Volume 2, No. 3, May-June 2011

ISSN No. 0976-5697

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Implementation of Hadamard Gate Using Quantum Artificial Neural Network

Shaktikanta Nayak*
PhD Student,
Department of Management Studies,
Indian Institute of Technology,
Roorkee,India
shakti302@gmail.com

Sitakanta Nayak
PhD Student ,
Department of Management Studies,
Indian Institute of Technology,
Roorkee,India
purieskay@gmail.com

Prof. J.P.Singh
Professors,
Department of Management Studies,
Indian Institute of Technology,
Roorkee, Uttarakhand,India
jpsiitr@gmail.com

Abstarct: The goal of the artificial neural network is to create powerful artificial problem solving systems. The field of quantum computation applies ideas from quantum mechanics to the study of computation and has made interesting progress. Quantum Artificial Neural Network (QANN) is one of the new paradigms built upon the combination of classical neural computation and quantum computation. It has great values for theoretic study and potential to applications. In this paper an attempt is made to show how a single quantum neuron is able to perform the walsh-hadamard transformation which is an important unitary transformation often employed in quantum computation.

Keywords: Quantum computation, walsh-hadamard transformation, Quantum neuron, Quantum artificial neural network

I. INTRODUCTION

The strong demand in neural and quantum computations is driven by the limitations in the hardware implementation of classical computations. Classical computers efficiently process numbers and symbols with relatively short bit registers d < 128.But it has two major limitations to process Patterns, which are wide-band signals having d>100 bits. The first shortcoming is due to the hardware implementation of it i.e. in case of pattern processing, classical computers require enormous number of gates $\propto d^{4.8}$ (According to Rent's Law) to process d-bit registers [9]. On the other hand a typical computer program able to perform universal calculations on patterns requires $\propto 2^d$ operators [10].This fact excludes the possibility to use algorithmic approach for pattern processing.

Artificial neural network (ANN) can solve this problem because it uses the novel architecture which is able to process long bit strings and learning by example not by programming.ANN can solve complex problems which have poor knowledge domains.ANN also have some other features like parallel distributed processing and robustness. The main objective of Quantum computation is to minimize the size of the computer elements, which will be governed by the quantum laws. The research in quantum computing deals with the quantum analog of classical computational architecture which operates with quantum bits and quantum gates.

Quantum computers retain many features inherent in classical computers. They cannot operate wide band signals and cannot be simply trained by examples. Their efficiency will depend on the powerful quantum algorithms. Classical neural networks also face many problems like absence of rule for optimal architectures; time consuming training, limited memory capacity Quantum computation based on the quantum mechanical nature of physics [3, 4, 5] , which is inherently a parallel distributed processor having exponential memory capacity and easily trainable, but it has severe hardware limitations. Hence it is needed to combine both quantum computation and neural computation (Which is called Quantum Neural Network) to overcome the difficulties of classical computers, Quantum computers and neurocomputers [1-2]The rest of the paper is organized as: section-2 introduces concepts of quantum computation section-3 introduces concepts of artificial neuron; section-4 provides correspondence between QC and ANN, in section-5 a quantum neuron model is explained, in section-6 realization of walsh-hadamard transformation with a single quantum neuron . Finally the conclusion and references.

II. HADAMARD GATE IN QUANTUM COMPUTION

A quantum computation [11, 12] is a collection of the following three elements:

A. A register or a set of registers.
B. A unitary matrix, which is used to execute the quantum algorithms.
C. Measurements to extract information.

Hence quantum computation is the set $\{H, U, \{M_m\}\}$, where H is the Hilbert space of n-qubit register .$U \in U(2^n)$ is the quantum algorithm. $\{M_m\}$ is the set of measurement operators .Actual quantum computation processes are very different from that of classical counterpart. In classical computer we give input data through the input devices.

The input signal is stored in computer memory ,then fed into the microprocessor and the result is stored in memory before it is displayed in the screen .Thus the information travels around the circuit .In contrast , information in quantum computation is first stored in a register and then external fields , such as oscillating magnetic fields , electric fields or laser beams are applied to produce gate operations on the register. These external fields are designed so that they produce desired gate operation, i.e. unitary matrix acting on a particular set of qubits .

Hence information sits in the register and they are updated each time the gate operation acts on the register. In quantum information processing the Hadmard transformation or Hadamard gate is a one qubit rotation mapping the qubit basis states $|0\rangle$ and $|1\rangle$ to two superposition states with equal weight of the computational basis states $|0\rangle$ and $|1\rangle$.Quantum algorithms use Hadamard gate as an initial step to map n-qubits with $|0\rangle$ to a superposition of all 2^n orthogonal states in the $|0\rangle$, $|1\rangle$ basis with equal weight. When the Hadamard gate is applied twice in succession, then the final state is always the same as the initial state.

