World Academy of Science, Engineering and Technology
International Journal of Nuclear and Quantum Engineering
Vol:7, No:7, 2013

Quantum Computing: A New Era of Computing

Jyoti Chaturvedi

***Abstract*—**Nature conducts its action in a very private manner. To reveal these actions classical science has done a great effort. But classical science can experiment only with the things that can be seen with eyes. Beyond the scope of classical science quantum science works very well. It is based on some postulates like qubit, superposition of two states, entanglement, measurement and evolution of states that are briefly described in the present paper.

One of the applications of quantum computing, implementation of a novel quantum evolutionary algorithm (QEA) to automate the time tabling problem of Dayalbagh Educational Institute (Deemed University) is also presented in this paper. Making a good timetable is a scheduling problem. It is NP-hard, multi-constrained, complex and a combinatorial optimization problem. The solution of this problem cannot be obtained in polynomial time. The QEA uses genetic operators on the Q-bit as well as updating operator of quantum gate which is introduced as a variation operator to converge toward better solutions.

***Keywords*—**Quantum computing, qubit, superposition, entanglement, measurement of states, evolution of states, Scheduling problem, hard and soft constraints, evolutionary algorithm, quantum evolutionary algorithm.

I. Introduction

WE, humans are living with nature, i.e. the most mysterious thing. Man always wants to reveal the fact behind the working of the nature. This gives the way to inventions and discoveries. But nature keeps its things extremely private. That's why it is really difficult to explore the nature.

Classical Science has done its best to explain the working of nature. But it can only deal with large physical objects that move with the speed much slower than the speed of light. Once the objects getting very small and start moving very fast, classical science fails to explain the facts like:

- Why elements are very much different from each other which differ in their atomic structure by only few electrons, for example why Neon is an inactive gas while Sodium is most active elements having only one extra electron?
- Why oxygen makes water after combining with hydrogen?
- Why energy of an atom is quantized i.e. discrete i.e. why it assumes only some definite values of energies and not the values between them?

To explain these types of facts we have shift to atomic level explanation, where objects are extremely small and move very fast. At atomic level quantum science explains most accurately the behavior of nature.

Jyoti Chaturvedi is with the Dayalbagh Educational Institute, Dayalbagh, Agra, U.P.-282005, India (e-mail:jyotiscience@gmail.com).

II. Motivation

Experiments are the building blocks of science. These experiments lead us to the discoveries and inventions of science. Scientists conduct the experiments on physical objects to reveal their properties. If the object is very small then scientists find their properties on the basis of the interaction of these objects with other physical objects. In the sequence of these experiments there is an experiment which had forced the scientists to develop a new science which is more flexible than classical science i.e. quantum mechanics to explain the phenomenon called wave particle duality.

In this experiment there is an electron gun which can throw electrons, two small slits and a screen (detector) in front of the electron gun. When electrons were thrown one at a time a strange intensity distribution was seen on the screen. After passing through the two slits, on the detector electron form an intensity curve which is same as the intensity curve of waves (shown in Fig. 1). But we know that electron is a particle (because it has quantized energy), therefore it should show the behavior like a particle and give the intensity curve like the curve obtained with marbles (shown in Fig. 1).

If we take intensity or probability curve with respect to only on slit then it will give probability distribution P1(x) or P2(x) of electrons on the detector depending upon first slit is open or the second one is open. But if both the slits are open then we get the probability distribution as $P_{12}(x) = P_1(x) + P_2(x)$, in case of marbles. In case of water waves the scenario is different, when both the slits are open; the intensity on the detector is not the sum of individual intensities with respect to first and second slit. In fact, the amplitude of the waves is the sum of individual amplitudes of the wave with respect to first and second slit i.e. $a_{12} = a_1 + a_2$. Since the intensity is proportional to the square of the amplitude, therefore $I_{12} = a_{12}^2$. This shows that electrons, hence every physical entity behaves both as wave as well as matter particle in this universe. This fails the laws of classical science and leads us to the new science, quantum science.

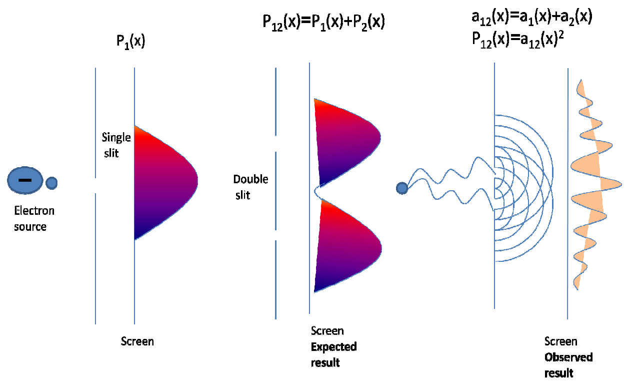

Fig. 1 Wave particle duality

III. Postulates and Phenomenon of Quantum Computing

Since quantum science deals with the entities which are very small (can't be seen by naked eyes), therefore we can't see the working and phenomenon of these entities. So we have to develop some postulates about the nature of these entities [14]-[17].

