Science Bulletin 64 (2019) 1402–1408

Contents lists available at ScienceDirect

Science Bulletin

journal homepage: www.elsevier.com/locate/scib

Article

An improved differential evolution algorithm for learning high-fidelity quantum controls

Xiaodong Yang [a], Jun Li [b,c,d,*], Xinhua Peng [a,e,*]

[a] CAS Key Laboratory of Microscale Magnetic Resonance and Department of Modern Physics, University of Science and Technology of China, Hefei 230026, China
[b] Shenzhen Institute for Quantum Science and Engineering and Department of Physics, Southern University of Science and Technology, Shenzhen 518055, China
[c] Center for Quantum Computing, Peng Cheng Laboratory, Shenzhen 518055, China
[d] Shenzhen Key Laboratory of Quantum Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, China
[e] Synergetic Innovation Centre of Quantum Information & Quantum Physics, University of Science and Technology of China, Hefei 230026, China

ARTICLE INFO

Article history:
Received 29 May 2019
Received in revised form 13 June 2019
Accepted 8 July 2019
Available online 15 July 2019

Keywords:
Control pulses searching
Differential evolution
Quantum states and gates preparation
Pulse imperfections
Random measurement errors

ABSTRACT

Precisely and efficiently designing control pulses for the preparation of quantum states and quantum gates are the fundamental tasks for quantum computation. Gradient-based optimal control methods are the routine to design such pulses. However, the gradient information is often difficult to calculate or measure, especially when the system is not well calibrated or in the presence of various uncertainties. Gradient-free evolutionary algorithm is an alternative choice to accomplish this task but usually with low-efficiency. Here, we design an efficient mutation rule by using the information of the current and the former individuals together. This leads to our improved differential evolution algorithm, called daDE. To demonstrate its performance, we numerically benchmark the pulse optimization for quantum states and quantum gates preparations on small-scale NMR system. Further numerical comparisons with conventional differential evolution algorithms show that daDE has great advantages on the convergence speed and robustness to several uncertainties including pulse imperfections and measurement errors.

1. Introduction

Quantum control is valuable for almost all areas of quantum technologies [1,2]. It plays a significant role in diverse quantum experimental platforms, such as magnetic resonance systems [3,4], superconducting qubits [5,6] and trapped ions [7,8]. A fundamental task in quantum control is to find appropriate control fields that can achieve target quantum operations. This is particularly important for the practical realization of quantum computation, due to the existence of noises and uncertainties unavoidable in realistic physical systems [9]. Quantum control searching is most often solved via a numerical optimization algorithm. In the past decades, researchers have devoted great efforts to designing fast, reliable, and robust quantum control algorithms [10]. Recently, there emerged great interests in developing learning-based quantum control algorithms. Broadly speaking, learning control algorithms can be divided into two classes, i.e., gradient-based and gradient-free. For example, the known gradient ascent pulse engineering (GRAPE) [11] is gradient-based, and has found substantial applications, both numerically [12,13] and experimentally [3,4,14]. However, in many cases, and especially for high dimensional quantum control tasks, computing or measuring the gradient information can be very hard, because it requires good knowledge of the controlled system or large amount of experimental measurements. This difficulty is further exacerbated when taking into account of control uncertainties. One can employ machine learning methods [15–21] to deal with this problem. Another promising approach to tackle this problem is to apply gradient-free algorithms, for example, Nelder-Mead algorithm (NM) [22,23] and differential evolution algorithm (DE) [5,6,24–26].

DE is a simple but competitive algorithm for real-parameter optimization problems. Since it was proposed [27], various variants and applications have appeared [28,29]. Belonging to the evolutionary algorithm family motivated by Darwin's theory of evolution, DE functions through simulating the natural evolution process by performing mutation, crossover and selection operations in the population space which consists of a set of individuals. Each individual is routinely represented by a real-valued vector containing the to-be-optimized parameters that relate to the settled quantum control problems. These individuals are then evaluated by setting a fitness function induced by the control target and iteratively renewed according to the algorithm procedures. The mutation strategy is very crucial for the efficiency of DE. Apart

* Corresponding authors.
E-mail addresses: lij3@sustech.edu.cn (J. Li), xhpeng@ustc.edu.cn (X. Peng).

https://doi.org/10.1016/j.scib.2019.07.013
2095-9273/

from the frequently used DE variants, such as "DE/rand/1", "DE/rand/2", "DE/best/1", "DE/best/2", "DE/current-to-best/1" [29] and their combinations, researchers have proposed new mutation strategies in recent years [30–32]. However, for most of these strategies, algorithm designers need to tune their parameters, often manually (a tedious and time-consuming task) and can only achieve limited improvement. Here, we use the "direction" information related to some individuals in the former and current generations to design new mutation strategies, referred to as direction averaged Differential Evolution (daDE). We then apply our daDE algorithm to the tasks of quantum state preparation and quantum gate implementation, and find that it shows several remarkable features including simplicity and robustness, as expected.

2. Direction averaged differential evolution algorithm

In this section, we introduce the proposed algorithm daDE in details. The standard DE algorithm can be improved in a number of different ways, and our work is mainly about how to modify the mutation rule. By convention, we use $P = \{X_1, \ldots, X_{NP}\}$ to represent the population with $X_i = (X_{i,1}, \ldots, X_{i,D})$ being the i-th individual, where D is the dimension of the optimization problem and NP is the population size. Here, each individual that represented by a vector X_i relates to the problem parameters to-be-optimized, and the population P contains NP candidate individuals for implementing the DE algorithm procedures.

2.1. Algorithm initialization

The algorithm starts by randomly generating an initial population $P = \{X_1^0, \ldots, X_{NP}^0\}$. As the search space is always constrained by some maximum bounds $X_{\max} = (X_{\max,1}, \ldots, X_{\max,D})$ and minimum bounds $X_{\min} = (X_{\min,1}, \ldots, X_{\min,D})$, the initial population can be generated as the following way:

$$X_{i,j}^0 = X_{\min,j} + rand_{i,j}[0,1](X_{\max,j} - X_{\min,j}), \tag{1}$$

where $rand_{i,j}[0,1]$ is a uniformly distributed random number in the range $[0,1]$.

