Enhanced quantum state preparation via stochastic predictions of neural networks

Chao-Chao Li,[1] Run-Hong He,[2] and Zhao-Ming Wang[1,*]

[1] *College of Physics and Optoelectronic Engineering, Ocean University of China, Qingdao 266100, China*

[2] *State Key Laboratory of Computer Science, Institute of Software Chinese Academy of Science, Beijing 101408, China*

(Received 1 August 2023; accepted 27 October 2023; published 17 November 2023)

In pursuit of enhancing the prediction capabilities of neural networks, it has been a longstanding objective to create datasets encompassing a diverse array of samples. The purpose is to broaden the horizons of the neural network and continually strive for improved prediction accuracy during the training process, which serves as the ultimate evaluation metric. However, in this paper, we explore an intriguing avenue for enhancing algorithm effectiveness through exploiting the knowledge blindness of the neural network. Our approach centers around a machine learning algorithm utilized for preparing arbitrary quantum states in a semiconductor double quantum dot system, a system characterized by highly constrained control degrees of freedom. We artificially define the characteristics of local optima in the state preparation task. When the optimization falls into a local optimum, disturbances are introduced through stochastic prediction, which gives a chance to escape from the local optimum. Notably, unlike previous methodologies that employ reinforcement learning to identify pulse patterns, we adopt a training approach akin to supervised learning, using it to dynamically design the pulse sequence. This approach has broad applicability and improves the efficiency of the algorithm.

DOI: 10.1103/PhysRevA.108.052418

I. INTRODUCTION

Quantum optimal control, which refers to the design of the external field shapes in quantum dynamical processes for given tasks in the best way possible, has become one of the cornerstones for current quantum technologies [1,2]. The physical platform includes nuclear magnetic resonance experiments [3], captured ions [4,5], superconducting qubits [6,7], nitrogen-vacancy centers [8], and semiconductor quantum dots [9–15]. Among these, spin qubits in semiconductor quantum dots show promise due to their scalability and long coherence times [16–22]. The singlet-triplet (S-T_0) qubit, which is encoded in the singlet-triplet spin subspace of two electrons trapped in a double quantum dot (DQD), is widely used. The advantage over other qubit candidates includes fast qubit operation and independence from uniform fluctuations in the magnetic field, allowing for complete control by electrical pulses [23–26].

Universal quantum computing relies on precise single-qubit rotations around different axes of the Bloch sphere and two-qubit gates capable of performing entanglement [27]. Efficient and precise quantum gate control constructed by deep reinforcement learning (RL) has been investigated [28], which involves executing gates within the constraints of platforms and mitigating errors during execution [29]. In the case of singlet-triplet spin qubits in semiconductor DQDs, fast electrical control of the exchange coupling is necessary for precise control of the rotation rate around the z axis of the Bloch sphere [30]. The quantum state preparation (QSP) algorithm is commonly employed as a subroutine for various tasks. In particular, Refs. [31,32] utilize QSP to implement general quantum measurements on quantum systems and to simulate noisy quantum channels, respectively.

Typically, performing an arbitrary quantum spin gate requires numerically solving a set of coupled nonlinear equations to determine a composite pulse sequence [23,29,30], which is resource and time consuming. Machine learning, a field born out of artificial intelligence, enables the analysis of vast amounts of data beyond human ability or previously imagined methods of enumeration [33,34], and has shown wide applicability on quantum control [35–44]. It is now an active research area and has demonstrated great success in solving lots of physical problems [45–48]. Reference [33] uses the supervised learning algorithm to design pulse sequences that closely match solutions of nonlinear equations. However, in practice complex pulse shapes and lengthy execution times limit the application [49]. RL, one of the subfields of machine learning, has also been used to solve various quantum optimal control problems [50,51]. References [49,52] utilize deep RL [28,37,41,42] to design discrete dynamic pulses for driving an initial state to a fixed state or resetting an arbitrary quantum state to a specific target state. In addition, deep RL has successfully generated arbitrary states from specific states in nitrogen-vacancy center systems [53]. By combining Refs. [52,53], driving between arbitrary quantum states can be realized.

For the pulse design, several optimization methods are available and have been widely used, such as greedy algorithm (GA) [54], gradient ascent pulse engineering (GRAPE) [55,56], and chopped random-basis optimization (CRAB) [57,58], while the latter two methods are based on gradient. These traditional methods have proven to be effective for the optimal control of lots of quantum systems. However, a major challenge with these methods is that they often converge to local optima instead of global maxima. Consequently, the

[*] Corresponding author: wangzhaoming@ouc.edu.cn

search may become stuck on a local maximum, leading to an insufficient fidelity. For gradient-based optimization algorithms, traps of local maxima can be avoided by considering a series of optimization processes with different trial control fields. Reference [59] employs a revised greedy (RG) algorithm to attain universal QSP with a high fidelity through the introduction of external interference, but it is less efficient because of trial and error at every step.

In this paper, we propose a stochastic prediction (SP) of neural network strategy, which can obtain reliable pulse sequences for high fidelity universal QSP. We use a large number of initial and target states to train the neural network and subsequently use the well-trained network to generate the pulse sequence, providing the control trajectory for state preparation. Our dataset solely consists of nonlocal optima, following the definition provided in Ref. [59]. A local optimum is defined as a scenario where the fidelity fails to improve compared to the previous step, indicating that the optimization process has reached a local maximum. During the state preparation process, we employ the knowledge blindness of the neural network to escape these local optima. When the network encounters such a situation, it randomly predicts an action as the introduction of external disturbance, allowing us to break free from the local optimum. RL specializes in addressing multistep tasks such as dynamic programming, which has been used to predict multiple impulses for QSP [52]. Although it shows that RL is effective to discern and navigate local optima, long optimization time is required. We propose a different route that can dynamically design control pulses. This approach simplifies the learning process and enhances the efficiency of the algorithm when compared to other methods. Our evaluation results show that our pulse design scheme is more efficient than traditional optimization methods in a discrete control space and higher fidelity can be obtained. Compared to conventional pulse optimization methods, our scheme jumps out of the local optimum by introducing disturbances through randomly predicted pulses, while improving the preparation efficiency. Our approach can be efficiently applied to small quantum systems with a limited number of qubits.

