A NEW IMPROVED QUANTUM EVOLUTIONARY ALGORITHM WITH MULTIPLICATIVE UPDATE FUNCTION

Mohsen Zamani[1], Mehdi Alinaghian[1], Mohammad Ali Yazdani[2]

[1]Department of Industrial and Systems Engineering, Isfahan University of Technology, 84156-83111 Isfahan, Iran.

[2]Department of Electrical Engineering, Shahid Bahonar University, Kerman, Iran.

ABSTRACT

The Quantum Evolutionary Algorithm (QEA) is a new subcategory of evolutionary computation in which the principles and concepts of quantum computation are used, and to display the solutions it utilizes a probabilistic structure.Therefore, it causes an increase in the solution space. This algorithm has two major problems: hitchhiking phenomenon and slow convergence speed.In this paper, to solve the problems a multiplicative update function called quantum gate is proposed that in addition to considering the best global solution considers the best solution of each generation. The results of one max and knapsack problems and five famous numerical functions show that the proposed method has a significant advantage compared with the basic algorithm in terms of performance, quality of solutions and convergence speed.

KEYWORD

quantum evolution algorithm, multiplicative update function,one maxproblem, knapsack problem.

1.INTRODUCTION

Evolutionary algorithms (EAs) are principally aStochastic Search and optimization method based on theprinciples of Darwin's theory of evolution[1]. Compared with traditional optimization methods, such as calculus-based methods and gradient-based methods EAs are robust.Most of the methods have the problemof trapping in local optimum whereasEAs solve this problem to some extent[2].In EA sthe parameters must be set so that the balancing between movingtowards the best solution and searching for better solutions is preserved. Focus on the best answer increases the Possibility of trapping in local optimumwhile focus on the searching for better solutions increases thesolution time.

Quantum computing was first introduced by Benioff and Feynman in early 1980s[3]. In recent years‹many efforts on the theory and design of algorithms and quantum computers have progressed.The research iscarried out in order to achieve higher computing power compared with conventional computers.As well asQuantum computing is employed in producing circuits of quantum computers[4].The concepts of quantum computing was successfully used to improve the

DOI:10.5121/ijcsa.2016.6102

efficiency of evolutionary algorithms on traditional PCs and led to the development of quantum evolutionary algorithms (QEA).Han and Kim proposed the QEA to solve the combinatorial optimization problems[5]. QEA uses quantum computing concepts such as quantum bit, linear superposition of states and quantum rotation gate and is the first approach that enables the use of quantum representation instead of conventional binary, numeric or symbolic representations. Platel and colleagues have investigated the weaknesses of QEA and detected The problem of premature convergence, and to solve this problem that caused by the Hitchhiking Phenomenon they proposed versatile quantum-inspired evolutionary algorithm(VQEA)[6]. Hitchhiking is a Phenomenon in which the amount of one of the bits generated by the optimization algorithm is incompatible with the expected value but by the amount of general fitness of global solution، the value of this bit remains unchanged during the run of the algorithm.

An advantage of the VQEA is that the Information obtained about the search space during the evolution of population does not remain at the level of one individual but shared Between members of the population.

Many papers have used QEA in solving combinatorial optimization problems[7-9].

In this paper the Quantum Evolutionary Algorithm with multiplicative Update function is presented where the value of Q-bits updated by a multiplicative function. The inputs of function are: present solution, the global best solution and the best solution to each generation and the output is rotation angle.This paper is organized as follows.Section 2 describes the Quantum evolutionary algorithms. Section 3 explains the proposed algorithm. Section 4 summarizes the results. Finally, concluding remarks follow in Section 5.

2.QUANTUM EVOLUTIONARY ALGORITHM

2.1.Concepts

QEA uses quantum computing concepts such as quantum bit, linear superposition of states and quantum rotation gate and is the first approach that enables the use of quantum representation instead of conventional binary, numeric or symbolic representations. Concepts of QEA are explained in the following.

