Extended Abstracts of the 2023 International Conference on Solid State Devices and Materials, Nagoya, 2023, pp71-72

Construction of an Environment Model for Auto-tuning Quantum Dot Devices Using Model-based Reinforcement Learning

Chihiro Kondo[1,*], Raisei Mizokuchi[1], Jun Yoneda[1], and Tetsuo Kodera[1]

[1] Tokyo Institute of Technology
2-12-1 O-okayama, Meguro-ku, Tokyo 152-8552, Japan
Phone :+81-3-5734-2508,
*E-mail: kondo.c.af@m.titech.ac.jp

Abstract

Semiconductor quantum dots (QDs) are a promising host for quantum computers because of their scalability. However, as the number of QDs grows, the time required to tune the potential increases, hampering scaling up. Machine learning is a promising approach to automate and expedite this tuning process. We propose to use model-based reinforcement learning (MBRL) for auto-tuning QDs. MBRL is expected to offer more generality because it models the environment and can divert the constructed model for other tasks. However, it remains to be seen whether the environment model can be constructed properly despite the sparse characteristic of QDs. In this work, we investigate the applicability of MBRL in this regard by emulating auto-tuning of a QD device to a single QD condition using MBRL on pre-measured data.

1. Introduction

Semiconductor QDs are promising for dense qubit integration because of their small footprint and compatibility with semiconductor technologies. One of the severe impediments to explore novel QD structures or materials is the labor-intensive potential tuning process required to make the devices function as qubits. To partially automate the tuning, machine learning techniques have been studied [1,2]; however, these efforts have been limited to specific tuning tasks or single device structures.

To overcome the limitation in the previous works, we propose a MBRL system for QD tuning. In MBRL, a model for the environment is constructed and is used for learning. Since this model can be diverted for other tasks and/or similar environments, we expect that the MBRL approach yields tuning protocols with greater generality.

Tuning of QDs is usually accomplished by finding specific target patterns in charge stability diagrams obtained by sweeping two gate voltages or in one-dimensional characteristics by ray-based method. Unfortunately, such target patterns are only sparsely distributed. Having to discover this sparse reward signal may be an obstacle to constructing the environment model in automating the tuning process using MBRL. In this work, we verify proper construction of an environment model and successful learning based on the constructed model – two first key steps towards MBRL-based QD tuning – by using a pre-measured charge stability diagram.

2. Learning system

Reinforcement learning (RL) is an area of machine learning concerned with behavior within a certain environment. In a RL framework, the "agent" (i.e., the learning system) interacts with the "environment" (i.e., a QD device in our case) and learns the "action" that maximizes "reward" obtained from the environment. Figure 1 shows our MBRL system. It consists of the following four major steps: (i) the agent measures a small charge stability diagram (red square in Fig. 1) that partially characterizes the QD and obtains its corresponding reward; (ii) the agent updates the construction of environment model based on the measurement results and the rewards; (iii) the agent learns the relation between actions and rewards in the constructed environment model many times, which is faster than in non-MBRL systems because in MBRL systems the agent does not interact with the environment itself through time-consuming measurements at this step; (iv) the agent determines which area will be measured in the next action to get higher reward based on the current learning situation. This cycle of four steps is repeated 5×10^6 times. We employ a neural-network reinforcement learning framework, "DreamerV2," as the algorithm for our agent. It is developed by DeepMind and outperforms the top single-GPU agents like Rainbow and IQN [3]. In this work, the task of the agent is to tune a multi-dot device to a single-QD state. For proof-of-concept of MBRL auto-tuning, we use a pre-measured wide-range charge stability diagram as environment for simplicity. It takes a couple of days to complete the entire learning cycle on a computer with a single GPU (RTX 2070 SUPER, NVIDIA).

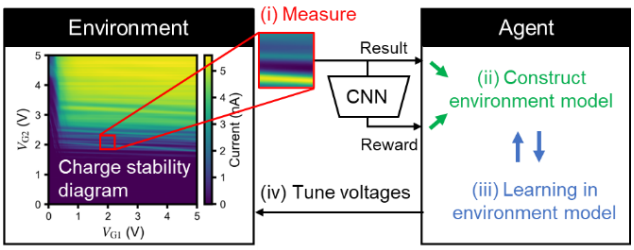

Fig. 1 Learning system. At the beginning of the cycle, the agent performs local characterization of the QD (red square). Next, reward is determined by image classification using CNN. Agent learns from measurement result and reward. Finally, the agent moves in the charge stability diagram to get more reward from the current learning situation, which corresponds to tuning of the gate voltages.