II. MODEL

Semiconductor quantum dots are promising candidates for quantum computing due to the advantage that they can be fully electrically driven [60]. Here, we describe the single-qubit and two-qubit models in S-T_0. The effective Hamiltonian of a single S-T_0 qubit controlled by an external electrical pulse is [19,61–64]

$$H = J\sigma_z + h\sigma_x, \tag{1}$$

under the computational basis states: spin singlet state $|0\rangle = |S\rangle = (|\uparrow\downarrow\rangle - |\downarrow\uparrow\rangle)/\sqrt{2}$, and spin triplet state $|1\rangle = |T_0\rangle = (|\uparrow\downarrow\rangle + |\downarrow\uparrow\rangle)/\sqrt{2}$. h is the Zeeman energy gap caused by magnetic field and it represents rotation around the x axis of the Bloch sphere. The exchange interaction J causes rotation around the z axis. h is not easy to be changed experimentally, we assume it to be a constant $h = 1$ [25]. The reduced Planck constant $\hbar = 1$ is assumed for simplicity throughout. Thus, the only controllable parameter is J between the two electrons, which can be adjusted by applying an external voltage. Due to the nature of the exchange coupling, J is finite and nonnegative [65], and these constraints allow the construction of composite pulses for the implementation of universal quantum gates.

Quantum information processing typically requires a two-qubit entanglement gate. In semiconductor DQDs, the Hamiltonian describing two entangled qubits based on Coulomb interactions can be expressed as follows [24,26,53,66–68]:

$$H_{2\text{-qubit}} = \frac{\hbar}{2}\{J_1(\sigma_z \otimes I) + J_2(I \otimes \sigma_z) + h_1(\sigma_x \otimes I) + h_2(I \otimes \sigma_x) + \frac{J_{12}}{2}[(\sigma_z - I) \otimes (\sigma_z - I)]\}, \tag{2}$$

under the basis states of $\{|SS\rangle, |ST_0\rangle, |T_0S\rangle, |T_0T_0\rangle\}$. J_i and h_i represent the exchange interaction and magnetic-field gradient across the double quantum dot, respectively, with the subscripts $i = 1, 2$ denoting the corresponding qubits. Experimentally, the coupling strength J_{12} between the qubits is proportional to J_1J_2, where both J_i values need to be positive. For simplicity, we set $J_{12} = J_1J_2/2$ and $h_1 = h_2 = 1$ as in Ref. [59]. To manipulate this two-qubit system, it is only necessary to control the electrical pulses that adjust J_1 and J_2.

III. METHODS

Now our task is to design discrete control pulses that can drive one arbitrary state to another arbitrary state. The pulse sequences are generated by the neural network. To optimize various parameters of the neural network, the scheme requires us to construct a dataset in advance. During the process of constructing composite pulses, the trained neural network can predict the appropriate pulse based on input that is not part of the training set. To reduce computational cost, the control pulses are discretized into the segmented constant function [56], with the maximum evolution time T uniformly divided into N segments and the pulse duration dt set as T/N. The fidelity F, which quantifies the distance between the evolution state and the target state, is used to assess the quality of the state preparation. $F = |\langle S_n|S_{\text{tar}}\rangle|^2$, where S_n denotes the evolution state at a time step of n, and S_{tar} represents the target state.

Our approach includes several steps: First, we construct a dataset with a large number of initial and target states and their corresponding actions, which is then used to put into a neural network for training. The trained network is saved as a model for use in future. Secondly, we begin by setting the initialization time step to step = 0 and feeding a pair of initial state S_{init} and target state S_{tar} into the network model to calculate the fidelity F of the initial state. And we define it as the maximum fidelity $F_{\max}$. After feature extraction in the fully connected layer, we obtain the output of the actions, which is a set of discrete actions output as a probability distribution under the activation function, and the sum of these probabilities is 1. We choose the best action $a_k = \text{argmax(action)}$, which represents the pulse strength $J(t)$. We use the word "action" to represent the intensity of the control pulse at each time step, which is similar as in the RL framework. Using the current quantum state S_{init} and the action obtained from the network prediction, we calculate the evolution state $S_n = \exp[-iH(a_k)dt]S_{\text{init}}$ and

Algorithm 1. The SP algorithm for designing control trajectory

```
Input: initial state $S_{\text{init}}$ and target state $S_{\text{tar}}$
Output: the maximum fidelity $F_{\max}$ and pulse sequence
    from step $= 0$ to step$_{\text{end}}$
Calculate the initial fidelity $F$ and let $F_{\max} = F$
Initialize the time step step $= 0$
while True do
    Feed initial and target states into the network model to
        predict the action probability distribution
    Choose the action $a_k = \text{argmax(action)}$
    Next state $S_{\text{step}}$ are the state obtained by performing $a_k$
        and calculate the corresponding fidelity $F_{\text{step}}$
    Compare the two fidelities $F_{\text{step}}$ and $F_{\max}$
    if $F_{\text{step}} > F_{\max}$ then
        Let $F_{\max} \leftarrow F_{\text{step}}$
    end if
    Let $S_{\text{init}} \leftarrow S_{\text{step}}$ and step $=$ step $+ 1$
    Break if $F_{\max} > 0.999$ or step $>$ step$_{\max}$
end while
```

its corresponding fidelity F_n for the next moment. Then we compare it with the previous maximum fidelity and select the larger of the two values as $F_{\max}$. The evolution state S_n is then fed into the network model as the new initial state with the target state at the time step step $=$ step $+ 1$. We repeat this process until either the time step reaches the maximum step N or the fidelity exceeds a satisfactory threshold. The control trajectory for the QSP consists of the sequence of actions predicted by the neural network, with this sequence representing the solution for obtaining the maximum of the fidelity. At last, the trained neural network can formulate appropriate control trajectories for quantum states in the test set or other states in the Hilbert space. The pulse design process is illustrated in Fig. 1. Algorithm 1 presents the pseudocode for the SP algorithm.

Figure 2 plots the process of data collection in detail. Taking a single qubit as an example, the data are uniformly sampled on the Bloch sphere to form a set of quantum states. This set of states is taken as both the initial and target quantum states. For a pair of initial and target states $|0\rangle$, $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, eight discrete actions $J \in [0, 1, 2, 3, 4, 5, 6, 7]$ are chosen as pulse intensities. The fidelity F_0 of the initial state is 0.5 initially. Executing these actions produces eight states with the fidelities 0.71, 0.65, 0.61, 0.56, 0.58, 0.62, 0.63, and 0.68. The maximum fidelity $F_1 = 0.71$ for this time step surpasses F_0. Then the initial state, target state, and corresponding action with F_1 are recorded in the dataset. Simultaneously, the state at the point of attaining the maximum fidelity serves as the new initial state for executing these actions, resulting in new states and corresponding fidelities 0.71, 0.67, 0.63, 0.59, 0.6, 0.64, 0.66, and 0.69. Notably, the new maximum fidelity $F_2 = 0.71$ does not exceed F_1, thus the data will not be included in the dataset in this step. The cycle continues, with the intermediate state corresponding to F_2 being the new initial state. These steps repeat until the fidelity exceeds a certain threshold or the step reaches the maximum limit. Finally the preparation data from the next initial state to the target state are collected and the dataset is constructed.

