Exploring the optimal cycle for a quantum heat engine using reinforcement learning

Gao-xiang Deng,[1] Haoqiang Ai,[2] Bingcheng Wang,[2] Wei Shao,[1,2,*] Yu Liu,[1] and Zheng Cui[2,†]
[1]*Institute of Thermal Science and Technology, Shandong University, Jinan, 250061, People's Republic of China*
[2]*Shandong Institute of Advanced Technology, Jinan, 250100, People's Republic of China*

(Received 21 September 2023; accepted 5 February 2024; published 28 February 2024)

Quantum thermodynamic relationships in emerging nanodevices are significant but often complex to deal with. The application of machine learning in quantum thermodynamics has provided a new perspective. This study employs reinforcement learning to output the optimal cycle of a quantum heat engine. Specifically, the soft actor-critic algorithm is adopted to optimize the cycle of a three-level coherent quantum heat engine with the aim of maximal average power. The results show that the optimal average output power of the coherent three-level heat engine is 1.28 times greater than the original cycle (steady limit). Meanwhile, the efficiency of the optimal cycle is greater than the Curzon-Ahlborn efficiency as well as efficiencies reported by other researchers. Notably, this optimal cycle can be fitted as an Otto-like cycle, which illustrates the effectiveness of the method.

DOI: 10.1103/PhysRevA.109.022246

I. INTRODUCTION

The rapid advancement of science and technology has led to the miniaturization of devices, such as nanoprocess chips and nanothermal engines [1]. Despite their small size, thermodynamic relationships within these microdevices, such as heat-dissipation and heat-work relationships (e.g., power, efficiency), remain crucial due to the quantum effects at the microscopic level. Quantum heat engines (QHEs) are devices that convert thermal energy to mechanical energy in a controlled way, using quantum-scale systems such as single particles or qubits [2] as the working fluid. Studying QHEs can contribute to the emerging interdisciplinary field of quantum thermodynamics [3], elucidate the microscopic thermodynamic principles in miniaturized devices, and promote the development of nanotechnology [4].

In the investigation of QHEs, an open question that remains unresolved is whether quantum effects can be utilized to enhance their performance [5–10]. Recently, several QHEs have been constructed experimentally to investigate the aforementioned query. These approaches involve the manipulation of atomic spins [11,12], ionic spins [13–15], or particle pair spins in crystals [16–18] through laser or magnetic field, the regulation of particle pair spins utilizing nuclear magnetic resonance (NMR) technology [19–21], and the control of a single electron on a microcircuit cooled by dilution refrigeration [22–25]. These experiments achieved the cycle by means of state manipulation of the quantum system (working fluid) via electromagnetic pulse or voltage, with subsequent measurement of the state changes before and after the cycle to obtain the corresponding heat flux and power.

Although some positive conclusions, i.e., quantum effects enhancing the performance of QHEs, were reported [16–18,20], the boosted performance typically requires a careful operation or specific condition. For example, the enhanced performance of QHE disappears when its thermal stroke time exceeds the decoherence time [16]. Moreover, when considering a specific cycle (e.g., Otto cycle), the impact of quantum effects on the performance becomes more ambiguous [10,26–33]. This is primarily because these studies generally assume a specific thermodynamic cycle and this cycle may not ensure optimal power extraction on a long timescale.

Furthermore, prior theoretical studies on maximal power extraction usually focused on slow or fast driving regimes [34–40], assuming specific cycles [29,41–47] such as the Otto cycle [29,44–47], designing adiabatic shortcuts [48–54], or utilizing variational optimization [32,55,56]. The theoretical derivation and calculation of quantum thermodynamics can be extremely challenging, showing a need for numerous assumptions and a narrowed scope in these theoretical studies to obtain an analyzable solution. However, the utilization of reinforcement learning (RL) can potentially identify the optimal long-term power extraction cycles without such limitations, and may thereby alleviate the need for tedious computations.

RL [57] has made significant progress in various fields, including computer games [58–60], robotics [61], and natural language processing [62]. These algorithms exhibit a much stronger exploration ability than humans and have been used for the quantum state preparation and quantum computing [63–71], surpassing the traditional methods used before. Furthermore, RL algorithms have been applied to explore the optimal cycle for two-level QHE and harmonic oscillator QHE [72–74]. Despite the potential of RL algorithms in many-body or multilevel quantum systems, their application has been relatively unreported due to the lack of proper dynamical evolution modeling that simplifies theoretical analysis.

This study employs the RL by the soft actor-critic (SAC) algorithm aiming to explore the optimal cycles with maximal long-term performance for coherent three-level QHE.

*shao@sdu.edu.cn
†zhengc@sdu.edu.cn

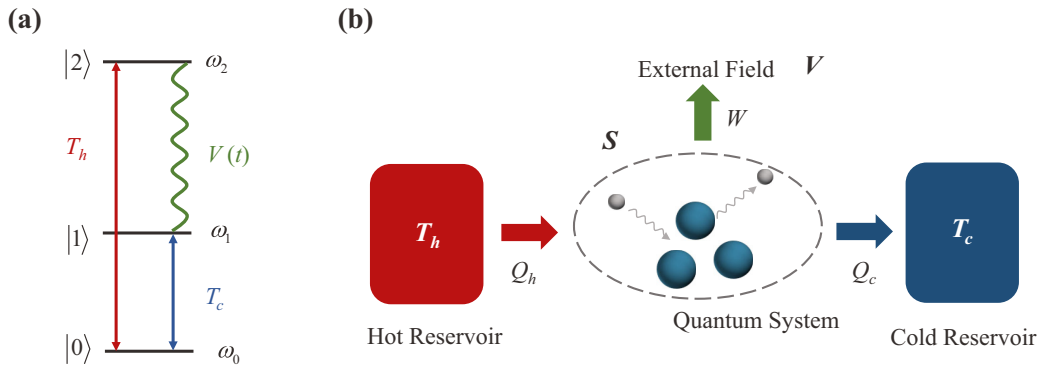

FIG. 1. Schematic of coherent three-level QHE. (a) Energy levels. $|0\rangle$, $|1\rangle$, and $|2\rangle$ are the eigenstates of the quantum system's free Hamiltonian, and ω_0, ω_1, and ω_2 are the corresponding eigenfrequencies. The thermodynamics between these energy levels are given in the main text. (b) Thermal processes. The quantum system undergoes three different processes: absorbing heat Q_h from the hot reservoir at temperature T_h, releasing heat Q_c to the cold reservoir at T_c, and outputting work W to the external field $V(t) = \lambda e^{i\omega t}|1\rangle\langle 2| + \lambda e^{-i\omega t}|2\rangle\langle 1|$. λ, ω, and t are the intensity, frequency, and evolution time of the external field, respectively.

Subsequently, the convergence of the SAC algorithm is analyzed through five consecutive trainings and the power and efficiency are discussed. Finally, applying the Boltzmann function during the compression and expansion processes approximates the optimal cycle as an Otto-like cycle.

II. MODELS AND METHOD

A. Thermodynamic model of coherent three-level QHE

The thermal processes of the coherent three-level QHE are governed by the dynamics of the transitions between the energy levels of the quantum system. Figure 1 depicts the thermodynamic model of the coherent three-level QHE. Figure 1(a) shows three energy levels, i.e., $|0\rangle$, $|1\rangle$, and $|2\rangle$, of the quantum system. The transition from $|0\rangle$ to $|2\rangle$, from $|1\rangle$ to $|0\rangle$, or between $|1\rangle$ and $|2\rangle$ occur when the quantum system couples with a hot reservoir at temperature T_h, a cold reservoir at temperature T_c, and an external field V, respectively. Figure 1(b) illustrates that the coupling to the hot reservoir, the cold reservoir, and the external field will lead to heat absorption Q_h, heat release Q_c, and work output W, respectively.

The quantum system S in this coherent three-level QHE is governed by the Gorini-Kossakowski-Lindblad-Sudarshan (GKLS) equation [75,76],

$$\partial_t \rho_S = -\frac{i}{\hbar}[H_S, \rho_S] + \sum_{i=c,h} D_i[\rho_S(t)], \tag{1}$$

where H_S denotes the Hamiltonian of the quantum system S, D_i is the dissipator which represents the heat dissipation to the reservoirs, and the subscripts c and h, represent the cold and hot reservoir, respectively. The corresponding dissipator is defined as

$$D_i = \sum_{\varepsilon} \Gamma_i(\varepsilon)\Big(L_i^{\varepsilon}(t)\rho_S\big[L_i^{\varepsilon}(t)\big]^{\dagger} - \frac{1}{2}\big\{\big[L_i^{\varepsilon}(t)\big]^{\dagger}L_i^{\varepsilon}(t)\rho_S + \rho_S\big[L_i^{\varepsilon}(t)\big]^{\dagger}L_i^{\varepsilon}(t)\big\}\Big), \tag{2}$$

and the projected jump operator is

$$L_i^{\varepsilon}(t) = \big[L_i^{-\varepsilon}(t)\big]^{\dagger} = \sum_{m,n=0}^{2} \delta_{\varepsilon,\varepsilon_m-\varepsilon_n}|\varepsilon_n(t)\rangle\langle\varepsilon_n(t)|L_i|\varepsilon_m(t)\rangle\langle\varepsilon_m(t)|, \tag{3}$$

with $L_c = |0\rangle\langle 1|$, $L_h = |0\rangle\langle 2|$. Here, the coupling parameter satisfies

$$\Gamma_i(-\varepsilon) = e^{-\beta_i\varepsilon}\Gamma_i(\varepsilon), \tag{4}$$

where $\beta_i = 1/k_B T_i$ is the inversed temperature of the reservoir; ε and $|\varepsilon(t)\rangle$ are the eigenvalues and eigenstates of H_S, respectively. In the following sections, both the Boltzmann constant k_B and reduced Planck constant $\hbar$ are set as 1.