3. Reward determination by image classification

In RL, reward plays an important role because the agent aims to maximize it and decides its action based on the reward predictions. In contrast to video games like Atari [4], where environment outputs a score that plays the role of reward, in QD measurements, rewards are not output per se. In order to evaluate the reward, we use image classification with convolutional neural network (CNN). In QD measurements, we know what kind of patterns in a stability diagram is expected for single QD characteristic (stripes with a negative slope); therefore, we calculate the reward based on the similarity to computer-generated "target patterns". We trained CNN with supervised learning for the classification task with the target pattern dataset (5000 images) and CIFAR-10 as dummy dataset (5000 images) [5]. These datasets are divided into training dataset (7000 images) and test dataset (3000 images). The training dataset is used for training the CNN, and the test dataset is used for evaluation only. As shown in Fig. 2, trained CNN achieved 99% accuracy with the test dataset. We use this trained CNN for the automatic reward determination in the MBRL system. The CNN gives the agent a base reward proportional to its confidence level (between 0 and 1) that the measured result has an expected characteristic. Additionally, a reward of +100 points is given upon reaching the goal, while a penalty of -100 points is issued when the gate voltages exceed the pre-determined limits.

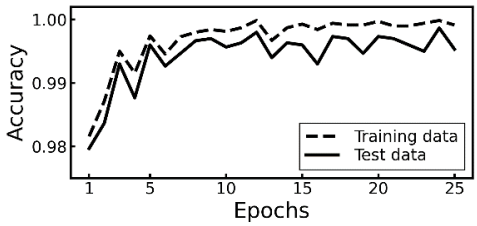

Fig. 2 Learning result of CNN with supervised learning for the classification task of whether or not a device is in a single QD region. Training and test data have 7000 and 3000 images, respectively. All images in the datasets are used in each epoch.

4. Result

Now the agent is ready to construct the environment model and learn from it. As an initial investigation of the feasibility of MBRL-based QD tuning, we perform two types of tests. First, we evaluate the constructed environment model by checking its reward prediction (Fig. 3). In this evaluation, the agent initially starts measurements at a given condition and then changes the measurement position, looking for better rewards. To decide the next action, the agent predicts rewards around the present measurement position. Figure 3 shows the predicted reward averaged over 1000 runs. Areas with high predicted rewards are roughly consistent with the single QD region identified by human eyes. This suggests that a proper environment model is constructed.

Second, we examine the learning in the environment model. Figure 4 (a) shows the episode reward, that is, the cumulative reward of each run. As learning progresses, the episode reward acquired by the agent increases and saturates at 100, meaning that the goal is regularly achieved. In the control experiment with random action selection, this value is much lower (~-70). The trajectories on the charge stability diagram during auto-tuning (Fig. 4 (b)) are reasonable in the sense that they smoothly tend towards the target region without significant detour. We note that the tuning is efficient, with the average number of measurements in these runs roughly 23 times (corresponding to ~3 minutes when converted into lab time). These results indicate that the agent was able to learn behavior in the environment model.

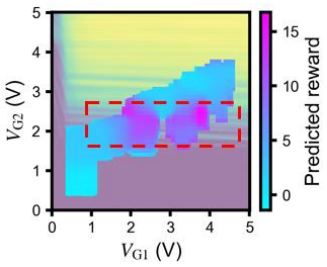

Fig. 3 Predicted rewards of the environment model plotted on top of the stability diagram. Reward prediction is output only in areas where the agent has passed. Red dashed square indicates the single QD region identified by human eyes.

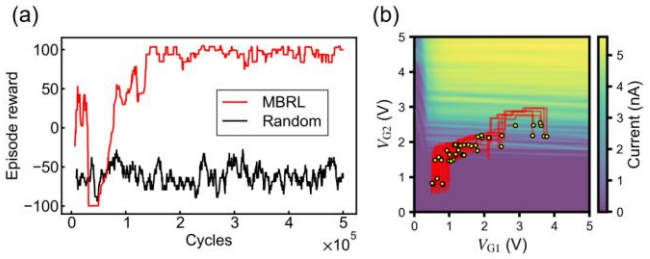

Fig. 4 (a) Evolution of episode reward. Red and black traces are for the agent and random action selection, respectively. (b) The agent's measurement trajectories on the charge stability diagram. The trajectories of 50 runs are shown.

5. Conclusions

We applied MBRL to the task of auto-tuning of a QD device to the single-QD region and investigated the applicability of MBRL in QD measurements. The determination of reward was automated by image classification using CNN. The results suggest that an appropriate environment model was constructed and that the agent successfully learned in the environment model. These are the first key steps towards MBRL-based QD tuning and support prospects of MBRL for more general QD auto-tuning technique.

Acknowledgements

This work was financially supported by MEXT Quantum Leap Flagship Program (MEXT QLEAP) grant no. JPMXS0118069228, JST Moonshot R&D grant no. JPMJMS2065, JST PRESTO grant no. JPMJPR21BA, and JSPS KAKENHI grant nos. JP23H05455 and JP23H01790.

References

[1] H. Moon *et al.*, Nat. Commun. 11, 4161 (2020).
[2] V. Nguyen *et al.*, NPJ Quantum Inf. 7, 100 (2021).
[3] D. Hafner *et al.*, arXiv:2010.02193 (2020).
[4] M. G. Bellemare *et al.*, arXiv:1207.4708 (2012).
[5] A. Krizhevsky, Technical report (2009).