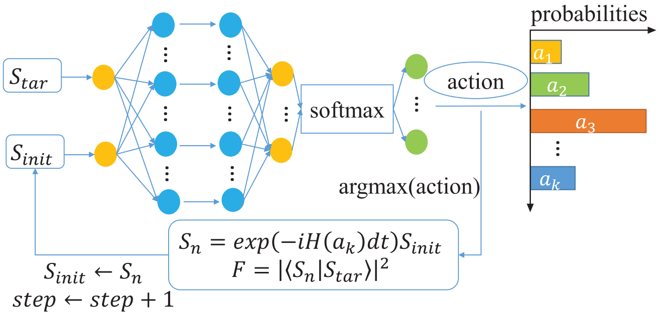

S_{tar}
S_{init}
softmax
action
probabilities
a_1
a_2
a_3
⋮
a_k
argmax(action)
$S_n = exp(-iH(a_k)dt)S_{init}$
$F = |\langle S_n|S_{tar}\rangle|^2$
$S_{init} \leftarrow S_n$
$step \leftarrow step + 1$

FIG. 1. Diagram of the stochastic prediction (SP) algorithm for designing control trajectory. The specifics of the algorithm are elaborated in Sec. III and the pseudocode is presented in Algorithm 1.

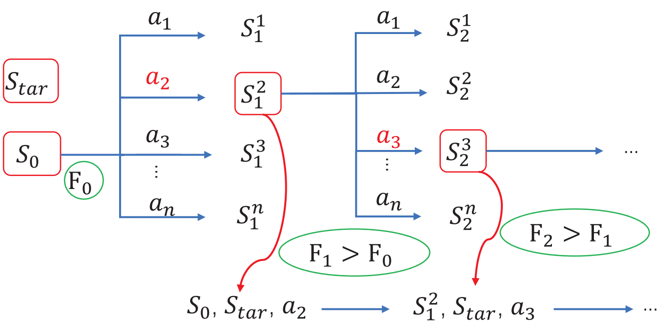


FIG. 2. The process of constructing the dataset.

During the process of dataset construction, we exclude data points that fall into local optima. We artificially define the characteristics of local optima, which means the maximum fidelity of the next step is not higher than the previous step. Consequently, the dataset solely comprises nonlocal optima, ensuring that the fidelity improves at each step.

Our scheme employs a classification mechanism of supervised learning to predict actions, effectively handling multistep tasks, as demonstrated in this paper where a multistep pulse is designed for state preparation. In contrast to the RG algorithm that relies on trial and error for exploring suitable actions, our approach directly and efficiently determines the next action. The training process of the neural network mirrors supervised learning, involving the construction of a dataset in advance. However, the dataset exclusively incorporates correct choices, with incorrect ones being excluded. In situations where the control process falls into a local optimum, randomly predicted errors can be introduced, serving as disruptions. This offers a chance for the control process to escape from the local optimum.

IV. RESULTS AND DISCUSSIONS

In this section, we focus on the state preparation of single-qubit and two-qubit states in semiconductor DQDs and compare our approach with conventional optimization methods. The details of the default parameters of the algorithm are listed in Table I.

TABLE I. Default parameters of the neural network.

Parameters	Single-qubit	Two-qubit
Total evolution time	4π	10π
Action duration	$\pi/5$	$\pi/2$
Maximum time step	20	20
Number of allowed actions	8	16
Batch size	64	128
Neurons per hidden layer	256/64/32/32/8	256/128/64/16
Learning rate	0.0005	0.001
Number of epochs	200	100
Activation function	softmax	softmax

A. Universal single-qubit state preparation

An arbitrary single-qubit state can be represented by a point on the Bloch sphere $|\psi(\theta,\varphi)\rangle = \cos(\frac{\theta}{2})|0\rangle + e^{i\varphi}\sin(\frac{\theta}{2})|1\rangle$, where the polar angle $\theta \in [0,\pi]$ and the azimuthal angle $\varphi \in [0,2\pi)$. We take the dataset for a single-qubit state preparation as in Ref. [59], where 128 testing points distributed uniformly at the angles θ and ϕ are sampled on the Bloch sphere. Each of these points is prepared in turn as a target state, enabling us to assess the performance of our method. For one preparation task, there is one fidelity $\bar{F}$. The mean of these average fidelities $\langle\bar{F}\rangle$ is calculated over all target states. For example, the single-qubit state preparation corresponds to $128 \times 127 = 16\,256$ tasks.

The state preparation can be achieved by performing successive rotations on a Bloch sphere, with the exchange coupling $J(t)$ as the only adjustable parameter [29]. In our approach, we use eight discrete control pulses, $J \in \{0, 1, 2, 3, 4, 5, 6, 7\}$. The total evolution time T is set to 4π, and the pulse duration dt is set to $\pi/5$, resulting in a maximum allowed time step of $N = T/dt = 20$ for the entire process. These parameters can be adjusted as required.

Figure 3 plots the test set accuracy, the average fidelity of the SP algorithm, and the average fidelity of the GA versus the number of epochs during the neural network training. Classical algorithms such as the GA are not involved in the training process, so the average fidelity of the GA is a constant. Figure 3 shows that after about 75 epochs, the test set accuracy of the network and the average fidelity of the SP algorithm do not improve significantly as the number of epoch increases, indicating that the network has converged and the two trends are consistent. Furthermore, the average fidelity of the SP algorithm is significantly better than that of the GA, demonstrating that our proposed scheme for preparing quantum states produces higher-quality results. Therefore, we conclude that our approach is viable, and the trained network can be applied to universal QSP tasks.

