Discovery of an exchange-only gate sequence for CNOT with record-low gate time using reinforcement learning

Violeta N. Ivanova-Rohling,[1, 2, †] Niklas Rohling,[1, ‡] and Guido Burkard[1, *]

[1]*Department of Physics, University of Konstanz, D-78457 Konstanz, Germany*
[2]*Zukunftskolleg, University of Konstanz, D-78457 Konstanz, Germany*

Exchange-only quantum computation is a version of spin-based quantum computation that entirely avoids the difficulty of controlling individual spins by a magnetic field and instead functions by sequences of exchange pulses. The challenge for exchange-only quantum computation is to find short sequences that generate the required logical quantum gates. A reduction of the total gate time of such synthesized quantum gates can help to minimize the effects of decoherence and control errors during the gate operation and thus increase the total gate fidelity. We apply reinforcement learning to the optimization of exchange-gate sequences realizing the CNOT and CZ two-qubit gates which lend themselves to the construction of universal gate sets for quantum computation. We obtain a significant improvement regarding the total gate time compared to previously known results.

I. INTRODUCTION

Quantum computing has been a strongly growing field in the last years due to its potential to solve certain problems efficiently that are hard on a classical computer. The growth of the field is driven by advances in gate fidelity and the number of qubits for scalable quantum computing platforms. Among those platforms are superconducting qubits [1, 2], Rydberg atoms [3], trapped ions [4], and spin qubits in semiconductor quantum dots [5]. Specifically, spin qubits are promising with respect to scaling due to their small size and synergy with silicon-based technology and due to recent advances in gate fidelity [6–8]. In the original spin-qubit setting [9], each qubit is represented by the spin of an electron or hole trapped in a semiconductor quantum dot. Computations in such single-electron or single-hole spin qubits are based on controlling the exchange interaction between the spins and the local time-dependent magnetic fields acting on individual spins. A logical two-qubit universal gate, such as the controlled-NOT (CNOT) gate, is implemented by using exchange-interaction-based SWAP^α operations controlled by inter-dot voltage combined with magnetic-field-controlled single-qubit gates. These physical SWAP^α gates are the result of the exchange operation, where α is the normalized time parameter for which the exchange interaction is pulsed and the SWAP gate is switched on. This means that α denotes the gate time in units of π/J where J is the strength of the exchange interactions when switched on. One of the challenges for quantum computing based on spins in quantum dots is the single-spin control that necessitates the modulation of a strongly non-homogeneous magnetic field on short time scales or the realization of a strongly inhomogeneous magnetic field using on-chip micro-magnets [10]. The necessity for this is completely avoided in an alterna-

tive approach which encodes one logical qubit using three physical spins. For this encoding, the exchange interaction is sufficient to implement universal quantum gates [11, 12], and thus the control of the local magnetic field is no longer necessary. This paradigm of quantum computation is referred to as spin *exchange-only* computation and has been subject to great experimental advances recently [13].

Various approaches to exchange-only computation exist, described in [14]. This physical platform has since been theoretically and experimentally investigated, and a large number of practical implementations of quantum dot systems for three-spin qubits have been developed, for more detail refer to [5].

We will consider the exchange-only computational model, described in Ref. [11], where each qubit is encoded using three physical spins (spin-$\frac{1}{2}$ particles), and where one- and two-qubit quantum gates on the logical qubits are implemented by sequences of SWAP^α gates (switching on and off the exchange interaction between pairs of spin particles) applied to the physical qubits. The exchange interaction can be completely switched off by a sufficiently large voltage barrier between the quantum dots (then the gate is off), and only through pulsing the voltage, the exchange interaction is switched on.

The cost of disposing of single-spin rotations in exchange-only quantum computation is twofold, (1) the necessity of an extended physical system, i.e., a larger number of physical spins to represent the qubit register, and (2) logical quantum gates that need to be synthesized by a sequence of several physical exchange operations, rather than a single application of exchange in the standard spin qubit paradigm. More specifically, in the case of two-qubit gates, several applications of the exchange interaction between at least five pairs of spins are involved in controlling the system of two logical qubits, see Fig. 1. A vital aspect for exchange-only computation to be practically relevant is thus to optimize the efficiency of the gates needed for quantum computation. This is where this paper provides a novel method that is shown to allow for a substantial improvement by applying re-

† violeta.ivanova-rohling@uni-konstanz.de
‡ niklas.rohling@uni-konstanz.de
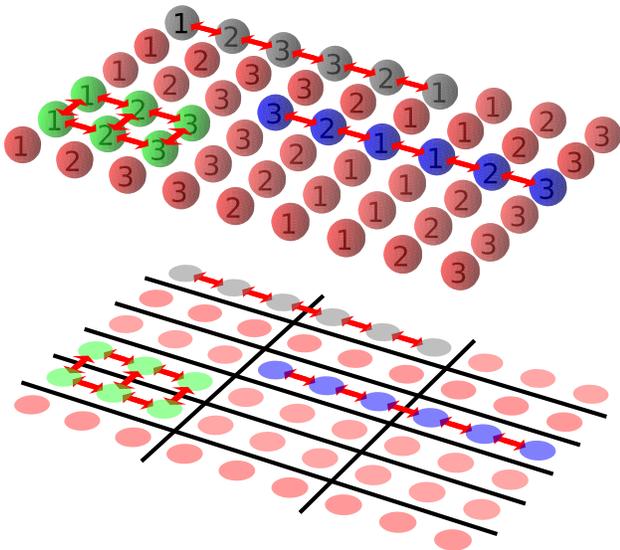* guido.burkard@uni-konstanz.de

FIG. 1. A possible two-dimensional arrangement of physical spins where each rectangle with three spins (labeled 1,2,3) defines one logical qubit. The definition of the logical qubit depends on the order of the physical qubit because each logical qubit, $a|0\rangle_l + b|1\rangle_l$ is defined for the total-spin-0 subspace as $|0\rangle_l = (|010\rangle - |100\rangle)/\sqrt{2}$, $|1\rangle_l = (2|001\rangle - |010\rangle - |100\rangle)/\sqrt{6}$, for the total-spin-1 subspace, see [15]. Here, $|ijk\rangle = |i\rangle_1|j\rangle_2|k\rangle_3$ where the numbers 1,2, and 3, correspond to the labels in the figure. We clearly see that the definition depends on the order of the physical qubits (spins), specifically the spins labeled 1 and 2 host a singlet in state $|0\rangle_l$ and a triplet in state $|1\rangle_l$. Further, note that the coupling of neighboring logical qubits depends on the arrangement and on the order of the physical qubits (within one row or within neighboring rows). This results in three distinct coupling scenarios between pairs of logical qubits (highlighted in gray, blue, and green) which we label "33" (gray), "11" (blue), and 2D (green) which are all considered in this paper. The exchange couplings that can be used in each scenario are indicated with red arrows.

inforcement learning (RL). Efficiency can be looked at in several terms: minimizing the number of pulses, time steps, or total time necessary. In this paper, we will focus on minimizing the total gate time for a fixed value J of the exchange coupling when switched on. We will consider both, the case of exchange gates applied in parallel when possible and exchange gates applied sequentially which can be advantageous for avoiding cross-talk [13]. Minimizing the time needed for a desired gate is beneficial with respect to gate fidelity as noise acts on the system for a shorter time while the gate sequence is performed. For pulses of fixed duration with varied exchange strength, as in [13], the actual time needed for the sequence will depend only on the number of pulses (sequential) or number of time steps for pulses applied in parallel. However, we note that there is a lower limit to the gate time set by the maximum available value of J. Additionally, minimizing the total time normalized to

a fixed J [16] then corresponds to operating at smaller exchange coupling which can reduce charge noise [13].