Quantum science is based on following postulates:

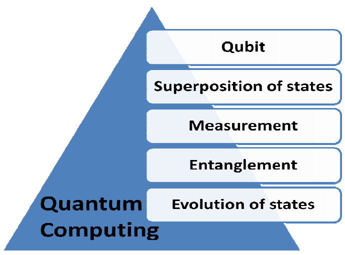

Fig. 2 Components of Quantum

A. Qubit (Quantum Bit)

Qubit is the basic building block of quantum computing. Qubit can be described in terms of the states of the system. These states belong to a state space defined as follows:

"Associated to any isolated physical system is a complex vector space with inner product (i.e. a Hilbert space) known as the state space of the system. The system is completely described by its state vector, which is a unit vector in the system's state space." [1]

Qubit is a two dimensional state space. These basis states of this space are generally taken as orthonormal states.

Let $|\Phi_0\rangle$ and $|\Phi_1\rangle$ be two orthonormal states, then a qubit state or qubit will be represented as the linear combination or superposition of states $|\Phi_0\rangle$ and $|\Phi_1\rangle$.

$$\text{i.e.} \quad |\psi\rangle = \alpha\,|\Phi_0\rangle + \beta\,|\Phi_1\rangle \quad , \alpha, \beta \in \mathbb{C} \qquad (1)$$
$$\text{s.t. } |\alpha|^2 + |\beta|^2 = 1$$
(Normalization condition).

α and β are the probability amplitudes of states $|\Phi_0\rangle$ and $|\Phi_1\rangle$ respectively. Square of these amplitudes give probabilities of the corresponding states. We can generalize (1) for classical binary states $|\Phi_0\rangle$ and $|\Phi_1\rangle$ by taking exactly one amplitude as 1 and other as 0.

In general, for simplicity we take $|\Phi_0\rangle$ as $|0\rangle$ and $|\Phi_1\rangle$ as $|1\rangle$ state respectively.

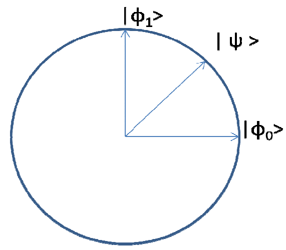

Fig. 3 Quantum state space

We can take electrons energy states to form a qubit. Here ground state is state $|0\rangle$ and excited state can be represented as $|1\rangle$.

B. Superposition Principle

Consider a k state system having distinct states 0, 1, …, k-1 in which a particle can reside.

Classical science says that the particle can be in exactly one state among these k states at a time.

According to quantum computing, if the particle can occupy these k states then it can also reside in between these states. Thus, Quantum superposition principle says that if a quantum system can be in one of the two states then it can also reside in linear superposition of these states.

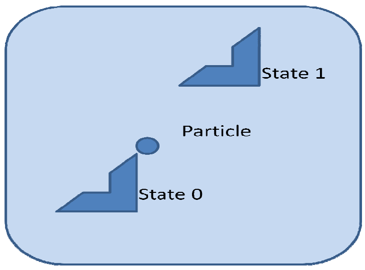

Fig. 4 Superposition state

Consider the analogy of stairs. Let us assume that two consecutive stairs form two states system. Classically a person climbing the stairs can be on lower stair (0 state) or on upper stair (1 state). But after leaving lower stair and before reaching the upper stair, the foot of the person passes through many different states which are the superposition of 0 and 1 states.

Consider another example of electron in hydrogen atom. According to this principle electron which has gained energy, does not make up its mind for a particular state (ground or excited state) and wander in between these states until it is measured.

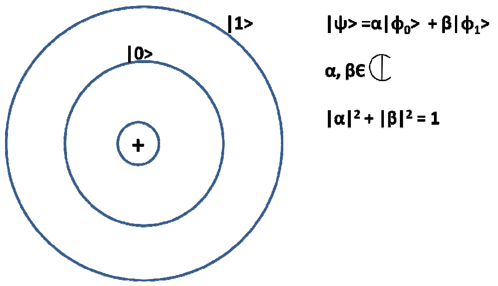

Fig. 5 Hydrogen atom

C. Measurement

In previous example linear superposition of 0 and 1 state is the hidden information of the private world of the electron. So, on measurement we can't observe this type of state in any system. In fact, on doing measurement electron makes up its mind and moves to one of the states as shown in Fig. 6. Hence the new state is exactly one among 0 or 1 states.

In measurement process the strange thing is that the state yielded by measurement process is the state actually occupied by the particle (experimentally seen).