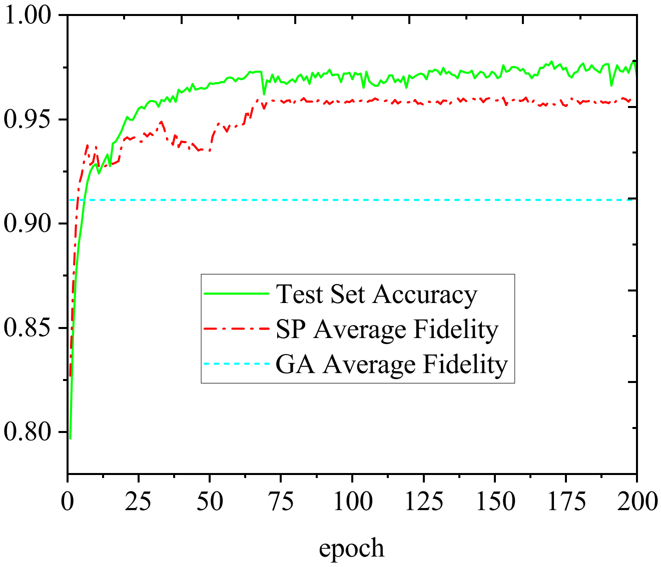


FIG. 3. The mean of all average fidelities $\langle\bar{F}\rangle$ of two algorithms and test set accuracy as the functions of the number of epochs in the training process for single-qubit preparation.

High efficiency QSP means high fidelity with short design time. To evaluate the efficiency of our SP algorithm against other methods, we present the distribution of the average fidelity $\bar{F}$ versus the average designing time $\bar{t}$ of the SP, GRAPE, CRAB, GA, and RG for preparing target states in Fig. 4. The control parameters are taken as the same as in Fig. 3. The average is based on the 128 state preparation tasks. To satisfy the discrete control requirement, we discretize the continuous control of GRAPE and CRAB to the nearest allowable action at the end of the execution [49]. As shown in Fig. 4, our SP algorithm outperforms all the other four conventional optimization algorithms in terms of efficiency in the discrete control space, with GRAPE and CRAB algorithms performing poorly in the same space. During optimization, the SP algorithm reduces the required time step adaptively to efficiently find the optimal solution. In contrast, GRAPE and CRAB use a fixed number of time steps and sometimes the optimal solution is missed. These two gradient-based methods require a prolonged optimization process and give better results under conditions of continuous control.

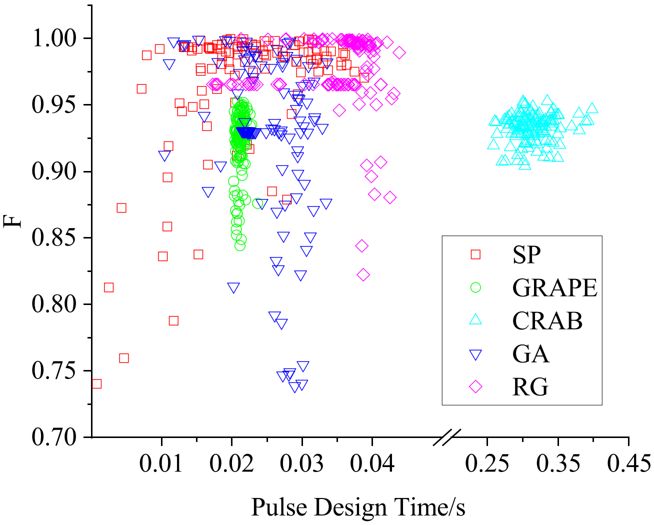


FIG. 4. The distribution of average fidelities $\bar{F}$ vs average design time $\bar{t}$ for the preparation of arbitrary single-qubit target states using various optimization algorithms, based on 128 sampled tasks. $\langle\bar{F}\rangle = 0.97$, 0.9121, 0.9117, 0.9206, and 0.97 and $\langle\bar{t}\rangle = 0.0211$, 0.0212, 0.3142, 0.0246, and 0.0347 with SP, GRAPE, CRAB, GA, and RG, respectively. $\langle\bar{F}\rangle$ and $\langle\bar{t}\rangle$ represent the mean of all average fidelities and all average pulse design times over 128 preparation tasks.

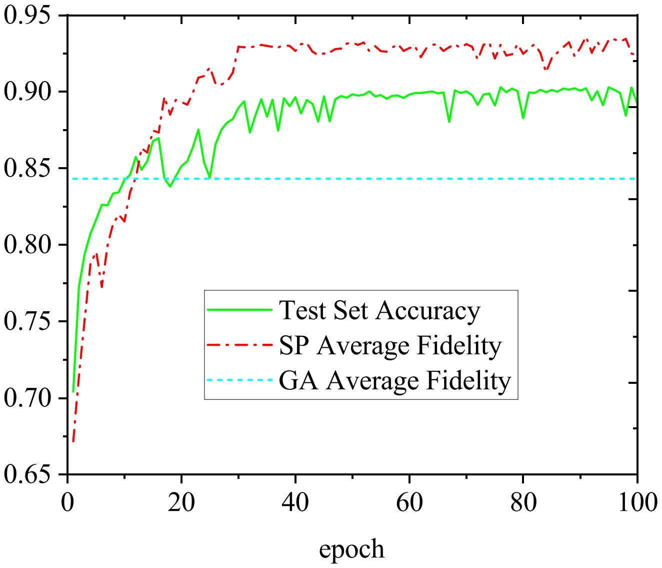


FIG. 5. The mean of the average fidelities $\langle \bar{F} \rangle$ of two algorithms and test set accuracy as the functions of the number of epochs in the training process for two-qubit preparation.

B. Universal two-qubit state preparation

For a two-qubit state preparation of a semiconductor DQD, the allowed control pulses for each qubit can be discretized as $\{(J_1, J_2)|J_1, J_2 \in \{1, 2, 3, 4\}\}$, resulting in a total of 16 allowed actions. During this process, the total evolution time is set to $T = 10\pi$ and the pulse duration is set to $dt = \pi/2$. The points in the data set for train and test are defined as $\{[a_1, a_2, a_3, a_4]^T\}$, where $a_j = e^{i\phi}c_j$ represents the probability amplitude of the corresponding jth basis state, and $\phi \in \{0, \pi/2, \pi, 3\pi/2\}$, and these $c_j s$ together represent the points on the hypersphere of the four-dimensional unit

$$\begin{cases} c_1 = \cos\theta_1, \\ c_2 = \sin\theta_1 \cos\theta_2, \\ c_3 = \sin\theta_1 \sin\theta_2 \cos\theta_3, \\ c_4 = \sin\theta_1 \sin\theta_2 \sin\theta_3, \end{cases} \tag{3}$$

with $\theta_i \in \{\pi/8, \pi/4, 3\pi/8\}$ [52]. We select randomly 256 testing points to form the data set.

