
| 0.2415 | -0.5403 |
| 0.2846 | -0.9317 |
| 0.3466 | -0.9672 |
| 0.4431 | -0.6333 |
| 0.6141 | -0.0575 |
| 1.0000 | 0.5403 |

On the basis of the data in Table IV, approximate the function $f(x) = \cos(1/x)$ in range [1/( $2\pi$ +1),1] by QNN. The network construct is set to 2-20-1, the learning rate is set to 0.5, the restriction error is set to 0.05, and the restriction steps are set to 15000. The QNN reaches convergence after 12242 steps, here, approximation error is 0.0499, and approximation curve is shown in Fig. 4. When the CBP holding the same structure as the QNN is applied to approximate $f(x) = \cos(1/x)$ , the network does not reach to convergence after 15000 steps, and the approximation error is up to 0.0627.

## V. Conclusion

The QNN is the combination of quantum computing and nerve computing, which has the advantage such as parallelism and high efficiency of quantum computation besides continuity, approximation capability, and computation capability that the general ANN has. In QNN, the weight value and the activation value are expressed by qubits, and the phase of each qubit is updated by the quantum rotation gate. Since both probability amplitudes participate in optimizing computation, the computation capability is evidently superior to the general BP network. Experiment proves that the QNN model and algorithm proposed in this paper is effective.

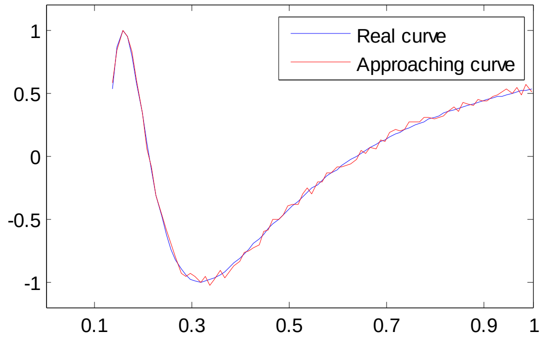

Fig. 4. Function approximation curve

## REFERENCES

[1] J. L. Cheng, D. H. Feng, and F. Liu, *Immune Optimization Computation, Learn and Recognition*, Science Press, Beijing, 2006.

[2] D. Ventura and T. R. Martinez, "Quantum associative memory," *Information Sciences*, vol. 124, pp. 273-296, 2000.

[3] M. Perus and P. Ecimovic, "Memory and pattern recognition in associative neural networks," *International Journal of Applied Science and Computation*, vol. 4, pp. 283-310, 1998.

[4] P. Pyllkkanen and P. Pylkko, "New directions in cognitive science," in *Proc. the International Symposium. Saariselka*, pp. 77-89, 1995.

[5] E. Behrman, "A quantum dot neural networks," in *Proc. Workshop Physics of Computation. Cambridge*, pp. 22-24, 1996.

[6] F. Shafee, "Neural networks with quantum gated nodes," *Engineering Applications of Artificial Intelligence*, vol. 20, no. 4, pp. 429-437, 2007.

[7] C. P. Li and S. Y. Li, "Learning algorithm and application of quantum BP neural networks based on universal quantum gates," *Journal of Systems Engineering and Electronics*, vol. 19, no. 1, pp. 167-174, 2008.

[8] S. Y. Li, *Fuzzy Control, Neurocontrol and Intelligent Cybernetics*, Harbin Institute of Technology Press, Harbin, 1998.

[9] M. Perus, "Neuro-quantum parallelism in brain-mind and computers," *Informatics*, vol. 20, pp. 173-183, 1996.

[10] L. J. Zhen, X. G. He, and D. S. Huang, "Super-linearly convergent bp learning algorithm for feed forward neural networks," *Journal of Software*, in Chinese, vol. 11, no. 8, pp. 1094-1096, 2000.

[11] P. Knight, "Quantum information processing without entanglement," *Science*, vol. 287, pp. 441-442, 2000.

**Dianbao Mu** was born on March 9th 1975 in Harbin Heilongjiang Province, China, graduated from the Daqing Petroleum Institute in 2000 years, majored in Computer Software. In 2004 start to learn in Jilin University in Jilin province, China and got master degree in 2007 majored in computational Mathematics. In 2012, he went to University of Waterloo Canada as a visit scholar. He started to work as a computer engineer in the No.1 Oil Production Company, in 2004, he moved to the Information Center of Daqing Oilfield Company. Ltd. In 2010, he started to work as a senior engineer and started the research of Quantum theory, and put emphases on the application in Oilfield. During these years, he finished more than 10 projects As technique expert, "Design of Takagi-Sugeno Fuzzy Controller Based on Improved Quantum Genetic Algorithm" was published in Computer Engineering in 2011.

**Zunyou Guan** was born on January 21, 1966 in Songyuan City, Jilin Province, China, got bachelor's degree in Daqing Petroleum Institute in 1988, majored in Drilling Engineering, got master's degree in Jilin University in 2007, majored in Software Engineering. He started to work in information center of No1 Drilling Company of the Daqing Petroleum Administration Bureau from 1991 to 1995, since 1995, worked in the Information Center of Daqing Oilfield Co., Ltd., a senior engineer, deputy director. He has made great achievements and published numerous papers during these years. "The Digital Development Framework and Implementation Strategy of Daqing Oilfield" was published in "The Digital Chemical" in 2004; in 2005, "A Scalable Data Quality Element Model" was published in "Computer Engineering", "A Large-scale Information Systems Project Implementation and Management Practice on Information Systems Engineering " published in 2012, organized by the Information Center of Tianjin.

**Hong Zhang** was born on October 27, 1975 in Daqing Heilongjiang Province, China. Graduated from the Shenyang Industrial College in 1995, majored in Machinery manufacturing. He started to work as a computer engineer in the No.1 Oil Production Company in 2000.in 2011, he moved to the Daqing oilfield exploration and Development Institute. During these years, he finished more than 4 projects as technology expert, and has published many articles. Now, he is in charge of the Oil and gas well production data management system (A2) of Daqing Oilfield Company.