Moreover, when arranging the physical qubits in a two-dimensional square lattice, different connections between neighboring logical qubits are present, see Fig. 1. Each of these different arrangements yields a distinct optimization problem. In quantum computing, a CNOT gate is universal when combined with single-qubit gates, which makes finding (efficient) exchange-only sequences to realize the CNOT gate an important problem. In [11], the first exchange-only universal gate set consisting of single-qubit rotations and a CNOT was presented. In [17], an exact specification of a universal logical gate-set using four spins to encode a single qubit was presented. The authors use extensive numerical optimization in order to obtain an optimized CNOT gate sequence with 27 parallel nearest-neighbor exchange interactions or 50 serial gates.

Different approaches have been utilized to find optimized sequences numerically [11, 15, 17]. The sequence for a CNOT found by Fong and Wandzura, via the use of genetic algorithms [15], see Fig. 2, is currently the most efficient exact CNOT sequence known when the physical qubits are connected via nearest-neighbor interactions and they are in a linear chain architecture, see the area in Fig. 1 highlighted in blue. Importantly, despite the fact that this solution has been discovered numerically, it has a precise analytical description. In [18], gate sequences were found for logical two-qubit gate locally equivalent to CNOT for various connectivities by applying exhaustive search under the condition that all exchange gates are $\sqrt{\text{SWAP}}$ or products thereof. Aside from a large number of purely numerical approaches, it has been possible to come up with an analytic derivation of the optimal Fong-Wandzura (FW) CNOT sequence [19]. Furthermore, analytical considerations with regards to *leakage* were utilized in combination with numerics [20] to simplify the search problem and construct another set of gate sequences realizing the CNOT gate. Under certain assumptions, the solution presented in Ref. [20] is more efficient than the FW sequence if one considers total time as the efficiency criterion. Other efficient universal two-qubit gates have also been investigated, such as a gate locally equivalent to the CPHASE gate [21] that is potentially valuable in the currently available NISQ quantum devices. Leakage errors in exchange-only spin qubits can be approached by a reset-if-leaked procedure and, via numerical optimization, by a leakage correcting gate sequence [22].

Numerous advances in implementing quantum dot systems for three-spin qubits have been made [23–31]. Recently, Weinstein *et al.* [13] presented a two-qubit exchange-only system implemented using an array of six ^{28}Si/SiGe quantum dots to achieve universal gates of very high operational fidelity. The fidelity of universal control of two encoded qubits was evaluated to be $96.3\% \pm 0.7\%$ for encoded CNOT operations, and even higher ($99.3\% \pm 0.5\%$) for encoded SWAP, demonstrat-
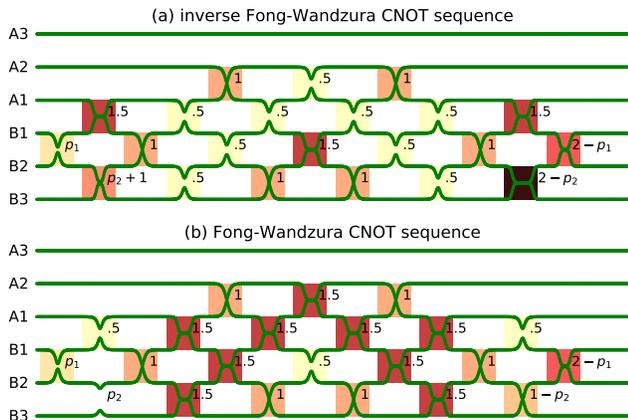
FIG. 2. (a) Inverse FW sequence compared to (b) the FW sequence from Ref. [15]. Both sequences require twenty-two pulse and 13 time steps, while their parallel times are $T_p = 13.89\,\pi/J$ and $T_p = 15.89\,\pi/J$, respectively, and their sequential times are $T_s = 20$ and $T_s = 24$, respectively. Logical qubits A and B are arranged as shown. Numbers after A and B label the physical qubits. The SWAP powers α are displayed explicitly in the gates. $\alpha = 1$ corresponds to a full SWAP operation, up to a global phase. Here, we have introduced $p_1 = \arccos(-1/\sqrt{3})/\pi$ and $p_2 = \arcsin(1/3)/\pi$. The representation of the SWAP$^\alpha$ gates in this figure is inspired by the representation in [13].

ing substantial progress towards achieving fault tolerance and computational acceleration with this approach.

The problem of optimizing gate sequences for two-qubit logical gates is high-dimensional. In our work, we use an intelligent optimization [32] approach enhanced by an RL algorithm, suitable for continuous search spaces. This allows us to explore a vastly larger search space by enforcing much fewer assumptions on the optimization problem in comparison to [18]. We aim to optimize the total time of the exchange-only gate sequences representing exact CNOT and exact CZ gates with varying connection topologies and find gate sequences for all three arrangements shown in Fig. 1. For the linear "11" arrangement highlighted in blue in Fig. 1, we find gate sequences representing CNOT gates. Notably one of the sequences we find, presented in Fig. 2 (a), has a shorter total time than the original FW sequence and the RL approach found it from scratch. We discuss the relation to other known gate sequences in Sec. IV.

Importantly, we demonstrate the usefulness of a reinforcement-learning-based approach for optimizing exchange-only sequences, which can be seamlessly extended to optimize different universal gates, and gate sequences with different architectures and different types of exchange interactions by simply redefining the cost function. The main aspects of the reinforcement learning approach we use for gate sequence optimization are visualized in Figure 3, and full details of the approach are given in Sec. II, as well as in Appendix A.

Additionally, we apply the RL algorithm to find optimized CZ gate sequences, see Sec. B, CNOT sequences for a linear arrangement with the singlet-triplet qubit part on the edges ("33" arrangement, highlighted in gray in Fig. 1) and obtain a sequence beating the one actually implemented in [13] with respect to sequential total gate time, see Sec. II E and Appendix C for details. Furthermore, we search for optimized gate sequences for CNOT gates in the 2D arrangement of spins, highlighted in green in Fig. 1, where seven pairs of spins can be coupled by exchange interactions, see Sec. II F and Appendix D.

II. METHODS

A. Reinforcement learning for optimization problems

Reinforcement learning is a class of machine learning algorithms, where an agent interacts with an environment and gets back a reward based on its actions. The goal of the agent is to learn a behavior that optimizes the total reward obtained. RL that uses neural networks as agents to learn the optimal policy is referred to as *deep RL*. Recently, RL, and especially deep RL have been used with great success for numerous problems in various areas of physics, in general, [33], as well as quantum computing [34], in particular. More recently, RL has been used to learn appropriate optimizers that solve difficult optimization problems, or to *learn to optimize*, examples include [35–40].

The RL approaches to optimization show advantages over automating and accelerating the optimization of complicated problems. Instead of manually crafting classical optimizers, one can parameterize and learn optimization rules in a data-driven fashion.

