Efficient quantum recurrent reinforcement learning via quantum reservoir computing

Samuel Yen-Chi Chen
Wells Fargo, New York, NY 10017, USA

***Abstract*—Quantum reinforcement learning (QRL) has emerged as a framework to solve sequential decision-making tasks, showcasing empirical quantum advantages. A notable development is through quantum recurrent neural networks (QRNNs) for memory-intensive tasks such as partially observable environments. However, QRL models incorporating QRNN encounter challenges such as inefficient training of QRL with QRNN, given that the computation of gradients in QRNN is both computationally expensive and time-consuming. This work presents a novel approach to address this challenge by constructing QRL agents utilizing QRNN-based reservoirs, specifically employing quantum long short-term memory (QLSTM). QLSTM parameters are randomly initialized and fixed without training. The model is trained using the asynchronous advantage actor-critic (A3C) algorithm. Through numerical simulations, we validate the efficacy of our QLSTM-Reservoir RL framework. Its performance is assessed on standard benchmarks, demonstrating comparable results to a fully trained QLSTM RL model with identical architecture and training settings.**

I. Motivation

Quantum computing (QC) offers potential for superior performance in complex computational tasks compared to classical methods. However, the absence of error correction in current quantum computers complicates deep quantum circuit implementation. These noisy intermediate-scale quantum (NISQ) devices necessitate specialized quantum circuit designs to maximize their benefits. A recent hybrid quantum-classical computing approach utilizes both systems, assigning quantum computers to advantageous tasks while classical computers handle tasks such as gradient calculations. Known as *variational quantum algorithms*, these methods have excelled in specific machine learning (ML) tasks. Reinforcement learning (RL), a subset of ML focused on sequential decision-making, has achieved remarkable success using deep neural networks. In contrast, the nascent field of quantum reinforcement learning (QRL) presents under-explored challenges.

II. Challenge

Most QRL approaches have traditionally emphasized variational quantum circuits (VQCs) devoid of recurrent connections. However, recent advancements, exemplified by the work presented in [1], have introduced Quantum Recurrent Neural Networks (QRNNs). These QRNNs show significant potential in partially observable environments, surpassing the performance of classical models. Nevertheless, the training of QRNNs remains a challenging task due to the computationally intensive nature of their gradient calculations. The calculation of gradients for a quantum function adheres to the parameter-shift rule [2], necessitating two quantum circuit evaluations per data point and per circuit parameter. Consequently, the number of required quantum circuit evaluations for gradient computation scales rapidly as $\mathcal{O}(NM)$, where N denotes the number of data points and M denotes the number of trainable parameters in the quantum circuit. This complexity is exacerbated when variational quantum circuits (VQCs) are interconnected to form QRNNs, such as Quantum Long Short-Term Memory (QLSTM) networks [3]. The gradient computation for QRNNs requires the application of backpropagation-through-time (BPTT), further increasing computational demands.

III. Approach

To address the challenge of gradient calculation, we propose the QLSTM-RC-RL framework. This approach leverages QLSTM [3] and reservoir computing (RC) [4] to optimize QRL training with a QRNN-based reservoir. In this framework, QLSTM functions as an untrained *reservoir*, with training applied solely to the classical neural network components preceding and succeeding it. The scheme is depicted in Figure 1 . Additionally, we enhance the training process through asynchronous methods as developed in [5, 6], utilizing existing multi-core CPU resources and, in the future, multiple-QPU systems. This asynchronous training accelerates the process and diversifies the trajectories encountered by the agent. The QLSTM,