As plotted in Fig. 5, the neural network converges after about 30 epochs. After 100 epochs of training, the average fidelity of the SP algorithm converges to 0.93. On the other hand, average fidelity of our proposed algorithm for the two-qubit state preparation still performs better than the GA. Figure 6 shows the frequency distribution of the average fidelity $\bar{F}$ for the 512 target states prepared by SP, GA, and RG, respectively. The results again verify that our algorithm outperforms the other two algorithms. Although some bad spots exist, the overall performance is excellent.

C. Universal state preparation in a noisy environment

The complete quantumness is always expected when performing universal QSP. However, normally the quantum noise will destroy the quantumness and then decrease the fidelity. How do the optimal pulse sequences perform when considering the noises? Next we introduce noises in the quantum line by adding the bit flip channel, phase flip channel, or amplitude damping channel, respectively. The bit flip and phase flip channels are modeled by applying an additional X or Z gate to the qubit with a probability of occurrence. These two noise channels are the so-called Pauli channels. We take the bit flip channel as an example, which can be expressed as

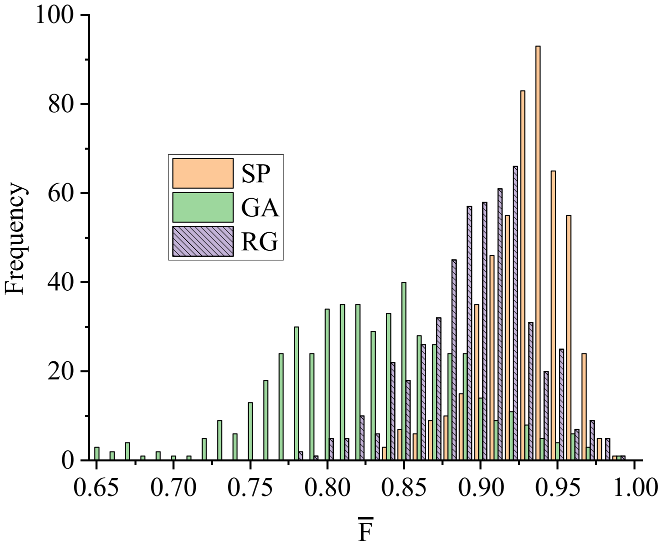


FIG. 6. The frequency distributions of average fidelities $\bar{F}$ for two-qubit preparation over 512 preparation tasks. The mean of all average fidelities $\langle \bar{F} \rangle = 0.9295$, 0.8381, and 0.8962 with SP, GA, and RG.

$$\epsilon(\rho) = (1 - p)I\rho I + pX\rho X, \tag{4}$$

where I is the unit matrix and X is the Pauli X gate. The corresponding Kraus operators for this channel are

$$E_0 = \sqrt{1-p}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad E_1 = \sqrt{p}\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \tag{5}$$

where p is the probability of occurrence of bit flip. The amplitude damping channel accounts for the dissipation of energy from the quantum system and the mathematical form can be expressed as

$$\epsilon(\rho) = E_0\rho E_0^\dagger + E_1\rho E_1^\dagger, \tag{6}$$

with Kraus operators

$$E_0 = \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{1-p} \end{bmatrix}, \quad E_1 = \begin{bmatrix} 0 & \sqrt{p} \\ 0 & 0 \end{bmatrix}, \tag{7}$$

where p is the dissipation factor.

We have found the ideal optimal pulse sequence that corresponds to the maximum fidelity through training the dataset in the absence of noises. For the noise model, we consider two cases. The first is that we use the ideal pulse sequence. We apply the noise channel after each time step of the pulse to create a noisy quantum line, which is used to drive the initial state to the final state. For the second case, we directly search the optimal pulse in the noise model. Now for the first cases, in Fig. 7 we plot the fidelity $\langle \bar{F} \rangle$ as a function of the occurrence probability (dissipation factor) p for single-qubit and two-qubit state preparation. For two-qubit state preparation,

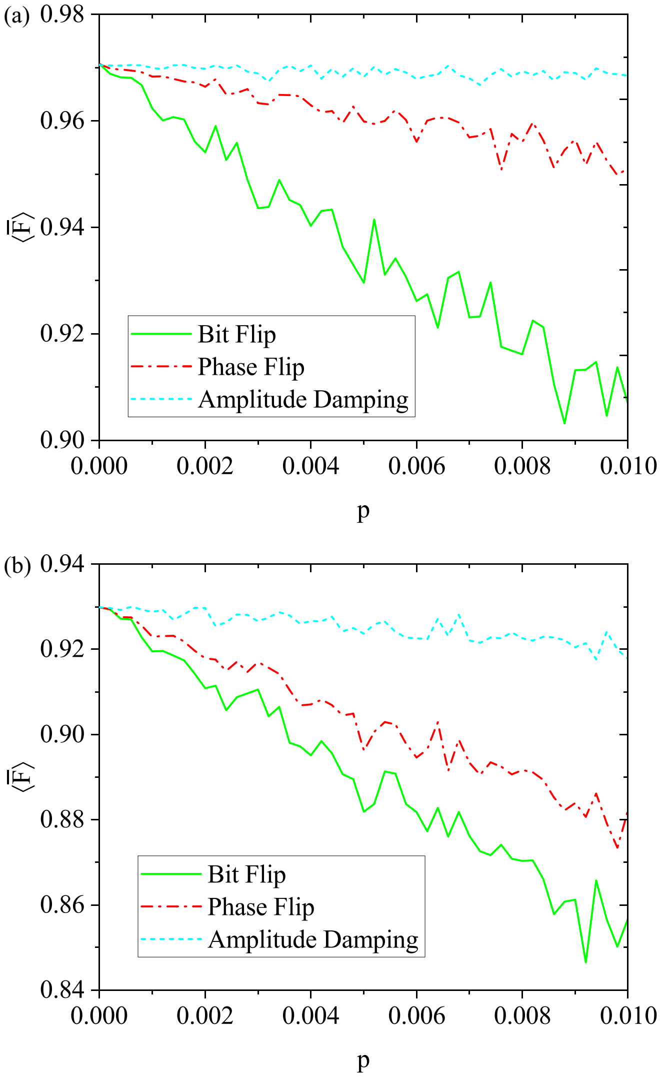


FIG. 7. The mean of average fidelities with the SP algorithm vs occurrence probability or dissipation factor of different noise quantum channels. (a) Single-qubit state preparation. (b) Two-qubit state preparation.

