Multiscale Quantum Gradual Approximation Algorithm: An Optimization Algorithm With a Step-by-Step Approximation Strategy

KUN CHENG1,2, PENG WANG3, AND ZHENDONG LI4,5

1University of Chinese Academy of Sciences, Beijing 100049, China
2Chengdu Institution of Computer Application, Chinese Academy of Sciences, Chengdu 100864, China
3School of Computer Science and Technology, Southwest Minzu University, Chengdu 610041, China
4School of Information Engineering, Ningxia University, Yinchuan 750021, China
5Collaborative Innovation Center for Ningsia Big Data and Artificial Intelligence (Co-founded by Ningsia Municipality and Ministry of Education), Yinchuan 750021, China

Corresponding author: Peng Wang (qhoalab@163.com)

This work was supported in part by the Fundamental Research Funds for the Central Universities, Southwest Minzu University, under Grant 2019NYB22 and Grant 2020NYB18, and in part by the National Natural Science Foundation of China under Grant 60702075 and Grant 71673032.

ABSTRACT In quantum swarm intelligence algorithms, the tunneling effect of the particles is determined by the potential energy acting on the particles. The tunneling effect of the particles affects the global search ability and convergence speed of the algorithm. Quantum algorithms with a single potential energy are prone to premature convergence under certain complex test functions. In this paper, we propose a multiscale quantum gradual approximation algorithm (MQGAA), which simply uses different approximation strategies to obtain different potential energy functions, to solve the premature problem of the optimization algorithm. In the MQGAA, particles undergo a transition from an unconstrained state to a constrained state at each scale. To demonstrate the effectiveness of the proposed algorithm, experiments are carried out with several common and effective stochastic algorithms on N-dimensional double-well potential functions and classical benchmark functions. We also use the Wilcoxon rank test to detect the performance of MQGAA. The experimental results show that the algorithm using a step-by-step approximation strategy achieves a better optimization performance on some complex test functions.

INDEX TERMS Taylor approximation, unconstrained state, constrained state, multiscale, multiscale quantum harmonic oscillator algorithm.

I. INTRODUCTION

Swarm intelligence is a kind of bionic algorithm inspired by the organization inherent in natural biological behavior. The algorithm simulates the mutual cooperation between natural biological groups. It is an algorithm in which one or more individuals with simple intelligence show advanced intelligence through the cooperation between them. In the past few decades, many swarm intelligence algorithms have been proposed, such as evolutionary strategy [1], particle swarm optimization [2], simulated annealing [3], genetic algorithm [4], differential evolution [5], ant colony optimization [6], fireworks algorithm [7], etc. These swarm intelligence algorithms have developed rapidly since they were proposed. The improvement of numerous algorithms greatly improves their performance.

In the swarm intelligence algorithm, particles exhibit aggregation. This aggregation, which means that the differences between individuals are limited, is a basic characteristic of swarm intelligence algorithms. In quantum mechanics, aggregation is described by the bound states of particles. The formation of the bound states is caused by the existence of an attractive potential field at the center of the particle motion. In recent years, many scholars have applied quantum theory to swarm intelligence algorithms. The quantum annealing algorithm (QA) is developed from the classical simulated annealing algorithm [8]. The algorithm uses the fluctuation characteristics of particles in quantum theory to design the
algorithm. The quantum tunneling effect of the quantum wavefunction will enable particles to cross the obstacles that cannot be crossed in classical physics, and the target system will be optimized by simulating this process. Sun et al. proposed a quantum behaved particle swarm optimization (QPSO) by combining the quantum system with particle swarm [9]. In the QPSO algorithm, a quantized attractive potential field is set up as the bound particle of the constrained state, which makes the particle move toward the aggregation state. The multiscale quantum harmonic oscillator optimization algorithm (MQHOA) is based on the physical meaning of the quantum theoretical wavefunction. The ground state of the quantum system corresponds to the probability distribution of the optimal solution of the objective function. In MQHOA, particles are attracted by the potential field of a harmonic oscillator [10]. The bare bones fireworks algorithm (BBFWA) is a simplified fireworks algorithm. The particles in BBFWA are evenly distributed in the definition domain, which is simple and easy to implement [11].

In the optimization process, we establish a quantized potential field to bind particles such that not only will the algorithm exhibit aggregation but also will the particles be able to appear in any position in the space with a certain probability. It is important to obtain and select a suitable potential constrained particle in the optimization algorithm.

In quantum mechanics, the bound state of a particle is determined by the wavefunction. The wavefunction can be solved by the Schrödinger equation, which is shown in (1), where \( \psi(x) \) is the attraction potential of particles in a quantum system, i.e., the potential energy function.

\[
E \psi(x) = (-\frac{\hbar^2}{2m} \frac{\partial^2}{\partial x^2} + v(x))\psi(x)
\]  

(1)

The objective function \( f(x) \) of the optimization problem corresponds to the potential energy \( v(x) \) in the Schrödinger' equation, and the square of the absolute value of the wavefunction \( \psi(x) \) corresponds to the probability distribution of the optimal solution in the solution space. In the process of optimization, particles will eventually concentrate near the optimal solution. That is, the probability distribution of particles will eventually be stable, which corresponds to the wavefunction of particles in the ground state in the quantum model. When we obtain the ground state wavefunction corresponding to the objective function, we can obtain the position of the optimal solution. When \( v(x) \) is complex, it is difficult to solve the wavefunction with the Schrödinger’ equation. Consequently, we need to simplify the objective function of the optimization.