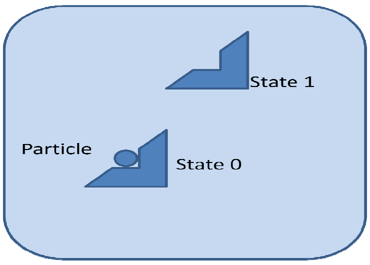

Fig. 6 Measurement state

D. Entanglement

In quantum computing we can deal with the states of more than one qubit simultaneously.

Let us assume that we have two qubits as follows:

$|\psi_1\rangle = \alpha_1 |0\rangle + \beta_1 |1\rangle$ and
$|\psi_2\rangle = \alpha_2 |0\rangle + \beta_2 |1\rangle$, $\alpha_1, \alpha_2, \beta_1, \beta_2 \in \mathbb{C}$
s.t. $|\alpha_1|^2 + |\beta_1|^2 = 1$
and $|\alpha_2|^2 + |\beta_2|^2 = 1$

Then the composite state of these qubits will be obtained with the help of tensor product of these states shown below:

$$|\psi\rangle = |\psi_1\rangle \otimes |\psi_2\rangle = (\alpha_1 |0\rangle + \beta_1 |1\rangle).(\alpha_2 |0\rangle + \beta_2 |1\rangle)$$
i.e. $$|\psi\rangle = \alpha_1\alpha_2|00\rangle + \alpha_1\beta_2|01\rangle + \beta_1\alpha_2 10\rangle + \beta_1\beta_2 |11\rangle \quad (2)$$

On seeing this, the obvious question that comes into our mind is: Can composite state be decomposed into its component states? The answer to this question is 'not always'. The states which cannot be decomposed into its component states are called entangled states. In this state identity of the individual qubit is lost. This phenomenon is called entanglement.

This is same as when we mix salt in pure water; they both lose their identities and reached in entangled state in which they cannot be separated.

To explain this let us consider famous bell state i.e. the composite state of two qubits:

$$|\psi\rangle = 1/\sqrt{2}\ |00\rangle + 1/\sqrt{2}\ |11\rangle$$

This cannot be decomposed into its component qubit states. If so then according to (2),
we have,

$\alpha_1\alpha_2 = 1/\sqrt{2}$ and $\beta_1\beta_2 = 1/\sqrt{2}$,
Thus $\alpha 1$, $\alpha 2$, β_1, β_2 will be non zero. (3)

But $\alpha_1\beta_2 = 0$ and $\beta_1\alpha_2 = 0$,
Thus α_1 or β_2 will be 0 and α_2 or β_1 will be 0, which is a contradiction to (3).

There is a strange behavior shown by entangled states i.e. if we find the state of one qubit then the state of the other qubit will be deterministic no matter how far the other qubit is placed from first qubit. Consider the same example to explain this. Let the first qubit is in state $|0\rangle$, then the new superposition state will be

$$|\psi\rangle = 1/\sqrt{2}\ |00\rangle$$

After normalizing this, we get
$|\psi\rangle = \dfrac{1/\sqrt{2}\ |00\rangle}{1/\sqrt{2}}$, which gives
$|\psi\rangle = |00\rangle$,

This gives probability 1 for other qubit to be in state 0. Hence measurement of one qubit affects the measurement of the other qubit.

E. Evolution of Quantum States

In order to explore the state space we have to evolve new states with the help of given states as boiling of water results in transformation of liquid state to gas state (vapors).

"The evolution of closed quantum system is described by a unitary transformation. That is, the state $|\psi_1\rangle$ of the system at time t_1 is related to the state of $|\psi_2\rangle$ of the system at time t2 by a unitary operator U which depends only on times t_1 and t_2"[1].

i.e. $|\psi_1\rangle = U\,|\psi_2\rangle$

Geometrically, unitary transformation is the rotation of the state space by some angle. Hence the given state vector, resulting in a state vector with the same length.

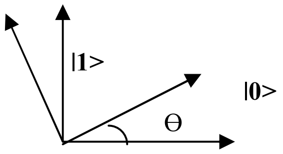

Fig. 7 Rotation of state space

Consider the qubit state is:

$| \psi_1 > = a |0> + b |1>$

and the unitary operator is:

$$U= \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \text{s.t.} \quad UU^*=I$$

Then evolution of state $| \psi_2 >$ will be as follows:

$$| \psi_2 > \; = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} * (a |0> + b |1>)$$

Hence, $| \psi_2 > = b |0> + a |1>$

Types of unitary transformation

- Hadmard Gate– This operator rotates the state space by Π/8 radian in the real plane.

$$H = 1/\sqrt{2}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

- Not Gate – This flips the bit from 0 to 1 and vice-versa.

$$NOT = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

- Rotation Gate– It rotates the plane by angle Θ.

$$U = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$

- Phase Flip– The phase flip is a NOT gate acting in basis:

$$|v> = 1/\sqrt{2}\ |0> + 1/\sqrt{2}\ |1>,$$
$$|v'> = 1/\sqrt{2}\ |0> - 1/\sqrt{2}\ |1> \qquad [3].$$

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

IV. Quantum Computing

To make human life simple, we have built computers which do the complex computation in few seconds. But today's computers are not so efficient with respect to some computational problems (e.g. NP problems).The reason is that these are classical computers which do irreversible computation. E.g. if we and two bits we get one resultant bit, thus we lose information about the other bit which released as heat in computer and make it inefficient. So we need a computer which can do reversible computation and does not lose any information.

