Optimization methodology based on Quantum computing applied to Fuzzy practical unit commitment problem

B. N. AL-Matrafi [a] , A. A. Mousa [a,b],

[a] Department of Mathematics and Statistics, Faculty of sciences, Taif University,
Taif, El-Haweiah, P.O. Box 888, Zip Code 21974, Saudi Arabia.
[b] Department of Basic Engineering Science, Faculty of Engineering, Menoufia University, Egypt

Abstract— In this Paper, an algorithm to solve constrained unit commitment problem (UCP) with operational and power flow constraints will be developed to plan an economic and secure generation schedule. This Paper presents a quantum gentic algorithm QGA for unit commitment problem (UCP) problem. Our approach integrates the merits of both Genetic algorithm and quantum computing and it has two characteristic features. Firstly, unit commitment problem UCP has been defined, where the input data involve many parameters whose possible values may be assigned by the expert. Secondly, quantum genetic algorithm can represent a linear superposition of states, and there is no need to include many individuals. QGA has an excellent ability of global search due its diversity caused by the probabilistic representation. Several optimization runs of the proposed approach will be carry out on the test problems to verify the validity of the proposed approach. In this perspective, having a quantum version of a genetic algorithm seems to be a relevant topic in the future, when quantum computers will be available. Moreover, the integration between the two paradigms can be a way of applying quantum computation to hard problems for which a quantum algorithm is not available yet.

Index Terms— unit commitment problem; quantum computing; Genetic algorithm.

1 Introduction

A problem that must be solved frequently by a power utility is to determine economically a schedule of what units will be used to meet the forecasted demand and operating constraints, such as spinning reserve requirements, over a short time horizon. This problem is commonly referred to as the unit commitment problem (UCP). The UCP is a mixed-integer programming problem and is in the class of NP-hard problems. In other words, the UCP is to determine a minimal cost turn-on and turn-off schedule of a set of electrical power generating units to meet a load demand while satisfying a set of operational constraints. Because of its size and NP-hardness, the true optimal solution of the UCP is normally difficult to obtain. Many optimization methods have been proposed to solve the UCP. For example, we mention the priority list method [1], branch-and-bound methods [2], dynamic programming approaches [3], and Lagrangian relaxation(LR) methods [4].

In this paper, an algorithm to solve environmental constrained unit commitment problem (UCP) with operational and power flow constraints will be developed to plan an economic and secure generation schedule.

Evolutionary Algorithms (EAs) [4] are population-based optimization techniques based on the classical laws of inheritance and Darwin's theory of evolution. EA is an umbrella term that covers several approaches, namely Genetic Algorithms (GA) [5], Evolutionary Strategies (ES) [6], Genetic Programming (GP) [7] and Differential Evolution (DE) [8] which are based on the same principles but differ in the application of these principles. EAs have been successfully used in many problem domains. GAs, ES and DE are mainly for search and optimization while GP is used more for automatic program generation, prediction and machine learning tasks.

Unfortunately, the UCP is a highly nonlinear and a multimodal optimization problem. Therefore, conventional optimization methods that make use of derivatives and gradients, in general, not able to locate or identify the global optimum. On the other hand, many mathematical assumptions such as analytic and differential objective functions have to be given to simplify the problem. Furthermore, this approach does not give any information regarding the trade-offs involved. Heuristic algorithms have been recently proposed for solving UCP [9-12]. The results reported were promising and encouraging for further research. Moreover the studies on heuristic algorithms over the past few years have shown that these methods can be efficiently used to eliminate most of difficulties of classical methods. It can be concluded from the foregoing that optimization evolutionary algorithms cannot simultaneously

- ***B. N. AL-Matrafi*** *is currently at Department of Mathematics and Statistics, Faculty of sciences, Taif University, Taif, El-Haweiah, P.O. Box 888, Zip Code 21974, Saudi Arabia.*
- ***A. A. Mousa*** *is currently at Department of Mathematics and Statistics, Faculty of sciences, Taif University, Taif, El-Haweiah, P.O. Box 888, Zip Code 21974, Saudi Arabia*

International Journal of Scientific & Engineering Research, Volume 4, Issue 11, November-2013 1139
ISSN 2229-5518

meet the requirements of convergence to the optimal solution; hence the need to incorporate a mechanism that improves convergence to the true optimal solution. Due to these drawbacks, more and more researches on hybridization algorithm are to find the effective hybrid algorithm. Recently, some quantum genetic algorithms (QMAs) [13-15] have been proposed for some combinatorial optimization problems, such as traveling salesman problem, knapsack problem, and filter design,…etc.

This paper intends to present quantum computing based on genetic algorithm for solving UCP. Our approach integrates the merits of both Genetic algorithm and quantum computing and it has two characteristic features. Firstly, unit commitment problem UCP has been defined, where the input data involve many parameters whose possible values may be assigned by the expert. Secondly, quantum genetic algorithm can represent a linear superposition of states, and there is no need to include many individuals. QGA has an excellent ability of global search due its diversity caused by the probabilistic representation. Several optimization runs of the proposed approach will be carry out on the standard IEEE systems to verify the validity of the proposed approach.

