Proceedings of Machine Learning Research vol 278:1–6, 2025 2025 2nd International Conference on Machine Learning and Intelligent Computing

Research on Multi-population Quantum Genetic Algorithm Based on Optimal Computation Allocation Technology

Ziyuan Guo 1175918197@QQ.COM
Institute of Artificial Intelligence Industry,ChangChun University of Architecture and Civil Engineering, No. 1519 Huoju Road, Chaoyang District, Changchun City, Jilin Province, China

Editors: Nianyin Zeng, Ram Bilas Pachori and Dongshu Wang

Abstract

Quantum genetic algorithms have proven their unique superiority in dealing with stochastic optimization problems. In this paper, we propose an innovative multi-population quantum genetic algorithm, which is based on optimal computational resource allocation techniques. By carefully optimizing the initialization strategy of the population and introducing the concept of an elite population, combined with optimal computational resource allocation techniques, we have significantly improved the performance of the algorithm on stochastic optimization problems. After a series of experimental verifications, we found that the proposed algorithm surpasses traditional quantum genetic algorithms and other classical optimization algorithms in terms of convergence speed and solution accuracy.

Keywords: Quantum Genetic Algorithm, Optimal Computation Allocation Technology, Risk Assessment.

1. Introduction

With the rapid advancement of technology, optimization problems have become increasingly important in many fields. Quantum genetic algorithms, as an efficient intelligent optimization method, have demonstrated outstanding performance when dealing with deterministic problems. However, when facing stochastic problems, traditional algorithms such as classical genetic algorithms and gradient descent methods often struggle to find the optimal solution. To address this challenge, optimal computational resource allocation technology has emerged. It saves computational re-sources through orderly comparisons, significantly increasing the likelihood of obtaining satisfactory solutions. Based on this, this paper proposes an improved quantum genetic algorithm that integrates optimal computational resource allocation technology, aiming to further enhance its performance and efficiency in stochastic optimization problems.

2. Algorithm Improvement

2.1. Population Initialization Improvement

Traditional quantum genetic algorithms typically initialize the probability amplitudes of all genes to equal values when initializing the population, which may lead to an excessively large initial search range for the population and slow convergence speed (Kiszka and Wozabal, 2022). This paper proposes a phased population initialization method based on Shannon's entropy theorem, initializing multiple populations according to the entropy change of the population, rapidly compressing the global search space, thereby improving the convergence speed of the algorithm.

Quantum bits can exist in a superposition of both quantum states simultaneously:

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

Among them, (α, β) is a pair of complex numbers representing probability amplitudes and satisfying $|\alpha|^2 + |\beta|^2 = 1$, where $|\alpha|^2$ is the probability of tending towards the quantum state 0, and $|\beta|^2$ is the probability of tending towards the quantum state 1 (Zhang et al., 2022).

Quantum genetic algorithms typically employ binary encoding, using 0 and 1 to rep-resent gene positions, where the gene encoding m parameters with multiple quantum bits is as follows:

$$q_j^t = \begin{pmatrix} \alpha_{11}^t & \alpha_{12}^t & \dots & \alpha_{1k}^t & \alpha_{21}^t & \alpha_{22}^t & \dots & \alpha_{2k}^t & \alpha_{m1}^t & \alpha_{m2}^t & \dots & \alpha_{mk}^t \\ \beta_{11}^t & \beta_{12}^t & \dots & \beta_{1k}^t & \beta_{21}^t & \beta_{22}^t & \dots & \beta_{2k}^t & \beta_{m1}^t & \beta_{m2}^t & \dots & \beta_{mk}^t \end{pmatrix} \tag{2}$$

Here, q_j^t represents the chromosome of individual j of the t-th generation, k is the number of qubits encoding each gene, and m is the number of genes in the chromosome.