2.2. Mutation rule: direction average strategy

After initialization, for each individual in the G-th generation X_i^G, a donor/mutant vector V_i^G is generated according to some mutation rule.

The mutation rule is crucial for the efficiency (convergence speed) of a DE algorithm. A good mutation rule should make the algorithm converge fast while avoiding getting trapped in any local extrema, depending on the balancing between the stochastic and deterministic nature of the algorithm. The known mutation rules contain three basic ingredients: (1) a current individual in the population of the G-th generation X_i^G; (2) the best individual in the population of the G-th generation X_b^G; (3) a randomly chosen individual in the population of the G-th generation X_r^G. The former two ingredients are deterministic while the last one is stochastic. Mutation rules are hence constructed via balancing the above three ingredients.

Here, we introduce a new ingredient: a current individual in the population of the $(G-1)$-th generation X_i^{G-1}. We use this ingredient in combination with the average of some chosen individuals in the G-th generation to form our new mutation rule. As well known from the DE procedure, the surviving individuals in the G-th generation have better or at least comparable fitness than the individuals in the $(G-1)$-th generation. Therefore, the "direction" information is embedded in this mutation rule, which provides a more definite and efficient determinism. Now we introduce the devised direction averaged mutation rule:

$$V_i^G = X_i^G + F_1 \cdot (X_{\text{avg}}^G - X_i^{G-1}) + F_2 \cdot (X_{r_1^i}^G - X_{r_2^i}^G), \tag{2}$$

where $G > 1$ and X_{avg}^G represents the best first S individuals in the G-th generation, namely $X_{\text{avg}}^G = \frac{1}{S}\sum_{j=1}^{S} X_j^G$. When $G = 1$ we use a basic mutation rule:

$$V_i^G = X_{r_3^i}^G + F_0 \cdot (X_{r_4^i}^G - X_{r_5^i}^G) + F_0 \cdot (X_{r_6^i}^G - X_{r_7^i}^G). \tag{3}$$

The subscripts $r_1^i, \ldots, r_7^i$ represent some mutually exclusive random integers selected from $\{0, 1, \ldots, NP\}$, such that $r_1^i, \ldots r_7^i = randn_i[0, NP]$. The parameters F_0, F_1, F_2 are some fixed numbers during the whole algorithm in the range $[0.5, 1]$ and the suggested S is set to be in the range $[0.1 \times NP, 0.4 \times NP]$ according to our tests.

2.3. Crossover

After generating the mutant vector $V_i^G = (V_{i,1}^G, \ldots, V_{i,D}^G)$, we use it to crossover with $X_i^G = (X_{i,1}^G, \ldots, X_{i,D}^G)$ to produce the trial vector $U_i^G = (U_{i,1}^G, \ldots, U_{i,D}^G)$ by the binomial crossover strategy:

$$U_{i,j}^G = \begin{cases} V_{i,j}^G, & \text{if } rand_{i,j}[0,1] \leqslant CR \text{ or } j = j_{\text{rand}}, \\ X_{i,j}^G, & \text{otherwise}, \end{cases} \tag{4}$$

where $j_{\text{rand}} \in [1, 2, \ldots, D]$ is a randomly chosen index and CR is a fixed number in the range $[0.8, 1]$.

Algorithm 1: daDE algorithm

```
G=0;
Initialize population P = {X_1^0, ..., X_NP^0} randomly;
Set the algorithm parameters F_0, F_1, F_2, S, CR;
// Main loop;
while The termination criteria are not met do
    G = G + 1;
    for i = 1 to NP do
        // Mutation;
        r_1^i, ..., r_7^i = randn_i[0, NP](mutually exclusive);
        if G = 1 then
            V_i^G = X_{r_3^i}^G + F_0.(X_{r_4^i}^G - X_{r_5^i}^G) + F_0.(X_{r_6^i}^G - X_{r_7^i}^G);
        else
            V_i^G = X_i^G + F_1.(X_avg^G - X_i^{G-1}) + F_2.(X_{r_1^i}^G - X_{r_2^i}^G);
        end
        // Crossover;
        for j = 1 to D do
            if rand_{i,j}[0, 1] ⩽ CR or j = j_rand then
                U_{i,j}^G = V_{i,j}^G;
            else
                U_{i,j}^G = X_{i,j}^G;
            end
        end
        // Selection;
        if f(U_i^G)) ⩽ f(X_i^G) then
            X_i^{G+1} = U_i^G;
        else
            X_i^{G+1} = X_i^G;
        end
    end
end
```

2.4. Selection

After all the trial vectors are generated, we decide the survivors for the next generation by a greedy selection strategy:

$$X_i^{G+1} = \begin{cases} U_i^G, & \text{if } f(U_i^G) \leqslant f(X_i^G), \\ X_i^G, & \text{otherwise}, \end{cases} \tag{5}$$

where $f(\cdot)$ is the fitness function to be minimized. With all these procedures defined, the pesudo-code of daDE is given in Algorithm 1.

3. Problem setup

An n-qubit quantum system with transverse time-varying magnetic control fields can be described by $\mathcal{H} = \mathcal{H}_S + \sum_{j=1}^{n}(u_x^j(t)\sigma_x^j + u_y^j(t)\sigma_y^j)$, where $\mathcal{H}_S$ is the system Hamiltonian, σ_x^j, σ_y^j are the Pauli operators on the j-th qubit, and $j = 1, 2, \ldots, n, t \in [0, T]$. These controls will generate an evolution operator $U(T)$, which can be optimized to achieve the target quantum states or gates. In the following, we formalize the quantum states and gates preparation problems.

