

Real-Observation Quantum-Inspired Evolutionary Algorithm for a Class of Numerical Optimization Problems*

Gexiang Zhang and Haina Rong

School of Electrical Engineering, Southwest Jiaotong University,
Chengdu 610031 Sichuan, China
gxzhang@ieee.org

Abstract. This paper proposes a real-observation quantum-inspired evolutionary algorithm (RQEA) to solve a class of globally numerical optimization problems with continuous variables. By introducing a real observation and an evolutionary strategy, suitable for real optimization problems, based on the concept of Q-bit phase, RQEA uses a Q-gate to drive the individuals toward better solutions and eventually toward a single state corresponding to a real number varying between 0 and 1. Experimental results show that RQEA is able to find optimal or close-to-optimal solutions, and is more powerful than conventional real-coded genetic algorithm in terms of fitness, convergence and robustness.

Keywords: Evolutionary computation, quantum-inspired evolutionary algorithm, real observation, numerical optimization.

1 Introduction

Quantum-inspired evolutionary algorithm (QEA) is an unconventional algorithm of evolutionary computation. QEA inherits the structure and probabilistic search way of conventional genetic algorithm (CGA), and some concepts and operations of quantum computing, such as quantum-inspired bit (Q-bit), quantum-inspired gate (Q-gate) and quantum operators including superposition, entanglement, interference and measurement [1,2]. Up to now, as a better optimization method than CGA, QEA has been used in several applications of knapsack problem [2,3], digital filter design [4], feature selection [1]. Extensively experimental results manifest its advantages of good global search capability, rapid convergence and speediness [1-4].

In the existing QEA, only binary strings can be obtained by observing the probability amplitudes of Q-bits. Accordingly, the evolutionary strategy (update strategy of the rotation angles of Q-gates) was derived from a class of combinatorial optimization problems and was represented with binary code. The QEA in the existing literature is called binary-observation QEA (BQEA). Like binary-coded CGA, BQEA suffers from several disadvantages when it involves real

* This work was supported by the Scientific and Technological Development Foundation of Southwest Jiaotong University under the grant No.2006A09.

number optimization problems [5-7]. First of all, there is Hamming Cliff when real variables are encoded as binary strings. The Hamming distances exist between the binary codes of adjacent integers. For example, the integers 31 and 32 are represented respectively with binary codes 011111 and 100000, which have a Hamming distance of 6. To improve the code of 31 to that of 32, BQEA must alter all bits simultaneously, which is a difficult problem. Then, in encoding real number as binary strings, discretization error will inevitably be introduced in BQEA that cannot operate on a continuous space but on an evenly discretized space. Discretization error comes from the discrepancy between the binary representation space and a real space. Two points close to each other in a real space might be very far in the binary representation space. Finally, the encoding and decoding operations make BQEA more computationally expensive because the binary chromosome will have a huge string length when binary substring representing each real parameter with the desired precision are concatenated to form a chromosome. What is more, Han [2] also made it clear that the representation of real number may be more suitable for numerical optimization than that of binary string.

To overcome the drawbacks of BQEA, this paper proposes a real-observation QGA (RQEA) which is more suitable than BQEA for a wide range of real-world numerical optimization problems. Experiments on several functions are carried out to verify the effectiveness. Experimental results show that RQEA is able to find optimal or close-to-optimal solutions, and is more powerful than conventional real-coded genetic algorithm (CRGA) in terms of fitness, convergence and robustness.

2 RQEA

Quantum mechanical system is a probabilistic system. Like a classical probabilistic system, the probabilities of each state need be specified to describe the behavior of quantum mechanical system [8]. A quantum state vector $|\Psi\rangle$ can best be used to describe the location of a quantum particle and a weighted sum which in the case of two possible locations A and B equals $\alpha|A\rangle + \beta|B\rangle$, where α and β are complex number weighting factors of the particle being in locations A and B , respectively, and where $\alpha|A\rangle$ and $\beta|B\rangle$ are themselves state vectors [9]. Each two state quantum system is referred to as a Q-bit, which is also the smallest information unit in a two-state quantum computer [8]. The quantum state $|\Psi\rangle$ may be in the A state, in the B state, or in any superposition of the two. The quantum state $|\Psi\rangle$ can be represented as

$$|\Psi\rangle = \alpha|A\rangle + \beta|B\rangle . \quad (1)$$

where α and β satisfy the normalization equality

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

where $|\alpha|^2$ and $|\beta|^2$ are the probabilities that the Q-bit will be observed in A state, in the B state, respectively, in the act of observing the quantum state.