Yet another application of RL for optimization is to use the RL agent as a hybrid aspect of the optimizer to automatically guide the behavior of the optimizer in an intelligent way, suitable in particular for the problem at hand. This does not involve "learning" to optimize on a similar task prior to the optimization task, but using the machinery of RL, and the stored experiences during the optimization procedure (the experience replay [41]), to select the appropriate next steps in the optimization search. Based on the agent's prior experience and obtained reward, the next optimization behaviors are selected, instead of encoded, such as selecting exploration vs. exploitation behaviors, or parameter values. Examples include [42, 43], where different global optimization heuristics were combined with a simple Q-learning approach to intelligently choose between exploitation or exploration behavior of the heuristic, as well as intelligently set other parameters of the optimization heuristics. These intelligent optimization approaches were tested on known hard mathematical functions as benchmarks and were found to outperform other state-of-the-art methods that were not enhanced by RL. In [44], a review of hybrid approaches
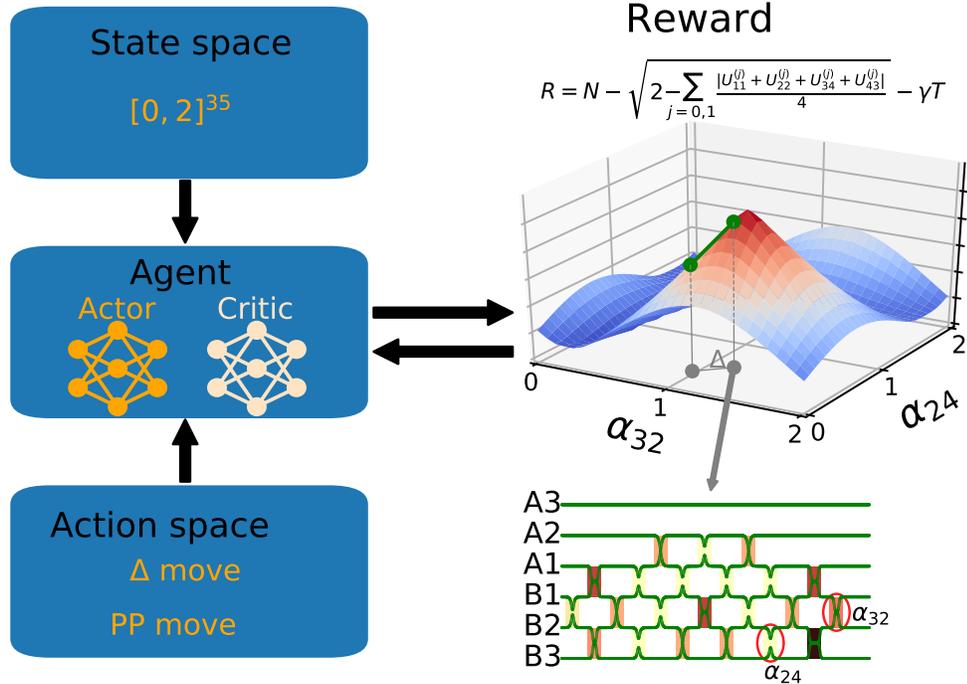
FIG. 3. The reinforcement learning (RL) algorithm used for CNOT gate sequence optimization. A deep deterministic policy gradient (DDPG) actor-critic algorithm was used as an agent. The state space is defined as 35 normalized time parameters $\alpha_i \in [0, 2]$ of SWAP^{α_i} gates, operating on six spins A_k and B_k ($k = 1, 2, 3$) representing two logical qubits A and B, and arranged in the five-brick structure. Each state corresponds to a sequence of exchange pulses with a length of up to 14 time steps using parallelism. The number of pulses can be smaller than 35 when some of the α_i are zero. The action space is defined as two possible actions – a Δ step, that potentially changes all parameters α_i, and a PP-move, i.e., a partial Powell local optimization step, where a local optimization with a fixed number of iterations is performed. The number of iterations is small ranging from 0 (no local optimization is performed) to 12. The reward is based on the distance to the CNOT gate used by Fong and Wandzura [15] and the total time, which can be either the time $T = T_p$ for applying the SWAP^{α_i} gates in parallel where possible, or the time $T = T_s$ for applying the SWAP^{α_i} gates sequentially.

for optimization, that use RL as well as metaheuristics for combinatorial optimization is presented. In [45], a memetic particle swarm optimizer that uses RL to control optimization operations, related to choosing local search behavior and particle selection, was introduced. The method turned out to be successful on several benchmark optimization problems. In this work, we use a deep RL approach to intelligently guide an optimizer to better optimize a gate sequence. The RL agent, based on previous experience, recorded in an experience buffer, and on previous rewards from the environment, predicts the optimality of an action. In this case, the action is a behavior of the optimizer.

B.   The five-brick structure

In the search for the reset-if-leaked sequence [22], a brick-like structure of repeated patterns of physical exchange gates (SWAP-α) was used. The brick structure is taking advantage of the commutation relation between the exchange interactions between the qubits in different subsystems. Two exchange interaction operators commute if the exchange interaction is applied to pairs of logical qubits that do not share any common spins. Then the gate sequence is invariant under the interchange of the order of these operators. Here, since we are not trying to reproduce the FW gate sequence, but aim to improve it, we loosen the four-brick pattern structure to a general five-brick structure (Fig. 4) that allows all six physical qubits to be affected by the sequence, which in principle enables a generalized search for other, potentially better sequences.
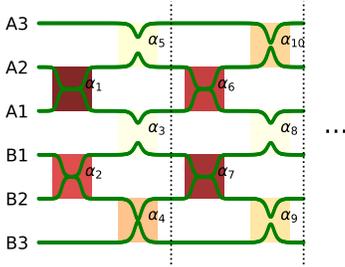
FIG. 4. The five-brick structure used to define the observation space of the RL algorithm.

C. Reward function for the CNOT-gate

In order to assess how well a gate sequence approximates a logical CNOT, the distance from CNOT is measured using the FW distance function introduced in [15],

$$
d_{\mathrm{FW}}(U(\boldsymbol{\alpha})) = \left[2 - \frac{1}{4} \left| U_{11}^{(0)} + U_{22}^{(0)} + U_{34}^{(0)} + U_{43}^{(0)} \right| \right.
$$
$$
\left. - \frac{1}{4} \left| U_{11}^{(1)} + U_{22}^{(1)} + U_{34}^{(1)} + U_{43}^{(1)} \right| \right]^{1/2}, \quad (1)
$$

where $U^{(0/1)}$ is the $\boldsymbol{\alpha}$-dependent unitary matrix describing the overall gate sequence on the subspace for total spin zero or total spin one, respectively. Here, $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots)$ represents the list of exchange time parameters. The function d_{FW} is a distance measure between a unitary matrix and the desired CNOT gate, taking advantage of the fact that CNOT as the target gate comprises only ones and zeros as matrix elements in the computational basis. Furthermore, note that while the unitarity of $U^{(0/1)}$ is used, d_{FW} allows for different phase factors in the spin-0 and in the spin-1 subspaces. The reward is given by

$$
R(\boldsymbol{\alpha}) = N - d_{\mathrm{FW}}(U) - \gamma T, \quad (2)
$$

where T denotes the total gate time. The last term rewards minimizing the total time needed for the gate sequence. N is a large number added for technical reasons, as negative rewards do not perform well.

D. Reinforcement learning for gate sequence optimization

Reinforcement learning (RL) can use the memory hash that is built from the learning experience in order to achieve intelligent optimization. The reward feedback, provided from the environment in the RL setting can improve the optimizer's behavior, and instead of choosing parameters of the optimization heuristics manually, the RL machinery can be used to guide the optimizer parameters in high-reward areas, with the actor-critic used to learn to predict the behavior of the optimizer that will optimize the reward.

A visual representation of our RL approach is shown in Figure 3, where the observation space consists of the possible values for the normalized times α_i for gate sequences of fixed length 35, the action space consists of two types of actions, a small change of the normalized times of the sequence, and a partial local optimization (the derivative-free Powell's method) of a fixed small number of iterations. The RL agent learns the best way to optimize the total time of the sequence in an actor-critic approach, where both the actor and the critic are neural networks. The reward obtained by the agent at each step is based on the sequence distance to the exact CNOT or CZ gate and the total time of the gate sequence. To optimize the gate sequence, given the difficulty of the problem, we utilize RL to learn strategies to optimize the sequence, instead of manually selecting and parametrizing an optimizer. We use the deep deterministic policy gradient (DDPG) [46] algorithm, which is an actor-critic algorithm [47] for RL with continuous state space, where the gate sequence is assumed to be constructed by a sequence of repeating the five-brick structure of a fixed length of 35 pulses, and the state space consists of the values of the normalized times α_i of the different SWAP$^{\alpha_i}$ gates, $i = 0, \ldots, 34$. We use hybrid control, namely the action space has both continuous as well as discrete components. The continuous components are values that change the normalized times α_i at a given step, while the discrete component determines the number of iterations of a partial derivative-free optimization (partial use of Powell's method, [48]). The number of possible iterations can be 0, which allows for the case where no derivative-free partial optimization takes place, and only the values of the normalized times are varied. By *partial optimization* here we mean that we fix the number of optimization iterations without the necessity of a local optimum to be achieved. The goal is to learn a sequence of parameter values (starting points of the partial Powell algorithm) that result in the best gate sequence. As a reward we use a function based on the FW measure for the distance from CNOT combined with the total time, see Eq. (2).

