Research Article

Quantum Genetic Algorithm Based on Qutrits and Its Application

Valerii Tkachuk (ORCID)

Vasyl Stefanyk Precarpathian National University, 57 Shevchenko Str., Ivano-Frankivsk 76018, Ukraine

Correspondence should be addressed to Valerii Tkachuk; tkachukv0@gmail.com

Typical approaches to designing quantum genetic algorithms are based on a concept of a qubit, a two-level quantum system. But many-valued quantum logic is more perspective from the point of view of the computational power. This paper proposes a quantum genetic algorithm based on a three-level quantum system in order to accelerate evolutionary process. Simulation using a set of standard test functions proves that the given algorithm is more effective and precise than the conventional quantum genetic algorithm.

1. Introduction

Evolutionary algorithms are one of the most widely used methods for solving the global optimization or search problem [1]. In its classical implementation, the smallest unit of information is a bit, a structure which may be in one of the following states: 0 and 1. It is important that a classical bit does not allow any intermediate states.

Quantum genetic algorithm (QGA) is a new evolutionary algorithm based on a combination of quantum computation and conventional genetic algorithms technology [2, 3]. This algorithm can be applied to the same set of problems the conventional genetic algorithm is used for, but it allows significantly accelerating the evolutionary process through quantum parallelization and entanglement of the quantum state. Probabilistic mechanism of the quantum computations combined with the evolutionary algorithm provides a global search for a solution with a rapid convergence and a small value of population. These algorithms have demonstrated their effectiveness for solving combinatorial and functional optimization problems, mechanical engineering optimization problems, image processing, and many others [4–8].

QGA main concepts were first proposed by Narayanan and Moore [9]. The basic unit of information used in a quantum computations is a qubit, a quantum system, which may be in the $|0\rangle$ basis state or the $|1\rangle$ basis state. Quantum nature of the qubit lies in the superposition principle, under which the qubit may be in any state which is a linear combination of basis states:

$$|q\rangle = \alpha_0 |0\rangle + \alpha_1 |1\rangle \qquad (1)$$

with a normalization constraint:

$$\alpha_0{}^2 + \alpha_1{}^2 = 1. \qquad (2)$$

As a consequence of the superposition principle, qubit state space is incomparably greater than state space of a classical bit. Information, stored in $\alpha_0{}^2$ and $\alpha_1{}^2$ amplitudes, is the quantum part of the information. Practically, $\alpha_0{}^2$ та $\alpha_1{}^2$ are numbers that specify the probability amplitudes of a qubit being in the states $|0\rangle$ and $|1\rangle$, respectively.

The outcome of quantum measurement is a qubit in one of the basis states. However, it is important that the result of the measurement is not deterministic, like it is in the classical calculation, but probabilistic.

QGA uses a matrix representation for practical implementation:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix},$$
$$|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \qquad (3)$$

Qubit (1) state in this representation can be defined as

$$|q\rangle = \begin{pmatrix} \alpha_0 \\ \alpha_1 \end{pmatrix}. \tag{4}$$

A structured set containing N qubits represents a quantum chromosome. State vector of such chromosome is a superposition state of 2^N basis states of the register $|i_1, i_2, \ldots, i_N\rangle$, $i_1, i_2, \ldots, i_N = \{0, 1\}$:

$$|\psi\rangle = \sum_{k=0}^{N} \alpha_{i_1, i_2, \ldots, i_N} |i_1, i_2, \ldots, i_N\rangle. \tag{5}$$

All the information about a qubits system is represented by state vector $|\psi\rangle$. The only thing that can be done to such a system is transforming the initial state vector $|\psi\rangle$ into a new state $|\psi'\rangle$ by applying a quantum gate. So quantum genetic algorithm is a state transition process from some starting state to an ending state using a quantum gate algorithm. The information about the problem solution is only contained by the vector of the system ending state $|\psi'\rangle$.