B. RL model of coherent three-level QHE

This section presents the RL model of the coherent three-level QHE based on the thermodynamic model, which begins with outlining the basic setting for the RL model, followed by the description of the long-term performance, the reward function, and the training details.

1. Basic settings for the RL model

Figure 2(a) demonstrates a coherent three-level QHE whose evolution is controlled by an RL agent. The objective of the RL agent is to identify the cycle that maximizes the long-term performance of the QHE by optimizing both the discrete control parameter, $d(t) = \{\text{hot}, \text{cold}, \text{work}\}$, and the continuous control parameter, $u(t)$. It is noteworthy that the optimized combinations of $a(t) = \{d(t), u(t)\}$ at different times are the cycles that maximize the long-term performance.

The discrete control parameter $d(t) =$ hot, cold, work determines the thermal process, with the settings for different thermal processes provided in Tables I and II. The frequency difference between energy levels $|1\rangle$ and $|0\rangle$ is chosen as reference and is written as follows:

$$\omega_{10} = \omega_1 - \omega_0. \tag{5}$$

Similarly, $\omega_{20} = \omega_2 - \omega_0$.

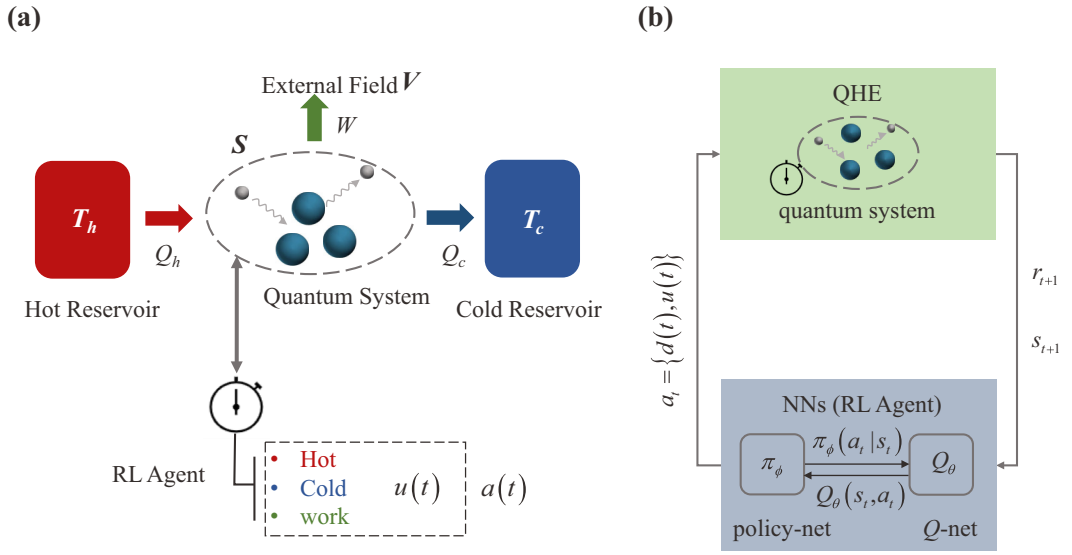

FIG. 2. Schematic of applying the RL algorithm to optimize the long-term performance of a coherent three-level QHE. (a) The RL agent controls the evolution of the QHE through $d(t) = \{\text{hot}, \text{cold}, \text{work}\}$, which determines the thermal process, and $u(t)$, which determines the corresponding system state. $a(t) = \{d(t), u(t)\}$ is a composite control parameter of $d(t)$ and $u(t)$. The specific definitions of $d(t)$ and $u(t)$ are provided in the main text. (b) RL training processes of the coherent three-level QHE. "QHE" refers to the thermodynamic model of QHE, representing the evolution of the quantum system with the control of the RL agent. "NNs" are the neural networks of the RL agent, which are composed of Q-net Q_θ and policy-net π_ϕ. The quantum system takes an action a_t based on the policy given by the RL agent and then transitions to a new state s_{t+1} while receiving a reward r_{t+1}. The NNs receive s_{t+1} and r_{t+1} as input and then output a new action based on a new policy. These steps are repeated until convergence to the optimal policy, which produces the optimal cycle with the maximal long-term performance. See Sec. II C for more details.

The continuous control parameter, denoted as $u(t)$, is initialized at the onset of each thermal process (t) and remains constant until the process concludes $(t + \Delta t)$. This parameter dictates the free Hamiltonian and energy gaps of the quantum system, thereby indicating how the agent manipulates the states of the quantum system throughout each thermal process. The Hamiltonians of the quantum system S at t and $t + \Delta t$ can be denoted, respectively, as

$$H_S(t) = u(t)H_{\text{free}} + V(0) = \begin{pmatrix} u(t)\omega_0 & 0 & 0 \\ 0 & u(t)\omega_1 & \lambda \\ 0 & \lambda & u(t)\omega_2 \end{pmatrix}, \tag{6}$$

$$H_S(t + \Delta t) = u(t)H_{\text{free}} + V(\Delta t) = \begin{pmatrix} u(t)\omega_0 & 0 & 0 \\ 0 & u(t)\omega_1 & \lambda e^{i\omega\Delta t} \\ 0 & \lambda e^{-i\omega\Delta t} & u(t)\omega_2 \end{pmatrix}, \tag{7}$$

TABLE I. Parameters of the reservoirs and external field for the coherent three-level QHE in different processes. g_1 and g_2 are coupling functions, which respectively represent the coupling to the cold and hot reservoir, $\varepsilon_{21} = \varepsilon_2 - \varepsilon_1$.

Process	$\beta_c\omega_{10}$	$\beta_h\omega_{10}$	ω_{20}/ω_{10}	λ/ω_{10}	ω [76]
Hot	0	1	2.5	0	0
Cold	5	0	2.5	0	0
Work	0	0	2.5	0.5	$\frac{\varepsilon_{21}^2 + \frac{1}{4}(g_1+g_2)^2}{\omega_2 - \omega_1}$

where the nonzero value of the nondiagonal elements in the Hamiltonians induces coherence within the QHE. Here,

$$H_{\text{free}} = \begin{pmatrix} \omega_0 & 0 & 0 \\ 0 & \omega_1 & 0 \\ 0 & 0 & \omega_2 \end{pmatrix} \tag{8}$$

is the free Hamiltonian, and

$$V(\tau) = \lambda e^{i\omega\tau}|1\rangle\langle 2| + \lambda e^{-i\omega\tau}|2\rangle\langle 1| = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & \lambda e^{i\omega\tau} \\ 0 & \lambda e^{-i\omega\tau} & 0 \end{pmatrix}, \quad \tau \in [0, \Delta t],$$
$$V(\tau + \Delta t) = V(\tau). \tag{9}$$

This implies that the external field is initialized at the commencement of each thermal process to avert phase accumulation across distinct processes. As per the configurations in Table I, the external field is activated solely during the work process, while it is deactivated during both the hot and cold processes.

TABLE II. Coupling parameters of the coherent three-level QHE in different processes.

Process	$\Gamma_c(\varepsilon_{10})/\omega_{10}$	$\Gamma_h(\varepsilon_{20})/\omega_{10}$	$\Gamma_c(\varepsilon_{20})/\omega_{10}$	$\Gamma_h(\varepsilon_{10})/\omega_{10}$
Hot	0	2	0	0
Cold	2	0	0	0
Work	0	0	0	0

As long as $d(t)$ and $u(t)$ are given by the RL agent, the evolution of S from t to $t+\Delta t$ can be obtained by solving Eq. (1). By applying Eqs. (6) and (7), and substituting the corresponding parameter values for different thermal processes as provided in Tables I and II, the value of $\rho_S(t+\Delta t)$ can be determined.

Figure 2(b) illustrates the RL training processes of the coherent three-level QHE shaded in green, while the RL agent, modeled by the neural networks (NNs) updating through the SAC algorithm, is shaded in blue. The quantum system takes an action a_t based on the policy provided by the RL agent and then transitions to a new state s_{t+1} and receives a reward r_{t+1}. Here, the thermodynamic model of the coherent QHE is designed to generate state transitions and rewards for the actions provided by the RL agent.

2. Long-term performance

The output power $P_{\rm he}$ is generally used as the performance metric for heat engines,

$$P_{\rm he}=\sum_{\alpha=c,h}J_\alpha(t), \tag{10}$$

where J is the heat current and the subscripts c and h refer to the cold reservoir and hot reservoir, respectively. However, the aim of this research is to obtain the optimal cycle with maximal long-term performance. Hence the average power [72],

$$\langle P_{\rm he}\rangle=\tilde{\gamma}\int_0^\infty e^{-\tilde{\gamma}t}P_{\rm he}(t)dt \tag{11}$$

was chosen as the long-term performance metric. The timescale of interest can be manipulated by adjusting $\tilde{\gamma}$, with smaller $\tilde{\gamma}$ and larger $\tilde{\gamma}$ corresponding to a longer and shorter timescale, respectively.

3. Reward function

In reinforcement learning, rewards are generally returned through a designed reward function. The designed reward function needs to possess a clear physical meaning and ensure convergence as well as maximizing the average power in Eq. (11).

