Quantum-Inspired Evolutionary Algorithm and its Application to Inverse Problems

S. L. Ho[1], Shiyou Yang[2], and Guangzheng Ni[2]
[1] Department of Electrical Engineering, The Hong Kong Polytechnic University, Hong Kong
[2] College of Electrical Engineering, Zhejiang University, Hangzhou, 310027, China

Abstract —This paper reports the investigations of the potential of a new evolutionary algorithm based on probabilistic models - the quantum-inspired evolutionary algorithm in finding solutions of inverse problems. To enhance the convergence speed without compromising the diversity performances of the populations, a new definition of local information sharing is introduced and implemented. Also, to guarantee the balance between explorations and exploitations, a mechanism for global information sharing is proposed. The proposed algorithm is evaluated on both low- and high-frequency inverse problems with promising results.

I. Quantum-Inspired Evolutionary Algorithm

Hitherto, evolutionary algorithms (EA) have become the standards for solving inverse problems [1],[2]. However, the three key operators such as selection, crossover and mutation are very complex in terms of both theory and numerical implementation. Moreover, the components of EAs, such as population size, variation operators, parent selection, reproduction and inheritance, survival competition methods, among others, must be designed properly before one can secure a good balance between exploration and exploitation searches [3]. In this regard, increasing efforts have been devoted to Evolutionary Algorithms based on Probabilistic Models (EAPM) to overcome the disruptive effects when the aforementioned genetic operators are implemented [4]. Since EAPMs explicitly extract global statistical information from their previous searches to build probability distribution models of promising solutions, there are no traditional genetic operators in EAPMs. The Quantum-inspired Evolutionary Algorithm (QEA) [3],[5], which applies quantum computing principle to enhance the classical evolutionary algorithms, is a kind of EMPMs, and it can, moreover, treat the balance between the explorations and exploitations readily. Nevertheless, the application of QEA to inverse problems has yet to be studied and reported.

A. Quantum Computing Principles [6]

In quantum computing, the smallest information unit is a quantum bit or *Q*-bit, a pair of complex numbers $[\alpha\ \beta]^T$. In contrast to a traditional binary bit, besides the state “1” or “0”, a *Q*-bit may also be in any superposition of both states. Thus, the state $|\Psi\rangle$ of a *Q*-bit $[\alpha\ \beta]^T$ is defined as

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

where $|\alpha|^2$ and $|\beta|^2$ give, respectively, the probabilities that the *Q*-bit will be found in “0” and “1” with the following normalization condition

$$|\alpha|^2 + |\beta|^2 = 1 \tag{2}$$

The corresponding analogue of a traditional individual in quantum computing is a *Q*-bit individual of a string of *m* *Q*-bits as given by

$$\begin{bmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_m \\ \beta_1 & \beta_2 & \cdots & \beta_m \end{bmatrix} \tag{3}$$

where $|\alpha_i|^2 + |\beta_i|^2 = 1\ (i = 1, 2, \cdots, m)$.

Obviously, evolutionary algorithms with a *Q*-bit probabilistic representation have better characteristics in population diversities [3]. To modify the probabilities α_i and β_i, some quantum gates are generally applied. Moreover, since (3) is a probabilistic representation of a decision parameter, it is generally observed/collapsed/measured to form a specific binary individual.

B. A QEA for Inverse Problems

The proposed QEA is a quantum population based evolutionary algorithm modified on the most illustrative version of [3]. In the description that follows, $Q(t) = \left\{q_1^t, q_2^t, \cdots, q_n^t\right\}$ is the current quantum population, q_j^t is its j^{th} *Q*-individual as given by

$$q_j^t = \begin{bmatrix} \alpha_{j1}^t & \alpha_{j2}^t & \cdots & \alpha_{jm}^t \\ \beta_{j1}^t & \beta_{j2}^t & \cdots & \beta_{jm}^t \end{bmatrix}, \tag{4}$$

and $p(t) = \left\{p_1^t, p_2^t, \cdots, p_n^t\right\}$ is the binary population obtained after observing $Q(t)$, and a comprehensive definition and description of these parameters are given in [3]. To facilitate the explanation of the proposed QEA, its iterative procedures are described as:

Procedure of QEA for Inverse Problems

1) initialize: *Q*(*t*), *B*(*t*) (the vector storing the *elitist* solutions of *Q*(*t*))
2) **while** (not termination condition) **do**
3) make *P* (*t*) by observing *Q*(*t*)
4) evaluate *P*(*t*)
5) update *B*(*t*)
6) update *Q*(*t*) using *Q*-gates to shift to *B*(*t*)
7) **if** (global synchronization condition) **then** update *B*(*t*) globally
8) **end while**

To share the information gathered by different individuals to enhance the convergence speed, in a metaphor to particle swarm optimization algorithms, the *elitist* solution b_i^t of a *Q*-bit individual q_i^t is defined as and updated to be the best one among all the explored solutions by q_i^t and its neighbors in the latest N_l-generations. The neighborhood is defined in a topological sense. Moreover, the usage of the best solutions in the latest *N*-generations rather than in the total generations so far explored will guarantee the diversity of the population since an *elitist* solution can be kept as the attractor for a specific Q-individual only for a maximum number of N_l-generations.

To compromise the best balance between explorations and exploitations, the global information sharing in Step 7 is controlled by a parameter S_g. The value of this parameter is automatically updated from its maximal to minimal values in the searching process such that the search will favor exploration in the starting stage, and it will be incrementally shifted to bias the exploitations around some promising solutions in the final stage of the search.