2. Nonlinear programming problem (NLPP)

Any evolutionary computation technique applied to a particular problem should address the issue of handling unfeasible individuals. In general, a search space S consists of two disjoint subsets of feasible and unfeasible subspaces F and U respectively. We do not make any assumptions about these sub-spaces; in particular they need not be convex and they need not be connected (e.g., as it is the case in the example in figure 1) [16,17] where feasible part F of the search space consist of two disjoined subsets). The general nonlinear programming problem [18] for continuous variables is to find $\bar{x}$ so as to

$$Min \quad f\left(\bar{x}\right), \bar{x} = \left(x_1,.....,x_n\right) \in R^n,$$

Where $\bar{x} \in F \subseteq S$. The set $S \subseteq R^n$ defines the search space and the set $F \subseteq S$ defines a feasible part of the search space. Usually, the search space S is defined as n-dimensional rectangle in R^n (domains of variables defined as lower and upper bounds): $\text{left}(i) \le x_i \le \text{right}(i),\ 1 \le i \le n$ Whereas the feasible set F is defined by the search space S and an additional set of constraints:

$$g_j\left(\bar{x}\right) \le 0, \quad for \qquad j = 1,.........,m$$

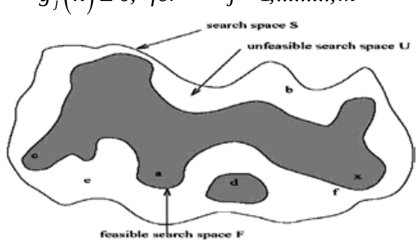

Fig.1. A search space and its feasible part.

Thus the nonlinear programming problem (NLPP) can be defined as follows:

$$NLPP: \ Min \quad f\left(\bar{x}\right)$$

$$\text{s.t.} \quad F = \left\{ \bar{x} \in R^n \middle| g_i\left(\bar{x}\right) \le 0, i = 1,2,...,k \quad and \ h_j\left(x\right) = 0 \right. \left., j = k+1,....m \right\}$$

$$S = \left\{ \bar{x} \in R^n \middle| l\left(x_i\right) \le x_i \le u\left(x_i\right), \quad i = 1,2,.....,n \right\}$$

At any point $\bar{x} \in F$, the constraint $g_k(.)$ satisfy $g_k\left(\bar{x}\right) = 0$ are called active constraints at $\bar{x}$. By extension, equality constraints $h_j(.)$ are called active at all points of F Nonlinear equations $h_j\left(\bar{x}\right) = 0$ requires an additional parameter (ψ) to define the precision of the system [19]. All nonlinear equations $h_j\left(\bar{x}\right) = 0$ (for j = k + 1, … , m) are replaced by pair of inequalities: $-\psi \le h_j\left(\bar{x}\right) \le \psi$ so we deal only with nonlinear inequalities.

$$NLPP: \ Min \quad f\left(\bar{x}\right)$$

$$\text{s.t.} \quad F = \left\{ \bar{x} \in R^n \middle| g_i\left(\bar{x}\right) \le 0, i = 1,2,.....,m \right\}$$

$$S = \left\{ \bar{x} \in R^n \middle| l\left(x_i\right) \le x_i \le u\left(x_i\right), \quad i = 1,2,.....,n \right\}$$

3- Unit commitmnet proble UCP

In the UCP under consideration, one is interested in a solution that minimizes the total operating cost of the generating units during the scheduling time horizon while several constraints are satisfied [3,20,21]

3.1 The objective function

The overall objective function of the UCP of N generating units of a scheduling time horizon T is

$$F_T = \sum_{t=1}^{T} \sum_{i=1}^{N} \left(U_{it} F_{it}(P_{it}) + V_{it} S_{it} \right) \$$$

Where

U_{it} : is status of unit i at hour t (ON=1, Off=0).

V_{it} : is start-up/ shut-down status of unit i at hour t.

P_{it} : is the output power of unit i at hour t.

The production cost, $F_{it}(P_{it})$, of a committed unit i, is conventionally taken in a quadratic form:

$$F_{it}(P_{it}) = A_i P^2_{\ it} + B_i P_{it} + C_i \ \$/\text{Hr}$$

Where A_i, B_i, C_i are the cost function parameter of unit i.

The start-up cost S_{it}, is a function of the down time of unit i

$$S_{it} = So_i[1 - D_i \exp(-T_{off_i} / T_{down_i})] + E_i \$$$

Where, So_i : is unit i cold start-up cost, and

D_i : is the cold start-up coefficients of unit i.

E_i : is the hot start-up coefficients of unit i.

3.2. The constraints

The constraints that have been taken into consideration in this work, may be classified into three main groups:

- **Load demand constraints:**

$$\sum_{i=1}^{N} U_{it} P_{it} = PD_t, \forall t$$

Where PD_t : is the system peak demand at hour t (MW)

- **Unit Constraints:**

The constraints on the generating units are

- **Generation limits**

$$U_{it}P_{\min_i} \le P_{it} \le P\max_i U_{it},\ \forall i,t$$

Where $U_{it}P_{\min_i}, P\max_i$ is minimum and maximum generation limit (MW) of unit i.