The MQHOA used the second-order Taylor approximation to simplify the optimization problems. MQHOA is an efficient quantum optimization algorithm that is proposed based on the wavefunction in quantum physics. MQHOA converts the optimization problem into solving the wavefunction of a time-dependent quantum system. The wavefunction defined in MQHOA reflects the potential field force on the particles in MQHOA. When the optimization problem is approximated by a second-order Taylor expansion, the wavefunction in MQHOA is the wavefunction of a quantum harmonic oscillator. The particles in MQHOA are bounded by the potential field of the quantum harmonic oscillator [10].

In quantum mechanics, the potential energy of a quantum system exerts a force field on the particles, which determines the tunneling effect of the particles in the system. Different potential energies determine different tunneling effects of the particles. By comparing the tunneling effects of different quantum systems, we propose a new algorithm called the multiscale quantum gradual approximation algorithm (MQGAA). In the MQGAA, at each scale, the particles undergo a transition from an unconstrained state to a constrained state. Particles in the unconstrained state have a strong global search ability, whereas the local search ability of the constrained particles is strong. After satisfying certain conditions, the particles can be transformed into the constrained state to enhance the local search ability, thus speeding up the convergence of the algorithm. The change in the particle state is realized by the change in the potential energy. The first- and second-order Taylor approximations are applied to the new algorithm to obtain quantum systems with different potential energies. Compared with other algorithms, this effectively enhances the exploitation ability in the global area; however, the convergence speed of the algorithm is lower.

To verify the performance of MQGAA, N-dimensional double-well potential functions and some classical benchmark functions are used as test functions. The double-well potential is one of a number of quartic potentials in quantum mechanics, in quantum field theory and elsewhere for the exploration of various physical phenomena and mathematical properties. The one-dimensional double-well function is an ideal potential well model objective function that has a globally optimal region and a locally optimal region. Many scholars have used the one-dimensional double-well potential function as the objective function to analyze the performance of quantum algorithms. The double-well potential function is used to analyze the performance of QA as a heuristic optimization algorithm [12]. Reference [13] used the one-dimensional double-well potential function as the ideal potential well model in quantum physics to study the annealing properties of wavefunctions in quantum systems. Reference [14] used the double-well potential function as a simple one-dimensional case study system to investigate the basic behavior and performance of simulated QA in comparison with classical annealing (CA). To further verify the performance difference between MQGAA and MQHOA, we use the bilateral Wilcoxon rank test to detect the experimental data of the two algorithms through the method described in [15].

This paper is organized as follows. In Section II, we introduce the basic definitions and formulas of Taylor’s formula. We briefly introduce the principle of MQHOA and analyze the approximation strategy of MQHOA in Section III. In Section IV, we give the approximation strategy of MQGAA.
and compare the tunneling effects of different approximation strategies for corresponding wavefunctions. The framework of MQGAA is also described in Section IV. In Section V, the efficiency of MQGAA is evaluated on benchmark functions with different characteristics and N-dimensional double-well potential functions. The Wilcoxon rank test is also in this section. The simulation results and performance analysis validate the effectiveness of the proposed approach. In Section VI, conclusions and future work are outlined.

II. TAYLOR’S FORMULA

In mathematics, Taylor’s formula describes the value of a function in its vicinity with the information of a point. If the function is sufficiently smooth, Taylor’s formula can construct a polynomial to approximate the value of the function in the neighborhood of this point by using the corresponding multiples of these derivatives as coefficients when the derivatives of the function at a certain point are known. In practical applications, Taylor’s formula needs to be truncated, taking only a finite number of terms. The Taylor series of the function is sufficiently smooth, Taylor’s formula can construct a polynomial to approximate the value of the function in its vicinity with the information of a point. If the function values near the optimal solution are known, we need to simplify complex functions as the simplest polynomial.

To conveniently describe the approximation strategy of the algorithm, we define Taylor’s first-order approximation and second-order approximation, which are described as follows. $f_1(x)$ in Formula (3) and $f_2(x)$ in Formula (4) are two finite terms of the Taylor formula shown in Formula (2).

A. FIRST-ORDER TAYLOR APPROXIMATION

We intercept the first two terms of Formula (2) for the approximation of $f(x)$ which is shown in Formula (3).

$$f_1(x) ≈ \frac{f(x_0)}{0!} + \frac{f'(x_0)}{1!} (x - x_0)$$

B. SECOND-ORDER TAYLOR APPROXIMATION

We intercept the first three terms of Formula (2) for the approximation of $f(x)$. The approximation of $f(x)$ is shown in (4)

$$f_2(x) ≈ \frac{f(x_0)}{0!} + \frac{f'(x_0)}{1!} (x - x_0) + \frac{f''(x_0)}{2!} (x - x_0)^2$$

III. MULTISCALE QUANTUM HARMONIC OSCILLATOR ALGORITHM

The MQHOA with the energy level stabilizing process proposed in 2016 [17] is an efficient quantum optimization algorithm that was proposed based on the wavefunction in quantum physics.

