COMPLEX SYSTEM MODELING AND SIMULATION
ISSN 2096-9929 05/06 pp 334−353
Volume 2, Number 4, December 2022
DOI: 10.23919/CSMS.2022.0021

Quantum-Inspired Distributed Memetic Algorithm

Guanghui Zhang*, Wenjing Ma, Keyi Xing, Lining Xing, and Kesheng Wang

Abstract: This paper proposed a novel distributed memetic evolutionary model, where four modules distributed exploration, intensified exploitation, knowledge transfer, and evolutionary restart are coevolved to maximize their strengths and achieve superior global optimality. Distributed exploration evolves three independent populations by heterogenous operators. Intensified exploitation evolves an external elite archive in parallel with exploration to balance global and local searches. Knowledge transfer is based on a point-ring communication topology to share successful experiences among distinct search agents. Evolutionary restart adopts an adaptive perturbation strategy to control search diversity reasonably. Quantum computation is a newly emerging technique, which has powerful computing power and parallelized ability. Therefore, this paper further fuses quantum mechanisms into the proposed evolutionary model to build a new evolutionary algorithm, referred to as quantum-inspired distributed memetic algorithm (QDMA). In QDMA, individuals are represented by the quantum characteristics and evolved by the quantum-inspired evolutionary optimizers in the quantum hyperspace. The QDMA integrates the superiorities of distributed, memetic, and quantum evolution. Computational experiments are carried out to evaluate the superior performance of QDMA. The results demonstrate the effectiveness of special designs and show that QDMA has greater superiority compared to the compared state-of-the-art algorithms based on Wilcoxon's rank-sum test. The superiority is attributed not only to good cooperative coevolution of distributed memetic evolutionary model, but also to superior designs of each special component.

Key words: distributed evolutionary algorithm; memetic algorithm; quantum-inspired evolutionary algorithm; quantum distributed memetic algorithm

1 Introduction

Although quantum computation is a newly emerging technique, it has achieved widespread attention from the scientific community due to its powerful computing power and outstanding parallelized ability in tackling various specialized problems[1−3]. So far, many studies on quantum computation have progressed actively, especially in the design of quantum algorithms. In the existing quantum algorithms, one of the main design ideas is to integrate superior characteristics of quantum computation into the architecture of some specific evolutionary algorithm (EA). Such algorithms are referred to as quantum EA (QEA). Compared to traditional EAs, QEA is characterized by principles of quantum computation such as concepts of qubits and superposition of states. Based on qubit representation and evolution, QEA can imitate the quantum computation process to achieve strong competitiveness

• Guanghui Zhang is with the School of Information Science and Technology and the Hebei Key Laboratory of Agricultural Big Data, Hebei Agricultural University, Baoding 071001, China. E-mail: zhangguanghui@stu.xjtu.edu.cn.
• Wenjing Ma is with the School of Information Science and Technology, Hebei Agricultural University, Baoding 071001, China. E-mail: wjma2022@hotmail.com.
• Keyi Xing is with the State Key Laboratory for Manufacturing System Engineering and the Systems Engineering Institute, Xi'an Jiaotong University, Xi'an 710049, China. E-mail: kyxing@mail.xjtu.edu.cn.
• Lining Xing is with the School of Electronic, Xidian University, Xi'an 710071, China. E-mail: lnxing@xidian.edu.cn.
• Kesheng Wang is with the Department of Production and Quality Engineering, Norwegian University of Science and Technology, Trondheim 7491, Norway. E-mail: kesheng.wang@ntnu.no.
∗ To whom correspondence should be addressed.
Manuscript received: 2022-08-28; revised: 2022-10-02; accepted: 2022-10-13

of good population diversity, powerful global exploration, fast convergence, and easy fusion with different EAs[4].

Presently, a variety of QEAs are designed based on a certain specific EA. Narayanan and Moore[5] fused genetic algorithm and quantum knowledge for the first time. The quantum genetic algorithm was proposed and since then the community of fusing quantum computation and evolutionary computation has been opened. In addition to quantum genetic algorithm[6, 7], there are also several other QEAs to be proposed including quantum differential evolution[8−12], quantum particle swarm optimization[13], quantum immune algorithm[14], quantum cooperative coevolutionary algorithm[15], quantum estimation of distribution algorithm[16], and so on. To the best of our knowledge, these variants of QEA basically use a specific and single EA as their evolutionary framework. However, with the rapid development of information science and technology, the problems that need to be addressed become more and more complex. This has put forward new challenges to the design of QEA considering that there exist plenty of local optima in the search space of these problems[17−19].

In order to obtain more efficient evolutionary architecture for the design of QEA, a few researchers have turned their attention to distributed computation (DC)[20] and memetic computation (MC)[21]. The DC architecture adopts the divide-and-conquer idea to decompose complex problem into several sub-problems, and coevolves them by independent search optimizers. The MC architecture concentrates on synergistic coordination between global exploration and local exploitation through integrating a population-based EA and one or more local search optimizers together. They have respective advantages in the applications. Most recently, some efforts have been paid to apply DC and MC architecture to build efficient QEA for different specialized problems. For example, Deng et al.[9] proposed a distributed QEA for global optimization, where the population is divided into three subpopulations, and is evolved through different mutant strategies to ensure the independence of each subpopulation and the diversity of overall population. Deng et al.[11] presented a distributed QEA for large scale optimization, where multipopulation is evolved by a quantum differential evolution in which different mutant strategies are applied in the early and later stages of the search, respectively. In Ref. [15], a multi-strategy distributed QEA was developed and applied to the knapsack problem and airport gate allocation problem, where cooperative coevolution, random rotation direction, and Hamming adaptive rotation angle are used to refine the overall performance. For the QEAs based on MC, Yang et al.[22] built a memetic QEA with levy flight for high dimension function optimization, where a memetic framework is used to balance the global and local search and a levy flight based local search is used to enhance searchability. Tang et al.[23] proposed a memetic QEA for global optimization, where two operators of memetic evolution and quantum evolution are coordinated for better performance.

These distributed and memetic QEA variants have shown great development prospect in solving complex continuous and combinatorial optimization problems. In view of the respective architecture superiority of DC and MC, the idea of integrating quantum mechanism with the architecture superiority of both DC and MC is naturally proposed. However, the related study is still very preliminary. This paper carries out the pioneering exploration and proposes a novel quantum-inspired distributed memetic algorithm (QDMA) to tackle this issue. The main novelties and contributions are as below.

(1) A novel distributed memetic evolutionary model is presented, which includes distributed quantum evolution, intensified quantum evolution, knowledge transfer, and evolutionary restart. This model effectively fuses three superiorities of quantum mechanism, distributed evolution, and memetic evolution.

(2) Global exploration campaign is achieved by coevolving three populations in the distributed and heterogenous way, which can effectively enhance search diversity.

(3) Local exploitation campaign is performed on an external elite archive in parallel with three populations, which can effectively enhance search intensification.

(4) A novel knowledge transfer model is proposed based on a point-ring communication topology, which can exchange superior experiences among search agents effectively.

(5) QDMA is proposed by fusing quantum mechanisms into distributed memetic evolutionary model for global optimization in quantum hyperspace. The computational result show that it significantly outperforms the compared state-of-the-art algorithms.

The rest of the paper is set as follows. Section 2 introduces the related algorithms. Section 3 details the proposed QDMA. Section 4 carried out extensive computational experiments. In Section 5, the conclusions and future works are provided.

2 Related Algorithms

In the following, we introduce the algorithms closely related to the proposed QDMA, including basic difterential evolution (DE), QEA, and EAs based on DC and MC.

2.1 DE

Differential evolution (DE) is a population-based EA, which drives the evolution by the individual difference. Because of its simplicity and efficiency, it has been successfully applied in diverse fields[24]. Denote the population of DE at generation g as $X_g = \{x_{i,g}, i = 1, 2, \ldots, \mathrm{PS}\}$, where $x_{i,g}$ is represented as $[x_{i,1,g}, x_{i,2,g}, \ldots, x_{i,d,g}]$ and referred to as the target individual, which is initialized randomly in the search space.

$$x_{i,j,g} = x_{\mathrm{L}} + r \times (x_{\mathrm{U}} - x_{\mathrm{L}}) \tag{1}$$

where g is the generation number, i is the individual index, $j \in \{1, 2, \ldots, d\}$ is the dimension index, $r \in [0, 1]$ is a random number, and x_{L} and x_{U} are the predefined lower and upper bounds of the search space, respectively. The evolution of DE at generation g will be achieved by the following three operators.

Mutation. For each individual $x_{i,g}$, the mutation operator is to generate the relevant mutant individual $v_{i,g}$. Presently, different mutant strategies have been proposed in the literature. In here, three most commonly-used ones are shown as below. They will be used in our QDMA, referred to as DE/rand/1, DE/best/1, and DE/rand-to-best/1, respectively.

$$v_{i,g} = x_{r_1,g} + F \times \left(x_{r_2,g} - x_{r_3,g}\right) \tag{2}$$

$$v_{i,g} = x_{\mathrm{best},g} + F \times \left(x_{r_1,g} - x_{r_2,g}\right) \tag{3}$$

$$v_{i,g} = x_{i,g} + F \times \left(x_{\mathrm{best},g} - x_{i,g}\right) + F \times \left(x_{r_1,g} - x_{r_2,g}\right) \tag{4}$$

where i, r_1, r_2, and r_3 are four different integers in range [1, PS], F is a parameter scaling the difference vector, and $x_{\mathrm{best},g}$ is the best individual in the current generation.

Crossover. Between each pair of $x_{i,g}$ and $v_{i,g}$, the binomial crossover operator is usually performed to get a trial individual $u_{i,g}$.

$$u_{i,j,g} = \begin{cases} v_{i,j,g}, & \text{if } r \leqslant \mathrm{CR} \text{ or } j = j_r; \\ x_{i,j,g}, & \text{otherwise} \end{cases} \tag{5}$$

where CR is a parameter controlling the inheritance rate from the mutant individual, and $j_r \in \{1, 2, \ldots, d\}$ is a random integer ensuring the mutant individual is inherited at least one element.

Selection. By comparing the fitness $f(x_{i,g})$ and $f(u_{i,g})$ of $x_{i,g}$ and $u_{i,g}$ greedily, the one with better fitness will survive in the population of next generation. For a minimization problem, the selection operation is performed as below.

$$x_{i,g+1} = \begin{cases} u_{i,g}, & \text{if } f\left(u_{i,g}\right) \leqslant f\left(x_{i,g}\right); \\ x_{i,g}, & \text{otherwise} \end{cases} \tag{6}$$

2.2 QEA

QEA is also a population dynamics based EA, which drives the evolution by imitating the concept and principle of quantum computation[25]. In basic QEA, the individual is represented by the quantum bit (Q-bit). A Q-bit is represented by a pair of numbers $[\alpha, \beta]^{\mathrm{T}}$ satisfying with $|\alpha|^2 + |\beta|^2 = 1$. Each Q-bit has three possible states: "0", "1", and a linear superposition of the two, which can be determined by Eq. (7).

$$|\varphi\rangle = \alpha|0\rangle + \beta|1\rangle \tag{7}$$

where $|0\rangle$ and $|1\rangle$ denote the states "0" and "1", respectively. $|\alpha|^2$ and $|\beta|^2$ specify the probability that the Q-bit is found in the states $|0\rangle$ and $|1\rangle$, respectively. Denote the QEA population as $Q_g = \{q_{i,g}, i = 1, 2, \ldots, \mathrm{PS}\}$, where each quantum individual includes d Q-bits. Naturally, $q_{i,g}$ can be represented as a string of d pairs of numbers $[\alpha, \beta]^{\mathrm{T}}$.

$$q_{i,g} = \left[\begin{array}{c|c|c|c} \alpha_{i,1,g} & \alpha_{i,2,g} & \cdots & \alpha_{i,d,g} \\ \beta_{i,1,g} & \beta_{i,2,g} & \cdots & \beta_{i,d,g} \end{array}\right] \tag{8}$$

where $|\alpha_{i,j,g}|^2 + |\beta_{i,j,g}|^2 = 1$, $j = 1, 2, \ldots, d$. In order to evaluate each $q_{i,g}$, quantum observation is carried out to map $q_{i,g}$ to be a binary solution $p_{i,g}$. Each dimension of $p_{i,g}$ is determined by comparing $\beta_{i,j,g}$ with a random number in range [0, 1]. Here, $p_{i,j,g} = 1$ if the random number is less to $|\beta_{i,j,g}|^2$; otherwise $p_{i,j,g} = 0$.

Basic QEA is still an EA, and evolves the individuals by quantum gate (Q-gate) until a termination criterion is satisfied. The Q-gate can be seen as a variation operator, which changes the Q-bits of individuals to drive the evolution process of QEA. The rotation Q-gate is commonly used in QEA, which updates the Q-bits as follows:

$$\begin{bmatrix} \alpha_{i,j,g+1} \\ \beta_{i,j,g+1} \end{bmatrix} = \begin{bmatrix} \cos\Delta\theta_{i,j,g} & -\sin\Delta\theta_{i,j,g} \\ \sin\Delta\theta_{i,j,g} & \cos\Delta\theta_{i,j,g} \end{bmatrix} \begin{bmatrix} \alpha_{i,j,g} \\ \beta_{i,j,g} \end{bmatrix} \tag{9}$$

where $\Delta\theta_{i,j,g}$ is rotation angle which controls the direction and magnitude of rotation. They should be designed in compliance with the addressed application problems. However, the Q-gate driven QEA evolution has the issues of low search accuracy and slow convergence. More often, it is based on the quantum representation and quantum mechanism to realize the evolution within the framework of some basic EAs such as DE[2], genetic algorithm[6], and particle swarm optimization[13]. In this sense, the performance of a QEA is greatly influenced by the adopted evolutionary architecture. For this reason, researchers began to try to adopt some new evolutionary models such as DC and MC when designing QEA, and have achieved several high-quality algorithms.

2.3 EAs based on DC and MC

For ease of description, denote the EAs based on DC and MC as distributed EA (DEA) and memetic EA (MEA), respectively. Next, we make a brief review on them.

DEA is a famous computational method, which decomposes complex problem into multiple subproblems and evolves them in the distributed way[20]. To realize the distributed evolution efficiently, different distributed evolutionary models have been proposed. Zhan et al.[18] proposed an adaptive DE-based DEA within a master-slave multipopulation distributed model, where a master node dominates three different-identity slave nodes, and different populations are co-evolved concurrently on these nodes. According to this master-slave distributed model, some DEA variants[26−30] have been built based on different EAs such as DE, genetic algorithm, and particle swarm optimization. Ge et al.[31] proposed a DEA within a balanced best-random distributed model for database fragmentation problem, where the population is divided into several subpopulations which are arranged in a ring topology. Communications occur when the best individual and a random individual of each subpopulation are sent to its neighbors according to the communication topology in the opposite directions. Ge et al.[32] was also based on this model to propose a DEA with adaptive split and mergence for large-scale optimization. More distributed evolutionary models and the corresponding DEAs can be found in Ref. [20]. Although plenty of efforts have been paid to DEAs, the reports on the fusion algorithms of quantum mechanism and DEA are still very limited. At present, there are only Refs. [8, 9, 11, 15, 22, 23].

MEA can be viewed as a computational method, which fuses a population-based EA and one or more problem-specific local optimizers to effectively balance exploration and exploitation. Due to successful applications in various specialized problems, MEA has attracted wide attention[21, 33]. Here, we only focus on the works in the fields of distributed MEA and quantum MEA. Zhang et al.[34] built a distributed memetic DE which fuses Lamarckian learning and Baldwinian learning. To gain a better tradeoff between exploration and exploitation, DE as an evolutionary framework is assisted by the Hooke-Jeeves local search algorithm. Zhang et al.[35] presented a multipopulation ant colony MEA with problem-specific local search for supply chain configuration problem. Sabar et al.[36] put forward a heterogeneous memetic DE for big data optimization problem, which fuses a cooperative co-evolution method with various memetic algorithms to increase the efficiency of the solving process. Zhang et al.[37] devised a distributed co-evolutionary MEA to address a realistic production scheduling problem, in which four newly devised modules are integrated reasonably. In addition, the works in the aspects of quantum MEA are very scarce, and we only retrieved Refs. [22, 23].

Overall, DEA and MEA represent two different evolutionary models, which have shown their respective superiority in the wide applications. Encouraged by these facts, this paper aims to integrate quantum evolution with the architecture superiority of both DEA and MEA to propose a novel and efficient QDMA metaheuristic.

3 Proposed QDMA

In this section, we elaborate the proposed QDMA including its evolutionary framework, specific designs, overall procedure, and the complexity analysis.

3.1 Distributed memetic evolutionary model

The QDMA adopts a novel distributed memetic evolutionary model to achieve the optimization process. All evolutions are performed completely in a quantum space. Figure 1 illustrates the model consisting of four modules, that is, distributed quantum evolution on multipopulation, intensified quantum evolution on elite archive, knowledge transfer with point-ring topology, and evolutionary restart with adaptive perturbation.

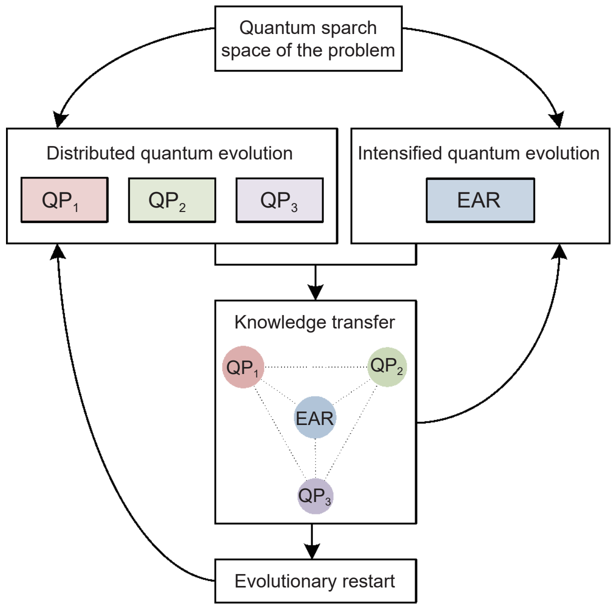

Fig. 1 Distributed memetic evolutionary model.