- **Minimum up/down time**

$$T_{off_i} \ge T_{down_i},\quad T_{on_i} \ge T_{up_i}$$

Where T_{up_i}, T_{down_i} are units i minimum up/down time.

T_{on_i}, T_{off_i} are time periods which unit i is continuously ON/OFF

- **Unit initial status.**
- **Crew constraints.**
- **Unit availability.**
- **Unit derating**
- ❖ **Spinning Reverse**

Spinning reserve, is the total amount of generation capacity available from all units synchronized (spinning) in the system minus the present load demand.

$$\sum_{i=1}^{N} U_{it}P_{\max_i} \ge (PD_t + R_t), \forall t$$

4- GENETIC ALGORITHM (GA)

The discovery of genetic algorithms (GA) was dated to the 1960s by Holland and further described by Goldberg [22]. The GAs have been applied successfully to problems in many fields such as optimization design, fuzzy logic control, neural networks, expert systems, scheduling, and many others [23,24,25]. For a specific problem, the GA codes a solution as an individual chromosome. It then defines an initial population of those individuals that represent a part of the solution space of the problem. The search space therefore, is defined as the solution space in which each feasible solution is represented by a distinct chromosome. Before the search starts, a set of chromosomes is randomly chosen from the search space to form the initial population. Next, through computations the individuals are selected in a competitive manner, based on their fitness as measured by a specific objective function.

The genetic search operators such as selection, mutation and crossover are then applied one after another to obtain a new generation of chromosomes in which the expected quality over all the chromosomes is better than that of the previous generation. This process is repeated until the termination criterion is met, and the best chromosome of the last generation is reported as the final solution. Figure 2 shows The pseudo code of the general GA algorithm .

```
Generate an initial population;
Evaluate fitness of individuals in the population;
Do:
    Select parents from the population;
    Recombine (mate (crossover and mutation operators))
    parents to produce children;
    Evaluate fitness of the children;
    Replace some or all of the population by the children;
While a satisfactory solution has been found.
```

Fig. 2. The pseudo code of the general GA algorithm .

5- QUANTUM COMPUTING

Quantum genetic computing QGA

QGA is based on the concepts of qubits and superposition of states of quantum mechanics. The smallest unit of information stored in a two- state quantum computer is called a quantum bit or qubit [26-30]. A qubit may be in the '1' state , in the '0' state, or in any superposition of the two. The state of a qubit can be represented as

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle \qquad (1)$$

Where and are complex numbers that specify the probability amplitudes of the corresponding states. $|\alpha|^2$ gives the probability that the qubit will be found in '0' state and $|\beta|^2$ gives the probability that the qubit will be found in '1' state. Normalization of the state to unity guarantees

$$|\alpha|^2 + |\beta|^2 = 1 \quad (2)$$

If there is a system of m-qubits, the system can represent 2^m states at the same time. However, in the act of observing a quantum gate, it collapses to a single state [27].

5.1. Representation

It is possible to use a number of different representations to encode the solutions onto chromosomes in evolutionary computation. The classical representations can be broadly classified as: binary, numeric, and symbolic [27]. GQA uses a novel representation that is based on the concept of qubits. One qubit is defined with a pair of complex numbers, (α,β), as

$$\begin{bmatrix}\alpha\\ \beta\end{bmatrix}$$

which is characterized by (1) and (2). And an m-qubits representation is defined as

$$\begin{bmatrix}\alpha_1 \\ \beta_1\end{matrix}\left|\begin{matrix}\alpha_2 \\ \beta_2\end{matrix}\right|\begin{matrix}\ldots \\ \ldots\end{matrix}\quad\left|\begin{matrix}\alpha_m \\ \beta_m\end{bmatrix}, \qquad (3)$$

Where $|\alpha|^2 + |\beta|^2 = 1, i = 1,2,\ldots\ldots,m$. This representation has the advantage that it is able to represent any superposition of states. If there is, for instance, a three-qubits system with three pairs of amplitudes such as

$$\begin{bmatrix}1/\sqrt{2} \\ 1/\sqrt{2}\end{matrix}\left|\begin{matrix}1 \\ 0\end{matrix}\right|\begin{matrix}\ldots \\ \ldots\end{matrix}\quad\left|\begin{matrix}1/2 \\ \sqrt{3}/2\end{bmatrix}, \qquad (4)$$

the state of the system can be represented as

$$\frac{1}{2\sqrt{2}}|000\rangle + \frac{\sqrt{3}}{2\sqrt{2}}|001\rangle + \frac{1}{2\sqrt{2}}|100\rangle + \frac{\sqrt{3}}{2\sqrt{2}}|101\rangle \qquad (5)$$

The above result means that the probabilities to represent the state $|000\rangle$, $|001\rangle$,$|100\rangle$ and $|101\rangle$ are $\frac{1}{8},\frac{3}{8},\frac{1}{8}$ and $\frac{3}{8}$ irespectively. Genetic algorithm with the qubit representation has a better characteristic of diversity than classical approaches, since it can represent superposition of states. Only one qubit chromosome such as (4) is enough to represent four states, but in classical representation at least four chromosomes (000),(001),(100), and (101)) are needed. Convergence can be also obtained with the qubit representation. As $|\alpha_i|^2$ or $|\beta_i|^2$ approaches to 1 1 or 0. the qubit chromosome converges to a single state and the property of diversity disappears gradually. That is, the qubit representation is able to possess the two characteristics of exploration and exploitation, simultaneously.

