

# A Local Search Operator in Quantum Evolutionary Algorithm and its Application in Fractal Image Compression

Hossein Mohamadi  
Islamic Azad University  
Azadshahr Branch  
H\_mohamadi@hotmail.com

Ali Nodehi  
Islamic Azad University  
Gorgan Branch  
alinodehi@gorganiau.ir

Mohamad Tayarani  
Islamic Azad University  
Mashhad Branch  
tayarani@ieee.org

**Abstract**—Fractal Image Compression is an optimization problem in the class of NP-Hard problems. Quantum Evolutionary Algorithm is a novel optimization algorithm proposed for class of combinatorial problems like Knapsack problem. While QEA is highly suitable for NP-Hard problems, QEA is not widely used in Fractal Image Compression. In order to improve the performance of QEA in Fractal Image Compression, this paper proposes a local search operator for QEA. The proposed algorithm uses Simulated Annealing algorithm in its search process. The SA is performed on observed possible solution to help the algorithm escaping from local optima. The proposed Simulated Annealing Quantum Evolutionary Algorithm (SAQEA) for fractal image compression is tested on several images like Lena, Pepper and Baboon for several times and is compared with QEA and GA. Experimental results show better performance for the proposed algorithm than QEA and GA and in comparison with full search, the proposed algorithm reaches suitable solutions with much less computation complexity.

**Keywords**- *Fractal Image Compression, Quantum Evolutionary Algorithm, Simulated Annealing.*

## I. INTRODUCTION

QUANTUM Evolutionary Algorithm is a novel optimization algorithm proposed for combinatorial optimization problems. The probabilistic representation of possible solutions in QEA makes the quantum individuals able to represent all the search space simultaneously. Nevertheless, like other evolutionary algorithms, QEA suffers from trapping in local optima. There are many approaches in evolutionary algorithms, trying to maintain the diversity in evolutionary algorithms. In order to maintain the diversity in QEA [1] proposes a diversity preserving operator for QEA which is called DPCQEA. Reference [2] uses a sinusoid sized population for QEA which preserves the diversity of the population and improves the performance of QEA. Reference [3] uses a diversity controlled GA for frequency-response masking (FRM) FIR digital filters over the double base number system multiplier coefficient space. In order to overcome premature convergence in GA, [4] proposes a novel adaptive genetic algorithm based on diversity maintaining.

Several works try to improve the algorithm of fractal image compression using Genetic algorithm. In [5] a new method for finding the IFS code of fractal image is

developed and the influence of mutation and the crossover is discussed. The low speed of fractal image compression blocks its way to practical application. In [6] a genetic algorithm approach is used to improve the speed of searching in fractal image compression. A new method for genetic fractal image compression based on an elitist model is proposed in [7]. In the proposed approach the search space for finding the best self similarity is greatly decreased. Reference [8] makes an improvement on the fractal image coding algorithm by applying genetic algorithm. Many researches increase the speed of fractal image compression but the quality of the image will decrease. In [9] the speed of fractal image compression is improved without significant loss of image quality. Reference [10] proposes a genetic algorithm approach which increases the speed of the fractal image compression without decreasing of the quality of the image. In the proposed approach a standard Barnsley algorithm, the Y. Fisher based in classification and the genetic compression algorithm with quad-tree partitioning are compared. In GA based algorithm a population of transformations is evolved for each range block. In order to prevent the premature convergence of GA in fractal image compression a new approach is proposed in [11], which controls the parameters of GA adaptively. A spatial correlation genetic algorithm is proposed in [12], which speeds up the fractal image compression algorithm. In the proposed algorithm there are two stages, first the spatial correlations in image for both the domain pool and the range pool is performed to exploit local optima. In the second stage if the local optima were not certifiable, the whole of image is searched to find the best self similarity. A schema genetic algorithm for fractal image compression is proposed in [13,17] to find the best self similarity in fractal image compression.

In order to help QEA escaping from local optima, this paper proposes a local search algorithm for QEA. In the proposed algorithm, after convergence of q-individuals, when the algorithm is trapped in local optima, a Simulated Annealing algorithm is performed on possible solution, to help the algorithm escaping from local optima. The SA algorithm is performed on all the possible solutions, until the algorithm escapes from the local optima. Then QEA starts its searches until its convergence. The proposed algorithm is used in fractal image compression and several experimental

results show better performance for the proposed algorithm than QEA and GA.

