A Generalized Quantum-Inspired Evolutionary Algorithm for Combinatorial Optimization Problems

Julio Manuel Alegría R.
School of Computer Science
Universidad Católica San Pablo (UCSP)
Arequipa, Peru
Email: julio.alegria@ucsp.pe

Yván Jesús Tupac V.
School of Computer Science
Universidad Católica San Pablo (UCSP)
Arequipa, Peru
Email: ytupac@ucsp.pe

***Abstract*—This paper proposes a generalization of the Quantum-Inspired Evolutionary Algorithm (QIEA): the Generalized Quantum-Inspired Evolutionary Algorithm (GQIEA). Same as QIEA, GQIEA is also based on the quantum computing principle of superposition of states, but instead of using only binary values (0 or 1) for the states, any finite set of values $(1, 2, \dots, n)$ can be used. GQIEA, as any other evolutionary algorithm, defines an individual representation , the evaluation function and population operators. As in QIEA, GQIEA also defines a generalized Q-gate operator, which is a variation operator to drive the individuals toward better solutions. To demonstrate its effectiveness and applicability, the proposal will be applied to the Knapsack Problem (KP), a classic combinatorial optimization problem. Results show that GQIEA has a good performance, even with a small population.**

Keywords*—Evolutionary Computation; Quantum-inspired algorithm; Knapsack problem.

I. INTRODUCTION

Evolutionary algorithms (EAs) are populational-based metaheuristic optimization algorithms based on the principles of natural biological evolution. EAs have several advantages over other classical optimization methods: they are robust, global, highly parallel, easily adaptable, and they need very few domain-specific knowledge to achieve good performance [1].

EAs are population algorithms, which means they operate with a set of individuals, applying concepts inspired by nature (such as survival of the fittest and genetic heritage) to successively approximate the individuals to optimal solutions. At each generation, new individuals are generated based on variation operations performed on individuals of the previous generation. The selection of the individuals that take part in this process is done based on their *fitness*. The fitness of an individual describes how well it adapts to the environment. The described process leads to generate individuals that are better suited to their environment than their predecessors, just as in natural adaptation.

Although it has been seen that EAs achieved good results for several optimization problems, some others have some characteristics that may impair the EAs performance. Because EAs need to validate their individuals constantly, problems with an expensive evaluation function will cause inefficient performance. In this sense, the Quantum-Inspired Evolutionary Algorithms [2] and some variants [3], [4], [5], [6] arise to address these cases.

Quantum mechanical computers were proposed in the early 1980s [7] and were formalized in the late 1980s [8]. Research on using quantum computing in EAs started since late 1990s. In 2002, Kuk-Hyun Han and Jong-Hwan Kim proposed the Quantum-Inspired Evolutionary Algorithm (QIEA) [2], where individuals are represented by a string of Q-bits, in a way that a "Q-bit individual" can represent a linear superposition of binary solutions.

But having only two possible states (0 or 1) has some disadvantages, such as: an excessive increase of dimensionality, numerical imprecision, or the need for a complex codification of solutions. In this paper we propose a generalization of the QIEA, so that individuals can represent a superposition of combinatorial solutions, in order to improve their performance in combinatorial optimization.

II. QIEA

To describe QIEA, we must briefly address some concepts of quantum computing. A Q-bit may be in the "0" state, in the "1" state, or in any superposition of the two [9]. The state of a Q-bit is represented in Equation 1.

$$|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle \tag{1}$$

where α and β are both complex numbers, so that $|\alpha|^2$ gives the probability that the Q-bit will be found in the "0" state and $|\beta|^2$ gives the probability that the Q-bit will be found in the "1" state. Fig. 1 will help us understand the relationship between α and β.

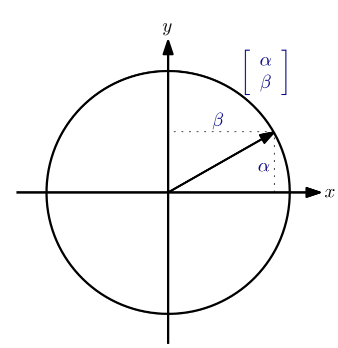

Fig. 1. Graphical representation of a Q-bit. Source: [10].

Inspired by these concepts, QIEA designs a Q-bit representation. In this representation, a Q-bit is defined as the smallest unit of information, as shown in Equation 2

$$\begin{bmatrix} \alpha \\ \beta \end{bmatrix} \tag{2}$$

where $|\alpha|^2 + |\beta|^2 = 1$.