The main disadvantage appearing during quantum state rotation is the need to use a table to look up the rotating angle, which limits the universality of the search. Fixed rotating angle has a negative impact on the search speed, which is the reason for QGA sometimes being implemented as an adaptive process for determining the rotating angle. To enhance the local searching ability and to get out of the local optimal solutions, GDA can be extended with conventional operators traditionally used in the classical genetic algorithm, such as quantum mutation operation and quantum disaster operation [10].

2. Quantum Genetic Algorithm Based on Qutrits (QGA_3)

Qudit is a structure with several, more than two, states, which can also be used to encode a chromosome. A qudit is basically a unit of quantum information, which may be in any of n states or in any superposition of those.

Ternary quantum logic is the simplest type of the many-valued logic. The basic unit of memory is called a qutrit. It has three basis states, $|0\rangle$, $|1\rangle$, and $|2\rangle$. The state of the qutrit can be represented as a superposition in the form of a linear combination:

$$|q\rangle = \alpha |0\rangle + \beta |1\rangle + \gamma |2\rangle \tag{6}$$

with a normalization constraint:

$$\alpha^2 + \beta^2 + \gamma^2 = 1. \tag{7}$$

Matrix representation:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix},$$

$$|1\rangle = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix},$$

$$|2\rangle = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}. \tag{8}$$

For the practical implementation, qutrit (6) state can be represented as

$$|q\rangle = \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix}. \tag{9}$$

A system containing N qutrits has 3^N basis states (as opposed to 2^N states for binary logic). When switching from binary to many-valued logic, we get an exponential increase in the number of basis states, which leads to an increase in the algorithm performance for the same search accuracy.

Quantum chromosome length is determined by the desired search accuracy ε, search area $[x_{\min}, x_{\max}]$, and the number of quantum system basis states:

$$N = \log_n \left(\frac{x_{\max} - x_{\min}}{\varepsilon} + 1 \right), \tag{10}$$

where n is a number of quantum system basis states.

With $n = 2$, search accuracy $\varepsilon = 10^{-6}$, and search area $[-1, 1]$, the length of the quantum chromosome has to be at least 21 qubits, while with $n = 3$ only 14 qutrits are required.

2.1. Qutrit Encoding. Matrix representation of a quantum chromosome is a structure which consists of N qutrits.

α_1	α_2	α_3	α_4	α_5	$\cdots$	$\cdots$	$\cdots$	α_N
β_1	β_2	β_3	β_4	β_5	$\cdots$	$\cdots$	$\cdots$	β_N
γ_1	γ_2	γ_3	γ_4	γ_5	$\cdots$	$\cdots$	$\cdots$	γ_N

Here $\{\alpha_i, \beta_i, \gamma_i\}$ determine the state of ith qutrit, and N qutrits determine one individual in the population. The initial state of the qutrits holds no information about the system state, so the easiest way to initialize the base population is by setting all probability state amplitudes to be equal to one another [2]. This means that at the end of the initialization each qutrit is in the state

$$|q\rangle = \frac{1}{\sqrt{3}} |0\rangle + \frac{1}{\sqrt{3}} |1\rangle + \frac{1}{\sqrt{3}} |2\rangle. \tag{11}$$

2.2. Observation of Genes. Classical information about the problem's solution is located in the ending state vector $|\psi\rangle'$ and can be retrieved as a result of quantum observation. Qutrit in one of the basis states, which is obtained as a result of quantum observation, is a classical representation of a quantum chromosome.

Based on the approach described in [11], the following pseudocode for quantum N-qutrit chromosome state observation can be suggested (Algorithm 1). The product of its

```
(1)  for i in 1, ..., N do
(2)     r ← random number in range [0, 1]
(3)     if r < ⌈a_i⌉² then
(4)        p ← 0
(5)     else if r < ⌈a_i⌉² + ⌈β_i⌉² then
(6)        p ← 1
(7)     else
(8)        p ← 2
(9)     end if
(10) end for
```

ALGORITHM 1: Qutrit state observation.

FIGURE 1: Average distribution of the population individuals n over the search area on different stages of the evolution using an example of F_4 function optimization for 100 runs.

work provides the ground state p (0, 1 or 2) with a probability of α_i^2, β_i^2 and γ_i^2, respectively.