Therefore, according to Eqs. (10) and (11), the reward function for the RL model of a coherent three-level QHE should be

$$r_{\rm QHE}=\delta_{d,\bar{d}}\frac{1}{\Delta t}\Delta\langle E_S\rangle, \tag{12}$$

which shows that the average internal energy change rate of quantum system S changes from t to $t+\Delta t$. Here, $d=\{\text{hot, cold, work}\}$, $\bar{d}=\{\text{hot, cold}\}$, Δt is the time step,

$$\delta_{d,\bar{d}}=\begin{cases}1, d=\text{hot or cold}\\ 0, d=\text{work}\end{cases}, \tag{13}$$

and

$$\Delta\langle E_S\rangle=\langle E_S(t+\Delta t)\rangle-\langle E_S(t)\rangle \tag{14}$$

is the change of internal energy $\langle E\rangle=\mathrm{tr}(\rho H)$.

The optimization of average power can be regarded as a discounted RL task [57,77], which operates in both continuous and discrete spaces. This approach has been demonstrated to be effective in learning far-from-equilibrium finite-time thermodynamics [69,72]. Specifically, the RL agent aims to maximize the total future reward [57,78],

$$r_{i+1}+\gamma r_{i+2}+\gamma^2 r_{i+3}+\cdots=\sum_{k=0}^{\infty}\gamma^k r_{i+1+k}, \tag{15}$$

where $i+1$ is the next step, and $\gamma\in[0,1)$ is the discount factor with smaller γ and larger γ corresponding to a shorter and longer future reward, respectively. According to Eq. (14), the reward r_{i+1} is given by

$$\begin{aligned}r_{i+1}=\delta_{d,\bar{d}}\frac{1}{\Delta t}\Delta\langle E_S\rangle=\delta_{d,\bar{d}}\frac{1}{\Delta t}\{&\mathrm{tr}[\rho_S(t+\Delta t)H_S(t+\Delta t)]\\&-\mathrm{tr}[\rho_S(t)H_S(t)]\},\end{aligned} \tag{16}$$

where

$$t=i\Delta t. \tag{17}$$

Substituting Eq. (16) into Eq. (15) brings us to the conclusion that the aim of the RL agent is to maximize Eq. (11) with $\tilde{\gamma}=-\ln\gamma/\Delta t$ [72]. Consequently, the future average power, $\langle P_{\rm he}\rangle$, in Eq. (11) and the average power of the current step i, $\langle P_{\rm he}\rangle_i$, can be, respectively, expressed as

$$\langle P_{\rm he}\rangle=(1-\gamma)\sum_{k=0}^{\infty}\gamma^k r_{i+1+k}, \tag{18}$$

$$\langle P_{\rm he}\rangle_i=(1-\gamma)\sum_{k=0}^{i}\gamma^k r_{i-k}. \tag{19}$$

It can be demonstrated that Eqs. (18) and (19) exhibit consistent convergence, indicating that they converge to the same value. This implies that maximizing either Eq. (18) or (19) will yield the same effect. Thus, it turns out that the aim of the RL agent is to find the optimal cycle that maximizes the average power defined in Eq. (11).

4. Training details

The training parameters of RL for the coherent three-level QHE are listed in Table III.

C. SAC algorithm of RL

The SAC algorithm is a type of actor-critic (AC) RL algorithm that can be divided into three parts: optimization objective, policy evaluation, and policy improvement. The SAC algorithm improves its stability and exploration by introducing an entropy term in the optimization objective [78,79]. As for the iteration steps, firstly, policy evaluation—the actor in this algorithm—selects an action based on a given policy, and the critic scores this selected action. Subsequently, in policy improvement, the actor improves its policy based on the scores from the critic. The "policy" which takes the form of a probability refers to how the actor chooses actions, and "scores" from the critic can be expressed as a Q function.

In order to avoid ambiguity, it is necessary to note that the following symbols and terms have no relation to those used in the preceding text. For instance, the symbol "H" in the SAC algorithm refers to the entropy of a specific policy, not the Hamiltonian.

TABLE III. Training parameters.

Parameter	Coherent three-level QHE
Optimizer	Adam
Learning rate	3×10^{-4}
Number of hidden layers	2
Number of hidden units per layer	256
Activation function	ReLU
Size of buffer R	160×10^3
Batch size	512
Discount γ	0.995
Time step Δt	0.5
"Polyak" coefficient τ	0.005
Update steps	50
Lower and upper limit of u	[0.3, 1.5]
$\bar{H}_{D,\text{init}}$	$0.98 \times \ln 3$
$\bar{H}_{D,\text{final}}$	0.03
$\bar{H}_{D,\text{decay}}$	144×10^3
$\bar{H}_{C,\text{init}}$	-0.72
$\bar{H}_{C,\text{final}}$	-3.8
$\bar{H}_{C,\text{decay}}$	144×10^3
Training steps	500×10^3

1. Optimization objective

The optimization objective can be expressed as

$$\pi^* = \arg\max_{\pi} \sum_{t=0}^{\infty} E_{(s_t,a_t)\sim\rho_\pi}\{r(s_t,\, a_t) + \alpha H[\pi(\centerdot|s_t)]\}, \quad (20)$$

where ρ_π is the state-action marginals of the trajectory distribution induced by a policy π. The current reward obtained by the actor taking an action a_t in the state s_t is denoted as $r(s_t, a_t)$, or r_t for simplicity. Additionally, H is the entropy of a probability distribution, and $\pi(\centerdot|s_t)$ represents the probability distribution of choosing an arbitrary action "$\centerdot$" in the state s_t. The temperature α represents the weight of entropy in the expected reward.

2. Policy evaluation

The SAC algorithm employs Q to evaluate the policy π:

$$Q^\pi = E_{(s_t,\,a_t)\sim\rho_\pi}\left\{\sum_{t=0}^{\infty}\gamma^t r(s_t,\, a_t) + \sum_{t=1}^{\infty}\gamma^t \alpha H[\pi(\centerdot|s_t)]|s_0 = s,\, a_0 = a\right\}. \quad (21)$$

Here, $\gamma \in [0, 1)$ is the discount factor with the consideration that the weight of the early training data should be reduced, and s_0 (a_0) is the initial state (action). The value of Q will be converged through iteration according to the Bellman backup operator,

$$B^\pi Q(s_t,\, a_t) \triangleq r(s_t, a_t) + \gamma E_{s_{t+1}\sim R}\{E_{a'\sim\pi}[Q(s_{t+1},\, a') - \alpha \ln \pi(a'|s_{t+1})]\}, \quad (22)$$

where R is the replay buffer which is used to replace the unknown distribution ρ_π, s_{t+1} is the state of next step, and a' is the action of next step (improved action). The iteration based on Bellman backup between training step k and step $k+1$ is

$$Q_{k+1} = B^\pi Q_k. \quad (23)$$

The SAC algorithm adopts two NNs with parameters θ_1 and θ_2 to fit the Q function, which is the "double Q-learning trick" for continuous control and other improvements [78]. These NNs are generally called Q-net and are denoted as Q_{θ_j}, with $j = \{1, 2\}$. Specifically, the inputs of Q_{θ_j} are (s, a) and the outputs are the value of the Q function. The outputs of Q_{θ_j} will be converged by minimizing the loss function,

$$L_Q(\theta_j) = E_{\substack{(s_t,a_t,r_t,s_{t+1})\sim R \\ a'\sim\pi}} \tfrac{1}{2}[Q_{\theta_j}(s_t,\, a_t) - y(r_t, s_{t+1})]^2, \quad (24)$$

where

$$y(r_t, s_{t+1}) = B^\pi Q_{\bar{\theta}}(s_t,\, a_t) = r(s_t,\, a_t) + \gamma E_{s_{t+1}\sim R}\Big\{E_{a'\sim\pi}\Big[\min_{j=1,2} Q_{\bar{\theta}_j}(s_{t+1},\, a') - \alpha \ln \pi(a'|s_{t+1})\Big]\Big\}, \quad (25)$$

and $\bar{\theta}_j$ is the parameters of the target Q-net $Q_{\bar{\theta}_j}$. Here, the parameters of $Q_{\bar{\theta}_j}$ are not updated during the backpropagation but are updated through "Polyak", i.e.,

$$\bar{\theta}_j \leftarrow \tau\theta_j + (1-\tau)\bar{\theta}_j, \quad (26)$$

to improve learning [78]. Here, τ is a hyperparameter and is listed in Table III.

The iteration step of policy evaluation is shaded in green in Fig. 3. The Q-net Q_{θ_j} and the target Q-net $Q_{\bar{\theta}_j}$ take data fetched from the replay buffer as input to output corresponding values and then update θ_j and $\bar{\theta}_j$ by minimizing the loss function $L_Q(\theta_j)$ given in Eq. (24) and by (26), respectively.