5.2. Selection operator

The widely used selection operator is proportional selection[18]. In this work, a rank-based selection is designed. In particular, all individuals of the population are first ordered from the best to the worst, then the top partial of individuals which predetermined by the analysit (e.g., N/5), are copied and the same size of the bottom (e.g., N/5), are discarded to maintain the size of population, N. In such away, good individuals also have more chance to be survive or to perform evolution.

5.3. Crossover operation

One point crossover is implemented for Q-bit, which is illustrated as follows. In particular, one crossover position is randomly determined (e.g. position *i*), and then the Q-bits of the parents before position *i* are reserved while the Q-bits after position i are exchanged, which shown in figure 3.

$$\downarrow$$

$$\left.\begin{matrix}\begin{bmatrix}\alpha_{1,1} & \alpha_{1,2} & \cdots & \alpha_{1,i} & \alpha_{1,i+1} & \cdots & \alpha_{1,m}\\ \beta_{1,1} & \beta_{1,2} & \cdots & \beta_{1,i} & \beta_{1,i+1} & \cdots & \beta_{1,m}\end{bmatrix}\\ \begin{bmatrix}\alpha_{2,1} & \alpha_{2,2} & \cdots & \alpha_{2,i} & \alpha_{2,i+1} & \cdots & \alpha_{2,m}\\ \beta_{2,1} & \beta_{2,2} & \cdots & \beta_{2,i} & \beta_{2,i+1} & \cdots & \beta_{2,m}\end{bmatrix}\end{matrix}\right\} \Rightarrow \left.\begin{matrix}\begin{bmatrix}\alpha_{1,1} & \alpha_{1,2} & \cdots & \alpha_{1,i} & \alpha_{2,i+1} & \cdots & \alpha_{2,m}\\ \beta_{1,1} & \beta_{1,2} & \cdots & \beta_{1,i} & \beta_{2,i+1} & \cdots & \beta_{2,m}\end{bmatrix}\\ \begin{bmatrix}\alpha_{2,1} & \alpha_{2,2} & \cdots & \alpha_{12i} & \alpha_{1,i+1} & \cdots & \alpha_{1,m}\\ \beta_{2,1} & \beta_{2,2} & \cdots & \beta_{2,i} & \beta_{1,i+1} & \cdots & \beta_{1,m}\end{bmatrix}\end{matrix}\right\}$$

Fig.3. Crossover operator

5.4. Mutation operator

Mutation operator is done by randomly one position is selected (e.g. position i), and then the corresponding α_i and β_i are exchanged, which shown in figure 4.

$$\downarrow$$

$$\begin{bmatrix}\alpha_{1,1} & \alpha_{1,2} & \cdots & \alpha_{1,i} & \alpha_{1,i+1} & \cdots & \alpha_{1,m}\\ \beta_{1,1} & \beta_{1,2} & \cdots & \beta_{1,i} & \beta_{1,i+1} & \cdots & \beta_{1,m}\end{bmatrix} \Rightarrow \begin{bmatrix}\alpha_{1,1} & \alpha_{1,2} & \cdots & \beta_{1,i} & \alpha_{1,i+1} & \cdots & \alpha_{1,m}\\ \beta_{1,1} & \beta_{1,2} & \cdots & \alpha_{1,i} & \beta_{1,i+1} & \cdots & \beta_{1,m}\end{bmatrix}$$

Fig. 4. Mutaution operator

5.5. Rotation gate for Q-bit

A qubit chromosome Q_j is updated [27] by using the rotation gate $U(\theta)$

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

In such a way that the i-th qubit value (α_i, β_i) is updated as

$$\begin{bmatrix}\alpha_i'\\ \beta_i'\end{bmatrix} = \begin{bmatrix}\cos(\theta) & -\sin(\theta)\\ \sin(\theta) & \cos(\theta)\end{bmatrix}\begin{bmatrix}\alpha_i\\ \beta_i\end{bmatrix}$$

Where θ_i is the rotation angle $\theta_i = s(\alpha_i, \beta_i)\Delta\theta_i$ where $s(\alpha_i, \beta_i)$ is the sign of θ_i that determines the direction , $\Delta\theta_i$ is the magnitude of rotation angle whose lookup table is shown in table 1 . In table1 b_i and x_i are the *i-th* bits of the best solution B and the binary solution R respectively.