To use the global statistical information extracted from the previous searches and build the probability distribution model of promising solutions, a quantum gate called rotation gate is used. To simplify the structure and implementation of the proposed QEA, the information sharing in the lowest level as commonly used in the available QEAs is deliberately removed.

II. Numerical Applications

A. Case Study One

To evaluate the performances of the proposed QEA, a high frequency inverse problem, the second example of [7], is firstly studied. The problem is to optimize the geometry of a 28-element antenna array for SideLobe levels (SLL) suppression in the region of [0^0, 180^0] and to prescribe nulls at 55^0, 57.5^0, 60^0, 120^0, 122.5^0, 126^0. After 10645 iterations in a typical run, the proposed algorithm converges to a final solution. The field pattern of the optimized antenna array of the proposed QEA is compared with that of the optimized array when a well developed PSO [7] is used to solve the same problem in Fig. 1. It is clear that the proposed QEA can suppress SLL to lower levels compared to that of the well developed PSO of [7] with the same main beam and null placements. It also produces a far closer pattern to the desired one when compared to the available optimizer, PSO.

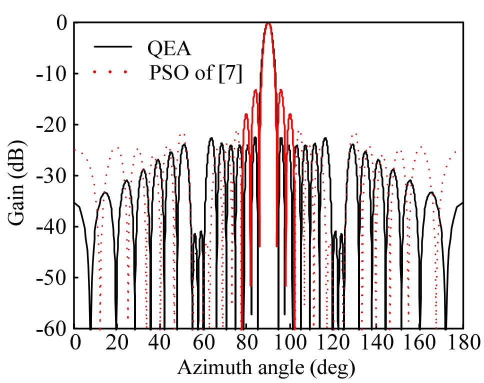

Fig.1. Comparison of field patterns for the optimized antenna array obtained using the proposed QEA and the well developed PSO of [7].

B. Case Study Two

A low frequency inverse problem, the Team Workshop problem 22 of a superconducting magnetic energy storage configuration with 8 free parameters as reported in [8] (Fig. 2), is solved by using the proposed QEA as a second application example. In the numerical experiments, the performance parameters are determined based on a two-dimensional finite element analysis. Under such conditions, the proposed algorithm is employed for searching the global optimal solution of the SMES device. Table I tabulates the finally optimized results of a typical run of the proposed QEA as well as the best ones obtained so far by IGTE [8]. It can be seen that the proposed QEA can find close approximations to the so far searched best ones with a small number of iterations in a relatively short span of time.

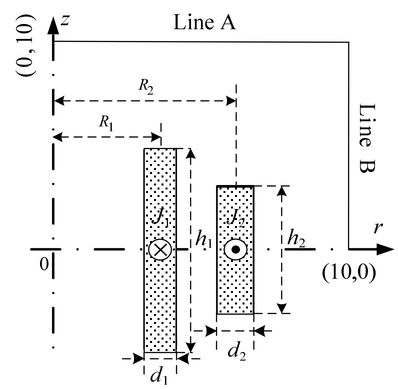

Fig. 2. The schematic diagram of a SMES.

III. References

[1] Dos Santos, Coelho L., and Alotto, P, "Multiobjective electromagnetic optimization based on a nondominated sorting genetic approach with a chaotic crossover operator," *IEEE Trans. Magn.*, vol.44, Iss. 6, Jun, 2008, pp. 1078-1081.

[2] Guimaraes, F.G., Campelo, F., Igarashi, H., Lowther, D.A., Ramirez J.A., "Optimization of cost functions using evolutionary algorithms with local learning and local Search," *IEEE Trans. Magn.*, vol.43, Iss. 4, Apr. 2007, pp. 1641-1644.

[3] Han, K. H., and Kim, J. H., "Quantum-inspired evolutionary algorithm for a class of combinatorial optimization," *IEEE Trans. Evol. Comput.*, vol. 6, Iss. 6, Dec. 2002, pp. 580-593.

[4] Lozano, J. A., Zhang, Q., and Larranaga, P., "Guest editorial: special issue on evolutionary algorithm based on probabilistic models," *IEEE Trans. Evol. Comput.*, vol. 13, Iss. 6, Dec. 2009, pp. 1197-1198.

[5] Platel, M. D., Schliebs, S., and Kasabov, N., "Quantum-inspired evolutionary algorithm: a multimodel EDA," *IEEE Trans. Evol. Comput.*, vol. 13, Iss. 6, Dec. 2009, pp. 1218-1232.

[6] Rieffel, E., and Polak, W., "An introduction to quantum computing for non-physicists," *ACM Computing Surveys*, vol. 32, pp. 300-335, 2000.

[7] Khodier, M. M., and Christodoulou, C. G., "Linear array geometry synthesis with minimum sidelobe level and null control using particle swarm optimization," *IEEE Transactions on Antennas and Propagation*, vol. 53, pp. 2674-2679, 2005.

[8] TEAM optimization benchmark problem 22 [online], available at: *www.igte.tugraz.at/archive/team/index.htm*.

TABLE I

Performance Comparison of the Proposed Method and the IGTE Solution for the SMES Configuration

Results	R_1(m)	R_2(m)	$h_1/2$(m)	$h_2/2$(m)	d_1(m)	d_2(m)	J_1(MA/m^2)	J_2(MA/m^2)	f_{obj}	No. iterations
Proposed	1.5704	2.1012	0.7844	1.4191	0.6001	0.2570	17.3358	-12.9658	6.7239×10^{-3}	2186
By IGTE	1.5703	2.0999	0.7846	1.4184	0.5943	0.2562	17.3367	-12.5738	5.5203×10^{-3}	/