In the evolutionary process, three quantum populations (QP_1, QP_2, and QP_3) are coevolved for global exploration in a distributed way. To obtain diversified exploring behaviors, heterogeneous optimizers are performed on three populations. In parallel with the quantum populations, an elite archive (EAR) consisting of some superior individuals is configurated. It is evolved by the local optimizer to exploit the promising area found by global exploration. This is not only different from traditional memetic algorithm (MA)[38] that evolves the global and local search agents in serial, but also different from traditional DEA[20] that evolves multiple global search agents parallelly. In this sense, the proposed model can cover both advantages of MA and DEA. After the global and local search campaign, the superior experiences from different search agents are exchanged by transferring the knowledge in a point-ring topology. Finally, the three populations are restarted by the evolutionary operators like quantum mutation, crossover, and selection. In this way, the homogeneous searches are expected to be relieved, especially in the later evolutionary phase. In this evolutionary model, the above process will be repeated until a termination criterion is met.

3.2 Quantum representation and measurement

In the population $X_g = \{x_{i,g}, i = 1,2,\ldots,\mathrm{PS}\}$ of QDMA, the individuals also employ the quantum representation as in the basic QEA. That is, $x_{i,g} = \left[x_{i,1,g}\left|x_{i,2,g}\right|\ldots\left|x_{i,D,g}\right.\right]$, where $x_{i,j,g}$ is a Q-bit. Because parameters α and β satisfy with $|\alpha|^2 + |\beta|^2 = 1$, the Q-bit can be expressed as $[\cos\ \theta,\ \sin\ \theta]^{\mathrm{T}}$ equivalently. Thus, the $x_{i,g}$ is re-expressed as below.

$$x_{i,g} = \left[\begin{array}{c|c|c|c} \cos\theta_{i,1,g} & \cos\theta_{i,2,g} & \cdots & \cos\theta_{i,D,g} \\ \sin\theta_{i,1,g} & \sin\theta_{i,2,g} & \cdots & \sin\theta_{i,D,g} \end{array}\right] \tag{10}$$

$$\theta_{i,j,g} = \mathrm{rand}_j \times \pi \tag{11}$$

where rand_j is a random number in range [0, 1], which means $\theta_{i,j,g}$ is always generated in range $[0,\ \pi]$ randomly.

Under this representation, the search space is limited into the range [−1, 1]. To calculate the fitness, the quantum individuals must be mapped to the solution space in range $[a,\ b]$, where a and b are the lower bound and upper bound of the variables of test functions in the experiment, respectively. For each Q-bit of $x_{i,g}$, the solution space mapping is performed by Eq. (12). Now, $x_{i,g}$ can be seen as the superposition of 2^d possible solutions. To further collapse $x_{i,g}$ into a unique solution vector, a quantum measurement strategy in Eq. (13) is carried out. Then, the fitness of the current $x_{i,g}$ can be calculated.

$$x_{i,j,g} = \frac{(b-a)\times x_{i,j,g} + (b+a)}{2} = \begin{bmatrix} \left((b-a)\times\cos\theta_{i,j,g} + (b+a)\right)/2 \\ \left((b-a)\times\sin\theta_{i,j,g} + (b+a)\right)/2 \end{bmatrix} \tag{12}$$

$$x_{i,j,g} = \begin{cases} \cos\theta_{i,j,g}, & \text{if } \cos^2\theta_{i,j,g} < \mathrm{rand}; \\ \sin\theta_{i,j,g}, & \text{otherwise} \end{cases} \tag{13}$$

Based on the quantum representation, three initial quantum populations QP_1, QP_2, and QP_3 are generated randomly, and then coevolved in distributed way.

3.3 Distributed quantum evolution on multipopulation

The global exploration campaign of QDMA is achieved by coevolving QP_1, QP_2, and QP_3 at each generation. Figure 2 shows the global exploration procedure. We can see that three subpopulations are coevolved by different quantum operators in distributed way. In this way, it is expected to increase the successful probability of the global exploration traversing the whole solution space. In view of the quantum characteristics of each population and in order to evolve them, we introduce the quantum computing into three operators (mutation, crossover, and selection) of the basic DE algorithm as follows.

Quantum mutation. For each quantum individual $x_{i,g}$ of the population $X_g = \{x_{i,g}, i = 1,2,\ldots,\mathrm{PS}\}$, there is a

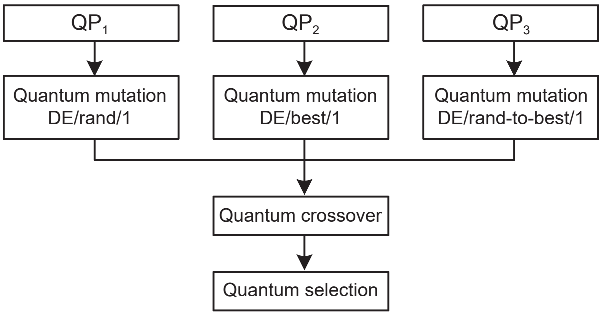

Fig. 2 Distributed quantum evolution on multipopulation.

unique phase angle vector $\theta_{i,g}=\left[\theta_{i,1,g}\left|\theta_{i,2,g}\right|\ldots\left|\theta_{i,D,g}\right.\right]$. For convenience, denote the set of all phase angle vectors as $\theta_g=\left\{\theta_{i,g}, i=1, 2,\ldots,\text{PS}\right\}$. Based on the framework of three basic mutation strategies DE/rand/1, DE/best/1, and DE/rand-to-best/1, three quantum mutation operators are used to update $\theta_{i,g}$ to generate the mutant vector.

$$\theta_{im,g}=\theta_{r_1,g}+F\times\left(\theta_{r_2,g}-\theta_{r_3,g}\right) \tag{14}$$

$$\theta_{im,g}=\theta_{\text{best},g}+F\times\left(\theta_{r_1,g}-\theta_{r_2,g}\right) \tag{15}$$

$$\theta_{im,g}=\theta_{i,g}+F\times\left(\theta_{\text{best},g}-\theta_{i,g}\right)+F\times\left(\theta_{r_1,g}-\theta_{r_2,g}\right) \tag{16}$$

where $\theta_{i,g}$ corresponds to $x_{i,g}$. In evolution, they are performed on three quantum populations QP_1, QP_2, and QP_3, respectively, which are used to enrich the search diversity.

Quantum crossover. To further diversify the population, the mutant vector and its parent vector are performed by the quantum crossover operator to generate a trial vector.

$$\theta_{ic,j,g}=\begin{cases}\theta_{im,j,g}, & \text{if } r\leqslant \text{CR or } j=j_r;\\ \theta_{i,j,g}, & \text{otherwise}\end{cases} \tag{17}$$

Quantum selection. To enable better individuals to enter the next generation of population, the fitness value of the quantum individual $x_{i,g}$ corresponding to the obtained trial vector $\theta_{ic,g}$ is calculated. Then, a greedy selection operator is performed. In this way, the quantum individual is evolved by the update of quantum phase angles.

$$\theta_{i,g+1}=\begin{cases}\theta_{ic,g}, & \text{if } f\left(x_{ic,g}\right)\leqslant f\left(x_{i,g}\right);\\ \theta_{i,g}, & \text{otherwise}\end{cases} \tag{18}$$

3.4 Intensified quantum evolution on elite archive

The local exploitation campaign is set in QDMA to enhance the search intensification for several promising areas found by global exploration. Different from the serial execution mode in traditional hybrid algorithms, however, QDMA performs the local exploitation in parallel with the proposed multipopulation global exploration. For details, the local exploitation campaign is performed on an external elite archive (EAR), which is independent of the multipopulation. In this way, the search intensification and diversity can be balanced more effectively. For the elite archive, its individual quality is expected to be as good as possible. Therefore, the elite archive is initialized by the three best individuals from the initial quantum populations QP_1, QP_2, and QP_3. When performing the local exploitation, two operators are adopted. One is simplex search operator, which is used to improve the worst individual of EAR, and the other is Cauchy mutation operator, which is used to perturb the other two EAR individuals.

Simplex search. It is a direct optimization technique, which does not depend on specific gradient information and has strong local search ability[39]. It first constructs a simplex based on current population, and rescales it to generate a better solution based on four operators: reflection, expansion, contraction, and shrinkage. Then, the worst solution in population is replaced by this new solution. Thus, the population quality can be gradually improved to approximate the global optimum. Denote the worst individual of EAR as $\theta_{w,g}$, the other two as $\theta_{a,g}$ and $\theta_{b,g}$, and the optimal of EAR as $\theta_{o,g}$. The steps of the proposed simplex search operator are as follows.

Step 1: Reflection. Calculate the middle point $\theta_{\text{mid},g}=(\theta_{a,g}+\theta_{b,g})\ /\ 2$, and generate a new point $\theta_{\text{ref},g}$ by reflecting $\theta_{w,g}$ based on Eq. (19). For a minimization problem, if $f(\theta_{\text{ref},g})<f(\theta_{o,g})$, perform the expansion; if $f(\theta_{\text{ref},g})>f(\theta_{w,g})$, perform the contraction; otherwise, perform the shrinkage.

$$\theta_{\text{ref},g}=\theta_{\text{mid},g}+\delta_1\times\left(\theta_{\text{mid},g}-\theta_{w,g}\right) \tag{19}$$

Step 2: Expansion. Expand $\theta_{\text{ref},g}$ to point $\theta_{\exp,g}$ along the same reflection direction by Eq. (20). If $f(\theta_{\exp,g})<f(\theta_{o,g})$, the better one of $\theta_{\exp,g}$ and $\theta_{\text{ref},g}$ replaces $\theta_{w,g}$; otherwise, $\theta_{\text{ref},g}$ replaces $\theta_{w,g}$.

$$\theta_{\exp,g}=\theta_{\text{mid},g}+\delta_2\times\left(\theta_{\text{ref},g}-\theta_{\text{mid},g}\right) \tag{20}$$

Step 3: Contraction. This operator is performed by Eq. (21) to generate the contraction point $\theta_{\text{con},g}$. If $f(\theta_{\text{con},g})<f(\theta_{w,g})$, $\theta_{\text{con},g}$ replaces $\theta_{w,g}$.

$$\theta_{\text{con},g}=\theta_{\text{mid},g}+\delta_3\times\left(\theta_{w,g}-\theta_{\text{mid},g}\right) \tag{21}$$

Step 4: Shrinkage. This operator is performed by Eq. (22) to generate the shrinkage point $\theta_{\text{shr},g}$. If $f(\theta_{\text{shr},g})<f(\theta_{w,g})$, the better one of $\theta_{\text{shr},g}$ and $\theta_{\text{ref},g}$ replaces $\theta_{w,g}$;

otherwise, $\theta_{\text{ref},g}$ replaces $\theta_{w,g}$.

$$\theta_{\text{shr},g}=\theta_{\text{mid},g}-\delta_4\times\left(\theta_{w,g}-\theta_{\text{mid},g}\right) \tag{22}$$

where δ_1 = 1, δ_2 = 2, and δ_3 = δ_4 = 0.5. These four operators are shown in Fig. 3.

Cauchy mutation. Due to the fast convergence brought by the simplex search strategy and the following knowledge transfer strategy, the improvement of EAR will become more and more difficult with evolution. It is expected to introduce an efficient strategy for EAR such that the search can jump out of the local optimal solution. For this purpose, the Cauchy mutation, which has successfully applied to improve many EAs[40−42], is used to perturb the $\theta_{a,g}$ and $\theta_{b,g}$ of EAR. Take $\theta_{a,g}$ as an example, the Cauchy mutation is performed based on Eq. (23), in which the generating function of Cauchy distribution random variable is in Eq. (24).

$$\theta_{a,g}=\theta_{a,g}+\theta_{\text{Gbest},g}\times\text{Cauchy}(0,1) \tag{23}$$

$$\text{Cauchy}(0,1)=\tan\left[(\xi-0.5)\pi\right] \tag{24}$$

where $\xi\in[0,1]$. $\theta_{\text{Gbest},g}$ is the global optimal solution found at generation g and the greedy acceptance criterion is used after each mutation.

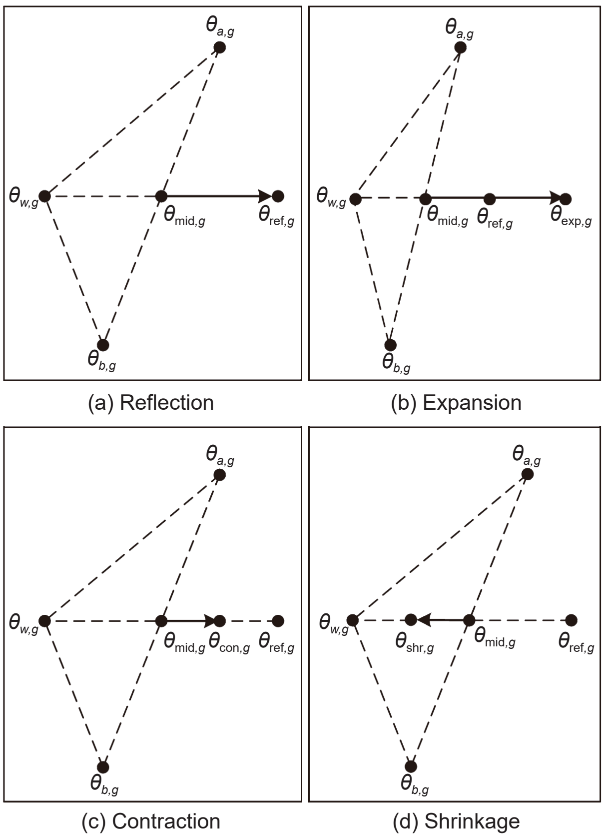

Fig. 3 Simplex operator.

3.5 Knowledge transfer based on point-ring topology

During the QDMA evolutionary process, the three quantum populations and the elite archive are evolved in the distributed and independent way. Because of their different evolutionary mechanisms, the superior information they have searched for at each generation is different and time-varying. Clearly, it is very necessary to exchange these excellent experiences among them to improve the overall searching ability of the algorithm. This is achieved by a knowledge transfer strategy, by which the search individuals are transferred across search agents reasonably and dynamically. Before devising this knowledge transfer model, we expect it to be a general model, that is, it can tackle different optimization problems directly. To this end, a new knowledge transfer model independent of specific problems is proposed based on the point-ring topology. Figure 4 shows the procedure of knowledge transfer.

Specifically, all search agents are configurated according to a point-ring topology, where the three quantum populations QP_1, QP_2, and QP_3 are arranged to the ring positions and the EAR is placed to the central point of the ring. First, the best individual of each population is migrated into the neighbor population clockwise. Therefore, the population diversity can be remained. Second, individual replacement operation is performed from EAR to each

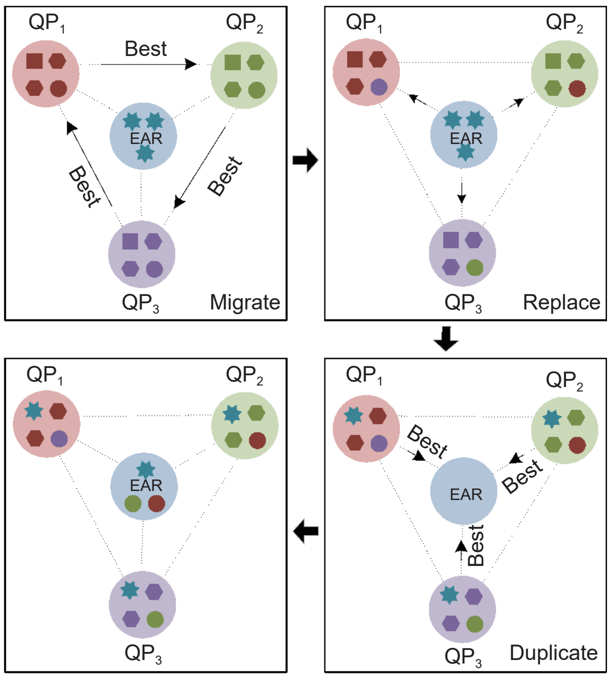

Fig. 4 Example of knowledge transfer.

population. In replacement, three elite individuals of EAR are removed to replace the worst individual of each population, respectively. Therefore, the overall quality of three populations can be continuously improved. Third, individual replication operation is executed from each population to EAR. The best individual of each current population is replicated and the copy is sent back to EAR. This can always maintain the overall quality of EAR at a high level, and can also introduce a certain diversity into EAR.

3.6 Evolutionary restart with adaptive perturbation

The strategies of quantum evolution and knowledge transfer can diversify search behaviors effectively. However, it is hoped to further enrich it especially in the late phase of evolution. So, an adaptive strategy is proposed to restart the global exploration populations (QP_1, QP_2, and QP_3). This restart strategy uses the framework of the quantum DE algorithm proposed previously. The mutation operator is executed by Eq. (25) and the crossover operator is the same as that in Eq. (17). The designed selection operator is described in Eq. (26). As can be seen, the trial individual after the mutation and crossover will enter the next generation if it is better than the parent individual. Otherwise, the parent individual is first perturbed in an adaptive way by Eq. (27), and then sent into next generation. From Eq. (27), we can see that the strength of the perturbation gradually increases with the evolution, which will provide more and more diversity for the populations, so that the search can escape the local optima and traverse more solution space.

$$\theta_{i,j,g} = \theta_{r_1,j,g} + F \times \mathrm{rand}_j \times \left(\theta_{r_2,j,g} - \theta_{r_3,j,g}\right) \tag{25}$$

$$\theta_{i,j,g+1} = \begin{cases} \theta_{ic,j,g}, & \text{if } f\left(x_{ic,g}\right) \leqslant f\left(x_{i,g}\right); \\ \theta_{i,j,g} + \Delta\theta_{i,j,g}, & \text{otherwise} \end{cases} \tag{26}$$

$$\Delta\theta_{i,j,g} = \theta_{\min} + \mathrm{fit} \times \mathrm{rand}_j \times (\theta_{\max} - \theta_{\min}) \times \exp\left(\frac{\mathrm{cputime}}{\mathrm{max\_cputime}}\right) \tag{27}$$

$$\mathrm{fit} = \frac{f\left(\theta_{i,g}\right) - f\left(\theta_{\mathrm{Gbest},g}\right)}{f\left(\theta_{\mathrm{Gbest},g}\right)} \tag{28}$$

where $\theta_{\min} = 0.001 \times \pi$, $\theta_{\max} = 0.05 \times \pi$, cputime is the current runtime, and max_cputime is the allowed maximum runtime of the algorithms.