In this paper, states A and B are not considered as only the states 1 and 0, respectively, but an arbitrary pair of states between 1 and 0, which satisfy (1) and (2).

For a quantum system with n Q-bits, there will be n quantum states and 2^n information states. Of course, n different pairs of complex number weighting factors are needed to describe the quantum system and each describes the weighted probability of the particle being at that location [9]. A basic operation in quantum computing is that of a *fair coin flip* performed on a single Q-bit whose states are A and B . In an 2^n information state quantum system, this operation is performed on each Q-bit independently and can change the state of each Q-bit [8]. Through this operation, a Q-bit in the state A or B can be transformed into a superposition of the two states. In RQEA, this operation is implemented by using a Q-gate. If there are n locations as given by n state vectors, the particle is said to be all n locations at the same time.

Quantum mechanical systems have a deeper structure, and consequently, besides having a certain probability of being in each state, they also have a phase associated with each state [8]. In quantum computing, four quantum operators, superposition, entanglement, interference and measurement, are mainly used [10]. Superposition operator is applied for junction of possible solution spaces in a single unified solution space. Entanglement operator is employed to search the optimal solution as an unknown marked state. Interference and measurement operators are applied to extract the marked state with highest probability.

Instead of numeric, binary or symbol representation, Q-bit representation is used to represent the individuals of population in RQEA [1-4]. The probability amplitude of a Q-bit is defined firstly.

Definition 1. *The probability amplitude of a Q-bit is defined by a pair of numbers (α, β) as*

$$[\alpha \ \beta]^T . \quad (3)$$

where α and β satisfy normalization equality (2). $|\alpha|^2$ and $|\beta|^2$ denote the probabilities that the qubit will be found in A state and in B state in the act of observing the quantum state, respectively. Note that in general, the probability amplitudes can be complex quantities. However, in this paper or in RQEA, we only need real amplitudes with either positive or negative signs.

For quantum systems, in addition to having a certain probability of being in each state, they also have a phase associated with each state. The definition of Q-bit phase is given in the following.

Definition 2. *The phase of a Q-bit is defined with an angle ξ as*

$$\xi = \arctan(\beta/\alpha) . \quad (4)$$

where $\xi \in [-\pi/2, \pi/2]$. The sign of Q-bit phase ξ indicates which quadrant the Q-bit lies in. If ξ is positive, the Q-bit is regarded as being in the first or third quadrant, otherwise, the Q-bit lies in the second or fourth quadrant.

According to Def.1, the probability amplitudes of n Q-bits are represented as

$$\begin{bmatrix} \alpha_1|\alpha_2|\dots|\alpha_n \\ \beta_1|\beta_2|\dots|\beta_n \end{bmatrix} . \tag{5}$$

where $|\alpha_i|^2 + |\beta_i|^2 = 1, i = 1, 2, \dots, n$. The phase of the i^{th} Q-bit is

$$\xi_i = \arctan(\beta_i/\alpha_i) . \tag{6}$$

Q-bit representation can represent a linear superposition of states probabilistically. As shown in (5), n Q-bits are able to represent a linear superposition of 2^n states. Different from CGA, Q-bit representation makes it possible that four main quantum operators including superposition, entanglement, interference and measurement are implemented. So Q-bit representation is greatly superior to other representations in population diversity. This is what distinguishes RQEA from CGA.

According to the above Q-bit representation, the structure of RQEA is described as Algorithm 1, in which each step is explained briefly as follows.

Algorithm 1. Algorithm of RQEA

Begin

- (1) Set initial values of parameters; % Evolutionary generation $g=0$;
 - (2) Initialize $P(g)$; %
 - While (not termination condition) do
 - $g=g+1$;
 - (3) Generate $R(g)$ by observing $P(g-1)$; %
 - (4) Evaluate $R(g)$; %
 - (5) Store the best solution among $R(g)$ and $B(g-1)$ into $B(g)$;
 - (6) Update $P(g)$ using Q-gates; %
 - (7) Migration operation;
 - If (catastrophe condition)
 - (8) Catastrophe operation;
 - End if
 - End
 - End
-

