QMARL: A QUANTUM MULTI-AGENT REINFORCEMENT LEARNING FRAMEWORK FOR SWARM ROBOTS NAVIGATION

*Weizhao Chen, Jiawang Wan, Fangwen Ye, Ran Wang, Cheng Xu**

School of Computer and Communication Engineering, University of Science and Technology Beijing

ABSTRACT

In the last decade, the field of reinforcement learning has evolved from single-agent paradigms to embrace multi-agent settings. However, as the number of agents increases, especially in intricate or stochastic environments, the efficacy of individual learning models tends to diminish. Moreover, applying experience replay techniques in multi-agent scenarios presents considerable challenges. To address these pressing issues, this paper introduces a straightforward yet highly effective approach known as Quantum-Based Multi-Agent Reinforcement Learning (QMARL). This approach revolves around the quantization of states and actions within the multi-agent reinforcement learning system. Leveraging the power of the Grover algorithm for action decision-making, we also introduce a novel quantum-based prioritized experience replay method. Our proposed approach has been rigorously validated through experiments conducted in the cooperative navigation environment provided by OpenAI. The results demonstrate its capacity to enhance multi-agent learning in complex settings. This research opens promising avenues for harnessing quantum computing techniques in the realm of reinforcement learning, paving the way for more robust and scalable solutions in multi-agent systems.

Index Terms— Multi-Agent Reinforcement Learning (MARL), Quantum Computing, Experience Replay Techniques, Grover Algorithm, Cooperative Navigation

1. INTRODUCTION

In the realm of Multi-Agent Reinforcement Learning (MARL), a critical challenge lies in the proliferation of agents [1]. As the number of agents increases, individual learning models struggle with scalability due to the escalating computational demands caused by the exponential growth of combinatorial possibilities [2]. Knowledge reuse strategies emerge as a solution, simplifying the learning process by leveraging prior knowledge in novel tasks, thus extending the reach of MARL to intricate problem domains [3].

Parameter sharing, a proven technique in various applications such as communication learning [4], agent modeling, and cooperative games in partially observable environments [5], has gained prominence. Additionally, the success of reinforcement learning methods, like deep Q-networks [6], hinges on an experience replay mechanism. Yet, employing experience replay in a multi-agent context is far from straightforward, as past experiences risk obsolescence due to evolving agent policies over time [1].

With the rapid development and widespread adoption of quantum computing technology, researchers have embarked on the integration of quantum computing with fields like machine learning and artificial intelligence. Current quantum reinforcement learning research falls into two primary categories. The first category harnesses quantum algorithms to enhance the efficiency of reinforcement learning. As early as 2008, Dong et al. [7] demonstrated that combining quantum algorithm characteristics could enhance traditional reinforcement learning algorithms, leading to a novel approach that blends quantum collapse and Grover's algorithm. The second category explores quantum-inspired interaction methods and classical environment quantization theories to devise innovative quantum reinforcement learning frameworks for efficiency enhancement [8-10].

Nonetheless, the majority of these studies concentrate on optimizing single-agent reinforcement learning techniques, with limited emphasis on optimization methods tailored specifically for multi-agent reinforcement learning using quantum paradigms. In this study, we delve into a novel Quantum-based Multi-Agent Reinforcement Learning (Q-MARL) method. The primary contributions of this paper are summarized as follows:

1) Quantum-Based Action Decision Method: We propose a quantum-based action decision method that represents states and actions in the multi-agent reinforcement learning system using quantum encoding, enriching the expression of state-action pairs through quantum properties. Action decisions are made employing the Grover algorithm, offering a quantum advantage in decision-making.

2) Quantum-Based Prioritized Experience Replay: To address the slow convergence issue in traditional reinforcement learning for path planning, we introduce a quantum-based prioritized experience replay method. This method combines the Grover search algorithm with the prioritized experience replay module, enabling parallel searching of higher-priority experiences in the quantum experience pool during training.

*Correspondence: xucheng@ustb.edu.cn. This work is supported in part by the National Natural Science Foundation of China under Grant 62101029, and in part by the China Scholarship Council Award under Grant 202006465043.