The algorithm leads to the quantum chromosome being transformed into its classical representation in the ternary numbering system.

$$
\begin{array}{ccccccccc}
\alpha_1 & \alpha_2 & \alpha_3 & \alpha_4 & \alpha_5 & \cdots & \cdots & \cdots & \alpha_N \\
\beta_1 & \beta_2 & \beta_3 & \beta_4 & \beta_5 & \cdots & \cdots & \cdots & \beta_N \\
\gamma_1 & \gamma_2 & \gamma_3 & \gamma_4 & \gamma_5 & \cdots & \cdots & \cdots & \gamma_N
\end{array}
$$

Observation

$$\downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow$$

$$0 \quad 2 \quad 0 \quad 1 \quad 1 \quad 2 \quad 1 \quad 0 \quad 0$$

2.3. Quantum Rotating Gates. All information about the problem and the algorithm for solving it is contained in the quantum gate, so its algorithm is the key issue of any quantum genetic algorithm construction.

The pseudocode for the genetic operator updating the qutrit state (quantum gate of the system evolution) as a result of probability amplitudes α_i, β_i, and γ_i modification is presented in Algorithm 2.

The algorithm's concept can be illustrated by the image given below. The vertical lines set the probability amplitudes α, β, and γ. If the classical representation of the population best individual has, for example, a 2 on the position i, all the amplitudes, except for γ, are multiplied by the factor μ. Its value lies in a range between 0 and 1 and is determined empirically. As a result, only the probability amplitude γ, which corresponds to the population best individual on the previous evolution stage, will increase. At the same time, probability amplitudes α and β will decrease.

$$b \in \{0 \quad 1 \quad 2\}$$

$$\downarrow \quad \downarrow \quad \uparrow$$

$$|q\rangle = [\alpha \quad \beta \quad \gamma]$$

It is important, that the described quantum gate algorithm eliminates the necessity to use a lookup table, which exists in the traditional QGA.

Initialized population ($t = 0$) includes all the possible solutions with the same probability. This means that the
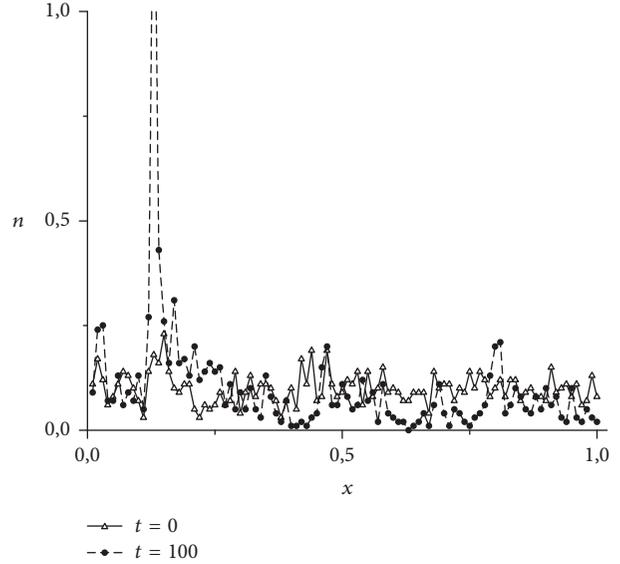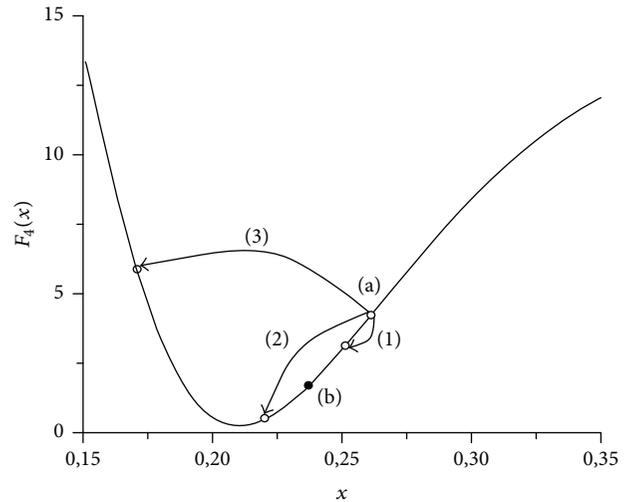

FIGURE 2: Possible outcomes of applying a quantum gate operator. Here (b) is the best individual of the population; (a) current individual of the population.