3. Policy improvement

The policy improvement step in the RL algorithm is implemented to obtain the optimal policy π^*. Ideally, the policy obeys the following probability distribution form:

$$\pi(a_t|s_t) \sim \exp[-\varepsilon(s_t,\, a_t)], \qquad \varepsilon(s_t,\, a_t) = -\frac{1}{\alpha}Q(s_t,\, a_t). \quad (27)$$

Equation (27) is the energy-based distribution function for complex tasks, but it cannot be sampled or provide a specific action. Therefore, the Gaussian distribution, denoted by π, is used to approximate the above energy-based distribution. This approximated probability distribution π should minimize the Kullback-Leibler (KL) divergence,

$$\pi_{\text{new}} = \arg\min_{\pi\sim\Pi} D_{\text{KL}}\left[\pi(\centerdot|s_t)\|\frac{\exp[1/\alpha Q^{\pi_{\text{old}}}(s_t,\, a_t)]}{Z^{\pi_{\text{old}}}(s_t,\, a_t)}\right], \quad (28)$$

where $D_{\text{KL}}(P\|Q) = -\sum_i p_i \ln(q_i/p_i)$, Π represents all possible sets of policy π, and Z is a normalization constant which can be ignored during the backpropagation. Consequently, the loss function of the policy neural network (policy-net for simplicity) π_ϕ can be obtained as

$$L_\pi(\phi) = E_{\substack{s_{t+1}\sim R \\ a'\sim\pi_\phi}}[\alpha \ln \pi_\phi(a'|s_{t+1}) - Q(s_{t+1},\, a')], \quad (29)$$

where ϕ is the set of the parameters of π_ϕ. Sampling from the Gaussian distribution $\pi(a'|s_{t+1})$ will produce the improved

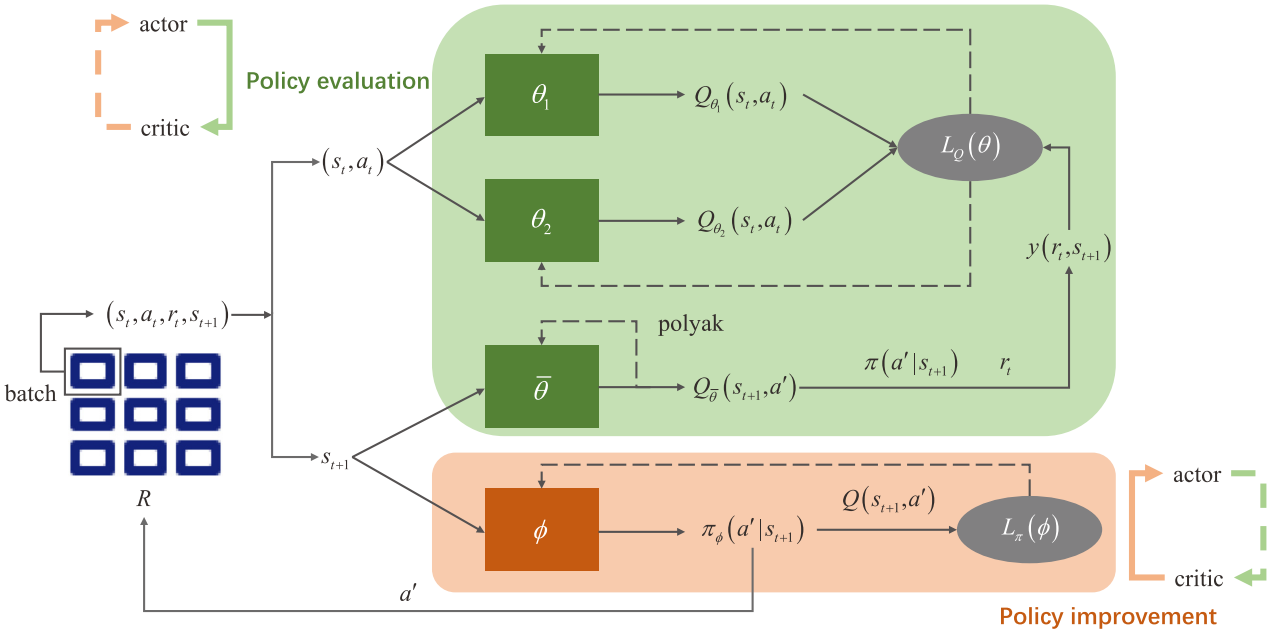

FIG. 3. Steps of SAC algorithm. Batch-sized data, which include the current state s_t, current action a_t, current reward r_t, and next state s_{t+1}, were firstly fetched from the replay buffer R for the training. The training of the SAC algorithm can be divided into two main steps: policy evaluation and policy improvement. (1) Policy evaluation (green). First, (s_t, a_t) are inputted into Q-net whose parameters are θ_1 and θ_2, and outputting $Q_{\theta_1}(s_t, a_t)$ and $Q_{\theta_2}(s_t, a_t)$ to evaluate the current action a_t. After that, $Q_{\bar{\theta}}(s_{t+1}, a')$ is obtained by inputting s_{t+1} and the next action a' given by the policy-net into the target Q-net whose parameters are $\bar{\theta}$. Subsequently, substitute policy $\pi(a'|s_{t+1})$, r_t, and $Q_{\bar{\theta}}(s_{t+1}, a')$ into Eq. (25) to obtain $y(r_t, s_{t+1})$, and yield the loss function of Q-net $L_Q(\theta)$ through Eq. (24). Lastly, the gradient of $L_Q(\theta)$ is backpropagated (dotted arrow) to update θ_1 and θ_2, and $\bar{\theta}$ is "soft" updated through "Polyak" (dotted arrow). (2) Policy improvement (orange). First, the policy-net receives the next state s_{t+1} and outputs the policy $\pi_\phi(a'|s_{t+1})$; then the next (improved) action a' is sampled from $\pi_\phi(a'|s_{t+1})$. Subsequently, substituting $\pi_\phi(a'|s_{t+1})$ and $Q(s_{t+1}, a')$ outputted by the Q-net into Eq. (29) gives the loss function of policy-net, $L_\pi(\phi)$, which was used lastly in backpropagation (dotted arrow) to update ϕ.

action a' for the next state s_{t+1}. The orange area in Fig. 3 illustrates the processes of the policy improvement step.

4. Modifications of SAC algorithm

Given that the SAC algorithm is designed for continuous action space, the optimization in this research is carried out both in discrete and continuous spaces. Therefore, it becomes necessary to make the following modifications to the SAC algorithm.

The temperature α is separated into two components: α_D for the discrete action $d(t)$ and α_C for the continuous action $u(t)$. The values of α_D and α_C can be updated by minimizing the following loss functions [73,78]:

$$\begin{aligned} L_{\alpha_D}(\alpha_D) &= \alpha_D E_{s\sim R}\big[H_D^\pi(s) - \bar{H}_D\big], \\ L_{\alpha_C}(\alpha_C) &= \alpha_C E_{s\sim R}\big[H_C^\pi(s) - \bar{H}_C\big], \end{aligned} \tag{30}$$

where

$$\begin{aligned} H_D^\pi(s) &= -\sum_d \pi_D(d|s)\ln\pi_D(d|s), \\ H_C^\pi(s) &= -\sum_d \pi_D(d|s) E_{u\sim\pi_C(\cdot|d,s)}[\ln\pi_C(u|d,s)] \end{aligned} \tag{31}$$

is the current entropy of the discrete policy π_D and the continuous policy π_C, respectively, and

$$\begin{aligned} \bar{H}_D &= \bar{H}_{D,\text{final}} + (\bar{H}_{D,\text{init}} - \bar{H}_{D,\text{final}})\exp(-n_{\text{steps}}/\bar{H}_{D,\text{decay}}), \\ \bar{H}_C &= \bar{H}_{C,\text{final}} + (\bar{H}_{C,\text{init}} - \bar{H}_{C,\text{final}})\exp(-n_{\text{steps}}/\bar{H}_{C,\text{decay}}) \end{aligned} \tag{32}$$

is the target entropy of the discrete and continuous policy, respectively; n_{steps} is the current trained step. It is noteworthy that both α_D and α_C are the parameter and the output of the NNs. Consequently, they are typically initialized as zero-dimensional tensors before training.

Replacing $E_{a'\sim\pi}[-\alpha\ln\pi(a'|s_{t+1})]$ in Eq. (25) with $\alpha_D H_D^\pi(s_{t+1}) + \alpha_C H_C^\pi(s_{t+1})$ yields the loss function for the Q-net:

$$L_Q(\theta_j) = E_{\substack{(s_t,a_t,r_t,s_{t+1})\sim R \\ a'\sim\pi}} \tfrac{1}{2}[Q_{\theta_j}(s_t, a_t) - y(r, s_{t+1})]^2. \tag{33}$$

Here,

$$\begin{aligned} y(r_t, s_{t+1}) = r(s_t, a_t) + \gamma E_{s_{t+1}\sim R}\Big\{E_{a'\sim\pi}\big[\min_{j=1,2} Q_{\bar{\theta}_j}(s_{t+1}, a')\big] \\ + \alpha_D H_D^\pi(s_{t+1}) + \alpha_C H_C^\pi(s_{t+1})\Big\}, \end{aligned} \tag{34}$$