In MQHOA, the optimization problem is transformed into the problem of solving for the time-dependent wavefunction. The Schrödinger equation is shown in (1). The objective function $f(x)$ of the optimization problem corresponds to the potential energy $v(x)$ in the Schrödinger equation, and the square of absolute value of the wavefunction $|\psi(x)|$ corresponds to the probability distribution of the optimal solution in the solution space. The optimal solutions correspond to the lowest potential energy. In other words, the ground state wavefunction of the quantum system reflects the distribution of the optimal solutions. The Schrödinger equation can only be used to obtain the wavefunctions of some simple potential energy functions. Therefore, we need to simplify complex objective functions by approximation.

In MQHOA, we approximate the objective function $f(x)$ by Taylor’s formula about the optimal solution $x_0$. The Taylor expansion is shown in formula (2). $x_0$ is the extremum point of the objective function; thus, $f'(x_0) = 0$. By substituting $f'(x_0) = 0$ for Formula (4), we obtain Formula (5).

$$f(x) ≈ \frac{f(x_0)}{0!} + \frac{f''(x_0)}{2!} (x - x_0)^2$$

The solution of the wavefunction of $f(x)$ is simplified to the solution of the wavefunction of the quantum harmonic oscillator system. The probability density of the wavefunction that is shown in (6) can be derived from Formula (1) [18].

$$|\psi_0(x)|^2 = \frac{a}{\sqrt{\pi}} \exp(-a^2x^2)$$

The wavefunction $\psi_{s_i}(x)$ of MQHOA shown in (7) is defined as a superposition of $n$ Gaussian probability density functions with $\mu_i$ as the sampling center, and $\sigma_s$ represents the current scale [19].

$$|\psi_{s_i}(x)|^2 = \sum_{i=1}^{n} |\psi_i(x)|^2 ≈ \sum_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma_s} \exp\left(-\frac{(x - \mu_i)^2}{2\sigma_s}\right)$$
IV. MULTISCALE QUANTUM GRADUAL APPROXIMATION ALGORITHM

The wavefunction in MQHOA can ensure that the solutions can escape from local optima with a certain probability; however, for certain complex problems, the solutions often fall into a local optimum, which makes MQHOA converge prematurely. To address this problem, we propose the MQGAA algorithm.

MQGAA is a quantum optimization algorithm based on the wavefunction, which consists of two key components: a diffusion (D) process and a multiscale (M) process. In the D process, the particles can be in one of two states: an unconstrained state or a constrained state. Particles in the unconstrained state have a strong global search ability. When a certain condition is reached, the particles transit from the unconstrained state to the constrained state. Particles in the constrained state have a weaker global search ability but a stronger local search ability, which can make the algorithm converge quickly. The transformation of the particle state is realized by the transformation of the wavefunction, and different wavefunctions are obtained by different approximation strategies for the objective function.

A. APPROXIMATION STRATEGY OF MQGAA

For an optimization algorithm, a stronger tunnel effect does not necessarily provide better performance by the algorithm, as we are uncertain that the position of the particle at the next moment will be closer to the optimal solution. In MQGAA, we adopt a step-by-step Taylor approximation strategy. At each scale, the first-order Taylor approximation is used to make the particles sufficiently disperse. When the sampled particles reach the metastable state, the second-order Taylor approximation is applied to the objective function to make the particles converge quickly.

We approximate the objective function $f(x)$ by Taylor’s formula at the optimal solution $x_0$. Because $f'(x_0) = 0$, Formula (3) can be simplified to Formula (8), where $C$ is constant.

$$f(x) \approx f(x_0) = C$$

The wavefunction of $f(x)$ can be obtained by solving the wavefunction of a free particle. The wavefunction of a free particle is shown in Formula (9).

$$\psi(x, t) = \psi_0 e^{-i \frac{p^2}{2m}(E - \mu)}$$

The probability density of a free particle described in Formula (10) is the square of the wavefunction shown in Formula (9).

$$|\psi|^2 = \psi \psi^* = |\psi_0|^2$$

The wavefunction $\psi_{\sigma_i}(x)$ corresponding to the first-order Taylor approximation of the objective function, which we used in MQGAA, shown in (11), is defined as a superposition of $n$ uniform distributions. The $a$ and $b$ are the upper and lower limit of the domain. The wavefunction corresponding to the second-order Taylor approximation of the objective function is the same as MQHOA, which is shown in (7).

$$|\psi_{\sigma_i}(x)|^2 = \sum_{i=1}^{n} |\psi_i(x)|^2 = \begin{cases} \frac{1}{R - L}, & L \leq x \leq R \\ 0, & x < L \text{ or } x > R \end{cases}$$

B. COMPARISON OF DIFFERENT WAVEFUNCTIONS

We compare the tunneling effects of different quantum systems by using the probability of solutions located outside the local optimum. The greater the probability of locating outside of the local optimum region is, the stronger the tunneling effect of the quantum system. As mentioned above, the wavefunctions corresponding to the first-order Taylor approximation and the second-order Taylor approximation are shown in Formulas (11) and (7), respectively.