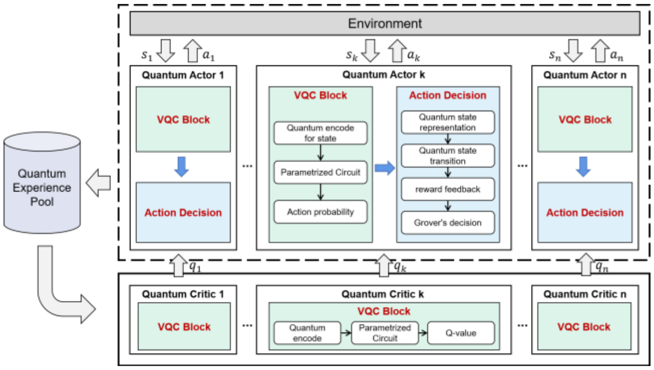

Figure 1. The framework of quantum-based multi-agent reinforcement learning (*QMARL)*.

3) **Quantum-Based Multi-Agent Deep Reinforcement Learning Framework:** We propose a comprehensive quantum-based multi-agent deep reinforcement learning framework. Leveraging the entanglement and superposition properties of quantum computing, our framework enhances the selection of high-quality actions from the set of available actions. Additionally, it retrieves a batch of more informative experiences during experience replay, which in turn aids in the training of the actor-critic networks.

In the subsequent sections, this paper elaborates on the proposed quantum-based multi-agent reinforcement learning method. Section 2 provides an in-depth description of the framework, while Section 3 encompasses the experimental setup, results, and comparative analyses against various advanced multi-agent reinforcement learning algorithms. Finally, Section 4 offers concluding remarks summarizing the contributions and implications of our approach.

2. QUANTUM-BASED MULTI-AGENT REINFORCEMENT LEARNING

2.1. Quantum-based MARL Framework

This study involves the quantization of the state-action space in traditional reinforcement learning methods. It replaces the conventional neural network with a quantum variational circuit, integrating components such as the Grover search algorithm, a prioritized experience replay module, and action decision-making. We propose QMARL, a quantum-based multi-agent reinforcement learning framework, as depicted in Figure 1.

2.2. Quantum Actor Module

In our approach, each agent incorporates two essential components: the variational quantum circuit (VQC) part and the action decision-making part.

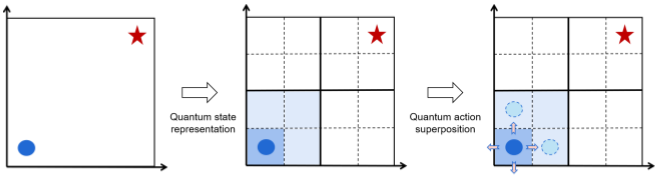

Figure 2. The schematic diagram of state and action quantization representations.

The VQC part receives the agent's observations, encoded as angles, as its input and feeds them into the quantum circuit [11]. These observations undergo processing within the parameterized network, ultimately yielding the agent's action probabilities. Subsequently, the output action probabilities are conveyed to the action decision-making component, denoted as

$$\pi_\theta(u_t \mid s_t) = \mathrm{softmax}\left(f(s_t; \theta)\right),$$

which is used for action selection reference.

In the action decision-making part, we integrate the principles of the Grover algorithm. Initially, we employ a quantum encoding scheme to represent the state-action pairs. Representing discrete states is straightforward, achieved by employing discrete quantum bits. Let the number of possible environment states be N_s. We then choose a number m satisfying the inequality $N_s \leq 2^m \leq 2N_s$ and use m qubits to represent the state set $S = \{|s_1\rangle, |s_2\rangle, |s_3\rangle, \dots \left|s_{N_s}\right|\}$.

$$\left|s^{(N_s)}\right\rangle = \sum_{i=1}^{N_s} C_i |s_i\rangle \leftrightarrow \left|s^{(m)}\right\rangle = \sum_{s=00\cdots0}^{\overbrace{11\dots1}^{m}} C_s |s\rangle$$

When the environment's state is continuous, we have the option to transform continuous positions into discrete centroids. This can be achieved by hierarchically partitioning the environment, as shown in Figure 2. This not only reduces the required number of qubits but also simplifies system operations. The spatial partitioning can be further refined according to the specific problem's resolution and accuracy requirements.

Similarly, the action space can also be represented using quantum bits.

$$\left|a_{s_i}^{(N_a)}\right\rangle = \sum_{j=1}^{N_a} C_j |a_j\rangle \leftrightarrow \left|a_s^{(n)}\right\rangle = \sum_{a=00\cdots0}^{\overbrace{11\dots1}^{a}} C_a |a\rangle$$

The quantumized state undergoes state transitions based on superpositioned quantumized actions, as illustrated in Figure 2. We employ the Grover algorithm for quantum action selection. Establishing a Grover process involves developing a suitable problem encoding and separately constructing the components of the Grover operator, namely the oracle U_ω and diffuser U_Ψ. These components facilitate amplitude amplification techniques that transform the initial uniform superposition state of the search space into a state corresponding to a solution.

