Archives of Computational Methods in Engineering (2019) 26:793–807
https://doi.org/10.1007/s11831-018-9269-0

ORIGINAL PAPER

Recent Developments and Applications in Quantum Neural Network: A Review

S. K. Jeswal[1] · S. Chakraverty[1]

Received: 6 August 2017 / Accepted: 28 April 2018 / Published online: 3 May 2018

Abstract

Quantum neural network is a useful tool which has seen more development over the years mainly after twentieth century. Like artificial neural network (ANN), a novel, useful and applicable concept has been proposed recently which is known as quantum neural network (QNN). QNN has been developed combining the basics of ANN with quantum computation paradigm which is superior than the traditional ANN. QNN is being used in computer games, function approximation, handling big data etc. Algorithms of QNN are also used in modelling social networks, associative memory devices, and automated control systems etc. Different models of QNN has been proposed by different researchers throughout the world but systematic study of these models have not been done till date. Moreover, application of QNN may also be seen in some of the related research papers. As such, this paper includes different models which have been developed and further the implement of the same in various applications. In order to understand the powerfulness of QNN, few results and reasons are incorporated to show that these new models are more useful and efficient than traditional ANN.

1 Introduction

The concept of artificial neural network has been proposed around 1950s mainly to mimic the different activities of human brain. An artificial neural network (ANN) is a parallel distributed information processor made up of identical units (neurons) capable of storing information and make it available for use. Over the years, quantum computing has seen exceptional development which has a great impact on faster computing. Moreover, quantum computing is more powerful in comparison to classical computing. After the evolution of quantum computing, it gives us unprecedented possibilities in solving problems beyond the abilities of classical computers. For example, Shor [1] proposed the first quantum algorithm that suggest how to factorize a very large integer and Grover's algorithm [2] gives a better performance as compared to the classical computation in searching an unsorted database. Quantum computing is added with traditional neural network in giving a new concept known as quantum neural network which is now explored by different researchers from all most every part of the world. Traditional ANN is not capable of low cost learning due to its computational power, whereas computational power of QNN is more as compared to its traditional counterpart, so one may use QNN instead of ANN. In 1980, Benioff [3] gave the concept of quantum computation. For the first time the concept of quantum neural computing has been given by Kak [4] in the year 1995. Further, Kak [4] examined the concept of quantum neural computing in the context of various new directions in neural network research. In 1995, A quantum-inspired neural network (QINN) has been proposed by Menneer and Narayanan [5]. Perus [6] discussed about absorbing comparability between neural networks and quantum parallelism. Hypothetical model of the QNN using optical interference has been proposed by Vlasov [7]. For the first time, a detail description as well as systematic examination of quantum neural network has been investigated in the Ph.D. thesis of Menneer [8] in 1998. In 2000, Ventura and Martinez [9] proposed a quantum implementation of the associative memory model. Behrman [10] with his colleagues gave the idea of physical implementation of the quantum neural network as an array of quantum dots. Narayanan and Menneer [11] experimentally as well as by simulation shown that quantum ANN is not only more

✉ S. Chakraverty
sne_chak@yahoo.com

S. K. Jeswal
sumit_jeswal@rediffmail.com

[1] Department of Mathematics, National Institute of Technology Rourkela, Rourkela, Odisha 769008, India

efficient but also more powerful as compared to traditional ANN for a number of experimental problems. In 2001, Altaisky [12] proposed a quantum perceptron model which is "network with a teacher". Mainly in the current scenario the quantum neural network model proposed are mostly self-organized network that is network without a teacher. Gupta and Zia [13] defined a novel computational model namely quantum neural network based on the concept of Deutsch's model of quantum computational network. In 2003, Qubit neural network has been introduced by Kouda et al. [14].

It has been shown that quantum neural networks can work better with some classical and quantum components as well. Schuld et al. [15] have given the quest for QNN model which combines the quantum computing with the unique properties of neural computing. They [15] studied different methods to develop QNN model and presented a systematic approach to quantum neural network research. To access the scope of the QNN models, Schuld et al. [15] proposed few requirements for meaningful QNN. The requirements are given as

a. Any binary string of length N can be encoded by the initial state of the quantum system.
b. Few basic neural computing mechanisms can be reflected by the quantum neural network.
c. The evolution is based on quantum phenomenon as for example entanglement, interference and linear superposition.

Ezhov and Ventura [16] have discussed advantages of quantum neural networks as compared to the classical ANN, from which few of them are listed below

- quantum parallelism;
- memory capacity increased exponentially as compared to classical counterpart;
- capacity to learn is faster as compared to traditional one;
- higher and good stability and reliability;
- performance is high with less number of hidden neurons;
- single layer network solution of linearly inseparable problems;
- information processing speed is high;

Due to these type of features, researchers motivated towards the development of quantum neural networks. there are various future scope for development in quantum neural network which makes it quite interesting.

Rest of the paper is organized in the following manner. In Sect. 2, we have given basic quantum concepts. Different QNN models are discussed in Sect. 3. In Sect. 4, some other models and different algorithms proposed by different researchers are addressed. Some applications of quantum neural network are then discussed in Sect. 5. Finally in Sect. 6, conclusions are drawn.

2 Basic Quantum Concepts

Smallest unit of classical computing is bit, similarly qubit (quantum bit) is the smallest unit in quantum information processing.

The notation $|\cdot\rangle, \langle\cdot|$ is called "Dirac notation" mainly used in quantum computation, which represents the standard notation for the states in the quantum mechanics given after the name of famous theoretical physicist Paul Dirac. $|\cdot\rangle$ is a ket vector which in general a column vector and $\langle\cdot|$ is a bra vector which is complex conjugate transpose of ket vector, represents a row vector. If we operate a matrix with ket vector, we get a ket vector again. Together bra and ket give an inner product that is combining $\langle x|$ and $|y\rangle$ as $\langle x \mid y\rangle$ denotes an inner product of two vectors which always gives a scalar quantity. For example $\langle 1 \mid 1\rangle = \langle 0 \mid 0\rangle = 1$ and $\langle 0 \mid 1\rangle = \langle 1 \mid 0\rangle = 0$. Combining ket and bra gives an outer product that is $|x\rangle\langle y|$ is the outer product of $|x\rangle$ and $\langle y|$ which is in fact an operator. For example

$$|0\rangle\langle 0| = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, |0\rangle\langle 1| = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, |1\rangle\langle 0| = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}, |1\rangle\langle 1| = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$$

A quantum state $|y\rangle$ is a superposition of the basis states $|0\rangle$ and $|1\rangle$ which can be expressed as

$$|y\rangle = \alpha_0|0\rangle + \beta_0|1\rangle = (\alpha_0, \beta_0)^T \tag{1}$$

where α_0 and β_0 are complex numbers such that $|\alpha_0|^2 + |\beta_0|^2 = 1$

α_0 and β_0 are also called as the probability amplitudes. A qubit is a unit vector in a two dimensional complex vector space for example Hilbert space for which a particular basis denoted by $\{|0\rangle, |1\rangle\}$ has been fixed. In general we can denote $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ and $|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$.

The qubit state $|y\rangle$ collapses into either $|0\rangle$ state with probability $|\alpha_0|^2$ or $|1\rangle$ state with probability $|\beta_0|^2$. These complex-valued probability amplitudes have four real numbers, one of these is fixed by the normalization condition. Then the qubit state (1) can be given by

$$|y\rangle = e^{i\gamma}\left(\cos\phi|0\rangle + e^{i\delta}.\sin\phi|1\rangle\right) \tag{2}$$

where γ, δ, ϕ are real valued parameters.

Now we systematically discuss some terms related to quantum computation such as (1) Linear Superposition, (2) Coherence and Decoherence, (3) Operator, (4) Interference, (5) Tensor product, (6) Entanglement, (7) Quantum Gates, (8) Quantum Parallelism.