QGA process starts with a random search. As the evolution progresses in time, the distribution undergoes changes (Figure 1).

The mechanisms for the local search with $t > 100$ are illustrated in Figure 2. They demonstrate the role of μ parameter. Its fixed value has a bad impact on the convergence speed. So, it is desirable to use an adaptive quantum gate operator. The listed data shows that it can influence the individual a in three possible ways:

(1) The fitness of the individual is improved, but it does not become the best;

```
(1)  for i in 1, . . . , N do
(2)    bestamp ← ith gene of the population's best individual
(3)    sum ← 0
(4)    for amp in {0, 1, 2} do
(5)      if amp ≠ bestamp then
(6)        q′[amp] = μ · q
(7)        sum ← sum + q′[amp]²
(8)      end if
(9)    end for
(10)   q′[bestamp] ← √(1 − sum)
(11) end for
```

ALGORITHM 2: Update of quantum states.

```
(1)  for i in 1, . . . , s do
(2)    if the chromosome is not the best then
(3)      if disaster condition then
(4)        for i in 1, . . . , N do
(5)          αᵢ = βᵢ = γᵢ = 1/√3
(6)        end for
(7)      end if
(8)    end if
(9)  end for
```

ALGORITHM 3: Quantum disaster.

(2) The fitness of the individual is improved, and it becomes the best;

(3) The fitness of the individual is worsened.

This means that the bigger the number of individuals located in the area of the best individual b, the more effective the process of the local convergence.

To sum up, QGA starts with the global search and switches to the local search automatically due to the change in the structural characteristics of the population individuals' distribution.

2.4. Quantum Disaster Operation. Quantum population evolution may cause its falling into local optimal solution. This question is particularly important when the search area is large (big number of parameters) or when the fitness function has a complex topology around the global optimum.

Taking into consideration the small number of the quantum population, which is usually not bigger than $s = 10$ individuals, getting out of the local optimal solution can be achieved by a quantum disaster operation [12]. It can be summarized as initialization of some population individuals, except for the best, with an initial state. Such operation "virtually" enlarges the size of the population and extends the area for optimal value search.

In the simplest case, quantum disaster operation can be implemented as Algorithm 3.

3. Simulation Test

QGA_3 based on a qutrits system is implemented using C++ programming language. The tests have been performed on the Intel Celeron CPU G1840 2.80 GHz, 4.0 GB RAM, Windows 7 Professional environment.

The majority of the quantum algorithms use similar primary ideas and approaches, which are based on the qubit representation of the population. For example, some algorithms based on the quantum computation ideas are presented in [13]. The studies of the quantum algorithms and the possibilities of their modification and improvement are carried out in [14, 15]. Other potential approaches also include the option of the transition to qutrits representation of the population and can be considered as directions for further research. Considering this, the only algorithms compared by effectiveness and performance were QGA and the suggested QGA_3.

The analysis and performance evaluation have been performed on a number of test functions [16]:

De Jong Function

$$F_1 = 100\left(x_1^2 - x_2^2\right) + \left(1 - x_1\right)^2, \tag{12}$$
$$-2.048 \le x_i \le 2.048, \ i = 1, 2.$$

The function has one global minimum solution:

$$F_1(1, 1) = 0. \tag{13}$$

Coldstein Price Function

$$\begin{aligned} F_2 = &\left[1 + \left(x_1 + x_2 + 1\right)^2\right. \\ &\left. \cdot \left(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2\right)\right] \cdot \left[30\right. \\ &+ \left(2x_1 - 3x_2\right)^2 \\ &\left. \cdot \left(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2\right)\right], \end{aligned} \tag{14}$$
$$-2 \le x_i \le 2, \ i = 1, 2.$$