The comparison of quantum tunneling effects is shown in Figure 1. The objective function $f(x)$, which we used here, is the one-dimensional double-well potential function. The expression of $f(x)$ is shown in Formula (12), where $V = 6$, $a = 2$, $\delta = 1$. $\psi_1(x)$ and $\psi_2(x)$, which are shown in (13), are the wavefunctions corresponding to the functions after the first- and second-order Taylor approximations of $f(x)$, respectively.

$$f(x) = V \left(\frac{x^2 - a^2}{d^2}\right)^2 + \delta x$$

$$\psi_1(x) = \frac{1}{R - L}, L \leq x \leq R$$

$$\psi_2(x) = \frac{1}{\sqrt{2\pi\sigma_s}} \exp \left(-\frac{(x - \mu_i)^2}{2\sigma_i^2}\right)$$

When the solution locates at point $A$, the algorithm falls into a local optimum. The problem of comparing the ability of algorithms to escape from local optima is transformed into a problem of comparing the quantum tunneling effects of particles from region $A$ to region $B$. We use the probabilities of the particle locating at point $A$ to the left region of point $C$, which are shown in the shadows of Figure 1, to represent the tunneling effect. Substituting the values in Figure 1 into Formula (13), we obtain Formula (14). We use Formula (14) to calculate two probabilities. Through numerical calculation, we obtain $p_1 = 0.5317$ and $p_2 = 0.2323$. The tunneling effect of the quantum system based on the first-order Taylor approximation of the objective function is stronger.

$$p_1 = \int_{-3}^{0.19} \psi_1(x) dx = \int_{-3}^{0.19} \frac{1}{R - L}, \text{where } L = -3, R = 3$$

$$p_2 = \int_{-3}^{0.19} \psi_2(x) dx = \int_{-3}^{0.19} \frac{1}{\sqrt{2\pi\sigma_s}} \exp \left(-\frac{(x - \mu_i)^2}{2\sigma_i^2}\right)$$

C. THE FRAMEWORK OF MQGAA

Algorithm 1 describes the MQGAA pseudocode. The notations are represented in Table 1. Similar to MQHOA, MQGAA is also a quantum optimization algorithm based on
As shown in Algorithm 1, when $\Delta \sigma < 1.5 \sigma_s$, the global diffusion reaches the stable state; when $\Delta \sigma > \sigma_s$, the local diffusion becomes stable. Choosing the right time to switch between global diffusion and local diffusion can not only improve the local search ability of MQGA, but also ensure the efficiency. We use experiments to compare the effects of different switching conditions in diffusion stages on the performance of MQGAA. The experimental results are shown in subsection V-B. When the diffusion is stable, the worst particle will be replaced by the mean of all particles.

As discussed in [20], the M process is a necessary and problem-independent process for optimization algorithms according to the principle of uncertainty. The M process is the variable sampling step size strategy. The MQGAA gradually reduces the search step to precisely obtain the globally optimal solutions. Large step sizes correspond to a global search, and small step sizes correspond to a local search. Different scales determine the different accuracy levels of the solutions.

**V. SIMULATION AND DISCUSSION**

In this section, we first introduce the experimental environment, the comparison algorithms and test functions used in the experiment. Then, we analyze the switching conditions of the diffusion stage by experiment. Subsequently, to prove the effectiveness of the MQGAA, two experiments are conducted on test functions with different characteristics.

One experiment is conducted to compare the global search abilities between MQGAA, MQHOA-SMC and MQHOA on the double-well potential function. The second experiment is conducted to compare with MQHOA-SMC, MQHOA, QPSO, SPSO2011, and BBFWA on classical benchmark functions. The 14 benchmark functions include 6 unimodal functions and 8 multi-modal functions. The functions $f_2$, $f_3$, $f_5$, $f_7$, $f_9$, and $f_{10}$ are unimodal functions. The last three functions are rotated multi-modal functions. The remaining 5 functions are multi-modal functions. In this paper, the orthogonal matrix $M$ is generated by Salomon’s method [21]. The function name, ID, search space and optimum are listed in Table 2. After the experiments, we use the bilateral Wilcoxon rank to compare MQGAA with MQHOA and MQHOA-SMC to judge the effectiveness of the successive approximation strategy.

**A. PARAMETER SETTINGS**

The parameters used are listed in Table 3. The experimental environment is as follows: MATLAB 2014b, Windows Server 2016, Intel Xeon E5-2630 (2.4 GHz) CPU, 32G of RAM. The results are recorded for 51 independent runs for each function.

---

**Algorithm 1: MQGAA Pseudocode**

Initialize $k$, $MaxFE$, $[d_{min}, d_{max}]$, $\sigma_s$, $x_i$ ($i = 1, \ldots, k$), $\Delta \sigma$.

Calculate $\sigma$, $F_i = f(x_i)$.

while ($FE \leq MaxFE$) do

while ($\Delta \sigma > \sigma_s$) do

if $(\Delta \sigma > 1.5 \sigma_s)$ then

$\forall x_i$, generate $x_i' \sim N(x_i, \sigma_i^2)$.

if $f(x_i') < f(x_i)$ then $x_i = x_i'$.

Calculate $\sigma'$.

$\Delta \sigma = \sigma' - \sigma$

else

$\forall x_i$, generate $x_i' \sim N(x_i, \sigma_i^2)$.

if $f(x_i') < f(x_i)$ then $x_i = x_i'$.

Calculate $\sigma'$.

$\Delta \sigma = \sigma' - \sigma$

end

end

Update $\sigma = \sigma'$.

$x_{worse} = x_{mean}$.

$\sigma_s = \sigma_s/2$.

Output $x_{best}$, $F_{best}$.