III. ARTIFICIAL NEURON

An artificial neuron is an adjustable unit performing, in general, a non linear mapping of the set of many, N, input (perceptron) values $x_1 \ldots x_N$, to a single output value y . The output value of a classical perceptron [6] is $y = f\left(\sum_{j=1}^{N} w_j x_j\right)$ (1)

Where f ( ) is the perceptron activation function takes the argument $a=W^T X$ and w_j are the weights tuning during learning process. The perceptron learning algorithm works as follows:

A. The weights w_j are initialized to small random numbers.

B. A *pattern* vector $(x_1 . . . x_N)$ is presented to the perceptron and the output y generated according to the rule (1)

C. The weights are updated according to the rule

$W_j(t+1) = w_j(t) + \eta(d - y)\, x_j$ **(2)**

Where *t* is discreet time, *d* is the desired output provided for training and $0<\eta<1$ is the step size.

IV. QUANTUM COMPUTATION AND ANN

Quantum computation is a linear theory but ANN depends upon non linear approach. The field of ANN contains several important ideas, which include the concept of a processing element (neuron), the transformation performed by this element (in general, input summation and nonlinear mapping of the result in to an output value), the interconnection structure between neurons, the network dynamics and the learning rule which governs the modification of interaction strengths. The main concepts of quantum mechanics are wave function, superposition (coherence), measurement (Decoherence), entanglement, and unitary transformations. In order to establish such correspondence is a major challenge in the development of a model of QNN. Many researchers use their own analogies in establishing a connection between quantum mechanics and neural networks [7], the correspondence between classical ANN and QNN is given below.

The power of ANN is due to their massive parallel, distributed processing of information and due to the nonlinearity of the transformation performed by the network nodes (neurons) .On the other hand; quantum mechanics offers the possibility of an even more powerful quantum parallelism which is expressed in the principle of superposition .This principle provides the advantage in processing large data sets.

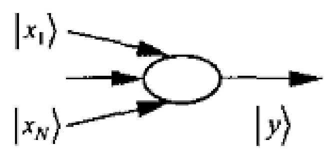
Classical Neural Network	**Quantum Neural Network**
Neuran $x_i \in \{0, 1\}$	Qubits $\|x\rangle = a\|0\rangle + b\|1\rangle$
Connection $\{w_{ij}\}_{ij=1}^{p-1}$	Entanglement $\|x_0, x_1 \ldots x_{p-1}\rangle$
LearningRule $\sum_{s=1}^{p} x_i^s x_j^s$	Superposition of Entangled state $\sum_{s=1}^{p} a_s \|x_0^s \ldots x_{p-1}^s\rangle$
Output Result N	Decoherence $\sum_{s=1} a_s \|x^s\rangle \Rightarrow \|x^k\rangle$

V. QUANTUM NEURON

Consider a neuron with *N* inputs $|x_1\rangle \ldots\ldots |x_N\rangle$ as shown in the fig. below x_j is a quantum bit (qubit) of the form ,

$|x_j\rangle = a_j|0\rangle + b_j|1\rangle = (a_j, b_j)^T$ (3)

Where $|a_j|^2 + |b_j|^2 = 1$



The output $|y\rangle$ can be derived by the rule [8] $|y\rangle = \hat{F} \sum_{j=1}^{N} \hat{w}_j |x_j\rangle$ (4)

Where $\hat{w}_j$ is 2×2 matrices acting on the basis $\{|0\rangle, |1\rangle\}$. $\hat{F}$ is an unknown operator that can be implemented by the network of quantum gates.

Let us consider $\hat{F}=\hat{I}$ be the identity operator .The output of the quantum perceprtron at the time t will be

$|y(t)\rangle = \hat{F} \sum_{j=1}^{N} \hat{w}_j(t) |x_j\rangle)$ (5)

In analogy with classical case equation (2) we can update the weights as follows

$\hat{w}(t+1) = \hat{w}_j(t) + \eta(|d\rangle - |y(t)\rangle)$ □$\langle x_j|$ (6)

Where $|d\rangle$ is the desired output .It can be shown the learning rule (6) derives the quantum neuron in to desired state $|d\rangle$ .Using rule (6) and taking modulo-square difference of real and desired outputs, we can get