A Q-bit individual is a string of m Q-bits. The advantage of Q-bit representation is that each individual is able to represent a linear superposition of states (binary solutions). If there is a system of m Q-bits, 2^m states can be represented at the same time. However, in the act of observing a quantum state, it collapses to a single state [2].

If there is, for instance, a three-Q-bit system, a individual can represent up to eight (2^3) states at the same time ($|000\rangle$, $|001\rangle$, $|010\rangle$, $|011\rangle$, $|100\rangle$, $|101\rangle$, $|110\rangle$, $|111\rangle$). Because of this, QIEA will have more diverse solutions that traditional EAs, even with a smaller population.

The update of the population in QIEA is performed by a variation operator called Q-gate [2]. This operator is defined as a rotation matrix, as shown in Equation 3.

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

This matrix is used to multiply each of the individual Q-bits. Each pair of values α and β will be treated as a two-dimensional vector and rotated using the matrix shown. This update will be performed for each Q-bit, according to Equation 4. Fig. 2 shows a graphical representation of the operator.

$$\begin{bmatrix} \alpha_i' \\ \beta_i' \end{bmatrix} = \mathbf{U}(\Delta\theta_i) \begin{bmatrix} \alpha_i \\ \beta_i \end{bmatrix} \tag{4}$$

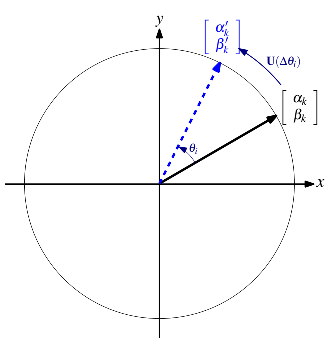

Fig. 2. Graphical representation of the Q-gate operator.

The values of $\Delta\theta$ are defined depending on the problem to be solved, while meeting the condition of being able to modify the values of α and β, so that they drive individuals toward optimal solutions [2].

Fig. 3 shows all the phases of the QIEA.

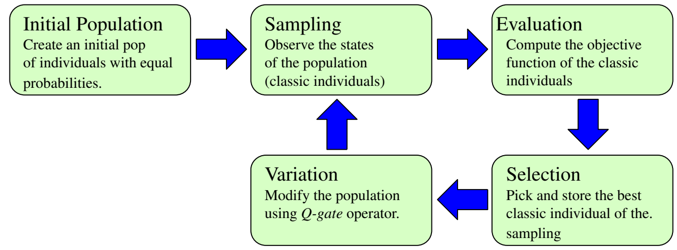

Fig. 3. Complete cycle of the QIEA.

III. The GQIEA Model

In QIEA, the pair (α, β) of a Q-bit defines the probability that the Q-bit is found in state "0" or in state "1". In GQIEA, instead of just two states, we generalize the idea to work with any finite number of states. So, the smallest unit of information is now a GQ-bit (generalized Q-bit), a tuple of size n, being n the total numbers of different possible S states, defined in Equation 5.

$$\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{bmatrix} \tag{5}$$

where $0 \leq \alpha_i \leq 1$ and $\sum_{i=1}^{n} \alpha_i = 1$.

In this representation, α_i defines the probability that the GQ-bit is found in state s_i. A "GQ-bit individual" will then be a string of m GQ-bits, defined in Equation 6.

$$\left[\begin{array}{c|c|c|c} \alpha_{1,1} & \alpha_{1,2} & \cdots & \alpha_{1,m} \\ \alpha_{2,1} & \alpha_{2,2} & \cdots & \alpha_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{n,1} & \alpha_{n,2} & \cdots & \alpha_{n,m} \end{array}\right] \tag{6}$$

This GQ-bit individual will be able to represent a superposition of all combinatorial solutions.

Same as the Q-gate in the original QIEA, we define a variation operator named GQ-gate (generalized Q-gate), responsible for driving the individuals toward better solutions, based on the best individual found in the generation. In this paper we propose two GQ-gate operators: the arithmetic GQ-gate and the geometric GQ-gate.