3.7 Overall procedure of QDMA

Combining the above special designs, the overall procedure of QDMA is illustrated in Algorithm 1. Note that the QDMA has two key parameters F and CR that

Algorithm 1 Overall procedure of QDMA

Input: population size PS, parameter pools of F and CR
Output: Global best solution gbest
/* Initialization */
Initialize quantum populations QP_1, QP_2, and QP_3 randomly
Solution space transformation in Eq. (12)
Quantum individual measurement in Eq. (13)
Calculate the fitness of quantum individuals
Choose the best of each population to generate elite archive EAR
Determine gbest
while the termination criterion is not satisfied **do**
  /* Distributed quantum evolution on multipopulation */
  for $k = 1$ to 3 **do**
    for $i = 1$ to PS **do**
      Randomly select a value of F from parameter pool
      Randomly select a value of CR from parameter pool
      Quantum mutation on individual i of QP_k, where DE/rand/1 for QP_1, DE/best/1 for QP_2, and DE/rand-to-best/1 for QP_3
      Quantum crossover in Eq. (17) on individual i of QP_k
      Quantum selection in Eq. (18) on individual i of QP_k
    endfor
  endfor
  /* Intensified quantum evolution on EAR */
  Simplex search on the worst individual of EAR
  Cauchy mutation on the other individuals of EAR
  /* Knowledge transfer with point-ring topology */
  Configure QP_1, QP_2, QP_3, and EAR in a point-ring topology
  Migrate the best individual of each population clockwise
  Replace the worst individual of each population by the elite of EAR
  Duplicate the best individual of each population to EAR
  /* Evolutionary restart with adaptive perturbation */
  for $k = 1$ to 3 **do**
    for $i = 1$ to PS **do**
      Randomly select a value of F from parameter pool
      Randomly select a value of CR from parameter pool
      Quantum mutation in Eq. (25) on individual i of QP_k
      Quantum crossover in Eq. (17) on individual i of QP_k
      Evolutionary restart in Eqs. (26) and (27) on individual i of QP_k
    endfor
  endfor
  Update gbest
endwhile
return gbest

need to be controlled. In evolution, we first adopt the experimental method to initialize a pool of good combinations (F, CR). Then (F, CR) are randomly selected from the pools at each usage. The detailed procedure is introduced in the experimental setup.

3.8 Analysis of QDMA on complexity

Denote the population size and the problem dimension as PS and D, respectively. First, in the initialization of quantum populations and elite archive, the time complexity is obtained as $O(\text{PS}\times D)$ and $O(\text{PS})$, respectively. In order to measure the population individuals, the solution space transformation and quantum observation are performed. The time complexity is calculated as $O(\text{PS}\times D)$. Second, the quantum populations are coevolved in distributed way by the three operators of quantum DE algorithm. The time complexity of the mutation operator, crossover operator, and selection operator is obtained as $O(\text{PS})$, $O(\text{PS}\times D)$, and $O(\text{PS})$, respectively. Third, the simplex search and Cauchy mutation strategies are applied to improve the elite archive, and the time complexity is obtained as $O(1)$. Fourth, the knowledge transfer among different search agents is used to exchange the superior search experience. The time complexity is also $O(1)$. Finally, in the restart strategy based on quantum DE, the quantum mutation, crossover, and selection operators are used to evolve or adaptively perturb the three populations. The time complexity is calculated as $O(\text{PS})$, $O(\text{PS}\times D)$, and $O(\text{PS})$, respectively. Therefore, the overall time complexity of QDMA is obtained as $O(\text{PS}\times D)$.

4 Computational Results

4.1 Experimental setup

To evaluate the performance of the proposed algorithms, a total of 18 representative and universal benchmark functions from CEC 2014−2018 are selected. All these functions have the global optimum value zero. They are considered four different dimensions, denoted as 10D, 30D, 50D, and 100D, respectively. Further, f_1–f_8 are unimodal functions, f_9 is a noisy quadratic function, and f_{10}–f_{18} are multimodal functions. In order to save space, the details of all these functions including expressions, variable ranges, and optimal values are provided in Table A1 in the Appendix.

The proposed QDMA applies three quantum populations for distributed evolution. The size of each population is set as PS = 30. As the original DE algorithm, the range of parameters F and CR are all limited into [0, 1]. To determine the most reasonable combination of them, we first select F and CR from the set {0.1, 0.2, …, 1.0}, which generates a total of 100 combinations (F, CR). Then, each combination is performed 30 times by QDMA on the 30-dimension function f_{14}. Finally, we adopt the average value of the optimal results in 30 times as the measure criterion of each combination. The top 12 combinations with the lowest measure criterion are selected. In this way, we obtain a good pool of combination (F, CR), where $F \in$ {0.6, 0.7, 0.8, 0.9} and CR $\in$ {0.7, 0.8, 0.9}. In the evolution of QDMA, (F, CR) are randomly selected from the pools at each usage.

We compare the QDMA against three state-of-the-art QEA variants, including MSIQEA[8], EMMSIQDE[9], and HMCFQDE[11]. In addition, the QDMA is also compared against three effective DE variants and a novel distributed DE variant, including SaDE[43], SaNSDE[44], SaNSDE+[45], and ADDE[18]. In order to execute a fair comparison, the parameters of these compared algorithms are set the same as those in the original papers. At the same time, the termination criterion is set as the specified maximal runtime, i.e., 10 000×D ms, in all algorithms.

The performance of all algorithms is measured by the fitness error value $f(x) - f(x^*)$, where x is the best solution obtained by a given algorithm in a single run and x^* is the global optimum of the test function. To generate reliable statistical results, each algorithm is run 30 times independently on each benchmark function. Among the 30 runs, the best error value (Bev), mean error value (Mev), and standard deviation (Std) are used for the final performance metrics. Besides, the Wilcoxon's rank-sum test[46] at 5% significance level is carried out to evaluate the statistical significance of the algorithmic comparison. Three symbols ("+", "≈", and "–") are used to represent that the QDMA is significantly better, similarly, or worse than the compared algorithm. For clarity, the results of the best algorithms are marked in boldface. The implementation setup is the Python3.7 simulation software and the PC with an Intel(R) Core(TM) i5-9300H CPU@2.4 GHz and 8 GB RAM under the 64-bit Windows 10 operating system.

4.2 Comparison with different QEA algorithms

The QDMA is first compared with the three state-of-the-art QEA variants including MSIQDE[8], EMMSIQDE[9], and HMCFQDE[11] to assess its overall performance. The detailed computational results are provided from Table 1 to Table 4 on four different dimensions, respectively.

For the $10D$ problems in Table 1, the Bev index shows that QDMA generates the best performance on all the functions, and the Mev and Std indexes show that QDMA outperforms the competitors on 10 benchmark functions except on f_4, f_9–f_{12}, f_{14}, f_{16}, and f_{17}. On these 8 functions, it is surpassed by MSIQDE on f_4, f_{11}, f_{12}, and f_{16} and EMMSIQDE on f_9, f_{10}, f_{14}, and f_{17}.

For the $30D$ and $50D$ problems in Tables 2 and 3, QDMA provides the almost same superiority on all these functions. Specifically, QDMA also generates all best values for the Bev index, and performs better in 12 values of the $30D$ problems and in 13 values of the $50D$ problems for both Mev and Std indexes.

For the larger $100D$ problems in Table 4, we can see that the superiority of QDMA becomes more and more obvious. It significantly outperforms the other competitors on all functions, except the Mev and Std indexes of functions f_2, f_4, and f_{10}. In addition, we can also see that QDMA can generate the global optimum zero in 12, 8, 8, and 5 of the 18 functions for the $10D$, $30D$, $50D$, and $100D$ problems, respectively. It is significantly better than the compared three algorithms.

As can be seen from the statistic test, QDMA is significantly better (+) than MSIQDE on 12, 16, 16, and 16 of 18 functions in the dimensions from $10D$ to $100D$, respectively. These data are 11, 14, 14, and 15 of 18 functions in the dimensions from $10D$ to $100D$ for EMMSIQDE. HMCFQDE cannot outperform QDMA on any function in the four dimensions. Thus, it can be concluded that QDMA achieves the best performance, followed by MSIQDE, EMMSIQDE, and HMCFQDE. According to the longitudinal comparison from $10D$ to $100D$, the results show that the performance of QDMA becomes better and better with the increase of problem scale, which also indicates that it has greater advantages in solving large-scale problems. As a result, QDMA generates the best performance.

In order to investigate the algorithmic evolutionary behavior, we further provide their convergence graphs to illustrate their evolutionary process. We choose f_5, f_9, f_{12}–f_{14}, and f_{16} as the representative functions and only test $50D$ problems. For other $10D$, $30D$, and $100D$ dimensions, our experiment shows that the convergence curves of the algorithms have the same change trend as those of $50D$. Therefore, they will not be provided here due to page limitation. From the convergence graphs in Fig. 5, we can see that the proposed QDMA shows three aspects of advantages. First, QDMA obtains the better solutions than its

Table 1 Comparison of QDMA and state-of-the-art QEAs on $10D$ problems.

Function	QDMA	MSIQDE	EMMSIQDE	HMCFQDE
	Bev/Mev/Std	Bev/Mev/Std/Test	Bev/Mev/Std/Test	Bev/Mev/Std/Test
f_1	$\mathbf{0.00\times10^{0}/1.80\times10^{-26}/3.55\times10^{-26}}$	$3.41\times10^{-10}/6.68\times10^{-6}/1.31\times10^{-5}/+$	$2.25\times10^{-6}/1.65\times10^{-3}/4.18\times10^{-3}/+$	$4.88\times10^{0}/1.41\times10^{2}/1.94\times10^{2}/+$
f_2	$\mathbf{0.00\times10^{0}/1.58\times10^{-24}/3.75\times10^{-24}}$	$3.01\times10^{-8}/1.91\times10^{-4}/3.41\times10^{-4}/+$	$1.39\times10^{-5}/2.97\times10^{-2}/5.19\times10^{-2}+$	$2.30\times10^{2}/7.48\times10^{2}/5.74\times10^{2}/+$
f_3	$\mathbf{0.00\times10^{0}/3.56\times10^{-20}/1.59\times10^{-19}}$	$3.92\times10^{-7}/2.49\times10^{-4}/2.84\times10^{-4}/+$	$1.02\times10^{-1}/1.94\times10^{2}/4.11\times10^{2}/+$	$1.39\times10^{3}/4.18\times10^{3}/2.03\times10^{3}/+$
f_4	$\mathbf{0.00\times10^{0}}/1.76\times10^{-2}/7.34\times10^{-2}$	$1.84\times10^{-5}/\mathbf{9.00\times10^{-4}/8.49\times10^{-4}/+}$	$1.13\times10^{-5}/9.39\times10^{-3}/9.81\times10^{-3}/+$	$7.69\times10^{-1}/1.62\times10^{0}/4.66\times10^{-1}/+$
f_5	$\mathbf{0.00\times10^{0}/1.73\times10^{-13}/2.86\times10^{-13}}$	$8.38\times10^{-5}/1.83\times10^{-3}/1.31\times10^{-3}/+$	$1.77\times10^{-4}/9.77\times10^{-3}/9.92\times10^{-3}+$	$8.64\times10^{0}/1.28\times10^{1}/2.48\times10^{0}/+$
f_6	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}/\approx}$	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}/\approx}$	$7.50\times10^{1}/3.86\times10^{2}/2.53\times10^{2}/+$
f_7	$\mathbf{0.00\times10^{0}/2.15\times10^{-25}/3.22\times10^{-25}}$	$7.24\times10^{-9}/1.30\times10^{-5}/2.15\times10^{-5}/+$	$5.38\times10^{-7}/6.99\times10^{-3}/1.26\times10^{-2}/+$	$7.43\times10^{0}/4.49\times10^{2}/2.61\times10^{2}/+$
f_8	$\mathbf{0.00\times10^{0}/4.90\times10^{-20}/8.67\times10^{-20}}$	$1.91\times10^{-3}/3.13\times10^{0}/5.44\times10^{0}/+$	$4.72\times10^{-2}/5.51\times10^{2}/8.65\times10^{2}/+$	$6.97\times10^{6}/1.08\times10^{8}/7.35\times10^{7}/+$
f_9	$\mathbf{9.48\times10^{-6}}/9.87\times10^{-3}/1.66\times10^{-2}$	$3.54\times10^{-4}/1.94\times10^{-3}/1.44\times10^{-3}/\approx$	$4.02\times10^{-5}/\mathbf{1.31\times10^{-3}/9.11\times10^{-4}}/\approx$	$1.52\times10^{-2}/7.52\times10^{-2}/3.19\times10^{-2}/+$
f_{10}	$\mathbf{1.59\times10^{-28}}/3.99\times10^{-1}/1.20\times10^{0}$	$5.44\times10^{0}/8.82\times10^{0}/6.32\times10^{-1}/+$	$2.80\times10^{-4}/\mathbf{1.44\times10^{-1}/2.34\times10^{-1}/+}$	$2.15\times10^{4}/2.87\times10^{5}/4.26\times10^{5}/+$
f_{11}	$\mathbf{0.00\times10^{0}}/6.40\times10^{-2}/1.30\times10^{-1}$	$4.39\times10^{-9}/\mathbf{3.27\times10^{-4}/5.38\times10^{-4}}/\approx$	$1.74\times10^{-6}/7.34\times10^{-3}/1.23\times10^{-2}/\approx$	$2.12\times10^{0}/4.37\times10^{0}/1.59\times10^{0}/+$
f_{12}	$\mathbf{4.44\times10^{-16}}/1.44\times10^{-1}/4.51\times10^{-1}$	$6.36\times10^{-5}/\mathbf{1.60\times10^{-3}/1.65\times10^{-3}/+}$	$2.12\times10^{-4}/1.08\times10^{-2}/1.29\times10^{-2}/+$	$4.95\times10^{0}/7.93\times10^{0}/1.53\times10^{0}/+$
f_{13}	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$5.97\times10^{-10}/1.28\times10^{-6}/2.81\times10^{-6}/+$	$1.57\times10^{-9}/4.86\times10^{-6}/1.09\times10^{-5}/+$	$1.25\times10^{-2}/1.02\times10^{0}/6.15\times10^{-1}/+$
f_{14}	$\mathbf{0.00\times10^{0}}/5.42\times10^{-1}/8.19\times10^{-1}$	$3.33\times10^{-2}/1.80\times10^{-1}/9.04\times10^{-2}/\approx$	$5.84\times10^{-3}/\mathbf{1.49\times10^{-1}/8.52\times10^{-2}}/\approx$	$3.11\times10^{0}/4.40\times10^{0}/6.10\times10^{-1}/+$
f_{15}	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$3.71\times10^{-11}/1.90\times10^{-8}/2.38\times10^{-8}/+$	$1.17\times10^{-13}/4.03\times10^{-8}/6.73\times10^{-8}/+$	$5.71\times10^{-3}/1.80\times10^{-2}/9.09\times10^{-3}/+$
f_{16}	$\mathbf{4.25\times10^{-97}}/1.67\times10^{-1}/2.57\times10^{-1}$	$2.13\times10^{-5}/\mathbf{4.71\times10^{-4}/5.34\times10^{-4}}/\approx$	$3.65\times10^{-5}/9.21\times10^{-3}/9.91\times10^{-3}/\approx$	$1.61\times10^{0}/2.62\times10^{0}/5.60\times10^{-1}/+$
f_{17}	$\mathbf{1.01\times10^{-31}}/6.02\times10^{-1}/2.16\times10^{0}$	$5.86\times10^{-5}/3.83\times10^{-2}/1.65\times10^{-2}/\approx$	$7.01\times10^{-8}/\mathbf{8.75\times10^{-5}/1.37\times10^{-4}}/\approx$	$4.47\times10^{0}/1.09\times10^{1}/4.78\times10^{0}/+$
f_{18}	$\mathbf{3.35\times10^{-30}/1.30\times10^{-3}/4.42\times10^{-3}}$	$4.10\times10^{-2}/1.61\times10^{-1}/6.61\times10^{-2}/+$	$2.92\times10^{-7}/7.53\times10^{-3}/2.12\times10^{-2}/\approx$	$3.49\times10^{0}/2.30\times10^{2}/8.35\times10^{2}/+$
Statistic test (+/≈/−)	−	12/6/0	11/7/0	18/0/0

Table 2 Comparison of QDMA and state-of-the-art QEAs on 30*D* problems.