To solve the problems of classical computing, concept of quantum computers has been introduced. Quantum computing is really efficient as shown in following example:

- we have 500 particles
- Thus, we have 2^{500} states.
- Each state consists of 500 1s and 0s.
- So we need only 500 qubits to represent these 2^{500} states (according to principle of superposition).
- 2^{500} >particles in the universe*age of the universe
- So, if we take number of classical computers = particles in the universe and make them work in parallel, even then it will take time i.e. more than the age of universe with only 500 particles. While with quantum computer it will take only few seconds to do this.

V. Application of Quantum Evolutionary Algorithm to the Timetabling Problem

The scheduling of courses in universities or institutions is known to be a highly constrained combinatorial optimization problem. Generally it is done manually. Solution of this problem usually involves taking the previous year's timetable and modifying that so it will work for the next year. Nowadays most of good universities provide much greater flexibility to students for selecting courses as well as greater choices. Also universities are enrolling more students into a wider variety of courses including an increasing number of combined degree courses. Therefore the process of finding a timeslot for each course so that no two subjects of any student clash, has been shown to be equivalent to assigning colors to vertices in a graph so that adjacent vertices always have different colors. This has been proved to lie in the set of NP-complete problems, which means that carrying out an exhaustive search for the timetable is not possible in a reasonable time. Hence there should be some automatic timetabling system, which creates timetable every year [1], [8]. Many course-timetabling algorithms have been proposed. The most popular methods that have and are being introduced for such a system are based on heuristics (e.g. graph coloring etc.), local search techniques (e.g. simulated annealing, tabu search). Some researchers have also employed evolutionary algorithms (mainly genetic algorithms and its variants) and got good solutions. Due to the complexity, the general genetic algorithm converges slowly and easily converges to local optima [6], [12], [7]. A novel quantum-inspired evolutionary algorithms (QEA) has been implemented for the Course Timetable Problem (CTP). The QEA uses genetic operators on the Q-bit as well as updating operator of quantum gate which is introduced as a variation operator to drive the individuals toward better solutions [2].

In this work we have implemented Quantum Evolutionary Algorithm for the Timetable Problem. The experimental results demonstrate that the QEA performs well and can also provide a set of high quality timetables.

VI. Timetable Problem

Timetable problem deals with construction of timetable of the courses. Timetable construction can be considered as generic scheduling activity. The scheduling problems are essentially problems that deal with effective distribution of resources among tasks.

University Timetable problem also presents a set of ***tasks (classes)*** and a set of ***resources (rooms, Labs, groups, instructors)***. Every task requests some resources for its realization on certain time slot. The goal is to make the timetable at least as good as experienced human (expert) would make it while satisfying the required constraints. Though each university may have different constraints, two types of constraints are commonly considered [4], [9]:

a) Hard constraints which must be satisfied in a timetable in order to make it usable (feasible).
b) Soft constraints which are desired but not absolutely essential. Soft constraints are those that are set by the user to produce a timetable that is more suited to their preferences. In other words, violation of only soft constraints means that a valid solution was produced, but only with less quality, depending on the frequency of soft constraint violations.

Some of the ***hard constraints*** for timetable problem are given here:

- A student should have only one class at a time.
- A teacher should have only one class at a time.
- A room should be booked for only one class at a time.
- Only one class of a course should be scheduled on a day.

Some of the ***soft constraints*** for timetable problem are given here:

- Student should not have any free time slot between two classes on a day.
- Classes of a teacher should be well spread over the week.
- A smaller class should not be scheduled in a room which can be used for a bigger class.
- Two or more number of classes should not be allotted to a teacher in a day.
- A class should be scheduled only in a specific room, if required, otherwise in a general room which has sufficient sitting capacity for the students of the class.
- A class should be scheduled only at a specific timeslot, if required.

A feasible timetable is one that does not violate any of the hard constraints. On the other hand, a "good" timetable is one that satisfies all hard constraints as well as a number of the soft constraints (or all if possible).

Making a valid timetable is not an easy task. The reason is that it is NP hard. Consider that there are ***'t' periods*** and ***'c' classes*** to be scheduled then there are ***t^c ways*** to do this. It increases exponentially when number of classes increase. Hence, time to finding the solution also increases exponentially. Therefore, we try to reach nearly optimal solution to this problem [10].