E. Optimizing T_s for CNOT with linear "33" arrangement

We investigate the performance of our approach for optimizing the CNOT gate sequence, imposing the same constraints on qubit arrangement as in [13] in order to be able to compare the resulting gate sequences to the one used in Weinstein *et al.* [13]. We enforce connectivity constraints of the physical qubits so that the singlet-triplet part of the logical qubit is on the outside of the gate sequence chain. This yields the order of the linear-chain arrangement highlighted in gray in Fig. 1 which is A1,

A2, A3, B3, B2, B1 (the "33" arrangement). This allows us to compare solutions discovered by our approach to the solution expressed in [13], where such additional requirement was imposed. We present the results in Sec. III.

F.   2D Connecting topology

In addition to the linear arrangement, we also consider the case where the three physical qubits of one logical qubit are coupled to their counterparts of the other logical qubit, see the qubits highlighted in green in Fig. 1. This requires an adjustment of the grouping of the exchange gates and complicates the optimization procedure, see Sec. III for the results.

III.   RESULTS

Multiple exact CNOT gate sequences of 14 time steps were discovered using the RL algorithm for gate sequence optimization. Most of the discovered sequences comprise 14 time steps, but several, including the FW gate sequence and the improved FW gate sequences, required only 13 time steps. The total times T_s and T_p for performing the exchange pulses sequentially and in parallelized form, respectively, of some of the discovered CNOT gate sequences are plotted as a function of the used training steps in Fig. 5. We find that with an increasing number of training steps, the solution discovered by the algorithm improves. The best solution, discovered by the algorithm has shorter total sequential and parallel times than the original sequence published by Fong and Wandzura in [15].

In addition to the results for the CNOT gate, the RL algorithm also produced several exact CZ sequences of length 14 time steps discovered by the RL algorithm, see Fig. 7. The dotted lines correspond to the total times required for the parallel and sequential operation of the CZ gate sequence described in [13]. The shortest sequence has total times $T_p = 11.5\,\pi/J$ and $T_s = 16.0\,\pi/J$, respectively, for parallel and sequential execution. This sequence (shown in Fig. 6) is equivalent to the CZ gate described in [13]. As the number of RL training steps increases, the corresponding best solutions discovered by the algorithm improve in efficiency. For details on the results for the CZ gate sequences, see Appendix B.

We also use the RL algorithm to optimize the CNOT gate with a different connecting topology. We again discover many exact solutions of a length of seven five-brick blocks, however, the best solution we discover is with a total sequential time $T_s = 20.4\,\pi/J$, and total parallel time $T_p = 15\,\pi/J$. Again the efficiency of the discovered gates depends on the number of training steps of the RL algorithm. For details on the results for the 2D topology, see Appendix D.

Finally, we also optimize a sequential CNOT gate sequence for the linear arrangement with singlet-triplet

pairs at the edges (the "33" arrangement shown in gray in Fig. 1). Under these constraints, we again discover multiple exact CNOT gates of various efficiencies. In this situation, we only evaluate the sequential total time T_s. The results are shown in Fig. 8 in the Appendix. Importantly, with our RL approach, we rediscover the gate sequence used in [13]. However, in addition, we discover a few solutions that are more efficient than the sequence in [13] with respect to total sequential time T_s. The best solution is shown in Fig. 9. This solution is identical to the Weinstein CNOT sequence with respect to the locally-equivalent part but with optimized local gates at the beginning and at the end of the sequence. The efficiency of the discovered CNOT gates again heavily depends on the training steps of the algorithm. For more details see Appendix C.

IV.   DISCUSSION OF THE RESULTS

For the linear arrangement with singlet-triplet pairs at the inside of the chain ("11" arrangement), our RL approach finds the realization of an exact CNOT gate which improves previously published results regarding the the total gate times T_s and T_p. For the arrangement with singlet-triplet pairs on the outside of a chain ("33" arrangement) as in [13] we found a sequence with reduced total sequential time T_s compared to the one implemented in [13]. These results demonstrate the power of RL applied to the optimization of quantum gates and quantum gate sequences.

We observed that some of the solutions for the CNOT gate sequence found by our RL approach are related to each other by symmetry operations. Those symmetries are presented in Appendix E in the form of mathematical lemmas. Importantly, explicitly implementing these symmetries in the future can boost the performance of the optimization strategy. These operations themselves are not difficult to understand and are implicitly used already in the literature at least partially, given that what we refer to as an 'inverse' FW sequence (or the non-local part of it) is also referred to as 'FW sequence' [18]. In general, the term 'FW sequence' refers to different sequences in the literature [13, 18, 21], see Fig. 2, which can be either an explicit CNOT sequence as in the original work by Fong and Wandzura [15] or locally equivalent [18] and which can be either using the same SWAP$^\alpha$ gates presented in [15] as in [21] or the inverse operations [13, 18]. Remarkably, our RL approach found both versions from scratch for the exact CNOT for the same linear arrangement of physical spin qubits as in [15], i.e., the chain highlighted in blue in Fig. 1, see Sec. II for the details of our optimization procedure. We note that the CNOT sequence presented here in Fig. 2 (a) requires a shorter gate time than the original FW sequence [15] and in contrast to the one presented in [18], it provides the full CNOT sequence rather than a sequence which is locally equivalent to CNOT. We further note that we did not im-
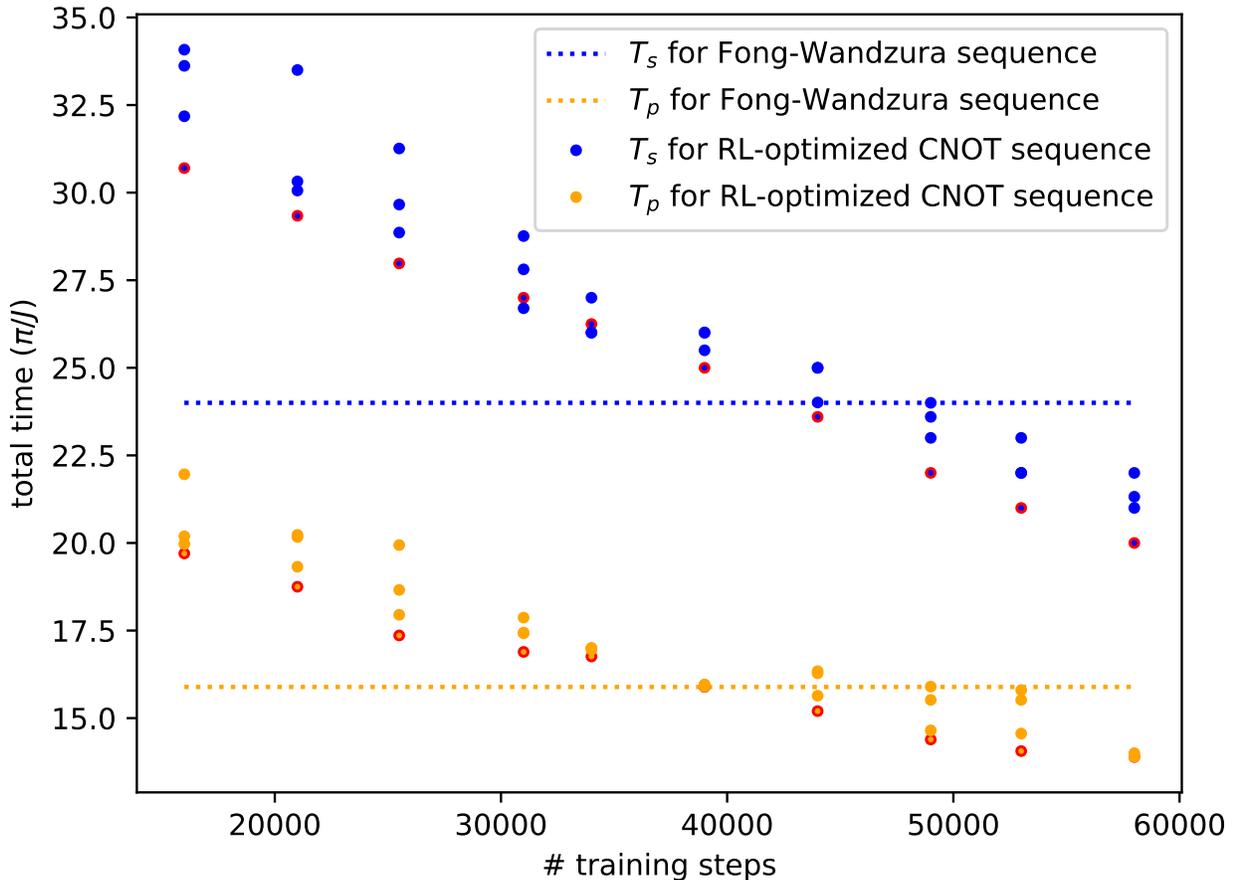
FIG. 5. Improvement of the total sequential time $T = T_s$ (in blue) and total parallel time $T = T_p$ (in orange) of the CNOT gate sequences discovered by the RL algorithm depending on the number of passed training steps (filled circles). The total gate time T in units of π/J, where J is the constant strength of the exchange interaction, is plotted as a function of the number of training steps. The solutions with optimal T_p for each investigated number of training steps are highlighted with a red circumference. Note that optimal T_s does not necessarily lead to optimal T_p. For comparison, we show T_p (T_s) of the FW gate sequence as an orange (blue) dotted horizontal line. The RL procedure is used as described in Sec. II D with a reward described in Eq. (2). We find that after about 45000 training steps, the sequences obtained from the algorithm trained using RL become more time-efficient than known sequences.