This paper is organized as follows: Section II discusses about QEA, in Section III the SAQEA is proposed. In Section IV, the proposed algorithm is applied on Fractal Image Compression problem. Section V discusses about the experimental results and finally section VI concludes the paper.

## II. QEA

QEA is inspired from the principles of quantum computation, and its superposition of states is based on qubits, the smallest unit of information stored in a two-state quantum computer. A qubit could be either in state “0” or “1”, or in any superposition of the two as described below:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \quad (1)$$

Where  $\alpha$  and  $\beta$  are complex number, which denote the corresponding state appearance probability, following below constraint:

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

This probabilistic representation implies that if there is a system of  $m$  qubits, the system can represent  $2^m$  states simultaneously. At each observation, a qubits quantum state collapses to a single state as determined by its corresponding probabilities.

### A. Representation

QEA uses a novel representation based on the above concept of qubits. Consider  $i$ -th individual in  $t$ -th generation defined as an  $m$ -qubit as below:

$$q_i^t = \begin{bmatrix} \alpha_{i1}^t & \alpha_{i2}^t & \dots & \alpha_{im}^t \\ \beta_{i1}^t & \beta_{i2}^t & \dots & \beta_{im}^t \end{bmatrix} \quad (3)$$

Where  $|\alpha_{ij}^t|^2 + |\beta_{ij}^t|^2 = 1, j=1,2,\dots,m$ ,  $m$  is the number of qubits, i.e., the string length of the qubit individual,  $i=1,2,\dots,n$ ,  $n$  is the number of possible solution in population and  $t$  is generation number of the evolution. Since a qubit is a probabilistic representation, any superposition of states is simultaneously represented. If there is, for instance, a three-qubits ( $m=3$ ) individual such as (4):

$$q_i^t = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{3}} & \frac{1}{2} \\ \frac{1}{\sqrt{2}} & \frac{\sqrt{2}}{\sqrt{3}} & \frac{\sqrt{3}}{2} \end{bmatrix} \quad (4)$$

Or alternatively, the possible states of the individual can be represented as:

$$\begin{aligned} q_i^t = & \frac{1}{2\sqrt{6}}|000\rangle + \frac{1}{2\sqrt{2}}|001\rangle + \frac{1}{2\sqrt{3}}|010\rangle + \frac{1}{2}|011\rangle \\ & + \frac{1}{2\sqrt{6}}|100\rangle + \frac{1}{2\sqrt{2}}|101\rangle + \frac{1}{2\sqrt{3}}|110\rangle \\ & + \frac{1}{2}|111\rangle \end{aligned} \quad (5)$$

Note that the square of above numbers are true probabilities, i.e. the above result means that the probabilities to represent the state  $|000\rangle, |001\rangle, |100\rangle, |010\rangle$  are  $1/24, 1/8, 1/24$  and  $1/12$  respectively. Consequently, the three-qubits system of (4) has all eight states information at the same time.

Evolutionary computing with the qubit representation has a better characteristic of diversity than classical approaches since it can represent superposition of states. Only one qubit individual such as (4) is enough to represent eight states, whereas in classical representation eight individuals are needed. Additionally, along with the convergence of the quantum individuals, the diversity will gradually fade away and the algorithm converges.

### B. QEA Structure

In the initialization step of QEA,  $[\alpha_{ij}^t \ \beta_{ij}^t]^T$  of all  $q_i^0$  are initialized with  $1/\sqrt{2}$ . This implies that each qubit individual  $q_i^0$  represents the linear superposition of all possible states with equal probability. The next step makes a set of binary instants;  $x_i^t$  by observing  $Q(t) = \{q_1^t, q_2^t, \dots, q_n^t\}$  states, where  $X(t) = \{x_1^t, x_2^t, \dots, x_n^t\}$  at generation  $t$  is a random instant of qubit population. Each binary instant,  $x_i^t$  of length  $m$ , is formed by selecting each bit using the probability of qubit, either  $|\alpha_{ij}^t|^2$  or  $|\beta_{ij}^t|^2$  of  $q_i^t$ . Each instant  $x_i^t$  is evaluated to give some measure of its fitness. The initial best solution  $b = \max_{i=1}^n \{f(x_i^t)\}$  is then selected and stored from among the binary instants of  $X(t)$ . Then, in ‘update  $Q(t)$ ,’ quantum gates  $U$  update this set of qubit individuals  $Q(t)$  as discussed below. This process is repeated in a while loop until convergence is achieved. The appropriate quantum gate is usually designed in accordance with problems under consideration.