A time and computational saving idea that is adopted in this work is to split the complete timetabling problem into two phases: time (day and hour) allocation and place (classroom) allocation [11]. The assumption is that, after obtaining a solution to the first phase, the second phase is much easier to solve, i.e. there are many solutions to the second phase that satisfy the first phase. The first phase is the most important and difficult one. The constraints are much stronger and the computation effort is much higher in this first phase. After finding a suitable schedule, the allocation of the rooms is a conceptually similar task with the first phase and consequently it can be solved in the same way. The argument behind this separation is that from the search space for n activities, each with a starting time ranging from 0 to m−1, and an allocated room, ranging from 0 to p−1, i.e. an overall search space of $m^n \times p^n$ possible solutions, the two-phase approach will derive a search space of $m^n + p^n$ possible solutions. This intuitive approach brings obviously an important improvement in the speed and effort of computation of any solving algorithm [3].

VII. Evolutionary Algorithm

Evolutionary algorithms are the algorithms which are inspired by the famous principle of "***Darwinian Evolution Theory***". These algorithms are a very impressive tool to solve complex combinatorial optimization problems. They generally only involve techniques implementing mechanisms inspired by biological evolution.

Candidate solutions to the optimization problem play the role of individuals in a population, and the cost function determines the environment within which the solutions "live". Evolution of the population then takes place after the repeated application of the above operators.

Recombination and ***mutation*** create the necessary diversity and thereby facilitate novelty, while ***selection*** acts as a force increasing quality. The net effect of ***survival of the fittest*** is that the average fitness of the population increases with each generation.

VIII. Quantum Evolutionary Algorithm(QEA)

Evolutionary algorithms (EAs) are characterized by the representation of the individual, the evaluation function representing the fitness level of the individuals, and the population dynamics such as population size, variation operators, parent selection, reproduction etc. To have a good balance between exploration of search space and exploitation of best solution, these components should be designed properly. Quantum evolutionary algorithm (QEA) can treat the balance between exploration and exploitation very easily. Also, QEA can explore the search space with a smaller number of individuals and exploit the search space for a global solution within a short span of time.

A.Q-bit & Q-bit Individual

Q-bit or quantum bit is the smallest unit of information in QEA. It is represented as a pair of numbers (α,β), where $|\alpha|^2 + |\beta|^2 = 1$. $|\alpha|^2$ gives the probability that qubit is found in 0 state and $|\beta|^2$ is the probability that qubit is found in 1 state. A Q-bit may also be in a linear superposition of the two states. A Q-bit individual as a string of m Q-bits is defined as [5],[13]

$$\begin{bmatrix} \alpha_1 \\ \beta_1 \end{bmatrix} \begin{matrix} \alpha_2 \\ \beta_2 \end{matrix} \begin{matrix} \cdots \\ \cdots \end{matrix} \begin{matrix} \alpha_m \\ \beta_m \end{matrix}$$

where $|\alpha|^2 + |\beta|^2 = 1$, for i=1,2,...m.

Qubit representation has the advantage that it is able to represent a linear superposition of states. If there is for instance, a three qubit system with three pairs of amplitudes, then it contains the information of eight classical states.

B. Population

QEA maintains a population of qubit individuals $Q(t)=\{q^t_1, q^t_2, \ldots, q^t_n\}$ at generation t, where n is the size of population, and q^t_j is a Q-bit individual defined as [5],[13]

$$q_j^t = \left[\begin{array}{c|c|c|c} \alpha_{j1}^t & \alpha_{j2}^t & \cdots & \alpha_{jm}^t \\ \beta_{j1}^t & \beta_{j2}^t & \cdots & \beta_{jm}^t \end{array}\right]$$

where m is the number of Q-bits, i.e., the string length of the Q-bit individual, and j = 1, 2, ……, n.

Using qubit individual population we can generate binary population.

C. Quantum Rotation Gate

Q-bit individuals in Q(t) are updated by applying Q-gates defined as a variation operator of QEA, by which operation the updated Q-bit should satisfy the normalization condition, $|\alpha'|^2 + |\beta'|^2 = 1$, where α' and β' are the values of the updated Q-bit. The following rotation gate is used as a basic Q-gate in QEA:

$$U(\Delta\theta_i) = \begin{bmatrix} \cos(\Delta\theta_i) & -\sin(\Delta\theta_i) \\ \sin(\Delta\theta_i) & \cos(\Delta\theta_i) \end{bmatrix},$$

where $\Delta\theta_i$, i = 1, 2……,m, is a rotation angle of each Q-bit toward either 0 or 1 state depending on its sign.

D. Migration

A migration in QEA is defined as the process of copying current best solution in binary population in place of previous solutions. A local migration is implemented by replacing some of the solution in the best individual's population, while global migration is implemented by replacing all the solution in best individuals population.

IX. Methodology

Here is the description about the implementation of the algorithm to find the solutions of the timetable problem.