- Quantum bit: Quantum bit or Q-bit is defined as the smallest unit of data storage which can be in the state of "0", in the state of "1" or a superposition of two states. Each Q-bit is defined by a pair of values $\begin{bmatrix}\alpha\\ \beta\end{bmatrix}$ The α^2 is the probability of state 0 and β^2 is the probability of state 1. Equation (1) expresses the relationship between these two values

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

- Q-bit individual: Q-bit individual is defined as a string of Q-bits. Each Q-bit individual can represent a linear superposition of states, probabilistically. In other words, each individual Q-bit with m bits can represent 2^m state simultaneously.For example,consider the following Q-bit individual with 3 Q-bits.

$$q = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{3}} \\ \frac{-1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & \frac{\sqrt{2}}{\sqrt{3}} \end{bmatrix} \tag{2}$$

The Q-bit individual is able to represent the eight states “000”،” 001” ،” 010” ،” 011” ،” 100” ،” 101” ،” 110” ،” 111” with the possibilities of 1/12،1/6،1/12،1/6،1/12،1/6،1/12،1/6 Respectively.

Each Q-bit individual considered as a distribution of all possible solutions in the search space.

The Q-bit representation has a better characteristic of generating diversity in population than any other representations like Binary.

- Quantum gate: In a QEA, Q-gate updates the values of Q-bits. Using Q-gate operator on each Q-bit simulates the evolutionary processes in the population.After each update by Q-gate, Q-bit values must satisfy relationship (1). To update the values of Q-bits, quantum rotation gate operator must be used in accordance with relationships (3) and (4).

$$U(\Delta\theta) = \begin{bmatrix} cos\Delta\theta & -sin\Delta\theta \\ sin\Delta\theta & cos\Delta\theta \end{bmatrix} \tag{3}$$

In the above relationship, Δθ is the Q-bit angle of rotation to state 0 or state 1. At each iteration, Q-gate operator updates Q-bits in accordance with equation (4).

$$\begin{bmatrix} \alpha_{i,j}(\mathrm{t+1}) \\ \beta_{i,j}(\mathrm{t+1}) \end{bmatrix} = U(\theta) * \begin{bmatrix} \alpha_{i,j}(\mathrm{t}) \\ \beta_{i,j}(\mathrm{t}) \end{bmatrix} \tag{4}$$

In the above relationship, $\alpha_{i,j}$ and $\beta_{i,j}$ are the parameters related to the j-th dimension of i-th Q-bit individual.

2.2.The Structure of Algorithm

According to the definitions above‹QEA pseudo-code is as follows:

```
Procedure QEA
begin
t=0
initialize Q(t)
make P(t) by observing Q(t)
evaluate P(t)
store the best solutions among P(t) into b
while (not termination-condition) do
  begin
  t = t+1
  update Q(t) using Q-gate
  make P(t) by observing the states ofQ(t-1)
  evaluate P(t)
  store the best solutions among P(t) into b
  end
end
```

In the step of "initialize Q(t) ," $\alpha_{i,j}(\mathrm{t})$ and $\beta_{i,j}(\mathrm{t})$ of all $q_{i,j}(\mathrm{t})$ are initialized with $1/\sqrt{2}$ .It means that one Q-bit individual, represents the linear superposition of all the possible states with the same probability.in the algorithm‹ n is the string length of the Q-bit individual, and is determined according to the dimensions of solution, and m is the number of Q-bit individuals that indicates the population.

The next step makes binary solutions in P(0) by observing the states of Q(0), where binary solution of each Q-bit individual is created at generation t=0 according to the $\alpha_{i,j}(\mathrm{t})$ and $\beta_{i,j}(\mathrm{t})$ .onebinary solution ,P_i^0‹ i=1,2,…,n , is a binarystring of length m, which is formed by selectingeither 0 or 1 for each bit using the probability of $q_i(0)$. in each stepto make a solution ‹equation (5) is used as follow:

$$p_{ij} = \begin{cases} 1 & if\ \ rand\,(0,1) \geq \alpha_{ij}^2 \\ 0 & otherwise \end{cases} \tag{5}$$

In this step Each binary solution X_i^0 is evaluated to give a measure of its fitness and the initial best solutions are then selected among the binary solutions P(0), and stored into until the termination condition is satisfied, QEA is running in the **while** loop as follows:

- In the **while** loop, binary solutions in p(t) are formed by observing the states of Q(t-1) According to equation (6).
- 2.After generation of new solutions،each binary solution is evaluated for the fitness value.
- 3.Q-bit individuals in Q(t) are updated by Q-gates so that the updated Q-bit should satisfy the equation(1). equation(2),(3) is used as a basic Q-gate in QEA where $\Delta\theta$ is the rotation angle of each Q-bit that can be obtained from Table (1)[10].