The function has one global minimum solution:

$$F_2(0, -1) = 3. \tag{15}$$

Schaffer Function

$$F_3 = 0.5 + \frac{\sin^2\left(x_1^2 + x_2^2\right) - 0.5}{\left[1.0 + 0.001\left(x_1^2 + x_2^2\right)\right]^2},$$ (16)

$$-100 \le x_i \le 100, \; i = 1, 2$$

The function has one global minimum solution:

$$F_3(0, 0) = 0.$$ (17)

Monopole and Six-Peak Camelback Function

$$F_4 = 10 + \frac{\sin(1/x)}{0.1 + (x - 0.16)^2}, \quad 0 \le x \le 1.$$ (18)

The function has one global maximum solution:

$$F_4(0.1275) = 19.8949.$$ (19)

Dual-Pole and Six-Peak Camelback Function

$$F_5 = \left(4 - 21x_1^2 + \frac{1}{3}x_1^4\right)x_1^2 + x_1 x_2 + \left(-4 + 4x_2^2\right)x_2^2,$$ (20)

$$-3 \le x_i \le 3, \; i = 1, 2.$$

The function has two global minimum solutions:

$$F_5(-0.0898, 0.7126) = F_5(0.0898, -0.7126)$$

$$= -1.031628.$$ (21)

Multipeak Positive Function

$$F_6 = e^{-0.001x}\cos^2(0.8x), \quad -2 \le x \le 5.$$ (22)

The function has one global maximum solution:

$$F_6(0) = 1.$$ (23)

Ackley Function

$$F_7 = -20e^{-0.2\sqrt{(1/n)\sum_{i=1}^{n} x_i^2}} - e^{(1/n)\sum_{i=1}^{n}\cos(2\pi x_i)}$$

$$+ 22.71828, \quad -5 \le x_i \le 5, \; i = 1, 2.$$ (24)

The function has one global minimum solution:

$$F_7(0, 0) = 0.$$ (25)

Rastrigin Function

$$F_8 = 10n + \sum_{i=1}^{n}\left[x_1^2 - 10\cos(2\pi x_i)\right],$$ (26)

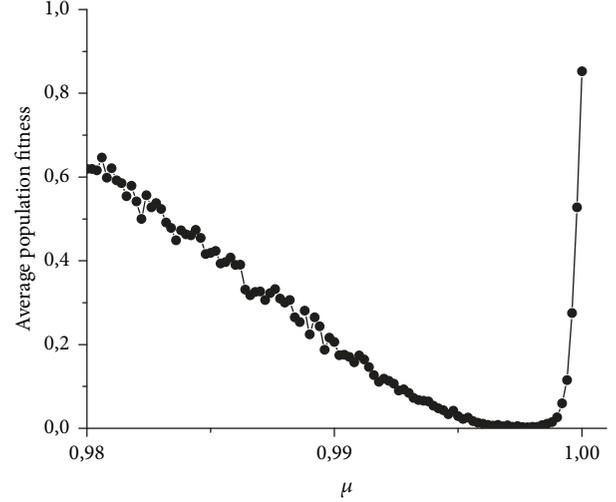$$-5.12 \le x_i \le 5.12, \; i = 1, 2.$$

FIGURE 3: Average population fitness as a function of μ parameter. Evolution time $t = 500$.

The function has one global minimum solution:

$$F_8(0, 0) = 0.$$ (27)

Optimization and Results. For QGA and QGA_3 simulation testing the following typical parameters were used: population size $s = 10$, total generations of iteration $t = 500$, and precision $\varepsilon = 1 \cdot 10^{-6}$.

The Choice of μ Parameter. The value of parameter μ determines the evolution of the quantum system and has a key role in the algorithm. It determines the balance between the global (quantum observation operation) and local (quantum gate) search for the found solution specification.

To choose the optimal μ value we have investigated its influence on the behavior of the main parameters which characterize the quantum system evolution, such as the average fitness of the best individual and the standard deviation of function value over 1000 algorithm runs. Its characteristic behavior for the example of F_8 function optimization is presented in Figures 3 and 4. The quantum disaster operation was not applied in this case.

