

## ARTICLE OPEN

## Generalizable control for quantum parameter estimation through reinforcement learning

Han Xu<sup>1,2,3</sup>, Junning Li<sup>1,2</sup>, Liqiang Liu<sup>4</sup>, Yu Wang<sup>3</sup>, Haidong Yuan<sup>4\*</sup> and Xin Wang<sup>1,2\*</sup>

Measurement and estimation of parameters are essential for science and engineering, where one of the main quests is to find systematic schemes that can achieve high precision. While conventional schemes for quantum parameter estimation focus on the optimization of the probe states and measurements, it has been recently realized that control during the evolution can significantly improve the precision. The identification of optimal controls, however, is often computationally demanding, as typically the optimal controls depend on the value of the parameter which then needs to be re-calculated after the update of the estimation in each iteration. Here we show that reinforcement learning provides an efficient way to identify the controls that can be employed to improve the precision. We also demonstrate that reinforcement learning is highly generalizable, namely the neural network trained under one particular value of the parameter can work for different values within a broad range. These desired features make reinforcement learning an efficient alternative to conventional optimal quantum control methods.

npj Quantum Information (2019)5:82

; <https://doi.org/10.1038/s41534-019-0198-z>

## INTRODUCTION

Metrology, which studies high precision measurement and estimation, has been one of the main driving forces in science and technology. Recently, quantum metrology, which uses quantum mechanical effects to improve the precision, has gained increasing attention for its potential applications in imaging and spectroscopy.<sup>1–6</sup>

One of the main quests in quantum metrology is to identify the highest precision that can be achieved with given resources. Typically the desired parameter,  $\omega$ , is encoded in a dynamics  $\Lambda_\omega$ . After an initial probe state  $\rho_0$  is prepared, the parameter is encoded in the output state as  $\rho_\omega = \Lambda_\omega(\rho_0)$ . Proper measurements on the output state then reveals the value of the parameter. To achieve the highest precision, one needs to optimize the probe states, the controls during the dynamics and the measurements on the output states. Previous studies have been mostly focused on the optimization of the probe states and measurements.<sup>6</sup> The control only starts to gain attention recently.<sup>7–18</sup> It has now been realized that properly designed controls can significantly improve the precision limits. The identification of optimal controls, however, is often highly complicated and time-consuming. This issue is particularly severe in quantum parameter estimation, as typically optimal controls depend on the value of the parameter, which can only be estimated from the measurement data. When more data are collected, the optimal controls also need to be updated, which is conventionally achieved by another run of the optimization algorithm. This creates a high demand for the identification of efficient algorithms to find the optimal controls in quantum parameter estimation.

Over the past few years, machine learning has demonstrated astonishing achievements in certain high-dimensional input-output problems, such as playing video games<sup>19</sup> and mastering the game of Go.<sup>20</sup> Reinforcement Learning (RL)<sup>21</sup> is one of the most basic yet powerful paradigms of machine learning. In RL, an agent interacts with an environment with certain rules and goals

set forth by the problem desired. By trial and error, the agent optimizes its strategy to achieve the goals, which is then translated to a solution to the problem. RL has been shown to provide improved solutions to many problems related to quantum information science, including quantum state transfer,<sup>22</sup> quantum error correction,<sup>23</sup> quantum communication,<sup>24</sup> quantum control<sup>25–27</sup>, and experiment design.<sup>28</sup>

Here we show that RL serves as an efficient alternative to identify controls that are helpful in quantum parameter estimation. A main advantage of RL is that it is highly generalizable, i.e., the agent trained through RL under one value of the parameter works for a broad range of the values. There is then no need for re-training after the update of the estimated value of the parameter from the accumulated measurement data, which makes the procedure less resource-consuming under certain situations.

## RESULTS

We consider a generic control problem described by the Hamiltonian:<sup>29</sup>

$$\hat{H}(t) = \hat{H}_0(\omega) + \sum_{k=1}^p u_k(t) \hat{H}_k, \quad (1)$$

where  $\hat{H}_0$  is the time-independent free evolution of the quantum state,  $\omega$  the parameter to be estimated,  $u_k(t)$  the  $k$ th time-dependent control field,  $p$  the dimensionality of the control field, and  $\hat{H}_k$  couples the control field to the state.

The density operator of a quantum state (pure or mixed) evolves according to the master equation,<sup>30</sup>

$$\partial_t \hat{\rho}(t) = -i[\hat{H}(t), \hat{\rho}(t)] + \Gamma[\hat{\rho}(t)], \quad (2)$$

where  $\Gamma[\hat{\rho}(t)]$  indicates a noisy process, the detailed form of which depends on the specific noise mechanism and will be detailed later.

<sup>1</sup>Department of Physics, City University of Hong Kong, Tat Chee Avenue, Kowloon, Hong Kong SAR, China. <sup>2</sup>City University of Hong Kong Shenzhen Research Institute, 518057 Shenzhen, Guangdong, China. <sup>3</sup>School of Physics and Technology, Wuhan University, 430072 Wuhan, China. <sup>4</sup>Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Shatin, Hong Kong SAR, China. \*email: hdyuan@mae.cuhk.edu.hk; x.wang@cityu.edu.hk

The key quantity in quantum parameter estimation is the QFI,<sup>31–34</sup> defined by

$$F(t) = \text{Tr}[\hat{\rho}(t)\hat{L}_s^2(t)], \quad (3)$$

where  $\hat{L}_s(t)$  is the so-called symmetric logarithmic derivative that can be obtained by solving the equation  $\partial_\omega \hat{\rho}(t) = \frac{1}{2}[\hat{\rho}(t)\hat{L}_s(t) + \hat{L}_s(t)\hat{\rho}(t)]$ .<sup>31,32,35</sup> According to the Cramér-Rao bound, the QFI provides a saturable lower bound on the estimation as  $\delta\hat{\omega} \geq \frac{1}{\sqrt{nF(t)}}$ , where  $\delta\hat{\omega} = \sqrt{E[(\hat{\omega} - \omega)^2]}$  is the standard deviation of an unbiased estimator  $\hat{\omega}$ , and  $n$  is the number of times the procedure is repeated. Our goal is therefore to search for optimal control sequences  $u_k(t)$  that maximize the QFI at time  $t = T$  (typically the conclusion of the control),  $F(T)$ , respecting all constraints possibly imposed in specific problems. Practically, we consider piecewise constant controls so the total evolution time  $T$  is discretized into  $N$  steps with equal length  $\Delta T$  labeled by  $j$ , and we use  $u_k^{(j)}$  to denote the strength of the control field  $u_k$  on the  $j$ th time step. Researches of such problem are frequently tackled by the Gradient Ascent Pulse Engineering (GRAPE) method,<sup>29</sup> which searches for an optimal set of control fields by updating their values according to the gradient of a cost function encapsulating the goal of the optimal control. It has been found that GRAPE is successful in preparing optimal control pulse sequences that improve the precision limit of quantum parameter estimation in noisy processes.<sup>11,12</sup> Many alternative algorithms can tackle this optimization problem such as the stochastic gradient ascent(descent) method and microbial genetic algorithm,<sup>36</sup> but the convergence to the optimal control fields becomes much slower when the dimensionality ( $p$ ) of the control field or the discretization steps ( $N$ ) increases. Other optimal quantum control algorithms, such as Krotov's method<sup>37–41</sup> and CRAB algorithm,<sup>42</sup> typically depend on the value of the parameter, thus need to be run repeatedly along the update of the estimation, which is highly time-consuming. More efficient algorithms are thus highly desired.