Table 1: Rotation angle in QEA algorithm

p_i	b_i	$F(p) \geq F(b)$	$\Delta\theta_{ij}^{t}$	$s(\alpha_{ij}^t, \beta_{ij}^t)$	
				$\alpha_{ij}^t, \beta_{ij}^t \geq 0$	$\alpha_{ij}^t, \beta_{ij}^t < 0$
0	0	false	0	±1	±1
0	0	true	0	±1	±1
0	1	false	0.01π	$+1$	-1
0	1	true	0	±1	±1
1	0	false	0.01π	-1	$+1$
1	0	true	0	±1	±1
1	1	false	0	±1	±1
1	1	true	0	±1	±1

Parameters presented in Table (1) are specifically for a minimization problem. In this table, p_i and b_i are the i-th dimension of current answer and best answer respectively; Δθis the rotation angle and s denotes the angle. In the basic quantum algorithm, α and β can take any quantity within the unit circle 4.The best solution is stored in b.

2.3.The Proposed Algorithm

The classical QEA described in the previous section.The result of researches on QEA shows that this algorithm has the premature convergence problem caused by Hitchhiking phenomenon. This phenomenon occurs when the amount of one of the bits generated by the optimization algorithm is incompatible with the expected value But by the amount of general fitness of global solution، the value of this bit remains unchanged during the implementation(execution) of the algorithm, it means that The bit of interest(concerned bit)، take a hitchhiker from other bits of global solution to solve this problem and improve the QEA، some solutions have been proposed, and in the following the description of each one is provided.

- Generate multiple solution by observing every individual

some changes have been made For greater diversity of solutions Such that each Q-bit individual $q_i(\mathrm{t}) \in \mathrm{Q(t)}$ generates L solution.Then the best solution is stored in the Present solution $c_i(\mathrm{t})$. in each generation the best individual (solution) is selected and stored as the current best solution in $\mathrm{z(t)}$, also the best global solution of b in each generation is updated.

- The use of current best solution to each generation

In this paper, to solve the problem of Hitchhiking phenomenon in the Q-bits update step، In addition to the global best solution، current best solution in each iteration is considered.To update Q(t) ، each Q-bit individual is updated According to (3) , (4) .The rotation angle is calculated by the formula provided below.

$$\Delta\theta_{ij}(t)=\gamma_1[\left(b_j-c_{ij}(t)\right)+\alpha\left(b_j.c_{ij}(t)\right)-\alpha\left(1-b_j\right).\left(1-c_{ij}(t)\right)] \quad (6)$$
$$-\gamma_2[\left(z_j(t)-c_{ij}(t)\right)+\alpha\left(z_j(t).c_{ij}(t)\right)-\alpha\left(1-z_j(t)\right).\left(1-c_{ij}(t)\right)]$$

The simplified equation is achieved as

$$\Delta\theta_{ij}(t)=\gamma_1\left[(\alpha+1)b_j+(\alpha-1)c_{ij}(t)-\alpha\right]+\gamma_2\left[(\alpha+1)z_j(t)+(\alpha-1)c_{ij}(t)-\alpha\right] \quad (7)$$

In equation (7)، γ_1 and γ_2 are two positive Coefficient that shows Confidence measure in global best solution and best solution of each generation respectively.To speed up the convergence,the algorithm designed in such a way that the bits that are similar to optimal global and current solutions bits، converge to their value faster than other bits. To consider this issue, the coefficient α that is a positive factor and greater than 1 is determined.equation (7) is defined so that The positive values of Δθ increase the probability of 1 and negative values increase the probability of 0.

It should be mentioned that in proposed Algorithm α and β take only the positive values.According to Figure (1)،the α , β range is only in the first quarter circle trigonometryThe technique presented in [4] has been used to avoid premature convergence, so the Q-bits cannot be closer to 0 or 1 than a specified ε value.