The research shows that if $\mu < 0.95$, the determinative operation in the population evolution is the quantum observation. Global search and the best individual's approximation to the optimal value take place because of its probabilistic nature. At the same time, the quantum rotation angle is greater than $15°$ and is too big to perform a local search.

As μ value increases, the rotation angle approaches its optimal value $\sim 4°$. It corresponds to $\mu \sim 0.996 \div 0.998$: with this parameter value the fitness function and standard deviation reach the minimum value (see Figures 3 and 4).

Quantum system behavior for $\mu > 0.998$ is critically dependent on its evolution time (see Figure 5). Indeed, with $t = 250$ system simply does not have enough time to relax to the ending state, which corresponds to the fitness function optimal value. Enlarging the value of evolution time parameter $t = 1000$ practically simply causes the

TABLE 1: Experimental results for F_1–F_8 test functions optimization.

Function	QGA				QGA_3			
	f_{opt}	$\overline{f}$	sd	$\overline{t}$ (sec)	f_{opt}	$\overline{f}$	sd	$\overline{t}$ (sec)
F_1	0.000	0.011	0.011	0.035	0.000	0.005	0.012	0.024
F_2	3.000	3.050	0.132	0.034	3.000	3.000	0.001	0.022
F_3	0.000	0.004	0.012	0.041	0.000	0.001	0.002	0.029
F_4	19.895	19.894	0.001	0.014	19.895	19.895	0.000	0.011
F_5	−1.031	−1.021	0.008	0.034	−1.031	−1.031	0.000	0.024
F_6	1.000	1.000	0.000	0.018	1.000	1.000	0.000	0.012
F_7	0.008	0.035	0.071	0.038	0.000	0.001	0.003	0.022
F_8	0.013	0.051	0.152	0.037	0.000	0.002	0.032	0.024

TABLE 2: Experimental results for test functions F_7 and F_8, parameters count $i = 6$.

Function	QGA				QGA_3			
	f_{opt}	$\overline{f}$	sd	$\overline{t}$ (sec)	f_{opt}	$\overline{f}$	sd	$\overline{t}$ (sec)
F_7	2.648	4.167	0.507	0.117	0.018	0.186	0.170	0.075
F_8	12.117	23.171	5.050	0.107	0.408	6.872	3.621	0.073
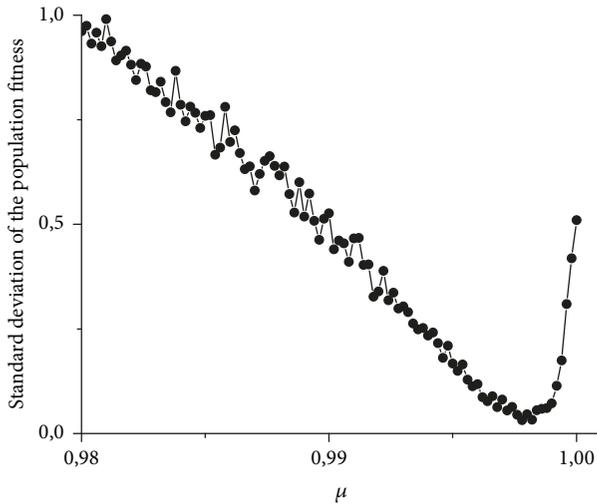
FIGURE 4: Standard deviation of the population fitness as a function of μ parameter. Evolution time $t = 500$.

— $t = 750$
···· $t = 500$
– – · $t = 250$

FIGURE 5: Average fitness of the best individual as a function of μ parameter for different evolution time parameter values.

algorithm running time to increase, with no optimal value improvement.

The role of the quantum disaster operation can be illustrated by Figure 6. The results of the F_8 function optimization are given for parameters count $i = 6$. Taking into consideration the F_8 complex topology around the optimum, the size of the population $s = 50$.

Based on the given analysis it has been established that for QGA_3 without the quantum disaster operation $\mu \approx 0.998$ is the optimal value. Adding the quantum disaster operation to the QGA_3 algorithm allows producing the best result with $\mu \approx 0.975$.