A. Representation of Solution Chromosome

- Quantum chromosome is represented as a two dimensional array with each cell containing the amplitudes of 0 and 1.
- On the other hand solution chromosome is represented as a two dimensional array having timeslots on horizontal axis and days as on vertical axis as shown below:

	10:30 am	**11:20 am**	**…**		**3:45 pm**
Monday					
Tuesday		Course1			
…					
Saturday	Course3			Course2	

Fig. 8 Solution Representation

- Solution chromosome (i.e. course code) is made up of binary bits.
- Each input given by the user is mapped with some integer value and used as the coded integer value.

B. Evaluation of Solution Chromosome

Evaluation function or fitness function takes an individual, evaluate its fitness according to the following constraints and return the fitness. The solution is called good or bad solution according to its fitness value. Here more the solution satisfies the constraints; the more it is called fit.

1. Following Hard Constraints Are Used in the Implementation

- Each teacher should have only one lecture in one period.
- Each student should have only one lecture in one period.
- Students should have classes on all six days of the week.
- Classes of the subjects from same group (e.g. pm, mc, bz etc.) should not be conducted in parallel.
- Class of a subject should be scheduled in one of the rooms that are assigned to that subject.
- Number of classes of a particular subject in a week should be according to the credit of that subject.
- A room should be scheduled only for one class at a time.
- Common classes of different years should be scheduled at the same time and same day.
- Classes of core courses should be scheduled in first or last period.
- Teacher's preferences.
- There should not be more than two classes of a subject on a day.

2. Following Soft Constraints Are Used in the Implementation

- Students should have continuous classes in a day.
- There should be gap between two lectures of a teacher.
- Labs should be scheduled in the second half.
- Classes should be best fit (for a room).
- Class of a particular subject should be scheduled in the same room throughout the week.
- Class of a particular subject should be held at the same time throughout the week.

C. Algorithm

Algorithm shown in Fig. 9 is used in the implementation. The algorithm contains the following functions:

- Main
- Input
- Make timetable
- Repair
- Fitness
- Update chromosome

```
main()
    begin
      Input to a file
      Make a timetable
    end
    function-input ()
    begin
      input from the user
    end
    Function-make_timetable ()
    begin
        • Initialize populations
        • Repair chromosomes
        • Find fitness of chromosomes

        While(generation<max_gen)
        begin
            • Generate solution population
            • Repair solution chromosome
            • Find fitness of solution chromosome
            • Find best fitness solution
            • compare it with previous best solution fitness and store the best
            • update quantum chromosome
      end

      Store gbest chromosome
    end
    Function-repair()
    begin
      satisfy hard constraints
    end
    Function-fitness ()
    begin
        fitness value of chromosome ( i.e. number of soft constraints satisfied)
    end
    Function-update_quantum_chromosome()
    begin
        update Q-chromosome using quantum rotation gate
    end
```

Fig. 9 Algorithm

➢ *Function- Make Timetable*

Formation of the timetable takes place in the function make timetable shown in Fig. 10.

Here quantum chromosomes (i.e. all α and β) are initialized with the value $1/\sqrt{2}$. Binary solution chromosomes are initialized by observing the states of initial quantum chromosome (i.e. by selecting either 0 or 1 for each bit using the probability of 0 and 1 in quantum chromosome). Initialization is done as shown in Fig. 11.

```
      begin
       t ← 0
i)    initialize q chromosome population Q(t)
ii).  make solution chromosome population P(t)
ii)   evaluate P(t)
iv).  store the best solution among P(t) into B(t)
      while(t<max_gen) do
      begin
          t ← t+1
v)        Make P(t) by observing the state of Q(t-1)
  vi)     Evaluate P(t)
  vii)    Update Q(t) using Q- gates
viii)     Store the best solutions among B(t-1) and P(t) into B(t)
  ix)     Store the best solution b among B(t)
  x)      Migrate b globally or locally.
```

Fig. 10 Make timetable

```
      begin
       i ← 0
       while(i<individual length)
       begin
i)         i ← i+1
ii)        if random [0,1) < |β|^2
         x_i (i^th bit of chromosome x) ← 1

         else
           x_i ← 0
       end
      end
```

Fig. 11 Initialization

After each iteration Q-chromosome is updated according to the fitness of previous solution chromosome as follows:

```
      begin
       i ← 0

       while(i < length of chromosome)
       begin
i)         i ← i+1
         determine ΔΘ_i, with the the lookup
             table, obtain (α_i', β_i') from the
           following

ii)        if(q is located in the first or third
             quadrant)

         [α_i', β_i']^T = U(ΔΘ_i) [α_i, β_i]^T

           else
         [α_i', β_i']^T = U(-ΔΘ_i) [α_i, β_i]^T
       end
iii)  q ← q'
```