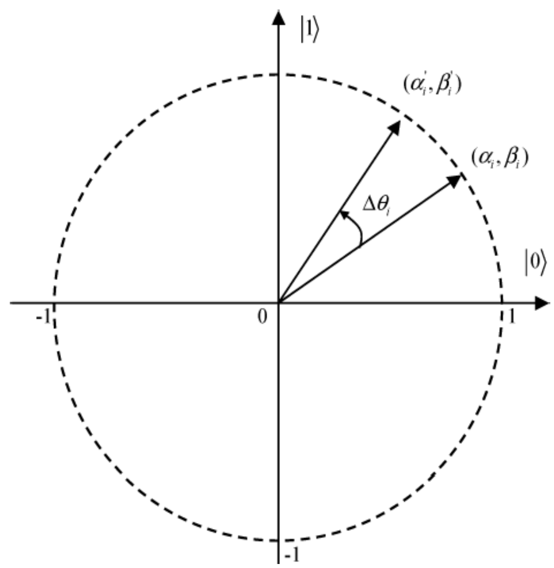

Figure1:range of rotation Angle

The proposed algorithm pseudo-code is given below:

procedure QEA
$t \leftarrow 0$
Initialize population $Q(t)$
while (not termination condition) do
$t \leftarrow t + 1$
make $P(t)$ by observing $Q(t)$
for each Q-bit individual $q_i(t)$ do
Run l times by observing $q_i(t)$
$c_i(t) \leftarrow$ best solution among l of times
end for
evaluate $P(t)$
$z(t) \leftarrow$ best solution among $P(t)$
Update b
Update $Q(t)$ using Q-gate by $P(t)$, $z(t)$, b, $c_i(t)$
end while
end procedure

3.EXPERIMENTS AND NUMERICAL RESULT

To demonstrate the performance and behavior of the proposed algorithm،one max and knapsack Problems and five other Numerical Functions are utilized.

3.1.Parameter Adjustment

Taguchi is a Powerful statistical method that is used to set parameters. In this method Factors are divided into two categories: controllable factors and Stochastic factors. Stochastic factors have inevitable effects.so Efforts are made tominimize the impact of these factors. controllable factors are determined at such levels that the method efficiency is at its maximum and its stability is preserved.In taguchi،instead of the value of solution، The ratio of the signal to noise S / N is used to examine the solution. In This ratio S is the utility (objective function value) and N is Undesirability (The standard deviation of the objective function values)[11].As a result، the aim is to increase the value of this ratio as possible.The proposed algorithm has 3 parameters that must be known before implementation.Each of the parameters are investigated at three levels.Table (2) shows the parameters and levels of assessment.

Table2: parameters and levels of assessment

value	level	parameter
$0.2*\pi$	1	
$0.25*\pi$	2	γ_1
$0.3*\pi$	3	
$0.1*\pi$	1	
$0.15*\pi$	2	γ_2
$0.2*\pi$	3	
1	1	
1.3	2	α
1.5	3	

Data analysis was performed by Minitab software. According to the number of selective factors and levels for the analysis، Standard orthogonal table L9(3*3)was selected.Five problems of one max and Five problems of knapsack were randomly chosen in order to investigate each of the vectors.each problem accordingto the intended values for parameters in each row of the Taguchi table that related to the Essence of the problem, was performed three times. the amount of error was considered as the criterion of solution to solve the problems.Then based on Taguchi performance in Minitab, in each problem for each parameter, the level that had the highest S / N ratio was selected as the best level for that parameter.Table (3) shows the value of levels selected for parameters.

Table3: the final parameters of the problem

parameter	value
γ_1	0.2*π
γ_2	0.15*π
α	1.3
l	0.05*n
t_{max}	0.3*n
pupsize	0.1*n

N is the dimension of solution

3.2.One Max Problem

In this problem a bit string of length k is Defined.The goal is to maximize the number of ones in bit string and the global Optimum value is k at x=111…1. To study the behavior of algorithm the method introduced in [4] is used. Figure (2) shows the Q-bits evolution of the individuals in the proposed algorithm. In the figure، the Color tones display the Square average value of β ($\bar{\beta}^2$). The black color Refers to $(\bar{\beta})^2 = 0$ and White Refers to $(\bar{\beta})^2 = 1$. With the Departure of Q-bits to the global optimum، the Q-bits color Varies from gray to white.