1. Linear Superposition [16, 17] is the linear combination of vectors. Quantum systems are given by a wave function y that exists in a Hilbert space. The Hilbert space has a set of states $|\phi_i\rangle$ that form a basis, and the system is described by a quantum state $|y\rangle$. In general we can represent it as

$$|y\rangle = \sum_i P_i|\phi_i\rangle$$

where $|y\rangle$ is said to be a linear superposition of the basis states $|\phi_i\rangle$ and $P_i = \langle\phi_i \mid y\rangle$ is the probability amplitude.
2. Coherence and Decoherence [16] are two useful ideas related to the concept of linear superposition. A quantum system is said to be coherent if it is in a linear superposition of its basis states. Decoherence is the destruction of quantum state correlations or a state in a linear superposition which interacts with the environment as a result the superposition is destroyed.
3. Operators [16, 17] are mainly used to describe the transformation of a wave function into another. We denote the operator by a capital letter, for example $\hat{U}$ and in general operators represent in the form of matrices acting on vectors. Based on the concept of operator, an eigenvalue equation is given as $\hat{U}|\phi_i\rangle = \lambda_i|\phi_i\rangle$, where λ_i's are the eigenvalues and $|\phi_i\rangle$ are the eigenstates of the solution to the given equation. In quantum environment, the operators are always linear and in addition to that they must also be unitary that is $\hat{U}^\dagger\hat{U} = \hat{U}\hat{U}^\dagger = \hat{I}$, where $\hat{U}^\dagger$ is the complex conjugate transpose and $\hat{I}$ is the identity operator.
4. Interference [16, 17] is an useful wave phenomenon. Wave peaks that in phase interfere constructively while those that are out of phase interfere destructively. Interference effects can be observed in all type of waves such as water, radio, light waves etc.
5. Tensor product is the process where two smaller vectors merge to form a larger one by combining elements from each in all possible ways that preserve both linearity and scalar multiplication. For example

$$|00\rangle = |0\rangle \otimes |0\rangle = \begin{pmatrix}1\\0\end{pmatrix} \otimes \begin{pmatrix}1\\0\end{pmatrix} = \begin{pmatrix}1\begin{pmatrix}1\\0\end{pmatrix}\\0\begin{pmatrix}1\\0\end{pmatrix}\end{pmatrix} = \begin{pmatrix}1\\0\\0\\0\end{pmatrix}$$

Likewise we can define

$$|01\rangle = \begin{pmatrix}0\\1\\0\\0\end{pmatrix}, |10\rangle = \begin{pmatrix}0\\0\\1\\0\end{pmatrix}, |11\rangle = \begin{pmatrix}0\\0\\0\\1\end{pmatrix}$$

Similarly we can generalize the product.
6. Entanglement is a quantum phenomenon which is a property of a multi-qubit system. Let us consider a quantum state $|y\rangle_1 \in H_1$ and another state $|y\rangle_2 \in H_2$ where H_1 and H_2 are the Hilbert Spaces, then after combining the states we have $|y\rangle_{12} \in H_1 \otimes H_2$. This combined state may be or may not be written as a tensor product. If the tensor product can be expressed as

$$|y\rangle_{12} = |y\rangle_1 \otimes |y\rangle_2$$

then we can say that the state is separable, otherwise the combined state is entangled. Hence the condition for entanglement is

$$|y\rangle_{12} \neq |y\rangle_1 \otimes |y\rangle_2$$

For example let us consider the quantum state

$$|\phi\rangle = \frac{1}{\sqrt{2}}|10\rangle + \frac{1}{\sqrt{2}}|11\rangle$$

which can also be written in vector form as

$$|\phi\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix}0\\0\\1\\1\end{pmatrix}$$

and the outer product of the given state is as follow

$$|\phi\rangle\langle\phi| = \frac{1}{2}\begin{pmatrix}0&0&0&0\\0&0&0&0\\0&0&1&1\\0&0&1&1\end{pmatrix}$$

Again the outer product matrix can be factorized as

$$\rho = \frac{1}{\sqrt{2}}\left(\begin{pmatrix}0&0\\0&1\end{pmatrix} \otimes \begin{pmatrix}1&1\\1&1\end{pmatrix}\right)$$

So the state is separable.

Few examples of entangled states are $|\phi\rangle = \frac{1}{\sqrt{2}}(|00\rangle \pm |11\rangle)$, $|\phi\rangle = \frac{1}{\sqrt{2}}(|01\rangle \pm |10\rangle)$.
7. Quantum Gates are unitary linear transformations of the quantum state vectors. Single qubit rotation gate can be represented as [18, 61]

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

Suppose the quantum state is given by $|\phi\rangle = \begin{bmatrix} \cos\phi \\ \sin\phi \end{bmatrix}$ and $|\phi\rangle$ can be transformed by the rotation gate $R(\theta)$ as [18]

$$R(\theta)|\phi\rangle = \begin{bmatrix} \cos(\phi+\theta) \\ \sin(\phi+\theta) \end{bmatrix}$$

where $R(\theta)$ shifts the phase of the quantum state $|\phi\rangle$.

There are gates that are most useful in quantum computing that may be single qubit gates or two qubit gates or three qubit gates etc. Here we have discussed few examples of single qubit gates, two qubit gates and three qubit gates below.

A single qubit gate is a unitary operator which transforms a single qubit state $|\phi\rangle_{in}$ to another single qubit state $|\phi\rangle_{out}$, that is $|\phi\rangle_{out} = \hat{U}|\phi\rangle_{in}$, where $\hat{U}$ is the unitary operator. Some of the useful gates in quantum computing are Pauli gates (Pauli-X, Pauli-Y, Pauli-Z), Hadamard gate, Phase shift gate and Rotation gates which are single qubit gates. Here we consider only Hadamard gate which is frequently used in QNN and so the same is discussed below.

Hadamard gate is a single qubit gate. The unitary matrix corresponding to this gate can be represented as [19, 39]

$$H = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

The action of this gate is given by

$$H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

and $H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$

We can represent it in bra-ket notation as follows

$$H = \frac{1}{\sqrt{2}}[(|0\rangle + |1\rangle)\langle 0| + (|0\rangle - |1\rangle\langle 1|)]$$

On the other hand, Controlled-NOT gate, Swap gate, Controlled-U gates are some examples of two qubit gates. Controlled-NOT gate is an important example of two qubit gate popularly known as CNOT gate, which complement the second qubit if the first qubit is in the state $|1\rangle$ and leaves the second qubit as it is otherwise. Then the matrix representation for CNOT gate is given by

$$CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

We can represent it in bra-ket notation as follows

$$CNOT = |00\rangle\langle 00| + |01\rangle\langle 01| + |11\rangle\langle 10| + |10\rangle\langle 11|$$

Similarly few three cubic gates will be mentioned as Toffoli gate, Fredkin gate, Deutsch gate.

8. Quantum parallelism is the method where a quantum computer is able to perform two computations simultaneously. Physicist David Deutsch coined the term, so as to distinguish it from classical parallel computation in standard computers. Quantum parallelism is a unique quantum mechanical phenomenon that can be used to build quantum algorithms.

   Learning process is the foremost and most important in traditional as well as in quantum neural networks. There are mainly two type of learning, that is supervised or learning with a teacher and unsupervised or learning without a teacher. In learning with a teacher where we know the desired output vector for each input vector from training set. Each time a vector from the training set is applied to the network, the weights of the network are updated, bringing the output pattern as close as possible to the pattern of the desired result. But in case of unsupervised learning, we do not know about the desired output.

3 Different QNN Models

Here we have systematically discussed various types of QNN model proposed till date to the best of our knowledge.

3.1 Quantum M-P Neural Network
3.2 Quantum Competitive Neural Network
3.3 Quantum-Inspired Neural Network
3.4 Quantum Dot Neural Network
3.5 Quantum Cellular Neural Network
3.6 Qubit Neural Network
3.7 Quantum Associative Neural Network

In the subsequent heads of (3.1), (3.2), (3.5) we have given the classical model as well as the quantum model.

3.1 Quantum M-P Neural Network

Using quantum linear superposition, A quantum M-P neural network model has been proposed by Zhou and Ding [20] after analyzing traditional M-P neural network based on the concept of quantum computation. They [20] have not only given the detail working principle of quantum M-P neural network but also the weight updating algorithm has been described for two cases of input state being in the orthogonal and non-orthogonal basic set respectively.

3.1.1 Scalar Product

The scalar product of two quantum state $\phi_s(a)$ and $\phi_t(a)$ is given by [20]

$$\phi_s(a) \cdot \phi_t(a) = \int_{\Omega} \phi_s(a) \cdot \phi_t(a) da$$

This is the scalar product of the continuous function. While computing numerically the continuous basis is converted to discrete one, and the scalar product can be computed as [20]

$$\phi_s(a) \cdot \phi_t(a) = \sum_a \phi_s(a) \cdot \phi_t(a) \tag{3}$$

Let us consider $\phi_s(a) = \left(a_1, \ldots, a_p\right)$ and $\phi_t(b) = \left(b_1, \ldots, b_p\right)$, then we can generalize (3) as

$$\phi_s(a) \cdot \phi_t(b) = a_1 b_1 + a_2 b_2 + \cdots + a_p b_p$$

3.1.2 Traditional M-P Model

McCulloch and Pitts developed the first logical model named as M-P model for ANN in 1943. In the traditional M-P model, the processing units of ANN must add the weight w_j with the input and process all to get an output. Let y_k is the summation of the input a_j and weight value w_j. The neuron activation mainly depends on a threshold value θ. If the threshold is more than y, then the neuron is activated and gives an output O_k. Input and output of this model is related and given by means of a function f which in general is a nonlinear function called as activation function. Mathematically, the above description can be written as [20]

$$y_k = \sum_j a_j w_j$$

$$O_k = f(y_k - \theta)$$

3.1.3 Quantum M-P Model

The quantum M-P model combined the traditional M-P model with the quantum basics. Here the output is given as [20]

$$O_k = \sum_j w_{kj} \phi_j,\ j = 1, 2, \ldots, 2^n$$

where n is the total number of qubits needed.