3.1. Quantum states preparation

The target is to prepare a quantum state ρ_t from a specific quantum state ρ_0. The controls induced evolution operator $U(T)$ can transform the initial state to $\rho_f = U(T)\rho_0 U(T)^\dagger$. A performance metric is then used to judge the distance between ρ_f and ρ_t [9], namely the state fidelity $F(\rho_f, \rho_t) = \mathrm{Tr}(\rho_f \rho_t)$, where we suppose both ρ_t and ρ_f are pure states. Thus, the state preparation problem can be formalized as

$$\begin{aligned} &\max \quad F(\rho_f, \rho_t) = \mathrm{Tr}(U(T)\rho_0 U(T)^\dagger \rho_t), \\ &\text{s.t.} \quad \dot{U}(t) = -\mathrm{i}\left(\mathcal{H}_S + \sum_{j=1}^{n}(u_x^j(t)\sigma_x^j + u_y^j(t)\sigma_y^j)\right)U(t). \end{aligned} \tag{6}$$

3.2. Quantum gates preparation

Similarly, to prepare a target quantum gate U_t, the controls will generate an evolution operator $U_f = U(T)$. It is judged by the performance metric [9] $F(U_t, U_f) = |\mathrm{Tr}(U_f U_t^\dagger)|/4^n$. Therefore, the gate preparation problem can be formalized as

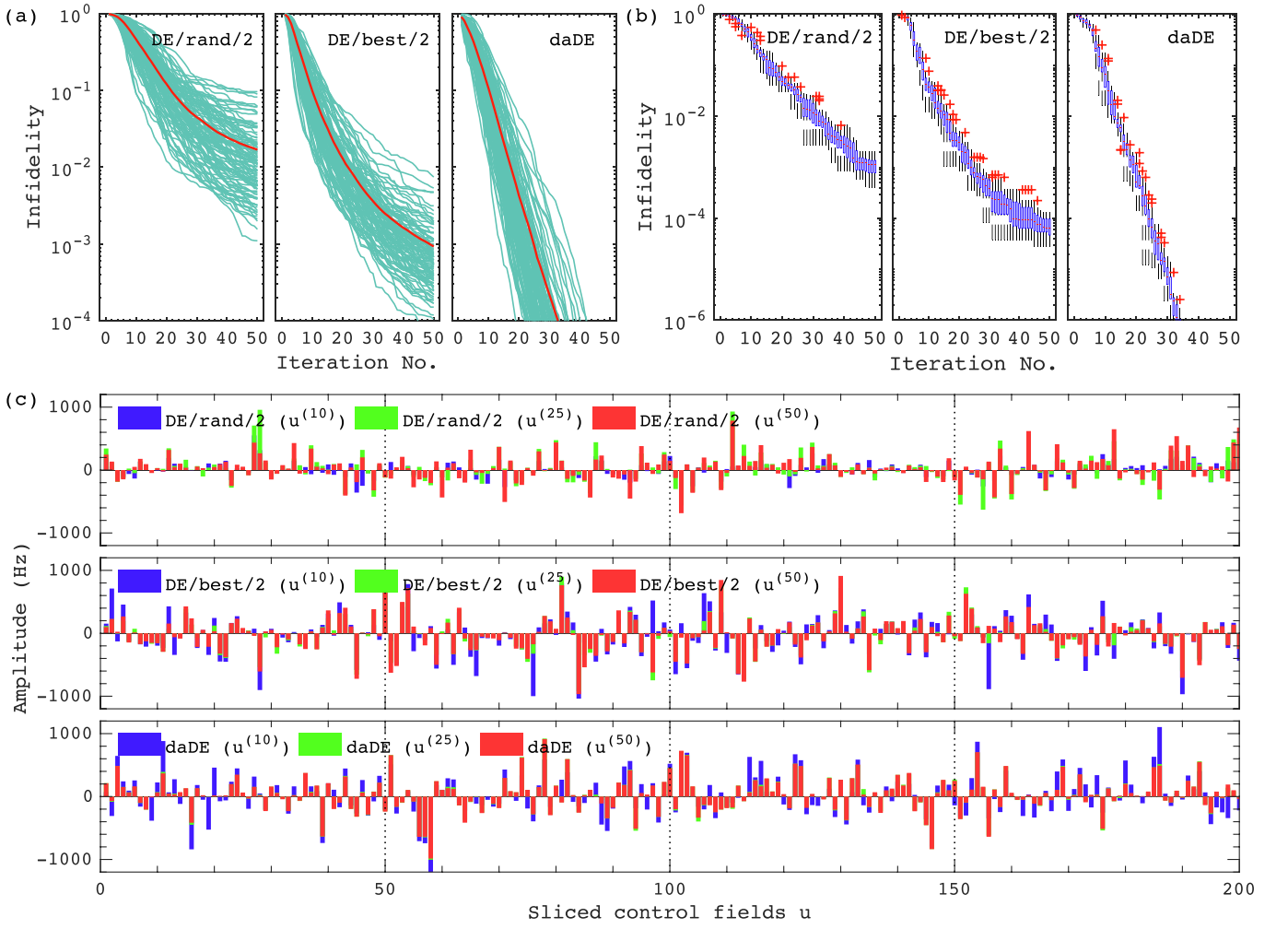

Fig. 1. (Color online) The numerical results of searching the control pulses for the Bell state preparation. (a) The state infidelity (i.e., $1 - F$) with respect to the iteration number for searching the Bell state. The control pulses are all divided into $M = D/4 = 50$ slices with $\Delta t = 10^{-4}$ s and the initial control pulses are all randomly generated in the range $[-50, 50]$. Each algorithm runs 100 times and stops when the iteration number reaches 50. The thick lines represent the averaged value over the 100 runs. (b) The corresponding box and whisker plot of the infidelity distribution for the best run among the 100 runs. (c) The corresponding sliced control fields when the iteration number is 10, 25 and 50. The algorithm parameters are set as follows (1) for "DE/rand/2": $NP = 20, D = 200, F = 0.3, CR = 0.9$. (2) for "DE/best/2": $NP = 20, D = 200, F = 0.5, CR = 0.95$. (3) for daDE: $NP = 20, D = 200, F_0 = 0.9, F_1 = 0.9, F_2 = 0.4, CR = 0.95, S = 0.25NP = 5$.