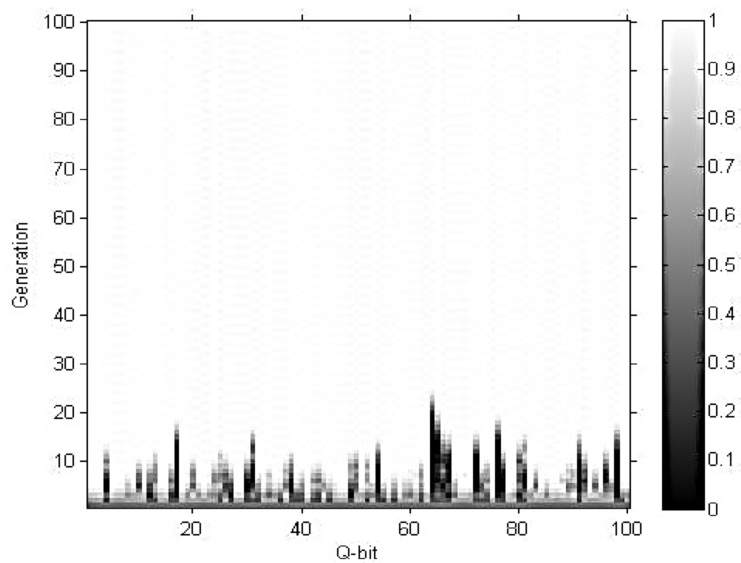

Figure2: the evolution of Q-bits

Figure (3) shows the best solution of each iteration (generation). As seen in Figure 3، The proposed algorithm can find the optimal solution faster than QEA for k = 100. On the other hand, in running step، sometimes the QEA converges to non-Optimum solutions. But this problem does not exist in the proposed algorithm

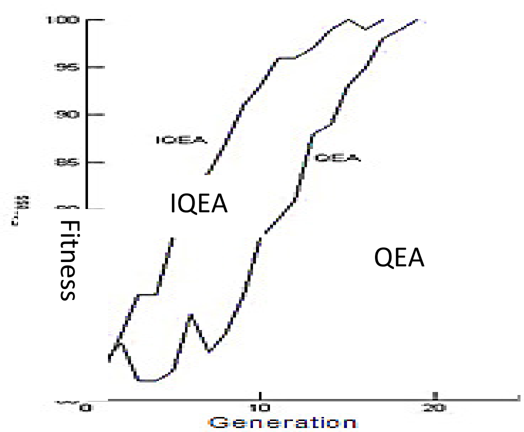

Figure3:Diagram of the best solution to each iteration (The horizontal axis is iteration number and The vertical axis is the best solution to each generation)

Table (4)shows the numerical results of experiments on one max problem in proposed and classic algorithm. Each experiment has been performed 10 times, the best and worst solution, and the average of solutions are given in table below

Table4: the numerical results of one max

		QEA				IQEA			
n	best solution	solution		gap%	time	solution		gap	time
100	100	best	100	0	2.13	best	100	0	1.9
		mean	100			mean	100		
		worst	100			worst	100		
250	250	best	250	0	7.1	best	250	0	6.8
		mean	249.5			mean	250		
		worst	249			worst	250		
350	350	best	350	0	16.2	best	350	0	15.14
		mean	348.12			mean	350		
		worst	344			worst	350		
500	500	best	496	0.8	39.9	best	500	0	38.2
		mean	494			mean	498		
		worst	491			worst	496		
650	650	best	639	1.6	52.3	best	650	0	49.7
		mean	630			mean	650		
		worst	622			worst	650		

3.3.Knapsack Problem

Knapsack problem which is a well-known combinatorial optimization problem is included in a class of NP-hard problems [12]. This problem is used in some papers for assessment. In this problem some objects are given by specified weight and value. The most famous kind of this problem is the 0–1 knapsack problem.Ifx_i=1 , the *i*th item is selected for the knapsack. The knapsack problem can be described as selecting asubset of items from among various items so that it is most profitable, given that the knapsack has limited capacity. Since there are n objects so 2^n possible combinations of objects can be built. The 0–1 knapsack problem is described as follows:

$$\max\left(\sum_{i=1}^{n} v_i x_i\right) \tag{8}$$

$$\textbf{Subject to :} \sum_{i=1}^{n} w_i x_i \le W \tag{9}$$

Consider the knapsack problem with 20 objects. The weight vector and its value is as follows:

$$w = (1,2,\ldots,20), \qquad v = (20,19,\ldots,1)$$

It is obvious because of ascending of weight vector and descending of the value vector,the optimal solution of objects choice, starts from object 1 and then object 2, and continue until the maximum weight limit allows. it is assumed to W=55. As a result, the optimal solution of choice of 10 first objects is f=20+19+…+11=155. After the implementation of the proposed algorithm the evolution of Q-bits
is shown in Figure (4).