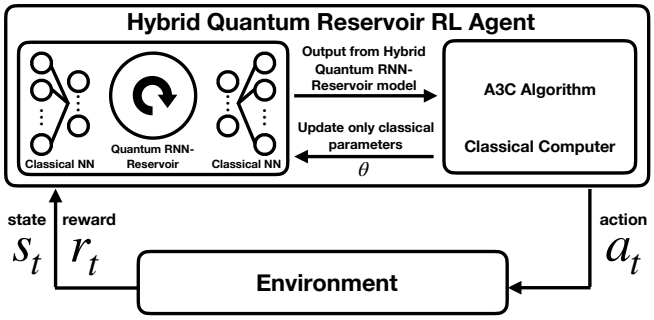

Fig. 1: **The hybrid quantum-classical framework for QLSTM-RC-RL.**

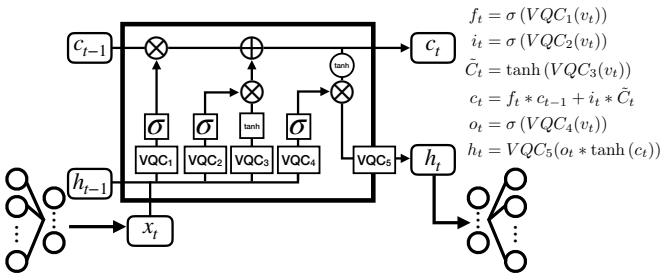

Fig. 2: **The QLSTM-Reservoir used in the proposed QA3C.** QLSTM is first proposed in the work [3]. In the proposed QLSTM-RC-RL framework, the VQC parameters are not trained. The input to the QLSTM is the concatenation v_t of the hidden state h_{t-1} from the previous time step and the current input vector x_t which is the processed observation from the environment.

depicted in Figure 2 and introduced by Chen et al. in [3], is a quantum adaptation of LSTM [7]. It employs VQCs instead of classical neural networks and excels in both time-series data and NLP

tasks [3, 8, 9]. The VQC incorporated into the QLSTM, as depicted in Figure 3, has shown impressive performance in time-series modeling [3]. It encompasses data encoding via R_y and R_z rotations, a variational component involving CNOT gates for qubit entanglement, trainable unitary R gates, and quantum measurement. The original QLSTM proposal, applicable to time-series modeling [3] and QRL [1], involves time-consuming training of VQC parameters. In this work, we adopt an approach using QLSTM as a reservoir to transform input data without the need for explicit VQC parameter training, as detailed in [10]. The proposed QLSTM-RC-RL framework includes

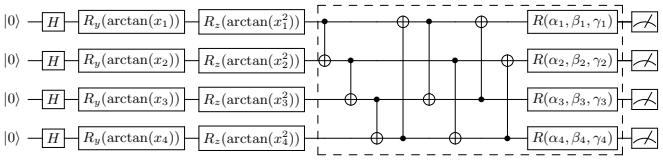

Fig. 3: **VQC architecture for QLSTM.** The VQC architecture here is inspired by the work [11]. The parameters α, β, γ are not trained in QLSTM-RC settings.

a *dressed QLSTM model* consisting of classical neural networks for preprocessing and postprocessing, with a QLSTM in between. We adopted the quantum asynchronous advantage actor-critic (QA3C) training method developed in the work [6]. Furthermore, we extend the original QA3C method to include the recurrent policy QLSTM. During asynchronous training, each local agent interacts with its own environment and stores the current trajectory in local memory. This trajectory is later used to calculate local gradients, which are then uploaded to the global shared model for updating.

IV. Results and Impact

a) Environment: In this study, we employ the MiniGrid-Empty environment, a widely utilized maze navigation scenario [12]. The primary objective for our QRL agent is to effectively generate appropriate actions sequence based on the observations it receives at each time step, enabling it to traverse from the initial location to the designated destination, represented as the green box in Figure 4. Notably, the MiniGrid-Empty environment is characterized by a 147-dimensional vector observation, denoted as s_t. It offers an action space $\mathcal{A}$ comprising six actions, namely *turn left*, *turn right*, *move forward*, *pick up an object*, *drop the object being carried*, and *toggle*. Of these actions, only the first three have practical consequences in this context, and the agent is expected to learn this distinction. Moreover, successful navigation to the goal rewards the agent with a score of 1, albeit subject to a penalty determined by the formula $1 - 0.9 \times (\textit{number of steps}/\textit{max steps allowed})$, with the maximum allowable steps set at $4 \times n \times n$, where n is the grid size [12]. Throughout our experimentation, we explore various configurations, encompassing different grid sizes and variations in the initial starting points.

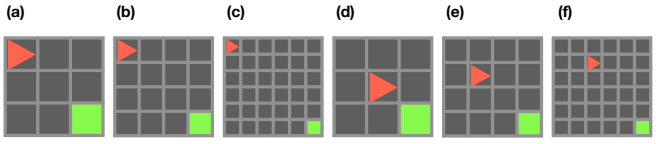

Fig. 4: **The MiniGrid environments.** (a) - (c) are MiniGrid environments with fixed starting points and (d) - (f) are with random starting points (starting points shown in (d) - (f) are a set of examples).