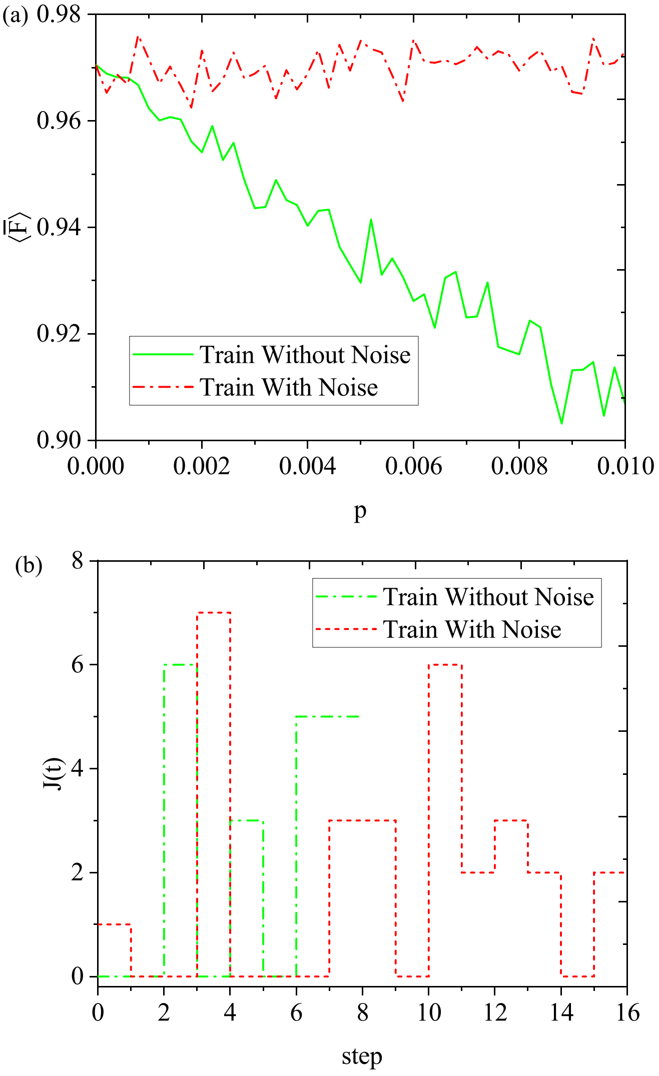
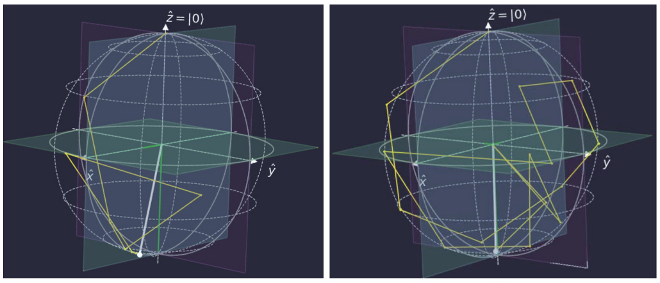


FIG. 8. (a) The mean of average fidelities using the SP algorithm vs occurrence probability of the bit flip channel with and without noise in the train process. (b) The designed control trajectory for these two cases with occurrence probability $p = 0.005$. The initial state and target state are set as $|0\rangle$ and $|1\rangle$.

we assume that both qubit noise channels are identical, and the probability or dissipation factor of the noise channels is the same ($p_1 = p_2 = p$). $\langle \bar{F} \rangle$ decreases with increasing p as expected. For single- and two-qubit cases and for the same p, $\langle \bar{F} \rangle$ decreases most significantly for the bit flip, phase flip is in the middle, and amplitude damping corresponds to the minimal impact.

It has been clearly shown that the fidelity will decrease in the presence of noise even with the ideal pulse sequence. Can we use our method to directly design the pulse sequence with noises? In this case, the optimal pulse sequence depends not only on the system but also on the noises. In a recent paper [69], optimized pulse sequences for the adiabatic speedup are found by using stochastic search procedures in a noisy environment. The detrimental effects of the environment on the system are reduced for the optimal pulses compared with the ideal closed-system pulses. Now we will train the network with noises. Initially, we add a noise channel on the Hamiltonian and construct a data set incorporating the noises. Then we obtain a model after training. For the demonstration, we take a single-qubit QSP with bit flip channel as an example. The initial state is taken as $|0\rangle$ and the target state is taken as $|1\rangle$. Figure 8(a) plots the mean of the average fidelity versus the flip probability for these two cases: train with (without) noise. Obviously, when using the ideal pulse sequences, $\langle \bar{F} \rangle$ decreases quickly with increasing p as expected. However, when training with noises, $\langle \bar{F} \rangle$ is almost stable for different p. The value of $\langle \bar{F} \rangle$ oscillates around 0.965. This result indicates that once the environmental parameter is given, the neural network is able to adjust its weight parameters appropriately during the training to combat the noises. Figure 8(b) shows the obtained control trajectories using the SP algorithm training with and without noises. These two trajectories are different from both the steps and the strengths. It only needs eight steps when training without noise and 16 steps are required with



(a) (b)

FIG. 9. The corresponding motion trail for the reset task on the Bloch sphere. (a) Train without noise, with the final fidelity $F = 0.9752$. (b) Train with noise, with the final fidelity $F = 0.9858$.

noises. Furthermore, we plot the corresponding motion trail of Fig. 8(b) for the reset task from $|0\rangle$ to $|1\rangle$ on the Bloch sphere in Fig. 9.

Given the limitations of available quantum computing, we simulate quantum computing on a classical computer and generate the corresponding data. Our algorithm was implemented using PYTHON 3.8.8, TENSORFLOW 2.12.0, QUTIP 4.7.1, and MINDQUANTUM 0.8.0, running on a computer with a six-core 2.60-GHz CPU and 16 GB of RAM.

V. CONCLUSIONS

In this paper, we propose an efficient SP algorithm for designing control trajectories that can prepare an arbitrary state from an arbitrary state. The scheme involves training a large number of initial and target states along with their corresponding actions using a neural network. Once the network is trained, it can be used to predict control pulses without any further training. We demonstrate the efficacy of our approach on single-qubit and two-qubit states of semiconductor quantum dots, highlighting its potential application in the future quantum computation. We construct a dataset composed of exclusively nonlocal optima. In instances where the control process falls into a local optimum, disturbances can be introduced by randomly predicting errors, providing an opportunity to escape the local optimum. Our SP algorithm shows its advantage over conventional pulse optimization algorithms by achieving a higher fidelity. Furthermore, the control pulses are predicted directly through the network, resulting in shorter pulse design time than other numerical optimization algorithms. At last, we consider noises including the Pauli channel and amplitude damping channel. We find that our SP algorithm is still effective for the design of pulse sequences when training with noises. Our investigation shows that machine learning is a powerful tool for the design of control pulse sequences in quantum information processing.