In this work, we employ RL to solve the problem and compare the results to GRAPE. Our implementation of GRAPE follows ref.<sup>11</sup> Figure 1 shows schematics of the RL procedure and the Actor-Critic algorithm<sup>21</sup> used in this work. In order to improve the efficiency of computation, we used a parallel version of the Actor-Critic algorithm called Asynchronous Advantage Actor-Critic (A3C) algorithm.<sup>43</sup> For more extensive reviews of RL, Actor-Critic algorithm and A3C, see Methods and the Supplementary Methods.

Next we apply the algorithm to two commonly considered noisy processes: dephasing and spontaneous emission, to demonstrate the effect of the algorithm.

### Dephasing dynamics

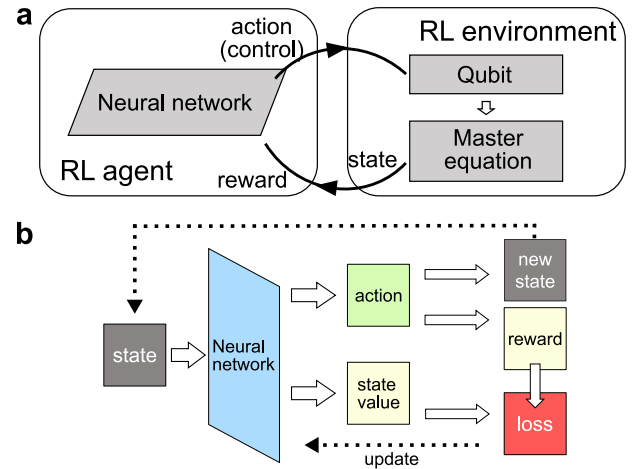
Under dephasing dynamics, the master equation, Eq. (2), takes the following form:<sup>11</sup>

$$\partial_t \hat{\rho}(t) = -i[\hat{H}(t), \hat{\rho}(t)] + \frac{\gamma}{2}[\hat{\sigma}_n \hat{\rho}(t) \hat{\sigma}_n - \hat{\rho}(t)], \quad (4)$$

where

$$\hat{H}(t) = \frac{1}{2}\omega_0 \hat{\sigma}_3 + \mathbf{u}(t) \cdot \boldsymbol{\sigma}, \quad (5)$$

the control field  $\mathbf{u}(t) = (u_1, u_2, u_3)$  is a magnetic field that couples to  $\boldsymbol{\sigma} = (\hat{\sigma}_1, \hat{\sigma}_2, \hat{\sigma}_3)$ , and  $\gamma$  is the dephasing rate which is taken as 0.1 throughout the paper. We consider a dephasing along a general direction given by  $\mathbf{n} = (\sin\vartheta \cos\phi, \sin\vartheta \sin\phi, \cos\vartheta)$ ,  $\hat{\sigma}_n = \mathbf{n} \cdot \boldsymbol{\sigma}$ . The parameter to be estimated is  $\omega_0$  in Eq. (5), the true value of which is assumed to be 1, and we take  $\omega_0^{-1} = 1$  as our time unit. We choose the probe state, i.e. the initial state of the evolution, as  $(|0\rangle + |1\rangle)/\sqrt{2}$  in all subsequent calculations, where  $|0\rangle, |1\rangle$  are the eigenstates of  $\hat{\sigma}_3$ .



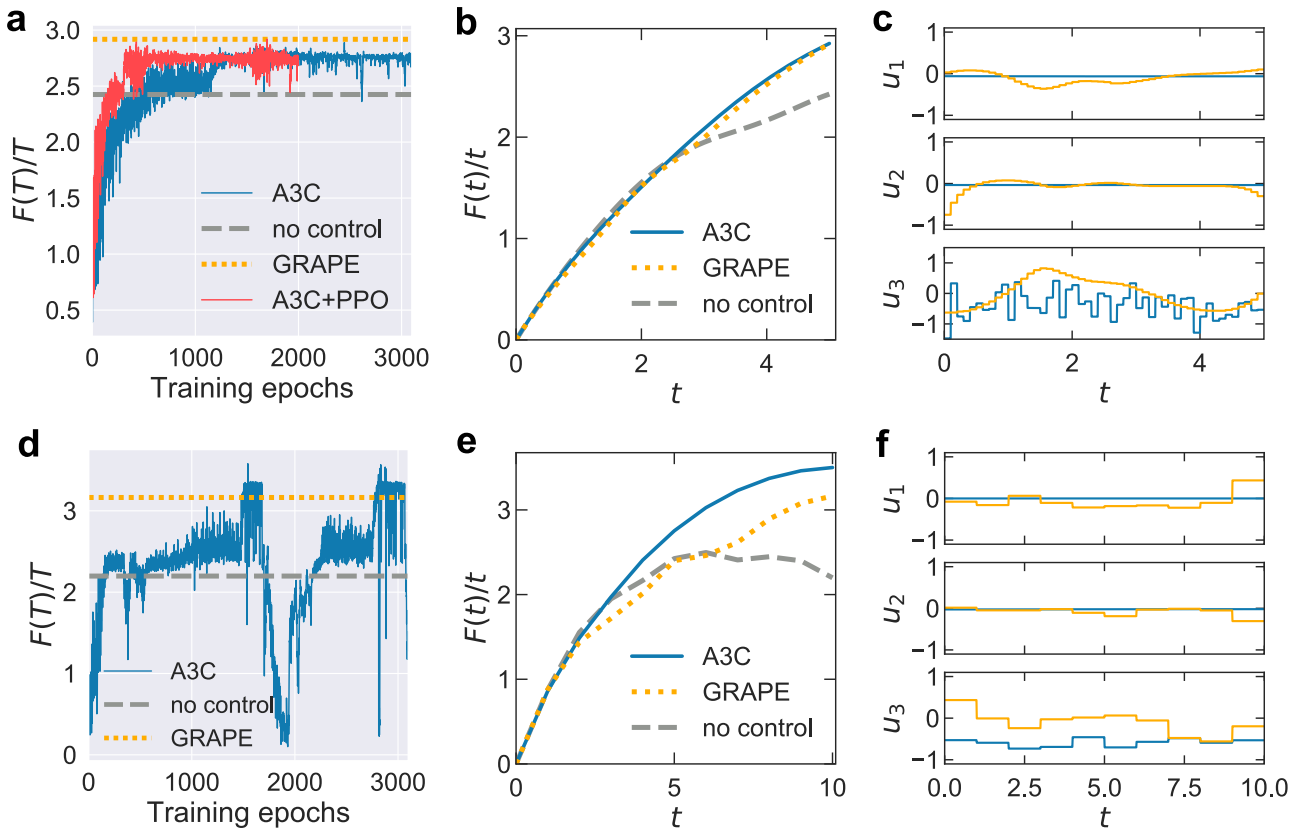
**Fig. 1** Schematics of the reinforcement learning procedure. **a** the RL agent-environment interaction as a Markovian decision process. The RL agent who first takes an action is prescribed by a neural network. The action is essentially the control field which steers the qubit. Then, depending on the consequence of the action, the agent receives a reward. **b** Schematic flow chart of one training step of the Actor-Critic algorithm. The hollow arrows show the data flow of the algorithm, and the dotted arrows show updates of the states and the neural network. In each time step, the state evolves according to the action chosen by the neural network, generating a new state which is used as the input to the network in the next time step. The loss function (detailed in Methods and the Supplementary Methods) is used to update the parameters of the neural network so as to optimize its choice of actions. The procedure is repeated until actions in all time steps are generated, forming the full evolution of the state and concluding one training episode