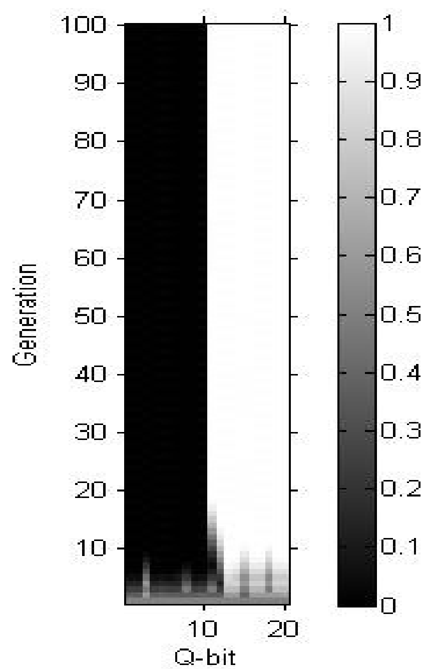

Figure4: the evolution of Q-bits

As seen in figure (4), Q-bits find their optimum value quickly. 10 first objects that their total weight is 55 have the highest worth and goes toward 1 at early generations. The diagram of global best solution versus iteration (generation) is shown in Figure (5).

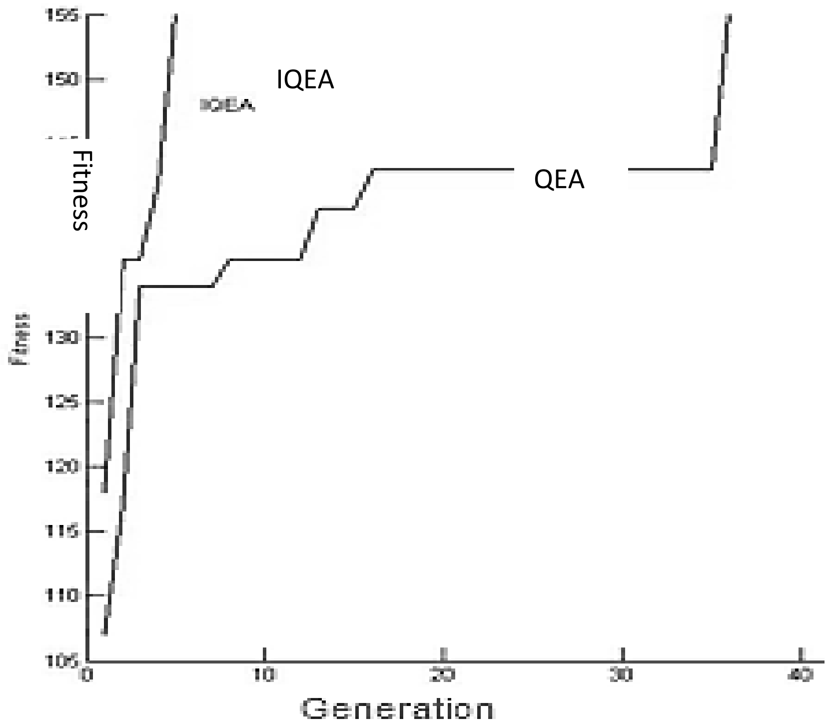

Figure5:Global best solution (the horizontal axis represents the number of generations and the vertical axis represents the global best answer)

As can be seen the proposed algorithm converges to the optimum solution with a higher speed. The algorithm has been implemented on several knapsack problems,weight، worth and capacity matrices are defined as following:

$$w_i = Uniformly\ Random\ [1,10], \tag{10}$$

$$v_i = w_i + 5 \tag{11}$$

$$C = \frac{1}{2}\sum w_i \tag{12}$$

The experiment conducted on various N and results can be observed in Table (5).Each experiment is performed 10 times. The best and worth solution and the average of solutions are given in Table (5).