Function	QDMA	MSIQDE	EMMSIQDE	HMCFQDE
	Bev/Mev/Std	Bev/Mev/Std/Test	Bev/Mev/Std/Test	Bev/Mev/Std/Test
f_1	$\mathbf{0.00\times10^{0}/1.95\times10^{-22}/6.02\times10^{-22}}$	$3.04\times10^{-10}/1.43\times10^{-6}/1.76\times10^{-6}/+$	$3.32\times10^{-7}/2.53\times10^{-3}/3.02\times10^{-3}/+$	$8.39\times10^{3}/1.69\times10^{4}/2.90\times10^{3}/+$
f_2	$\mathbf{2.58\times10^{-88}}/2.02\times10^{0}/9.26\times10^{0}$	$1.56\times10^{-7}/\mathbf{1.12\times10^{-3}/2.87\times10^{-3}}/+$	$1.86\times10^{-5}/5.65\times10^{-1}/8.84\times10^{-1}/+$	$2.45\times10^{4}/3.34\times10^{4}/5.31\times10^{3}/+$
f_3	$\mathbf{0.00\times10^{0}}/1.10\times10^{4}/3.88\times10^{4}$	$7.05\times10^{-7}/\mathbf{9.42\times10^{-4}/1.77\times10^{-3}}/+$	$8.74\times10^{-2}/5.93\times10^{2}/1.10\times10^{3}+$	$1.28\times10^{7}/2.31\times10^{7}/6.16\times10^{6}/+$
f_4	$\mathbf{2.41\times10^{-59}}/9.04\times10^{0}/1.52\times10^{1}$	$2.24\times10^{-7}/\mathbf{1.03\times10^{-3}/1.18\times10^{-3}}/–$	$1.70\times10^{-4}/3.68\times10^{-2}/3.69\times10^{-2}/–$	$1.88\times10^{1}/3.65\times10^{1}/7.79\times10^{0}/+$
f_5	$\mathbf{4.95\times10^{-44}}/2.00\times10^{0}/1.08\times10^{1}$	$1.49\times10^{-5}/\mathbf{2.55\times10^{-3}/2.52\times10^{-3}}/+$	$2.02\times10^{-5}/1.04\times10^{-2}/9.08\times10^{-3}/+$	$5.34\times10^{1}/6.02\times10^{1}/2.80\times10^{0}/+$
f_6	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}/\approx$	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}/\approx$	$1.06\times10^{4}/1.61\times10^{4}/2.17\times10^{3}/+$
f_7	$\mathbf{0.00\times10^{0}/4.39\times10^{-85}/3.08\times10^{-85}}$	$7.26\times10^{-9}/6.74\times10^{-6}/9.32\times10^{-6}/+$	$4.43\times10^{-4}/1.07\times10^{-1}/1.92\times10^{-1}/+$	$1.12\times10^{5}/1.71\times10^{5}/3.59\times10^{4}/+$
f_8	$\mathbf{1.83\times10^{-92}/3.97\times10^{-14}/2.14\times10^{-13}}$	$2.75\times10^{-4}/2.47\times10^{0}/4.13\times10^{0}/+$	$2.09\times10^{0}/3.09\times10^{3}/4.41\times10^{3}/+$	$8.10\times10^{9}/1.45\times10^{10}/2.98\times10^{9}/+$
f_9	$\mathbf{1.63\times10^{-6}/5.01\times10^{-4}/7.13\times10^{-4}}$	$1.85\times10^{-4}/1.96\times10^{-3}/1.35\times10^{-3}/+$	$4.13\times10^{-4}/3.09\times10^{-3}/2.43\times10^{-3}/+$	$4.77\times10^{0}/1.14\times10^{1}/3.14\times10^{0}/+$
f_{10}	$\mathbf{1.92\times10^{-9}/1.95\times10^{1}/1.32\times10^{1}}$	$2.89\times10^{1}/2.89\times10^{1}/\mathbf{1.81\times10^{-3}}/+$	$\mathbf{6.81\times10^{-4}}/7.48\times10^{-1}/9.26\times10^{-1}/–$	$9.89\times10^{8}/3.3\times10^{9}/1.05\times10^{9}/+$
f_{11}	$\mathbf{0.00\times10^{0}}/5.56\times10^{-3}/1.87\times10^{-2}$	$9.91\times10^{-8}/\mathbf{3.72\times10^{-4}/9.37\times10^{-4}}/+$	$1.24\times10^{-6}/1.40\times10^{-2}/2.60\times10^{-2}/+$	$7.98\times10^{1}/1.51\times10^{2}/2.53\times10^{1}/+$
f_{12}	$\mathbf{4.44\times10^{-16}/3.11\times10^{-11}/1.55\times10^{-10}}$	$5.67\times10^{-5}/1.22\times10^{-3}/1.07\times10^{-3}/+$	$8.11\times10^{-4}/1.44\times10^{-2}/1.40\times10^{-2}/+$	$1.56\times10^{1}/1.71\times10^{1}/6.26\times10^{-1}/+$
f_{13}	$\mathbf{0.00\times10^{0}/3.79\times10^{-15}/2.04\times10^{-14}}$	$4.63\times10^{-9}/6.36\times10^{-6}/1.03\times10^{-5}/+$	$4.23\times10^{-8}/1.99\times10^{-5}/5.83\times10^{-5}/+$	$2.15\times10^{1}/5.11\times10^{1}/1.35\times10^{1}/+$
f_{14}	$\mathbf{0.00\times10^{0}/6.33\times10^{-4}/3.41\times10^{-3}}$	$3.23\times10^{-2}/5.87\times10^{-1}/3.73\times10^{-1}/+$	$6.76\times10^{-2}/5.11\times10^{-1}/3.03\times10^{-1}/+$	$2.77\times10^{1}/3.22\times10^{1}/1.30\times10^{0}/+$
f_{15}	$\mathbf{0.00\times10^{0}/4.44\times10^{-17}/2.39\times10^{-16}}$	$4.32\times10^{-11}/9.05\times10^{-8}/1.92\times10^{-7}+$	$2.56\times10^{-9}/2.39\times10^{-7}/3.03\times10^{-7}+$	$5.26\times10^{-1}/7.75\times10^{-1}/1.42\times10^{-1}/+$
f_{16}	$\mathbf{3.53\times10^{-54}/1.47\times10^{-44}/7.80\times10^{-45}}$	$4.76\times10^{-6}/5.16\times10^{-4}/6.58\times10^{-4}/+$	$2.81\times10^{-4}/4.70\times10^{-2}/4.34\times10^{-2}/+$	$9.22\times10^{0}/1.31\times10^{1}/1.62\times10^{0}/+$
f_{17}	$\mathbf{6.28\times10^{-29}/6.42\times10^{-11}/3.39\times10^{-10}}$	$2.36\times10^{-2}/7.38\times10^{-2}/3.73\times10^{-2}/+$	$8.24\times10^{-8}/2.27\times10^{-4}/4.42\times10^{-4}/+$	$9.06\times10^{6}/3.62\times10^{7}/1.37\times10^{7}/+$
f_{18}	$\mathbf{2.08\times10^{-27}}/5.97\times10^{-2}/1.04\times10^{-1}$	$2.21\times10^{-1}/6.08\times10^{-1}/2.09\times10^{-1}/+$	$1.44\times10^{-6}/\mathbf{1.67\times10^{-2}/2.57\times10^{-2}}/\approx$	$4.50\times10^{7}/9.86\times10^{7}/2.87\times10^{7}/+$
Statistic test (+/≈/−)	–	16/1/1	14/2/2	18/0/0

Table 3 Comparison of QDMA and state-of-the-art QEAs on 50*D* problems.

Function	QDMA	MSIQDE	EMMSIQDE	HMCFQDE
	Bev/Mev/Std	Bev/Mev/Std/Test	Bev/Mev/Std/Test	Bev/Mev/Std/Test
f_1	$\mathbf{4.19\times10^{-91}/5.84\times10^{-15}/0.00\times10^{0}}$	$2.64\times10^{-10}/1.17\times10^{-6}/1.53\times10^{-6}/+$	$1.39\times10^{-8}/1.45\times10^{-2}/3.48\times10^{-2}/+$	$3.21\times10^{4}/4.64\times10^{4}/4.57\times10^{3}/+$
f_2	$\mathbf{0.00\times10^{0}}/3.32\times10^{3}/1.79\times10^{4}$	$9.34\times10^{-8}/\mathbf{2.69\times10^{-3}/4.67}\times10^{-3}/+$	$9.29\times10^{-3}/7.25\times10^{0}/1.38\times10^{1}/+$	$7.42\times10^{4}/9.94\times10^{4}/1.10\times10^{4}/+$
f_3	$\mathbf{3.46\times10^{-99}}/1.72\times10^{5}/9.24\times10^{5}$	$1.26\times10^{-9}/\mathbf{1.99\times10^{-3}/3.96}\times10^{-3}/+$	$1.58\times10^{0}/2.73\times10^{2}/4.18\times10^{2}/+$	$1.64\times10^{8}/2.49\times10^{8}/4.54\times10^{7}/+$
f_4	$\mathbf{1.27\times10^{-43}}/5.12\times10^{1}/4.35\times10^{1}$	$9.90\times10^{-5}/\mathbf{1.35\times10^{-3}/1.11}\times10^{-3}/–$	$1.49\times10^{-3}/3.12\times10^{-1}/2.68\times10^{-1}/–$	$7.68\times10^{1}/9.27\times10^{1}/8.60\times10^{0}/+$
f_5	$\mathbf{5.96\times10^{-44}/6.82\times10^{-44}/2.18\times10^{-45}}$	$6.12\times10^{-5}/1.29\times10^{-3}/1.24\times10^{-3}/+$	$1.24\times10^{-3}/1.38\times10^{-2}/1.19\times10^{-2}/+$	$6.70\times10^{1}/7.09\times10^{1}/1.66\times10^{0}/+$
f_6	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}/\approx$	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}/\approx$	$3.51\times10^{4}/4.69\times10^{4}/4.62\times10^{3}/+$
f_7	$\mathbf{0.00\times10^{0}/7.42\times10^{-21}/4.00\times10^{-20}}$	$2.66\times10^{-10}/1.34\times10^{-5}/3.06\times10^{-5}/+$	$5.17\times10^{-4}/3.94\times10^{-1}/7.67\times10^{-1}/+$	$7.14\times10^{5}/9.82\times10^{5}/1.01\times10^{5}/+$
f_8	$\mathbf{0.00\times10^{0}/3.20\times10^{0}/1.46\times10^{1}}$	$8.66\times10^{-4}/2.61\times10^{0}/3.96\times10^{0}/+$	$9.79\times10^{0}/8.22\times10^{3}/9.83\times10^{3}/+$	$3.61\times10^{10}/4.52\times10^{10}/4.43\times10^{9}/+$
f_9	$\mathbf{1.28\times10^{6}/6.28\times10^{-4}/9.90\times10^{-4}}$	$1.02\times10^{-4}/1.77\times10^{-3}/1.12\times10^{-3}/+$	$7.52\times10^{-5}/2.12\times10^{-3}/1.53\times10^{-3}/+$	$4.66\times10^{1}/8.02\times10^{1}/1.30\times10^{1}/+$
f_{10}	$\mathbf{4.49\times10^{-6}/3.96\times10^{1}}/1.11\times10^{1}$	$4.88\times10^{1}/4.89\times10^{1}/\mathbf{2.07\times10^{-2}}/+$	$5.25\times10^{-3}/4.28\times10^{0}/9.14\times10^{0}/–$	$1.17\times10^{10}/1.48\times10^{10}/1.47\times10^{9}/+$
f_{11}	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$1.47\times10^{-6}/4.01\times10^{-4}/7.42\times10^{-4}/+$	$8.10\times10^{-6}/8.89\times10^{-3}/1.16\times10^{-2}/+$	$3.45\times10^{2}/4.30\times10^{2}/3.25\times10^{1}/+$
f_{12}	$\mathbf{4.44\times10^{-16}/4.44\times10^{-16}/0.00\times10^{0}}$	$2.53\times10^{-6}/4.19\times10^{-4}/6.55\times10^{-4}/+$	$1.40\times10^{-4}/1.71\times10^{-2}/1.74\times10^{-2}/+$	$1.83\times10^{1}/1.88\times10^{1}/2.23\times10^{-1}/+$
f_{13}	$\mathbf{0.00\times10^{0}/1.14\times10^{-14}/6.12\times10^{-14}}$	$1.41\times10^{-9}/5.13\times10^{-6}/9.69\times10^{-6}/+$	$3.68\times10^{-10}/4.79\times10^{-5}/7.51\times10^{-5}/+$	$1.24\times10^{2}/1.56\times10^{2}/1.29\times10^{1}/+$
f_{14}	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$7.56\times10^{-2}/8.92\times10^{-1}/6.76\times10^{-1}/+$	$1.12\times10^{-1}/7.54\times10^{-1}/4.17\times10^{-1}/+$	$5.67\times10^{1}/6.15\times10^{1}/1.66\times10^{0}/+$
f_{15}	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$6.22\times10^{-11}/1.05\times10^{-7}/2.47\times10^{-7}/+$	$6.56\times10^{-10}/4.45\times10^{-7}/5.67\times10^{-7}/+$	$2.00\times10^{0}/2.28\times10^{0}/1.66\times10^{-1}/+$
f_{16}	$\mathbf{2.04\times10^{-44}/2.56\times10^{-44}/2.52\times10^{-45}}$	$4.26\times10^{-7}/2.58\times10^{-4}/2.32\times10^{-4}/+$	$1.76\times10^{-3}/4.19\times10^{-2}/4.13\times10^{-2}/+$	$2.08\times10^{1}/2.25\times10^{1}/8.66\times10^{-1}/+$
f_{17}	$\mathbf{2.32\times10^{-27}}/1.84\times10^{-4}/6.62\times10^{-4}$	$2.52\times10^{-2}/7.18\times10^{-2}/3.49\times10^{-2}/+$	$1.70\times10^{-7}/\mathbf{1.54\times10^{-4}/2.97\times10^{-4}}/+$	$9.79\times10^{7}/1.87\times10^{8}/3.59\times10^{7}/+$
f_{18}	$\mathbf{7.54\times10^{-25}/7.79\times10^{-4}/2.81\times10^{-2}}$	$3.36\times10^{-1}/1.07\times10^{0}/3.31\times10^{-1}/+$	$2.03\times10^{-5}/2.46\times10^{-2}/4.08\times10^{-2}/\approx$	$2.53\times10^{8}/4.63\times10^{8}/7.87\times10^{7}/+$
Statistic test (+/≈/−)	–	16/1/1	14/2/2	18/0/0

competitors in solving all six test functions. Second, QDMA converges to the solutions at a faster speed. Third, the solution of each function obtained by QDMA is equal to or very close to zero, i.e., the optimal solution. Therefore, we can conclude that QDMA clearly outperforms the compared three state-

Table 4 Comparison of QDMA and state-of-the-art QEAs on 100*D* problems.

Function	QDMA	MSIQDE	EMMSIQDE	HMCFQDE
	Bev/Mev/Std	Bev/Mev/Std/Test	Bev/Mev/Std/Test	Bev/Mev/Std/Test
f_1	**$1.18\times10^{-85}/1.46\times10^{-85}/1.94\times10^{-86}$**	$1.76\times10^{-9}/5.15\times10^{-6}/7.17^{-6}$/+	$1.17\times10^{-5}/2.01\times10^{-2}/4.14\times10^{-2}$/+	$1.20\times10^{5}/1.34\times10^{5}/7.15\times10^{3}$/+
f_2	**4.05×10^{-86}**/$3.65\times10^{4}/9.37\times10^{4}$	2.43×10^{-9}/**$4.33\times10^{-3}/6.22\times10^{-3}$/+**	$1.20\times10^{-2}/7.41\times10^{1}/1.75\times10^{2}$/+	$2.90\times10^{5}/4.29\times10^{5}/6.68\times10^{4}$/+
f_3	**$6.12\times10^{-81}/1.10\times10^{-80}/3.30\times10^{-81}$**	$3.12\times10^{-7}/2.26\times10^{-3}/6.16\times10^{-3}$/+	$3.39\times10^{-1}/1.06\times10^{3}/1.91\times10^{3}$/+	$1.68\times10^{9}/2.09\times10^{9}/2.96\times10^{8}$/+
f_4	**2.83×10^{-43}**/$7.68\times10^{4}/3.15\times10^{5}$	1.39×10^{-4}/**$2.31\times10^{-3}/2.07\times10^{-3}$/–**	$8.69\times10^{-3}/1.13\times10^{0}/1.17\times10^{0}$/–	$1.97\times10^{2}/4.46\times10^{7}/1.71\times10^{8}$/+
f_5	**$6.48\times10^{-44}/6.91\times10^{-44}/1.24\times10^{-45}$**	$4.42\times10^{-5}/2.62\times10^{-3}/2.60\times10^{-3}$/+	$6.24\times10^{-7}/9.11\times10^{-3}/9.32\times10^{-3}$/+	$7.84\times10^{1}/8.11\times10^{1}/1.29\times10^{0}$/+
f_6	**$0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}$**	**$0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}$/≈**	**$0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}$/≈**	$1.15\times10^{5}/1.37\times10^{5}/8.49\times10^{3}$/+
f_7	**$7.15\times10^{-86}/7.33\times10^{-84}/1.75\times10^{-84}$**	$5.53\times10^{-8}/1.23\times10^{-5}/2.17\times10^{-5}$/+	$1.14\times10^{-3}/1.17\times10^{0}/2.61\times10^{0}$/+	$5.11\times10^{6}/6.06\times10^{6}/4.10\times10^{5}$/+
f_8	**$1.12\times10^{-79}/1.45\times10^{-79}/2.00\times10^{-80}$**	$1.07\times10^{-2}/2.37\times10^{0}/4.80\times10^{0}$/+	$1.17\times10^{0}/2.63\times10^{4}/6.65\times10^{4}$/+	$1.22\times10^{11}/1.36\times10^{11}1/5.70\times10^{9}$/+
f_9	**$1.56\times10^{-6}/6.75\times10^{-4}/1.07\times10^{-3}$**	$2.52\times10^{-4}/1.96\times10^{-3}/1.44\times10^{-3}$/+	$6.79\times10^{-4}/4.56\times10^{-3}/3.64\times10^{-3}$/+	$3.72\times10^{2}/6.08\times10^{2}/5.81\times10^{1}$/+
f_{10}	**6.61×10^{-4}**/$8.54\times10^{1}/2.79\times10^{1}$	$9.86\times10^{1}/9.89\times10^{1}$/**$5.73\times10^{-2}$**/+	9.90×10^{-4}/**6.18×10^{0}**/1.26×10^{1}/–	$4.19\times10^{10}/5.11\times10^{10}/3.19\times10^{9}$/+
f_{11}	**$0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}$**	$7.93\times10^{-8}/1.71\times10^{-4}/2.20\times10^{-4}$/+	$1.13\times10^{-7}/1.49\times10^{-2}/2.23\times10^{-2}$/+	$1.08\times10^{3}/1.23\times10^{3}/6.73\times10^{1}$/+
f_{12}	**$4.44\times10^{-16}/5.63\times10^{-16}/6.38\times10^{-16}$**	$4.42\times10^{-6}/2.95\times10^{-4}/3.21\times10^{-4}$/+	$1.88\times10^{-4}/1.01\times10^{-2}/1.36\times10^{-2}$/+	$1.96\times10^{1}/1.98\times10^{1}/9.39\times10^{-2}$/+
f_{13}	**$0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}$**	$1.28\times10^{-9}/7.31\times10^{-6}/1.18\times10^{-5}$/+	$3.37\times10^{-8}/7.59\times10^{-5}/1.68\times10^{-4}$/+	$4.09\times10^{2}/4.50\times10^{2}/1.92\times10^{1}$/+
f_{14}	**$0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}0$**	$4.60\times10^{-2}/2.09\times10^{0}/1.32\times10^{0}$+	$2.39\times10^{-1}/1.61\times10^{0}/1.02\times10^{0}$/+	$1.32\times10^{2}/1.3\times10^{2}/3.98\times10^{0}$/+
f_{15}	**$0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}$**	$7.32\times10^{-10}/2.23\times10^{-7}/4.98\times10^{-7}$/+	$6.23\times10^{-11}/1.07\times10^{-6}/1.34\times10^{-6}$/+	$6.21\times10^{0}/6.57\times10^{0}/1.96\times10^{-1}$/+
f_{16}	**$3.34\times10^{-44}/3.79\times10^{-44}/2.55\times10^{-45}$**	$7.85\times10^{-6}/2.90\times10^{-4}/3.94\times10^{-4}$/+	$1.98\times10^{-4}/5.61\times10^{-2}/5.18\times10^{-2}$/+	$3.52\times10^{1}/3.74\times10^{1}/8.73\times10^{-1}$/+
f_{17}	**$2.13\times10^{-22}/1.24\times10^{-4}/3.01\times10^{-4}$**	$2.94\times10^{-2}/7.87\times10^{-2}/3.17\times10^{-2}$/+	$7.10\times10^{-8}/7.22\times10^{-4}/9.19\times10^{-4}$/+	$5.26\times10^{8}/7.74\times10^{8}/9.81\times10^{7}$/+
f_{18}	**$6.38\times10^{-12}/1.03\times10^{-2}/1.34\times10^{-2}$**	$5.77\times10^{-1}/1.84\times10^{0}/5.59\times10^{-1}$/+	$2.20\times10^{-8}/1.90\times10^{-2}/2.56\times10^{-2}$/+	$1.30\times10^{9}/1.71\times10^{9}/1.44\times10^{8}$/+
Statistic test (+/≈/−)	−	16/1/1	15/1/2	18/0/0