b) Hyperparameters: The hyperparameters for the proposed QLSTM-RC-RL are: Adam with learning rate: 1×10^{-4}, $beta_1 = 0.92$, $beta_2 = 0.999$, model lookup steps $L = 5$ and discount factor $\gamma = 0.9$. The local agents/models calculate their own gradients for classical NN every L steps (the length of trajectory used during model updates) and update the model following the QA3C method. The number of parallel processes (local agents) is 80.

c) Model Sizes: In our study, we employ hybrid QLSTM models composed of four key components: a classical NN for environmental observation preprocessing, a QLSTM that can be fully trained or initialized randomly and fixed in the RC scenario, and two classical NNs for processing QLSTM outputs to produce action logits and state values. In our study, we utilize an 8-qubit VQC-based QLSTM model with input and hidden dimensions of 4. The internal state is 8-dimensional. We explore QLSTM variations with 1, 2, and 4 VQC layers, as shown in dashed box in Figure 3 All hybrid models share identical configurations for their classical neural networks. Specifically, the preprocessing NN consists of $147 \times 4 + 4 = 592$ parameters, the NN for action logits has $4 \times 6 + 6 = 30$ parameters, and the NN for state values comprises $4 \times 1 + 1 = 5$ parameters. The number of parameters of QLSTM with n VQC layer is: $8 \times 3 \times 5 \times n = 120n$ in which the VQCs are 8-qubit and each general rotation gate is parameterized by 3 parameters. There are 5 VQCs in a QLSTM as shown in Figure 2.

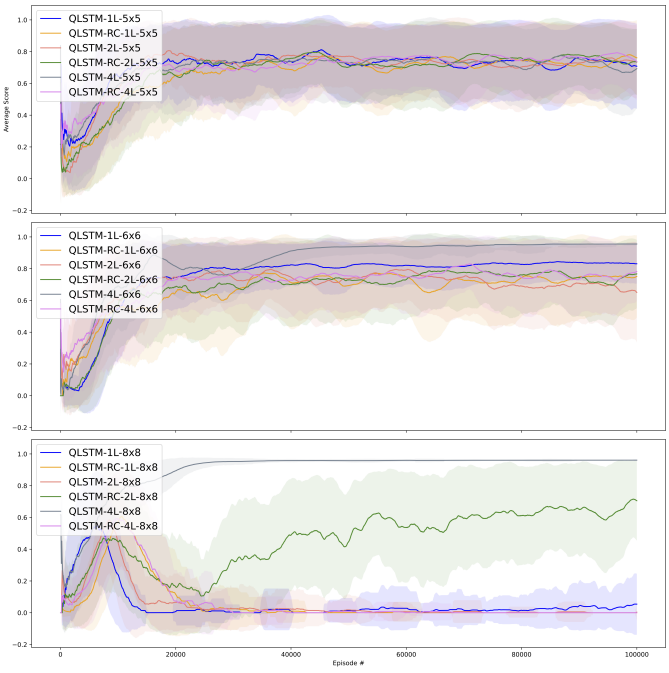

Fig. 5: **Results: QLSTM-RC-RL in MiniGrid-Empty environment with fixed starting point.**

d) Results: **MiniGrid with fixed starting point** We first consider the setting that the RL agent start from a fixed point in the environment. In the MiniGrid environment, this is at the upper-left corner of the maze as described in Section **??** and Figure 4. The results are shown in the Figure 5. We can observe that among the three environments settings we tested, QLSTM and QLSTM-RC with 1, 2 or 4 VQC layers reach similar performance in the MiniGrid-5x5. In the MiniGrid-6x6, fully-trained QLSTM with 4 VQC layers achieve the best performance. Other models still achieve good performance, which are close to the best one. The QLSTM-RC performs very similar to the fully-trained QLSTM. In the most difficult MiniGrid-