Table 5:.results of Knapsack problem

knapsack problem		QEA				IQEA			
N	best solution	solution		gap%	time(s)	solution		gap	time
40	240	best	240	0	1.84	best	240	0	0.96
		mean	236			mean	240		
		worst	230			worst	240		
80	493	best	484	1.8	2.8	best	493	0	1.7
		mean	465			mean	489		
		worst	444			worst	474		
100	602	best	589	2.1	2.5	best	597	0.8	2.1
		mean	574			mean	592		
		worst	544			worst	589		
150	932	best	891	4.4	25	best	921	1.1	17.2
		mean	860			mean	350		
		worst	831			worst	863		
250	1522	best	1478	2.9	24.3	best	1497	1.6	22.8
		mean	1450			mean	1773		
		worst	1410			worst	1458		

Five Numerical functions such as Sphere, Ackley, Griewank, Rastrigin, Schwefel, Rosenbrock are applied to evaluate the proposed algorithm. Information on this functions can be seen in Table (6) and Figure (6).

Table 6: Numerical functions

	Function name	Function	Domain	Minimum value
1	Sphere (N=30)	$\sum_{i=1}^{N} x_i^2$	[-100,100]	0
2	Ackley (N=30)	$xp(-0.2\sqrt{\frac{1}{N}\sum_{i=1}^{N} x_i^2}) - \exp(\frac{1}{N}\sum_{i=1}^{N}\cos(2\pi x_i) + 20 + e$	[-32,32]	0

3	Griewank (N=30)	$\frac{1}{4000}\sum_{i=1}^{N} x_i^2 - \prod_{i=1}^{N}\cos(\frac{x_i}{\sqrt{i}}) + 1$	[-600,600]	0
4	Rastrigin (N=30)	$10N + \sum_{i=1}^{N}(x_i^2 - 10\cos(2\pi x_i))$	[-5,5]	0
5	Rosenbrock (N=30)	$\sum_{i=1}^{N-1}(100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$	[-30,30]	0