The value of $\Delta\theta_i$ has an effect on the speed of convergence, but if it is too big the solution may diverge to alocal optimum. The sign $s(\alpha_i, \beta_i)$ determines the direction of convergence to a global optimum. the lookup table can be used as strategy for convergence this update procedure can be described as follows.

5.6. Evaluation

Binary string X with length m is firstly constructed according to the probability amplitudes of individual p with Q-bit representation s follows:

For every bit $x_i (i = 1, 2, ...m)$ of the string X first generate a random number $\eta \in [0,1]$. The pseudo code of Evaluation algorithm are declared in figue 5.

Procedure make ($X = \{x_i : (i = 1, 2, ...m)\}$)
Begin
$i \leftarrow 0$
While $(i < m)$ do
$i \leftarrow i + 1$
generate a random number $\eta \in [0,1]$
If $\eta > |\alpha_i|^2$
Then $x_i \leftarrow 1$
Else $x_i \leftarrow 0$
End
End

Fig. 5. The pseudo code of Evaluation algorithm.

5.7. Repair procedure

Constraint handing techniques for evolutionary algorithms can be grouped into few categories [31]. One way is to generate solutions without considering the constraints then penalize them in the fitness function, this method have been used in many previous published work. Another category is based on the application of special repair algorithm to correct any infeasible solution so generated. The third category concentrates on the use of special mapping (decodes) which guarantee the generation of feasible solution or the use of problem specific operator s which preserve the feasibility of the solution.

6-QUANTUM GENETIC ALGORITHM FOR UCP

6.1. Solution coding

The solution of UCP is represented by binary matrix (U) of dimension $T \times N$, where N is the generating units and T is the scheduling time horizon [20,21]. Figure 6 illustrate the binary solution matrix

Hours	Units						
	1	2	3	4	..	..	N

ISSN 2229-5518

1	**1**	**0**	**1**	**1**	**...**	**..**	**1**
2	**1**	**0**	**1**	**1**	**..**	**..**	**1**
4	**0**	**1**	**0**	**1**	**..**	**..**	**1**
	..	**..**	**..**	**..**	**..**	**..**	**..**
.	**..**	**..**	**..**	**..**	**..**	**..**	**..**
	..	**..**	**..**	**..**	**..**	**..**	**..**
T	**0**	**0**	**1**	**1**	**...**	**..**	**1**

Fig.6. The Candidate solution matrix

This matrix is represented using Q-bit as follows in figure 7.

$$\left\{ \begin{array}{c} \left[\begin{array}{c|c|c|c} \alpha_{1,1} & \alpha_{1,2} & \cdots\cdots\cdots & \alpha_{1,N} \\ \beta_{1,1} & \beta_{1,2} & \cdots\cdots\cdots & \beta_{1,N} \end{array}\right] \\ \left[\begin{array}{c|c|c|c} \alpha_{2,1} & \alpha_{2,2} & \cdots\cdots\cdots & \alpha_{2,N} \\ \beta_{2,1} & \beta_{2,2} & \cdots\cdots\cdots & \beta_{2,N} \end{array}\right] \\ \cdots\cdots \\ \cdots\cdots \\ \left[\begin{array}{c|c|c|c} \alpha_{T,1} & \alpha_{T,2} & \cdots\cdots\cdots & \alpha_{T,N} \\ \beta_{T,1} & \beta_{T,2} & \cdots\cdots\cdots & \beta_{T,N} \end{array}\right] \end{array} \right\}$$

Fig.7. The Candidate solution representation using Q-bit.

The Candidate solution representation using Q-bit is transformed using evalution process to the following matrix of dimesion $T \times N$ as follows :

$$\begin{bmatrix} 1 & 0 & 1 & \ldots & \ldots & 0 \\ 1 & 0 & 0 & .. & .. & 0 \\ 1 & 1 & 0 & .. & .. & 1 \\ \ldots & \ldots & \ldots & \ldots & .. & 0 \\ \ldots. & \ldots & \ldots & \ldots & .. & 0 \\ 0 & 1 & 0 & .. & .. & 0 \end{bmatrix}$$

6.2. Fitness function: The fitness is taken as the reciprocal of the total operating cost without any penalty terms, the reason is that repair procedure were applied to deal with unfeasible individuals.

6.3.Selection : The widely used selection operator is proportional selection. In this work, a rank-based selection is designed. In particular, all individuals of the population are first ordered from the best to the worst, then the top N/10 individuals are copied and the bottom N/10 are discarded to maintain the size of population, N. In such away, good individuals also have more chance to be reserved or to perform evolution.

6.4. Crossover operation : One point crossover is implemented for Q-bit, which is illustrated as follows. In particular, one crossover position is randomly determined (e.g. position *i*), and then the Q-bits of the parents before position *i* are reserved while the Q-bits after position i are exchanged.

6.5.Mutation operator: Mutation operator is done by randomly one position is selected (e.g. position i), and then the corresponding α_i and β_i are exchanged.