- (1) Population size n_p , the number n_v of variables and the initial evolutionary generation g need be set.
- (2) In this step, population $P(g)=\{p_1^g, p_2^g, \dots, p_{n_p}^g\}$, where $p_i^g (i = 1, 2, \dots, n_p)$ is an arbitrary individual in population $P(g)$ and p_i^g is represented as

$$p_i^g = \begin{bmatrix} \alpha_{i1}^g|\alpha_{i2}^g|\dots|\alpha_{in_v}^g \\ \beta_{i1}^g|\beta_{i2}^g|\dots|\beta_{in_v}^g \end{bmatrix} . \tag{7}$$

where $\alpha_{ij}^g = \beta_{ij}^g = 1/\sqrt{2} (j = 1, 2, \dots, n_v)$, which means that all states are superposed with the same probability.

Table 1. Look-up table of function $f(\alpha, \beta)$ (Sign is a symbolic function)

$\xi_1 > 0$	$\xi_2 > 0$	$f(\alpha, \beta)$	
		$\xi_1 \geq \xi_2$	$\xi_1 < \xi_2$
True	True	+1	-1
True	False	sign($\alpha_1 \cdot \alpha_2$)	
False	True	-sign($\alpha_1 \cdot \alpha_2$)	
False	False	sign($\alpha_1 \cdot \alpha_2$)	-sign($\alpha_1 \cdot \alpha_2$)
$\xi_1, \xi_2 = 0$ or $\pi/2$		± 1	

- (3) According to probability amplitudes of all individuals in $\mathbf{P}(g - 1)$, observed states $\mathbf{R}(g)$ is constructed by observing $\mathbf{P}(g - 1)$. Here $\mathbf{R}(g) = \{\mathbf{a}_1^g, \mathbf{a}_2^g, \dots, \mathbf{a}_{n_p}^g\}$, where $\mathbf{a}_i^g (i = 1, 2, \dots, n_p)$ is an observed state of an individual $\mathbf{p}_i^{g-1} (i = 1, 2, \dots, n_p)$. \mathbf{a}_i^g is a real number with the dimension n_v , that is $\mathbf{a}_i^g = b_1 b_2 \dots b_{n_v}$, where $b_j (j = 1, 2, \dots, n_v)$ is a real number between 0 and 1. Observed states $\mathbf{R}(g)$ is generated in probabilistic way. For the probability amplitude $[\alpha \ \beta]^T$ of a Q-bit, a random number r in the range $[0, 1]$ is generated. If $r < |\alpha|^2$, the corresponding observed value is set to $|\alpha|^2$, otherwise, the value is set to $|\beta|^2$.
- (4) The fitness are calculated by using the obtained real parameter values.
- (5) The best solution are stored into $B(g)$.
- (6) In this step, the probability amplitudes of all Q-bits in population $\mathbf{P}(g)$ are updated by using Q-gates given in (8).

$$\mathbf{G} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} . \tag{8}$$

where θ is the rotation angle of Q-gate and θ is defined as $\theta = k \cdot f(\alpha, \beta)$, where k is chosen as

$$k = 0.5\pi e^{-\frac{\text{mod}(g, 100)}{10}} . \tag{9}$$

and $f(\alpha, \beta)$ is obtained by using the look-up table shown in Table 1, in which $\xi_1 = \arctan(\beta_1/\alpha_1)$ and $\xi_2 = \arctan(\beta_2/\alpha_2)$, where α_1, β_1 are the probability amplitudes of the best solution stored in $B(g)$ and α_2, β_2 are the probability amplitudes of the current solution.

- (7) Within an individual, the probability amplitudes of one Q-bit are migrated to those of another Q-bit, i.e.

$$\begin{bmatrix} |\alpha_{11}\rangle \leftrightarrow |\alpha_{12}\rangle \leftrightarrow \dots \leftrightarrow |\alpha_{1i}\rangle \leftrightarrow \dots \leftrightarrow |\alpha_{in_v}\rangle \\ |\beta_{11}\rangle \leftrightarrow |\beta_{12}\rangle \leftrightarrow \dots \leftrightarrow |\beta_{1i}\rangle \leftrightarrow \dots \leftrightarrow |\beta_{in_v}\rangle \end{bmatrix}$$

- (8) The catastrophe condition is a prescribed generation C_g , such as 10 or 20.