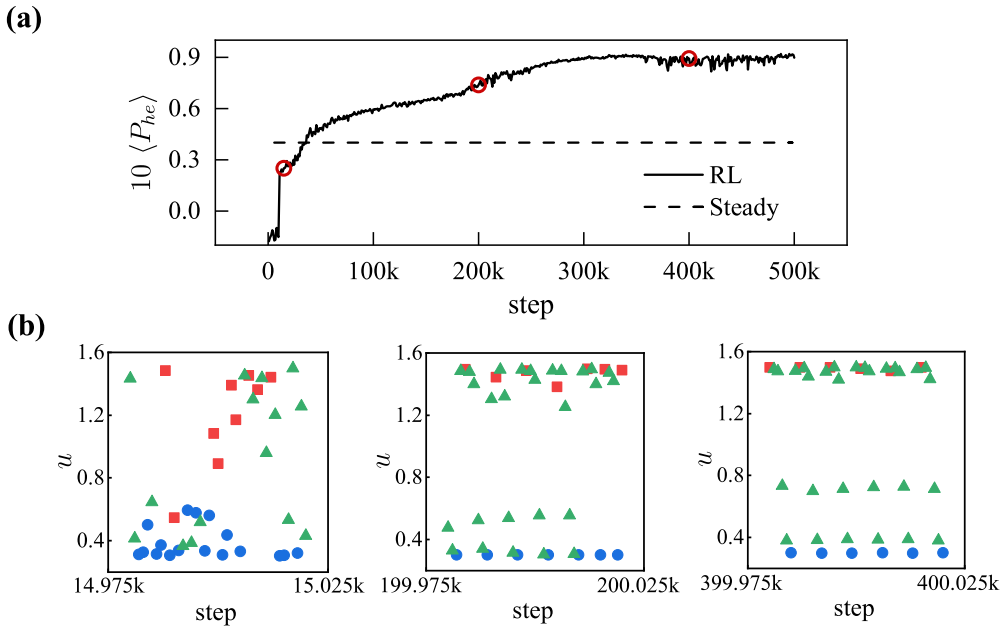

FIG. 4. Training results of the entangled three-level QHE. (a) Average output power $P_{\rm he}$ during the training. The solid line represents the average output power obtained by the RL agent and the dashed line represents the "Steady" limit derived in Ref. [76]. One step corresponds to time step Δt in Table III. (b) Different policies (cycle) given by the RL agent during different training periods marked in (a). The red, blue, and green points represent hot, cold, and work processes, respectively, and u denotes the corresponding system state of these processes. See Table III for details of the training parameters.

and the value of the target Q-net is [77,80]

$$E_{a'\sim\pi}\big[\min_{j=1,2} Q_{\bar{\theta}_j}(s_{t+1},\, a')\big] = \sum_{d'} \pi_D(d'|s_{t+1}) E_{u'\sim\pi_C(\cdot|d,s_{t+1})}\big[\min_{j=1,2} Q_{\bar{\theta}_j}(s_{t+1}, d', u')\big]. \tag{35}$$

The entropy of the current policy is

$$\alpha_D H_D^{\pi}(s_{t+1}) + \alpha_C H_C^{\pi}(s_{t+1}) = -\alpha_D \sum_{d'} \pi_D(d'|s_{t+1}) \ln \pi_D(d'|s_{t+1}) - \alpha_C \sum_{d'} \pi_D(d'|s_{t+1}) \mathrm{E}_{u'\sim\pi_C(\cdot|d',s_{t+1})}[\ln \pi_C(u'|d', s_{t+1})]. \tag{36}$$

Similarly, replacing $E_{a_t\sim\tilde{\pi}_\phi}[\alpha \ln \pi_\phi(a_t|s_t)]$ in Eq. (29) with $\alpha_D H_D^{\pi_\phi}(s) + \alpha_C H_C^{\pi_\phi}(s)$ gives the loss function of the policy-net:

$$\begin{aligned} L_\pi(\phi) &= E_{\substack{s_t\sim R \\ a_t\sim\pi_\phi}} [\alpha \ln \pi_\phi(a_t|s_t) - Q_\theta(s_t,\, a_t)] \\ &= E_{s_t\sim R}\Big[\sum_d \pi_{D,\phi}(d|s_t)\alpha_D \ln \pi_{D,\phi}(d|s_t) \\ &\quad + \alpha_C \ln \pi_{C,\phi}(u|d, s_t) \\ &\quad - \sum_d \pi_{D,\phi}(d|s_t) \min_{j=1,2} Q_{\theta_j}(s_t, d, u) \Big]. \end{aligned} \tag{37}$$

III. RESULTS AND DISCUSSIONS

A. Training results

The average output power $\langle P_{\rm he}\rangle$ and the policies during the training are given in Figs. 4(a) and 4(b), respectively. Figure 4(a) demonstrates that the maximum average output power of the "RL Cycle" (solid line) converges to about 0.91, which is approximately 1.28 times greater than 0.399 of the "Steady" limit (dashed line). The reason is that the agent designs different u for different processes d, strengthening or weakening the corresponding processes to improve the average output power. Additionally, Fig. 4(b) shows the convergence of different policies (cycles) provided by the RL agent as the number of training steps increases.

The well-trained RL agent produces the optimal cycle as shown in Fig. 5. The optimal u's for the hot and cold processes are fixed at 1.5 and 0.3, respectively. However, optimal u for the work processes varies with time between 0.4 and 1.5. At the same time, the corresponding process time steps for the hot, cold, and work processes are 1, 1, and 5, respectively. A further discussion on these results will be provided in the following section.

B. Convergence of the SAC algorithm

The SAC algorithm incorporates randomness within its operations. For example, the "batch" used for training is randomly sampled from a dynamically changing "replay buffer" [57]. Moreover, the agent's policy is derived from probability sampling of the output probability distribution [78]. Therefore, its stability needs to be scrutinized. Thus, Fig. 6 gives the average power of five consecutive trainings

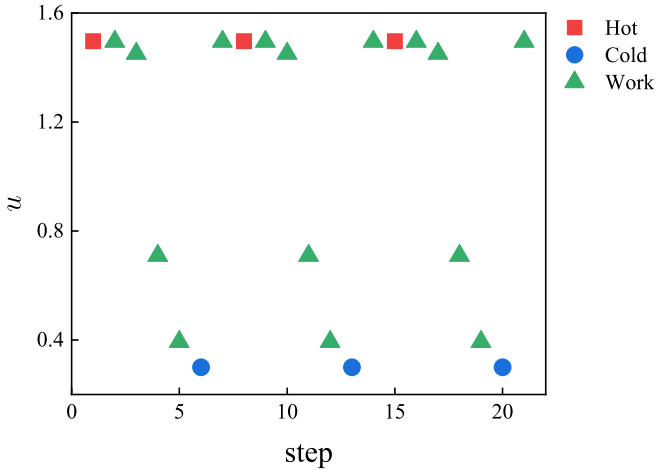

FIG. 5. Optimal cycle produced by the NNs trained by the RL algorithm. One step corresponds to Δt in Table III, controlling parameter u represents the system state, and the different processes are denoted by different colors. The red, blue, and green markers indicate the hot, cold, and work processes, respectively.

of the coherent three-level QHE. These results demonstrate a similar converged average power of all five trainings, which indicates the reliability of this algorithm.

C. Comparative analysis of average power across different cycles

In the section, the hot process and cold process were unchanged, but the work process was modified based on the cycle given by the RL agent. Figures 7(a)–7(c) show the patterns of cycle 1, cycle 2, and cycle 3, respectively. Each cycle shares the same hot and cold processes, but the work process varies. The u's of the work processes in cycles 1 and 2 change linearly, with the work process of cycle 2 enduring longer to achieve sufficient work. Conversely, cycle 3 adopts the work processes given by the RL agent.

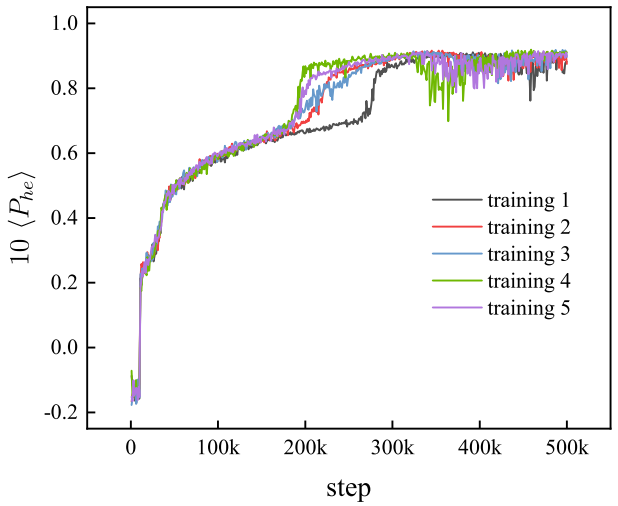

FIG. 6. Averaged power curves of five consecutive trainings for the coherent three-level QHE.

The system is initialized to the Gibbs state,

$$\rho_S(0) = \frac{e^{-\beta_S(0)H_S(0)}}{\mathrm{tr}(e^{-\beta_S(0)H_S(0)})}, \tag{38}$$

where

$$\beta_S(0) = 3/\omega_{10},$$
$$H_S(0) = \begin{pmatrix} \omega_0 & 0 & 0 \\ 0 & \omega_1 & 0 \\ 0 & 0 & \omega_2 \end{pmatrix}. \tag{39}$$

The final average power over 1k steps, corresponding to the three cycle modes discussed above, is depicted in Fig. 7(d). It can be seen from the figure that the average power of cycles 1, 2, and 3 is 0.204, 0.591, and 0.837, respectively. This indicates that cycle 3 given by the RL agent, holds a significant advantage over cycles 1 and 2. Specifically, when compared to cycle 2, cycle 3 can improve the performance by approximately 41.6%.