In Fig. 2 we present our numerical results on QFI under dephasing dynamics with  $\vartheta = \pi/4$ ,  $\phi = 0$  using square pulses. Figure 2a–c show the results for  $\Delta T = 0.1$ . Figure 2a shows the training process in terms of  $F(T)/T$  as functions of the number of training epochs. The blue line shows results from the training using A3C algorithm. The value of  $F(T)/T$  corresponding to results from GRAPE and the case with no control are shown as the orange dotted line and gray dashed line, respectively. The red line shows results from “A3C + PPO”, an enhanced version of A3C which converges faster.<sup>44</sup> The details of this algorithm is explained in the Supplementary Methods. We can see that after sufficient training epochs, results from A3C exceed that for the case with no control, and approaches the optimal results found by GRAPE. On the other hand, “A3C + PPO” converges more quickly to essentially the same result of A3C.

We select one training outcome from those with best performances in Fig. 2a and show  $F(t)/t$  and the pulse profiles in Fig. 2b, c respectively. As can be seen from Fig. 2b, both GRAPE and A3C outperform the case with no control, while the results of A3C are comparable to those from GRAPE.

Figure 2d–f show results with a larger time step,  $\Delta T = 1$ . From the training results shown in Fig. 2d, we see that results from A3C occasionally exceed those from GRAPE, for example at training epoch  $\sim 1600$  and 3000.  $F(t)/t$  and the pulse profile of one of the best-performing results is again shown in Fig. 2e, f, and we see from Fig. 2e that A3C indeed outperforms GRAPE in this case.

We have discussed dephasing dynamics along a particular axis pertaining to Fig. 2, and the results for several other dephasing axes are shown in the Supplementary Discussion. We conclude from these results that in most cases, the A3C algorithm is capable to produce results comparable to those from GRAPE, while in selected situations (e.g. larger  $\Delta T$ ) A3C may outperform GRAPE.



**Fig. 2** Quantum parameter estimation under dephasing dynamics with  $\vartheta = \pi/4$ ,  $\phi = 0$  using square pulses. **a–c** results for  $\Delta T = 0.1$ ,  $T = 5$ . **d–f** results for  $\Delta T = 1$ ,  $T = 10$ . **a, d** show the learning procedure, namely  $F(T)/T$  as functions of training epochs. **b, e** show  $F(t)/t$  for one of the best training results selected from **a** and **d** respectively. **c** and **f** show the pulse profiles corresponding to **b** and **e**

We now discuss the generalizability of the control sequences for quantum parameter estimation, a key result of this paper. As the true value of  $\omega_0$  is not known a priori, the control sequence has to be found optimal for a chosen  $\omega_0$ . When such sequence is applied in situations under other  $\omega_0$  values, the true value is still measured, but the resulting QFI is lower than when the optimal control for true  $\omega_0$  is used. In order to raise the QFI, one must then perform a second measurement using control sequences optimized for the estimated true value of  $\omega_0$ . The entire procedure therefore involves two steps, using different pulse sequences. This is fundamentally different than other typical measurements in quantum control, e.g. evaluation of fidelities of quantum gates,<sup>45</sup> for which there is no need for a second pulse sequence or a second measurement.

The dotted lines in the left column of Fig. 3 show the QFI resulting from measurements with the optimal control found for  $\omega_0 = 1$  with GRAPE. Results without control are shown as gray dashed lines for comparison. The range of  $\omega_0$  covers a period of  $2\pi/T$ . As expected, the QFI is largest at  $\omega_0 = 1$ , but reduces as  $\omega_0$  deviates from 1. As  $\omega_0$  further varies, the QFI increases at some values of  $\omega_0$  which may be due to the geometric relationship of the phase that corresponding to those  $\omega_0$  values and the phase at  $\omega_0 = 1$ . In any case, these QFI values are consistently lower than the value at  $\omega_0 = 1$ . An obvious way to improve the QFI is to generate new optimal control sequences for each value of  $\omega_0$  from GRAPE, but this is costly as the computational complexity scales as  $\mathcal{O}(N^3)$ . A detailed discussion on the computational complexity can be found in Supplementary Discussion.

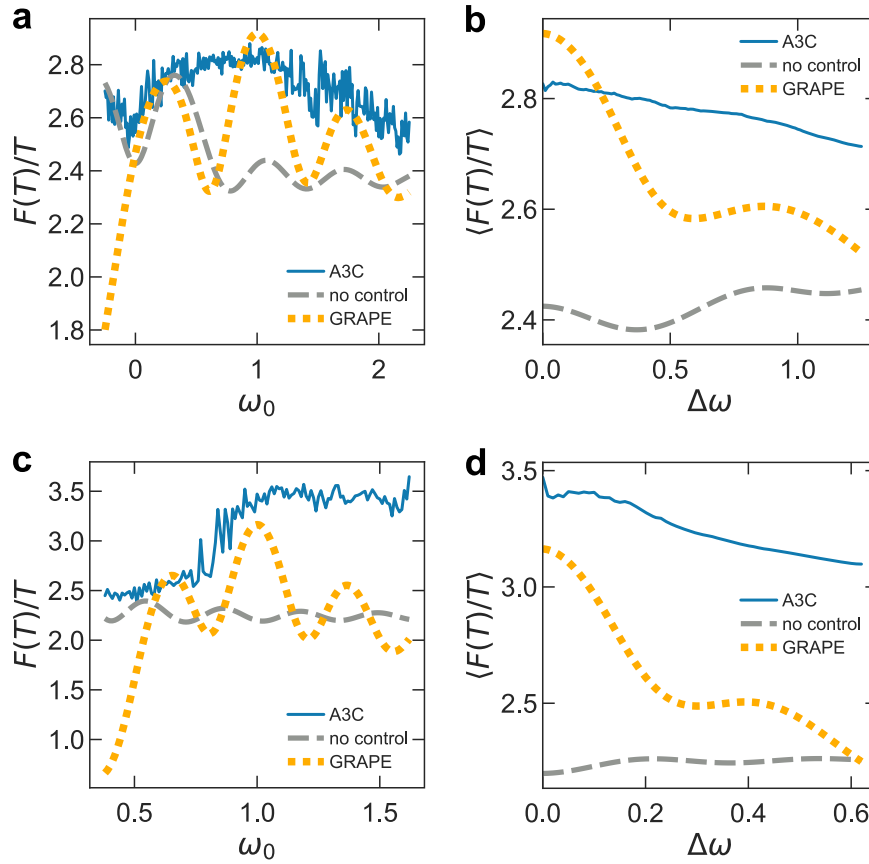
With A3C we have an efficient solution to this problem. We can train the neural network at  $\omega_0 = 1$ , and use this particular network to generate control sequences for different  $\omega_0$  values. The neural

