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 extends it not only to be used for binary values (0 or 1), but for any finite set of values ($1, \dots, n$). 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 Assignment Problem (AP), a classic combinatorial optimization problem. Results show that GQIEA has a good performance, even with a small population.**

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

I. Introduction

Evolutionary algorithms (EAs) are a 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].

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.

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. 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.

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

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].

III. The GQIEA Model

In QIEA, the pair (α, β) of a Q-bit define 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 arithmetical GQ-gate and the geometrical 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 arithmetical GQ-gate, or following Equation 8, for the geometrical 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 arithmetical GQ-gate, the new individual will be determined as shown in Equation 9.

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

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 Assignment Problem (AP), that consists of optimizing the assignment of a set of tasks T to a set of agents A. For this experiment, we use a variation of the AP called *Task BGAP* [10], where multiple tasks can be assigned per agent, and the goal is to minimize the maximum cost of the assignments made, as shown in Equation 10.

$$\text{Minimize} \quad \max_{\forall a \in A} \sum_{\forall t \in T} c_{a,t} \cdot x_{a,t} \tag{10}$$

where $c_{t,a}$ is the cost of assigning agent a to task t, and $x_{t,a} = 1$ if agent a was assigned to task t or $x_{t,a} = 0$ if not. Each task can be only be assigned to one agent, but each agent can be assigned more than one task.

The proposal model will be compared with a traditional Combinatorial EA [11]. In this approach, each individual x of the population is conformed by genes $x = (g_1, g_2, \ldots, g_n)$, where each gene g_i can take one of the set of states S. The Combinatorial EA uses traditional crossover and mutation operators to drive individuals towards optimal solutions.

Fig. 1 shows a comparison between the Combinatorial EA and the GQIEA, using the arithmetic GQ-gate. While Fig. 2 uses the geometric GQ-gate.

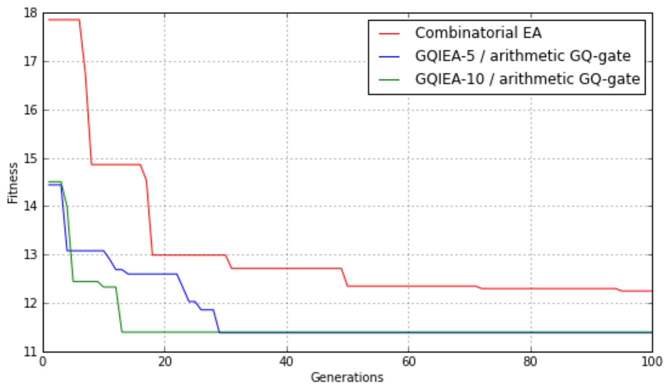

Fig. 1. Comparison between traditional Combinatorial EA and GQIEA using arithmetic GQ-gate, applied to the Assignment Problem (*Task BGAP*).

TABLE I. PARAMETERS USED IN FIG. 1.

Algorithm(s)	Parameter	Value
Combinatorial EA	Population size	100
Combinatorial EA	Selection	Roulette wheel
Combinatorial EA	Crossover	Uniform
Combinatorial EA	Crossover prob.	0.65
Combinatorial EA	Mutation prob.	0.08
GQIEA-5	Population size	5
GQIEA-10	Population size	10
GQIEA-5, GQIEA-10	GQ-Gate	Arithmetic
GQIEA-5, GQIEA-10	Δ	0.2

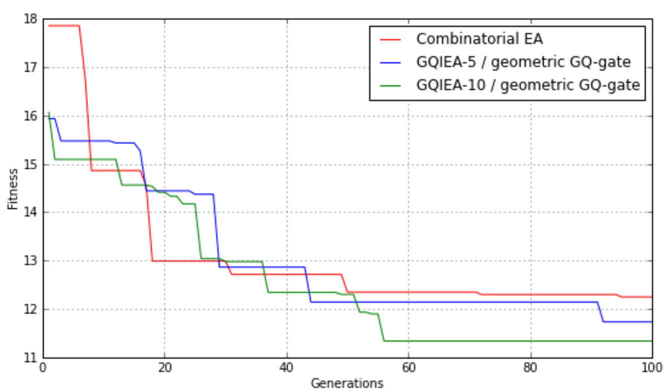

Fig. 2. Comparison between traditional Combinatorial EA and GQIEA using geometric GQ-gate, applied to the Assignment Problem (*Task BGAP*).

V. CONCLUSIONS

In this paper a generalization of QIEA (GQIEA) was proposed, defining a GQ-bit individual and designing two variation operators, called GQ-gate operators. GQIEA, using quantum computing principles, represents individuals as a superposition of combinatorial solutions.

TABLE II. PARAMETERS USED IN FIG. 2.

Algorithm(s)	Parameter	Value
Combinatorial EA	Population size	100
Combinatorial EA	Selection	Roulette wheel
Combinatorial EA	Crossover	Uniform
Combinatorial EA	Crossover prob.	0.65
Combinatorial EA	Mutation prob.	0.08
GQIEA-5	Population size	5
GQIEA-10	Population size	10
GQIEA-5, GQIEA-10	GQ-Gate	Geometric
GQIEA-5, GQIEA-10	Δ	1.1

The Assignment Problem was applied as a testbed to demonstrate the performance of the proposed algorithm. In the experiments, GQIEA outperforms the results of the traditional Combinatorial EA, even with a much smaller population. It is still intended to continue refining this model and its GQ-gate operators to further increase performance.

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] J. Mazzola and A. Neebe, "Bottleneck generalized assignment problems," *Engineering Costs and Production Economics*, vol. 14, pp. 61–65, 1988.

[11] 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