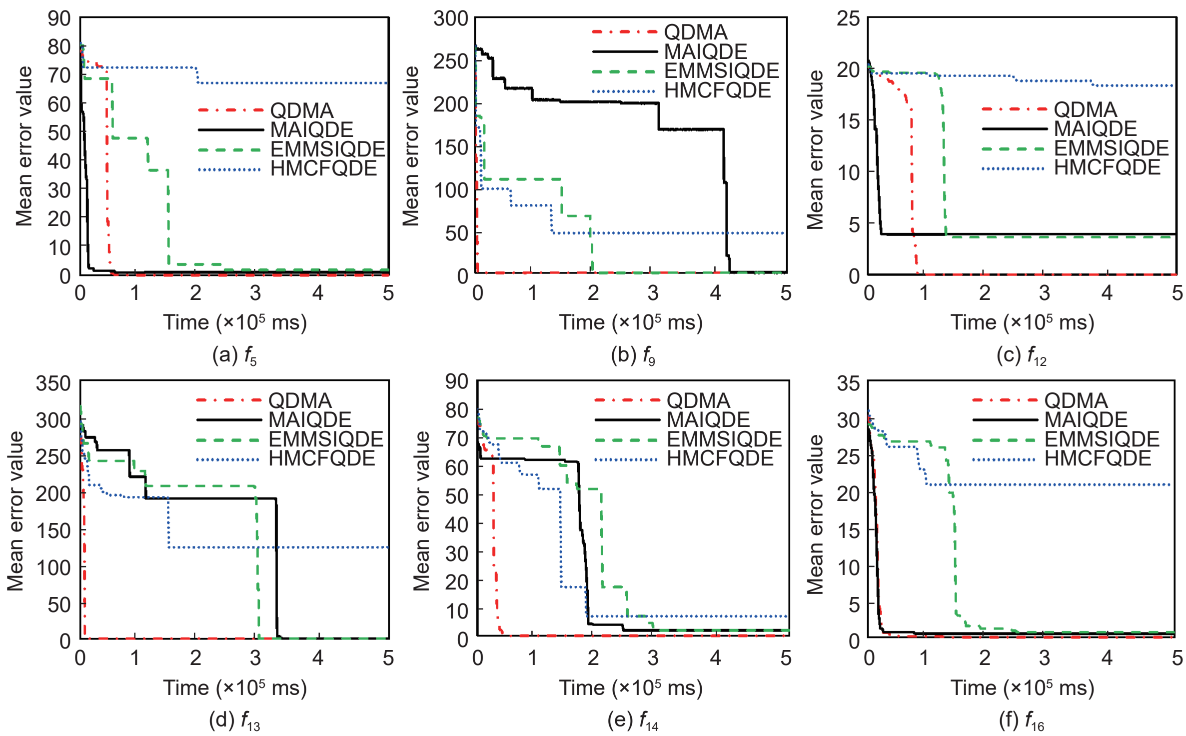

Fig. 5 Convergence graphs of QDMA and QEA variants on 50*D* problems.

of-the-art QEA variants. In addition, MSIQDE and EMMSIQDE show similar performance, and their convergence solutions are also very competitive. Overall, in terms of convergence speed and accuracy, the performance of HMCFQDE is worst in solving each test function.

4.3 Comparison with different DE algorithms

Next, the QDMA is further compared against three efficient DE variants SaDE[43], SaNSDE[44], and SaNSDE+[45], and a state-of-the-art distributed DE variant ADDE[18]. Table 5 shows the statistical results between QDMA and the candidate algorithms, while the detailed comparison results are attached from Tables A2−A5 in the Appendix.

For the 10*D* problems in Table A2 in the Appendix, we can see that QDMA performs clearly better than all compared algorithms according to the Bev index. It can generate the better Bev values in 15 of the 18 functions, and finds 12 global optimal solutions of the 18 functions. According to the Mev and Std indexes, although QDMA clearly wins ADDE, we also see that it is dominated by SaDE, SaNSDE, and SaNSDE+ on almost all functions except f_6, f_{13}, and f_{15}. These results suggest that QDMA has efficient global optimization ability but its robustness needs to be further improved in solving small-scaled problems.

For the 30*D* problems in Table A3 in the Appendix, the conclusion is overall the same as those in the 10*D* problems. As the increase of problem scale, we can see that QDMA starts to perform better in more Mev and Std indexes. For the 50*D* problems in Table A4 in the Appendix, QDMA still keeps 15 better Bev values. However, the number of the better Mev and Std indexes has increased to 11. This implies that QDMA is gradually improving its robustness and has the better performance in solving more complex problems.

For the 100*D* problems in Table A5 in the Appendix, QDMA shows a more obvious dominant position among the compared algorithms no matter which of the three indexes is. In the Bev index, QDMA obtains the better values in 17 of the 18 functions, except that SaDE wins it on f_{18}. In the Mev and Std indexes, QDMA obtains the better values in 15 of the 18 functions. It is only won by SaDE on f_4 in Mev index, SaNSDE on f_4 in Std index, and SaDE on f_{17} and f_{18} in both Mev and Std indexes. This fully shows the superiority of QDMA in addressing the large-scaled optimization problems.

Table 5 Comparison of QDMA and state-of-the-art DE variants.

Dimension	Statistical result				
	QDMA	SaDE	SaNSDE	SaNSDE+	ADDE
10*D*	+	3	1	2	18
	≈	7	7	8	0
	−	8	10	8	0
30*D*	+	6	5	11	18
	≈	5	4	3	0
	−	7	9	4	0
50*D*	+	11	9	15	18
	≈	5	4	1	0
	−	2	5	2	0
100*D*	+	13	13	17	18
	≈	2	2	0	0
	−	3	3	1	0

The statistic test in Table 5 further confirms the analysis and conclusion obtained above. Overall, QDMA only wins SaDE, SaNSDE, and SaNSDE+ on 3, 1, and 2 functions, respectively, in 10*D* problems. These data are swiftly improved when we add the problem scales. For example, there are 6, 5, and 11 functions in 30*D* problems, 11, 9, and 15 functions in 50*D* problems, and 13, 13, and 17 functions in 100*D* problems. QDMA performs better than ADDE in all benchmark functions and measurement indexes. Therefore, QDMA shows the best performance among the compared algorithms.

We further give the convergence graphs of these algorithms to analyze the evolutionary process. The convergence graphs are shown in Fig. 6. As can be seen from Fig. 6, QDMA has the best convergence speed and optimization accuracy for all six test functions. Figure 6 also shows that ADDE performs worst on all the test functions except f_{14} shown in Fig. 6e. On function f_{14}, its performance is second only to QDMA, that is, better than another three DE variants. In addition, the performances of SaDE, SaNSDE, and SaNSDE+ are very close and between QDMA and ADDE. Therefore, we can conclude that the proposed QDMA outperforms the compared three effective and efficient DE algorithms and the distributed DE algorithm at a considerable margin.

4.4 Effect of QDMA special designs

The key special designs of QDMA are distributed quantum evolution on QPs, intensified quantum evolution on EAR, and knowledge transfer with point-ring topology. Next, we discuss the effect of these special designs on the overall performance of QDMA. For this purpose, three variants of QDMA are tested. They are denoted as QDMA_k, $k \in \{1, 2, 3\}$. QDMA_1 is a single QEA obtained by removing the modules of distributed quantum evolution on QPs and intensified quantum evolution on EAR from QDMA. QDMA_2 is QDMA_1 plus the distributed quantum evolution module. QDMA_3 is QDMA_2 combing the intensified quantum evolution module. Overall QDMA is QDMA_3 plus the knowledge transfer module. In addition, the basic DE algorithm with a single population and the

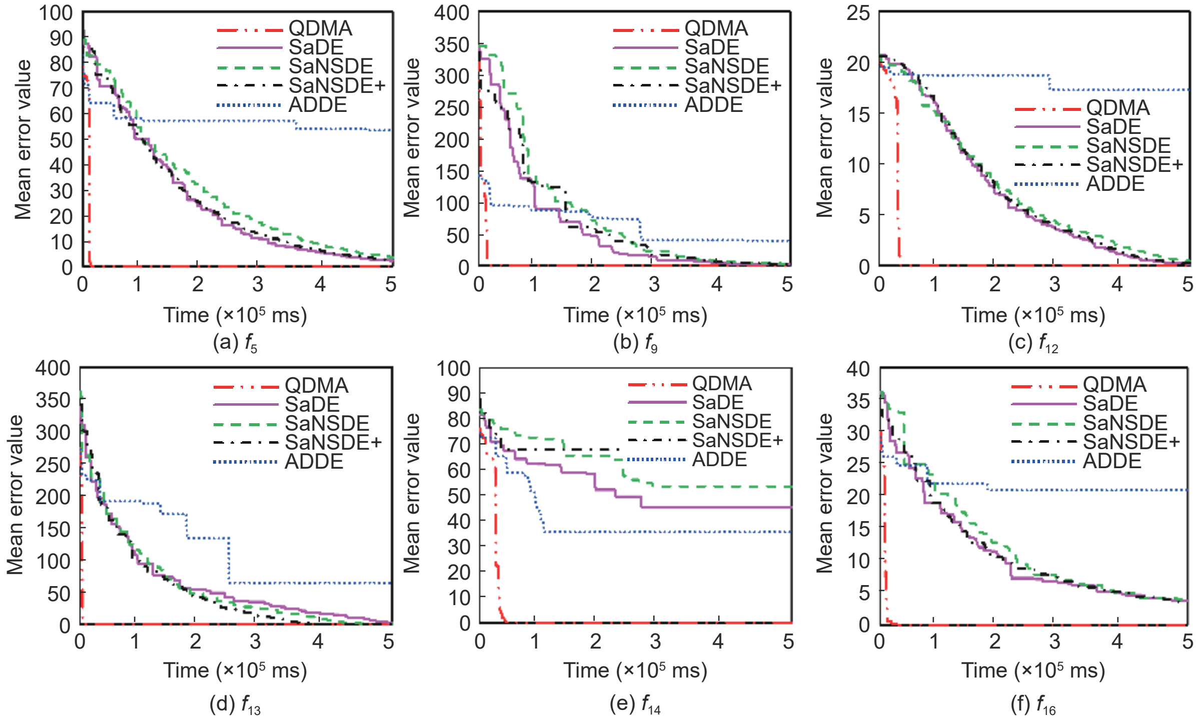

Fig. 6 Convergence graphs of QDMA and DE variants on 50*D* problems.

DE/rand/1 mutation operator is used as the reference algorithm. For the test functions, we only consider the 50*D* dimension. Table 6 shows the computational results.

From the comparison between QDMA_1 and DE in Table 6, QDMA_1 performs better in each of three indexes on all tested functions, except the Std index on f_5, f_9, and f_{14}. Therefore, it can be concluded that the proposed QEA is effective. This is because this QEA is built by introducing the quantum evolution into the

Table 6 Comparison of QDMA and its special designs on 50*D* problems.

Function	QDMA	DE	QDMA_1	QDMA_2	QDMA_3
	Bev/Mev/Std	Bev/Mev/Std/Test	Bev/Mev/Std/Test	Bev/Mev/Std/Test	Bev/Mev/Std/Test
f_1	$\mathbf{4.19\times10^{-91}/5.84\times10^{-15}/0.00\times10^{0}}$	$2.64\times10^{3}/2.11\times10^{4}/1.24\times10^{4}/+$	$3.30\times10^{-27}/1.47\times10^{-1}/7.62\times10^{-1}/+$	$3.20\times10^{-12}/2.33\times10^{1}/1.18\times10^{2}/+$	$4.95\times10^{-1}/2.94\times10^{3}/4.32\times10^{3}/+$
f_2	$\mathbf{0.00\times10^{0}/3.32\times10^{3}/1.79\times10^{3}}$	$5.77\times10^{4}/7.29\times10^{4}/7.98\times10^{3}/+$	$4.72\times10^{4}/6.82\times10^{4}/6.51\times10^{3}/+$	$1.42\times10^{4}/5.73\times10^{4}/2.00\times10^{4}/+$	$1.82\times10^{-1}/4.66\times10^{4}/3.45\times10^{4}/+$
f_3	$\mathbf{3.46\times10^{-99}/1.72\times10^{5}}/9.24\times10^{5}$	$1.31\times10^{7}/9.82\times10^{7}/8.83\times10^{7}/+$	$4.58\times10^{4}/2.71\times10^{6}/2.61\times10^{6}/+$	$4.33\times10^{-12}/3.19\times10^{4}/\mathbf{5.97\times10^{4}}/+$	$5.87\times10^{1}/2.40\times10^{6}/1.02\times10^{7}/+$
f_4	$\mathbf{1.27\times10^{-43}/5.12\times10^{1}}/4.35\times10^{1}$	$1.03\times10^{2}/2.80\times10^{2}/8.61\times10^{2}/+$	$8.45\times10^{1}/1.71\times10^{2}/3.38\times10^{2}/+$	$3.83\times10^{1}/9.89\times10^{1}/\mathbf{2.30\times10^{1}}/+$	$4.65\times10^{1}/1.40\times10^{1}/2.06\times10^{1}/+$
f_5	$\mathbf{5.96\times10^{-44}/6.82\times10^{-44}/2.18\times10^{-45}}$	$5.96\times10^{1}/6.59\times10^{1}/2.33\times10^{0}/+$	$5.63\times10^{1}/6.53\times10^{1}/2.49\times10^{0}/+$	$1.56\times10^{1}/6.09\times10^{1}/1.15\times10^{1}/+$	$2.01\times10^{-1}/2.22\times10^{1}/2.24\times10^{1}/+$
f_6	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$6.96\times10^{3}/2.20\times10^{4}/1.12\times10^{4}/+$	$\mathbf{0.00\times10^{0}}/5.46\times10^{1}/7.76\times10^{1}/+$	$1.20\times10^{1}/7.48\times10^{2}/1.56\times10^{3}/+$	$\mathbf{0.00\times10^{0}}/3.94\times10^{2}/5.80\times10^{3}/+$
f_7	$\mathbf{0.00\times10^{0}/7.42\times10^{-21}/4.00\times10^{-20}}$	$7.10\times10^{4}/5.43\times10^{5}/2.08\times10^{5}/+$	$4.36\times10^{3}/8.12\times10^{4}/6.15\times10^{4}/+$	$5.24\times10^{-2}/3.98\times10^{4}/8.53\times10^{4}/+$	$2.13\times10^{-2}/2.91\times10^{4}/3.61\times10^{4}/+$
f_8	$\mathbf{0.00\times10^{0}}/3.20\times10^{0}/1.46\times10^{1}$	$3.72\times10^{9}/2.73\times10^{10}/9.25\times10^{9}/+$	$3.07\times10^{-21}/9.08\times10^{5}/3.88\times10^{6}/+$	$8.38\times10^{-21}/\mathbf{1.23\times10^{0}/6.64\times10^{0}}/+$	$3.39\times10^{1}/1.59\times10^{9}/2.44\times10^{9}/+$
f_9	$\mathbf{1.28\times10^{-6}/6.28\times10^{-4}/9.90\times10^{-4}}$	$3.18\times10^{1}/5.55\times10^{1}/1.10\times10^{1}/+$	$1.23\times10^{1}/4.35\times10^{1}/1.12\times10^{1}/+$	$1.51\times10^{1}/4.10\times10^{1}/1.31\times10^{1}/+$	$1.04\times10^{-1}/9.94\times10^{0}/1.65\times10^{1}/+$
f_{10}	$\mathbf{4.49\times10^{-6}/3.96\times10^{1}/1.11\times10^{1}}$	$6.03\times10^{9}/1.11\times10^{10}/1.93\times10^{9}/+$	$5.80\times10^{6}/2.89\times10^{9}/1.67\times10^{9}/+$	$2.19\times10^{2}/3.31\times10^{9}/2.43\times10^{9}/+$	$1.53\times10^{2}/2.76\times10^{9}/4.13\times10^{9}/+$
f_{11}	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$8.33\times10^{1}/3.05\times10^{2}/6.93\times10^{1}/+$	$2.45\times10^{0}/6.10\times10^{1}/5.28\times10^{1}/+$	$\mathbf{0.00\times10^{0}}/2.24\times10^{-2}/4.20\times10^{-2}/+$	$9.85\times10^{-2}/3.98\times10^{1}/5.40\times10^{1}/+$
f_{12}	$\mathbf{4.44\times10^{-16}/4.44\times10^{-16}/0.00\times10^{0}}$	$1.20\times10^{1}/1.71\times10^{1}/1.49\times10^{0}/+$	$1.72\times10^{0}/3.19\times10^{0}/7.35\times10^{-1}/+$	$3.10\times10^{0}/8.02\times10^{0}/2.68\times10^{0}/+$	$5.97\times10^{-1}/7.01\times10^{0}/3.80\times10^{0}/+$
f_{13}	$\mathbf{0.00\times10^{0}/1.14\times10^{-14}/6.12\times10^{-14}}$	$1.68\times10^{1}/6.49\times10^{1}/3.09\times10^{1}/+$	$\mathbf{0.00\times10^{0}}/4.71\times10^{-3}/2.41\times10^{-2}/+$	$2.09\times10^{-9}/3.97\times10^{-3}/2.13\times10^{-2}/+$	$1.89\times10^{-6}/1.40\times10^{0}/4.15\times10^{0}/+$
f_{14}	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$4.81\times10^{1}/5.88\times10^{1}/3.24\times10^{0}/+$	$3.41\times10^{1}/5.22\times10^{1}/6.11\times10^{0}/+$	$4.12\times10^{0}/8.31\times10^{0}/2.43\times10^{0}/+$	$2.31\times10^{0}/3.22\times10^{1}/1.50\times10^{1}/+$
f_{15}	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$7.37\times10^{-1}/1.54\times10^{0}/2.80\times10^{-1}/+$	$1.03\times10^{-4}/2.82\times10^{-1}/2.03\times10^{-1}/+$	$7.93\times10^{-8}/9.66\times10^{-2}/1.48\times10^{-1}/+$	$1.50\times10^{-5}/9.90\times10^{-2}/1.60\times10^{-1}/+$
f_{16}	$\mathbf{2.04\times10^{-44}/2.56\times10^{-44}/2.52\times10^{-45}}$	$8.03\times10^{0}/1.67\times10^{1}/3.02\times10^{0}/+$	$5.00\times10^{-1}/1.90\times10^{0}/8.71\times10^{-1}/+$	$6.00\times10^{-1}/1.91\times10^{0}/1.02\times10^{0}/+$	$1.16\times10^{-1}/5.74\times10^{0}/3.96\times10^{0}/+$
f_{17}	$\mathbf{2.32\times10^{-27}/1.84\times10^{-4}/6.62\times10^{-4}}$	$6.56\times10^{7}/1.31\times10^{8}/3.21\times10^{7}/+$	$4.81\times10^{7}/1.24\times10^{8}/3.07\times10^{7}/+$	$4.31\times10^{0}/1.05\times10^{8}/2.08\times10^{7}/+$	$3.93\times10^{-1}/2.97\times10^{7}/5.88\times10^{7}/+$
f_{18}	$\mathbf{7.54\times10^{-25}/7.79\times10^{-4}/2.81\times10^{-2}}$	$2.16\times10^{8}/3.08\times10^{8}/6.00\times10^{7}/+$	$1.32\times10^{8}/1.96\times10^{8}/3.45\times10^{7}/+$	$1.84\times10^{2}/1.59\times10^{8}/2.63\times10^{7}/+$	$4.59\times10^{0}/5.91\times10^{7}/1.10\times10^{8}/+$
Statistic test (+/≈/−)	−	18/0/0	18/0/0	18/0/0	18/0/0