D. Efficiency of RL cycle

The quantum system is initialized to the same Gibbs state given in Eq. (38) and then evolves for 1k steps according to the RL cycle. After the evolution is over, the cycle efficiency is calculated by the following formula [73],

$$\eta = \frac{\eta_C}{1 + \frac{\langle\sigma\rangle}{\beta_c\langle P_{\mathrm{he}}\rangle}}, \tag{40}$$

where Carnot efficiency $\eta_C = 1 - \beta_h/\beta_c$, average power $\langle P_{\mathrm{he}}\rangle$ is given by Eq. (11), and average entropy production,

$$\langle\sigma\rangle = \tilde{\gamma}\int_0^\infty e^{-\tilde{\gamma}t}\sigma(t)dt. \tag{41}$$

Here, the instantaneous entropy production is given by

$$\sigma(t) = \sum_{\alpha=c,h}\beta_\alpha J_\alpha(t). \tag{42}$$

According to Eq. (40), the efficiency of the RL cycle can be obtained at approximately 65.4%. This efficiency is greater than the Curzon-Ahlborn efficiency which is the efficiency at maximum power (EMP) derived by Curzon and Ahlborn [81], with $\eta^{\mathrm{EMP}} = 1-\sqrt{\beta_h/\beta_c} = (1-\sqrt{1/5}) \times 100\% = 55.3\%$. However, recent research [45,82–84] shows that the EMP of QHE may exceed η^{EMP}, which is consistent with our results.

E. Fitting of RL cycle

The cycle (Fig. 5) obtained by the RL agent in this study can be fitted to a periodic cycle as shown in Fig. 8. The durations of working-1, heating, working-2, and cooling process are τ_1, τ_2, τ_3, and τ_4, respectively. As can be observed from the figure, u remains unchanged at 1.495 and 0.300 during the heating and cooling processes, respectively. Meanwhile, the value of u during the working-2 process can be fitted by the Boltzmann function [85], as described in Eq. (43). The specific fitting parameters are provided in Table IV. The coefficient of determination, R^2, for the working-2 process is 0.999, demonstrating an approximation of good acceptance. Consequently, we hypothesize that the working-1 process

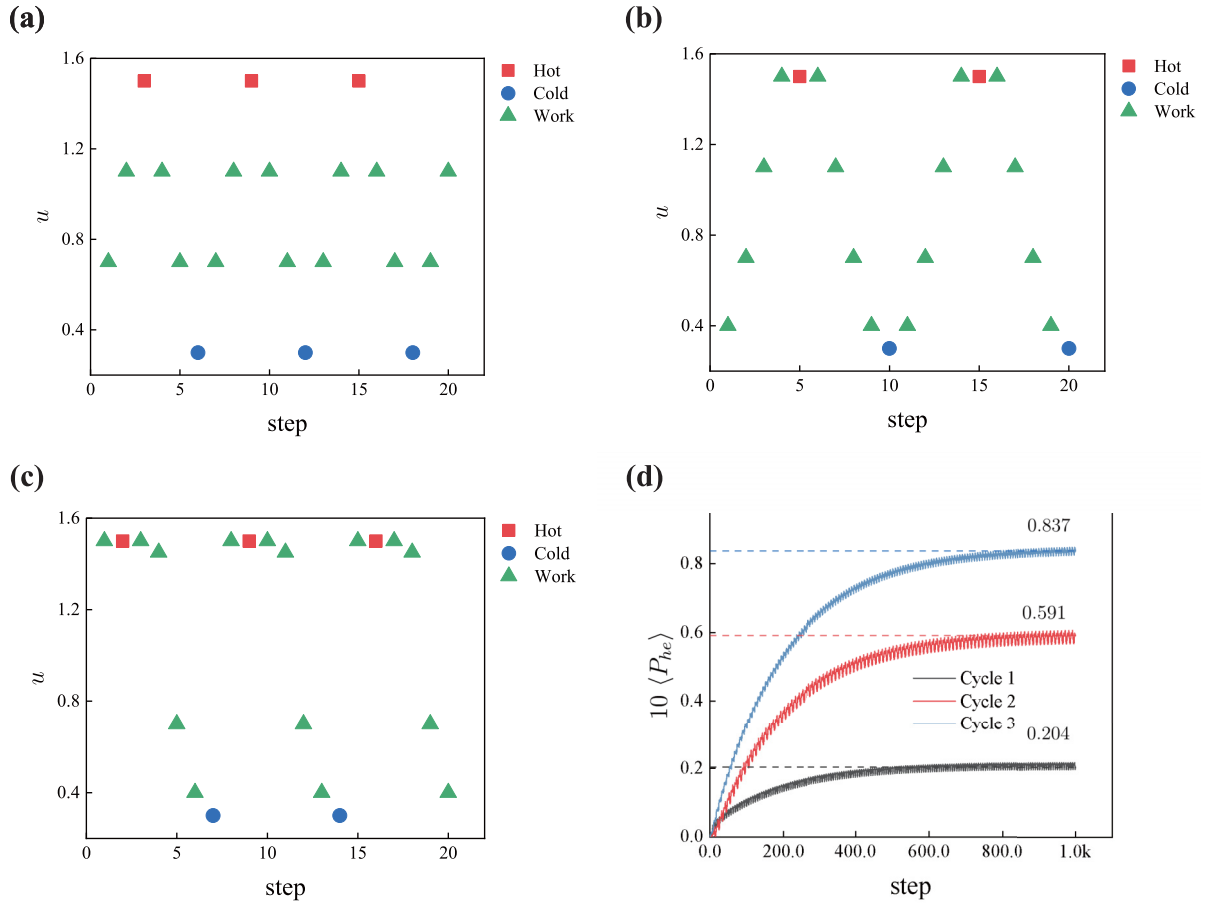

FIG. 7. Different cycle modes and their average power. (a)–(c) represent cycles 1–3 as described in the main text, respectively. Each step signifies a time of 0.5. The hot, cold, and work processes are represented by red, blue, and green, respectively, while u denotes the system state. (d) depicts the average power under three distinct cycle modes over 1k steps. The black, red, and blue solid lines show the evolution of average power changes under cycles 1–3, respectively. The dotted lines indicate the final average power of corresponding cycle.

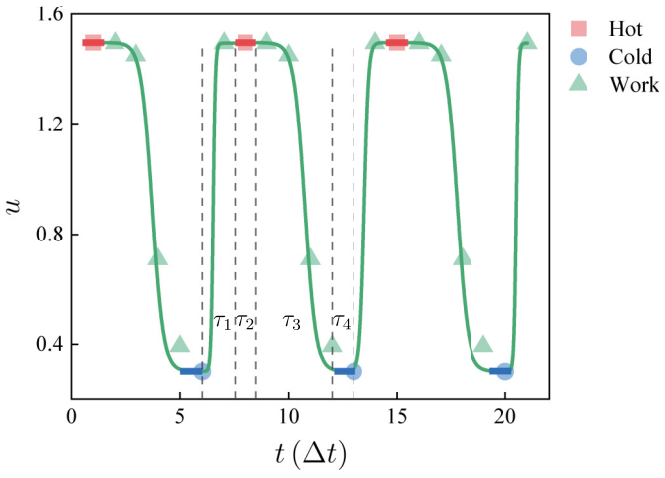

FIG. 8. Fitted cycle based on the RL cycle depicted in Fig. 5. The red, blue, and green solid lines represent the heating, cooling, and working processes, respectively. The translucent markers indicate the RL cycle. A single cycle of the fitted Otto cycle is marked by dashed lines, with τ_1, τ_2, τ_3, and τ_4 representing the duration of the working-1, heating, working-2, and cooling processes, respectively.

could also be approximated by the Boltzmann function. It should be note that the working-1 process only has one data point within one period, meaning it could be fitted by any function. We attempted to reduce Δt to obtain more data points, but the SAC algorithm failed to converge.

$$u(t) = \frac{A_1 - A_2}{1 + e^{(t-t_0)/dt}} + A_2. \tag{43}$$

TABLE IV. Parameter values of the fitted Boltzmann function for different processes. t denotes the duration of the process with a unity of Δt, while A_1, A_2, t_0, and dt are the fitted Boltzmann function parameters; R^2 is the coefficient of determination. The last row displays the parameters of the general working process, where the values outside and inside the brackets refer to the working-1 and working-2 processes, respectively.

Process	t	A_1	A_2	t_0	dt	R^2
Working-1	[6, 7.5]	0.300	1.495	6.75	0.05	
Working-2	[8.5, 12]	1.497	0.300	10.25	0.25	0.990
Working	$[t_{\min}, t_{\max}]$	0.3(1.5)	1.5(0.3)	$\frac{t_{\min}+t_{\max}}{2}$	0.05(0.25)	

This fitted cycle can be regarded as an Otto-like cycle. The quantum Otto cycle generally includes four processes: adiabatic compression, isochoric heating, adiabatic expansion, and isochoric cooling [86]. The working-1 and working-2 processes, respectively, increase and decrease the energy gap, which are similar to the compression and expansion processes within the quantum Otto cycle. Therefore, this fitted cycle is analogous to the Otto cycle for they both maximize the power, demonstrating the effectiveness of our method.

F. Discussion with the finite-time Otto cycle

Applying the RL algorithm also provides a new perspective for investigating the finite-time Otto cycle with the advantage of alleviating tedious analysis. The finite-time Otto cycle [29,37,38,44–54,87] is proposed to deal with the practical application defects of the ideal Otto cycle. The ideal Otto cycle generally needs to meet two assumptions. Firstly, the compression and expansion process should be quasistatic to prevent heat leakage. Secondly, the system should be in the Gibbs state after the isochoric processes. However, in actual processes, the quasistatic processes and the slow isochoric processes required by the Gibbs state could lead to a decreased power. To this end, some finite-time Otto cycles, such as those utilizing the "adiabatic shortcut" [48–54] to speed up the adiabatic processes, are designed to improve output power. However, most of these shortcuts rely on experience or require complex theoretical derivations which hinder the corresponding research.