Say we have the j-th GQ-bit of a GQ-bit individual, which represents a superposition of n states $\{s_1, s_2, \ldots, s_n\}$. If the state s_ϕ was the one found in j-th position of the best individual of the generation, the new probabilities of the GQ-bit will be determined following Equation 7, for the arithmetic GQ-gate, or following Equation 8, for the geometric GQ-gate.

$$\alpha_i = \begin{cases} \alpha_i + \Delta_t & \text{if } i = \phi \\ \alpha_i & \text{if } i \neq \phi \end{cases} \quad , \quad i = 1, 2, \ldots, n \tag{7}$$

$$\alpha_i = \begin{cases} \alpha_i \cdot \Delta_t & \text{if } i = \phi \\ \alpha_i & \text{if } i \neq \phi \end{cases} \quad , \quad i = 1, 2, \ldots, n \tag{8}$$

If there is, for instance, a three-GQ-bit system, where each GQ-bit of an individual can be observed in four different states $S = \{s_1, s_2, s_3, s_4\}$, up to 64 ($4^3$) combinatorial solutions can be represented at the same time in a single individual. If we say that the best individual found in a specific generation is, for example: s_1, s_4, s_2, and we are using the arithmetic GQ-gate, the new individual will be determined as shown in Equation 9.

$$\left[\begin{array}{c|c|c} \alpha_{1,1} + \Delta_t & \alpha_{1,2} & \alpha_{1,3} \\ \alpha_{2,1} & \alpha_{2,2} & \alpha_{2,3} + \Delta_t \\ \alpha_{3,1} & \alpha_{3,2} & \alpha_{3,3} \\ \alpha_{4,1} & \alpha_{4,2} + \Delta_t & \alpha_{4,3} \end{array}\right] \quad (9)$$

where Δ_t is a parameter of the algorithm. Each GQ-bit will then need to be normalized to continue fulfilling the property $\sum_{i=1}^{n} \alpha_i = 1$.

The structure of GQIEA is described in Algorithm 1.

Algorithm 1 Structure of the GQIEA.

```
t ← 0
initialize Q_t
while not termination-condition do
    t ← t + 1
    generate P_t observing the states of Q_{t−1}
    evaluate P_t
    store best solution in b_t
    b_t ← best between b_t and b_{t−1}
    Q_t ← GQ-gate(Q_{t−1}, b_t, Δ_t)
end while
```

Q_t is the algorithm population, which may have one or more GQ-bit individuals, while P_t is a population of classical individuals, generated for each generation. b_t will used to store the best individuals generated bt the GQIEA along its evolutionary process.

IV. Experiments

In order to demonstrate its effectiveness and applicability, the proposal will be applied to the Knapsack Problem (KP), that consists of finding the optimum quantity of objects that can be included in a container with a maximum weight, while optimizing the total value of the included values, as long as the total weight does not surpass the maximum [11]. In the original KP, proposed in 1972, the goal was to maximize the sum of included values, given a maximum weight W, as shown in Equation 10.

$$\begin{aligned} &\text{Maximize} \quad \sum_{\forall k \in K} v_k \cdot x_k \qquad x_k \in \{0, 1\} \\ &\text{such that} \quad \sum_{\forall k \in K} w_k \cdot x_k \leq W \end{aligned} \quad (10)$$

Where v_k and w_k are the value and weight of object k, and $x_k = 1$ if object k was included in the container, or $x_k = 0$ otherwise.

For this experiment, we use a variation of the KP called *Bounded Knapsack Problem* (BKP). In this variation, the goal remains the same, but the difference is that several copies of the same object k can be included, as long as the maximum of that specific object is not exceeded (m_k). The function is expressed in Equation **??**.

$$\begin{aligned} &\text{Maximize} \quad \sum_{\forall k \in K} v_k \cdot x_k \qquad x_k \in \{0, 1, \ldots, m_k\} \\ &\text{such that} \quad \sum_{\forall k \in K} w_k \cdot x_k \leq W \end{aligned} \quad (11)$$

Where v_k and $w_k$are the values and weights of object k, and x_k represents the number of copies of object k included in the container.

The proposal model will be compared with a traditional Combinatorial EA [12] and the QIEA [2].

The first approach uses a combinatorial representation, where each individual of the population is as follows: $x = (x_1, x_2, \ldots, x_n)$, where n equals the total number of objects, and $0 \leq x_i < m_i$.

QIEA uses a quantum binary representation, where each individual wil be conformed by a string of Q-bits. Each Q-bit will be related to a specific copy of an object k, defining the probability of including the copy or not in the container. Therefore, the dimension of each indivual is the sum of the maximum quantities of each object (m_i).

Our proposal uses a quantum combinatorial representation, with a string of n GQ-bits, being n the total number of objects. In this representation, each GQ-bit is related to a specific a object k, defining the probability of having a certain amount of copies.

Before the experiment, we performed a parameter adjustment for each algorithm.