The pseudo-code of QEA algorithm is defined as [4]:

```

Procedure QEA
  begin
     $t=0$ 
    1. initialize  $Q(0)$ .
    2. make  $X(0)$  by observing the states of  $Q(0)$ .
    3. evaluate  $X(0)$ .
    4. Store  $X(0)$  into  $B(0)$ . Store the best solution among  $X(0)$  into  $\mathbf{b}$ .
    5. while not termination condition do
      begin
         $t=t+1$ 
        6. make  $X(t)$  by observing the states of  $Q(t-1)$ 
        7. evaluate  $X(t)$ 
        8. update  $Q(t)$  using Q-gates
        9. store the best solutions among  $B(t-1)$  and  $X(t)$  into  $B(t)$ 
        10. store the best solution among  $B(t)$  into  $\mathbf{b}$ 
        11. if global migration condition then migrate  $\mathbf{b}$  to  $B(t)$  globally
        12. else if local migration condition then migrate  $b_k^t$  in  $B(t)$  to  $B(t)$  locally
      end
    end
  
```

QEA has a population of quantum

individuals  $Q(t) = \{q_1^t, q_2^t, \dots, q_n^t\}$ , where  $t$  is generation step and  $n$  is the size of population.

A comprehensive description of QEA can be found in [4]. The QEA procedure is described as:

1. In the initialization step all qubits  $\alpha_{ij}^0$  and  $\beta_{ij}^0$ ,  $i=1,2,\dots,n$  and  $j=1,2,\dots,m$  are initialized with  $1/\sqrt{2}$ . It means the probability of observing "0" and "1" for all qubits is equal.

2. In this step the binary solutions  $X(0) = \{x_1^0, x_2^0, \dots, x_n^0\}$  at generation  $t=0$  are creating by observing  $Q(0)$ . Observing  $x_{ij}^t$  from qubit  $[\alpha_{ij}^t \ \beta_{ij}^t]^T$  is performed as below:

$$x_{ij}^t = \begin{cases} 0 & \text{if } R(0,1) < |\alpha_{ij}^t|^2 \\ 1 & \text{otherwise} \end{cases} \quad (6)$$

Where  $R(\cdot, \cdot)$ , is a uniform random number generator.

3. All solutions in  $X(t)$  are evaluated with fitness function.

4. Store  $X(0)$  into  $B(0)$ . Select best solution among  $X(0)$  and store it to  $\mathbf{b}$ .

5. The while loop is running until termination condition is satisfied. Termination condition can be considered as maximum generation condition or convergence condition.

6. Observing  $X(t)$  from  $Q(t-1)$ .

7. Evaluate  $X(t)$  by fitness function

8. Update  $Q(t)$

9, 10. Store the best solutions among  $B(t-1)$  and  $X(t)$  to  $B(t)$ . If the fittest solution among  $B(t)$  is fitter than  $\mathbf{b}$  then store the best solution into  $\mathbf{b}$ .

11, 12. If global migration condition is satisfied copy  $\mathbf{b}$  to all the solutions in  $B(t)$ . If local migration condition is satisfied replace some of solutions in  $B(t)$  with best one of them.

### C. Quantum Gates Assignment

The common mutation is a random disturbance of each individual, promoting exploration while also slowing convergence. Here, the quantum bit representation can be simply interpreted as a biased mutation operator. Therefore, the current best individual can be used to steer the direction of this mutation operator, which will speed up the convergence. The evolutionary process of quantum individual is completed through the step of "update  $Q(t)$ ." A crossover operator, quantum rotation gate, is described below. Specifically, a qubit individual  $q_i^t$  is updated by using the rotation gate  $U(\theta)$  in this algorithm. The  $j$ -th qubit value of  $i$ -th quantum individual in generation  $t$   $[\alpha_{ij}^t \ \beta_{ij}^t]^T$  is updated as:

$$\begin{bmatrix} \alpha_{ij}^{t+1} \\ \beta_{ij}^{t+1} \end{bmatrix} = \begin{bmatrix} \cos(\Delta\theta) & -\sin(\Delta\theta) \\ \sin(\Delta\theta) & \cos(\Delta\theta) \end{bmatrix} \begin{bmatrix} \alpha_{ij}^t \\ \beta_{ij}^t \end{bmatrix} \quad (7)$$

Where  $\Delta\theta$  is rotation angle and controls the speed of convergence and determined from Table I. Reference [14] shows that these values for  $\Delta\theta$  have better performance.