The optimization results of the F_1–F_8 test functions with a standard set of input parameters are listed in Table 1.

In Table 1, f_{opt} denotes the function value of optimum; $\overline{f}$ is the average value of the best individual fitness function; sd is standard deviation of fitness function; $\overline{t}$ is average elapsed time need for the algorithm to run.

A search for F_7 and F_8 functions global minimum has been carried out separately, using parameter count $i = 6$ and a standard set of other algorithm parameters (see Table 2). The results reveal that QGA_3 is more, but not enough, effective, especially in regard to function F_8.

The result can be significantly improved by enlarging the population size at least to $s = 50$ and adding a quantum

TABLE 3: The influence of the quantum disaster operation on QGA_3 efficacy.

Function	QGA_3				QGA_3 added disaster operation			
	f_{opt}	$\overline{f}$	sd	$\overline{t}$ (sec)	f_{opt}	$\overline{f}$	sd	$\overline{t}$ (sec)
F_7	0.001	0.095	0.118	0.384	0.000	0.001	0.001	0.399
F_8	0.019	3.135	2.353	0.369	0.000	2.025	1.889	0.375
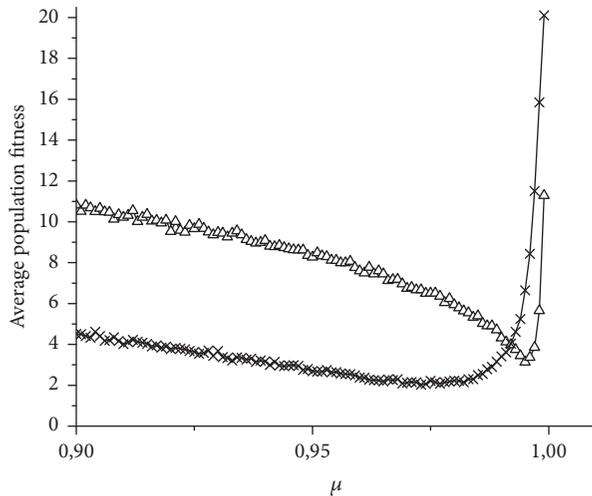
FIGURE 6: Quantum disaster operation influence on the average population fitness as a function of μ parameter: $\triangle$ –QGA_3 without the disaster operation; $\times$ – QGA_3 with the disaster operation.

disaster operation, with the other parameters remaining unchanged. The results of experiments after the described improvements to the algorithm are listed in Table 3.

4. Conclusions

In this paper a new quantum genetic algorithm QGA_3 has been proposed. The algorithm is based on the idea of using qutrits for the representation of the population individuals. A simple algorithm, which does not require a lookup table, has been used for constructing the evolution operator which is responsible for the quantum system state updates.

The experiments have been performed using a typical set of test functions. On the basis of the experimental results we have concluded that QGA_3 allows decreasing the computational time while being more productive and having better convergence, compared to the conventional QGA.

We have also obtained that, in a case with a lot of parameters to be optimized, the QGA_3 efficacy can be improved by adding a quantum disaster operation. In the same time, the increase in algorithm operating time does not exceed 4%.

Further promising investigations include the research of n-valued quantum logic units, qudits and their possible applications for representation of the population individuals. This is expected to be a subject of separate studies.
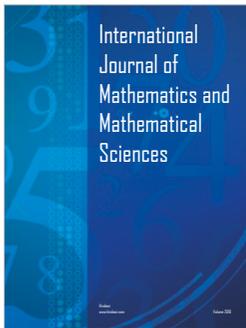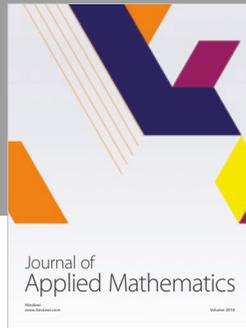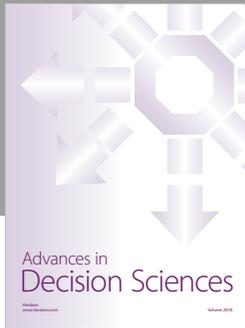
Conflicts of Interest

The author declares that there are no conflicts of interest regarding the publication of this paper.