$$\max \quad F(U_f, U_t) = |\mathrm{Tr}(U(T)U_t^{\dagger})|/4^n, \tag{7}$$
$$\text{s.t.} \quad \dot{U}(t) = -\mathrm{i}\left(\mathcal{H}_S + \sum_{j=1}^{n}(u_x^j(t)\sigma_x^j + u_y^j(t)\sigma_y^j)\right)U(t).$$

4. Numerical simulations

To solve the above state and gate optimization problems, iterative algorithms are always used, here we consider daDE and two traditional DE algorithms "DE/rand/2" and "DE/best/2" [29]. Firstly, we divide the total controls period T into M equal slices, and in each slice ($\Delta t = T/M$) the control amplitude is a constant. Therefore, the evolution operator for m-th slice can be expressed as

$$U_m = \exp\left\{-\mathrm{i}\Delta t\left[\mathcal{H}_S + \sum_{j=1}^{n}(u_x^j[m]\sigma_x^j + u_y^j[m]\sigma_y^j)\right]\right\}. \tag{8}$$

The total evolution operator can then be calculated as $U(T) = \prod_{m=1}^{M} U_m$. Secondly, the DE algorithms are used to optimize the piece-wise constant controls $u = (u_x^j[m], u_y^j[m])$, where $j = 1, \ldots, n, m = 1, \ldots, M$. Thus, the optimization problem is to maximize the state or gate fidelity over u.

We conduct numerical simulations on a two-qubit nuclear magnetic resonance (NMR) system. The sample chloroform we use consists of two nuclear spins ^{1}H and ^{13}C. Label the spins 1, 2, the internal Hamiltonian can be described as $\mathcal{H}_S = \sum_{i=1}^{2} \Delta\omega_0^i\sigma_z^i/2 + \pi J_{12}\sigma_z^1\sigma_z^2/2$, where $\Delta\omega_0^i$ represents the offset of the i-th spin in the rotating frame, and $J_{12} = 217.4$ Hz is the scalar coupling strength between the spins 1 and 2.

4.1. Direct benchmarking

For the state preparation problem, we choose the Bell state $\rho_t = |+\rangle\langle+|$ as our target state, where $|+\rangle = (|0\rangle + 1\rangle)/\sqrt{2}$. As shown in Fig. 1a, each DE algorithm is run 100 times and the stopping criterion is that the iteration number exceeds 50. The total time length of the control pulses is set as $T = 5$ ms, which is much larger than the minimum time needed [33] (about 2.3 ms) so that all these three DE algorithms are capable of accomplishing this task in an acceptable running time with limited resource. It should be noted that all the algorithm parameters are tuned carefully to make sure each of them is in its nearly best performance. The results show that daDE converges fastest and the infidelity will continue to decrease sharply when applying more iterations, while