pose any restrictions on the values of α for the SWAP$^\alpha$ gates in contrast to [18]. While a restriction to $\sqrt{\text{SWAP}}$ gates and products as made in [18] cannot yield an exact CNOT gate, it can, however, provide a gate sequence that is locally equivalent to the CNOT or CZ gate. The independence from such restrictions on the gate sequence demonstrates the flexibility of our approach. Regarding the more complex optimization problem for the 2D connectivity, we note that while the RL algorithm can tackle also this problem, it is challenging to obtain a solution comparable in total time to the most efficient sequence for linear connectivity.

V. CONCLUSION

We have shown that machine learning and intelligent optimization through RL are working approaches for finding optimal exchange-only gate sequences. Specifically, we have discovered optimized solutions for a variety of gates and connectivities. In this work, RL has demonstrated its flexibility and usefulness in optimizing the total times of exchange-only CZ and CNOT gate sequences. The results demonstrate that RL helps finding such sequences and improves the total gate times of state-of-the-art solutions with fewer prior assumptions compared to other approaches. In the optimization problems considered in this work, we have used a brick (base)

structure that encodes the commutation relations of the exchange coupling. By enforcing a fixed connectivity we have turned the problem into a continuous optimization problem, for which efficient methods exist. We then use the RL as a tool for intelligent optimization that learns the appropriate starting points of a local optimizer. We find optimized solutions that are better or equivalent in terms of total times to known state-of-the-art solutions.

A limitation of our approach is the use of a fixed brick structure, which captures the commutation relations of operators but is not unique. Ideally, different brick structures could be used in optimization. For a more flexible approach, the symmetries that follow from the commutator relations can be encoded in an equivariant neural network, instead of using a fixed brick structure. This is meaningful also for other symmetries in the search space. Moreover, symmetries arising from the commutation relationships, as well as the other discovered symmetries, could be exploited by directly incorporating them in various ways in the optimization problem. As the approach is flexible, it allows for the investigation of different connection topologies in future work. Instead of optimizing the total gate time, the objectives could also be to minimize the number of exchange gates or the number of time steps. This might be particularly promising for gates other than CNOT and CZ. Additionally, one could extend the problem of optimal exchange-only gate sequences to a more realistic scenario where the gate fidelity is optimized in the e presence of state leakage and noise.

The RL-based approach presented here is by no means limited to spin exchange-only qubits. In contrast, it can be broadly applied for finding sequences for various quantum computing hardware platforms or for optimizing compiling sequences of quantum gates.

Appendix A: Reinforcement learning used for optimization

For all optimization problems investigated here, the RL algorithm used is the Deep Deterministic Policy Gradient algorithm (DDPG) [46] which uses the interaction between an actor network and a critic network to learn [47] a policy. Moreover, it is a model-free algorithm that does not impose a model or prior knowledge of the world.

The RL algorithm uses a deterministic policy gradient and can operate over continuous action spaces. We use the DDPG as implemented in the "stable baselines-3" package [49]. For the definition of the environment, we use the API provided by the package gym in OpenAI [50].

We use a gate sequence of a fixed number of 35 parameters, organized into seven blocks of five SWAP$^\alpha$ gates, i.e. bricks, of parameters (physical gate times), similar to the structure FW sequence but without restrictions on the individual exchange pulses. However, an optimal shorter gate sequence length can potentially be reached via this setting. Note that effectively the sequence is shorter if gate times were found to be zero. Observation, action space, and steps of the algorithm were implemented in OpenAI. The actor and critic networks used are standard fully connected multilayer perceptron (MLP) networks with three hidden layers, size 64. The following hyperparameter values [51] are used: an initial learning rate $\alpha_a = 0.0001$ of the actor, and initial learning rate $\alpha_c = 0.00001$ of the critic networks with a linear learning rate schedule of decrease, and a batch size of 256. A large buffer size of 5000000 allows for a large number of experiences to be stored. The observation space, describing the parameter values of the optimization search, is the 35-dimensional space $[0,2]^{35}$. The action space used is hybrid, which has a continuous component, as well as a discrete component, $[-0.4, 0.4]^{35} \otimes \{0, \ldots, 12\}$. Namely, at each step, the agent performs an action that is a change of the normalized times $\alpha_i, i = 0, \ldots, 34$, as well as a partial Powell optimization is performed with a possible number of iterations from 0 to 12, where 0 iterations means that no Powell derivative-free optimization is performed. The goal is for the algorithm to learn, starting from a random point, an appropriate sequence of parameter changes for the respective $\{\alpha_0, \ldots \alpha_{34}\}$ as well as appropriate Powell iteration values to navigate the search space.

Appendix B: Results for the CZ-gate

In order to adapt the algorithm for the discovery of the CZ gate, all we need is to redefine the reward function. The five-brick structure and the parametrization can be reused directly as defined in the search for optimal CNOT gate sequence. This demonstrates the flexibility of the approach. The quality measure, which evaluates a gate sequence's distance from the CZ gate is given by:

$$d_{CZ} = \left[2 - \frac{1}{4} \left| U_{11}^{(0)} + U_{22}^{(0)} + U_{33}^{(0)} - U_{44}^{(0)} \right| \right.$$
$$\left. - \frac{1}{4} \left| U_{11}^{(1)} + U_{22}^{(1)} + U_{33}^{(1)} - U_{44}^{(1)} \right| \right]^{1/2}. \quad \text{(B1)}$$

Accordingly, the reward is given by

$$R(\boldsymbol{\alpha}) = N - d_{CZ}(\boldsymbol{\alpha}) - \gamma T. \quad \text{(B2)}$$

In the following, we present the results found by our RL approach for an exact CZ gate, using the objective function from Eq. (B2). The results obtained are similar to the sequences found for CNOT, see Fig. 2. Again there are solutions which are related to each other by symmetry transformations, see Appendix E. The sequence in
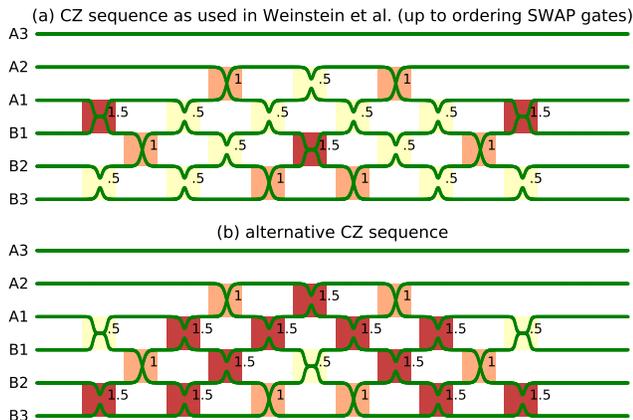
FIG. 6. (a) The optimal CZ sequence rediscovered by the RL algorithm equivalent to the CZ gate sequence in Weinstein *et al.* [13] (up to spin-order related SWAP gates) and up to local gates to the sequence in [18]. (b) Alternative CZ gate sequence also found by our RL approach and related to the original FW sequence [15] by altering the local operations on logical qubits A and B at the beginning and end of the sequence. The sequences shown in (a) and (b) are related to each other by 'inverting' the sequence, see Appendix E. They are equivalent in number of exchange pulses and number of time steps when using parallelism. Sequence (a) is superior in both, T_p and T_s.