Fig. 12 Chromosome updation

where

$\Theta_1 = \Theta_2 = \Theta_4 = \Theta_6 = \Theta_7 = \Theta_8 = 0$

$\Theta_3 = 0.01\pi$

$\Theta_5 = -0.01\pi$

TABLE I
LOOKUP TABLE

T	b_i	f(x) > f(b)	$\Delta\theta_i$
0	0	False	θ_i
0	0	True	θ_2
0	1	False	θ_3
0	1	True	θ_4
1	0	False	θ_5
1	0	True	θ_6
1	1	False	θ_7
1	1	True	θ_8

X. RESULTS

A. The Results Obtained from QEA for Following Input is Shown in Fig.13

Inputs:

TABLE II
INPUT (A)

Faculty	Science
Department	Mathematics
Programme	B.Sc. 1st semester
Group	PM

TABLE III
INPUT (B)

Preferred rooms
Lt1
Lt2

TABLE IV
INPUT (C)

Course code	Course name	Credit	Teacher
Mah101	Mathematics I	3.0	Dr. S. Srivastava
Mah102	Mathematics II	3.0	Dr. S. Srivastava
Mam101	Calculus I	3.5	Dr. S. Sharma
Mam102	Discrete mathematics	3.5	Dr. D. Pandey

Result:

```
following is the generated timetable

            8:50    9:30    10:30   11:20   12:10   1:00    2:15    3:00    3:45

Monday      ------  ------  mah102  ------  ------  ------  ------  ------  ------

Tuesday     ------  ------  ------  mam101  ------  mam101  ------  ------  ------

Wednesday   ------  ------  ------  ------  ------  mam102  ------  mam102  ------

Thursday    ------  ------  ------  ------  ------  mam101  ------  ------  mah101

Friday      ------  ------  ------  ------  ------  ------  mah102  mam101  mah101

Saturday    ------  ------  ------  mam102  mah102  mah101  mah102  mam102  mah101

following is the timetable for rooms

            8:50    9:30    10:30   11:20   12:10   1:00    2:15    3:00    3:45
Monday      ------  ------  lt1     ------  ------  ------  ------  ------  ------
Tuesday     ------  ------  ------  lt1     ------  lt1     ------  ------  ------
Wednesday   ------  ------  ------  ------  ------  lt1     ------  lt1     ------
Thursday    ------  ------  ------  ------  ------  lt1     ------  ------  lt1
Friday      ------  ------  ------  ------  ------  ------  lt1     lt1     lt1
Saturday    ------  ------  ------  lt1     lt1     lt1     lt1     lt1     lt1
```

Fig. 13 Result

XI. EXPERIMENTAL GRAPHS

It was seen that the fitness of solution i.e. quality of the timetable increases with the increase in population size and generations.

- The algorithm increases sharply when generation number is between zero and a thousand, however after 5000 generations nearly the same fitness value is observed. Although we reached 30000th generation, we did not get a solution with a fitness value of 1. It is almost impossible that generating timetables which satisfy all individual preferences.

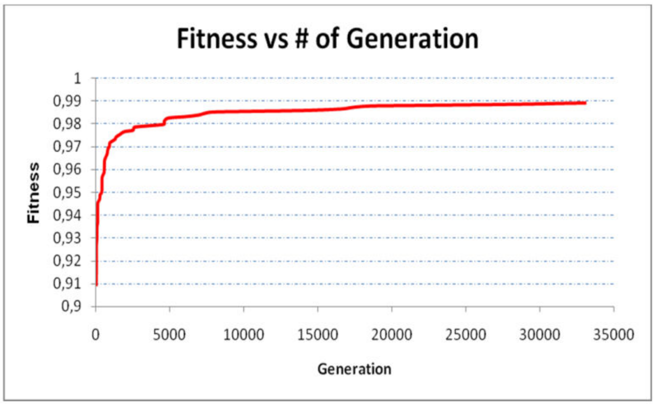

Fig. 14 Fitness vs. Generation (a)

It can also be seen from the graph shown in figure below that fitness becomes consistent and doesn't increase by a large amount after some time. Hence, after some amount of increment in generation fitness becomes consistent.

It can also be seen from the graph shown in figure below that fitness becomes consistent and doesn't increase by a large amount after some time. Hence, after some amount of increment in generation fitness becomes consistent.

TABLE V
EFFECT OF POPULATION SIZE ON FITNESS IN QEA

Population size	Fitness	Generations
20	0.01186	120
40	0.01205	110
100	0.01258	90
200	0.01283	80

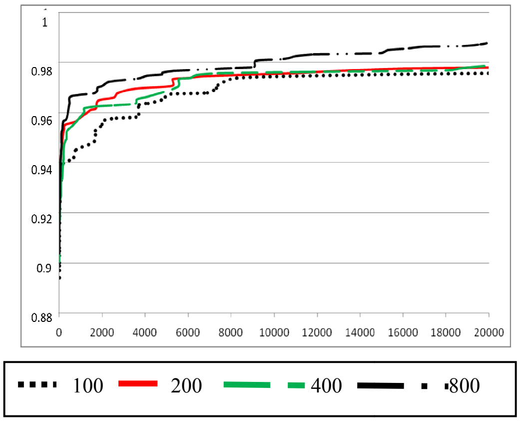

Fig. 15 Fitness vs Generation (b)

XII. Conclusion

Quantum computing with its various postulates is well suited to describe those elements of the universe which are very small in size and very fast moving. Thus, with the help of this type of computing we can describe the working of basic building block of the universe, hence the working of the universe.