network is only trained at  $\omega_0 = 1$ . However, the trained neural network works for a broad range of parameter values. There is no need to re-train the neural network with the updated estimation of the parameter. The computational cost is thus simply  $\mathcal{O}(N)$  so it is much more efficient than generating new sequences with GRAPE. These results from A3C are shown in the left column of Fig. 3 as blue solid lines which represents the best-performing sequence from 100 trials generated from the trained neural network. For  $\Delta T = 0.1$  (Fig. 3a), although the QFI in the training  $\omega_0 = 1$  is slightly lower for A3C than that of GRAPE, A3C demonstrates higher generalizability as the QFI decreases slowly when  $\omega_0$  deviates from 1. For  $\Delta T = 1$  (Fig. 3c), the QFI of A3C is consistently higher than GRAPE except a narrow range of  $\omega_0$  around 0.65.

To further reveal the generalizability of different methods, we consider the measurement in an ensemble with  $\omega_0$  uniformly distributed in  $[1 - \Delta\omega, 1 + \Delta\omega]$ . The performance of the quantum parameter estimation is therefore given by the average  $F(T)/T$ ,

$$\langle F(T)/T \rangle = \frac{1}{2\Delta\omega} \int_{1-\Delta\omega}^{1+\Delta\omega} F(T)/T d\omega. \quad (6)$$

These results are shown in the right column of Fig. 3, which are averages of the data in the corresponding panels in the left column. As seen from Fig. 3b ( $\Delta T = 0.1$ ),  $\langle F(T)/T \rangle$  for GRAPE is high at small  $\Delta\omega$  but drops quickly as  $\Delta\omega$  is increased. On the contrary,  $\langle F(T)/T \rangle$  for A3C is lower than that for GRAPE at small  $\Delta\omega$ , but decays much more slowly. As a consequence,  $\langle F(T)/T \rangle$  for A3C exceeds that for GRAPE beyond  $\Delta\omega \gtrsim 0.22$ . This result indicates that for measurements involving a reasonably varying parameter, A3C demonstrates higher generalizability. For  $\Delta T = 1$ , the results of A3C always exceed GRAPE as seen from Fig. 3d. The result for



**Fig. 3** Generalizability of the control under dephasing dynamics. **a, c**  $F(T)/T$  vs  $\omega_0$  for three different methods. Note that the results from the GRAPE method are obtained using the pulses generated for  $\omega_0 = 1$  only, while those from A3C are obtained using a neural network trained at  $\omega_0 = 1$ . **b, d** average  $F(T)/T$  in a range  $[1 - \Delta\omega, 1 + \Delta\omega]$  corresponding to the results of **a** and **c** respectively. **a, b**  $\Delta T = 0.1$ ,  $T = 5$ ; **c, d**  $\Delta T = 1$ ,  $T = 10$

A3C decays much more slowly than that for GRAPE, in consistency with the  $\Delta T = 0.1$  case.

Intuitively without control and noise, the optimal strategy is preparing the initial probe state as  $(|0\rangle + |1\rangle)/\sqrt{2}$ , since this state has the fastest rate of rotations under the Hamiltonian. Since the evolution of the state is also affected by dephasing, competitions exist between the parametrization and the effect of noise. When the evolution time is short, the parametrization dominates, in which case the control does not help much. However, in experimentally relevant situations the evolution time is typically long enough for noises to dominate. The controls are therefore useful as they can steer the states to regions where those states are less affected by the noise, even if such states may have a slower speed of parametrization. GRAPE and RL-based methods are both systematical ways to find controls, however, as we have demonstrated, A3C is more generalizable.

#### Spontaneous emission

A process involving the spontaneous emission is described by the Lindblad master equation:<sup>11</sup>

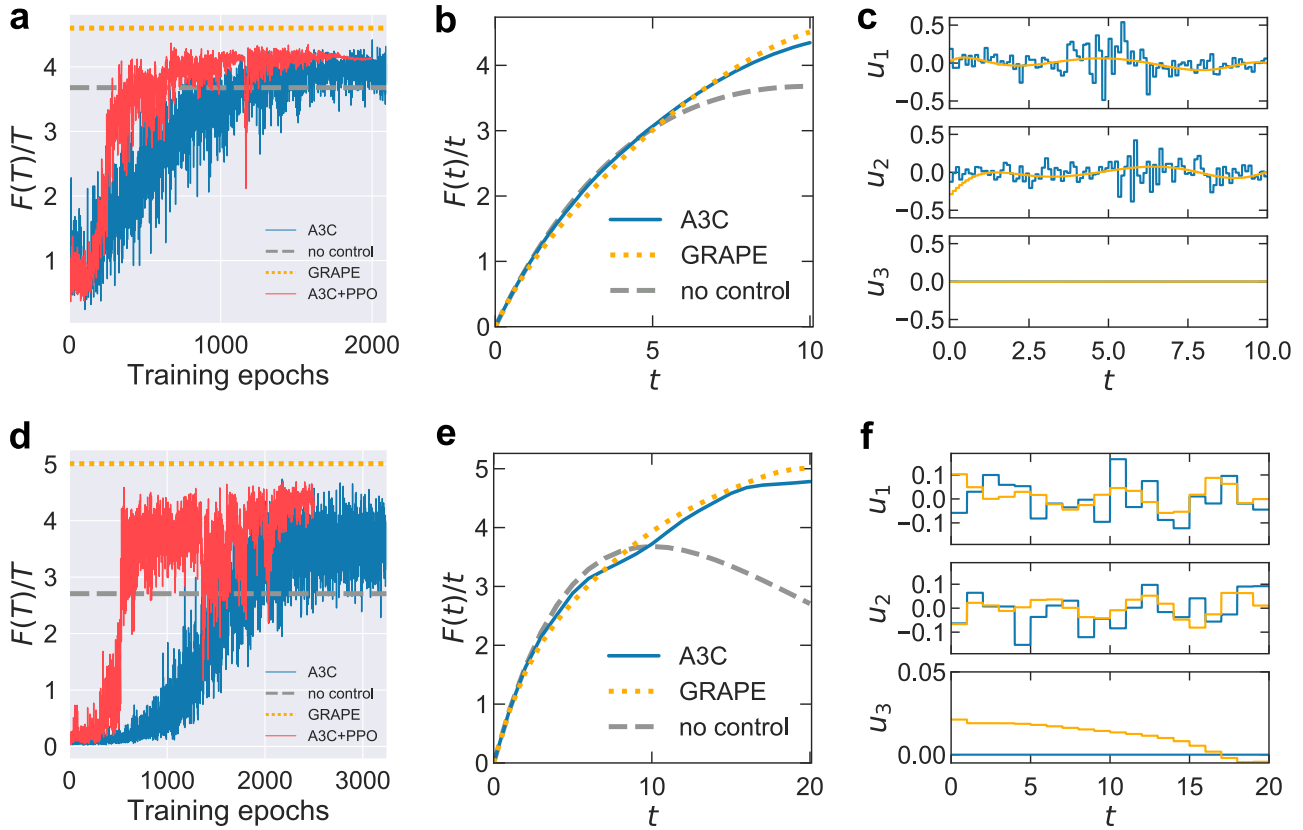
$$\begin{aligned} \partial_t \hat{\rho}(t) = & -i[\hat{H}(t), \hat{\rho}(t)] + \gamma_+ [\hat{\sigma}_+ \hat{\rho}(t) \hat{\sigma}_- - \frac{1}{2} \{\hat{\sigma}_- \hat{\sigma}_+, \hat{\rho}(t)\}] \\ & + \gamma_- [\hat{\sigma}_- \hat{\rho}(t) \hat{\sigma}_+ - \frac{1}{2} \{\hat{\sigma}_+ \hat{\sigma}_-, \hat{\rho}(t)\}], \end{aligned} \quad (7)$$