6.6.Update procedure: The following pseudo in figure 8, is the pseudo code of the upading procedure

Procedure update
($Q_j = \{q_i = (\alpha_i, \beta_i) : (i = 1, 2, \ldots m)\}, j = 1, 2, \ldots N$)
Begin
$i \leftarrow 0$
While $(i < m)$ do
$i \leftarrow i + 1$
Determine θ_i with the lookup table
Obtain (α_i', β_i') as
$[\alpha_i', \beta_i']^T = U(\theta_i)[\alpha_i, \beta_i]^T$
End
$q \leftarrow q'$
End

Fig.8. The pseudo of update procedure:

6.7. Repairing Infeasible Individuals:

The idea of this technique is to separate any feasible individuals in a population from those that are infeasible by repairing infeasible individuals. This approach co-evolves the population of infeasible individuals until they become feasible individuals as in the pseudo code in figure 9.

Procedure Update
($Q_j = \{q_i = (\alpha_i, \beta_i) : (i = 1, 2, \ldots m)\}, j = 1, 2, \ldots N$)
Begin
$i \leftarrow 0$
While (q' is infeasible) do
$i \leftarrow i + 1$
Create random individual z as
Check feasibility
End
$q \leftarrow q'$
End

Fig.9. The pseudo code of repair algorithm

Now all individual are in the feasible space.

6.8.Elitist strategy: Using an elitist strategy to produce a faster convergence of the algorithm to the optimal solution of the problem. The elitist individual represents the more fit point of the population. The use of elitist individual guarantees that the best fitness individual never increase (Minimization problem) from one generation to the next.

6.9. Stopping rule: The algorithm is terminated for either one of the following conditions is satisfied:

- The maximum number of generations is achieved.
- When the genotypes (the genotypes structures) of the population of individuals converges, convergence of the genotype structure occur when all bit positions in all strings are identical. In this case, crossover will have no further effect.

ISSN 2229-5518

x_i	b_i	$f(X) \geq f(B)$	$s(\alpha_i, \beta_i)$				
			$\Delta\theta_i$	$\alpha_i\beta_i > 0$	$\alpha_i\beta_i <$	$\alpha_i = 0$	$\beta_i = 0$
0	0	false	0	0	0	0	0
0	0	true	0	0	0	0	0
0	1	false	0	0	0	0	0
0	1	true	0.05π	-1	+1	±1	0
1	0	false	0.01π	-1	+1	±1	0
1	0	true	0.025π	+1	-1	0	±1
1	1	false	0.005π	+1	-1	0	±1
1	1	true	0.025π	+1	-1	0	±1

Table.1. lookup table is shown, where b_i and x_i are the *i-th* bits of the best solution B and the binary solution R respectively.

	Unit1	**Unit 2**	**Unit 3**	**Unit 4**	**Unit 5**	**Unit 6**	**Unit 7**	**Unit 8**	**Unit 9**	**Unit 10**
$P_{max}(MW)$	455	455	130	130	162	80	85	55	55	55
$P_{min}(MW)$	150	150	20	20	25	20	25	10	10	10
$a(\$/h)$	1000	970	700	680	450	370	480	660	665	670
$b(\$/Mwh)$	16.19	17.26	16.60	16.50	19.70	22.26	27.74	29.92	27.27	27.79
$b(\$/Mw^2h)$	0.00048	0.00031	.002	.002211	.00398	.00712	.0079	.00413	.00222	.00173
Min up time MUT (h)	8	8	5	5	6	3	3	1	1	1
min down time MDT(h)	8	8	5	5	6	3	3	1	1	1
Hot start cost ($)	4500	5000	550	560	900	170	260	30	30	30
Cold start cost ($)	9000	1000	1100	1120	1800	340	520	60	60	60
Cold start Hours (h)	5	5	4	4	4	2	2	0	0	0
Initial status (h)	8	8	-5	-5	-6	-3	-3	-1	-1	-1

Table 2. The data for the 10 unit base system for comparison was taken from

Hour	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
Load (MW)	700	750	850	950	1000	1100	1150	1200	1300	1400	1450	1500
Hour	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**
Load (MW)	1400	1300	1200	1050	1000	1100	1200	1400	1300	1100	900	800

Table 3. Load data

Hours	Unit1	Unit 2	Unit 3	Unit 4	Unit 5	Unit 6	Unit 7	Unit 8	Unit 9	Unit 10
1	1	1	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0	0
3	1	1	0	0	1	0	0	0	0	0
4	1	1	0	1	0	0	0	0	0	0
5	1	1	0	1	1	0	0	0	0	0
6	1	1	1	1	1	0	0	0	0	0
7	1	1	1	1	1	0	0	0	0	0
8	1	1	1	1	1	0	0	0	0	0
9	1	1	1	1	1	1	1	0	0	0
10	1	1	1	1	1	1	1	1	0	0
11	1	1	1	1	1	1	1	1	1	0
12	1	1	1	1	1	1	1	1	1	1
13	1	1	1	1	1	1	1	1	0	0
14	1	1	1	1	1	1	1	0	0	0
15	1	1	1	1	1	0	0	0	0	0
16	1	1	1	1	1	0	0	0	0	0
17	1	1	1	1	1	0	0	0	0	0
18	1	1	1	1	1	0	0	0	0	0
19	1	1	1	1	1	0	0	0	0	0
20	1	1	1	1	1	1	1	1	0	0
21	1	1	1	1	1	1	1	0	0	0
22	1	1	1	1	1	1	1	0	0	0
23	1	1	1	0	0	0	0	0	0	0
24	1	1	0	0	0	0	0	0	0	0