It is reasonable to question whether the effectiveness of the algorithm can be extended to complex systems. Our algorithm is suitable for simple tasks characterized by shallow local optima, but it might be challenged when encountering complex tasks. The prediction approach based on supervised learning may result in a limited number of classes that it can accurately categorize. Moreover, our algorithm is specifically applicable when a strong similarity exists among objects to be recognized, such as predicting the correct action based on quantum states. For scenarios such as dynamic scene changes during the control process or the existence of a policy link between the front and back steps, the algorithm will lose its effectiveness. For the complexity of the control, discrete actions with limited numbers are required in our scheme.

ACKNOWLEDGMENTS

This paper is based upon work supported by the Natural Science Foundation of Shandong Province (Grant No. ZR2021LLZ004), Fundamental Research Funds for the Central Universities (Grant No. 202364008), and the Young Scientists Fund of the National Natural Science Foundation of China (Grant No. 62002349).

[1] S. J. Glaser, U. Boscain, T. Calarco, C. P. Koch, W. Köckenberger, R. Kosloff, I. Kuprov, B. Luy, S. Schirmer, T. Schulte-Herbrüggen *et al.*, Eur. Phys. J. D **69**, 1 (2015).

[2] C. P. Koch, U. Boscain, T. Calarco, G. Dirr, S. Filipp, S. J. Glaser, R. Kosloff, S. Montangero, T. Schulte-Herbrüggen, D. Sugny *et al.*, EPJ Quantum Technol. **9**, 19 (2022).

[3] L. M. Vandersypen and I. L. Chuang, Rev. Mod. Phys. **76**, 1037 (2005).

[4] P. Richerme, Z.-X. Gong, A. Lee, C. Senko, J. Smith, M. Foss-Feig, S. Michalakis, A. V. Gorshkov, and C. Monroe, Nature (London) **511**, 198 (2014).

[5] M.-H. Yung, J. Casanova, A. Mezzacapo, J. Mcclean, L. Lamata, A. Aspuru-Guzik, and E. Solano, Sci. Rep. **4**, 3589 (2014).

[6] M. H. Devoret and R. J. Schoelkopf, Science **339**, 1169 (2013).

[7] G. Wendin, Rep. Prog. Phys. **80**, 106001 (2017).

[8] L. Childress and R. Hanson, MRS Bull. **38**, 134 (2013).

[9] D. M. Zajac, A. J. Sigillito, M. Russ, F. Borjans, J. M. Taylor, G. Burkard, and J. R. Petta, Science **359**, 439 (2018).

[10] W. Huang, C. Yang, K. Chan, T. Tanttu, B. Hensen, R. Leon, M. Fogarty, J. Hwang, F. Hudson, K. M. Itoh *et al.*, Nature (London) **569**, 532 (2019).

[11] T. Watson, S. Philips, E. Kawakami, D. Ward, P. Scarlino, M. Veldhorst, D. Savage, M. Lagally, M. Friesen, S. Coppersmith *et al.*, Nature (London) **555**, 633 (2018).

[12] W. Jang, M.-K. Cho, J. Kim, H. Chung, V. Umansky, and D. Kim, Appl. Phys. Lett. **117**, 234001 (2020).

[13] R. Hanson, L. P. Kouwenhoven, J. R. Petta, S. Tarucha, and L. M. K. Vandersypen, Rev. Mod. Phys. **79**, 1217 (2007).

[14] M. A. Eriksson, M. Friesen, S. N. Coppersmith, R. Joynt, L. J. Klein, K. Slinker, C. Tahan, P. Mooney, J. Chu, and S. Koester, Quant. Info. Proc. **3**, 133 (2004).

[15] F. A. Zwanenburg, A. S. Dzurak, A. Morello, M. Y. Simmons, L. C. L. Hollenberg, G. Klimeck, S. Rogge, S. N. Coppersmith, and M. A. Eriksson, Rev. Mod. Phys. **85**, 961 (2013).

[16] D. Kim, Z. Shi, C. Simmons, D. Ward, J. Prance, T. S. Koh, J. K. Gamble, D. Savage, M. Lagally, M. Friesen *et al.*, Nature (London) **511**, 70 (2014).

[17] E. Kawakami, T. Jullien, P. Scarlino, D. R. Ward, D. E. Savage, M. G. Lagally, V. V. Dobrovitski, M. Friesen, S. N. Coppersmith, M. A. Eriksson *et al.*, Proc. Natl. Acad. Sci. USA **113**, 11738 (2016).

[18] J. T. Muhonen, J. P. Dehollain, A. Laucht, F. E. Hudson, R. Kalra, T. Sekiguchi, K. M. Itoh, D. N. Jamieson, J. C. McCallum, A. S. Dzurak *et al.*, Nat. Nanotechnol. **9**, 986 (2014).

[19] B. M. Maune, M. G. Borselli, B. Huang, T. D. Ladd, P. W. Deelman, K. S. Holabird, A. A. Kiselev, I. Alvarado-Rodriguez, R. S. Ross, A. E. Schmitz *et al.*, Nature (London) **481**, 344 (2012).

[20] H. Bluhm, S. Foletti, I. Neder, M. Rudner, D. Mahalu, V. Umansky, and A. Yacoby, Nat. Phys. **7**, 109 (2011).

[21] C. Barthel, J. Medford, C. M. Marcus, M. P. Hanson, and A. C. Gossard, Phys. Rev. Lett. **105**, 266808 (2010).

[22] J. J. Pla, K. Y. Tan, J. P. Dehollain, W. H. Lim, J. J. Morton, F. A. Zwanenburg, D. N. Jamieson, A. S. Dzurak, and A. Morello, Nature (London) **496**, 334 (2013).

[23] X. Wang, L. S. Bishop, E. Barnes, J. P. Kestner, and S. Das Sarma, Phys. Rev. A **89**, 022310 (2014).

[24] J. Taylor, H.-A. Engel, W. Dür, A. Yacoby, C. Marcus, P. Zoller, and M. Lukin, Nat. Phys. **1**, 177 (2005).

[25] X. Wu, D. R. Ward, J. Prance, D. Kim, J. K. Gamble, R. Mohr, Z. Shi, D. Savage, M. Lagally, M. Friesen *et al.*, Proc. Natl. Acad. Sci. USA **111**, 11938 (2014).