References

[1] J. H. Holland, *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*, University of Michigan Press, Oxford, UK, 1975.

[2] K. H. Han and J. H. Kim, "Genetic quantum algorithm and its application to combinatorial optimization problem," in *Proceedings of the IEEE International Conference on Evolutionary Computation*, pp. 1354–1360, San Diego, Calif, USA, July 2000.

[3] . Kuk-Hyun Han, . Kui-Hong Park, . Ci-Ho Lee, and . Jong-Hwan Kim, "Parallel quantum-inspired genetic algorithm for combinatorial optimization problem," in *Proceedings of the 2001 Congress on Evolutionary Computation*, pp. 1422–1429, Seoul, South Korea.

[4] P. Jantos, D. Grzechca, and J. Rutkowski, "Evolutionary algorithms for global parametric fault diagnosis in analogue integrated circuits," *Bulletin of the Polish Academy of Sciences—Technical Sciences*, vol. 60, no. 1, pp. 133–142, 2012.

[5] H. Talbi, M. Batouche, and A. Draa, "A quantum-inspired evolutionary algorithm for multiobjective image segmentation," *International Journal of Mathematical, Physical and Engineering Sciences*, vol. 1, pp. 109–114, 2007.

[6] B. Li and L. Wang, "A hybrid quantum-inspired genetic algorithm for multiobjective flow shop scheduling," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 37, no. 3, pp. 576–591, 2007.

[7] T. W. Lau, C. Y. Chung, K. P. Wong, T. S. Chung, and S. L. Ho, "Quantum-inspired evolutionary algorithm approach for unit commitment," *IEEE Transactions on Power Systems*, vol. 24, no. 3, pp. 1503–1512, 2009.

[8] L. Su-Hua, W. Yao-Wu, P. Lei, and X. Xin-Yin, "Application of quantum-inspired evolutionary algorithm in reactive power optimization," *Relay*, vol. 33, pp. 30–35, 2005.

[9] A. Narayanan and M. Moore, "Quantum-inspired genetic algorithms," in *Proceedings of the IEEE International Conference on Evolutionary Computation*, pp. 61–66, 1996.

[10] H. Wang, J. Liu, J. Zhi, and C. Fu, "The improvement of quantum genetic algorithm and its application on function optimization," *Mathematical Problems in Engineering*, vol. 2013, Article ID 730749, 10 pages, 2013.

[11] R. Nowotniak and J. Kucharski, "Higher-order quantum-inspired genetic algorithms," in *Proceedings of the 2014 Federated Conference on Computer Science and Information Systems, FedCSIS 2014*, pp. 465–470, Poland, September 2014.

[12] H. Miao, H. Wang, and Z. Deng, "Quantum genetic algorithm and its application in power system reactive power

optimization," in *Proceedings of the International Conference on Computational Intelligence and Security (CIS '09)*, vol. 1, pp. 107–111, Beijing, China, 2009.

[13] G. Zhang, "Quantum-inspired evolutionary algorithms: a survey and empirical study," *Journal of Heuristics*, vol. 17, pp. 303–351, 2011.

[14] M. Soleimanpour-moghadam, H. Nezamabadi-pour, and M. M. Farsangi, "A quantum inspired gravitational search algorithm for numerical function optimization," *Information Sciences*, vol. 267, pp. 83–100, 2014.

[15] G. Zhang and H. Rong, "Real-Observation Quantum-Inspired Evolutionary Algorithm for a Class of Numerical Optimization Problems," in *Computational Science – ICCS 2007*, vol. 4490 of *Lecture Notes in Computer Science*, pp. 989–996, Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[16] Y. Sun and H. Xiong, "Function optimization based on quantum genetic algorithm," *Research Journal of Applied Sciences, Engineering & Technology*, vol. 7, no. 1, pp. 144–149, 2014.