---

**TABLE 1. Notations in Algorithm 1.**

| $k$ | the population size |
| $FE$ | the number of function evaluations |
| $MaxFE$ | the maximum number of function evaluations |
| $d_{min}$ | the lower bound of the feasible space |
| $d_{max}$ | the upper bound of the feasible space |
| $\sigma_s$ | the sampling size, the initial value of which is $d_{max} - d_{min}$ |
| $x_i$ | the $i$th particle of the population |
| $F_i$ | the fitness value of $x_i$ |
| $\sigma$ | the standard deviation of the population |
| $\Delta \sigma$ | the standard deviation of the population’s variance in two samplings |
| $s_{mean}$ | the mean position of the population |
| $x_{worse} = \min(F_i)$ | the particle position with the worst fitness value |
| $x_{best}$ | the particle position with $F_{best}$ |
TABLE 2. The classical benchmark functions.

| Function Name | ID | Benchmark Function | Search Space | Optimum |
|---------------|----|--------------------|--------------|---------|
| Griewank      | $f_3$ | $f(x) = \frac{1}{4000} \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos \left( \frac{x_i}{\sqrt{i}} \right) + 1$ | [-100,100] | 0       |
| Sphere        | $f_2$ | $f(x) = \sum_{i=1}^{n} x_i^2$ | [-100,100] | 0       |
| Zakhvor        | $f_3$ | $f(x) = \sum_{i=1}^{n} x_i^2 + \left( \sum_{i=1}^{n} 0.5 \omega_i \right)^2 + \left( \sum_{i=1}^{n} 0.5 \omega_i \right)^4$ | [-5,10] | 0       |
| Sum of Different Powers | $f_4$ | $f(x) = \sum_{i=1}^{n} (|x_i^{11}|^{11})$ | [-100,100] | 0       |
| Ellipsoidial   | $f_5$ | $f(x) = \sum_{i=1}^{n} (x_i - i)^2$ | [-100,100] | 0       |
| Levy          | $f_6$ | $f(x) = \frac{1}{1 + \left( \sum_{i=1}^{n} \frac{1}{1 + 0.001 \sum_{i=1}^{n} x_i} \right)^2}$ | [-10,10] | 0       |
| Sum Squares   | $f_7$ | $f(x) = \sum_{i=1}^{n} x_i^2$ | [-10,10] | 0       |
| Alpine        | $f_8$ | $f(x) = \sum_{i=1}^{n} |x(i) + \sin(x(i)) + 0.1 \cdot x(i)|$ | [0,10] | 0       |
| Quadric       | $f_9$ | $f(x) = \sum_{i=1}^{n} (\sum_{j=1}^{i} x(j))^{1/2}$ | [-100,100] | 0       |
| Rotated Hyper-ellipsoid | $f_{10}$ | $f(x) = \sum_{i=1}^{n} \sum_{j=1}^{i} x(j)^2$ | [-65.536,65.536] | 0       |
| Weierstrass   | $f_{11}$ | $f(x) = \frac{1}{n \sum_{k=0}^{N} a^k \cos(2\pi b^k (x_i - 1)^2))}$ | $\sum_{k=0}^{N} a^k \cos(2\pi b^k (x_i - 1)^2))$ | $\sum_{k=0}^{N} a^k \cos(2\pi b^k (x_i - 1)^2))$ | 0       |

B. EXPERIMENTS ON THE SWITCHING CONDITION OF THE DIFFUSION STAGE

To analyze the influence of different switching conditions on the performance of the MQGAA, we choose different switching conditions of the diffusion stage, such as $\sigma_s, 1.25 \times \sigma_s, 1.5 \times \sigma_s, 1.75 \times \sigma_s, 2 \times \sigma_s, 2.5 \times \sigma_s$. In this subsection, we use the 14 benchmark functions shown in Table 2 as test functions. For the dimension of test functions, we choose 10 dimensions. The results are listed in Table 4. According to the results, different switching conditions have little effect on the performance of MQGAA. To balance the efficiency and global search ability of MQGAA, we choose the switching condition as $1.5 \times \sigma_s$.

C. EXPERIMENTS BASED ON DOUBLE-WELL POTENTIAL FUNCTION

The one-dimensional double-well potential function that is used in this paper is represented in Formula (12). To better determine the performance of the algorithm, we define a multidimensional double-well potential function. The higher the
TABLE 4. Comparison of MQGAA with different switching conditions of the diffusion stage on benchmark functions with 10 dimensions. The termination criterion is set to $FE \leq MaxFE$. The experiments are repeated 51 times for each method.

| Conditions | $\alpha_s$ | $\beta_s$ | $\gamma_s$ | $\delta_s$ | $\epsilon_s$ | $\nu_s$ |
|------------|------------|------------|------------|------------|------------|--------|
| State | 90.56% | 100.00% | 100.00% | 100.00% | 100.00% | 100.00% |
| Best | 0.80% | 1.00% | 1.00% | 1.00% | 1.00% | 1.00% |
| Mean | 3.24% | 3.02% | 3.02% | 3.02% | 3.02% | 3.02% |
| Std | 1.01% | 0.51% | 0.51% | 0.51% | 0.51% | 0.51% |

Table 6. Comparison of MQGAA, MQHOA-SMC, and MQHOA on the N-dimensional double-well potential function. The termination criterion is set to $FE \leq MaxFE$. The experiments are repeated 51 times for each method.