framework of basic DE, which can effectively enhance the evolutionary diversity. Compared with $QDMA_1$, $QDMA_2$ achieves the better performance on 13 functions for the Bev and Mev indexes. It indicates that the distributed quantum evolution module is effective in the evolutionary process of QDMA. By further integrating EAR-driven intensified quantum evolution and executing it in parallel with distributed quantum evolution, $QDMA_3$ outperforms $QDMA_2$ on 11 functions in the Bev index and on 10 functions in the Mev index. As a result, the proposed local search strategy and the parallel evolutionary design with the global search are successful. Finally, QDMA significantly outperforms all considered variants, which demonstrates that the knowledge transfer strategy plays an important role for enhancing the performance of overall QDMA.

From the above analysis, we can see that the special designs proposed in QDMA are effective and very important to achieve the superior overall performance of QDMA.

5 Conclusion

To achieve the high-quality global optimization, a QDMA metaheuristic is proposed based on quantum computation and a novel distributed memetic evolutionary framework. In QDMA, individuals are represented and evolved by quantum computing characteristics, which can effectively enhance the evolutionary diversity. Within the proposed framework, four modules called distributed quantum evolution, intensified quantum evolution, knowledge transfer, and evolutionary restart are cooperated, which can maximize their strengths and achieve superior global optimality. For details, distributed quantum evolution explores three populations independently by the heterogenous operators. Intensified quantum evolution exploits an external elite archive to balance global and local searches. Knowledge transfer uses a point-ring topology to exchange successful experiences among all search agents. Evolutionary restart module uses an adaptive perturbation strategy to control the diversity of the populations reasonably. Extensive computational experiments are executed to evaluate the proposed algorithm. The results demonstrate the effectiveness of each special design and show that the QDMA can outperform the compared state-of-the-art algorithms based on Wilcoxon's rank-sum test. These superiorities are attributed not only to excellent cooperative coevolution mechanism from distributed memetic evolutionary framework, but also to good designs of each special component.

Inspired by the superior performance, in future work we will apply the QDMA to deal with large-scale and multiobjective optimization[47] as well as some practical applications such as intelligent optimization for manufacturing scheduling[48].

Appendix

In this appendix, Table A1 is the test functions used in the experiment, and Tables A2−A5 are the detailed comparison results between our proposed QDMA, SaDE, SaNSDE, SaNSDE+, and ADDE in Section 4.3.

Table A1 Test functions.

Function	Expression	Variable range	Optimal value
f_1	$f_1(x)=\sum_{i=1}^{D} x_i^2$	[−100, 100]	0
f_2	$f_2(x)=\sum_{i=1}^{D}(\sum_{j=1}^{i} x_j)^2$	[−100, 100]	0
f_3	$f_3(x)=\sum_{i=1}^{D}(10^6)^{\frac{i-1}{D-1}} x_i^2$	[−100, 100]	0
f_4	$f_4(x)=\sum_{i=1}^{D}\lvert x_i\rvert+\prod_{i=1}^{D}\lvert x_i\rvert$	[−10, 10]	0
f_5	$f_5(x)=\max\{\lvert x_i\rvert, 1\leqslant i\leqslant D\}$	[−100, 100]	0
f_6	$f_6(x)=\sum_{i=1}^{D}(\lfloor x_i+0.5\rfloor)^2$	[−100, 100]	0
f_7	$f_7(x)=\sum_{i=1}^{D}(ix_i^2)$	[−100, 100]	0
f_8	$f_8(x)=\sum_{i=1}^{D}[\sum_{j=1}^{i}(\sum_{k=1}^{i} x_k)]^2$	[−100, 100]	0
f_9	$f_9(x)=\sum_{i=1}^{D} ix_i^4+\text{random}[0,1)$	[−1.28, 1.28]	0
f_{10}	$f_{10}(x)=\sum_{i=1}^{D-1}[100(x_{i+1}-x_i^2)^2+(1-x_i)^2]$	[−100, 100]	0

(to be continued)

Table A1 Test functions. (continued)

Function	Expression	Variable range	Optimal value
f_{11}	$f_{11}(x)=\sum_{i=1}^{D}\frac{x_i^2}{4000}-\prod_{i=1}^{D}\cos(\frac{x_i}{\sqrt{i}})+1$	[–600, 600]	0
f_{12}	$f_{12}(x)=-20\exp(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D}x_i^2})-\exp(\frac{1}{D}\sum_{i=1}^{D}\cos 2\pi x_i)+20+\mathrm{e}$	[–32, 32]	0
f_{13}	$f_{13}(x)=10D+\sum_{i=1}^{D}(x_i^2-10\cos(2\pi x_i))$	[–0.5, 0.5]	0
f_{14}	$f_{14}(x)=\sum_{i=1}^{D}(\sum_{k=0}^{k_{\max}}[a^k\cos(2\pi b^k(x_i+0.5))])-D\sum_{k=0}^{k_{\max}}[a^k\cos(2\pi b^k\times 0.5)]$ $a=0.5, b=3, k_{\max}=20$	[–0.5, 0.5]	0
f_{15}	$f_{15}(x)=\sum_{i=1}^{D}(0.5+\frac{\sin^2(\sqrt{x_i^2+x_{i+1}^2})-0.5}{[1+0.001(x_i^2+x_{i+1}^2)]^2}), x_{D+1}=x_1$	[–0.5, 0.5]	0
f_{16}	$f_{16}(x)=1-\cos(2\pi\sqrt{\sum_{i=1}^{D}x_i^2})+0.1\sqrt{\sum_{i=1}^{D}x_i^2}$	[–100, 100]	0
f_{17}	$f_{17}(x)=\frac{\pi}{D}\{10\sin^2(\pi y_1)+\sum_{i=1}^{D-1}(y_i-1)^2[1+10\sin^2(\pi y_{i+1})]+(y_D-1)^2\}+\sum_{i=1}^{D}u(x_i,10,100,4)$ $y_i=1+\frac{1}{4}(x_i+1), u(x_i,a,k,m)=\begin{cases}k(x_i-a)^m, x_i>a;\\0, \quad -a<x_i<a;\\k(-x_i-a)^m, x_i<-a\end{cases}$	[–50, 50]	0
f_{18}	$f_{18}(x)=0.1\{10\sin^2(3\pi x_1)+\sum_{i=1}^{D-1}(x_i-1)^2[1+\sin^2(3\pi x_{i+1})]+(x_D-1)^2[1+\sin^2(2\pi x_D)]\}+$ $\sum_{i=1}^{D}u(x_i,5,100,4)$	[–50, 50]	0

Table A2 Comparison of QDMA and different DE variants on 10*D* problems.

Function	QDMA	SaDE	SaNSDE	SaNSDE+	ADDE
	Bev/Mev/Std	Bev/Mev/Std/Test	Bev/Mev/Std/Test	Bev/Mev/Std/Test	Bev/Mev/Std/Test
f_1	$\mathbf{0.00\times10^{0}}$/$1.80\times10^{-26}$/$3.55\times10^{-26}$	$\mathbf{0.00\times10^{0}}$/$1.77\times10^{-80}$/$7.40\times10^{-80}$/≈	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/≈	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/–	1.76×10^{-31}/6.88×10^{-31}/5.34×10^{-31}/+
f_2	$\mathbf{0.00\times10^{0}}$/$1.58\times10^{-24}$/$3.75\times10^{-24}$	6.90×10^{-24}/2.46×10^{-11}/8.88×10^{-11}/+	$\mathbf{0.00\times10^{0}}$/$\mathbf{2.15\times10^{-95}}$/$\mathbf{1.16\times10^{-94}}$/–	1.45×10^{-71}/1.81×10^{-63}/9.30×10^{-63}/–	6.45×10^{2}/1.49×10^{3}/4.06×10^{2}/+
f_3	$\mathbf{0.00\times10^{0}}$/$3.56\times10^{-20}$/$1.59\times10^{-19}$	$\mathbf{0.00\times10^{0}}$/$3.04\times10^{-69}$/$1.45\times10^{-68}$/–	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/–	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/–	1.04×10^{6}/4.45×10^{6}/2.11×10^{6}/+
f_4	$\mathbf{0.00\times10^{0}}$/$1.76\times10^{-2}$/$7.34\times10^{-2}$	4.46×10^{-50}/2.31×10^{-28}/9.72×10^{-28}/–	1.05×10^{-53}/$\mathbf{1.61\times10^{-47}}$/$\mathbf{5.90\times10^{-47}}$/–	4.10×10^{-25}/3.66×10^{-23}/4.91×10^{-23}/–	7.21×10^{0}/1.09×10^{1}/1.31×10^{0}/+
f_5	$\mathbf{0.00\times10^{0}}$/$1.73\times10^{-13}$/$2.86\times10^{-13}$	7.71×10^{-32}/1.19×10^{-17}/4.15×10^{-17}/≈	3.30×10^{-46}/$\mathbf{4.32\times10^{-42}}$/$\mathbf{1.99\times10^{-41}}$/≈	1.76×10^{-29}/2.69×10^{-26}/7.77×10^{-26}/≈	1.55×10^{1}/2.12×10^{1}/2.53×10^{0}/+
f_6	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/≈	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/≈	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/≈	7.09×10^{2}/1.43×10^{3}/4.55×10^{2}/+
f_7	$\mathbf{0.00\times10^{0}}$/$2.15\times10^{-25}$/$3.22\times10^{-25}$	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/–	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/–	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/≈	3.08×10^{3}/6.14×10^{3}/1.60×10^{3}/+
f_8	$\mathbf{0.00\times10^{0}}$/$4.90\times10^{-20}$/$8.67\times10^{-20}$	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/–	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/–	2.37×10^{-93}/1.03×10^{-46}/1.06×10^{-46}/–	5.66×10^{8}/1.00×10^{9}/1.84×10^{8}/+
f_9	$\mathbf{9.48\times10^{-6}}$/$9.87\times10^{-3}$/$1.66\times10^{-2}$	2.07×10^{-3}/6.41×10^{-3}/$\mathbf{2.09\times10^{-3}}$/≈	2.72×10^{-3}/$\mathbf{6.37\times10^{-3}}$/$2.11\times10^{-3}$/≈	1.74×10^{-3}/7.06×10^{-3}/3.06×10^{-3}/≈	4.63×10^{-3}/7.05×10^{-2}/4.89×10^{-2}/+
f_{10}	1.59×10^{-28}/3.99×10^{-1}/1.20×10^{0}	9.47×10^{-13}/1.23×10^{-8}/3.12×10^{-8}/+	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/–	9.87×10^{-22}/2.41×10^{-4}/1.09×10^{-3}/+	3.19×10^{6}/2.19×10^{7}/1.14×10^{7}/+
f_{11}	$\mathbf{0.00\times10^{0}}$/$6.40\times10^{-2}$/$1.30\times10^{-1}$	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/–	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/–	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/–	8.24×10^{0}/1.29×10^{1}/2.82×10^{0}/+
f_{12}	$\mathbf{4.44\times10^{-16}}$/$1.44\times10^{-1}$/$4.51\times10^{-1}$	4.44×10^{-16}/$\mathbf{3.76\times10^{-15}}$/$8.86\times10^{-16}$/–	4.00×10^{-15}/4.00×10^{-15}/$\mathbf{0.00\times10^{0}}$/–	4.00×10^{-15}/9.40×10^{-11}/2.91×10^{-10}/≈	7.98×10^{0}/1.22×10^{1}/1.64×10^{0}/+
f_{13}	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/≈	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/≈	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/≈	4.67×10^{0}/7.46×10^{0}/1.91×10^{0}/+
f_{14}	$\mathbf{0.00\times10^{0}}$/$\mathbf{5.42\times10^{-1}}$/$8.19\times10^{-1}$	1.48×10^{0}/2.96×10^{0}/$\mathbf{6.29\times10^{-1}}$/+	2.50×10^{0}/4.68×10^{0}/6.69×10^{-1}/+	5.17×10^{0}/8.30×10^{0}/9.89×10^{-1}/+	7.15×10^{0}/9.14×10^{0}/7.21×10^{-1}/+
f_{15}	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/≈	$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/$\mathbf{0.00\times10^{0}}$/≈	$\mathbf{0.00\times10^{0}}$/$1.68\times10^{-13}$/$9.07\times10^{-13}$/≈	2.51×10^{-2}/5.51×10^{-2}/1.60×10^{-2}/+
f_{16}	$\mathbf{4.25\times10^{-97}}$/$1.67\times10^{-1}$/$2.57\times10^{-1}$	9.99×10^{-2}/$\mathbf{9.99\times10^{-2}}$/$1.93\times10^{-11}$/≈	9.99×10^{-2}/$\mathbf{9.99\times10^{-2}}$/$7.09\times10^{-13}$/≈	9.99×10^{-2}/$\mathbf{9.99\times10^{-2}}$/$\mathbf{7.33\times10^{-12}}$/≈	2.64×10^{0}/3.92×10^{0}/4.94×10^{-1}/+
f_{17}	1.01×10^{-31}/6.02×10^{-1}/2.16×10^{0}	$\mathbf{4.71\times10^{-32}}$/$\mathbf{4.71\times10^{-32}}$/$\mathbf{1.09\times10^{-47}}$/–	$\mathbf{4.71\times10^{-32}}$/$\mathbf{4.71\times10^{-32}}$/$\mathbf{1.09\times10^{-47}}$/–	1.05×10^{-30}/1.65×10^{-29}/1.69×10^{-29}/–	9.37×10^{0}/1.27×10^{2}/2.69×10^{2}/+
f_{18}	3.35×10^{-30}/1.30×10^{-1}/4.42×10^{-1}	$\mathbf{1.35\times10^{-31}}$/$\mathbf{1.35\times10^{-31}}$/$\mathbf{6.57\times10^{-47}}$/–	$\mathbf{1.35\times10^{-31}}$/$\mathbf{1.35\times10^{-31}}$/$\mathbf{6.57\times10^{-47}}$/–	1.76×10^{-31}/6.88×10^{-31}/5.34×10^{-31}/–	1.16×10^{2}/1.19×10^{5}/1.44×10^{5}/+
Statistic test (+/≈/–)	–	3/7/8	1/7/10	2/8/8	18/0/0

Table A3 Comparison of QDMA and different DE variants on 30*D* problems.