A two qubits quantum M-P model is given with the four kinds of input such as $|00\rangle, |01\rangle, |10\rangle, |11\rangle$ and then output can be given by

$$\begin{aligned} O = {} & w_1 \phi_{00}(a_1, a_2) + w_2 \phi_{01}(a_1, a_2) + w_3 \phi_{10}(a_1, a_2) \\ & + w_4 \phi_{11}(a_1, a_2) \end{aligned} \tag{4}$$

where a_1, a_2 are the qubit of input, $w_k = \left(w_{k1}, w_{k2}, \ldots, w_{kj}\right)$ indicates a vector.

3.1.4 Model with the Orthogonal State

If the states are orthogonal, their scalar product vanishes, for example $|00\rangle$ and $|01\rangle$ is orthogonal. With the quantum computation, the output of quantum M-P model can be written as [20]

$$O_k = \sum_j w_{kj} |a_1, a_2, \ldots, a_n\rangle, \quad j = 1, 2, \ldots, 2^n$$

i.e. $O_k = w_{k1}|0, 0, \ldots, 0\rangle + w_{k2}|0, 0, \ldots, 1\rangle + \cdots + w_{k2^n}$ $|1, 1, \ldots, 1\rangle$

3.1.5 Model with the Non-orthogonal State

If the states are non-orthogonal, the input quantum state is $\alpha|0\rangle + \beta|1\rangle$ which is non-orthogonal with the basic state $|0\rangle$ or $|1\rangle$. Then the input and output relation is given by [20]

$$O_{km} = \sum_j w_{kj} \phi_j \cdot \phi_m, \quad j = 1, 2, \ldots, 2^n;\ m = 1, 2, \ldots, 2^n$$

where ϕ_j and ϕ_m are two states and $\phi_j \cdot \phi_m$ is the scalar product of these states.

3.1.6 Weight Learning Algorithm

A weight updating algorithm for the network is discussed by Zhou and Ding [20] as below

a. Initialize a weight matrix w^0.
b. Given a set of quantum examples, that is the pairs of the input–output $(|\phi\rangle, |O\rangle)$.
c. Calculate the output using $|\gamma\rangle = w^c|\phi\rangle$, where c denotes the number of iterations with initial value $c = 0$.
d. Update weight with the formula $w_{kj}^{c+1} = w_{kj}^c + \tau\left(|O\rangle_k - |\gamma\rangle_k\right)|\phi\rangle_j$, where w_{kj} are matrix entries indexed by the row k and column j; τ is the learning constant.
e. Repeat the process of (c) and (d) step up to the acceptable errors.

The weight of quantum M-P neural network can be stored in a square matrix $|w\rangle$. Suppose $|u\rangle$ is the output and input is $|v\rangle$. The input output relation can be defined [20] as

$$|u\rangle = w|v\rangle$$

Silva et al. [21] have shown that the learning algorithm that has been proposed by Zhou and Ding [20] did not follow an unitary evolution. For this they [21] have given an example that is if $w(0) = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$, the input $|v\rangle =$

$\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ and the desired output $|d\rangle = \frac{1}{\sqrt{2}}(|0\rangle + 1)$, then using the weight updating formula

$$w_{ij}(t+1) = w_{ij}(t) + \eta\left(|d\rangle_i - |u\rangle_i\right)|v\rangle_i$$

the next iteration can be calculated as

$$w(1) = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}$$

which did not follow the unitary evolution.

3.2 Quantum Competitive Neural Network (QCNN)

Quantum competitive neural network (QCNN) has been proposed by Zhou [22] combining classical competitive neural network (CCNN) with quantum concept. Zhou [22] has given two important concepts, first one is that of the basic concept of QCNN with the practical model and secondly he has pointed out the storage capacity which exponentially increases as compared to its classical counterpart. The memory capacity of the QCNN has been increased by an exponential factor of 2^m as compared to the CCNN, where m is the number of quantum bit. The most important thing to remember about QCNN is that, it does not have weights, because of which it neither required any learning algorithm nor weight updating which makes the network to train quickly. QCNN has a number of applications in the area of pattern recognition, image recognition and image clustering.

3.2.1 Classical Competitive Neural Network

The detail architecture of the CCNN is depicted in Fig. 1. The CCNN consists of two layers. The first one is the input matching subnetwork L1 which computes the Euclidean distance between input and storage pattern and the second layer is known as competitive subnetwork L2 which is used to search for the most preferable storage pattern to the input pattern using the rule of minimum distance.

Let the input pattern is $\hat{Y}(y_1, y_2, \ldots, y_t)$, the learning is narrated as given below

1. Vector normalization

   In the first instance, the input pattern $\hat{Y}$ as well as the weight vector $\hat{W}_l, l = 1, 2, \ldots, s$ of the network are normalized as follows

$$\hat{Y} = \frac{Y}{\|Y\|} = \left[\frac{y_1}{\sqrt{\sum_{k=1}^{t} y_k^2}}, \ldots\ldots\ldots, \frac{y_t}{\sqrt{\sum_{k=1}^{t} y_k^2}}\right] \quad (5)$$

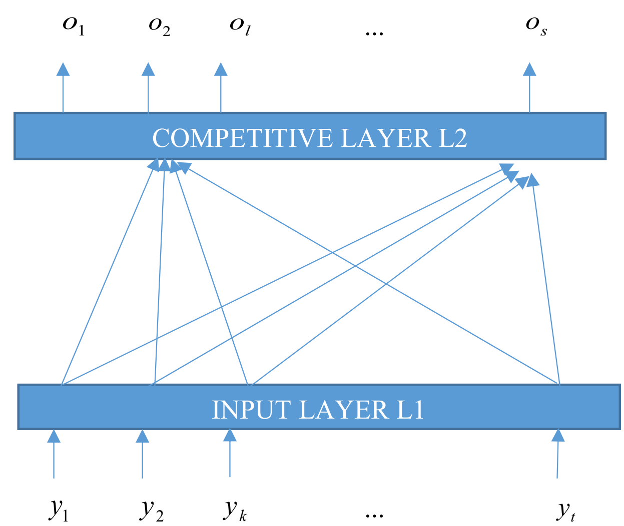


Fig. 1 The structure of classical competitive neural network [22]

$$\hat{W}_l = \frac{W_l}{\|W_l\|} = \left[\frac{w_{1l}}{\sqrt{\sum_{k=1}^{t} w_{kl}^2}}, \ldots\ldots\ldots, \frac{w_{sl}}{\sqrt{\sum_{k=1}^{t} w_{kl}^2}}\right] \quad (6)$$

2. Comparing comparability and after that finding the wining neuron

   When a pattern $\hat{Y}$ is given to the network, there is a comparison between all the weight vectors $\hat{W}_l$ of the competitive layer and $\hat{Y}$, then the corresponding neuron of weight vector which is almost similar to the pattern $\hat{Y}$ is chosen to be the winning neuron. The determinant principle [22] is given on the basis of cosine pattern method as $\cos\phi = \frac{\hat{W}_l \cdot \hat{Y}^T}{\|\hat{W}_l\| \cdot \|\hat{Y}\|}$. Two patterns are more similar, when the angle ϕ reduces and the value of cosine becomes larger.

3.2.2 QCNN

Like CCNN, it has also two layers namely input and competitive layers. Patterns that required to be identified by the network, stored in input layer and winning pattern is chosen by the competitive layer. Now the next thing is to know about storing of quantum patterns by input layer and their competition with each other is described as below.

3.2.2.1 Quantum Pattern Storage Quantum pattern storage is not similar to that of classical pattern storage and the working process is more complicated than the classical one, since every quantum operator have to be quantum unitary operator [22]. Input layer uses three registers: a first register $|p\rangle$ consists of n qubits feeds the pattern p^i to be stored, second register $|m\rangle$ consists of n qubits used for holding the memory, and third register is the control register $|c\rangle$ prepared in the state $|01\rangle$. A more detail explanation with an algorithm and application example is given in Zhou [22].