Fig. 6 (b) is related to the original FW sequence [15] by changes only in the local gates for the A and for the B qubit at the beginning and at the end of the sequence. On the other hand, the sequence in Fig. 6 (a) is up to reordering of the spins-related SWAP gates identical to the CZ sequence from [13] where it is referred to as 'FW CZ' sequence.

The performance of the RL algorithm as a function of training steps is presented in Fig. 7. While the CNOT or CZ sequences can be generated from each other by padding with local single-qubit gates acting on the qubits A and B at the beginning and at the end of the sequence, the fact that our RL approach performs well for finding both, CNOT and CZ sequences from scratch is an important indication for the flexibility of the ansatz.

Appendix C: Optimizing gate sequences with fixed order gates to compare to the solution presented in [13]

The arrangement in Ref. [13] is such that the singlet-triplet pairs are on the ends of the chain of six spin qubits, as in the chain highlighted in gray in Fig. 1. This explains that the sequence implemented by Weinstein *et al.* has eight additional SWAP gates which are in some sense switching between two distinct linear orders (highlighted in blue and gray in Fig. 1). The total sequential time of the Weinstein CNOT sequence, see Fig. 9, is $26\pi/J$

assuming a constant exchange coupling J for each of the exchange gates in contrast to the actual implementation in [13], while the total time of the improved FW sequence together with eight ordering SWAP gates is $28\pi/J$. However, this is an unfair comparison, as some gates at the beginning and the end are shifted relative to the ordering SWAP gates to transform them in a more efficient way. In order to use the Weinstein CNOT sequence as a benchmark, we need to use our RL approach applied to their architecture and then minimize the sequential time.

To be able to fairly compare to the sequence discussed in [13], we set specific constraints for the gate sequence – sequential execution of the physical gates and we use the same pairs spins coupled by exchange gates as in [13]. RL was used to optimize a CNOT sequence that is sequential and with order gates using the form from [13]. We achieved sequences with better total sequential time than the one in [13], see Fig. 9.

Appendix D: 2D architecture optimization

In addition, we apply RL to optimize a two-dimensional (2D) topology where each spin is exchange-coupled to a spin of the other qubit. The constraints of the 2D arrangement, highlighted in green in Fig. 1, lead to a modification of the 5-brick structure used in the FW optimization, yielding a seven-component structure as shown in Fig 10. In each block, we first apply the exchange gates between the logical qubits A1 and A2 as well as B1 and B2 in parallel. Second, we apply the gates between A2 and A3 as well as between B2 and B3 in parallel. Finally, we apply the interactions J_{AjBj} with $j = 1, 2, 3$ in parallel. We find a gate sequence for CNOT with total time $T_s = 20\,\pi/J$ ($T_p = 15.89\,\pi/J$).

Appendix E: Theoretical derivation of observed symmetries

Among the solutions obtained by RL, we observe that some are related to each other by a symmetry operation that leaves the resulting logical quantum gate unaffected. We elaborate on such symmetry operations in the following, using some specific properties of the CNOT and CZ gates, namely that these unitaries are also Hermitian and that – in matrix form – they have real entries.

We start by reminding ourselves that switching the sign of the time a constant Hamiltonian is "switched on" translates from a unitary operation to its inverse:

Lemma E.1 *A quantum gate given by* $U(t) = \exp\left(\frac{-it}{\hbar}H\right)$ *with a time-independent Hamiltonian H, fulfills* $U^{-1}(t) = U(-t)$.

In the context described in this paper, unitary operations are given by a sequence of consecutive gates of the form
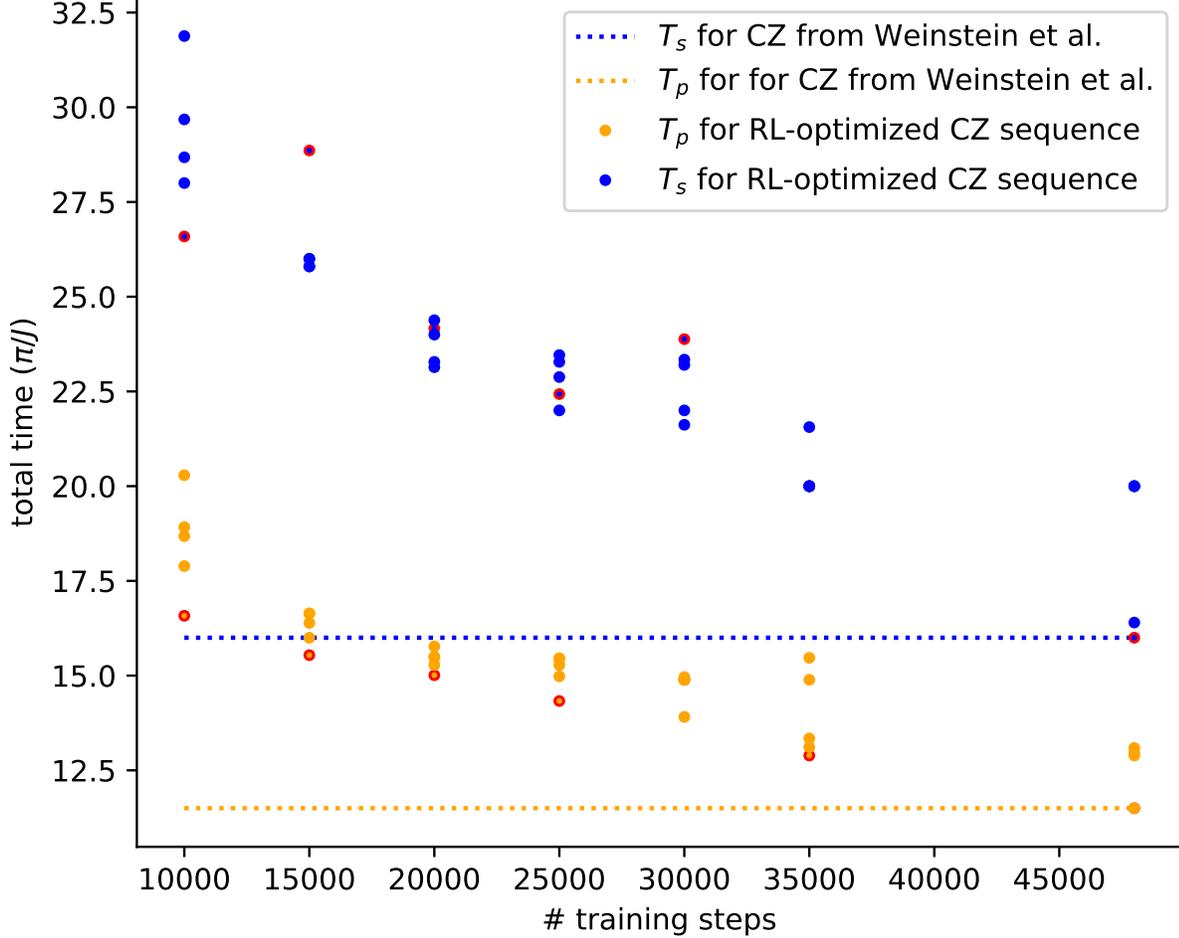
FIG. 7. Improvement of T_s (in blue) and T_p (in orange) of the CZ gate sequences units of π/J discovered by the RL algorithm depending on the number of training steps used. Further details, see Fig. 5.

given in the lemma above. This means constant Hamiltonians H_j are switched on for times t_j. We directly obtain the following statement about those sequences.