Function	QDMA	SaDE	SaNSDE	SaNSDE+	ADDE
	Bev/Mev/Std	Bev/Mev/Std/Test	Bev/Mev/Std/Test	Bev/Mev/Std/Test/	Bev/Mev/Std/Test
f_1	$\mathbf{0.00\times10^{0}}/1.95\times10^{-22}/6.02\times10^{-22}$	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/–	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/–	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/–	$7.64\times10^{-12}/3.22\times10^{-11}/1.92\times10^{-11}$/+
f_2	$\mathbf{2.58\times10^{-88}/2.02\times10^{0}/9.26\times10^{0}}$	$8.19\times10^{2}/2.75\times10^{3}/1.03\times10^{3}$/+	$6.35\times10^{-2}/2.18\times10^{3}/4.94\times10^{3}$/–	$1.51\times10^{1}/2.65\times10^{3}/5.85\times10^{3}$/+	$1.52\times10^{4}/2.26\times10^{4}/3.62\times10^{3}$/+
f_3	$\mathbf{0.00\times10^{0}}/1.10\times10^{4}/3.88\times10^{4}$	$3.37\times10^{-92}/3.41\times10^{-90}/5.97\times10^{-90}$/–	$1.55\times10^{-95}/\mathbf{1.55\times10^{-92}/2.21\times10^{-92}}$/–	$3.63\times10^{-61}/8.42\times10^{-60}/1.25\times10^{-59}$/–	$9.59\times10^{7}/1.56\times10^{8}/2.84\times10^{7}$+
f_4	$\mathbf{2.41\times10^{-59}}/9.04\times10^{0}/1.52\times10^{1}$	$1.09\times10^{-27}/5.08\times10^{-27}/2.55\times10^{-27}$/–	$8.53\times10^{-28}/\mathbf{2.08\times10^{-27}/8.90\times10^{-28}}$/–	$9.30\times10^{-10}/2.07\times10^{-9}/8.27\times10^{-10}$/–	$5.76\times10^{1}/7.99\times10^{1}/2.39\times10^{1}$/+
f_5	$\mathbf{4.95\times10^{-44}}/2.00\times10^{0}/1.08\times10^{1}$	$1.28\times10^{-10}/\mathbf{3.53\times10^{-10}/1.28\times10^{-10}}$/+	$1.79\times10^{-3}/2.37\times10^{-1}/4.15\times10^{-1}$/+	$8.62\times10^{-5}/1.35\times10^{-1}/2.22\times10^{-1}$/+	$4.07\times10^{1}/4.67\times10^{1}/2.81\times10^{0}$/+
f_6	$\mathbf{0.00\times10^{0}}/3.33\times10^{-2}/1.80\times10^{-1}$	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/≈	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/≈	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/≈	$1.19\times10^{4}/1.60\times10^{4}/1.68\times10^{3}$/+
f_7	$\mathbf{0.00\times10^{0}}/4.39\times10^{-85}/3.08\times10^{-85}$	$9.14\times10^{-95}/1.25\times10^{-92}/2.78\times10^{-92}$/–	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/–	$3.81\times10^{-91}/4.58\times10^{-84}/1.45\times10^{-83}$/≈	$1.33\times10^{5}/1.93\times10^{5}/2.40\times10^{4}$/+
f_8	$1.83\times10^{-92}/3.97\times10^{-14}/2.14\times10^{-13}$	$1.05\times10^{-95}/1.67\times10^{-93}/3.36\times10^{-93}$/–	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/–	$1.39\times10^{-27}/2.17\times10^{-25}/2.64\times10^{-25}$/+	$8.54\times10^{9}/1.35\times10^{10}/2.12\times10^{9}$/+
f_9	$\mathbf{1.63\times10^{-6}/5.01\times10^{-4}/7.13\times10^{-4}}$	$4.90\times10^{-2}/6.18\times10^{-2}/9.74\times10^{-3}$/+	$5.57\times10^{-2}/8.22\times10^{-2}/1.23\times10^{-2}$/+	$4.50\times10^{-2}/8.66\times10^{-2}/1.52\times10^{-2}$/+	$9.66\times10^{-1}/5.41\times10^{0}/1.79\times10^{0}$/+
f_{10}	$\mathbf{1.92\times10^{-9}/1.95\times10^{1}/1.32\times10^{1}}$	$5.08\times10^{0}/2.05\times10^{1}/2.57\times10^{1}$/≈	$2.27\times10^{-3}/1.17\times10^{0}/1.88\times10^{0}$/–	$2.41\times10^{1}/3.78\times10^{1}/2.53\times10^{1}$/+	$9.27\times10^{8}/1.83\times10^{9}/4.85\times10^{8}$/+
f_{11}	$\mathbf{0.00\times10^{0}}/5.56\times10^{-3}/1.87\times10^{-2}$	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/≈	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/≈	$\mathbf{0.00\times10^{0}}/8.50\times10^{-9}/1.18\times10^{-9}$/≈	$1.07\times10^{2}/1.47\times10^{2}/1.75\times10^{1}$/+
f_{12}	$\mathbf{4.44\times10^{-16}}/3.11\times10^{-11}/1.55\times10^{-10}$	$4.00\times10^{-15}/\mathbf{6.01\times10^{-15}}/1.76\times10^{-15}$/+	$4.00\times10^{-15}/6.96\times10^{-15}/\mathbf{1.32\times10^{-15}}$/+	$2.02\times10^{-8}/1.08\times10^{-5}/2.86\times10^{-5}$/+	$1.62\times10^{1}/1.72\times10^{1}/4.61\times10^{-1}$/+
f_{13}	$\mathbf{0.00\times10^{0}}/3.79\times10^{-15}/2.04\times10^{-14}$	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/≈	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/≈	$6.22\times10^{-6}/2.84\times10^{-3}/6.73\times10^{-3}$/+	$6.46\times10^{1}/8.16\times10^{1}/6.38\times10^{0}$/+
f_{14}	$\mathbf{0.00\times10^{0}/6.33\times10^{-4}/3.41\times10^{-3}}$	$2.10\times10^{1}/2.52\times10^{1}/1.68\times10^{0}$/+	$2.50\times10^{1}/2.82\times10^{1}/1.52\times10^{0}$/+	$3.41\times10^{1}/3.90\times10^{1}/1.87\times10^{0}$/+	$3.46\times10^{1}/3.81\times10^{1}/1.64\times10^{0}$/+
f_{15}	$\mathbf{0.00\times10^{0}}/4.44\times10^{-17}/2.39\times10^{-16}$	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/≈	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/≈	$1.39\times10^{-14}/9.32\times10^{-14}/9.34\times10^{-14}$/+	$5.29\times10^{-1}/8.12\times10^{-1}/1.14\times10^{-1}$/+
f_{16}	$\mathbf{3.53\times10^{-54}/1.47\times10^{-44}/7.80\times10^{-45}}$	$9.99\times10^{-2}/1.77\times10^{-1}/4.20\times10^{-2}$/+	$9.99\times10^{-2}/1.67\times10^{-1}/4.71\times10^{-2}$/+	$9.99\times10^{-2}/1.67\times10^{-1}/4.71\times10^{-2}$/+	$1.02\times10^{1}/1.25\times10^{1}/9.00\times10^{-1}$/+
f_{17}	$6.28\times10^{-29}/6.42\times10^{-11}/3.39\times10^{-10}$	$\mathbf{1.57\times10^{-32}/1.57\times10^{-32}/2.74\times10^{-48}}$/–	$6.49\times10^{-31}/1.22\times10^{-29}/1.02\times10^{-29}$/–	$3.64\times10^{-11}/1.39\times10^{-10}/1.00\times10^{-10}$/+	$4.77\times10^{6}/1.47\times10^{7}/7.47\times10^{6}$/+
f_{18}	$2.08\times10^{-27}/5.97\times10^{-2}/1.04\times10^{-1}$	$\mathbf{1.35\times10^{-31}/1.35\times10^{-31}/6.57\times10^{-47}}$/–	$1.35\times10^{-31}/1.72\times10^{-31}/5.35\times10^{-32}$/–	$7.64\times10^{-12}/3.22\times10^{-11}/1.92\times10^{-11}$/–	$1.17\times10^{7}/5.07\times10^{7}/2.26\times10^{7}$/+
Statistic test (+/≈/−)	−	6/5/7	5/4/9	11/3/4	18/0/0

Table A4 Comparison of QDMA and different DE variants on 50*D* problems.

Function	QDMA	SaDE	SaNSDE	SaNSDE+	ADDE
	Bev/Mev/Std	Bev/Mev/Std/Test	Bev/Mev/Std/Test	Bev/Mev/Std/Test	Bev/Mev/Std/Test
f_1	$4.19\times10^{-91}/5.84\times10^{-15}/\mathbf{0.00\times10^{0}}$	$8.97\times10^{-97}/2.76\times10^{-95}/3.45\times10^{-95}$/–	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/–	$5.62\times10^{-87}/1.92\times10^{-84}/6.68\times10^{-84}$/+	$7.64\times10^{-12}/3.22\times10^{-11}/1.92\times10^{-11}$/+
f_2	$\mathbf{0.00\times10^{0}/3.32\times10^{3}}/1.79\times10^{4}$	$6.04\times10^{4}/8.31\times10^{4}/\mathbf{1.18\times10^{4}}$/+	$2.74\times10^{2}/1.64\times10^{4}/2.59\times10^{4}$+	$1.31\times10^{3}/3.81\times10^{4}/3.85\times10^{4}$/+	$1.52\times10^{4}/2.26\times10^{4}/3.62\times10^{3}$/+
f_3	$\mathbf{3.46\times10^{-99}}/1.72\times10^{5}/9.24\times10^{5}$	$9.32\times10^{-71}/5.91\times10^{-69}/8.57\times10^{-69}$/+	$1.80\times10^{-74}/\mathbf{3.61\times10^{-72}/7.29\times10^{-72}}$/+	$8.61\times10^{-41}/1.39\times10^{-39}/1.81\times10^{-39}$+	$9.59\times10^{7}/1.56\times10^{8}/2.84\times10^{7}$/+
f_4	$\mathbf{1.27\times10^{-43}}/5.12\times10^{1}/4.35\times10^{1}$	$8.15\times10^{-19}/1.73\times10^{-18}/5.23\times10^{-19}$/–	$9.17\times10^{-21}/\mathbf{3.08\times10^{-19}/1.82\times10^{-19}}$/–	$7.28\times10^{-6}/1.48\times10^{-5}/4.98\times10^{-6}$/–	$5.76\times10^{1}/7.99\times10^{1}/2.39\times10^{1}$/+
f_5	$\mathbf{5.96\times10^{-44}/6.82\times10^{-44}/2.18\times10^{-45}}$	$1.07\times10^{-4}/1.08\times10^{-1}/1.88\times10^{-1}$/+	$3.94\times10^{-1}/7.53\times10^{0}/3.37\times10^{0}$/+	$7.19\times10^{-1}/6.55\times10^{0}/3.40\times10^{0}$/+	$4.07\times10^{1}/4.67\times10^{1}/2.81\times10^{0}$/+
f_6	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/00.00\times10^{0}}$/≈	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/≈	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/≈	$1.19\times10^{4}/1.60\times10^{4}/1.68\times10^{3}$/+
f_7	$\mathbf{0.00\times10^{0}}/7.42\times10^{-21}/4.00\times10^{-20}$	$1.54\times10^{-74}/2.15\times10^{-73}/2.77\times10^{-73}$/+	$1.34\times10^{-87}/\mathbf{8.63\times10^{-84}/1.16\times10^{-83}}$/+	$6.91\times10^{-64}/8.67\times10^{-60}/3.63\times10^{-59}$/+	$1.33\times10^{5}/1.93\times10^{5}/2.40\times10^{4}$/+
f_8	$\mathbf{0.00\times10^{0}}/3.20\times10^{0}/1.46\times10^{1}$	$4.04\times10^{-76}/6.83\times10^{-75}/7.43\times10^{-75}$/+	$5.30\times10^{-90}/\mathbf{2.64\times10^{-85}/9.37\times10^{-85}}$/–	$9.76\times10^{-42}/8.34\times10^{-16}/1.18\times10^{-15}$/+	$8.54\times10^{9}/1.35\times10^{10}/2.12\times10^{9}$/+
f_9	$\mathbf{1.28\times10^{-6}/6.28\times10^{-4}/9.90\times10^{-4}}$	$1.00\times10^{-1}/1.32\times10^{-1}/1.41\times10^{-2}$/+	$1.34\times10^{-1}/1.79\times10^{-1}/2.72\times10^{-2}$/+	$1.33\times10^{-1}/1.96\times10^{-1}/2.47\times10^{-2}$/+	$9.66\times10^{-1}/5.41\times10^{0}/1.79\times10^{0}$/+
f_{10}	$\mathbf{4.49\times10^{-6}}/3.96\times10^{1}/\mathbf{1.11\times10^{1}}$	$2.68\times10^{1}/\mathbf{3.86\times10^{1}}/2.20\times10^{1}$/≈	$2.70\times10^{1}/5.18\times10^{1}/2.89\times10^{1}$/+	$4.27\times10^{1}/1.14\times10^{2}/8.85\times10^{1}$/+	$9.27\times10^{8}/1.83\times10^{9}/4.85\times10^{8}$/+
f_{11}	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/≈	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/≈	$1.80\times10^{-7}/4.16\times10^{-5}/9.19\times10^{-5}$/+	$1.07\times10^{2}/1.47\times10^{2}/1.75\times10^{1}$/+
f_{12}	$\mathbf{4.44\times10^{-16}}/4.44\times10^{-16}/\mathbf{0.00\times10^{0}}$	$7.55\times10^{-15}/\mathbf{2.73\times10^{-16}}/1.76\times10^{-15}$/+	$7.55\times10^{-15}/8.26\times10^{-15}/1.69\times10^{-15}$/+	$3.12\times10^{-5}/4.95\times10^{-4}/5.90\times10^{-4}$/+	$1.62\times10^{1}/1.72\times10^{1}/4.61\times10^{-1}$/+
f_{13}	$\mathbf{0.00\times10^{0}}/1.14\times10^{-14}/6.12\times10^{-14}$	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/≈	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/≈	$4.89\times10^{-1}/6.49\times10^{0}/4.97\times10^{0}$/+	$6.46\times10^{1}/8.16\times10^{1}/6.38\times10^{0}$/+
f_{14}	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$4.55\times10^{1}/5.35\times10^{1}/2.76\times10^{0}$/+	$5.35\times10^{1}/5.69\times10^{1}/1.85\times10^{0}$/+	$6.80\times10^{1}/7.26\times10^{1}/1.95\times10^{0}$/+	$3.46\times10^{1}/3.81\times10^{1}/1.64\times10^{0}$/+
f_{15}	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/00.00\times10^{0}}$/≈	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$/≈	$1.93\times10^{-9}/7.18\times10^{-9}/6.26\times10^{-9}$/+	$5.29\times10^{-1}/8.12\times10^{-1}/1.14\times10^{-1}$+
f_{16}	$\mathbf{2.04\times10^{-44}/2.56\times10^{-44}/2.52\times10^{-45}}$	$2.00\times10^{-1}/2.00\times10^{-1}/7.72\times10^{-12}$/+	$2.00\times10^{-1}/2.20\times10^{-1}/4.00\times10^{-2}$/+	$2.00\times10^{-1}/2.03\times10^{-1}/1.80\times10^{-2}$/+	$1.02\times10^{1}/1.25\times10^{1}/9.00\times10^{-1}$/+
f_{17}	$2.32\times10^{-27}/1.84\times10^{-3}/6.62\times10^{-3}$	$\mathbf{9.42\times10^{-33}/9.42\times10^{-33}/2.74\times10^{-48}}$/+	$2.39\times10^{-21}/7.38\times^{-19}/6.76\times^{-19}$/–	$1.11\times10^{-5}/3.64\times10^{-5}/2.34\times10^{-5}$/+	$4.77\times10^{6}/1.47\times10^{7}/7.47\times10^{6}$/+
f_{18}	$7.54\times10^{-25}/7.79\times10^{-2}/2.81\times10^{-1}$	$\mathbf{1.35\times10^{-31}/1.35\times10^{-31}/6.57\times10^{-47}}$/+	$3.41\times10^{-21}/1.32\times10^{-20}/9.92\times10^{-21}$/–	$8.88\times10^{-6}/2.87\times10^{-5}/1.64\times10^{-5}$/–	$1.17\times10^{7}/5.07\times10^{7}/2.26\times10^{7}$/+
Statistic test (+/≈/−)	−	11/5/2	9/4/5	15/1/2	18/0/0

Table A5 Comparison of QDMA and different DE variants on 100*D* problems.