3.2.2.2 Quantum Pattern Competition In the process of pattern competition a copy of the memory $|m\rangle$ is formed which is used in the retrieval algorithm. While one is given a binary input i, that may be a corrupted version of the patterns in the memory, the network works from the L1 layer that has been learned well to the competitive layer and after pattern competition the winning pattern will be outputted. The competition between two patterns is completed by evaluating their Euclidean distance, and there is a criterion for choosing the winning pattern which will be decided if the pattern's Euclidean distance with input pattern is small. The competitive layer consists of three registers. The first one holds the input pattern, second one holds the memory and third one is a control register.

A quantum competition neural network (QCNN) has been presented by Zhong and Yuan [23]. Because of the presence of the pseudo states, QCNN is capable of quantum memory. They [23] have presented a competitive algorithm based on quantum concept and an application problem related to pattern recognition. With simulations, they [23] concluded that the presented algorithm is far superior to its classical counterpart.

3.3 Quantum-Inspired Neural Network (QINN)

In 1995, Menneer and Narayanan [5] proposed a hypothetical neural network model known as quantum inspired neural network. Introducing quantum rotation gates to the back propagation network, a novel QINN model has been proposed. Quantum Inspired Neural Network has been discussed by different authors [18, 24–27].

3.3.1 Quantum-Inspired Neuron Model (QINM)

In QINM, the weights are given by qubits $|\phi_i\rangle = a_i|0\rangle + b_i|1\rangle = [a_i, b_i]^T$. Let $I = (i_1, i_2, \ldots, i_n)^T$ is the real input vector, y is the output, the quantum weight is given by $[a_i \quad b_i]^T$, then the input–output relation of quantum inspired neuron is given by [24]

$$y = f(IR|\phi\rangle) = B \cdot \sum_{i=1}^{n} i_i R_i |\phi_i\rangle \tag{7}$$

where $\cdot$ is the inner product operator, $f(I) = B \cdot I, B = [1, 1]^T$, R_i is a quantum rotation gate to update the phase of qubit $|\phi_i\rangle$. The detail architecture of QINM is given in below Fig. 2.

3.3.2 Quantum Inspired Neural Network (QINN) model

QINN has three layers namely input, hidden, and output layer same as the traditional ANN. In this model the input, output and weights may be qubits. The QINN model which includes only quantum neurons called as normalization QINN where as if it includes both quantum neurons and classical neurons is called as the hybrid QINN. Shang [24] described about the hybrid QINN and the input–output relation is given as [24]

$$\begin{aligned} h_j &= B_j \cdot \sum_{i=1}^{n} x_i R_{ij} |\phi_{ij}\rangle \\ y_k &= s\left(\sum_{j=1}^{p} w_{jk} h_j\right) \end{aligned} \tag{8}$$

where $i = 1, 2, \ldots, n$; $j = 1, 2, \ldots, p$; $k = 1, 2, \ldots, m$; w_{jk} is the weight between the jth neuron in the hidden layer and the kth neuron in output layer, s is the sigmoid function. The error function of the output is given by

$$E = \frac{1}{2}\sum_{k=1}^{m} (\hat{y}_k - y_k)^2 \tag{9}$$

where $\hat{y}_k$ is the desired output.

A novel model and algorithm of quantum inspired neural network with sequence input (QINNSI) based on controlled rotation gates has been proposed by Li and Xiao [18]. Example to predict breast cancer using the QINNSI model and ANN have been examined in the paper [18]. Shang [24] proposed a QINN with both quantum weights as well as real weights. A novel hybrid quantum inspired neural networks (QINNs) model has been proposed by Li et al. [25] with sequence inputs based on the controlled hadamard gates. The model [25] has a three layer architecture. A learning algorithm has been given based on the concept of quantum computing. This proposed model [25] is superior as compared to the Back Propagation Neural Networks.

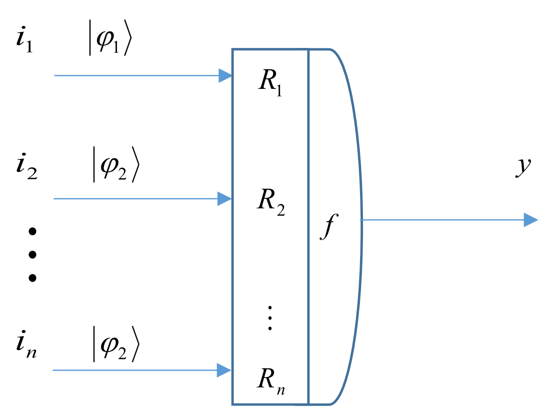


Fig. 2 The quantum-inspired neuron model [24]

A CNOT gated QNNs have been designed by Cao and Li [26] combining the basics of quantum rotation gates and CNOT gate. They have not only designed the network but also proposed an algorithm for the model. Moreover two application problems are given, one of them is wine recognition and another one is modeling of acrylamide homogeneous polymerization process. Li and Li [27] have generalized the ability of ANN with the help of the characteristics of quantum rotation gates and CNOT gate. They [27] proposed a model for quantum inspired neuron with sequence input.

3.4 Quantum Dot Neural Network

In 1996, Quantum Dot Neural Network has been proposed by Behrman et al. [28]. Quantum dots are very small particle in between 2 and 10 nm in diameter which is equivalent to around 50 atoms. Basically quantum dot consists of a single electron trapped inside a cage of atoms.

Basically in almost every ANN model, the neurons get inputs from other processors through the weighted connections and compute an output which is passed on to other neurons. The computed output y_i of the ith neuron is performed on the signals $\{y_j\}$ from the other neurons in the network and is given as [28]

$$y_i = \sum_j w_{ij} f_j(y_j) \tag{10}$$

where w_{ij} denotes the weight between the neuron j to the neuron i, and f_j is an activation function for neuron j. Accordingly the expression for the time evolution of the quantum mechanical state of a system is written as [28]

$$\begin{aligned} \left|\Omega(y_f, S)\right\rangle &= G(y_f, S; y_0, 0)|\Omega(y_0, 0)\rangle \\ &= \int_{(y_0,0)}^{(y_f,S)} D[y(t)] \exp\left(\frac{i}{h}\int_0^S d\tau[\frac{1}{2}m\dot{y}^2 - V(y)]\right)|\Omega(y_0,0)\rangle \\ &= \lim_{N\to\infty} \int_{(y_0,0)}^{(y_{N+1}y_f,S)} dy_1 \cdots dy_N \left(\frac{m}{2\pi in\Delta t}\right)^{(N+1)/2} \\ &\quad \times \exp\left(\frac{i\Delta t}{n}\sum_{j=0}^{N}\left[\frac{m}{2}\left(\frac{y_{j+1}-y_j}{\Delta t}\right)^2 - V(y_j)\right]|\Omega(y_0,0)\rangle\right) \end{aligned} \tag{11}$$

In above description as mentioned in the paper [28], $|\Omega(y_0, 0)\rangle$ denotes the input state of the quantum system and $\left|\Omega(y_f, S)\right\rangle$ is the output state, the state of the system at time $t = S$. G is the Green's function which propagates the system forward in time, from initial position y_0 at time $t = 0$ to final position y_f at time $t = S$. Second line of the Eq. (11) expresses G in the Feynman path integral formulation of quantum mechanics.

Quantum dot model system consists of a quantum dot molecule where five dots arranged as the pips on a playing card. The dots are closely connected to each other so that tunneling is possible between any two neighboring dot molecules. Again equation (11) can be rewrite as

$$\begin{aligned} &|\Omega(\sigma_z(N,\Delta t),S)\rangle \\ &= \sum_{\sigma_z(j\Delta t)} \exp\left(\frac{i}{h}\sum_j [K\sigma_x(j\Delta t) + \in (j\Delta t)\sigma_z(j\Delta t)]\right) I[\sigma_z(t)]| \\ &\quad \Omega(\sigma_z(0)), 0\rangle \end{aligned} \tag{12}$$

where $y(t)$, has been written as a finite set of sums over states of the polarization,σ_z, at each time slice $j\Delta t$ at each time slice, the polarization can be either $+1$ or -1 [28] (Fig. 3).

3.5 Quantum Cellular Neural Networks

The concept of quantum cellular neural network (QCNN) has been put forth by Toth et al. [29] in 1996. They [29] used coupled quantum dot cells in an architecture to build an analog Cellular Neural Network.