It should be noted that due to the complexity of theoretical derivation and calculation, research on the finite-time Otto cycle of the three-level QHE is currently challenging and limited [4,46,76,82,88]. Thus this study only compares the power of the steady state and does not delve into the power of the Otto cycle.

IV. CONCLUSIONS

This research employed the RL via the SAC algorithm to optimize the long-term performance of the coherent three-level quantum heat engine (QHE), specifically aiming to maximize the average output power. Remarkably, the RL agent gave a cycle with an average output power of approximately 1.28 times greater than the steady limit. Furthermore, the convergence of the SAC algorithm was verified through five consecutive trainings and the efficiency of this cycle was found to be larger than the Curzon-Ahlborn efficiency. Finally, the results also showed that the optimal cycle could be fitted as an Otto-like cycle by adopting the Boltzmann function during the compression and expansion processes. This demonstrates the feasibility of utilizing reinforcement learning within the power optimization of QHE.

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ACKNOWLEDGMENTS

This work was supported by the Taishan Scholar Project (Grant No. tsqn202103142), and the Natural Science Foundation of Shandong Province (Grant No. ZR2021QE033).

G.-x.D., W.S., and Z.C. designed this research and its corresponding model and method. G.-x.D. wrote the code, carried out the training, and processed the data. G.-x.D., H.A., B.W., W.S., and Y.L. wrote the paper.

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

[1] J. Parker, C. W. Peterson, Y. Yifat, S. A. Rice, Z. Yan, S. K. Gray, and N. F. Scherer, Optical matter machines: Angular momentum conversion by collective modes in optically bound nanoparticle arrays, Optica **7**, 1341 (2020).

[2] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, 2010).

[3] R. Alicki and R. Kosloff, Introduction to quantum thermodynamics: History and prospects, in *Thermodynamics in the Quantum Regime* (Springer, Berlin, 2018), pp. 1–33.

[4] S. Bhattacharjee and A. Dutta, Quantum thermal machines and batteries, Eur. Phys. J. B **94**, 1 (2021).

[5] G. Manzano, F. Plastina, and R. Zambrini, Optimal work extraction and thermodynamics of quantum measurements and correlations, Phys. Rev. Lett. **121**, 120602 (2018).

[6] L. Bresque, P. A. Camati, S. Rogers, K. Murch, A. N. Jordan, and A. Auffèves, Two-qubit engine fueled by entanglement and local measurements, Phys. Rev. Lett. **126**, 120605 (2021).

[7] S. Seah, S. Nimmrichter, and V. Scarani, Maxwell's lesser demon: A quantum engine driven by pointer measurements, Phys. Rev. Lett. **124**, 100603 (2020).

[8] K. V. Hovhannisyan, M. Perarnau-Llobet, M. Huber, and A. Acín, Entanglement generation is not necessary for optimal work extraction, Phys. Rev. Lett. **111**, 240401 (2013).

[9] M. Perarnau-Llobet, K. V. Hovhannisyan, M. Huber, P. Skrzypczyk, N. Brunner, and A. Acín, Extractable work from correlations, Phys. Rev. X **5**, 041011 (2015).

[10] J. P. Pekola, B. Karimi, G. Thomas, and D. V. Averin, Supremacy of incoherent sudden cycles, Phys. Rev. B **100**, 085405 (2019).

[11] Y. Zou, Y. Jiang, Y. Mei, X. Guo, and S. Du, Quantum heat engine using electromagnetically induced transparency, Phys. Rev. Lett. **119**, 050602 (2017).

[12] Q. Bouton, J. Nettersheim, S. Burgardt, D. Adam, E. Lutz, and A. Widera, A quantum heat engine driven by atomic collisions, Nat. Commun. **12**, 2063 (2021).

[13] J. Roßnagel, S. T. Dawkins, K. N. Tolazzi, O. Abah, E. Lutz, F. Schmidt-Kaler, and K. Singer, A single-atom heat engine, Science **352**, 325 (2016).

[14] G. Maslennikov, S. Ding, R. Hablützel, J. Gan, A. Roulet, S. Nimmrichter, J. Dai, V. Scarani, and D. Matsukevich, Quantum absorption refrigerator with trapped ions, Nat. Commun. **10**, 202 (2019).

[15] N. Van Horne, D. Yum, T. Dutta, P. Hänggi, J. Gong, D. Poletti, and M. Mukherjee, Single-atom energy-conversion device with a quantum load, npj Quantum Inf. **6**, 37 (2020).

[16] J. Klatzow, J. N. Becker, P. M. Ledingham, C. Weinzetl, K. T. Kaczmarek, D. J. Saunders, J. Nunn, I. A. Walmsley, R. Uzdin, and E. Poem, Experimental demonstration of quantum effects in the operation of microscopic heat engines, Phys. Rev. Lett. **122**, 110601 (2019).

[17] K. Ono, S. N. Shevchenko, T. Mori, S. Moriyama, and F. Nori, Analog of a quantum heat engine using a single-spin qubit, Phys. Rev. Lett. **125**, 166802 (2020).

[18] W. Ji, Z. Chai, M. Wang, Y. Guo, X. Rong, F. Shi, C. Ren, Y. Wang, and J. Du, Spin quantum heat engine quantified by quantum steering, Phys. Rev. Lett. **128**, 090602 (2022).

[19] P. A. Camati, J. P. S. Peterson, T. B. Batalhão, K. Micadei, A. M. Souza, R. S. Sarthour, I. S. Oliveira, and R. M. Serra, Experimental rectification of entropy production by Maxwell's demon in a quantum system, Phys. Rev. Lett. **117**, 240502 (2016).

[20] J. P. S. Peterson, T. B. Batalhão, M. Herrera, A. M. Souza, R. S. Sarthour, I. S. Oliveira, and R. M. Serra, Experimental characterization of a spin quantum heat engine, Phys. Rev. Lett. **123**, 240601 (2019).

[21] R. J. de Assis, T. M. de Mendonça, C. J. Villas-Boas, A. M. de Souza, R. S. Sarthour, I. S. Oliveira, and N. G. de Almeida, Efficiency of a quantum Otto heat engine operating under a reservoir at effective negative temperatures, Phys. Rev. Lett. **122**, 240602 (2019).

[22] J. V. Koski, V. F. Maisi, T. Sagawa, and J. P. Pekola, Experimental observation of the role of mutual information in the nonequilibrium dynamics of a Maxwell demon, Phys. Rev. Lett. **113**, 030601 (2014).

[23] J. V. Koski, V. F. Maisi, J. P. Pekola, and D. V. Averin, Experimental realization of a Szilard engine with a single electron, Proc. Natl. Acad. Sci. USA **111**, 13786 (2014).

[24] J. V. Koski, A. Kutvonen, I. M. Khaymovich, T. Ala-Nissila, and J. P. Pekola, On-chip Maxwell's demon as an information-powered refrigerator, Phys. Rev. Lett. **115**, 260602 (2015).

[25] G. Manzano, D. Subero, O. Maillet, R. Fazio, J. P. Pekola, and É. Roldán, Thermodynamics of gambling demons, Phys. Rev. Lett. **126**, 080603 (2021).

[26] B. Karimi and J. P. Pekola, Otto refrigerator based on a superconducting qubit: Classical and quantum performance, Phys. Rev. B **94**, 184503 (2016).

[27] G. Watanabe, B. P. Venkatesh, P. Talkner, and A. Del Campo, Quantum performance of thermal machines over many cycles, Phys. Rev. Lett. **118**, 050601 (2017).

[28] S. Deffner, Efficiency of harmonic quantum Otto engines at maximal power, Entropy **20**, 875 (2018).

[29] A. Das and V. Mukherjee, Quantum-enhanced finite-time Otto cycle, Phys. Rev. Res. **2**, 033083 (2020).

[30] R. Uzdin, A. Levy, and R. Kosloff, Equivalence of quantum heat machines, and quantum-thermodynamic signatures, Phys. Rev. X **5**, 031044 (2015).

[31] J. Jaramillo, M. Beau, and A. del Campo, Quantum supremacy of many-particle thermal machines, New J. Phys. **18**, 075019 (2016).

[32] V. Cavina, A. Mari, A. Carlini, and V. Giovannetti, Optimal thermodynamic control in open quantum systems, Phys. Rev. A **98**, 012139 (2018).

[33] K. Brandner, M. Bauer, and U. Seifert, Universal coherence-induced power losses of quantum heat engines in linear response, Phys. Rev. Lett. **119**, 170602 (2017).

[34] M. Esposito, R. Kawai, K. Lindenberg, and C. Van den Broeck, Efficiency at maximum power of low-dissipation Carnot engines, Phys. Rev. Lett. **105**, 150603 (2010).

[35] J. H. Wang, J. Z. He, and X. He, Performance analysis of a two-state quantum heat engine working with a single-mode radiation field in a cavity, Phys. Rev. E **84**, 041127 (2011).

[36] V. Cavina, A. Mari, and V. Giovannetti, Slow dynamics and thermodynamics of open quantum systems, Phys. Rev. Lett. **119**, 050601 (2017).

[37] P. Abiuso and V. Giovannetti, Non-Markov enhancement of maximum power for quantum thermal machines, Phys. Rev. A **99**, 052106 (2019).

[38] P. Abiuso and M. Perarnau-Llobet, Optimal cycles for low-dissipation heat engines, Phys. Rev. Lett. **124**, 110606 (2020).