3 Experiments

To test the effectiveness of RQEA, 13 functions $f_1 - f_{13}$ are used to bring into comparison with CGA [5-7]. The evolutionary strategies of CRGA include elitism

selection, uniform crossover and uniform mutation. The crossover and mutation probabilities are set to 0.8 and 0.1, respectively. RQEA and CRGA use the same population size 20 and the same maximal generation 500. The parameter C_g is set to 20 in RQEA. RQEA and CRGA are performed 50 independent runs for each test function, respectively. The mean best values and the standard deviations are recorded for each test function. Experimental results are listed in Table 2, in which m, σ, g and p represent the mean best, the standard deviation, the maximal number of generations and the population size, respectively. The results are averaged over 50 runs. From Table 2, it can be seen that RQEA obtains far better results than CRGA in terms of both the mean best solutions and the standard deviations for all of the test functions.

(I) Sphere function

$$f_1(\mathbf{x}) = \sum_{i=1}^N x_i^2, \quad -100.0 \leq x_i \leq 100.0, \quad N = 30. \quad (10)$$

(II) Ackley function

$$f_2(\mathbf{x}) = 20 + e - 20 \exp\left(-0.2\sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2}\right) - \exp\left(\frac{1}{N} \sum_{i=1}^N \cos(2\pi x_i)\right) \\ -32.0 \leq x_i \leq 32.0, \quad N = 30 \quad (11)$$

(III) Griewank function

$$f_3(\mathbf{x}) = \frac{1}{4000} \sum_{i=1}^N x_i^2 - \prod_{i=1}^N \left(\frac{x_i}{\sqrt{i}}\right) + 1, \quad -600.0 \leq x_i \leq 600.0, \quad N = 30. \quad (12)$$

(IV) Rastrigin function

$$f_4(\mathbf{x}) = 10N + \sum_{i=1}^N (x_i^2 - 10 \cos(2\pi x_i)), \quad -5.12 \leq x_i \leq 5.12, \quad N = 30. \quad (13)$$

(V) Schwefel function

$$f_5(\mathbf{x}) = 418.9829N - \sum_{i=1}^N (x_i \sin(\sqrt{|x_i|})), \quad -500.0 \leq x_i \leq 500.0, \quad N = 30. \quad (14)$$

(VI) Schwefel's problem 2.22

$$f_6(\mathbf{x}) = \sum_{i=1}^N |x_i| + \prod_{i=1}^N |x_i|, \quad -10 \leq x_i \leq 10, \quad N = 30. \quad (15)$$

(VII) Schwefel's problem 1.2

$$f_7(\mathbf{x}) = \sum_{i=1}^N \left(\sum_{j=1}^i x_j\right)^2, \quad -100 \leq x_j \leq 100, \quad N = 30. \quad (16)$$

Table 2. Comparisons of RQEA and CRGA

	RQEA				CRGA		Global minimum
	<i>g</i>	<i>p</i>	<i>m</i>	σ	<i>m</i>	σ	
f_1	500	20	1.11×10^{-7}	2.19×10^{-7}	1.55×10^4	2.20×10^3	0
f_2	500	20	2.62×10^{-4}	1.29×10^{-3}	15.67567	0.478948	0
f_3	500	20	1.50×10^{-6}	5.63×10^{-6}	141.9507	15.16764	0
f_4	500	20	1.44×10^{-7}	3.21×10^{-7}	239.6483	12.61521	0
f_5	500	20	0.194603	0.684961	9.80×10^3	3.21×10^2	0
f_6	500	20	1.78×10^{-4}	2.82×10^{-4}	1.29×10^9	2.32×10^9	0
f_7	500	20	3.07×10^{-6}	8.15×10^{-6}	6.29×10^4	1.74×10^4	0
f_8	500	20	6.02×10^{-5}	7.34×10^{-5}	39.63123	2.649442	0
f_9	500	20	0	0	1.60×10^4	1.51×10^4	0
f_{10}	500	20	2.06×10^{-3}	1.92×10^{-3}	12.15373	3.226462	0
f_{11}	500	20	0.998004	1.82×10^{-10}	1.578839	0.658432	≈ 1
f_{12}	500	20	-1.031628	4.42×10^{-9}	-0.966542	0.069581	-1.031628
f_{13}	500	20	0.397904	8.63×10^{-5}	0.525740	0.169221	0.397877