| Algorithm | SR | $50.82\%$ | $1.96\%$ | Best | $0.2290937$ | $7.169253$ |
|------------|----|-----------|---------|------|-------------|------------|
| MQGAA | $94.12\%$ | $50.00\%$ | $0.00\%$ | $1.7764E-15$ | $3.9935013$ |
| MQHOA-SMC | $94.68\%$ | $49.02\%$ | $0.00\%$ | $3.67211346$ | $7.6105523$ |
| MQHOA | $97.06\%$ | $49.02\%$ | $0.00\%$ | $0.00145636$ | $0.3198094$ |

D. EXPERIMENTS BASED ON THE CLASSICAL BENCHMARK FUNCTIONS

In this subsection, MQGAA is compared with MQHOA-SMC, MQHOA, QPSO, SPSO2011, and BBFWA. The experimental results are listed in Table 7, Table 8 and Table 9, corresponding to 10, 30 and 60 dimensions, respectively. The best results are marked in boldface. The parameters used are listed in Table 3.

Table 7 lists the results for 10 dimensions. For the unimodal functions $f_2, f_4, f_5, f_9$ and $f_{10}$, although MQGAA does not achieve higher accuracy, it can find the optimal solution in each of the 51 runs. For multi-modal functions, there are large numbers of local optima that are more difficult to locate. Therefore, the success rate is more reflective of the performance of the algorithm. Based on the results, MQGAA achieved success rates of 100% on three multi-modal functions. For the rotated multi-modal functions $f_{12}-f_{14}$, MQGAA achieved success rates of 100% on $f_{12}$ and $f_{13}$, as did most of the other algorithms. MQHOA-SMC achieved the best performance on $f_{14}$, with a success rate of 92.31%.
TABLE 7. Comparison of MQGAA, MQHOA-SMC, MQHOA, QPSO, SPSO2011, and BBFWA on benchmark functions with 10 dimensions. The termination criterion is set to $FE \leq \text{MaxFE}$. The experiments are repeated 51 times for each method.

| Algorithm | 10 | 20 | 30 | 40 | 50 | 60 |
|-----------|----|----|----|----|----|----|
| MQGAA | 93.51% | 100.00% | 100.00% | 100.00% | 100.00% | 100.00% |
| PFOS | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| QPSO | 91.25% | 100.00% | 100.00% | 100.00% | 100.00% | 100.00% |
| SPSO2011 | 91.25% | 100.00% | 100.00% | 100.00% | 100.00% | 100.00% |
| BBFWA | 91.25% | 100.00% | 100.00% | 100.00% | 100.00% | 100.00% |

TABLE 8. Comparison of MQGAA, MQHOA-SMC, MQHOA, QPSO, SPSO2011, and BBFWA on benchmark functions with 30 dimensions. The termination criterion is set to $FE \leq \text{MaxFE}$. The experiments are repeated 51 times for each method.

| Algorithm | 10 | 20 | 30 | 40 | 50 | 60 |
|-----------|----|----|----|----|----|----|
| MQGAA | 100.00% | 100.00% | 100.00% | 100.00% | 100.00% | 100.00% |
| PFOS | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% | 0.00% |
| QPSO | 100.00% | 100.00% | 100.00% | 100.00% | 100.00% | 100.00% |
| SPSO2011 | 100.00% | 100.00% | 100.00% | 100.00% | 100.00% | 100.00% |
| BBFWA | 100.00% | 100.00% | 100.00% | 100.00% | 100.00% | 100.00% |

Table 8 lists the results for 30 dimensions. For the unimodal functions $f_2$, $f_4$, $f_7$, and $f_{10}$, although MQGAA does not achieve higher accuracy, it can find the optimal solution in each of 51 runs. For $f_3$, MQGAA achieved the best results out of all the evaluation options. For $f_9$, only SPSO2011 and MQGAA find the optimal solution in 51 runs, with success rates of 100% and 51.92%, respectively. For multi-modal functions, MQGAA achieved a success rate of 100% on $f_1$, $f_3$, and $f_{11}$. For rotated multi-modal functions, MQGAA achieved a success rate of 100% on $f_{12}$ and $f_{14}$. Only SPSO2011 achieved a better performance on $f_{13}$, with a success rate of 5.88%; all other algorithms had a success rate of 0%.
TABLE 9. Comparison of MQGA, MQHOA-SMC, MQHOA, QPSO, SPSO2011, and BBFWA on benchmark functions with 60 dimensions. The termination criterion is set to \( FE \leq MaxFE \). The experiments are repeated 51 times for each method.

| Algorithm      | \( f_1 \) | \( f_2 \) | \( f_3 \) | \( f_4 \) | \( f_5 \) | \( f_6 \) | \( f_7 \) |
|----------------|---------|---------|---------|---------|---------|---------|---------|
| MQGA           | 100.00% | 100.00% | 0.00%   | 0.00%   | 0.00%   | 0.00%   | 62.75%  |
| QPSO           | 91.00%  | 92.00%  | 93.00%  | 94.00%  | 95.00%  | 96.00%  | 97.00%  |
| SPSO2011       | 73.00%  | 74.00%  | 75.00%  | 76.00%  | 77.00%  | 78.00%  | 79.00%  |
| MQHOA-SMC      | 90.00%  | 91.00%  | 92.00%  | 93.00%  | 94.00%  | 95.00%  | 96.00%  |
| MQHOA          | 88.71%  | 89.71%  | 90.71%  | 91.71%  | 92.71%  | 93.71%  | 94.71%  |
| BBFWA          | 78.71%  | 79.71%  | 80.71%  | 81.71%  | 82.71%  | 83.71%  | 84.71%  |