3.5.1 Cellular Neural Network (CNN)

The concept of CNN has been given by Chua and Yang [30] which is consisting of two or three dimensional array of similar cells. Every cell, indexed by i, has dynamical state variables y_i, external inputs u_i and internal constant cell data v_i. A neighboring synaptic input I_i^T is always going to influence each cell which in general depends on cell states values as well as cell inputs within a sphere S_i centered on cell i. Then the cellular neural network can be defined using synaptic laws as [29]

$$I_i^T = \sum_\lambda A_i^\lambda x_{i+\lambda} + \sum_\lambda B_i^\lambda f(x_i, x_{i+\lambda}) + \sum_\lambda C_i^\lambda u_{i+\lambda} \tag{13}$$

The cell dynamics can be found using the CNN state equation as [29]

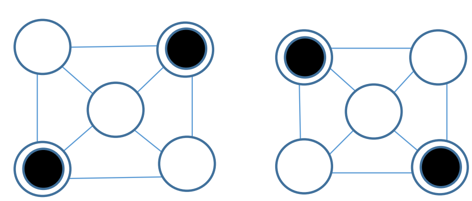


Fig. 3 The circles represents the quantum dots arranged in a manner that tunneling between the neighboring quantum dots is possible in the molecule [28]

$$\frac{\partial}{\partial t}x_i = -g\left(x_i, v_i, u_i, I_i^T\right) \tag{14}$$

If there is no external inputs, then the cellular neural network is known as autonomous.

3.5.2 Quantum Model of Cell Array

Figure 4 shows a model of an array of interacting quantum cells where every cell holds four quantum dots with two extra electrons.

The cell polarization C can be defined using the expectation values σ_i of the charge on each dot.

$$C \equiv \frac{(\sigma_1 + \sigma_3) - (\sigma_2 + \sigma_4)}{\sigma_1 + \sigma_2 + \sigma_3 + \sigma_4} \tag{15}$$

The polarization is always in between − 1 and + 1.

Fortuna and Porto [31] presented a paper on the chaotic phenomenon of QCNN. They [31] have given an example on how to evaluate the polarization between two adjoining cells and shown their chaotic behavior. Sen et al. [32] described the chaotic phenomenon in QCNN using coupled Josephson circuits. Using the coupled Josephson circuits, the paper describes the nonlinear dynamical behavior of a QCNN.

3.6 Qubit Neural Network

In 2000, Matsui et al. [33] proposed a qubit neuron model which shows quantum learning abilities. This qubit neuron model has a high efficiency in solving problems like data compression. Kouda et al. [14] proposed qubit neural network where the interaction between the states of the neurons with other neurons are based on the laws of quantum mechanics. The qubit neuron model leads to a new multi-layered quantum feed forward neural network which implements the quantum phenomenon given by Kouda et al. [34]. In recent times some works and research have been carried out in the area of quantum neural networks so as to improve the computational power of neural networks. It is a multi-layered neural network consisting of quantum bit neurons.

3.6.1 Qubit Neuron Model

We can rewrite the neuron state in (1) as

$$f(\theta) = \cos\theta + i\,\sin\theta = e^{i\theta} \tag{16}$$

Figure 5 shows a general model of quantum neuron. The neuron state w that gets inputs from L, other neurons are given as [14]

$$v = \sum_{l}^{L} f(\theta_l).x_l - f(\lambda) = \sum_{l}^{L} f(\theta_l).f(y_l) - f(\lambda) \tag{17}$$

$$y = \frac{\pi}{2}h(\phi) - \arg(v) \tag{18}$$

$$w = f(y) \tag{19}$$

where v is the internal state of a qubit neuron w. x_l is the qubit neuron state of the lth neuron, f is the same function as given in (16), y and y_l are quantum phases of w and x_l respectively. Here h is the sigmoid function defined as

$$h(\phi) = \frac{1}{1 + e^{-\phi}}$$

This neuron model includes two kinds of parameters namely phase parameters in the form of weight connection θ_l and threshold λ, and the reversal parameter ϕ.

Qubit neural network has a three layer architecture where first one is the input layer $\{X_i\}(i = 1, 2, \ldots, I)$, second one is the hidden layer $\{H_j\}(j = 1, 2, \ldots, J)$ and the last one is the output layer $\{Y_k\}(k = 1, 2, \ldots, K)$ where I, J and K are the number of neurons present in the input, hidden and output layer respectively.

When input data (denoted by $input_i$) is given into the network, the input layer consisting of the neurons in $\{X_i\}$ which converts input values from $[0, 1]$ into quantum states with phase values in the range $\left[0, \frac{\pi}{2}\right]$. The output z_i^X of the input neuron X_i becomes the output to the hidden layer as

$$z_i^X = f\left(\frac{\pi}{2}.input_i\right) \tag{20}$$

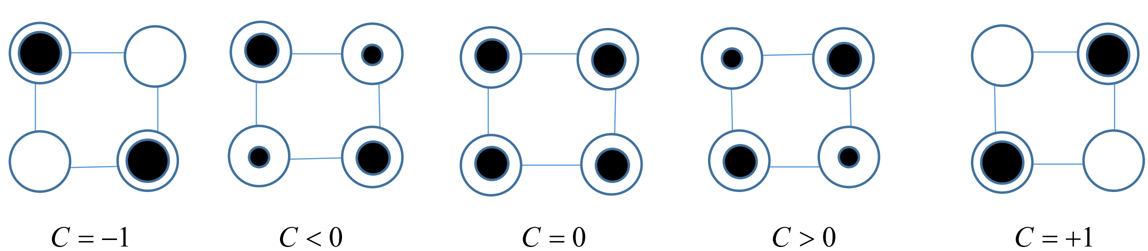


Fig. 4 Cell polarization [29]

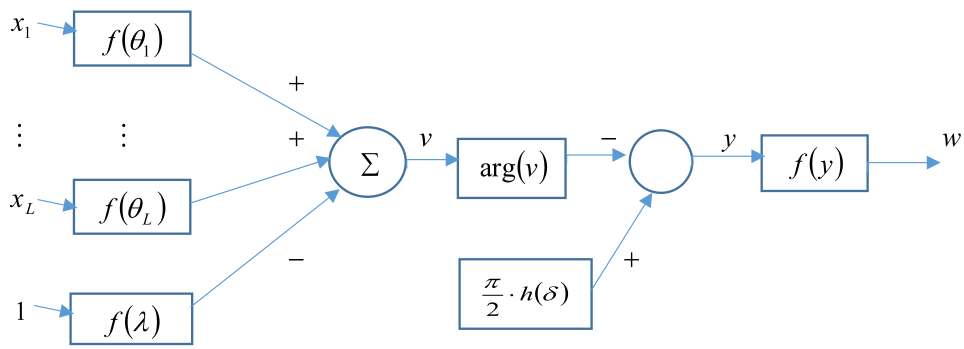


Fig. 5 Qubit neuron model [14]

We obtain the output to the network, denoted by $output_k$, by calculating the probability for the basic state $|1\rangle$ in the nth neuron state z_k^Y in the output layer [14]

$$output_k = \left|\mathrm{Im}\left(z_k^Y\right)\right|^2 \tag{21}$$

3.6.2 Quantum Modified Back Propagation Learning

Authors [14] defined modified version of back propagation algorithm with quantum concept in order to incorporate learning process in qubit neural network. Back propagation algorithm is employed as the learning rule. This is described by the equations [14]

$$\begin{aligned} \theta_l^{new} &= \theta_l^{old} - \eta \frac{\partial E_{total}}{\partial \theta_l} \\ \lambda^{new} &= \lambda^{old} - \eta \frac{\partial E_{total}}{\partial \lambda} \\ \delta^{new} &= \delta^{old} - \eta \frac{\partial E_{total}}{\partial \delta} \end{aligned} \tag{22}$$

where η is the learning parameter. E_{total} is the sum squared error given by [14]

$$E_{total} = \frac{1}{2} \sum_p^P \sum_n^N \left(t_{p,n} - output_{p,n}\right)^2 \tag{23}$$

This quantity is the cost function to be minimized. Here, P is the number of learning patterns, $t_{p,n}$ is the target signal for the nth neuron and $output_{p,n}$ means $output_n$ of the network when it learns the pth pattern.

The efficiency of qubit neuron model has been discussed by Kouda et al. [14]. With the simulation results, they [14] have concluded that the qubit model has been very much efficient in learning problems as compared to the classical counterpart and further the learning ability of the qubit neural network using 4-bit parity check problem has been explored. Next, the learning ability of qubit neural network has been examined by Kouda et al. [35]. The simulation results show that it is efficient in solving problems like data compression. With reference to Kouda et al. [14], they [35] confirmed the results in that paper [14] and further investigated to validate the efficiency of the qubit neuron model through 4 bit as well as 6 bit parity check problems. From the simulation results they [35] concluded that the better performance of the qubit model is because of the superposition of states. Kouda et al. [36] examined the effectiveness of qubit neural network for a control application problem of an inverted pendulum on a cart.