Table 4. Unit commitment schedule

Hours	Unit1	Unit 2	Unit 3	Unit 4	Unit 5	Unit 6	Unit 7	Unit 8	Unit 9	Unit 10
1	453	247	0	0	0	0	0	0	0	0
2	455	295	0	0	0	0	0	0	0	0
3	450	375	0	0	1	0	0	0	0	0
4	455	455	0	1	0	0	0	0	0	0
5	450	370	0	130	50	0	0	0	0	0
6	455	350	130	130	35	0	0	0	0	0
7	455	410	130	130	25	0	0	0	0	0
8	455	455	125	125	40	0	0	0	0	0
9	450	450	120	130	90	30	30	0	0	0
10	455	455	130	130	165	30	25	10	0	0
11	455	450	125	125	165	80	18	20	12	0
12	450	440	127	132	158	70	28	33	28	34
13	455	455	125	125	162	43	25	10	0	0
14	455	450	130	130	95	23	17	0	0	0
15	455	455	130	130	30	0	0	0	0	0
16	420	430	100	90	10	0	0	0	0	0
17	455	455	20	45	25	0	0	0	0	0
18	450	450	45	130	35	0	0	0	0	0
19	440	440	130	130	60	0	0	0	0	0
20	455	455	130	130	160	35	20	15	0	0
21	450	455	130	130	90	25	20	0	0	0
22	455	455	20	100	25	20	25	0	0	0
23	400	400	100	0	0	0	0	0	0	0
24	455	345	0	0	0	0	0	0	0	0

Table 5. Power sharing (MW) of Unit Commitment problem

7. Numerical Computation

In order to validate the proposed approach, an example to ten units were considered, which include 10 generator units with scheduling time horizon of 24 hours. Table 2 contains the detailed data of 10 units test problem. Also table3 gives the houly load demand for the horizon T. Table 4 present the finel schedule of the 24 hours, which givem in table 5, in the form of its mega watt.

8. Conclusion

In this paper, we proposed a new enriched algorithm as applies to the unit commitment problem in electric power systems. The proposed algorithm differes from the other evolutionary algorithms in four respects, First, The UCP solution is encoed using aquantum bit, thau saving computer memaory as well computational time of the algorithm search procedure, where the qibit representation is able to process the two characteristics of exploration and explotation, simultaneosly.

Second the fitness function is constructed only from the total operating cost without including penalty terms, ans this due to implementation of repai algorithm, which evolve each individual until it become feasible. Third, to improve the speed of calculation, the reproduction operator are implemented in adifferent way, using quantum operation forth, the hyperdization of genetic algorithm with comutum computing enriched the search algorithm.

One test problem from the literature are solved. it shown that the proposed algorithm is cablel for gerenting optimal solution

References

[1] CHANDLER, W. G., DANDENO, P. L., GILMN, A. F., and KIRCHMAYER, L. K., Short-Range Operation of a Combined Thermal and Hydroelectric Power System, AIEE Transactions, Vol. 73, No. 3, pp. 1057–1065, 1953.

[2] DILLON, T. S., Integer Programming Approach to the Problem of Optimal Unit Commitment with Probabilistic Reserve Determination, IEEE Transactions on Power Apparatus and Systems, Vol. 97, No. 6, pp. 2154–2164, 1978.

[3] WOOD, A. J., and WOLLENBERG, B. F., Power Generation, Operation, and Control, John Wiley and Sons, New York, NY, 1984.

[4] A. E. Eiben, J. E. Smith, Introduction to Evolutionary Computing, Springer Verlag, 2003.

[5] J. H. Holland, Adaptation in Natural and Artificial Systems, University of Michigan Press, 1975.

[6] I. Rechenberg, Evolutionstrategie: Optimierung Technischer Systeme nach Prinzipien der Biologischen Evolution, Fomman Holzboog Verlag, 1973.

[7] L. J. Fogel, A. J. Owens, M. J. Walsh, Artificial Intelligence through Simulated Evolution, Wiley, New York,1966.

[8] [12] K. V. Price, R. M. Storn, J. A. Lampinen, Differential Evolution: A Practical Approach to Global Optimization, Springer, 2005.

[9] Dionisios K. Dimitroulas, Pavlos S. Georgilakis ,A new memetic algorithm approach for the price based unit commitment problem, Applied Energy, Volume 88, Issue 12, December 2011, Pages 4687-4699

[10] K. Chandrasekaran, S. Hemamalini, Sishaj P. Simon, Narayana Prasad Padhy, Thermal unit commitment using binary/real coded artificial bee colony algorithm, Electric Power Systems Research, Volume 84, Issue 1, March 2012, Pages 109-119.