TABLE 10. The Wilcoxon rank of MQGA, MQHOA and MQHOA-SMC on benchmark functions with 10 dimensions. The termination criterion is set to \( FE \leq MaxFE \). The experiments are repeated 51 times for each method.

| Algorithm      | \( f_1 \) | \( f_2 \) | \( f_3 \) | \( f_4 \) | \( f_5 \) | \( f_6 \) |
|----------------|---------|---------|---------|---------|---------|---------|
| MQGA           | 3.40E-01| 3.62E-01| 5.06E-01| 4.62E-01| 5.06E-01| 4.62E-01|
| MQHOA          | 5.06E-01| 5.06E-01| 5.06E-01| 4.62E-01| 5.06E-01| 4.62E-01|
| MQHOA-SMC      | 4.62E-01| 5.06E-01| 5.06E-01| 4.62E-01| 5.06E-01| 4.62E-01|

we compare the seven functions of SPSO2011 with those of MQGA, we find that the functions of SPSO2011 are mainly the unimodal functions while the functions of MQGA are mostly multi-mode functions.

Compared with other algorithms, the performance of MQGA is more stable on 14 test functions, and for multi-mode function, the performance of MQGA is better than other algorithms. \( f_j \) has many regularly distributed, widespread local minima; thus, it is very difficult to find the true solution. For all of the 10, 30 and 60 dimensions, MQGA achieves the highest success rate on \( f_1 \) out of all tested algorithms. Compared with MQHOA-SMC and MQHOA, MQGA performs better on \( f_7 \), \( f_9 \) and \( f_11 \). This means that MQGA, which uses the step-by-step Taylor approximation strategy, has a stronger global search ability and can effectively avoid premature convergence.

MQGA uses two different wavefunctions to enhance the ability of the algorithm to escape from the local optimum when there are many local optimins. Higher dimensions cannot increase the number of local optimal solutions, so we did not experiment with a higher dimension.

E. THE BILATERAL WILCOXON RANK

We use the bilateral Wilcoxon rank test to detect the experimental data of MQGA, MQHOA and MQHOA-SMC by the method described in [15]. We use the experimental results of the benchmark functions for rank detection. The results of different dimensions are shown in Table 10 and Table 11. When the \( p \) value is less than the confidence value 0.05, MQGA is proved to be effective.

Table 10 lists the Wilcoxon rank results of MQGA, MQHOA and MQHOA-SMC for 10 dimensions. In Table 10 there are some results indicated as NaN. In MATLAB, NaN means calculation error, which usually occurs when the divisor or denominator is 0 or the data exceed the accuracy. Here, we ignore these results. In the results of the comparison of MQGA and MQHOA, the \( p \) value of 2 functions is less than the confidence value, including \( f_2 \) and \( f_{10} \). The \( p \) value of 12 functions is less than 0.05, except \( f_1 \) and \( f_{11} \) in the results of the comparison of MQGA and MQHOA-SMC.

Table 11 lists the Wilcoxon rank results of MQGA, MQHOA and MQHOA-SMC for 30 dimensions. In the results of the comparison of MQGA and MQHOA, the \( p \) value of 3 functions is less than the 0.05, including \( f_2 \) and \( f_{10} \). The \( p \) value of 11 functions is less than the
TABLE 11. The Wilcoxon rank of MQGAA, MQHOA and MQHOA-SMC on benchmark functions with 30 dimensions. The termination criterion is set to $FE \leq \text{MaxFE}$. The experiments are repeated 51 times for each method.

| Function | MQGAA | MQHOA | MQHOA-SMC |
|----------|-------|-------|-----------|
|          | $f_1$ | $f_2$ | $f_3$ | $f_4$ | $f_5$ | $f_6$ | $f_7$ | $f_8$ | $f_9$ | $f_{10}$ | $f_{11}$ | $f_{12}$ | $f_{13}$ | $f_{14}$ |
| p-value  | 5.07E-01 | 2.15E-01 | 1.37E-01 | 7.41E-02 | 5.48E-01 | 2.32E-01 | 5.89E-01 | 1.42E-01 | 9.35E-01 | 1.44E-01 | 6.62E-01 | 4.89E-01 | 1.11E-01 | 7.48E-01 |
| Function | $f_1$ | $f_2$ | $f_{11}$ | $f_{12}$ | $f_{13}$ | $f_{14}$ |
| p-value  | 2.15E-02 | 3.02E-01 | 6.12E-01 | 1.17E-01 | 1.81E-01 | 1.21E-01 | 2.15E-01 | 1.02E-01 | 9.35E-01 | 7.48E-01 | 1.74E-01 | 1.27E-02 | 5.07E-01 | 2.15E-01 | 3.02E-01 |

VI. CONCLUSION

This paper proposes an MQGAA with a step-by-step Taylor approximation strategy. Theoretical analysis and experimental results indicate that the new strategy enhances the global exploitation ability. Statistical analysis of the experimental results also shows that MQGAA effectively improves the robustness and exploitation ability of the original algorithm. Comprehensive simulations between MQGAA and some efficient meta-heuristic methods, including MQHOA-SMC, MQHOA, QPSO, SPSO2011 and BBFWA under different dimensionalities, are conducted on N-dimensional double-well potential functions and classical benchmark functions. The simulation results reveal that the MQGAA is a competitive algorithm.