With some basics of quantum computation, a multilayer qubit neural network with qubit neurons has been proposed by Matsui et al. [37]. Moreover they [37] have included two more applications problems for example image compression and pattern recognition to examine the performance of qubit neural network.

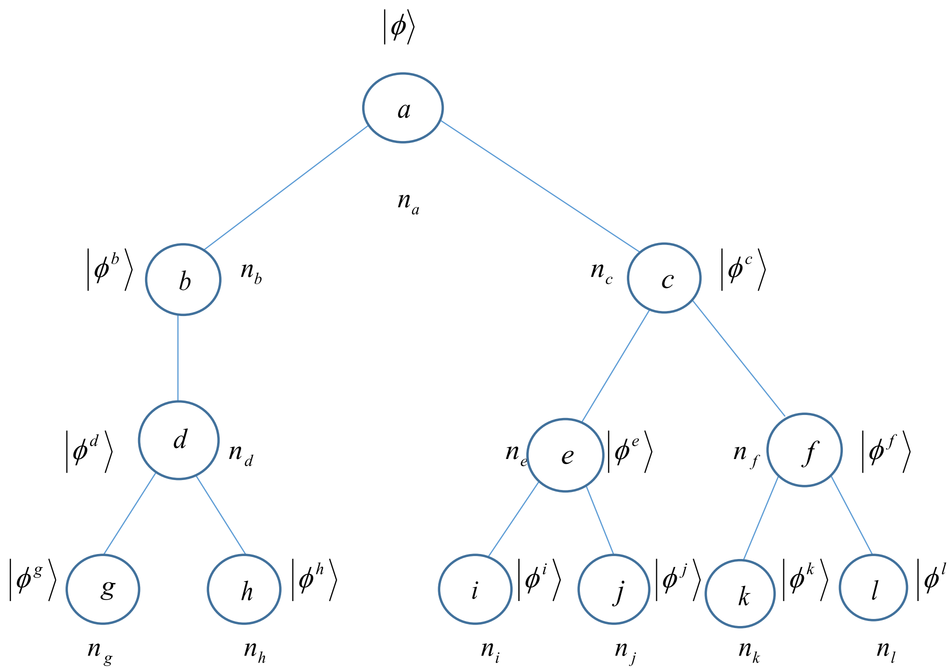
3.7 Quantum Associative Neural Network

In 2000, a quantum associative memory model has been presented by Ventura and Martinez [9]. They [9] produced a quantum associative memory after combining two quantum computational algorithms as a result the memory capacity increased exponentially as compared to its traditional counterpart. Perus [38] proposed a model known as quantum associative network. Overcoming the classical associative neural network, Zhou et al. [39] proposed a model of quantum associative neural network (QANN) (combing the classical one and the quantum associative memory) with nonlinear search algorithm. The working of QANN has mainly two parts, first part is storing patterns and second one is recalling pattern similar to its classical counterpart. Storing of patterns need different quantum operations and quantum gates.

3.7.1 Procedure of Storing Patterns

For storing patterns, Zhou et al. [39] defined a quantum binary decision diagram using the concept of binary tree. The quantum binary decision diagram built using the concept of superposition of states which is shown in the Fig. 6.

Fig. 6 Quantum binary decision diagram [39]



In Fig. 6, $a, b, c, \ldots, l$ represent the quantum gates, $|\phi\rangle, |\phi^b\rangle, \ldots, |\phi^l\rangle$ are quantum states. Out of them $|\phi^g\rangle, |\phi^h\rangle, \ldots, |\phi^l\rangle$ are the desired quantum states, and $n_a, n_b, \ldots, n_l$ are nodes where n_a is the root node.

Recalling pattern is quite alike to the classical counterpart.

4 Some Other Models and Learning Algorithm Related to QNN

A new network model namely self-organizing quantum neural network in short SOQNN has been proposed by Zhou et al. [40] after combining the concept of quantum computation with classical ANN. In the proposed model, the input pattern, output pattern and weights all are qubit. Pattern classification could be performed using SOQNN using a process called quantum competitive. Shafee [41] examined a quantum model with quantum bits and quantum operators. Shafee [41] discussed few beautiful results of QNN using quantum gates such as CNOT and AND gates.

After summarizing the basic model and learning algorithm of quantum neuron, Xuan [42] proposed a novel model named as quantum adaptive resonance theory neural network (QARTNN). QARTNN combines quantum computing with adaptive resonance theory and further a learning algorithm for the network has been proposed by Xuan [42]. This QARTNN model is very much useful and we can apply it in pattern recognition. Quantum neural like network has been proposed by Perus et al. [43]. They [43] wrote a concise review of current research of quantum Hopfield-like information processing. Sagheer and Zidan [44] proposed an autonomous quantum perceptron neural network (AQPNN) where the input and output are quantum whereas weights are taken to be real. Authors have given a learning algorithm for AQPNN and tested the computational power of this algorithm. After examining the algorithm, they [44] concluded that it would be a good computational alternate compared to classical counterpart.

A new and useful single hidden layer feedforward neural network model based on the principle of quantum computing has been proposed by Liu et al. [45]. They [45] have used Grover's algorithm for the training of network. To examine the performance of the proposed model some simulation results have been included. Behrman and Stack [46] uses two basic and useful concepts of quantum computing that is interference and entanglement which enhance the computing power of a QNN. They [46] used a two qubit system as QNN and trained the network for computing and giving the output of its own relative phase.

QNN models have also been discussed by different authors [47–51]. Purushothaman and Karayiannis [47] found that QNNs were very useful in solving pattern classification problems. Experimentally, authors [47] have shown that QNN can recognize structures in data, a property absent in the traditional feedforward neural network with sigmoidal hidden units. Zhong and Yuan [48] presented a QNN model using quantum neuron model of wave function. They [48] analyzed the learning ability of QNN and found that the learning ability of quantum neuron is very high. Few quantum neural computing problems have also been examined. A supervised learning algorithm for QNN has been presented and examined by Silva et al. [49] using quantum neuron node. They have introduced a quantum weightless neural network (QWNN). Ventura and Martinez [50] discussed quantum theory and on the basis of the quantum theory presented a model for quantum neuron. Quantum computation solves different computational tasks and few results shows that it is exponentially faster as compared to the classical computation. Similar to classical ANNs (CANNs) Cao et al. [51] introduced a novel model namely quantum artificial neural network (QANN). They have included a universal approximation theorem (UAT) to prove the capability of approximation of a QANN. UAT theorem [51] concludes that every continuous function which maps n quantum states as a non-normalized quantum state can be uniformly approximated using QANN.

For traditional neural network without having weights, Silva et al. [52] investigated the network using a quantum algorithm namely single shot learning algorithm (SSLA) which requires only one epoch. The proposed algorithm [52] generates a superposition of all probable neural network configuration for a given network topology.

5 Applications of Quantum Neural Network

Quantum neural network has a number of applications in real life problems, few of them are listed below

1. Breast Cancer Prediction [18]
2. Image Compression [37, 53, 54]
3. Pattern Recognition [37, 55–58]
4. Function Approximation [57, 58]
5. Time series prediction [59, 60]
6. Electronic remaining useful life (RUL) prediction [60]
7. Forecasting series problems [61]
8. Fault diagnosis of transformer [62]
9. Portfolio Selection [63]
10. Vehicle Classification [64]
11. Automatic detection of premature ventricular contraction [65]
12. To recognize the handwritten numerals [66]
13. Curve fitting [67]
14. To recognize Electroencephalograph signals [68]
15. Temperature Control [69]
16. Other Applications [70–74]

QNN has applications in different areas of real world. We have already discussed about the application for the breast cancer prediction [18] and image processing [37]. Kouda et al. [53] studied the performance of QNN of large size in image compression problems. Simulation results in the paper [53] proved that QNN has high processing abilities in image compression as compared to the classical neural network. A novel three layer quantum back propagation network model for image compression based on the concept of QNN has been presented by Li and Li [54]. Due to the slower convergence of the initial weight, authors [54] used the concept of genetic algorithm in optimizing the initial weight. Moreover authors [54] introduced a concept called clamping which helps in the better performance of genetic algorithm.