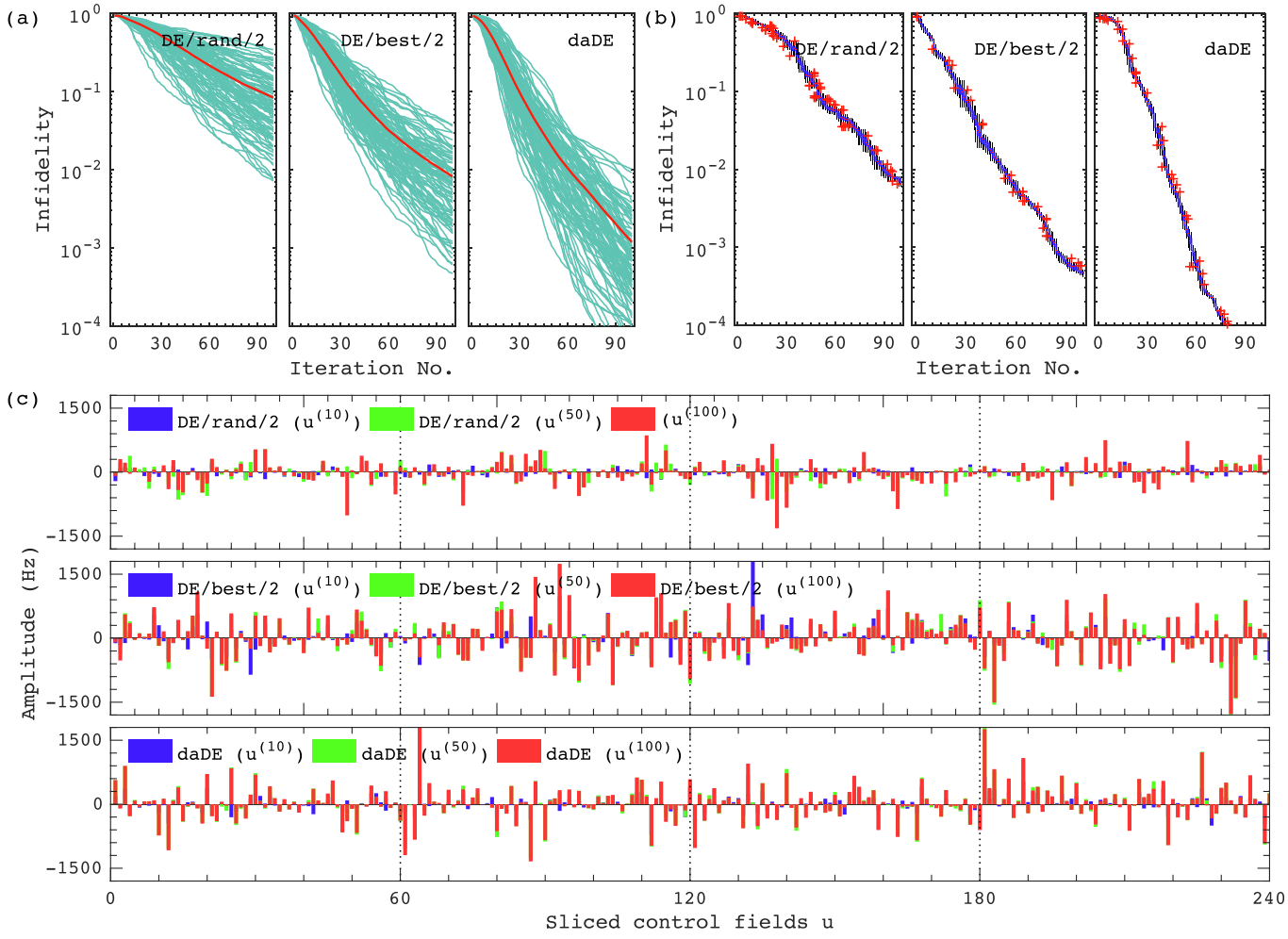

Fig. 2. (Color online) The numerical results of searching the control pulses for the CNOT gate preparation. (a) The gate infidelity (i.e., $1 - F$) with respect to the iteration number for searching the CNOT gate. The control pulses are all divided into $M = D/4 = 60$ slices with $\Delta t = 10^{-4}$ s and the initial control pulses are all randomly generated in the range $[-50, 50]$. Each algorithm runs 100 times and stops when the iteration number reaches 100. The thick lines represent the averaged value over the 100 runs. (b) The corresponding box and whisker plot of the infidelity distribution for the best run among the 100 runs. (c) The corresponding sliced control fields when the iteration number is 10, 50 and 100. The algorithm parameters are set as follows (1) for "DE/rand/2": $NP = 20, D = 240, F = 0.3, CR = 0.9$. (2) for "DE/best/2": $NP = 20, D = 240, F = 0.5, CR = 0.95$. (3) for daDE: $NP = 20, D = 240, F_0 = 0.9, F_1 = 0.9, F_2 = 0.4, CR = 0.95, S = 0.25NP = 5$.

"DE/rand/2" and "DE/best/2" converge slower and the infidelity decrement with respect to the number of iterations will become smaller and smaller. To further characterize the evolution of the population distribution of these DE algorithms, we demonstrate the box and whisker plot (a five-number summary contain the maximum, the minimum, the lower and upper quartiles, and the median in the data set) using the best run among their 100 runs in Fig. 1b. One can clearly see that the populations in the beginning are all very tightly grouped, this is because the initial individuals in the populations are generated randomly in the same small range. However, when the iteration number increases, the populations in "DE/rand/2" and "DE/best/2" become disperse gradually, while the population in daDE does not disperse much. This indicates that the individuals with better performance in daDE are selected and maintained more efficiently, thus demonstrating the effectiveness of our proposed mutation rule. What's more, we show the sliced control fields of the current best individual when the iteration number equals to 10, 25 and 50 for each DE algorithm in Fig. 1c. From these comparisons, we can not tell that there exist significant difference for the final sliced control fields, but one can find that for "DE/rand/2", roughly speaking, the amplitudes are a little smaller. In addition, the control fields in the 25-th iteration for "DE/best/2", especially for daDE, are very close to that in the 50-th iteration. This indicates that these two algorithms have very fast searching speed towards the final optimal control fields in the first half of the optimization process.

For the quantum gate preparation problem, the results are similar, as shown in Fig. 2. We directly run these three DE Algorithms 100 times for searching the CNOT gate and stop them when the iteration number exceeds 100, see the results in Fig. 2a. We also show the corresponding box and whisker plot of the infidelity distribution for the best run among the 100 runs in Fig. 2b, and the sliced control fields for the best individual when the iteration number is 10, 50 and 100 in Fig. 2c. Again, we can conclude that daDE converges fastest and its best run has very concentrate population distributions. The amplitudes of the control fields of the best individual for "DE/rand/2" are much smaller than that for "DE/best/2" and daDE, while the control fields in 50-th iteration for daDE is very close to that in 100-th iteration.

4.2. Under uncertainties

In the following, we consider several uncertainties to compare the performance of daDE and the other two basic DE algorithms "DE/rand/2" and "DE/best/2". In real quantum control processes, there exist many kinds of uncertainties that distort the performance of control pulses, such as model and parameter errors, control imperfections and measurement errors [34]. Here, we first consider the control imperfections, which are induced by hardware limitations, the field inhomogeneity and some other factors. For simplicity, we model this pulse distortion by a linear filter [13], i.e., $u'(t) = \mathcal{D}[\boldsymbol{u}(t)] = \int_0^\infty h(t-\tau)\boldsymbol{u}(t)\mathrm{d}\tau$, where $h(t) = \frac{1}{t_r}\mathrm{e}^{-\frac{1}{t_r}}, t \geqslant 0$ is the impulse function. The pulse distortion degree is characterized by the time constant t_r, the heavier of the distortion, the bigger of t_r. In Fig. 3, we set the stopping criterion as the final state or gate fidelity reaches 0.999 and display the iteration number with respect to the pulse distortion $t_r/\Delta t$. Here, we only consider the runs that the final fidelity reaches 0.999 within 200 iterations for the Bell state and 300 iterations for the CNOT gate. Therefore, the runs not satisfying this condition will not be included and we give the corresponding success rates at the bottom. From these two figures, we can find that for this kind of uncertainty, "DE/rand" has very low success rates and the biggest iteration number variances, daDE owns the smallest iteration number dispersion and almost 100% success probabilities. This indicates that daDE is the most robust to the assumed control imperfections.