In the near future, we will further theoretically study the approximation strategy used by MQGAA and apply it to solve real-world engineering optimization problems.

REFERENCES

[1] H. G. Beyer and H.-P. Schwefel, “Evolution strategies—A comprehensive introduction,” Natural Comput., vol. 1, no. 1, pp. 3–52, 2002.
[2] J. Kennedy, “Particle swarm optimization,” in Encyclopedia of Machine Learning, Springer, 2010, pp. 760–766.
[3] P. J. M. L. van Emile Aarts, “Simulated annealing: Theory and applications,” ACTA Applicandae Math., vol. 12, no. 1, pp. 108–111, 1987.
[4] D. E. Goldberg and J. H. Holland, “Genetic algorithms and machine learning,” Mach. Learn., vol. 3, nos. 2–3, pp. 95–99, 1988.
[5] R. Storn and K. Price, “Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces,” J. Global Optim., vol. 11, no. 4, pp. 341–359, 1997.
[6] M. Dorigo and L. M. Gambardella, “Ant colony system: A cooperative learning approach to the traveling salesman problem,” IEEE Trans. Evol. Comput., vol. 1, no. 1, pp. 53–66, Apr. 1997.
[7] Y. T. Zhu, “Fireworks algorithm for optimization,” in Advances in Swarm Intelligence, Berlin, Germany: Springer, 2010, pp. 335–364.
[8] R. Martonak, G. E. Santoro, and E. Tosatti, “Quantum annealing of the traveling salesman problem,” Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top., vol. 70, no. 5, 2004, Art. no. 057701.
[9] J. Sun, W. Xu, and B. Feng, “A global search strategy of quantum-behaved particle swarm optimization,” in Proc. IEEE Conf. Cybern. Intell. Syst., Dec. 2004, pp. 111–116.
[10] P. Wang, Y. Huang, C. Ren, and Y. M. Guo, “Multi-scale quantum harmonic oscillator for high-dimensional function global optimization algorithm,” Chin. J. Electron., vol. 41, no. 12, pp. 2468–2473, Dec. 2013.
[11] J. Li and Y. Tan, “The bare bones fireworks algorithm: A minimalist global optimizer,” Appl. Soft Comput., vol. 62, pp. 454–462, Jan. 2018.
[12] E. M. Inack and S. Pilati, “Simulated quantum annealing of double-well and multiwell potentials,” Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top., vol. 92, no. 5, Nov. 2015, Art. no. 053304.
[13] L. Stella, G. E. Santoro, and E. Tosatti, “Monte Carlo studies of quantum and classical annealing on a double well,” Phys. Rev. B, vol. 73, no. 14, Apr. 2006, Art. no. 144302.
[14] L. Stella, G. E. Santoro, and E. Tosatti, “Optimization by quantum annealing: Lessons from simple cases,” Phys. Rev. B, vol. 72, no. 1, Jul. 2005, Art. no. 014303.
[15] G. E. Noether, Introduction to Wilcoxon (1945) Individual Comparisons by Ranking Methods. New York, NY, USA: Springer, 1992.
[16] T. U. Department of Mathematics, Advanced Mathematics, 7th ed. Beijing, China: Higher Education Press, 2014.
[17] W. Peng and H. Yan, “MQHOA algorithm with energy level stabilizing process,” J. Commun., vol. 37, no. 7, pp. 79–86, 2016.
[18] D. J. Griffiths, Introduction to Quantum Mechanics. London, U.K.: Pearson, 2005.
[19] P. Wang and Y. Huang, “Physical model of multi-scale quantum harmonic oscillator optimization algorithm,” J. Frontiers Comput. Sci. Technol., vol. 9, no. 10, pp. 1271–1280, 2015.
[20] P. Wang, Multiscale Quantum Harmonic Oscillator Algorithm, nos. 52–53, Beijing, China: The People’s Posts and Telecommunications Press, 2016.
[21] R. Salomon, “Re-evaluating genetic algorithm performance under coordinate rotation of benchmark functions. A survey of some theoretical and practical aspects of genetic algorithms,” Biosystems, vol. 39, no. 3, pp. 263–278, Jan. 1996.
[22] M. Zambrano-Bigiarini, M. Clerc, and R. Rojas, “Standard particle swarm optimisation 2011 at CEC-2013: A baseline for future PSO improvements,” in Proc. IEEE Congr. Evol. Comput., Jun. 2013, pp. 2357–2344.
[23] P. Wang, X. Ye, B. Li, and K. Cheng, “Multi-scale quantum harmonic oscillator algorithm for global numerical optimization,” Appl. Soft Comput., vol. 69, pp. 655–670, Aug. 2018.
[24] B. Li, P. Wang, and J. Jin, “Multiscale quantum harmonic oscillator algorithm with strict metastability constraints for multi-modal optimization,” IEEE Access, vol. 7, pp. 17377–17388, 2019.