where  $\hat{\sigma}_\pm = (\hat{\sigma}_1 \pm i\hat{\sigma}_2)/2$  and  $\hat{H}$  is defined as Eq. (5). The relaxation rates are taken as  $\gamma_+ = 0.1$ ,  $\gamma_- = 0$  throughout our discussion.

Figure 4 shows numerical results on QFI with spontaneous emission. Figure 4a–c are for  $\Delta T = 0.1$ ,  $T = 10$ , and Fig. 4d–f show

calculations with a larger time step  $\Delta T = 1$ ,  $T = 20$ . Figure 4a, d [left column] show the A3C training processes, in which the results from GRAPE are indicated as orange dotted line for reference. We see that “A3C + PPO” converges faster, and both A3C and “A3C + PPO” saturate to values slightly lower than GRAPE. Again, one of the best-performing control is picked out and the corresponding  $F(t)/t$  and pulse profiles are shown in the middle and right column respectively. From Fig. 4b, e we see that for the best result from A3C, the QFI is lower than, but comparable to results from GRAPE.

As in the case of dephasing dynamics, we consider the generalizability of different methods in a situation involving  $\omega_0$  that distributes uniformly in a range. Again, we use GRAPE to obtain optimal control sequences for  $\omega_0 = 1$  and apply that to other values. For A3C, we trained the neural network at  $\omega_0 = 1$ ; the resulting sequence is then used to obtain an estimate of the true  $\omega_0$  value. A new sequence is then generated using the neural network already trained at  $\omega_0 = 1$  with the estimated  $\omega_0$ . The best-performing results out of 100 A3C outputs are shown as the blue solid lines in Fig. 5, while the results from GRAPE are shown as the orange dotted lines. The left column of Fig. 5 shows  $F(T)/T$  as functions of  $\omega_0$  for two  $\Delta T$  values. In both cases, the GRAPE method outperforms A3C in a narrow neighborhood around  $\omega_0 = 1$ , but its QFI decreases substantially as  $\omega_0$  further deviates. On the other hand, A3C exhibits great generalizability: for  $\Delta T = 0.1$  the QFI does not decrease until  $\omega_0$  is reduced to  $\omega_0 \lesssim 0.6$ , while for  $\Delta T = 1$  the QFI remains approximately the same for the entire range of  $\omega_0$  considered. The average  $F(T)/T$  in the range  $[1 - \Delta\omega, 1 + \Delta\omega]$  are shown in the right column of Fig. 5. In Fig. 5b, A3C



**Fig. 4** Quantum parameter estimation under spontaneous emission using square pulses. **a–c** results for  $\Delta T = 0.1$ ,  $T = 10$ . **d–f** results for  $\Delta T = 1$ ,  $T = 20$ . **a, d** show the learning procedure. **b** and **e** show  $F(t)/t$  for one of the best training results selected from **a** and **d** respectively. **c** and **f** show the pulse profiles corresponding to **b** and **e**

outperforms GRAPE when  $\Delta\omega \gtrsim 0.22$ , while in Fig. 5d, A3C outperforms GRAPE in an even larger range  $\Delta\omega \gtrsim 0.07$ .

Overall we conclude that in the case of spontaneous emission, the A3C algorithm provides comparable results to GRAPE, although it cannot give higher QFIs. Nevertheless, A3C has much greater generalizability, as is consistent with the case concerning the dephasing dynamics.

#### Sequences with Gaussian pulses

For all results shown above, the control sequences involve square pulses only. In practical experiments, shaped pulses are sometimes used. Therefore in this section we consider Gaussian pulses as an example. The total time  $T$  is still divided into smaller pieces with  $\Delta T$ . However, at the  $j$ th piece the piecewise constant pulse is replaced by a Gaussian centering on that piece and truncated on the ends:

$$u^{(j)}(t) = A^{(j)} \exp\left\{-\left[\frac{(t - t^{(j)})}{\sigma^{g,(j)}}\right]^2\right\}, \quad (8)$$

where  $A^{(j)}$  indicates the amplitude and  $\sigma^{g,(j)}$  the flatness of the pulse. We demonstrate here that with A3C method it is natural to accommodate non-boxcar pulses.

In Fig. 6 we show A3C results using Gaussian pulses and compare them to GRAPE results using square pulses. Figure 6a–c show results under dephasing dynamics with  $\vartheta = \pi/4$ , and Fig. 6d–f show results under the spontaneous emission. In both cases  $\Delta T = 1$ ,  $T = 10$ . For dephasing dynamics, our best results from A3C outperform GRAPE, as is also the case for square pulses generated by A3C. For spontaneous emission, our best-performing result has a QFI value slightly lower than those from GRAPE with square pulses, but their values are very close. These results indicate that