[26] J. M. Nichol, L. A. Orona, S. P. Harvey, S. Fallahi, G. C. Gardner, M. J. Manfra, and A. Yacoby, npj Quantum Inf. **3**, 3 (2017).

[27] M. A. Nielsen and I. L. Chuang, Phys. Today(2) **54**, 60 (2001).

[28] Z. An and D. Zhou, Europhys. Lett. **126**, 60002 (2019).

[29] R. E. Throckmorton, C. Zhang, X.-C. Yang, X. Wang, E. Barnes, and S. Das Sarma, Phys. Rev. B **96**, 195424 (2017).

[30] X. Wang, L. S. Bishop, J. Kestner, E. Barnes, K. Sun, and S. Das Sarma, Nat. Commun. **3**, 997 (2012).

[31] D. F. Pinto, M. S. Zanetti, M. L. W. Basso, and J. Maziero, Phys. Rev. A **107**, 022411 (2023).

[32] M. S. Zanetti, D. F. Pinto, M. L. Basso, and J. Maziero, J. Phys. B **56**, 115501 (2023).

[33] X.-C. Yang, M.-H. Yung, and X. Wang, Phys. Rev. A **97**, 042324 (2018).

[34] J. Heaton, Genet. Program. Evolvable Mach. **19**, 305 (2018).

[35] X.-M. Zhang, Z.-W. Cui, X. Wang, and M.-H. Yung, Phys. Rev. A **97**, 052333 (2018).

[36] X. Yang, R. Liu, J. Li, and X. Peng, Phys. Rev. A **102**, 012614 (2020).

[37] J. Lin, Z. Y. Lai, and X. Li, Phys. Rev. A **101**, 052327 (2020).

[38] M. Bukov, Phys. Rev. B **98**, 224305 (2018).

[39] X. Kong, L. Zhou, Z. Li, Z. Yang, B. Qiu, X. Wu, F. Shi, and J. Du, npj Quantum Inf. **6**, 79 (2020).

[40] A. M. Palmieri, E. Kovlakov, F. Bianchi, D. Yudin, S. Straupe, J. D. Biamonte, and S. Kulik, npj Quantum Inf. **6**, 20 (2020).

[41] Z. T. Wang, Y. Ashida, and M. Ueda, Phys. Rev. Lett. **125**, 100401 (2020).

[42] M. Y. Niu, S. Boixo, V. N. Smelyanskiy, and H. Neven, npj Quantum Inf. **5**, 33 (2019).

[43] A. Gratsea, F. Metz, and T. Busch, J. Phys. A **53**, 445306 (2020).

[44] H. Ma, D. Dong, S. X. Ding, and C. Chen, IEEE Trans. Neural Netw. Learn. Syst. **34**, 8852 (2022).

[45] M. I. Jordan and T. M. Mitchell, Science **349**, 255 (2015).

[46] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot *et al.*, Nature (London) **529**, 484 (2016).

[47] G. Carleo and M. Troyer, Science **355**, 602 (2017).

[48] G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby, L. Vogt-Maranto, and L. Zdeborová, Rev. Mod. Phys. **91**, 045002 (2019).

[49] X.-M. Zhang, Z. Wei, R. Asad, X.-C. Yang, and X. Wang, npj Quantum Inf. **5**, 85 (2019).

[50] L. Giannelli, P. Sgroi, J. Brown, G. S. Paraoanu, M. Paternostro, E. Paladino, and G. Falci, Phys. Lett. A **434**, 128054 (2022).

[51] Z. An, H.-J. Song, Q.-K. He, and D. L. Zhou, Phys. Rev. A **103**, 012404 (2021).

[52] R.-H. He, R. Wang, S.-S. Nie, J. Wu, J.-H. Zhang, and Z.-M. Wang, EPJ Quantum Technol. **8**, 29 (2021).

[53] T. Haug, W.-K. Mok, J.-B. You, W. Zhang, C. E. Png, and L.-C. Kwek, Mach. Learn.: Sci. Technol. **2**, 01LT02 (2021).

[54] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms* (MIT, Cambridge, MA, 2022).

[55] N. Khaneja, T. Reiss, C. Kehlet, T. Schulte-Herbrüggen, and S. J. Glaser, J. Magn. Reson. **172**, 296 (2005).

[56] B. Rowland and J. A. Jones, Phil. Trans. R. Soc. A **370**, 4636 (2012).

[57] P. Doria, T. Calarco, and S. Montangero, Phys. Rev. Lett. **106**, 190501 (2011).

[58] T. Caneva, T. Calarco, and S. Montangero, Phys. Rev. A **84**, 022326 (2011).

[59] R.-H. He, H.-D. Liu, S.-B. Wang, J. Wu, S.-S. Nie, and Z.-M. Wang, Quantum Sci. Technol. **6**, 045021 (2021).

[60] X. Zhang, H.-O. Li, K. Wang, G. Cao, M. Xiao, and G.-P. Guo, Chin. Phys. B **27**, 020305 (2018).

[61] J. R. Petta, A. C. Johnson, J. M. Taylor, E. A. Laird, A. Yacoby, M. D. Lukin, C. M. Marcus, M. P. Hanson, and A. C. Gossard, Science **309**, 2180 (2005).

[62] J. Levy, Phys. Rev. Lett. **89**, 147902 (2002).

[63] F. K. Malinowski, F. Martins, P. D. Nissen, E. Barnes, Ł. Cywiński, M. S. Rudner, S. Fallahi, G. C. Gardner, M. J. Manfra, C. M. Marcus *et al.*, Nat. Nanotechnol. **12**, 16 (2017).

[64] S. Foletti, H. Bluhm, D. Mahalu, V. Umansky, and A. Yacoby, Nat. Phys. **5**, 903 (2009).

[65] X. Zhang, H.-O. Li, G. Cao, M. Xiao, G.-C. Guo, and G.-P. Guo, Natl. Sci. Rev. **6**, 32 (2019).

[66] M. D. Shulman, O. E. Dial, S. P. Harvey, H. Bluhm, V. Umansky, and A. Yacoby, Science **336**, 202 (2012).

[67] X. Wang, E. Barnes, and S. Das Sarma, npj Quantum Inf. **1**, 15003 (2015).

[68] I. Van Weperen, B. D. Armstrong, E. A. Laird, J. Medford, C. M. Marcus, M. P. Hanson, and A. C. Gossard, Phys. Rev. Lett. **107**, 030506 (2011).

[69] Y.-Y. Xie, F.-H. Ren, R.-H. He, A. Ablimit, and Z.-M. Wang, Phys. Rev. A **106**, 062612 (2022).