The Combinatorial EA uses the following parameters: selection method (roulette wheel, tournament, stochastic remainder), crossover method (one-point, two-point, uniform), mutation method (disorder, change), crossover coefficient ($[0.4, 0.9]$), and mutation coefficient ($[0.02, 0.2]$). The best values found for these parameters, obtained based on iterative tests, are shown in Table I.

TABLE I. Best values for parameters for Combinatorial EA.

Parameter	Value
Selection method	Roulette wheel
Crossover method	Uniform
Mutation method	Change
Crossover coefficient	0.5
Mutation coefficient	0.14

QIEA uses only one parameter: theta ($[0.01\pi, 0.2\pi]$). The best value found for this parameter, obtained based on iterative tests, is shown in Table II.

TABLE II. Best values for parameters for QIEA.

Parameter	Value
Theta	0.10π

GQIEA also uses only one parameter: delta, both for arithmetic and geometric GQ-gate. In the first case, its value will be between $[0.05, 0.25]$), and for the other case, its value will be between $[1.05, 1.25]$. The best values found for this parameter, obtained based on iterative tests, are shown in Table III.

TABLE III. BEST VALUES FOR PARAMETERS FOR GQIEA.

Parameter	GQ-gate	Value
Delta	Arithmetic	0.06
Delta	Geometric	1.11

The results of the algorithms compared (using the parameter values defined previously) applied to the Knapsack Problem are shown in Table IV.

TABLE IV. RESULTS OF THE ALGORITHMS COMPARED, APPLIED TO THE KNAPSACK PROBLEM.

Algorithm	Result	Runtime
Combinatorial EA	12.65	145.962 seconds
QIEA	12.51	49.924 seconds
GQIEA, using Arithmetic GQ-gate	12.65	31.970 seconds
GQIEA, using Geometric GQ-gate	12.65	36.649 seconds

Fig. 4 shows a graphical comparison between the Combinatorial EA, the QIEA and the GQIEA. Fig. 5 shows an average of 10 executions of the same algorithms.

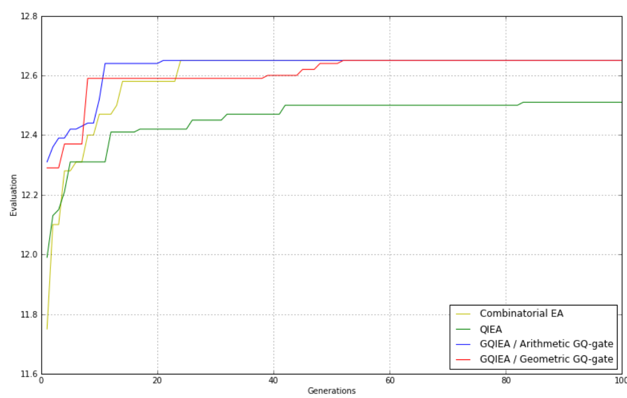

Fig. 4. Comparison of the Combinatorial EA, the QIEA and the GQIEA, applied to the Knapsack Problem.

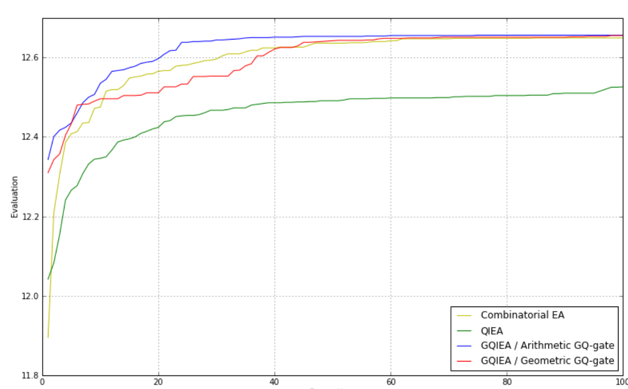

Fig. 5. Comparison of an average of 10 executions of the Combinatorial EA, the QIEA and the GQIEA, applied to the Knapsack Problem.

Table IV shows us that three out of four algorithms reached the same final result, while one (QIEA) was left behind. The algorithm with the slowest runtime was the Combinatorial EA, as it was expected. We can also see that both GQIEA had faster runtimes than the QIEA. This because there was a dimensionality reduction.

We must be careful interpreting Fig. 4. At first glance it may seem that the Combinatorial EA is converging faster than the rest of the algorithms. But as we can see, the X axis is based on generations and not on time. Based on time, both instances of the GQIEA are the ones converging faster.