Secondly, we consider measurement errors that induced by technical limitations, the statistic properties of the measurement strategies and some other factors. They can be further classified into

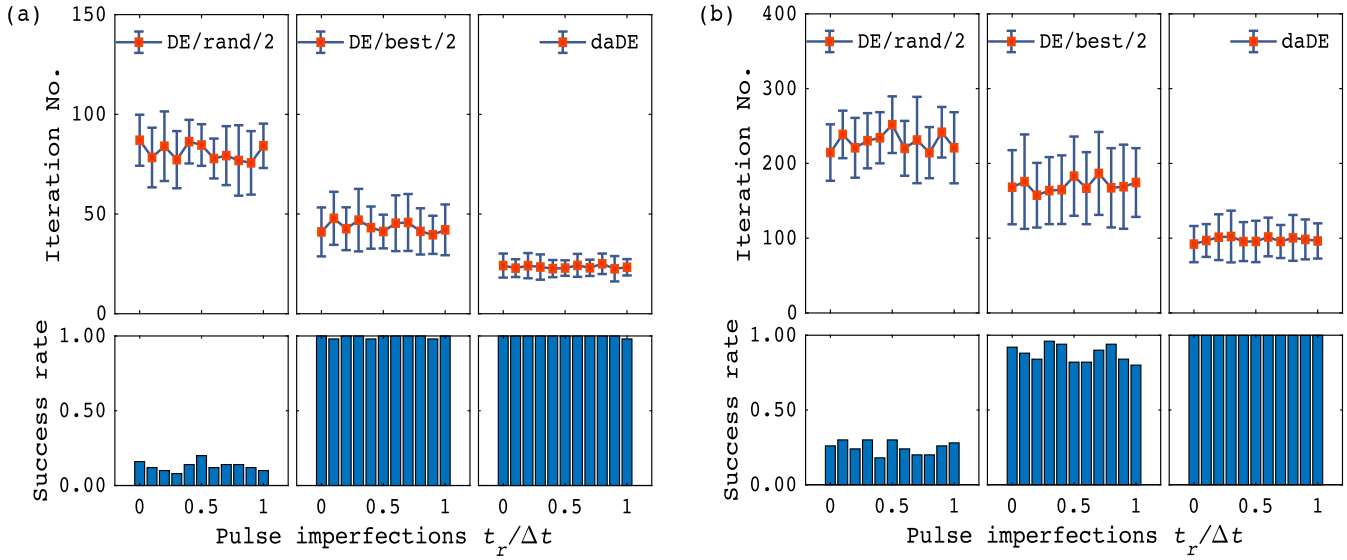

Fig. 3. (Color online) The numerical results of searching the control pulses for the Bell state and the CNOT gate when pulse imperfections exist. (a) The iteration number with respect to the pulse imperfections $t_r/\Delta t$ for searching the Bell state. The stopping criterion is the state fidelity reaches 0.999 within 200 iterations, thus we exclude the failure runs and give the corresponding success rates at the bottom. Each algorithm runs 100 times and the algorithm parameters are set as follows. (1) For "DE/rand/2": $NP = 20, D = 200, F = 0.3, CR = 0.9$. (2) For "DE/best/2": $NP = 20, D = 200, F = 0.5, CR = 0.95$. (3) For daDE: $NP = 20, D = 200, F_0 = 0.9, F_1 = 0.9, F_2 = 0.4, CR = 0.95, S = 0.25NP = 5$. (b) The iteration number with respect to the pulse imperfections $t_r/\Delta t$ for searching the CNOT gate. The stopping criterion is the gate fidelity reaches 0.999 within 300 iterations, thus we exclude the failure runs and give the corresponding success rates at the bottom. Each algorithm runs 100 times and the algorithm parameters are set as follows. (1) For "DE/rand/2": $NP = 20, D = 240, F = 0.3, CR = 0.9$. (2) For "DE/best/2": $NP = 20, D = 240, F = 0.5, CR = 0.95$. (3) For daDE: $NP = 20, D = 240, F_0 = 0.9, F_1 = 0.9, F_2 = 0.4, CR = 0.95, S = 0.25NP = 5$.

systematic and random measurement errors. Here, we only include an additive random measurement error which can be described by a normal distribution with the mean 0 and variance γ^2, i.e., $F'(\rho_f,\rho_t)=F(\rho_f,\rho_t)+N(0,\gamma^2)$ or $F'(U_f,U_t)=F(U_f,U_t)+N(0,\gamma^2)$. As shown in Fig. 4, we run each Algorithm 50 times and stop when the gate fidelity reaches 0.999 or the maximum iteration number exceeds 200 for Bell state and 300 for CNOT gate. Similarly, we exclude the failed runs that can not reach the gate fidelity 0.999 and plot the corresponding success rates at the bottom. From these two figures, we can find that this kind of uncertainty influences the performance of these DE algorithms a lot, for "DE/rand/2" and "DE/best/2", the success rates are very low for both the Bell state and the CNOT gate preparations, while daDE shows sufficient high success rates and relatively small iteration number variances, thus generally speaking, daDE is the most robust in this case.

5. Discussions and conclusions

High-fidelity quantum control has great significance for quantum computation and quantum engineering. In this work, we proposed an improved differential evolution algorithm, namely daDE, to solve the quantum state and gate preparation problems instead of the frequently-used gradient based optimal control methods. Compared with two basic DE variants "DE/rand/2" and "DE/best/2", we consider the Bell state and the CNOT gate preparation tasks on a two-qubit NMR system. Numerical simulations show that daDE has fastest convergence speed for these tasks. What's more, we consider the control imperfections and random measurement errors in the control process to further characterize the performance of daDE. Numerical comparisons reveal that daDE owns strongest robustness against these two uncertainties and high success rates with the limited resource.

Differential evolution algorithms are representative and competitive gradient-free methods to solve many real-valued optimization problems, including quantum states/gate preparation tasks or some other optimization tasks in quantum computation. The avoid of calculating or measuring the gradient information greatly saves the resource needed and hence makes it easier to implement experimentally. However, the low convergence speed of the traditional DE algorithms often makes the performance of their real applications below our expectations. This can be mitigated by designing more efficient algorithmic procedures and parameter tuning strategies. What's more, some powerful methods, such as machine learning [5], have been combined to improve the DE algorithm efficiency further. Our proposed daDE optimizes the algorithmic procedure itself to achieve higher convergence speed and stronger robustness.