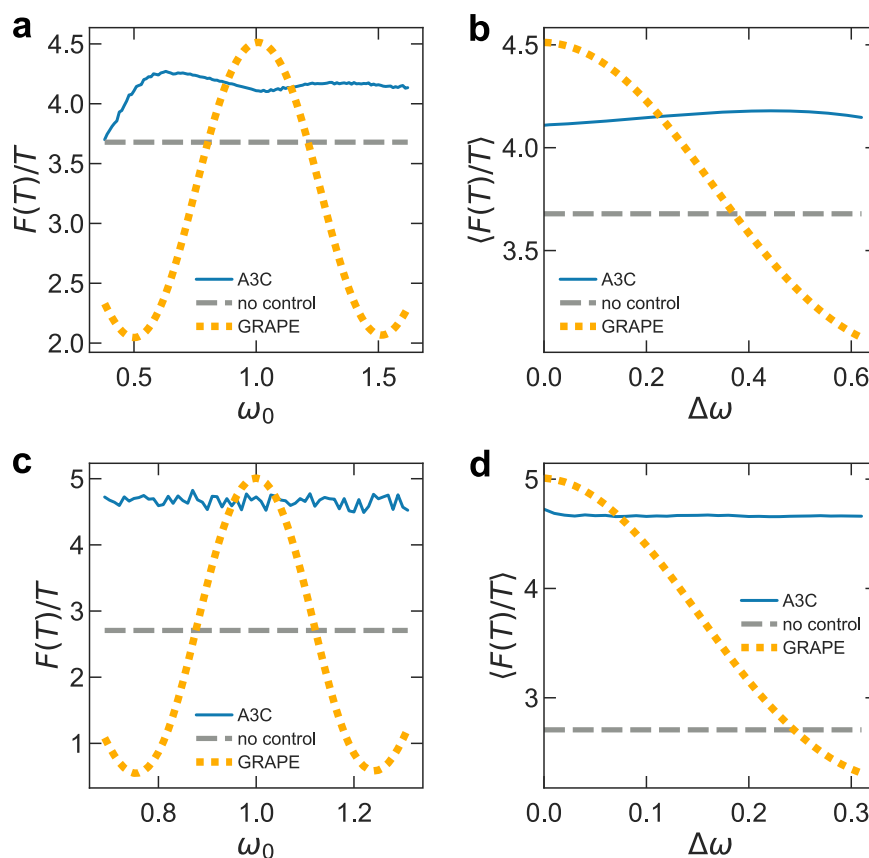
A3C method can naturally accommodate pulses other than square shape. We note that our use of Gaussian pulses is theoretical, and in practical situations, experimentally more relevant ones such as the Blackman pulses<sup>45</sup> should be used. These shaped pulses are implemented by introducing constraints to the gradient in GRAPE<sup>46</sup> or by modifying the action from the RL agent directly.

#### DISCUSSION

The generalizability of RL, or sometimes called “generalization” in the literature, is an actively studied topic in computer science, for example on problems related to game playing where the RL agent trained under one level of the game can be used to clear other levels.<sup>47–50</sup> While the reason why RL is generalizable is not completely clear, one suggestion has it that it likely arises from the underfitting by the neural network to the training data,<sup>51</sup> which is supported by studies showing that reducing overfitting improves generalizability.<sup>50</sup>

The generalizability in fact has a much wider scope than what has been studied here. In the so-called “transfer learning”,<sup>52</sup> experiences gained from one training of the RL agent can be used to improve its performance on different but related tasks by, for example, minimal updates of the network parameters. In contrast, our method does not alter network parameters while only generalizes the neural network in new RL environments with different parameters to estimate. We therefore believe that RL can be made even more generalizable by further studies involving more sophisticated algorithms.

To summarize, RL, in particular the A3C algorithm, is capable of finding the control protocol that enhances QFI in a way comparable to the traditionally used GRAPE method, and is in certain situations superior than GRAPE, e.g. for pulse sequences



**Fig. 5** Generalizability of the control under spontaneous emission. **a, c**  $F(T)/T$  vs  $\omega_0$  for three different methods. Note that the results from the GRAPE method are obtained using the pulses generated for  $\omega_0 = 1$  only, while those from A3C are obtained using a neural network trained at  $\omega_0 = 1$ . **b, d** average  $F(T)/T$  in a range  $[1 - \Delta\omega, 1 + \Delta\omega]$  corresponding to the results of **a** and **c** respectively. **a, b**  $\Delta T = 0.1$ ,  $T = 10$ ; **c, d**  $\Delta T = 1$ ,  $T = 20$

with larger time steps. Moreover, RL can naturally accommodate non-boxcar pulse shapes. Nevertheless, the key advantage afforded by RL is the generalizability, namely the neural network trained for one estimated parameter value can efficiently generate pulse sequences that provide reasonably enhanced QFI for a broad range of parameter values, while in order to achieve the same level of QFI the GRAPE algorithm has to be applied in full each time with a new parameter estimation. Our results therefore suggest that RL-based methods can be powerful alternatives to commonly used gradient-based ones, capable to find control protocols that could be more efficient in practical quantum parameter estimation.

## METHODS

In this section we describe the RL framework shown in Fig. 1. We also provide an expansive review of the RL methods and the detail on implementation in the Supplementary Methods.

Figure 1a shows the RL agent who takes an action as prescribed by a neural network. In our problem, the action is essentially the control field which steers the qubit according to the master equation, Eq. (2), and the resulting state of the evolution determines the reward the agent receives. In practice, the reward encodes the QFI, i.e. higher reward will be obtained when greater QFI is given by the control.

The action taken by the agent implies a time evolution of the quantum state according to Eq. (2) with the control field,  $u_k(t)$ . All possible actions therefore form a continuous set. We solve this problem using the Actor-Critic algorithm,<sup>21</sup> as shown in Fig. 1b. Such algorithm is particularly suitable to our problem as it can treat continuous actions. The key of the algorithm is that the neural network is not only updated using the reward, but also a state value, the latter of which greatly improves the efficiency of the training procedure. At certain time step, the neural network takes the density matrix of the quantum state as an input, and outputs both an

action, and a state value which assesses how likely the state will lead to a larger QFI. The state is then evolved using the output action, obtaining the new state and QFI, which is then implemented into the reward. The reward and state value combines into a so-called “loss function” that provides feedback, by updating the neural network, for the RL agent to make better decisions. The RL agent takes the new quantum state to repeat the above step until time  $T$  is reached, concluding one “episode” of training. After that, the quantum state is reset for the next episode to begin with. A completed episode outputs a pulse profile by sequencing the actions taken in each time step.