If all genes (α_i^t, β_i^t) of all chromosomes in the initial population $Q(t_0)$ are initialized to be $(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})$, then the chromosome represents an equal probability superposition of all possible states:

$$\left|\varphi_{q_j}^0\right\rangle = \sum_{k=1}^{2^m} \frac{1}{\sqrt{2^m}} |S_k\rangle \tag{3}$$

Among them, S_k is the k-th state of the chromosome, which is represented by a binary string of length m $(x_1, x_2, x_3, \dots x_m)$, where $x_i = 0, 1$, for i $= 1, 2, \dots, m$ (Hejazi et al., 2017).

According to Shannon's entropy theorem, there is:

$$H = -\sum_{i=1}^{n} P(x_i) \log_2 P(x_i) \tag{4}$$

Assuming that a random number Rand is generated during encoding, where $Rand \in (0, 1)$, if $Rand > \alpha_i^2$, then the current bit binary encoding is 1, and if $Rand \leq \alpha_i^2$, then the current bit binary encoding is 0. thus there is:

$$H(\delta) = -\sum_{h=0}^{m} C_m^h \left[\delta^h (1-\delta)^{m-h} \cdot \log_2 \delta^h (1-\delta)^{m-h}\right] \tag{5}$$

Where δ is the value of α_i^2, m is the number of gene loci, and h is the number of gene loci where Randδ. Obviously, the above formula has a maximum value when $\alpha_i^2 = 1/2$, meaning that when all genes (α_i, β_i) in the population are initialized to $(1/\sqrt{2}, 1/\sqrt{2})$, the entropy is the highest and the initial search range of the population is the largest. However, when $\alpha_i^2 = 0$ or 1, the search range is the smallest. Therefore, when initializing multiple populations, one should not directly assign values to n populations. Instead, one should initialize all genes of the first population to $(1/\sqrt{2}, 1/\sqrt{2})$, ensuring the search range, and then use the high-quality individuals obtained through population updating to initialize the next one or more populations, until all n populations are completed (Kim and Ahn, 2022).

2.2. Introduction of elite populations

To further enhance the performance of the algorithm, this paper introduces an elite population into the multi-population quantum genetic algorithm. The elite population consists of the optimal solutions from the iterative processes of each sub-population, providing reference rotation angles for population evolution, assisting in the evolution of sub-populations, thereby improving the convergence ability and search efficiency of the population (Cheng et al., 2022).

On the basis of the original sub-population size, an additional population is generated based on the optimal solutions during the iteration process of each sub-population. After the elite population is generated, reference rotation angles $(\theta_1, \theta_2, \ldots \theta_i)$ are provided for population evolution (Al-Betar et al., 2021). The calculation rule for all population rotation angles becomes:

$$\theta_{Ei}^{t} = \sum_{i=1}^{N} (C_i \cdot \theta_i) \tag{6}$$

When the population of elites is generated, the individual fitness gap is very small, and individual fitness is an evaluation of the objective function (Liu and Liang, 2022). A very small fitness gap proves that the solutions are very similar, so all populations $\{X_1, X_2, \ldots X_m\}$ can be approximated as independent and identically distributed sequences (Xiao and Gao, 2018). According to the Central Limit Theorem of Levy-Lindberg, it follows that:

$$\lim_{x \to \infty} P(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{\frac{1}{2}t^2} dt \tag{7}$$

2.3. Optimal Computational Load Allocation

Optimal computational resource allocation is based on the probability of selecting objectives, allocating more simulation runs to feasible solutions with higher satisfaction levels to enhance the efficiency of limited resource utilization. This paper applies this method to a multi-population quantum genetic algorithm, achieving computational resource conservation and acceleration of algorithm search speed through probabilistic risk model evaluation (Jiao et al., 2020).The traditional simulation optimization mini-mum value model is described as follows:

$$\begin{gathered} f(\theta) = E[M(\theta, \xi)] \\ \text{s.t. } \theta \in \Theta \end{gathered} \tag{8}$$

θ represents any feasible solution, and ξ denotes the error caused by uncertain factors. If the model variables follow a normal distribution, ξ is the variance of the normal distribution. Assuming that the variables follow a normal distribution is for mathematical convenience, theoretical support, adaptability to practical applications, and simplification of the model. This helps to enhance the operability and solution efficiency of the model, while reasonably addressing the randomness and uncertainty in practical problems. f usually refers to the mathematical expectation or mean of M sampling, serving as a performance indicator for the objective function. Θ represents the space of feasible solutions (Pan et al., 2023).