Using QNN, Li and Li [55] introduced a novel approach to recognize facial expression and given an example of pattern recognition. For the training and testing purpose of the QNN, the Japanese female facial expression database has been taken into consideration. Xu et al. [56] proposed a way for face recognition using QNN. It involves three steps that is image preprocessing, feature extraction and face recognition. For the training and testing purpose of the QNN, authors [56] used ORL faces digital database. A QNN model has been presented by Mu et al. [57] where the input and output of the network are real vectors, but on the other hand the connected weights are qubits. With the help of two examples that is pattern recognition and function approximation, authors [57] have shown the convergence rate of the QNN model. Based on the concepts of quantum rotation gates and quantum CNOT gate, Li and Zhao [58] presented QNNs model with a learning algorithm based on the concept of gradient descent algorithm. Authors [58] have discussed two important application problems of pattern recognition and function approximation to conclude that the proposed QNN is more efficient as compare to the traditional Back Propagation networks with respect to the number of iterations and convergence rate.

Araujo et al. [59] described a new technique namely quantum inspired hybrid which helped in overcoming the random work dilemma for financial time series prediction. This methodology consists of a hybrid model which combines a Quantum Multilayer Perceptron with a quantum-inspired evolutionary algorithm. Cui et al. [60] proposed a new model namely Complex Rotation Quantum Dynamic Neural Networks (CRQDNN) on the basis of complex quantum neuron. It has a three layer architecture where one of them is a quantum layer which contains the Quantum Complex Rotation Gate and controlled NOT gate while the hidden layer comprises of the IIR filters as the memory unit which is used to handle time series input. In comparison with real QNN, complex QNN model increases the level of quantum entanglement. The networks model introduced by Cui et al. [60] is useful for time series prediction. Authors [60] discussed about two applications that is chaotic time series prediction and RUL prediction.

Azevedo and Ferreira [61] described a QNN using the concept of qubit neuron model which is used in time series forecasting problems. They [61] have predicted two forecasting series problems, one is chaotic series and another one is stock market series. Ren et al. [62] studied the classified effect of QNN on the basis of Rough set which was applicable in the fault diagnosis of transformer. Authors [62] concluded that QNNs based on the concept of rough set theory is very much useful in handling uncertain data. Mahajan [63] applied the QNN model to solve the financial engineering problem of portfolio selection. A new learning model has been proposed by Yu and Ma [64] with qubit neuron which they applied in vehicle classification. For this classification problem, authors [64] constructed a three layered QNN and taken four kinds of vehicles such as saloon car, micro bus, truck and motor coach. Zhou [65] experimentally shown usefulness of QNN in automatic detection of premature ventricular contraction based on ECG recordings. After combining the concepts of fuzzy and ANN, author [65] gave the concept of quantum neural network with the detailed training algorithm. QNN gives better diagnosis accuracy in this case. But the proposed QNN takes bit more time to train and needs extra storage space. Zhou et al. [66] presented a QNN which unites the advantages of neural network and principles of fuzzy theory. Zhou et al. [66] discussed the QNN's application to recognize the handwritten numerals and experimentally found that QNN is able to recognize confusing digit pairs. Zhang et al. [67] given an interesting application of quantum back propagation neural network (QBPNN) in curve fitting. From the simulation results authors [67] concluded that QBPNN learns quickly and better fitting ability as compared to the classical back propagation neural network (CBPNN).

Aljazeera et al. [68] described how to recognize Electroencephalograph signals using QNN. To recognize the signals, authors [68] used three concepts for the feature extraction which are independent component analysis, wavelet transform and Fourier transform. A Takagi–Sugeno-Kang-type quantum neural fuzzy network has been introduced by Lin et al. [69] for controlling the temperature where the network has a five layered architecture. A learning algorithm has been proposed which combines the self-clustering algorithm and the BP algorithm.

Combining the concept of quantum mechanics and Schrodinger wave equation Gandhi and McGinnity [70] proposed a new technique for filtering the electromyogram (EMG) signals. The model discussed by Gandhi and McGinnity [70] is termed as Recurrent Quantum Neural Network (RQNN). Authors [70] included an unsupervised learning rule which helps the RQNN to trap the statistical behavior of the input signal and estimate the EMG signal embedded in noise. Further Behera and Sundaram [71] presented a paper which discussed about stochastic filtering using RQNN. A QNN model based on the concept of nonlinear Schrodinger wave equation has been introduced by Behera et al. [72] which explains the eye movements while tracking moving objects. Ivancevic and Reid [73] introduced a novel method based on a coupled QNNs for modelling dynamics of confined crowds driven by Entropic Stochastic Resonance (ESR). A multi-layer QNN has been examined by Takahashi et al. [74] and explored its application in control system.

6 Conclusion

Recent developments in QNN by various researchers are discussed in detail which will certainly help the researchers to work on this new vista. Starting with basics of quantum computation and neural network, it slowly introduced the basics of QNN. Various models as proposed by previous authors are addressed one by one to understand the developments. The powerfulness of the quantum computing in neural network will certainly lead artificial intelligence (AI) to achieve a new height. It may be noted that quantum computing plays an important and vital role for improving the computational efficiency of the neural networks. As such, traditional ANN has become more powerful and efficient over the years after the addition of quantum computing. Different simulation results cited in various papers mentioned above show that quantum neural network is very useful in solving important and challenging problems. Although QNN is being applied to different problems but it is still in infant stage and the same need to be investigated further to take advantage of its efficiency and powerfulness in other subject areas.

Compliance with Ethical Standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Shor PW (1994) Algorithms for quantum computation: discrete logarithms and factoring. In: Foundations of computer science, 1994 proceedings. 35th annual symposium on IEEE, pp 124–134
2. Grover LK (1996) A fast quantum mechanical algorithm for database search. In: Proceedings of the twenty-eighth annual ACM symposium on theory of computing, pp 212–219
3. Benioff P (1982) Quantum mechanical Hamiltonian models of Turing machines. J Stat Phys 29(3):515–546
4. Kak S (1995) On quantum neural computing. Inf Sci 83:143–163
5. Menneer T, Narayanan A (1995) Quantum-inspired neural networks, Department of Computer Science, University of Exeter, Exeter, United Kingdom, Technical Report, p 329