Lemma E.2 *For a gate sequence* $U(t_1, \ldots, t_n) = U_n(t_n) \cdots U_1(t_1)$ *with* $U_j(t_j) = \exp\left(\frac{-it_j}{\hbar} H_j\right)$, *the inverse unitary operation is given by the sequence* $[U(t_1, \ldots, t_n)]^{-1} = U_1(-t_1) \cdots U_n(-t_n)$.

Consequently, we can express a unitary operation which is Hermitian, like CNOT:

Lemma E.3 *For a gate sequence* $U(t_1, \ldots, t_n)$, *defined in the same way as in the lemma above with* $U^\dagger = U$, *the reverse sequence with inverted time arguments represents the same unitary,*

$$U_n(t_n) \cdots U_1(t_1) = U_1(-t_1) \cdots U_n(-t_n).$$

For sequences of palindromic structure, this yields:

Corollary E.3.1 *If a gate sequence is given by* $U(t_1, \ldots, t_k) = U_1(t_1) \cdots U_k(t_k) \cdots U_1(t_1)$, *i.e., it is of palindromic structure, and obeys* $U^\dagger = U$, *then the same unitary operation is represented by the sequence with negative time arguments,*

$$U(t_1, \ldots, t_k) = U(-t_1, \ldots, -t_k).$$

Note that the non-local part of the FW sequence for CNOT and also an exact CZ sequence are palindromic.

Now, we will use the properties of the distance measures d_{FW} and d_{CZ} and properties of the individual exchange gates. These individual gates for the exchange-only qubits are SWAP$^\alpha$ gates applied to two of the physical qubits. A SWAP$^\alpha$ gate is symmetric in the standard basis $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$ as well as in the representation used for computing the exchange-only sequences for

FIG. 8. Total times of the resulting CNOT gate sequences with fixed ordered gates discovered by the RL algorithm and the sufficient number of training steps, needed to obtain them.

FIG. 9. CNOT gate sequences for the linear arrangement with the singlet-triplet pairs on the outside (highlighted in gray in Fig. 1): The RL-optimized sequence of sequential time $T_s = 24\pi/J$ and the sequence from [13] of $T_s = 26\pi/J$.

the logical subsystem (5×5 for zero spin, 9×9 matrices for spin one), see Ref. [15]. From this it follows that a sequence of SWAP^α gates, $U(\alpha_0, \ldots, \alpha_{n-1}) = \mathrm{SWAP}^{\alpha_n}_{i_n j_n} \cdots \mathrm{SWAP}^{\alpha_1}_{i_1 j_1}$ fulfills

$$U(2 - \alpha_1, \ldots, 2 - \alpha_n) = [U(\alpha_1, \ldots, \alpha_n)]^*,$$

where $*$ denotes the complex conjugate (not the Hermitian conjugate).

FIG. 10. The seven-parameter block of quantum gates used for the physical qubits being arranged in a two-by-three rectangle with nearest-neighbor coupling. Note that the exchange gates between two spins where one belongs to logical qubit A and the other to logical qubit B (corresponding normalized gate times are α_5, α_6, α_7 or α_{12}, α_{13}, α_{14}) can be performed in parallel.

Note that CNOT has only real entries in the logical subsystem in the standard basis. Consequently, two gates which are complex conjugate to each other, have the same distance to CNOT. This holds for the Euclidean distance as well as for the FW loss function, d_{FW}.

Lemma E.4 *For sequences of* $\mathrm{SWAP}^\alpha_{ij}$ *gates,*

$$U(\alpha_1, \ldots, \alpha_n) = \mathrm{SWAP}^{\alpha_n}_{i_n j_n} \cdots \mathrm{SWAP}^{\alpha_1}_{i_1 j_1},$$

the following holds

$$d_{\mathrm{FW}}(U(\alpha_1, \ldots, \alpha_n)) = d_{\mathrm{FW}}(U(2 - \alpha_1, \ldots, 2 - \alpha_n)).$$

Note that the lemma above also holds for d_{CZ}. Further note that Lemma E.3 and Lemma E.4 applied to a CNOT (CZ) sequence yield two sequences also representing CNOT (CZ). Only if the sequence is of palindromic structure these two sequences are identical to each other, compare Corollary E.3.1.

[1] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. Brandao, D. A. Buell, *et al.*, Quantum supremacy using a programmable superconducting processor, Nature **574**, 505 (2019).

[2] M. Kjaergaard, M. E. Schwartz, J. Braumüller, P. Krantz, J. I.-J. Wang, S. Gustavsson, and W. D. Oliver, Superconducting qubits: Current state of play, Annual Review of Condensed Matter Physics **11**, 369 (2020).

[3] S. J. Evered, D. Bluvstein, M. Kalinowski, S. Ebadi, T. Manovitz, H. Zhou, S. H. Li, A. A. Geim, T. T. Wang, N. Maskara, *et al.*, High-fidelity parallel entangling gates on a neutral atom quantum computer, arXiv preprint arXiv:2304.05420 (2023).

[4] C. D. Bruzewicz, J. Chiaverini, R. McConnell, and J. M. Sage, Trapped-ion quantum computing: Progress and challenges, Applied Physics Reviews **6**, 021314 (2019).

[5] G. Burkard, T. D. Ladd, A. Pan, J. M. Nichol, and J. R. Petta, Semiconductor spin qubits, Rev. Mod. Phys. **95**, 025003 (2023).

[6] X. Xue, M. Russ, N. Samkharadze, B. Undseth, A. Sammak, G. Scappucci, and L. M. Vandersypen, Quantum logic with spin qubits crossing the surface code threshold, Nature **601**, 343 (2022).

[7] A. Noiri, K. Takeda, T. Nakajima, T. Kobayashi, A. Sammak, G. Scappucci, and S. Tarucha, Fast universal quantum gate above the fault-tolerance threshold in silicon, Nature **601**, 338 (2022).

[8] M. T. Madzik, S. Asaad, A. Youssry, B. Joecker, K. M. Rudinger, E. Nielsen, K. C. Young, T. J. Proctor, A. D. Baczewski, A. Laucht, *et al.*, Precision tomography of a three-qubit donor quantum processor in silicon, Nature **601**, 348 (2022).

[9] D. Loss and D. P. DiVincenzo, Quantum computation with quantum dots, Physical Review A **57**, 120 (1998).

[10] M. Pioro-Ladriere, T. Obata, Y. Tokura, Y.-S. Shin, T. Kubo, K. Yoshida, T. Taniyama, and S. Tarucha, Electrically driven single-electron spin resonance in a slanting zeeman field, Nature Physics **4**, 776 (2008).

[11] D. P. DiVincenzo, D. Bacon, J. Kempe, G. Burkard, and K. B. Whaley, Universal quantum computation with the exchange interaction, nature **408**, 339 (2000).

[12] D. Bacon, J. Kempe, D. A. Lidar, and K. B. Whaley, Universal fault-tolerant quantum computation on decoherence-free subspaces, Phys. Rev. Lett. **85**, 1758 (2000).

[13] A. J. Weinstein, M. D. Reed, A. M. Jones, R. W. Andrews, D. Barnes, J. Z. Blumoff, L. E. Euliss, K. Eng, B. H. Fong, S. D. Ha, *et al.*, Universal logic with encoded spin qubits in silicon, Nature **615**, 817 (2023).

[14] M. Russ and G. Burkard, Three-electron spin qubits, Journal of Physics: Condensed Matter **29**, 393001 (2017).