3. Multi-population Quantum Genetic Algorithm for Stochastic Optimization Problems

3.1. Test Function

In order to verify the improved algorithm and optimal computational load allocation technique, the following three classical continuous complex functions are selected as test functions.

Test function 1:

$$f(x) = x_1^2 + x_2^2 - 10\cos(2\pi x_1) - 10\cos(2\pi x_2)$$
$$\text{s.t}\begin{cases} -10 \leq x_i \leq 10, i = 1, 2 \\ x_i \sim N\left(\overline{x_i}, \frac{\overline{x_i^2}}{T \cdot D}\right) \end{cases}$$

Test function 2:

$$f(x) = \sum_{i=1}^{D}\left(x_i^2 - 5\right)$$
$$\text{s. t}\left(x_i \sim N\left(\bar{x}_i, \frac{\bar{x}_i}{T \cdot D}\right), -10 \leq \bar{x}_\imath \leq 10\right)$$

Test function 3:

$$f(x_1, x_2) = m_1 + m_2 + x_1\sin(4\pi x_1) + x_2\sin(20\pi x_2)$$
$$\text{s. } t\begin{cases} -3.0 \leq x_1 \leq 12.1 & m_1 \sim N\left(x_1, \frac{x_1}{T \cdot D}\right) \\ 4.1 \leq x_1 \leq 5.8 & m_2 \sim N\left(x_2, \frac{x_2}{T \cdot D}\right) \end{cases}$$

3.2. Experimental Results

Here, T is set to 5, and D is the number of variables. All test functions have random factors in their independent variables, which are simulated and compared using five different algorithms: MIQGA-OCBA, MIQGA, QGA, MSGA, and SGA. The other four algorithms all use Monte Carlo simulation, with each sampling performed 10 times. The three quantum genetic algorithms all employ the H_ε quantum rotation gate, with the initial rotation angle θ_i set to 0.01π. For MSGA and SGA, the crossover rate P_c is 0.8, the mutation rate P_m is 0.05, the selection method uses a roulette wheel, the crossover method uses single-point crossover, and the mutation method uses random inversion of one gene within the chromosome. MIQGA-OCBA, MIQGA, and MSGA all have 10 subpopulations, with a generational gap GGNP of 0.9 between populations. Each algorithm is run independently 10 times to take the average, obtaining the optimal value Y, the values of the independent variable X, and the absolute error.The experimental results are shown in Table 1, Table 2, and Table 3.

Therefore, it is evident that MIQGA-OCBA not only converges at the fastest speed but also yields values that are more accurate and closer to theoretical values. Consequently, MIQGA-OCBA's algorithm performance in uncertain environments is significantly superior to MIQGA, QGA, MSGA, and SGA, while MIQGA's performance is also slightly better than the other three algorithms in most cases.

Table 1: Results of Test Function 1.

Algorithm	Optimal value Y	X (x1,x2)	Absolute error
MIQGA-OCBA	201.4265	(-9.54976,-9.54976)	0
MIQGA	201.4252	(-9.5472,-9.5498)	0.0007
QGA	201.4030	(9.5572,9.5575)	0.0043
MSGA	201.4237	(-9.5525,-9.5481)	0.0027
SGA	201.4233	(9.5528,9.5477)	0.0028

Table 2: Results of Test Function 2.