Future researches can be conducted on applying daDE to large quantum systems or other optimization problems in quantum computation to investigate its abilities and properties. Advanced DE variant can also be designed by combing daDE with many other strategies. In addition, these gradient-free DE algorithms can be easily modified to closed-loop versions to further improve the control precision experimentally.

Conflict of interest

The authors declare that they have no conflict of interest.

Acknowledgments

Jun Li was supported by the National Natural Science Foundation of China (11605005, 11875159, and U1801661), Science, Technology and Innovation Commission of Shenzhen Municipality (ZDSYS20170303165926217 and JCYJ20180302174036418), Guangdong Innovative and Entrepreneurial Research Team Program (2016ZT06D348). Xinhua Peng was supported by the National Key Research and Development Program of China (2018YFA0306600), the National Science Fund for Distinguished Young Scholars (11425523), Projects of International Cooperation

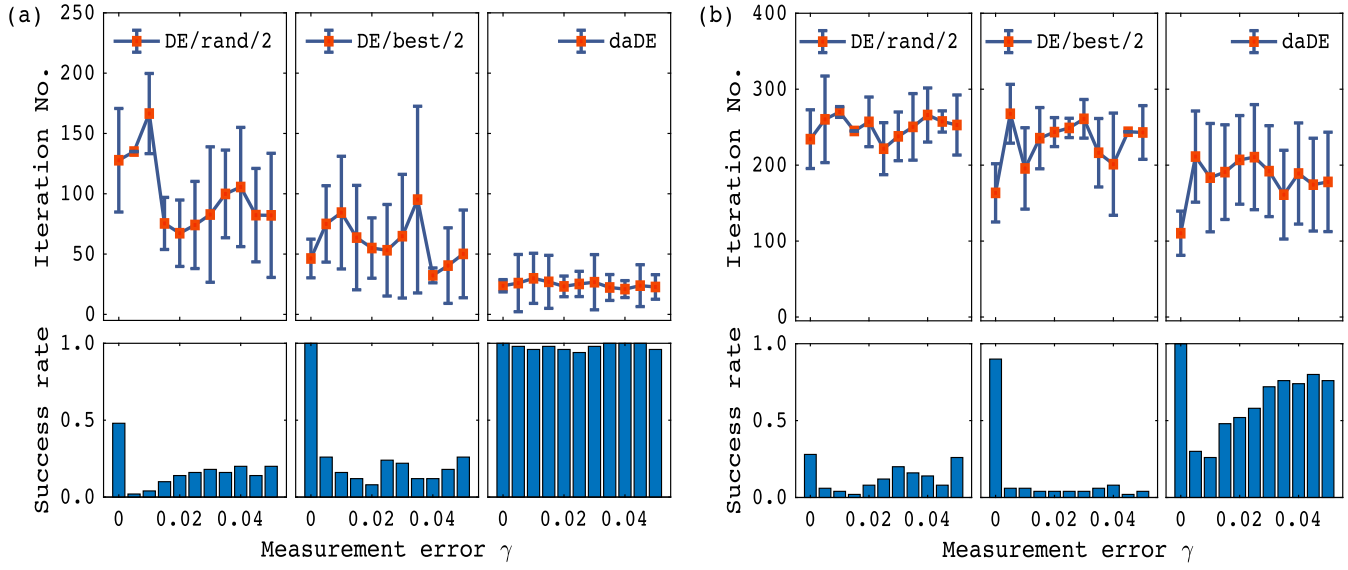

Fig. 4. (Color online) The numerical results of searching the control pulses for the Bell state and the CNOT gate when measurement errors exist. (a) The iteration number with respect to the measurement errors γ for searching the Bell state. The stopping criterion is the state fidelity reaches 0.999 within 200 iterations, thus we exclude the failure runs and give the corresponding success rates at the bottom. Each algorithm runs 100 times and the algorithm parameters are set as follows. (1) For "DE/rand/2": $NP=20, D=200, F=0.3, CR=0.9$. (2) For "DE/best/2": $NP=20, D=200, F=0.5, CR=0.95$. (3) For daDE: $NP=20, D=200, F_0=0.9, F_1=0.9, F_2=0.4, CR=0.95, S=0.25NP=5$. (b) The iteration number with respect to the measurement errors γ for searching the CNOT gate. The stopping criterion is the gate fidelity reaches 0.999 within 300 iterations, thus we exclude the failure runs and give the corresponding success rate at the bottom. Each algorithm runs 100 times and the algorithm parameters are set as follows. (1) For "DE/rand/2": $NP=20, D=240, F=0.3, CR=0.9$. (2) For "DE/best/2": $NP=20, D=240, F=0.5, CR=0.95$. (3) For daDE: $NP=20, D=240, F_0=0.9, F_1=0.9, F_2=0.6, CR=0.95, S=0.25NP=5$.

and Exchanges NSFC (11661161018) and Anhui Initiative in Quantum Information Technologies (AHY050000). The authors also thank Xiaopeng Luo and Xi Xing for helpful discussions.

Author contributions

X. Yang proposed this project and did the numerical simulations, as well as wrote the manuscript; J. Li discussed and revised the manuscript; X. Peng supervised the whole project.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.scib.2019.07.013.