[39] V. Cavina, P. A. Erdman, P. Abiuso, L. Tolomeo, and V. Giovannetti, Maximum-power heat engines and refrigerators in the fast-driving regime, Phys. Rev. A **104**, 032226 (2021).

[40] T. Villazon, A. Polkovnikov, and A. Chandran, Swift heat transfer by fast-forward driving in open quantum systems, Phys. Rev. A **100**, 012126 (2019).

[41] R. Dann and R. Kosloff, Quantum signatures in the quantum Carnot cycle, New J. Phys. **22**, 013055 (2020).

[42] H. T. Quan, Y. X. Liu, C. P. Sun, and F. Nori, Quantum thermodynamic cycles and quantum heat engines, Phys. Rev. E **76**, 031105 (2007).

[43] A. E. Allahverdyan, K. V. Hovhannisyan, A. V. Melkikh, and S. G. Gevorkian, Carnot cycle at finite power: Attainability of maximal efficiency, Phys. Rev. Lett. **111**, 050601 (2013).

[44] M. Campisi and R. Fazio, The power of a critical heat engine, Nat. Commun. **7**, 11895 (2016).

[45] J. F. Chen, C. P. Sun, and H. Dong, Boosting the performance of quantum Otto heat engines, Phys. Rev. E **100**, 032144 (2019).

[46] P. A. Camati, J. F. G. Santos, and R. M. Serra, Coherence effects in the performance of the quantum Otto heat engine, Phys. Rev. A **99**, 062103 (2019).

[47] Z. Fei, J.-F. Chen, and Y.-H. Ma, Efficiency statistics of a quantum Otto cycle, Phys. Rev. A **105**, 022609 (2022).

[48] J. W. Deng, Q. H. Wang, Z. H. Liu, P. Hanggi, and J. B. Gong, Boosting work characteristics and overall heat-engine performance via shortcuts to adiabaticity: Quantum and classical systems, Phys. Rev. E **88**, 062122 (2013).

[49] B. Cakmak and O. E. Mustecaplioglu, Spin quantum heat engines with shortcuts to adiabaticity, Phys. Rev. E **99**, 032108 (2019).

[50] K. Funo, N. Lambert, B. Karimi, J. P. Pekola, Y. Masuyama, and F. Nori, Speeding up a quantum refrigerator via counterdiabatic driving, Phys. Rev. B **100**, 035407 (2019).

[51] O. Abah and M. Paternostro, Shortcut-to-adiabaticity Otto engine: A twist to finite-time thermodynamics, Phys. Rev. E **99**, 022110 (2019).

[52] R. Kosloff and Y. Rezek, The quantum harmonic Otto cycle, Entropy **19**, 136 (2017).

[53] S. J. Deng, A. Chenu, P. P. Diao, F. Li, S. Yu, I. Coulamy, A. del Campo, and H. B. Wu, Superadiabatic quantum friction suppression in finite-time thermodynamics, Sci. Adv. **4**, eaar5909 (2018).

[54] E. Torrontegui, S. Ibáñez, S. Martínez-Garaot, M. Modugno, A. del Campo, D. Guéry-Odelin, A. Ruschhaupt, X. Chen, and J. G. Muga, Shortcuts to adiabaticity, in *Advances in Atomic, Molecular, and Optical Physics* (Elsevier, Amsterdam, 2013), Vol. 62, Chap. 2, pp. 117–169.

[55] N. Suri, F. C. Binder, B. Muralidharan, and S. Vinjanampathy, Speeding up thermalisation via open quantum system variational optimisation, Eur. Phys. J. Spec. Top. **227**, 203 (2018).

[56] P. Menczel, T. Pyharanta, C. Flindt, and K. Brandner, Two-stroke optimization scheme for mesoscopic refrigerators, Phys. Rev. B **99**, 224306 (2019).

[57] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction* (MIT Press, Cambridge, MA, 2018).

[58] V. Mnih *et al.*, Human-level control through deep reinforcement learning, Nature (London) **518**, 529 (2015).

[59] O. Vinyals *et al.*, Grandmaster level in StarCraft II using multi-agent reinforcement learning, Nature (London) **575**, 350 (2019).

[60] D. Silver *et al.*, Mastering the game of Go without human knowledge, Nature (London) **550**, 354 (2017).

[61] T. Haarnoja, S. Ha, A. Zhou, J. Tan, G. Tucker, and S. Levine, Learning to walk via deep reinforcement learning, arXiv:1812.11103.

[62] L. Ouyang *et al.*, Training language models to follow instructions with human feedback, Adv. Neural Inf. Processing Syst. **35**, 27730 (2022).

[63] M. Bukov, A. G. R. Day, D. Sels, P. Weinberg, A. Polkovnikov, and P. Mehta, Reinforcement learning in different phases of quantum control, Phys. Rev. X **8**, 031086 (2018).

[64] Z. An and D. L. Zhou, Deep reinforcement learning for quantum gate control, Europhys. Lett. **126**, 60002 (2019).

[65] M. Dalgaard, F. Motzoi, J. J. Sørensen, and J. Sherson, Global optimization of quantum dynamics with AlphaZero deep exploration, npj Quantum Inf. **6**, 6 (2020).

[66] J. Mackeprang, D. B. R. Dasari, and J. Wrachtrup, A reinforcement learning approach for quantum state engineering, Quantum Mach. Intell. **2**, 5 (2020).

[67] M. Y. Niu, S. Boixo, V. N. Smelyanskiy, and H. Neven, Universal quantum control through deep reinforcement learning, npj Quantum Inf. **5**, 33 (2019).

[68] X.-M. Zhang, Z. Wei, R. Asad, X.-C. Yang, and X. Wang, When does reinforcement learning stand out in quantum control? A comparative study on state preparation, npj Quantum Inf. **5**, 85 (2019).

[69] P. Sgroi, G. M. Palma, and M. Paternostro, Reinforcement learning approach to nonequilibrium quantum thermodynamics, Phys. Rev. Lett. **126**, 020601 (2021).

[70] R. Sweke, M. S. Kesselring, E. P. L. van Nieuwenburg, and J. Eisert, Reinforcement learning decoders for fault-tolerant quantum computation, Mach. Learn.: Sci. Technol. **2**, 025005 (2020).

[71] F. S. Luiz, A. de Oliveira Junior, F. F. Fanchini, and G. T. Landi, Machine classification for probe-based quantum thermometry, Phys. Rev. A **105**, 022413 (2022).

[72] P. A. Erdman and F. Noé, Identifying optimal cycles in quantum thermal machines with reinforcement-learning, npj Quantum Inf. **8**, 1 (2022).

[73] P. A. Erdman and F. Noé, Model-free optimization of power/efficiency tradeoffs in quantum thermal machines using reinforcement learning, PNAS nexus **2**, pgad248 (2023).

[74] I. Khait, J. Carrasquilla, and D. Segal, Optimal control of quantum thermal machines using machine learning, Phys. Rev. Res. **4**, L012029 (2022).

[75] D. Chruściński and S. Pascazio, A brief history of the GKLS equation, Open Sys. Inf. Dynamics **24**, 1740001 (2017).

[76] P. Bayona-Pena and K. Takahashi, Thermodynamics of a continuous quantum heat engine: Interplay between population and coherence, Phys. Rev. A **104**, 042203 (2021).

[77] O. Delalleau, M. Peter, E. Alonso, and A. Logut, Discrete and continuous action representation for practical RL in video games, arXiv:1912.11077.

[78] T. Haarnoja *et al.*, Soft actor-critic algorithms and applications, arXiv:1812.05905.

[79] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor, in *Proceedings of the 35th International Conference on Machine Learning*, Vol. PMLR 80 (ML Research Press, Maastricht, 2018), pp. 1861–1870.

[80] P. Christodoulou, Soft actor-critic for discrete action settings, arXiv:1910.07207.

[81] F. L. Curzon and B. Ahlborn, Efficiency of a Carnot engine at maximum power output, Am. J. Phys. **43**, 22 (1975).

[82] K. E. Dorfman, D. Xu, and J. Cao, Efficiency at maximum power of a laser quantum heat engine enhanced by noise-induced coherence, Phys. Rev. E **97**, 042120 (2018).

[83] J.-F. Chen, C.-P. Sun, and H. Dong, Achieve higher efficiency at maximum power with finite-time quantum Otto cycle, Phys. Rev. E **100**, 062140 (2019).

[84] Y.-H. Ma, D. Xu, H. Dong, and C.-P. Sun, Universal constraint for efficiency and power of a low-dissipation heat engine, Phys. Rev. E **98**, 042112 (2018).

[85] X. Zhang, F. Zhou, Z. Liu, Z. Zhang, Y. Qin, S. Zhuo, X. Luo, E. Gao, and H. Li, Quadruple plasmon-induced transparency of polarization desensitization caused by the Boltzmann function, Opt. Express **29**, 29387 (2021).

[86] S. Vinjanampathy and J. Anders, Quantum thermodynamics, Contemp. Phys. **57**, 545 (2016).

[87] P. A. Erdman, V. Cavina, R. Fazio, F. Taddei, and V. Giovannetti, Maximum power and corresponding efficiency for two-level heat engines and refrigerators: Optimality of fast cycles, New J. Phys. **21**, 103049 (2019).

[88] G.-x. Deng, W. Shao, Y. Liu, and Z. Cui, Continuous three-level quantum heat engine with high performance under medium temperature difference, J. Appl. Phys. **133**, 124903 (2023).