Algorithm	Optimal value Y	X (x1,x2)	Absolute error
MIQGA-OCBA	190.0	(10.00, 10.00)	0
MIQGA	190.1472	(9.996,9.992)	0.1463
QGA	190.133	(9.99,9.99)	0.133
MSGA	189.941	(-10,9.992)	0.065
SGA	189.9925	(9.9989,9.9962)	0.0096

Table 3: Results of Test Function 3.

Algorithm	Optimal value Y	X (x1,x2)	Absolute error
MIQGA-OCBA	35.2000	(12.1000,5.72500)	0
MIQGA	35.1988	(12.1000,5.72450)	0.0014
QGA	35.0578	(12.1000,5.72507)	0.1520
MSGA	3.9169	(12.0973,5.72528)	0.2823
SGA	3.9810	(12.0975,5.72574)	0.2195

4. Conclusion and Outlook

This paper proposes a multi-population quantum genetic algorithm based on optimal computational resource allocation technology and applies it to stochastic optimization problems. Experimental results indicate that the algorithm outperforms traditional quantum genetic algorithms and other classical optimization algorithms in terms of convergence speed and solution accuracy. Future research will further improve the algorithm by considering more complex factors in practical problems to enhance its applicability and robustness.

References

Mohammed Azmi Al-Betar, Mohammed A. Awadallah, Ali Asghar Heidari, Huiling Chen, Habes Al-khraisat, and Chengye Li. Survival exploration strategies for harris hawks optimizer. *Expert Systems with Applications*, 168:114243, 2021. doi: https://doi.org/10.1016/j.eswa.2020.114243.

Ji-Wei Cheng, Feng Zhang, and Xiang-Yang Li. Nonlinear amplitude inversion using a hybrid quantum genetic algorithm and the exact zoeppritz equation. *Petroleum Science*, 19(3):1048–1064, 2022. doi: https://doi.org/10.1016/j.petsci.2021.12.014.

Taha Hossein Hejazi, Mirmehdi Seyyed-Esfahani, and Mehdi Taghinia. Fully pca-based approach to optimization of multiresponse-multistage problems with stochastic considerations. *Applied Mathematical Modelling*, 45:530–550, 2017. doi: https://doi.org/10.1016/j.apm.2017.01.011.

Shan Jiao, Guoshuang Chong, Changcheng Huang, Hanqing Hu, Mingjing Wang, Ali Asghar Heidari, Huiling Chen, and Xuehua Zhao. Orthogonally adapted harris hawks optimization for parameter estimation of photovoltaic models. *Energy*, 203:117804, 2020. doi: https://doi.org/10.1016/j.energy.2020.117804.

Jun Suk Kim and Chang Wook Ahn. Size-efficient sparse population for strictly structured quantum genetic algorithm. *Future Generation Computer Systems*, 135:159–171, 2022. doi: https://doi.org/10.1016/j.future.2022.04.030.

Adriana Kiszka and David Wozabal. A stability result for linear markovian stochastic optimization problems. *Math. Program.*, 191(2):871–906, February 2022. doi: 10.1007/s10107-020-01573-3.

Xiaolong Liu and Tongying Liang. Harris hawk optimization algorithm based on square neighborhood and random ar-ray. *Control and Decision*, 37(10):2467–2476, 10 2022.

S. Pan, T. K. Gupta, and K. Raza. Batts: a hybrid method for optimizing deep feedforward neural network. *PeerJ Comput Sci*, 9:e1194, 2023. doi: 10.7717/peerj-cs.1194.

Hui Xiao and Siyang Gao. Simulation budget allocation for selecting the top-m designs with input uncertainty. *IEEE Transactions on Automatic Control*, 63(9):3127–3134, 2018. doi: 10.1109/TAC.2018.2791425.

Liwei Zhang, Yule Zhang, Jia Wu, and Xiantao Xiao. Solving stochastic optimization with expectation constraints efficiently by a stochastic augmented lagrangian-type algorithm. *INFORMS Journal on Computing*, 34(6):2989–3006, 2022. doi: 10.1287/ijoc.2022.1228.