TABLE I. LOOKUP TABLE OF  $\Delta\theta$ .

$x_i$	$b_i$	$f(x) \geq f(b)$	$\Delta\theta$
0	0	false	0
0	0	true	0
0	1	false	$0.01\pi$
0	1	true	0
1	0	false	$-0.01\pi$
1	0	true	0
1	1	false	0
1	1	true	0

### III. SIMULATED ANNEALING QEA

Evolutionary algorithms suffer from trapping in local optima and early convergence and Quantum Evolutionary Algorithm is not an exception. In QEA, q-individuals search the search space and after some iterations trap in true states of  $[0 \ 1]^T$  or  $[1 \ 0]^T$ . In this condition, q-individuals have not much chance to escape from the local optima. Several works have tried to proposed new methods to help the algorithm searching other parts of the search space. This paper proposes a novel local search algorithm based on Simulated Annealing for Quantum Evolutionary Algorithms. In the proposed algorithm in each iteration, the convergence of the population is checked, and if the population converged, a Simulated Annealing algorithm is performed on possible solutions. The procedure of the proposed algorithm is as follows:

The pseudo-code of SAQEA algorithm is defined as [4]:

```

Procedure SAQEA
  begin
     $t=0$ 
    1. initialize  $Q(0)$ .
    2. make  $X(0)$  by observing the states of  $Q(0)$ .
    3. evaluate  $X(0)$ .
    4. Store  $X(0)$  into  $B(0)$ . Store the best
       solution among  $X(0)$  into  $\mathbf{b}$ .
    5. while not termination condition do
       begin
          $t=t+1$ 
        6. make  $X(t)$  by observing the states of
            $Q(t-1)$ 
        7. evaluate  $X(t)$ 
        8. update  $Q(t)$  using Q-gates
        9. store the best solutions among  $B(t-1)$ 
           and  $X(t)$  into  $B(t)$ 
        10. store the best solution among  $B(t)$ 
            into  $\mathbf{b}$ 
        11. if global migration condition
            then migrate  $\mathbf{b}$  to  $B(t)$  globally
        12. else if local migration condition
  
```

TABLE II. EXPERIMENTAL RESULTS ON LENA, PEPPER AND BABOON

Picture	Method	Population Size	MSE computations	PSNR
Lena	Full Search	-	59,474,944	28.85
	QEA	30	6,144,000	28.49
		25	5,120,000	28.28
		20	4,096,000	27.95
		15	3,072,000	27.43
	SAQEA	30	6,144,000	28.56
		25	5,120,000	28.35
		20	4,096,000	28.16
		15	3,072,000	27.53
	GA	30	6,144,000	28.11
		25	5,120,000	28.04
		20	4,096,000	27.55
		15	3,072,000	27.27
Pepper	Full Search	-	59,474,944	29.85
	QEA	30	6,144,000	29.55
		25	5,120,000	29.09
		20	4,096,000	28.87
		15	3,072,000	28.12
	SAQEA	30	6,144,000	29.63
		25	5,120,000	29.21
		20	4,096,000	28.98
		15	3,072,000	28.51
	GA	30	6,144,000	29.14
		25	5,120,000	28.92
		20	4,096,000	28.64
		15	3,072,000	28.11
Baboon	Full Search	-	59,474,944	20.04
	QEA	30	6,144,000	19.28
		25	5,120,000	19.18
		20	4,096,000	18.95
		15	3,072,000	18.62
	SAQEA	30	6,144,000	19.62
		25	5,120,000	19.31
		20	4,096,000	19.08
		15	3,072,000	18.56
	GA	30	6,144,000	19.17
		25	5,120,000	19.02
		20	4,096,000	18.65
		15	3,072,000	18.41

13. **then** migrate  $b_k^t$  in  $B(t)$  to  $B(t)$  locally
  14. **if** the population is converged perform SA on all the possible solutions until the population leaves the converged status  
**end**
- end**

The proposed SAQEA is like QEA except in line 14. In this step the convergence status of the population is calculated. The convergence of the population is calculated as [16]:

$$C = \frac{1}{n \times m} \sum_{i=1}^m \sum_{j=1}^n \left| 1 - 2|\alpha_{ij}|^2 \right| \quad (8)$$

Where  $C$  is the convergence of the population,  $m$  is the size of population (the number of the q-individuals in the population) and  $n$  is the number of q-bits in the q-individuals (the dimension of the problem).