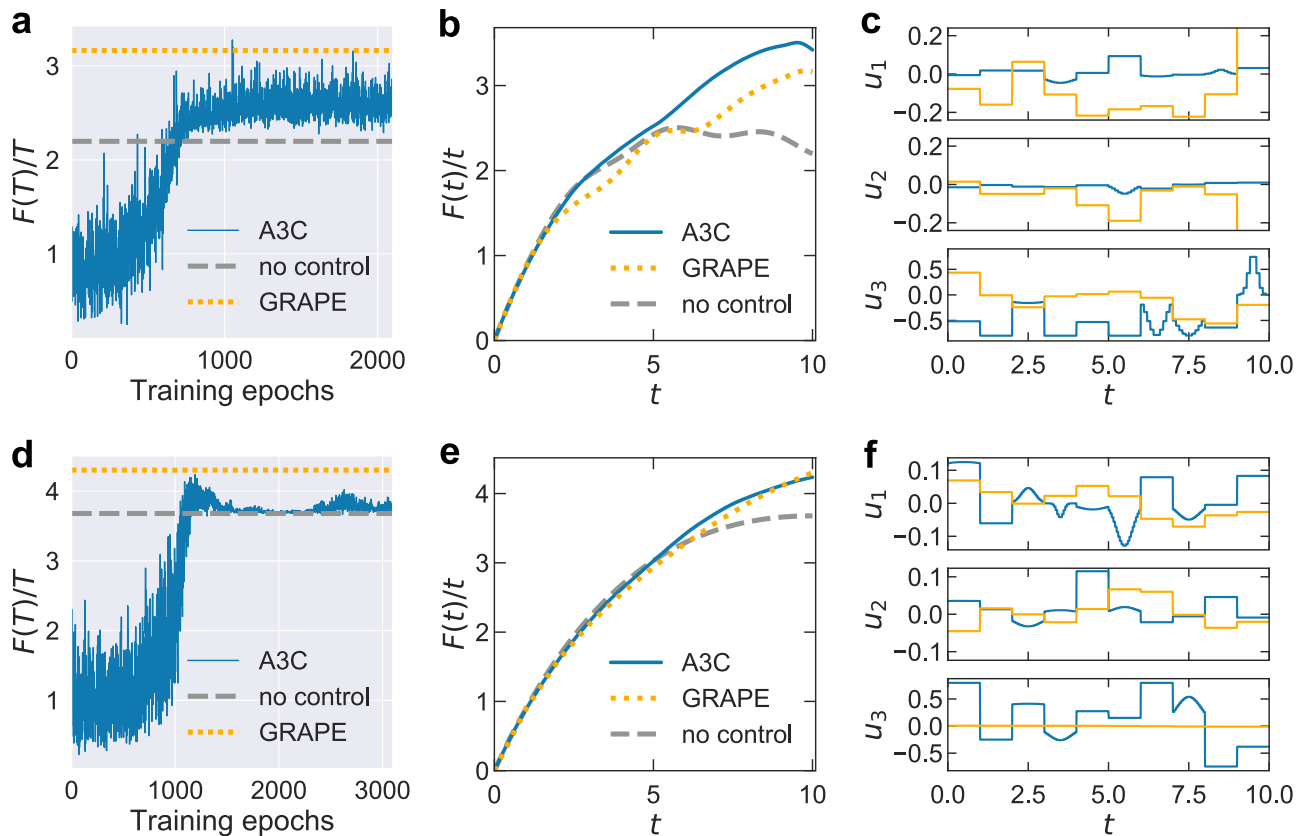
In order to improve the efficiency of computation, we used a parallel version of the Actor-Critic algorithm called Asynchronous Advantage Actor-Critic (A3C) algorithm.<sup>43</sup> In this case, several copies of the agent and environment (called local agents and environments) run in parallel, and as each of them finishes one episode, the solution is delivered to a global agent for further optimization. The optimal policy among these results is then regarded as the output from one “epoch” of training, i.e. one epoch involves several episodes of training from different local agents. Since different local agents deliver their results at different times, the procedure is asynchronous. The details of both the Actor-Critic and the A3C algorithm are described in the Supplementary Methods, as well as the pseudo-code describing the implementation of the algorithm.

## DATA AVAILABILITY

The datasets generated during this study are available from the corresponding author upon reasonable request.

## CODE AVAILABILITY

The code used to generate data is available from the corresponding author upon reasonable request.



**Fig. 6** Quantum parameter estimation using Gaussian pulses as building blocks for A3C. **a–c** dephasing dynamics with  $\vartheta = \pi/4$ . **d–f** spontaneous emission. **a, d** show the learning procedures. **b, e** show  $F(t)/t$  for the best training results selected from each case. **c, f** show the Gaussian pulse profiles, respectively. Note that the GRAPE results shown here use square pulses. Parameters:  $\Delta T = 1$ ,  $T = 10$

Received: 29 April 2019; Accepted: 9 September 2019;  
Published online: 04 October 2019

## REFERENCES

- Kolobov, M. I. The spatial behavior of nonclassical light. *Rev. Mod. Phys.* **71**, 1539 (1999).
- Lugiato, L., Gatti, A. & Brambilla, E. Quantum imaging. *J. Opt. B-Quantum Semi-classical Opt.* **4**, S176 (2002).
- Morris, P. A., Aspden, R. S., Bell, J. E., Boyd, R. W. & Padgett, M. J. Imaging with a small number of photons. *Nat. Commun.* **6**, 5913 (2015).
- Roga, W. & Jeffers, J. Security against jamming and noise exclusion in imaging. *Phys. Rev. A* **94**, 032301 (2016).
- Tsang, M., Nair, R. & Lu, X.-M. Quantum theory of superresolution for two incoherent optical point sources. *Phys. Rev. X* **6**, 031033 (2016).
- Giovannetti, V., Lloyd, S. & Maccone, L. Advances in quantum metrology. *Nat. Photonics* **5**, 222–229 (2011).
- Yuan, H. & Fung, C.-H. F. Optimal feedback scheme and universal time scaling for Hamiltonian parameter estimation. *Phys. Rev. Lett.* **115**, 110401 (2015).
- Yuan, H. Sequential feedback scheme outperforms the parallel scheme for Hamiltonian parameter estimation. *Phys. Rev. Lett.* **117**, 160801 (2016).
- Pang, S. & Jordan, A. N. Optimal adaptive control for quantum metrology with time-dependent Hamiltonians. *Nat. Commun.* **8**, 14695 (2017).
- Pang, S. & Brun, T. A. Quantum metrology for a general Hamiltonian parameter. *Phys. Rev. A* **90**, 022117 (2014).
- Liu, J. & Yuan, H. Quantum parameter estimation with optimal control. *Phys. Rev. A* **96**, 012117 (2017).
- Liu, J. & Yuan, H. Control-enhanced multiparameter quantum estimation. *Phys. Rev. A* **96**, 042114 (2017).
- Yang, J., Pang, S. & Jordan, A. N. Quantum parameter estimation with the Landau-Zener transition. *Phys. Rev. A* **96**, 020301 (2017).
- Naghiloo, M., Jordan, A. N. & Murch, K. W. Achieving optimal quantum acceleration of frequency estimation using adaptive coherent control. *Phys. Rev. Lett.* **119**, 180801 (2017).
- Sekatski, P., Skotiniotis, M., Kolodynski, J. & Dur, W. Quantum metrology with full and fast quantum control. *Quantum* **1**, 27 (2017).
- Fraïsse, J. M. E. & Braun, D. Enhancing sensitivity in quantum metrology by Hamiltonian extensions. *Phys. Rev. A* **95**, 062342 (2017).
- Degen, C. L., Reinhard, F. & Cappellaro, P. Quantum sensing. *Rev. Mod. Phys.* **89**, 035002 (2017).
- Braun, D. et al. Quantum-enhanced measurements without entanglement. *Rev. Mod. Phys.* **90**, 035006 (2018).
- Mnih, V. et al. Human-level control through deep reinforcement learning. *Nature* **518**, 529–533 (2015).
- Silver, D. et al. Mastering the game of Go with deep neural networks and tree search. *Nature* **529**, 484–489 (2016).
- Sutton, R. S. & Barto, A. G. *Reinforcement Learning: An Introduction* (MIT press, 2018).
- Zhang, X.-M., Cui, Z.-W., Wang, X. & Yung, M.-H. Automatic spin-chain learning to explore the quantum speed limit. *Phys. Rev. A* **97**, 052333 (2018).
- Fösel, T., Tighineanu, P., Weiss, T. & Marquardt, F. Reinforcement learning with neural networks for quantum feedback. *Phys. Rev. X* **8**, 031084 (2018).
- Wallnöfer, J., Melnikov, A. A., Dür, W. & Briegel, H. J. Machine learning for long-distance quantum communication. *arXiv preprint arXiv:1904.10797* (2019).
- Bukov, M. et al. Reinforcement learning in different phases of quantum control. *Phys. Rev. X* **8**, 031086 (2018).
- Niu, M. Y., Boixo, S., Smelyanskiy, V. N. & Neven, H. Universal quantum control through deep reinforcement learning. *arXiv preprint arXiv:1803.01857* (2018).
- An, Z. & Zhou, D. Deep reinforcement learning for quantum gate control. *arXiv preprint arXiv:1902.08418* (2019).
- Melnikov, A. A. et al. Active learning machine learns to create new quantum experiments. *Proc. Natl Acad. Sci. USA* **115**, 1221–1226 (2018).
- Khaneja, N., Reiss, T., Kehlet, C., Schulte-Herbrüggen, T. & Glaser, S. J. Optimal control of coupled spin dynamics: design of NMR pulse sequences by gradient ascent algorithms. *J. Magn. Reson.* **172**, 296–305 (2005).