Function	QDMA	SaDE	SaNSDE	SaNSDE+	ADDE
	Bev/Mev/Std	Bev/Mev/Std/Test	Bev/Mev/Std/Test	Bev/Mev/Std/Test	Bev/Mev/Std/Test
f_1	$\mathbf{1.18\times10^{-85}/1.46\times10^{-85}/1.94\times10^{-86}}$	$7.19\times10^{-73}/2.08\times10^{-65}/4.38\times10^{-65}/+$	$4.29\times10^{-77}/3.36\times10^{-68}/1.42\times10^{-67}/+$	$3.48\times10^{-54}/3.71\times10^{-51}/8.27\times10^{-51}/+$	$2.51\times10^{0}/4.66\times10^{1}/2.07\times10^{2}/+$
f_2	$\mathbf{4.05\times10^{-86}/3.65\times10^{4}/9.37\times10^{4}}$	$3.83\times10^{5}/5.08\times10^{5}/6.66\times10^{4}/+$	$9.32\times10^{3}/1.53\times10^{5}/1.27\times10^{5}/+$	$2.56\times10^{4}/3.54\times10^{5}/2.19\times10^{5}/+$	$2.52\times10^{5}/3.26\times10^{5}/4.28\times10^{4}/+$
f_3	$\mathbf{6.12\times10^{-81}/1.10\times10^{-80}/3.30\times10^{-81}}$	$4.96\times10^{-50}/3.42\times10^{-43}/7.12\times10^{-43}/+$	$1.03\times10^{-49}/4.19\times10^{-45}/1.62\times10^{-44}/+$	$5.88\times10^{-21}/2.74\times10^{-19}/3.61\times10^{-19}/+$	$2.09\times10^{9}/2.78\times10^{9}/3.11\times10^{8}/+$
f_4	$\mathbf{2.83\times10^{-43}}/7.68\times10^{4}/3.15\times10^{5}$	$4.15\times10^{-12}/\mathbf{1.42\times10^{-10}}/2.16\times10^{-10}/–$	$5.49\times10^{-12}/2.21\times10^{-10}/\mathbf{1.99\times10^{-10}}/–$	$9.00\times10^{-1}/6.09\times10^{0}/3.50\times10^{0}/–$	$1.35\times10^{7}/5.35\times10^{21}/1.70\times10^{22}/+$
f_5	$\mathbf{6.48\times10^{-44}/6.91\times10^{-44}/1.24\times10^{-45}}$	$8.71\times10^{0}/2.40\times10^{1}/7.42\times10^{0}/+$	$2.01\times10^{1}/3.74\times10^{1}/8.51\times10^{0}/+$	$1.94\times10^{1}/3.58\times10^{1}/7.64\times10^{0}/+$	$6.18\times10^{1}/6.79\times10^{1}/1.91\times10^{0}/+$
f_6	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}/\approx}$	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}/\approx}$	$\mathbf{0.00\times10^{0}}/2.20\times10^{1}/5.76\times10^{1}/+$	$7.95\times10^{4}/9.65\times10^{4}/9.08\times10^{3}/+$
f_7	$\mathbf{7.15\times10^{-86}/7.33\times10^{-84}/1.75\times10^{-84}}$	$1.32\times10^{-48}/9.80\times10^{-48}/1.36\times10^{-47}/+$	$2.66\times10^{-61}/5.06\times10^{-57}/2.44\times10^{-56}/+$	$5.32\times10^{-37}/1.07\times10^{-33}/5.30\times10^{-33}/+$	$3.40\times10^{6}/4.17\times10^{6}/3.31\times10^{5}/+$
f_8	$\mathbf{1.12\times10^{-79}/1.45\times10^{-79}/2.00\times10^{-80}}$	$1.36\times10^{-49}/1.69\times10^{-48}/2.89\times10^{-48}/+$	$8.80\times10^{-64}/1.51\times10^{-59}/4.20\times10^{-59}/+$	$1.26\times10^{-28}/4.80\times10^{-6}/1.27\times10^{-5}/+$	$7.04\times10^{10}/8.43\times10^{10}/6.47\times10^{9}/+$
f_9	$\mathbf{1.56\times10^{-6}/6.75\times10^{-4}/1.07\times10^{-3}}$	$2.39\times10^{-1}/3.20\times10^{-1}/2.86\times10^{-2}/+$	$3.27\times10^{-1}/4.26\times10^{-1}/6.56\times10^{-2}+$	$3.62\times10^{-1}/4.62\times10^{-1}/5.87\times10^{-2}/+$	$1.26\times10^{2}/2.13\times10^{2}/4.30\times10^{1}/+$
f_{10}	$\mathbf{6.61\times10^{-2}/8.54\times10^{1}/2.79\times10^{1}}$	$8.43\times10^{1}/1.18\times10^{2}/3.03\times10^{1}/+$	$8.57\times10^{1}/1.29\times10^{2}/4.36\times10^{1}/+$	$9.79\times10^{1}/1.95\times10^{3}/5.01\times10^{3}/+$	$1.38\times10^{10}/2.23\times10^{10}/3.28\times10^{9}/+$
f_{11}	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$\mathbf{0.00\times10^{0}}/5.31\times10^{-15}/3.10\times10^{-15}/+$	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}/\approx}$	$2.33\times10^{-2}/2.42\times10^{-1}/1.89\times10^{-1}/+$	$6.78\times10^{2}/8.57\times10^{2}/8.63\times10^{1}/+$
f_{12}	$\mathbf{4.44\times10^{-16}/5.63\times10^{-16}/6.38\times10^{-16}}$	$6.21\times10^{-10}/1.48\times10^{-9}/4.24\times10^{-10}/+$	$7.61\times10^{-13}/1.26\times10^{-11}/7.53\times10^{-12}/+$	$1.12\times10^{-3}/5.49\times10^{-2}/3.95\times10^{-2}/+$	$1.84\times10^{1}/1.89\times10^{1}/2.23\times10^{-1}/+$
f_{13}	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$8.98\times10^{-12}/3.65\times10^{-1}/7.03\times10^{-1}/+$	$\mathbf{0.00\times10^{0}}/1.99\times10^{-1}/3.98\times10^{-1}/+$	$6.30\times10^{1}/9.09\times10^{1}/1.35\times10^{1}/+$	$3.04\times10^{2}/4.04\times10^{2}/3.94\times10^{1}/+$
f_{14}	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$1.19\times10^{2}/1.32\times10^{2}/4.72\times10^{0}/+$	$1.31\times10^{2}/1.37\times10^{2}/2.77\times10^{0}/+$	$1.54\times10^{2}/1.62\times10^{2}/3.30\times10^{0}/+$	$1.35\times10^{2}/1.43\times10^{2}/3.13\times10^{0}/+$
f_{15}	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}}$	$\mathbf{0.00\times10^{0}/0.00\times10^{0}/0.00\times10^{0}/\approx}$	$2.22\times10^{-16}/1.43\times10^{-14}/1.08\times10^{-14}/+$	$5.57\times10^{-5}/9.08\times10^{-5}/2.02\times10^{-5}/+$	$3.77\times10^{0}/4.97\times10^{0}/5.64\times10^{-1}/+$
f_{16}	$\mathbf{3.34\times10^{-44}/3.79\times10^{-44}/2.55\times10^{-45}}$	$3.00\times10^{-1}/3.23\times10^{-1}/4.23\times10^{-2}/+$	$3.00\times10^{-1}/4.43\times10^{-1}/1.17\times10^{-1}/+$	$3.00\times10^{-1}/3.74\times10^{-1}/9.42\times10^{-2}/+$	$2.56\times10^{1}/2.96\times10^{1}/1.54\times10^{0}/+$
f_{17}	$\mathbf{2.13\times10^{-22}}/1.24\times10^{-2}/3.01\times10^{-2}$	$3.77\times10^{-20}/\mathbf{2.96\times10^{-17}}/\mathbf{1.50\times10^{-16}}/–$	$4.81\times10^{-10}/9.3\times10^{-3}/3.52\times10^{-2}/–$	$1.53\times10^{0}/4.96\times10^{0}/2.01\times10^{0}/+$	$2.11\times10^{8}/3.32\times10^{8}/5.97\times10^{7}/+$
f_{18}	$6.38\times10^{-12}/3.03\times10^{-1}/6.34\times10^{-1}$	$\mathbf{8.74\times10^{-23}/2.58\times10^{-19}/1.33\times10^{-18}}/–$	$7.01\times10^{-11}/8.90\times10^{-2}/4.79\times10^{-1}/–$	$2.51\times10^{0}/4.66\times10^{1}/2.07\times10^{2}/+$	$2.25\times10^{8}/7.90\times10^{8}/1.72\times10^{8}/+$
Statistic test (+/≈/−)	−	13/2/3	13/2/3	17/0/1	18/0/0

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China (No. 62273193) and the Talent Introducing Project of Hebei Agricultural University (Nos. KY201903 and YJ201953).

References

[1] T. C. Lu, G. R. Yu, and J. C. Juang, Quantum-based algorithm for optimizing artificial neural networks, *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 8, pp. 1266–1278, 2013.

[2] Y. Y. Li, M. Z. Tian, G. Y. Liu, C. Peng, and L. C. Jiao, Quantum optimization and quantum learning: A survey, *IEEE Access*, vol. 8, pp. 23568–23593, 2020.

[3] C. -H. Cho, C. -Y. Chen, K. -C. Chen, T. -W. Huang, M. -C. Hsu, N. -P. Cao, B. Zeng, S. -G. Tan, and C. -R. Chang, Quantum computation: Algorithms and applications, *Chinese Journal of Physics*, vol. 72, pp. 248–269, 2021.

[4] K. -H. Han and J. -H. Kim, Quantum-inspired evolutionary algorithm with a new termination criterion, H/sub/spl epsi// gate, and two-phase scheme, *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 2, pp. 156–169, 2004.

[5] A. Narayanan and M. Moore, Quantum-inspired genetic algorithms, in *Proc. IEEE International Conference on Evolutionary Computation,* Nagoya, Japan, 1996, pp. 61–66.

[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] F. D. Martino and S. Sessa, A novel quantum inspired genetic algorithm to initialize cluster centers in fuzzy C-means, *Expert Systems with Applications*, vol. 191, p. 116340, 2022.

[8] W. Deng, H. Liu, J. Xu, H. Zhao, and Y. Song, An improved quantum-inspired differential evolution algorithm for deep belief network, *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 10, pp. 7319–7327, 2020.

[9] W. Deng, J. Xu, X. Z. Gao, and H. Zhao, An enhanced MSIQDE algorithm with novel multiple strategies for global optimization problems, *IEEE Transactions on Systems*, *Man*, *and Cybernetics*: *Systems*, vol. 52, no. 3, pp. 1578–1587, 2022.

[10] Y. Wang and W. Wang, Quantum-inspired differential evolution with grey wolf optimizer for 0-1 knapsack problem, *Mathematics*, vol. 9, no. 11, p. 1233, 2021.

[11] W. Deng, S. F. Shang, X. Cai, H. Zhao, Y. Zhou, H. Chen, and W. Deng, Quantum differential evolution with cooperative coevolution framework and hybrid mutation strategy for large scale optimization, *Knowledge-Based Systems*, vol. 224, p. 107080, 2021.

[12] H. Su and Y. Yang, Differential evolution and quantum-

inquired differential evolution for evolving Takagi–Sugeno fuzzy models, *Expert Systems with Applications*, vol. 38, no. 6, pp. 6447–6451, 2011.

[13] W. Deng, J. Xu, H. Zhao, and Y. Song, A novel gate resource allocation method using improved PSO-based QEA, *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 3, pp. 1737–1745, 2022.

[14] R. Shang, B. Du, K. Dai, L. Jiao, A. M. G. Esfahani, and R. Stolkin, Quantum-inspired immune clonal algorithm for solving large-scale capacitated arc routing problems, *Memetic Computing*, vol. 10, no. 1, pp. 81–102, 2018.

[15] X. Cai, H. Zhao, S. Shang, Y. Zhou, W. Deng, H. Chen, and W. Deng, An improved quantum-inspired cooperative co-evolution algorithm with muli-strategy and its application, *Expert Systems with Applications*, vol. 171, p. 114629, 2021.

[16] M. D. Platel, S. Schliebs, and N. Kasabov, Quantum-inspired evolutionary algorithm: A multimodel EDA, *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 6, pp. 1218–1232, 2009.

[17] X. F. Liu, Z. H. Zhan, T. L. Gu, S. Kwong, Z. Lu, H. B. -L. Duh, and J. Zhang, Neural network-based information transfer for dynamic optimization, *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 5, pp. 1557–1570, 2020.

[18] Z. H. Zhan, Z. J. Wang, H. Jin, and J. Zhang, Adaptive distributed differential evolution, *IEEE Transactions on Cybernetics*, vol. 50, no. 11, pp. 4633–4647, 2020.

[19] N. M. Hamza, D. L. Essam, and R. A. Sarker, Constraint consensus mutation-based differential evolution for constrained optimization, *IEEE Transactions on Evolutionary Computation*, vol. 20, no. 3, pp. 447–459, 2016.

[20] Y. -J. Gong, W. -N. Chen, Z. -H. Zhan, J. Zhang, Y. Li, Q. Zhang, and J. -J. Li, Distributed evolutionary algorithms and their models: A survey of the state-of-the-art, *Applied Soft Computing*, vol. 34, pp. 286–300, 2015.

[21] F. Neri and C. Cotta, Memetic algorithms and memetic computing optimization: A literature review, *Swarm and Evolutionary Computation*, vol. 2, pp. 1–14, 2012.

[22] J. Yang, Y. Cai, D. Tang, W. Chen, and L. Hu, Memetic quantum optimization algorithm with levy flight for high dimension function optimization, *Applied Intelligence*, doi: 10.1007/s10489-022-03429-z.

[23] D. Tang, Z. Liu, J. Zhao, S. Dong, and Y. Cai, Memetic quantum evolution algorithm for global optimization, *Neural Computing and Applications*, vol. 32, pp. 9299–9329, 2020.

[24] S. Das and P. N. Suganthan, Differential evolution: A survey of the state-of-the-art, *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 4–31, 2011.

[25] K. -H. Han and J. -H. Kim, Quantum-inspired evolutionary algorithm for a class of combinatorial optimization, *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 6, pp. 580–593, 2002.

[26] Z. J. Wang, Z. H. Zhan, S. Kwong, H. Jin, and J. Zhang, Adaptive granularity learning distributed particle swarm optimization for large-scale optimization, *IEEE Transactions on Cybernetics*, vol. 51, no. 3, pp. 1175–1188, 2021.

[27] Z. J. Wang, Z. -H. Zhan, W. -J. Yu, Y. Lin, J. Zhang, T. -L. Gu, and J. Zhang, Dynamic group learning distributed particle swarm optimization for large-scale optimization and its application in cloud workflow scheduling, *IEEE Transactions on Cybernetics*, vol. 50, no. 6, pp. 2715–2729, 2020.

[28] Y. Guo, J. -Y. Li, and Z. -H. Zhan, Efficient hyperparameter optimization for convolution neural networks in deep learning: A distributed particle swarm optimization approach, *Cybernetics and Systems*, vol. 52, no. 1, pp. 36–57, 2021.

[29] Z. J. Wang, Y. R. Zhou, and J. Zhang, Adaptive estimation distribution distributed differential evolution for multimodal optimization problems, *IEEE Transactions on Cybernetics*, vol. 52, no. 7, pp. 6059–6070, 2022.

[30] Z. J. Wang, Z. H. Zhan, and J. Zhang, Solving the energy efficient coverage problem in wireless sensor networks: A distributed genetic algorithm approach with hierarchical fitness evaluation, *Energies*, vol. 11, no. 12, p. 3526, 2018.

[31] Y. F. Ge, W. J. Yu, J. Cao, H. Wang, Z. -H. Zhan, Y. Zhang, and J. Zhang, Distributed memetic algorithm for outsourced database fragmentation, *IEEE Transactions on Cybernetics*, vol. 51, no. 10, pp. 4808–4821, 2021.

[32] Y. F. Ge, W. J. Yu, Y. Lin, Y. J. Gong, Z. H. Zhan, W. N. Chen, and J. Zhang, Distributed differential evolution based on adaptive mergence and split for large-scale optimization, *IEEE Transactions on Cybernetics*, vol. 48, no. 7, pp. 2166–2180, 2018.

[33] G. H. Zhang, X. J. Ma, L. Wang, and K. Y. Xing, Elite archive-assisted adaptive memetic algorithm for a realistic hybrid differentiation flowshop scheduling problem, *IEEE Transactions on Evolutionary Computation*, vol. 26, no. 1, pp. 100–114, 2022.

[34] C. M. Zhang, J. Chen, and B. Xin, Distributed memetic differential evolution with the synergy of Lamarckian and Baldwinian learning, *Applied Soft Computing*, vol. 13, no. 5, pp. 2947–2959, 2013.

[35] X. Zhang, Z. H. Zhan, W. Fang, P. Qian, and J. Zhang, Multi population ant colony system with knowledge-based local searches for multiobjective supply chain configuration, *IEEE Transactions on Evolutionary Computation*, vol. 26, no. 3, pp. 512–526, 2021.

[36] N. R. Sabar, J. Abawajy, and J. Yearwood, Heterogeneous cooperative co-evolution memetic differential evolution algorithm for big data optimization problems, *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 2, pp. 315–327, 2017.

[37] G. H. Zhang, B. Liu, L. Wang, D. X. Yu, and K. Y. Xing, Distributed co-evolutionary memetic algorithm for distributed hybrid differentiation flowshop scheduling problem, *IEEE Transactions on Evolutionary Computation*, vol. 26, no. 5, pp. 1043–1057, 2022.

[38] X. Chen, Y. S. Ong, M. H. Lim, and K. C. Tan, A multi-facet survey on memetic computation, *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 5, pp. 591–607, 2011.

[39] G. H. Zhang, K. Y. Xing, G. Y. Zhang, and Z. Y. He,

Memetic algorithm with meta-Lamarckian learning and simplex search for distributed flexible assembly permutation flowshop scheduling problem, *IEEE Access*, vol. 8, no. 1, pp. 96115–96128, 2020.
[40] T. J. Choi, J. Togelius, and Y. G. Cheong, Advanced Cauchy mutation for differential evolution in numerical optimization, *IEEE Access*, vol. 8, pp. 8720–8734, 2020.
[41] M. Ma, J. Wu, Y. Shi, L. Yue, C. Yang, and X. Chen, Chaotic random opposition-based learning and Cauchy mutation improved moth-flame optimization algorithm for intelligent route planning of multiple UAVs, *IEEE Access*, vol. 10, pp. 49385–49397, 2022.
[42] G. Rudolph, Local convergence rates of simple evolutionary algorithms with Cauchy mutations, *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 4, pp. 249–258, 1997.
[43] A. K. Qin, V. L. Huang, and P. N. Suganthan, Differential evolution algorithm with strategy adaptation for global numerical optimization, *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 2, pp. 398–417, 2009.
[44] Z. Yang, K. Tang, and X. Yao, Self-adaptive differential evolution with neighborhood search, in *Proc. 2008 IEEE Congress on Evolutionary Computation* (*IEEE World Congress on Computational Intelligence*), Hong Kong, China, 2008, pp. 1110–1116.
[45] Z. Yang, X. Li, C. P. Bowers, T. Schnier, K. Tang, and X. Yao, An efficient evolutionary approach to parameter identification in a building thermal model, *IEEE Transactions on Systems*, *Man*, *and Cybernetics*, *Part C*, vol. 42, no. 6, pp. 957–969, 2012.
[46] J. Derrac, S. García, D. Molina, and F. Herrera, A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms, *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 3–18, 2011.
[47] M. Sun, C. Sun, X. Li, G. Zhang, and F. Akhtar, Large-scale expensive optimization with a switching strategy, *Complex System Modeling and Simulation*, vol. 2, no. 3, pp. 253–263, 2022.
[48] L. Wang, Z. Pan, and J. Wang, A review of reinforcement learning based intelligent optimization for manufacturing scheduling, *Complex System Modeling and Simulation*, vol. 1, no. 4, pp. 257–270, 2021.

Guanghui Zhang received the PhD degree in control science and engineering from Xi'an Jiaotong University, Xi'an, China in 2018. Since 2019, he has been with the School of Information Science and Technology, Hebei Agricultural University, Baoding, China. His current research interests include control and scheduling of flexible and distributed manufacturing system.

Wenjing Ma is currently pursuing the BS degree in data science and big data technology at Hebei Agricultural University, Baoding, China. Her current research interests include quanta inspired evolutionary algorithm and distributed evolutionary algorithms.

Keyi Xing received the BS degree in mathematics from Northwest University, Xi'an, China, in 1982, the MS degree in applied mathematics from Xidian University, Xi'an, China in 1985, and the PhD degree in systems engineering from Xi'an Jiaotong University, Xi'an, China in 1994. He was with Xidian University in 1985. Since 2004, he has been with Xi'an Jiaotong University. His research interests include control and scheduling of automated manufacturing and hybrid systems.

Lining Xing received the bachelor degree in economics and science from Xi'an Jiaotong University, Xi'an, China, in 2002, and the PhD degree in management science from National University of Defense Technology, Changsha, China, in 2009. He is currently a professor with Xidian University. He visited the School of Computer, University of Birmingham, Birmingham, UK, from November 2007 to November 2008. His research interests include intelligent optimization, mission planning, and resource scheduling.

Kesheng Wang is a professor at Norwegian University of Science and Technology. His research interests include manufacturing system theory, applied computational intelligence, and swarm intelligence. He is the fellow of the Norwegian Academy of Engineering.