The population is converged if satisfy the below constraint:

$$C = \frac{1}{n \times m} \sum_{i=1}^m \sum_{j=1}^n \left| 1 - 2|\alpha_{ij}| \right|^2 > \gamma \quad (9)$$

If the population is converged the Simulated Annealing algorithm is performed on all the possible solutions. The SA algorithm is as follows:

**Procedure Local search SA**  
**begin**

1.  $T = T_0$
2.  $t \leftarrow 0$
3. **while** not (termination condition) **do**  
    **begin**
4.     **for** all possible solutions  $x^i$  in  $X(t)$   
        **begin**
5.          $E_1 = \text{evaluation}(x^i)$
6.         perform change( $x^i$ ) and store the result in  $y^i$
7.          $E_2 = \text{evaluation}(y^i)$
8.          $\Delta E = E_2 - E_1$
9.         **if**  $\Delta E > 0$  **then**
10.              $x^i = y^i$
11.         **else**
12.              $p = e^{-\frac{\Delta E}{T}}$
13.         **if**  $p < U(1,0)$  **then**  $x^i = y^i$   
            **end**
14.      $T = \eta \times T$   
        **end**

**end**

The termination condition in step 3, is when the population escapes from convergence status. The termination condition is as follows:

$$C = \frac{1}{n \times m} \sum_{i=1}^m \sum_{j=1}^n \left| 1 - 2|\alpha_{ij}| \right|^2 < \gamma \quad (10)$$

Which means the population has escaped from local optimum.

The step 5 performs a change on possible solutions. The change is like mutation in Genetic Algorithms and is defined as:

**Procedure Change**  
**begin**

1. **for** all alleles  $x^i_{i,k}$  in  $x^i$  **do**
  2.     **if**  $R(0,1) < \text{mutation rate}$
  3.          $x^i_{i,k} = R(l_k, u_k)$
- end**

#### IV. SAQEA FOR FRACTAL IMAGE COMPRESSION

The proposed SAQEA for fractal image compression searches among the domain blocks to find the best domain block and the best transformation for each range block. For each range block, SAQEA searches among all the domain pool to find the best domain block and the best transformation. The coding method for the q-individuals in the proposed method is as below:

$p_x$	$p_y$	$p_t$
-------	-------	-------

In the proposed approach each q-individual, has three parts:  $p_x$  shows the horizontal position of domain block,  $p_y$  shows the vertical position of the domain block and  $p_t$  shows the transformation. The transformations are the 8 ordinary transformations: rotate 0°, 90°, 180°, 270°, flip vertically, horizontally, flip relative to 45°, and relative to 135°. Each part of each solution is a real number and is converted to an integer number before evaluation process.

#### V. EXPERIMENTAL RESULTS

This section experiments the proposed algorithm and compares the proposed algorithm with the performance of GA and QEA in fractal image compression. The proposed algorithm is examined on images Lena, Pepper and Baboon with the size of 256×256 and gray scale. The size of range blocks is considered as 8×8 and the size of domain blocks is considered as 16×16. In order to compare the quality of results, the PSNR test is performed:

$$\text{PSNR} = 10 \times \log \left( \frac{255^2}{\frac{1}{m \times n} \sum_{i=1}^n \sum_{j=1}^m (f(i,j) - g(i,j))^2} \right)$$

Where  $m \times n$  is the size of image.

The crossover rate in GA is 0.8 and the probability of mutation in GA and SAQEA is 0.003 for each allele. The parameter  $\gamma$  in (9) and (10) is considered as 0.9, and  $\eta$  is considered as 0.95. Table II shows the experimental results on the proposed algorithm and GA. The number of iterations for GA, QEA and SAQEA for all the experiments is 200. According to table II the proposed algorithm improves the performance of fractal image compression for all the experimental results.

#### VI. CONCLUSION

This paper proposes a Simulated Annealing QEA for fractal image compression. In evolutionary algorithms during the search process the diversity in the population is decreased, and the algorithm is trapped in the local optima. This is true for QEA which uses a probabilistic representation for the individuals. Here we propose a novel local reach operator for QEA to help the algorithm escaping from the local optima. In the proposed algorithm, after the convergence of population, Simulated Annealing algorithm is performed on possible solutions until algorithm escapes from local optima. The proposed SAQEA has some parameters and this paper finds the best parameters for the proposed algorithm. There are some other questions that should be focused in our future works. The main question is the best parameters of the proposed algorithm. As it is seen the proposed algorithm has some parameters:  $\gamma$  and  $\eta$ . Finding the best parameters for the proposed algorithm will be answered in our future researches. Finally experimental results on Lena, Pepper, and Baboon picture show an improvement on fractal image compression. The time

complexity of the proposed SAQEA is equal to original version of QEA.