$\| |d\rangle - |y(t+1)\rangle \|^2$

$= \| |d\rangle - \sum_{j=1}^{N} \hat{w}_j(t+1) |x_j\rangle \|^2$

$= \| |d\rangle - \sum_{j=1}^{N} \hat{w}_j(t) |x_j\rangle + \eta(|d\rangle - |y(t)\rangle \langle x_j | x_j\rangle) \|^2$

$= \| |d\rangle - |y(t)\rangle - \sum_{j=1}^{N} \eta(|d\rangle - |y(t)\rangle \|^2$

$= (1-N\eta)^2 \; \| |d\rangle - |y(t)\rangle \|^2$ (7)

For small η ($0<\eta<\frac{1}{N}$) and normalized input states $\langle x_j | x_j \rangle = 1$ the result of iteration converges to the desired state $|d\rangle$ .The whole network can be then composed from the primitives elements using the standard rules of ANN architecture .

VI. WALASH-HADAMARD RANSFORMATION IN QANN

Considering a quantum neuron with input $|x\rangle$ and output $|y\rangle = \hat{F} \sum_{j=1}^{N} \hat{w}_j |x_j\rangle$

Taking weight $\hat{w}$ and operator $\hat{F}$ as $\hat{w} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ and

$$\hat{F} = \hat{H} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad (8)$$

Where $\hat{H}$ is the Walsh- Hadamard transformation.

When input $|x\rangle = |0\rangle$

$$|y\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

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

When input $|x\rangle = |1\rangle$

$$|y\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

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

When $\hat{H}$ is applied to n qubits in the state $|0\rangle$ individually, $\hat{H}$ generates a superposition of all 2^n possible states .i.e 0 to $2^n - 1$.

$$(\hat{H} \otimes \hat{H} \otimes \ldots\ldots\ldots \otimes \hat{H})|000\ldots.0\rangle$$

$$= \frac{1}{\sqrt{2^n}} (|0\rangle + |1\rangle) \otimes (|0\rangle + |1\rangle) \otimes \ldots\ldots\ldots \otimes (|0\rangle + |1\rangle) \; |000\ldots0\rangle$$

$$= \frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n - 1} |x_j\rangle \quad (9)$$

Comparing equation (9) with equation (4) We can conclude that Walsh-Hardamard transformation can be implemented by a quantum neuron as given in the fig. with 2^n inputs and one output choosing $\hat{w} = \frac{1}{\sqrt{2^n}} \hat{I}$ and $\hat{F} = \hat{I}$

VII. CONCLUSION

In quantum computing walash-hardmard gate works as a unitary transformation to generate a superposition state from $|0\rangle$ and $|1\rangle$ .Also there are numerous important applications of the Hadamard transformation in quantum computation and neural network . In this paper we have discussed the basic concepts of QNN, a model of QNN, discussed the mechanism and the training algorithm and shown how a single quantum neuron is able to perform Walash-Hardamard transformation similar to classical neural network.

VIII. REFERENCES

[I] S. C. Kak, "On Quantum Neural Computing", Information Science, vo1.83, pp.143-160, 1995.

[2] T. Menneer, and A. Narayanan, "Quantum-inspired Neural Networks", Tech. Rep. R329, Univ. of Exeter, 1995.

[3] P. A. Benioff, "Quantum Mechanical Hamiltonian Model of Turing Machine",J Stat. Phy,r., vo1.29(3),pp.515-546, 1982. [4] R. P. Feynman, "Simulating physics with computers", Inr. J. of Theo. Physics, vo1.21(6/7), pp.467-488,1982.

[5] D. Deutsch, "Quantum theory, the Church-Turing principle and the universal quantum computer", Pmc. of Roy. Soc. of LondonSer A, voLA400, , pp.97-117, 1985.

[6] M.Minsky and P.Papert, Perceptrons: An Introduction to Computational Geometry, MIT Press, Cambridge, Mass. 1969.

[7] A. Ezhov and D. Ventura, "Quantum Neural Networks", N. Kanabov (Ed.), Future Directions .for lnielligent Systems und Information Sciences, Springer-Verlag, pp.213-234, 2000.

[8] M V. Altaisky "Quantum neural network", ArXive: qunat-ph/O l07012, 2001

[9] B.S. Landman and R.L.Russo, IEEE Transactions on computers, 20, 1469-1479(1971)

[10] A.A. Ezhov, S.A. Shumsky, Neurocomputing and its application in economics and business.Moscow Engineering Physics Institutes, Moscow,1999.

[11] Quantum computing by Mikio nakahara and Tetsuo ohmi ,CRC press

[12] Quantum computation and Quantum information by Michel A. Nielsen and Isaac L. Chuang , Cambridge university press.