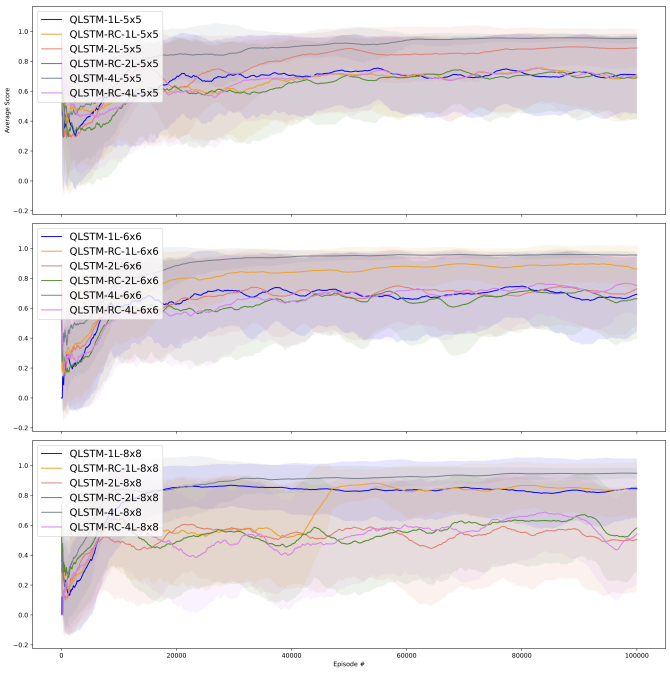

Fig. 6: **Results: QLSTM-RC-RL in MiniGrid-Empty environment with random starting point.**

8x8 case, only the fully-trained QLSTM with 4 VQC layers reaches the optimal performance. The QLSTM-RC with 2 VQC layers still learns but slowly. Other model configurations struggle to learn the good policies.

MiniGrid with random starting point We further consider the setting that, in each episode, the RL agent starts from a random point in the environment. We provide a set of examples in Figure 4. The results are shown in the Figure 6. We can observe that among the three environments settings we tested, the fully-trained QLSTM with 4 VQC layers outperform other models in all three cases. An interesting result is that the QLSTM-RC with only one VQC layer still reaches performance very close to the best performing agent. Other agents, either QLSTM or QLSTM-RC, perform very similar. Overall, the performance of QLSTM and QLSTM-RC agents in this environment are comparable or superior than in the environments with fixed starting point. A possible reason is that the agent can experience different situations more frequently. The agent may start from a location closer to the goal and achieve the goal with a positive reward. This is crucial since the MiniGrid is a sparse environment and the agent may require a large number of trial and error to obtain a positive reward in the non-random environment.

e) Conclusion: In this paper, we introduce a novel approach to reinforcement learning (RL) utilizing quantum recurrent neural network (QRNN)-based reservoir computing. Specifically, we implement a hybrid QLSTM reservoir as the function approximator to realize the quantum Asynchronous Advantage Actor-Critic (A3C) algorithm. Our experimental results across various testing environments demonstrate that the proposed framework achieves stability and average performance comparable to fully-trained counterparts, given fixed model sizes, architectures, and training hyperparameters. This method provides an efficient pathway for advancing quantum reinforcement learning (QRL) with recurrent structures.

References

[1] S. Y.-C. Chen, "Quantum deep recurrent reinforcement learning," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[2] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii, "Quantum circuit learning," *Physical Review A*, vol. 98, no. 3, p. 032309, 2018.

[3] S. Y.-C. Chen, S. Yoo, and Y.-L. L. Fang, "Quantum long short-term memory," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 8622–8626.

[4] M. Lukoševičius and H. Jaeger, "Reservoir computing approaches to recurrent neural network training," *Computer Science Review*, vol. 3, no. 3, pp. 127–149, 2009.

[5] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *International conference on machine learning*. PMLR, 2016, pp. 1928–1937.

[6] S. Y.-C. Chen, "Asynchronous training of quantum reinforcement learning," *Procedia Computer Science*, vol. 222, pp. 321–330, 2023, international Neural Network Society Workshop on Deep Learning Innovations and Applications (INNS DLIA 2023). [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050923009365

[7] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[8] R. Di Sipio, J.-H. Huang, S. Y.-C. Chen, S. Mangini, and M. Worring, "The dawn of quantum natural language processing," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 8612–8616.

[9] S. S. Li, X. Zhang, S. Zhou, H. Shu, R. Liang, H. Liu, and L. P. Garcia, "Pqlm-multilingual decentralized portable quantum language model," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[10] S. Y.-C. Chen, D. Fry, A. Deshmukh, V. Rastunkov, and C. Stefanski, "Reservoir computing via quantum recurrent neural networks," *arXiv preprint arXiv:2211.02612*, 2022.

[11] S. Y.-C. Chen, C.-H. H. Yang, J. Qi, P.-Y. Chen, X. Ma, and H.-S. Goan, "Variational quantum circuits for deep reinforcement learning," *IEEE Access*, vol. 8, pp. 141 007–141 024, 2020.

[12] M. Chevalier-Boisvert, L. Willems, and S. Pal, "Minimalistic gridworld environment for openai gym," https://github.com/maximecb/gym-minigrid, 2018.