#### REFERENCES

- [1] M.-H. Tayarani- N and M.- R. Akbarzadeh-T “A Cellular Structure and Diversity Preserving operator in Quantum Evolutionary Algorithms”. *IEEE World Conference on Computational Intelligence* 2008.
- [2] M.-H. Tayarani- N and M.- R. Akbarzadeh-T “A Sinusoid Size Ring Structure Quantum Evolutionary Algorithm”. *IEEE International Conference on Cybernetics and Intelligent Systems Robotics, Automation and Mechanics* 2008.
- [3] Sai Mohan Kilambi Nowrouzian, B, “A Diversity Controlled Genetic Algorithm for Optimization of FRM Digital Filters over DBNS Multiplier Coefficient Space”. *IEEE International Symposium on Circuits and Systems*, 2007.
- [4] Xiaojun Xing Qiuling Jia Zhigang Ling Dongli Yuan, “A Novel Genetic Algorithm Based on Individual and Gene Diversity Maintaining and it’s Simulation” *IEEE International Conference on Automation and Logistics*, 2007.
- [5] Xi'an Jiaotong Univ “An improved genetic algorithm of solving IFS code of fractal image” *IEEE 3<sup>rd</sup> international conference on signal processing* 1996.
- [6] Xiao Chen, Guang-Xi Zhu, and Yaoting Zhu, “Fractal image coding method based on genetic algorithms” *International Symposium on Multispectral Image Processing* 1998.
- [7] Mitra, S.K. Murthy, C.A. Kundu, M.K “Technique for fractal image compression using genetic algorithm”, *IEEE Trans on Image Processing* Vol 7 no 4 pp 586-593 1998.
- [8] Lu Xun Yu Zhongqiu “The application of GA in fractal image compression”, 3<sup>rd</sup> *IEEE World Congress on Intelligent Control and Automation*, 2000.
- [9] Gafour, A. Faraoun, K. Lehireche, “A Genetic fractal image compression”. *ACS/IEEE International Conference on Computer Systems and Applications*, 2003.
- [10] Faraoun Kamel Mohamed, Boukelif Aoued, “Speeding Up Fractal Image Compression by Genetic Algorithms” *Springer Journal of Multidimension Systems and Signal processing*. Vol 16, No 2, 2005.
- [11] Lifeng Xi, Liangbin Zhang, “A Study of Fractal Image Compression Based on an Improved Genetic Algorithm”, *International Journal of Nonlinear Science* Vol 3, No 2, pp 116-124m 2007.
- [12] Ming-Sheng Wu, Wei-Chih Teng, Jyh-Horng Jeng and Jer-Guang Hsieh, “Spatial correlation genetic algorithm for fractal image compression” *Elsevier Journal of Chaos, Solitons and Fractals*. Volume 28, Issue 2, pp 497-510, 2006.
- [13] Ming-Sheng Wu, Jyh-Horng Jeng, and Jer-Guang Hsieh, “Schema genetic algorithm for fractal image compression” *Elsevier Journal of Engineering Applications of Artificial Intelligence* Vol 20, Issue 4, pp 531-538 2007.
- [14] K. Han, J. Kim, “Quantum-inspired evolutionary algorithm for a class of combinatorial optimization,” *IEEE Trans. on Evolutionary Computation*, vol. 6, no. 6, 2002.
- [15] V. K. Koumoussis, C. P. Katsaras. "A Saw-Tooth Genetic Algorithm Combining the Effects of Variable Population Size and Reinitialization to Enhance Performance," *IEEE Trans. Evol. Comput.* vol. 10, pp. 19–28, 2006.
- [16] Han, J. Kim, “Quantum-Inspired Evolutionary Algorithms with a New Termination Criterion,  $H_c$  Gate, and Two-Phase Scheme,” *IEEE Trans. on Evolutionary Computation*, vol. 8, no. 2, 2004.
- [17] A. Nodehi and M.-H. Tayarani- N and F. Mahmoudi “A Novel Functional Sized Population Quantum Evolutionary Algorithm for Fractal Image Compression ”. *International Conference on Computer* 2009.