[15] B. H. Fong and S. M. Wandzura, Universal quantum computation and leakage reduction in the 3-qubit decoherence free subsystem, Quantum Info. Comput. **11**, 1003–1018 (2011).

[16] The quantity we refer to as normalized sequential time is termed *exchange angle* in [13].

[17] M. Hsieh, J. Kempe, S. Myrgren, and K. B. Whaley, An explicit universal gate-set for exchange-only quantum computation, Quantum Information Processing **2**, 289 (2003).

[18] F. Setiawan, H.-Y. Hui, J. P. Kestner, X. Wang, and S. D. Sarma, Robust two-qubit gates for exchange-coupled qubits, Phys. Rev. B **89**, 085314 (2014).

[19] D. Zeuch and N. Bonesteel, Simple derivation of the fong-wandzura pulse sequence, Physical Review A **93**, 010303 (2016).

[20] J. R. van Meter and E. Knill, Approximate exchange-only entangling gates for the three-spin-1/2 decoherence-free subsystem, Physical Review A **99**, 042331 (2019).

[21] D. Zeuch and N. Bonesteel, Efficient two-qubit pulse sequences beyond cnot, Physical Review B **102**, 075311 (2020).

[22] V. Langrock and D. P. DiVincenzo, A reset-if-leaked procedure for encoded spin qubits, arXiv preprint arXiv:2012.09517 10.48550/arXiv.2012.09517 (2020).

[23] E. A. Laird, J. M. Taylor, D. P. DiVincenzo, C. M. Marcus, M. P. Hanson, and A. C. Gossard, Coherent spin manipulation in an exchange-only qubit, Phys. Rev. B **82**, 075403 (2010).

[24] L. Gaudreau, G. Granger, A. Kam, G. Aers, S. Studenikin, P. Zawadzki, M. Pioro-Ladriere, Z. Wasilewski, and A. Sachrajda, Coherent control of three-spin states in a triple quantum dot, Nature Physics **8**, 54 (2012).

[25] J. Medford, J. Beil, J. Taylor, S. Bartlett, A. Doherty, E. Rashba, D. DiVincenzo, H. Lu, A. Gossard, and C. M. Marcus, Self-consistent measurement and state tomography of an exchange-only spin qubit, Nature nanotechnology **8**, 654 (2013).

[26] J. Medford, J. Beil, J. Taylor, E. Rashba, H. Lu, A. Gossard, and C. M. Marcus, Quantum-dot-based resonant exchange qubit, Physical review letters **111**, 050501 (2013).

[27] D. Kim, Z. Shi, C. Simmons, D. Ward, J. Prance, T. S. Koh, J. K. Gamble, D. Savage, M. Lagally, M. Friesen, *et al.*, Quantum control and process tomography of a semiconductor quantum dot hybrid qubit, Nature **511**, 70 (2014).

[28] K. Eng, T. D. Ladd, A. Smith, M. G. Borselli, A. A. Kiselev, B. H. Fong, K. S. Holabird, T. M. Hazard, B. Huang, P. W. Deelman, *et al.*, Isotopically enhanced triple-quantum-dot qubit, Science advances **1**, e1500214 (2015).

[29] M. Reed, B. Maune, R. Andrews, M. Borselli, K. Eng, M. Jura, A. Kiselev, T. Ladd, S. Merkel, I. Milosavljevic, *et al.*, Reduced sensitivity to charge noise in semiconductor spin qubits via symmetric operation, Physical review letters **116**, 110402 (2016).

[30] G. Cao, H.-O. Li, G.-D. Yu, B.-C. Wang, B.-B. Chen, X.-X. Song, M. Xiao, G.-C. Guo, H.-W. Jiang, X. Hu, *et al.*, Tunable hybrid qubit in a gaas double quantum

dot, Physical review letters **116**, 086801 (2016).

[31] B. Thorgrimsson, D. Kim, Y.-C. Yang, L. Smith, C. Simmons, D. R. Ward, R. H. Foote, J. Corrigan, D. Savage, M. Lagally, *et al.*, Extending the coherence of a quantum dot hybrid qubit, npj Quantum Information **3**, 32 (2017).

[32] D. Pham and D. Karaboga, *Intelligent optimisation techniques: genetic algorithms, tabu search, simulated annealing and neural networks* (Springer Science & Business Media, 2012).

[33] J. D. Martín-Guerrero and L. Lamata, Reinforcement learning and physics, Applied Sciences **11**, 8589 (2021).

[34] M. Krenn, J. Landgraf, T. Foesel, and F. Marquardt, Artificial intelligence and machine learning for quantum technologies, Phys. Rev. A **107**, 010101 (2023).

[35] T. Chen, X. Chen, W. Chen, Z. Wang, H. Heaton, J. Liu, and W. Yin, Learning to optimize: A primer and a benchmark, J. Mach. Learn. Res. **23** (2022).

[36] K. Gregor and Y. LeCun, Learning fast approximations of sparse coding, in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, ICML'10 (Omnipress, Madison, WI, USA, 2010) p. 399–406.

[37] K. Li and J. Malik, Learning to optimize, in *International Conference on Learning Representations* (2017).

[38] X. Chen, T. Chen, Y. Cheng, W. Chen, A. Awadallah, and Z. Wang, Scalable learning to optimize: A learned optimizer can train big models, in *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXIII* (Springer-Verlag, Berlin, Heidelberg, 2022) p. 389–405.

[39] Y. Chen, M. W. Hoffman, S. G. Colmenarejo, M. Denil, T. P. Lillicrap, M. Botvinick, and N. de Freitas, Learning to learn without gradient descent by gradient descent, in *Proceedings of the 34th International Conference on Machine Learning*, Proceedings of Machine Learning Research, Vol. 70, edited by D. Precup and Y. W. Teh (PMLR, 2017) pp. 748–756.

[40] H. Dai, E. B. Khalil, Y. Zhang, B. Dilkina, and L. Song, Learning combinatorial optimization algorithms over graphs, in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17 (Curran Associates Inc., Red Hook, NY, USA, 2017) p. 6351–6361.

[41] W. Fedus, P. Ramachandran, R. Agarwal, Y. Bengio,

H. Larochelle, M. Rowland, and W. Dabney, Revisiting fundamentals of experience replay, in *International Conference on Machine Learning* (PMLR, 2020) pp. 3061–3071.

[42] A. Seyyedabbasi, A reinforcement learning-based metaheuristic algorithm for solving global optimization problems, Advances in Engineering Software **178**, 103411 (2023).

[43] A. Seyyedabbasi, R. Aliyev, F. Kiani, M. U. Gulle, H. Basyildiz, and M. A. Shah, Hybrid algorithms based on combining reinforcement learning and metaheuristic methods to solve global optimization problems, Knowledge-Based Systems **223**, 107044 (2021).

[44] M. Karimi-Mamaghan, M. Mohammadi, P. Meyer, A. M. Karimi-Mamaghan, and E.-G. Talbi, Machine learning at the service of meta-heuristics for solving combinatorial optimization problems: A state-of-the-art, European Journal of Operational Research **296**, 393 (2022).

[45] H. Samma, C. P. Lim, and J. M. Saleh, A new reinforcement learning-based memetic particle swarm optimizer, Applied Soft Computing **43**, 276 (2016).

[46] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, Continuous control with deep reinforcement learning, arXiv preprint arXiv:1509.02971 (2015).

[47] V. Konda and J. Tsitsiklis, Actor-critic algorithms, Advances in neural information processing systems **12** (1999).

[48] R. Fletcher and M. J. Powell, A rapidly convergent descent method for minimization, The computer journal **6**, 163 (1963).

[49] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, Stable-baselines3: Reliable reinforcement learning implementations, Journal of Machine Learning Research **22**, 1 (2021).

[50] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, Openai gym, arXiv preprint arXiv:1606.01540 (2016).

[51] L. N. Smith, A disciplined approach to neural network hyper-parameters: Part 1–learning rate, batch size, momentum, and weight decay, arXiv preprint arXiv:1803.09820 (2018).