Finally, Fig. 5 shows us an average of 10 executions of all algorithms. This figure gives us a better idea about the trends in the behaviour of these algorithms. We can see that the quality of the response and the convergence remains stable.

V. CONCLUSIONS

In this paper a new evolutionary model was proposed, which generalizes the QIEA. In this proposal, the representation of individuals allows the superposition of solutions conformed by any set of finite states, in order to improve the performance achieved by the QIEA in combinatorial optimization problems with expensive evaluation function.

A representation was presented, which allows the superposition of combinatorial solutions. This representations reduces dimensionality, and the complex encoding of solution becomes unnecesary.

In addition, two variation operators were proposed, called GQ-gate operators: arithmetic GQ-gate and geometric GQ-gate. These operators are responsible for driving the individuals toward better solutions. Results show that the arithmetic GQ-gate has a better convergence at a smaller runtime.

The proposed model was compared with two algorithms used in combinatorial optimization: the Combinatorial EA and the QIEA. Experiments showed that the GQIEA has a good quality of results and convergence, improving in most cases the performance of the algorithms with which it was compared.

Is important to note that the process of adjusting the parameters for the Combinatorial EA (5 parameters) was much longer than the adjustment of parameters for the QIEA and GQIEA (1 parameter).

In conclusion, the Generalized Quantum-Inspired Evolutionary Algorithm is a good alternative when handling a combinatorial optimization problem with a expensive evaluation function. The small size of population needed, the low dimensionality of its individuals, and the absense of a complex encoding of solutions are its main advantages.

VI. FUTURE WORK

Applying parallelism in the phases of sampling and evaluation would accelerate significantly the runtime of the GQIEA.

Designing ad-hoc GQ-gate operators for the problem domain would improve the quality of results.

REFERENCES

[1] T. Back, D. B. Fogel, and Z. Michalewicz, *Handbook of Evolutionary Computation*. Institute of Physics Publishing Ltd. Bristol, UK, 1997.

[2] K.-H. Han and J.-H. Kim, "Quantum-inspired evolutionary algorithm for a class of combinatorial optimization," *Evolutionary Computation, IEEE Transactions on*, vol. 6, pp. 580–593, 2002.

[3] A. da Cruz, M. M. B. R. Vellasco, and M. A. C. Pacheco, "Quantum-inspired evolutionary algorithms applied to numerical optimization problems," in *Evolutionary Computation (CEC), 2010 IEEE Congress on*, 2010, pp. 1–6.

[4] C. Chung, H. Yu, and K.-P. Wong, "An advanced quantum-inspired evolutionary algorithm for unit commitment," *Power Systems, IEEE Transactions on*, vol. 26, no. 2, pp. 847–854, 2011.

[5] F. Leon, "Real-valued quantum-inspired evolutionary algorithm for multi-issue multi-lateral negotiation," in *Intelligent Computer Communication and Processing (ICCP), 2012 IEEE International Conference on*, 2012, pp. 41–48.

[6] J.-T. Tsai, J.-H. Chou, and W.-H. Ho, "Improved quantum-inspired evolutionary algorithm for engineering design optimization," *Mathematical Problems in Engineering*, vol. 1, p. 27, 2012.

[7] P. Benioff, "The computer as a physical system: A microscopic quantum mechanical hamiltonian model of computers as represented by turing machines," *Journal of Statistical Physics*, vol. 22, pp. 563–591, 1980.

[8] D. Deutsch, "Quantum theory, the church-turing principle and the universal quantum computer," *Proceedings Royal Society*, vol. 400, pp. 97–117, 1985.

[9] T. Hey, "Quantum computing: an introduction," *Computing & Control Engineering Journal*, vol. 10, pp. 105–112, 1999.

[10] A. V. Abs da Cruz, M. A. C. Pacheco, and M. M. B. R. Vellasco, "Quantum-inspired evolutionary algorithms for problems based on numerical representation," Ph.D. dissertation, Pontificia Universidade Catolica do Rio de Janeiro, 2007.

[11] H. Kellerer, U. Pferschy, and D. Pisinger, *Knapsack Problems*. Springer Berlin Heidelberg, 2004.

[12] V. Snášel, J. Platoš, P. Krömer, and N. Ouddane, "Genetic algorithms for the use in combinatorial problems," in *Foundations of Computational Intelligence Volume 3*, ser. Studies in Computational Intelligence, A. Abraham, A.-E. Hassanien, P. Siarry, and A. Engelbrecht, Eds. Springer Berlin Heidelberg, 2009, vol. 203, pp. 3–22. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-01085-9_1