30. Breuer, H.-P. & Petruccione, F. *The Theory Of Open Quantum Systems* (Oxford University Press, 2002).
31. Helstrom, C. W. *Quantum Detection And Estimation Theory* (Academic press, 1976).
32. Holevo, A. *Probabilistic and Quantum Aspects of Quantum Theory* (North-Holland, Amsterdam, 1982).
33. Petz, D. & Ghinea, C. Introduction to quantum Fisher information. In *Quantum Probability and Related Topics*, 261–281 (World Scientific, 2011).
34. Braunstein, S. L. & Caves, C. M. Statistical distance and the geometry of quantum states. *Phys. Rev. Lett.* **72**, 3439–3443 (1994).
35. Braunstein, S. L., Caves, C. M. & Milburn, G. Generalized uncertainty relations: theory, examples, and Lorentz invariance. *Ann. Phys.* **247**, 135–173 (1996).
36. Harvey, I. The microbial genetic algorithm. In *Advances in Artificial Life. Darwin Meets von Neumann* (eds Kampis, G., Karsai, I. & Szathmáry, E.) 126–133 (Springer Berlin Heidelberg, Berlin, Heidelberg, 2011).
37. Sklarz, S. E. & Tannor, D. J. Loading a Bose-Einstein condensate onto an optical lattice: an application of optimal control theory to the nonlinear Schrödinger equation. *Phys. Rev. A* **66**, 053619 (2002).
38. Palao, J. P. & Kosloff, R. Optimal control theory for unitary transformations. *Phys. Rev. A* **68**, 062308 (2003).
39. Machnes, S. et al. Comparing, optimizing, and benchmarking quantum-control algorithms in a unifying programming framework. *Phys. Rev. A* **84**, 022305 (2011).
40. Reich, D. M., Ndong, M. & Koch, C. P. Monotonically convergent optimization in quantum control using Krotov's method. *J. Chem. Phys.* **136**, 104103 (2012).
41. Goerz, M. H., Whaley, K. B. & Koch, C. P. Hybrid optimization schemes for quantum control. *EPJ Quantum Technol.* **2**, 21 (2015).
42. Doria, P., Calarco, T. & Montangero, S. Optimal control technique for many-body quantum dynamics. *Phys. Rev. Lett.* **106**, 190501 (2011).
43. Mnih, V. et al. Asynchronous methods for deep reinforcement learning. *arXiv preprint arXiv:1602.01783v2* (2016).
44. Schulman, J., Wolski, F., Dhariwal, P., Radford, A. & Klimov, O. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347v2* (2017).
45. Goerz, M. H., Halperin, E. J., Aytac, J. M., Koch, C. P. & Whaley, K. B. Robustness of high-fidelity Rydberg gates with single-site addressability. *Phys. Rev. A* **90**, 032329 (2014).
46. Skinner, T. E. & Gershenson, N. I. Optimal control design of pulse shapes as analytic functions. *J. Magn. Reson.* **204**, 248–255 (2010).
47. Pathak, D., Agrawal, P., Efros, A. A. & Darrell, T. Curiosity-driven exploration by self-supervised prediction. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 16–17 (2017).
48. Burda, Y. et al. Large-scale study of curiosity-driven learning. *arXiv preprint arXiv:1808.04355* (2018).
49. Nichol, A., Pfau, V., Hesse, C., Klimov, O. & Schulman, J. Gotta learn fast: A new benchmark for generalization in RL. *arXiv preprint arXiv:1804.03720* (2018).
50. Cobbe, K., Klimov, O., Hesse, C., Kim, T. & Schulman, J. Quantifying generalization in reinforcement learning. *arXiv preprint arXiv:1812.02341* (2018).
51. MacKay, D. J. C. *Information Theory, Inference And Learning Algorithms* (Cambridge University Press, 2003).
52. Taylor, M. E. & Stone, P. Transfer learning for reinforcement learning domains: a survey. *J. Mach. Learn. Res.* **10**, 1633–1685 (2009).

## ACKNOWLEDGEMENTS

This work is supported by the Research Grants Council of the Hong Kong Special Administrative Region, China (Grant Nos. CityU 21300116, CityU 11303617, CityU 11304018, and CUHK 14207717), the National Natural Science Foundation of China (Grant Nos. 11874312, 11604277, 11874292, 11729402, and 11574238), the Guangdong Innovative and Entrepreneurial Research Team Program (Grant No. 2016ZT06D348), and the Key R&D Program of Guangdong province (Grant No. 2018B030326001).

## AUTHOR CONTRIBUTIONS

X.W. and H.Y. conceived the project, H.X. and J.L. performed calculations. All authors discussed the results and implications at all stages and wrote the paper.

## COMPETING INTERESTS

The authors declare no competing interests.

## ADDITIONAL INFORMATION

**Supplementary information** is available for this paper at <https://doi.org/10.1038/s41534-019-0198-z>.

**Correspondence** and requests for materials should be addressed to H.Y. or X.W.

**Reprints and permission information** is available at <http://www.nature.com/reprints>

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2019