(VIII) Schwefel’s problem 2.21

$$f_8(\mathbf{x}) = \max_{i=1} \{|x_i|, 1 \leq i \leq 30\}, \quad -100 \leq x_i \leq 100. \quad (17)$$

(IX) Step function

$$f_9(\mathbf{x}) = \sum_{i=1}^N ([x_i + 0.5])^2, \quad -100 \leq x_i \leq 100. \quad (18)$$

(X) Quartic function, i.e. noise

$$f_{10}(\mathbf{x}) = \sum_{i=1}^N ix_i^4 + \text{random}[0, 1), \quad -1.28 \leq x_i \leq 1.28. \quad (19)$$

(XI) Shekel’s Foxholes function

$$f_{11}(\mathbf{x}) = \left[\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^2 (x_i - a_{ij})^6} \right]^{-1}, \quad -10 \leq x_j \leq 10. \quad (20)$$

where $(a_{ij}) = \begin{pmatrix} -32 & -16 & 0 & 16 & 32 & -32 & \dots & 0 & 16 & 32 \\ -32 & -32 & -32 & -32 & -32 & -16 & \dots & 32 & 32 & 32 \end{pmatrix}$.

(XII) Six-hump camel-back function

$$f_{12}(\mathbf{x}) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4, \quad -5 \leq x_i \leq 5. \quad (21)$$

(XIII) Branin function

$$f_{13}(\mathbf{x}) = \left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos(x_1) + 10. \quad (22)$$

$$-5 \leq x_1 \leq 10, \quad 0 \leq x_2 \leq 15$$

4 Concluding Remarks

Extending two states 1 and 0 to an arbitrary pair of states between 1 and 0 in quantum system, this paper proposes RQEA to solve numerical optimization problems. RQEA can be considered as the extensive version of BQEA to real number solution space. Extensive experiments show that RQEA is a competitive algorithm for numerical optimization problems. Our future work will be concentrated on the applications of RQEA.

References

1. Zhang, G.X., Hu, L.Z., Jin, W.D.: Quantum Computing Based Machine Learning Method and Its Application in Radar Emitter Signal Recognition. In: Torra, V., Narukawa, Y., (eds.): *Lecture Notes in Artificial Intelligence*, Vol.3131. Springer-Verlag, Berlin Heidelberg New York (2004) 92-103
2. Han, K.H., Kim, J.H.: Quantum-Inspired Evolutionary Algorithms with a New Termination Criterion, H_ϵ Gate, and Two-Phase Scheme. *IEEE Transactions on Evolutionary Computation* **8** (2004) 156-169
3. Han, K.H., Kim, J.H.: Quantum-Inspired Evolutionary Algorithms for a Class of Combinatorial Optimization. *IEEE Transactions on Evolutionary Computation* **6** (2002) 580-593
4. Zhang, G.X., Jin, W.D., Li, N.: An Improved Quantum Genetic Algorithm and Its Application. In: Wang, G., et al., (eds.): *Lecture Notes in Artificial Intelligence*, Vol.2639. Springer-Verlag, Berlin Heidelberg New York (2003) 449-452
5. Oyama, A., Obayashi, S., Nakahashi, K.: Real-Coded Adaptive Range Genetic Algorithm and Its Application to Aerodynamic Design. *International Journal of Japan Society of Mechanical Engineers, Series A* **43** (2000) 124-129
6. Qing, A.Y., Lee, C.K., Jen, L.: Electromagnetic Inverse Scattering of Two-Dimensional Perfectly Conducting Objects by Real-Coded Genetic Algorithm. *IEEE Transactions on Geoscience and Remote Sensing* **39** (2001) 665-676
7. Wang, J.L., Tan, Y.J.: 2-D MT Inversion Using Genetic Algorithm. *Journal of Physics: Conference Series* **12** (2005) 165-170
8. Grover, L.K.: Quantum Computation. In: *Proceedings of the 12th Int. Conf. on VLSI Design* (1999) 548-553
9. Narayanan, A.: Quantum Computing for Beginners. In: *Proc. of the 1999 Congress Evolutionary Computation* (1999) 2231-2238
10. Ulyanov, S.V.: Quantum Soft Computing in Control Process Design: Quantum Genetic Algorithm and Quantum Neural Network Approaches. In: *Proc. of World Automation Congress vol.17* (2004) 99-104