Besides this, the time Table problem that is presented in this paper is a NP hard problem with various typical hard and soft constraints. Here this problem has been solved using quantum inspired Evolutionary Algorithm (QEA). The results show that QEA performs well and gives a good solution. With the help of QEA, generation of good time table for different courses offered by the Faculties of Dayalbagh Educational Institute, Dayalbagh, Agra after considering many complex hard and soft constraints is possible.

References

[1] Nilesh Gambhava and Gopi Sanghani, "University Examination Timetabling using Genetic Algorithm", *in proc International Conference on Applied Artificial Intelligence ICAAI'03, , Kolhapur, INDIA,* Monday, 15th December 2003.

[2] Yu Zheng , Jing-fa Liu ,Wue-hua Geng and Jing-yu Yang , "Quantum-Inspired Genetic Evolutionary Algorithm for Course Timetabling", *in Proc of 3rd International Conference on Genetic and Evolutionary Computing, 2009. WGEC '09,* pp 750 – 753, 14-17 Oct. 2009

[3] Liviu Lalescu and Costin Badica, "An Evolutionary Approach for School Timetabling", *in Proc. 14th International Conference on Control Systems and Computer Science, vol. II,* pp114, July 2-5, 2003.

[4] E.K. Burke, J. Kingston and K. Jackson, "Automated university timetabling: the state of the art", *The Computer Journal,* Vol. 40, No. 9, pp. 565-571, 1997

[5] Kuk-Hyun Han and Jong-Hwan Kim , "On the Analysis of the Quantum-inspired Evolutionary Algorithm with a Single Individual*", in Proceedings of the 2001 International Conference on Artificial Intelligence,* July 16-21, 2006.

[6] Kuldeep Kumar, Sikander, Ramesh Sharma AND Kaushal Mehta, " Genetic Algorithm Approach to Automate University Timetable", *International Journal of Technical Research(Ijtr)*, Vol. 1, Issue 1,PP. 33-37, Mar-Apr2012.

[7] Ho Sung C. Lee, "*Timetabling Constrained System via Genetic Algorithm*", Masters Thesis. University of Philippines, Diliman, Quezon City, 2000.

[8] Liviu Lalescu and Costin Badica, "*Timetabling Experiments Using Genetic Algorithms",* University of Craiova, Faculty of Control, Computers and Electronics Software Engineering Department, Romania, July 2003.

[9] Branimir Sigl, Marin Golub, Vedran Mornar, "Solving Timetable Scheduling Problem by Using Genetic Algorithms", *in proc. of 25th Int. Conf. Information Technology Interfaces IT1, Cavtat, Croatia*, pp. 519-524, June 2003.

[10] C. Y. Cheong, K. C. Tan and B. Veeravalli, "Solving the Exam Timetabling Problem via a Multi-Objective Evolutionary Algorithm – a more general approach", in *Proceedings of the IEEE Symposium on Computational Intelligence in Scheduling, CI-Sched, Honolulu, HI, USA*, pp. 165 – 172, 2007.

[11] M. Al-Betar, A. Khader and T. Gani, "A Harmony Search Algorithm for University Course Timetabling ", in proc. *of 7th Intl. Conf. on the Practice and Theory of Automated Timetabling.MONTREAL, CANADA*, Aug. 2008.

[12] Kuldeep Singh Sandhu, " Automating Class Schedule Generation in the Context of a University Timetabling Information System", Ph.D. Dissertation, Nathan Campus, Griffith University., 21st September 2001.

[13] Kuk-Hyun Han and Jong-Hwan Kim, "Quantum-inspired Evolutionary Algorithm for a Class of Combinatorial Optimization*", IEEE transaction on Evolutionary Computation,* Vol. 6, No. 6, December 2002.

[14] Mark Oskin, *Lecture Notes for Quantum Computing* , Department of Computer Science and Engineering, University of Washington, Washington.

[15] U.V. Vazirani, *lecture notes on Qubits, Quantum Mechanics, and Computers for Chem/CS/Phys191,* University of California, Berkeley, 2012.

[16] Vishal Sahni, *Quantum Computing.* New Delhi: TATA McGRAW – HILL Publishing Company LTD., 2007.

[17] Ajoy Ghatak and S. Lokanathan, *Quantum Mechanics Theory and Application.* New Delhi: MACMILAN INDIA LTD., 2004.