6. Perus M (1996) Neuro-quantum parallelism in brain-mind and computer. Informatica 20:173–183
7. Vlasov A (1997) Quantum computations and images recognition. arXiv preprint arXiv:quant-ph/9703010
8. Menneer T (1998) Quantum artificial neural networks. PhD thesis, University of Exeter
9. Ventura D, Martinez T (2000) Quantum associative memory. Inf Sci 124(1–4):273–296
10. Behrman EC, Nash LR, Steck JE, Chandrashekar VG, Skinner SR (2000) Simulations of quantum neural networks. Inf Sci 128(3–4):257–269
11. Narayanan A, Menneer T (2000) Quantum artificial neural network architectures and components. Inf Sci 128(3–4):231–255
12. Altaisky MV (2001) Quantum neural network. arXiv preprint arXiv:quant-ph/0107012
13. Gupta S, Zia RKP (2001) Quantum neural networks. J Comput Syst Sci 63(3):355–383
14. Kouda N, Matsui N, Nishimura H, Peper F (2003) Qubit neural network and its efficiency. In: International conference on knowledge-based and intelligent information and engineering systems, pp 304–310
15. Schuld M, Sinayskiy I, Petruccione F (2014) The quest for a quantum neural network. Quantum Inf Process 13(11):2567–2586
16. Ezhov AA, Ventura D (2000) Quantum neural networks. In: Kasabov N (ed) Future directions for intelligent systems and information sciences. Physica, Heidelberg, pp 213–235
17. Ricks B, Ventura D (2004) Training a quantum neural network. In: Advances in neural information processing systems, pp 1019–1026
18. Li P, Xiao H (2014) Model and algorithm of quantum-inspired neural network with sequence input based on controlled rotation gates. Appl Intell 40:107
19. Imre S, Balazs F (2013) Quantum computing and communications: an engineering approach. Wiley, New York
20. Zhou R, Ding Q (2007) Quantum MP neural network. Int J Theor Phys 46:3209–3215
21. da Silva AJ, de Oliveira WR, Ludermir TB (2015) Comments on “quantum MP neural network”. Int J Theor Phys 54:1878–1881
22. Zhou R (2010) Quantum competitive neural network. Int J Theor Phys 49:110–119
23. Zhong Y, Yuan C (2012) Quantum competition network model based on quantum entanglement. J Comput 7(9):2312–2317
24. Shang F (2015) Quantum-inspired neural network with quantum weights and real weights. Open J Appl Sci 5:609–617
25. Li P, Xiao H, Shang F, Tong X, Li X, Cao M (2013) A hybrid quantum-inspired neural networks with sequence inputs. Neurocomputing 117:81–90
26. Cao M, Li P (2014) Quantum-inspired neural networks with applications. Int J Comput Inf Technol 3(1):83–92
27. Li Z, Li P (2015) Quantum-inspired neural network with sequence input. Open J Appl Sci 5:259–269
28. Behrman EC, Niemel J, Steck JE, Skinner SR (1996) A quantum dot neural network. In: Proceedings of the 4th workshop on physics of computation, pp 22–24
29. Toth G, Lent CS, Tougaw PD, Brazhnik Y, Weng W, Porod W, Liu RW, Huang YF (1996) Quantum cellular neural networks. Superlattices Microstruct 20(4):473–478
30. Chua LO, Yang L (1988) Cellular neural networks: applications. IEEE Trans Circuits Syst 35(10):1273–1290
31. Fortuna L, Porto D (2002) Chaotic phenomena in quantum cellular neural networks. In: Cellular neural networks and their applications. Proceedings of 7th IEEE international workshop, pp 369–376
32. Sen W, Li C, Qin L, Gang W (2007) Chaotic phenomena in Josephson circuits coupled quantum cellular neural networks. Chin Phys 16(9):2631–2634
33. Matsui N, Takai M, Nishimura H (2000) A network model based on qubit-like neuron corresponding to quantum circuit. Electron Commun Jpn (Part III: Fundam Electron Sci) 83(10):67–73
34. Kouda N, Matsui N, Nishimura H (2004) A multilayered feed-forward network based on qubit neuron model. Syst Comp Jpn 35(13):43–51
35. Kouda N, Matsui N, Nishimura H, Peper F (2005) Qubit neural network and its learning efficiency. Neural Comput Appl 14(2):114–121
36. Kouda N, Matsui N, Nishimura H, Peper F (2005) An examination of qubit neural network in controlling an inverted pendulum. Neural Process Lett 22(3):277–290
37. Matsui N, Nishimura H, Isokawa T (2009) Qubit neural networks: its performance and applications. In: Nitta T (ed) Complex-valued neural networks: utilizing high-dimensional parameters, information science reference. IGI Global, Hershey, pp 325–351
38. Perus M (2000) Neural networks as a basis for quantum associative networks. Neural Netw World 10(6):1001–1013
39. Zhou R, Wang H, Wu Q, Shi Y (2012) Quantum associative neural network with nonlinear search algorithm. Int J Theor Phys 51(3):705–723
40. Zhou R, Zheng HY, Jiang N, Ding Q (2006) Self-organizing quantum neural network. In: Neural networks, 2006. IJCNN’06. International joint conference on IEEE, pp 1067–1072
41. Shafee F (2007) Neural networks with quantum gated nodes. Eng Appl Artif Intell 20(4):429–437
42. Xuan H (2011) Research on quantum adaptive resonance theory neural network. In: Electronic and mechanical engineering and information technology (EMEIT), international conference on IEEE, vol 8, pp 3885–3888
43. Perus M, Bischof H, Hadzibeganovic T (2005) A natural quantum neural-like network. NeuroQuantology 3(3):151–163
44. Sagheer A, Zidan M (2013) Autonomous quantum perceptron neural network. arXiv preprint arXiv:1312.4149
45. Liu CY, Chen C, Chang CT, Shih LM (2013) Single-hidden-layer feed-forward quantum neural network based on Grover learning. Neural Netw 45:144–150
46. Behrman EC, Steck JE (2013) A quantum neural network computes its own relative phase. In: Swarm intelligence (SIS), 2013 IEEE symposium, pp 119–124
47. Purushothaman G, Karayiannis NB (1997) Quantum neural networks (QNNs): inherently fuzzy feedforward neural networks. Neural Netw IEEE Trans Neural Netw 8(3):679–693
48. Zhong YH, Yuan CQ (2013) Analysis of quantum neural network learning ability. Appl Math & Inf Sci 7(2L):679–683
49. da Silva AJ, de Oliveira WR, Ludermir TB (2012) Classical and superposed learning for quantum weightless neural networks. Neurocomputing 75(1):52–60
50. Ventura D, Martinez T (1998) An artificial neuron with quantum mechanical properties. In: Artificial neural nets and genetic algorithms. Springer, Vienna, pp 482–485
51. Cao H, Cao F, Wang D (2015) Quantum artificial neural networks with applications. Inf Sci 290:1–6
52. da Silva AJ, de Oliveira WR, Ludermir TB (2014) Training a classical weightless neural network in a quantum computer. In: European symposium on artificial neural networks, computational intelligence and machine learning, pp 523–528
53. Kouda N, Matsui N, Nishimura H (2002) Image compression by layered quantum neural networks. Neural Process Lett 16(1):67–80
54. Li H, Li M (2010) A new method of image compression based on quantum neural network. In: Information science and management engineering (ISME), 2010 international conference of IEEE, vol 1, pp 567–570

55. Li P, Li J (2007) A facial expression recognition method based on quantum neural networks. In: Proceedings of international conference on intelligent systems and knowledge engineering, pp 74–78
56. Xu Y, Zhang X, Gai H (2011) Quantum neural networks for face recognition classifier. Procedia Eng 15:1319–1323
57. Mu D, Guan Z, Zhang H (2013) Learning algorithm and application of quantum neural networks with quantum weights. Int J Comput Theory Eng 5(5):788–792
58. Li J, Zhao S, Zhang H, Liu Y (2015) Neural networks based on quantum gated nodes. Int J Comput Inf Technol 4:583–589
59. Araújo RDA, de Oliveira AL, Soares SC (2010) A quantum-inspired hybrid methodology for financial time series prediction. In: The 2010 international joint conference on IEEE, Neural networks (IJCNN), pp 1–8
60. Cui Y, Shi J, Wang Z (2015) Complex rotation quantum dynamic neural networks (CRQDNN) using complex quantum neuron (CQN): applications to time series prediction. Neural Netw 71:11–26
61. Azevedo CR, Ferreira TA (2007) The application of qubit neural networks for time series forecasting with automatic phase adjustment mechanism. In: Proceedings of XXVII congress of the Brazilian computer science society (VI national meeting of artificial intelligence), pp 1112–1121
62. Ren X, Zhang F, Zheng L, Men X (2010) Application of quantum neural network based on rough set in transformer fault diagnosis. In: Power and energy engineering conference (APPEEC), 2010 Asia-Pacific IEEE, pp 1–4
63. Mahajan RP (2011) A quantum neural network approach for portfolio selection. Int J Comput Appl 29(4):47–54
64. Yu S, Ma N (2008) Quantum neural network and its application in vehicle classification. In: Natural computation, ICNC'08. Fourth international conference on IEEE, vol 2, pp 499–503
65. Zhou J (2003) Automatic detection of premature ventricular contraction using quantum neural networks. In: Bioinformatics and bioengineering, 2003. Proceedings. Third IEEE symposium, pp 169–173
66. Zhou J, Gan Q, Krzyżak A, Suen CY (1999) Recognition of handwritten numerals by quantum neural network with fuzzy features. Int J Doc Anal Recogn 2(1):30–36
67. Zhang Z, Liu Y, Xie J (2013) Study on application of quantum BP neural network to curve fitting. TELKOMNIKA Indones J Electr Eng 11(10):5636–5643
68. Aljazaery IA, Ali AA, Abdulridha HM (2011) Classification of Electroencephalograph (EEG) signals using quantum neural network. Signal Process Int J (SPIJ) 4(6):329–337
69. Lin CJ, Chen CH, Lee CY (2006) TSK-type quantum neural fuzzy network for temperature control. Int Math Forum 1(18):853–866
70. Gandhi VS, McGinnity TM (2013) Quantum neural network based surface EMG signal filtering for control of robotic hand. In: Neural networks (IJCNN), the 2013 international joint conference on IEEE, pp 1–7
71. Behera L, Sundaram B (2004) Stochastic filtering and speech enhancement using a recurrent quantum neural network. In: Intelligent sensing and information processing. Proceedings of international conference on IEEE, pp 165–170
72. Behera L, Kar I, Elitzur AC (2005) A recurrent quantum neural network model to describe eye tracking of moving targets. Found Phys Lett 18(4):357–370
73. Ivancevic VG, Reid DJ (2009) Dynamics of confined crowds modelled using entropic stochastic resonance and quantum neural networks. Int J Intell Def Support Syst 2(4):269–289
74. Takahashi K, Kurokawa M, Hashimoto M (2012) Controller application of a multi-layer quantum neural network with qubit neurons. J Adv Mech Des Syst Manuf 6(4):526–540