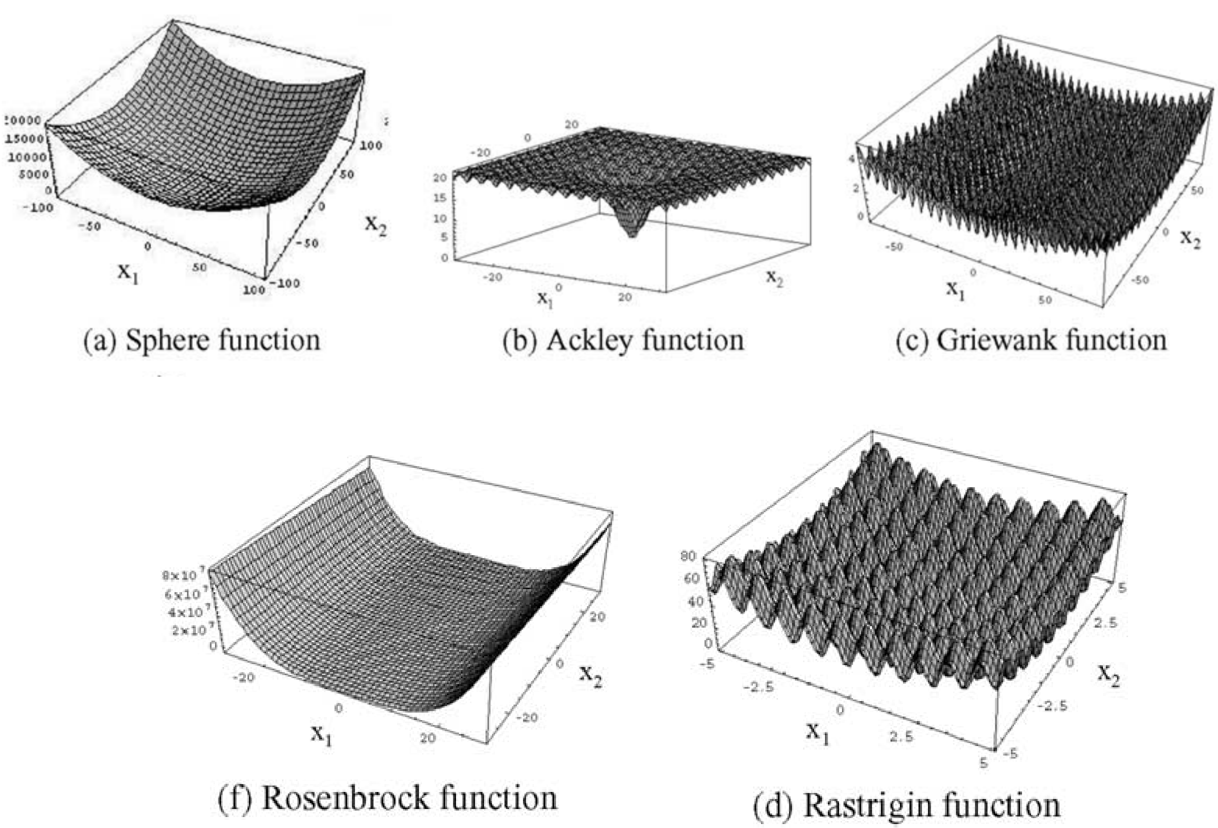

Figure6: Numerical functions diagram

The results of numerical functions can be observed in Table (7). Each experiment is performed 10 time, the best and worth solution and the average of solutions are given.

Table 7: The results of numerical functions

numerical function		QEA		IQEA	
Function	best solution	solution		solution	
Sphere	0	best	35612	best	0
		mean	37254	mean	16
		worst	44081	worst	44
Akley	0	best	0.2	best	0
		mean	1.8	mean	0.04
		worst	4.02	worst	0.1
Rastrigin	0	best	32	best	0
		mean	43	mean	0
		worst	67	worst	0
Griewank	0	best	8.01	best	0
		mean	8.6	mean	1.2
		worst	9.28	worst	1.3
Rosenbrock	0	best	M	best	11
		mean	M	mean	18.5
		worst	M	worst	29

During the experiment the run time(duration of solution) of two algorithms was almost equal، But as shown in Table (7) The proposed algorithm is converged into better solutions. In Table (7), M is a big number (M>10000).

4.CONCLUSION

In this paper the Quantum Evolutionary Algorithm with multiplicative Update function is presented.In this function to update the Q-bits in each generation، the best solution and the global best solution is used.The Q-bit rotation is limited only to the first quarter of trigonometry.it causes the simplicity of the algorithm and the computational complexity is reduced.Experiments on onemax and knapsack problems and five numerical functions prove that this algorithm has high convergence speedand because of using two types of solutions، the Hitchhiking Phenomenon reduces in comparison with the classical algorithm.

5. REFERENCE

[1] Haupt, R. L., & Haupt, S. E. (2004). Practical genetic algorithms. John Wiley & Sons.

[2] Bäck, T., Fogel, D. B., & Michalewicz, Z. (Eds.). (2000). Evolutionary computation 1: Basic algorithms and operators (Vol. 1). CRC Press.

[3] Kaye, P., Laflamme, R., & Mosca, M. (2007). An introduction to quantum computing.

[4] Han, K. H., & Kim, J. H. (2004). Quantum-inspired evolutionary algorithms with a new termination criterion, H ε gate, and two-phase scheme. Evolutionary Computation, IEEE Transactions on, 8(2), 156-169.

[5] Narayanan, A., & Moore, M. (1996, May). Quantum-inspired genetic algorithms. In Evolutionary Computation, 1996., Proceedings of IEEE International Conference on (pp. 61-66). IEEE.

[6] Platel, M. D., Schliebs, S., & Kasabov, N. (2007, September). A versatile quantum-inspired evolutionary algorithm. In Evolutionary Computation, 2007.CEC 2007. IEEE Congress on (pp. 423-430). IEEE.

[7] Mahdabi, P., Jalili, S., & Abadi, M. (2008, July). A multi-start quantum-inspired evolutionary algorithm for solving combinatorial optimization problems. In Proceedings of the 10th annual conference on Genetic and evolutionary computation (pp. 613-614). ACM.

[8] Zhang, R., & Gao, H. (2007, August). Improved quantum evolutionary algorithm for combinatorial optimization problem. In Machine Learning and Cybernetics, 2007 International Conference on (Vol. 6, pp. 3501-3505). IEEE.

[9] da Cruz, A. V. A., Vellasco, M. M., & Pacheco, M. A. C. (2008). Quantum-inspired evolutionary algorithm for numerical optimization. In Quantum inspired intelligent systems (pp. 115-132). Springer Berlin Heidelberg.

[10] Zhang, G. (2011). Quantum-inspired evolutionary algorithms: a survey and empirical study. Journal of Heuristics, 17(3), 303-351.

[11] C Montgomery, D. (1997). Montgomery Design and Analysis of Experiments.

[12] Garey, M. R., & Johnson, D. S. (1979). Computers and intractability: a guide to the theory of NP-completeness. 1979. San Francisco, LA: Freeman.