References

[1] Rabitz H, de Vivie-Riedle R, Motzkus M, et al. Whither the future of controlling quantum phenomena? Science 2000;288:824–8.
[2] Brif C, Chakrabarti R, Rabitz H. Control of quantum phenomena: past, present and future. New J Phys 2010;12:075008.
[3] Li J, Yang XD, Peng XD, et al. Hybrid quantum-classical approach to quantum optimal control. Phys Rev Lett 2017;118:150503.
[4] Lu DW, Li KR, Li J, et al. Enhancing quantum control by bootstrapping a quantum processor of 12 qubits. npj Quantum Inf 2017;3:45.
[5] Zahedinejad E, Ghosh J, Sanders BC. High-fidelity single-shot toffoli gate via quantum control. Phys Rev Lett 2015;114:200502.
[6] Zahedinejad E, Ghosh J, Sanders BC. Designing high-fidelity single-shot three-qubit gates: a machine-learning approach. Phys Rev Appl 2016;6:054005.
[7] Gaebler JP, Tan TR, Lin Y, et al. High-fidelity universal gate set for $^{9}Be^{+}$ ion qubits. Phys Rev Lett 2016;117:060505.
[8] Monz T, Kim K, Hänsel W, et al. Realization of the quantum toffoli gate with trapped ions. Phys Rev Lett 2009;102:040501.
[9] Nielsen MA, Chuang IL. Quantum computation and quantum information. 10th ed. Cambridge University Press; 2010.
[10] Glaser SJ, Boscain U, Calarco T, et al. Training Schrödingers cat: quantum optimal control. Eur Phys J D 2015;69:279.
[11] Khaneja N, Reiss T, Kehlet C, et al. Optimal control of coupled spin dynamics: design of nmr pulse sequences by gradient ascent algorithms. J Magn Reson 2005;172:296–305.
[12] Ryan CA, Negrevergne C, Laforest M, et al. Liquid-state nuclear magnetic resonance as a testbed for developing quantum control methods. Phys Rev A 2008;78:012328.
[13] Wu RB, Chu B, Owens DH, et al. Data-driven gradient algorithm for high-precision quantum control. Phys Rev A 2018;97:042122.
[14] Feng G, Cho FH, Katiyar H, et al. Gradient-based closed-loop quantum optimal control in a solid-state two-qubit system. Phys Rev A 2018;98:052341.
[15] Bukov M, Day AGR, Sels D, et al. Reinforcement learning in different phases of quantum control. Phys Rev X 2018;8:031086.
[16] Arrazola JM, Bromley TR, Izaac J, et al. Machine learning method for state preparation and gate synthesis on photonic quantum computers. Quantum Sci Technol 2018;4:024004.
[17] Gao J, Qiao LF, Jiao ZQ, et al. Experimental machine learning of quantum states. Phys Rev Lett 2018;120:240501.
[18] An Z, Zhou D. Deep reinforcement learning for quantum gate control. arXiv:190208418, 2019.
[19] Zhang XM, Wei ZZ, Asad R, et al. When reinforcement learning stands out in quantum control? A comparative study on state preparation. arXiv:190202157, 2019.
[20] Banchi L, Pancotti N, Bose S. Quantum gate learning in qubit networks: toffoli gate without time-dependent control. npj Quantum Inf 2016;2:16019.
[21] Wu RB, Ding HJ, Dong DY, et al. Learning robust and high-precision quantum controls. Phys Rev A 2019;99:042327.
[22] Kelly J, Barends R, Campbell B, et al. Optimal quantum control using randomized benchmarking. Phys Rev Lett 2014;112:240504.
[23] Egger DJ, Wilhelm FK. Adaptive hybrid optimal quantum control for imprecisely characterized systems. Phys Rev Lett 2014;112:240503.
[24] Shir OM, Roslund J, Leghtas Z, et al. Quantum control experiments as a testbed for evolutionary multi-objective algorithms. Genet Program Evol M 2012;13:445–91.
[25] Dong DY, Xing X, Ma HL, et al. Differential evolution for quantum robust control: algorithm, applications and experiments. arXiv:1702.03946, 2017.
[26] Sun YY, Ma HL, Wu CZ, et al. Ensemble control of open quantum systems using differential evolution. In: 2015 10th Asian Control Conference (ASCC). IEEE; 2015. p. 1–6.
[27] Storn R, Price K. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. J Global Optim 1997;11:341–59.
[28] Das S, Suganthan PN. Differential evolution: a survey of the state-of-the-art. IEEE Trans Evol Comput 2011;15:4–31.
[29] Das S, Mullick SS, Suganthan PN. Recent advances in differential evolution-an updated survey. Swarm Evol Comput 2016;27:1–30.
[30] Ali M. Differential evolution with generalized differentials. J Comput Appl Math 2011;235:2205–16.
[31] Cai YQ, Wang JH. Differential evolution with neighborhood and direction information for numerical optimization. IEEE T Cybern 2013;43:2202–15.
[32] Zhang X, Yuen SY. A directional mutation operator for differential evolution algorithms. Appl Soft Comput 2015;30:529–48.
[33] Khaneja N, Brockett R, Glaser SJ. Time optimal control in spin systems. Phys Rev A 2001;63:032308.
[34] Ferrie C, Moussa O. Robust and efficient in situ quantum control. Phys Rev A 2015;91:052306.

Xiaodong Yang is now studying at the Department of Modern Physics, University of Science and Technology of China (Hefei, China) for his Ph.D. degree. His research interests include nuclear magnetic resonance (NMR) based quantum simulation, closed-loop learning control and quantum optimization algorithms.

Jun Li received his Ph.D. degree in physics in 2015 from the Department of Modern Physics, University of Science and Technology of China (Hefei, China). He is now an associate researcher in Institute for Quantum Science and Engineering, Southern University of Science and Technology of China (Shenzhen, China). His research interests include developing scalable quantum control, exploring potentials of quantum computing, studying complex spin dynamics, and building portable nuclear magnetic resonance spectrometer.

Xinhua Peng received her Ph.D. degree in atomic and molecular physics in 2003 from the Wuhan Institute of Physics and Mathematics, Chinese Academy of Sciences (Wuhan, China). She is now a professor in University of Science and Technology of China. Her research interests include nuclear magnetic resonance (NMR) based quantum computation, quantum simulation, quantum control, as well as atomic magnetometer based low-field nuclear magnetic resonance and quantum metrology.