[11] Dilip Datta, Saptarshi Dutta, A binary-real-coded differential evolution for unit commitment problem, International Journal of Electrical Power & Energy Systems, Volume 42, Issue 1, November 2012, Pages 517-524.

[12] S. Najafi, Y. pourjamal, A New Heuristic Algorithm for Unit Commitment Problem, Energy Procedia, Volume 14, 2012, Pages 2005-2011.

[13] Jiaquan Gao, Jun Wang, A hybrid quantum-inspired immune algorithm for multiobjective optimization, Applied Mathematics and Computation, Volume 217, Issue 9, 1 January 2011, Pages 4754-4770.

[14] Gwo-Ching Liao ,Solve environmental economic dispatch of Smart MicroGrid containing distributed generation system – Using chaotic quantum genetic algorithm , International Journal of Electrical Power & Energy Systems, Volume 43, Issue 1, December 2012, Pages 779-787.

[15] Jia-Chu Lee, Whei-Min Lin, Gwo-Ching Liao, Ta-Peng Tsao , Quantum genetic algorithm for dynamic economic dispatch with valve-point effects and including wind power system , International Journal of Electrical Power & Energy Systems, Volume 33, Issue 2, February 2011, Pages 189-197.

[16] Osman M.S., M.A.Abo-Sinna, and A.A. Mousa " IT-CEMOP: An Iterative Co-evolutionary Algorithm for Multiobjective Optimization Problem with Nonlinear Constraints" Journal of Applied Mathematics & Computation (AMC) 183, pp373-389, (2006).

[17] M.S.Osman , M.A.Abo-Sinna , and A.A. Mousa " A Solution to the Optimal Power Flow Using Genetic Algorithm "Journal of Applied Mathematics & Computation (AMC) vol 155, No. 2, 6 August (2004) pp 391-405.

[18] A.A.Mousa, “Recent trends for solving fuzzy multi-objective programming problems: methodology and application, ISBN 978-3-8484-9205-3, Lambert academic publishing GmbH& Co.kG, Berlin, 2012.

[19] Z. Michalewicz, Genetic algorithm + Data structures =evolution programs, third ed., springer-verlag, (1996).

[20] Mantawy, A.H., Abdel-Magid, Y. L. and Selim, S. Z.,integrating genetic algorithms, tabu search, and simulated annealing for the unit commitment problem, Feb 1998, Power Systems, IEEE Transactions on (Volume:14 , Issue: 3)

[21] Mantawy, A.H., Abdel-Magid, Y. L. and Selim, S. Z.,, A simulated annealing algorithm for unit commitment, Feb 1998, Power Systems, IEEE Transactions on (Volume:13 , Issue: 1)

[22] D.E. Goldberg, Genetic Algorithms in Search, Optimiza-

tion & Machine Learning, Addison-Wesley, Reading, MA, 1989.

[23] M. Azzam and A. A. Mousa, " Using genetic algorithm and topsis technique for multiobjective reactive power compensation", Electric Power Systems Research 80 (2010) 675–681.

[24] Abd Allah A. Galal, Abd Allah A. Mousa, Bekheet N. Al-Matrafi , Ant Colony Optimization Approach Based Genetic Algorithms for Multiobjective Optimal Power Flow Problem under Fuzziness , Applied Mathematics, 2013, 4, 595-603, doi:10.4236/am.2013.44084

[25] A. A. Mousa and Kotb A. Kotb, Hybrid Multiobjective Evolutionary Algorithm Based Technique for Economic Emission Load Dispatch Optimization Problem, Journal of Natural Sciences and Mathematics, Qassim University, Vol. 5, No. 1, PP 9-26 (June 2011/Rajab 1432H.)

[26] Abdel-Khalek, S. , Mousa, A.A., Nofal, T.A., Dynamic properties of wehrl information entropy and wehrl phase distribution for a moving four-level atom, Journal of Russian Laser Research, Volume 33, Issue 6, December 2012, Pages 547-558.

[27] Ling Wang, Fang Tang, Hao Wu,Hybrid genetic algorithm based on quantum computing for numerical optimization and parameter estimation, Applied Mathematics and Computation, Volume 171, Issue 2, 15 December 2005, Pages 1141–1156

[28] T. Hey, ”Quantum computing: an introduction,” Computing & Control Engineering Journal, pp. 105-112,Jun 1999.

[29] A. Narayanan, “Quantum computing for beginners,” in Proceedings of the 1999 Congress on Evolutionary Computation, pp. 2231-2238, Jul 1999.

[30] R. Hinterding, ”Representation, Constraint Satisfaction and the Knapsack Problem,” in Proceedings of the 1999 Congress on Evolutionary Computation, pp. 1286- 1292, Jul 1999.

[31] Michalewicz Z., " A Survey of Constraint Handling Techniques In Evolutionary Computation Methods", proceeding of the 4th Annual Conference on Evolutionary Programming, pp135-155, MIT Press , Cambridge , MA, (1995).