The U_ω operator consists of two sub-modules: the T module and the R module. The T module performs quantum transitions by taking the state register and the superposition of actions as inputs, and it outputs the combination of all valid states to which the current state can transition. The R module, referred to as the reward feedback, identifies the target cell with a reward exceeding a certain threshold (the sum of the rewards for the output states of the T operator and the output

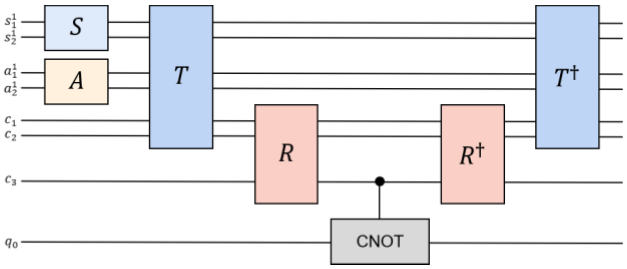

Figure 3. Block scheme of the $\mathbf{U}_\omega$ oracle for the 2 × 2 case.

values from the network) by flipping the phase of the oracle qubit. After applying the required T and R operators, information about the solution states is conveyed through quantum bits that encode the search space during non-computational stages, as depicted in Figure 3.

Following the modification of the action qubits' amplitudes using the Grover algorithm with a specific number of iterations, we observe the action qubits to determine the desired action for execution.

2.3. Quantum Experience Replay Module

In a prioritized experience replay mechanism, the priority of an experience, denoted as P_k, is defined based on a criterion that effectively identifies important and valuable experiences for replay [12]. During each training iteration, the agent interacts with the environment at time step t, retrieves necessary state and reward information, and generates a state transition by selecting an action. This sequence of states, actions, rewards, and next states, denoted as $<s_t, a_t, r_t, s_{t+1},>$, is considered as an experience. The TD-error of this experience is computed and stored together with the experience in an experience pool.

We introduce a hyperparameter, denoted as α to control the preference of sampling between uniform and greedy strategies. When α =0, uniform sampling is employed, whereas α=1 corresponds to greedy sampling. By adjusting this control value, we prioritize experiences with higher priority pk during the search process.

For an experience pool with size N, it requires ⌈log2(N) ⌉quantum bits for storage and ⌈log2(k)⌉⌉ quantum bits for indexing, encoding the pool into quantum state basis vectors. Each data entry necessitates ⌈log2(N)⌉ conditional non-quantum gates. With this encoding, each experience corresponds to an index. By utilizing Grover's search on the experience pool, we can search for experiences with higher priority. When observing the index, we obtain the experience index that satisfies our search task, as shown in Figure 4.

2.4. Quantum Critic Module

The Quantum Critic Module consists of a Quantum Critic, where we employ Concentrated Temporal Difference (CTDE) as the state-value function. States and actions are encoded using quantum angles and input into a Variational Quantum Circuit (VQC). The output is the state-action value function, which is used to update the network in the Quantum Actor. The Q-network loss function is defined as follows:

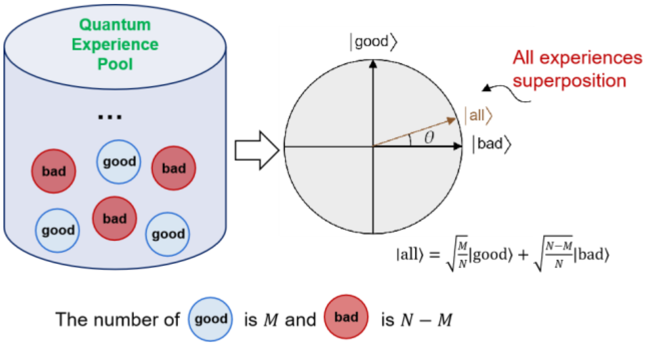
Quantum Experience Pool
...
bad
good
bad
bad
good
good
|good⟩
All experiences superposition
|all⟩
θ
|bad⟩
$|\text{all}\rangle = \sqrt{\frac{M}{N}}|\text{good}\rangle + \sqrt{\frac{N-M}{N}}|\text{bad}\rangle$
The number of good is M and bad is N − M

Figure 4. Quantum superposition processing of experiences categorized as good or bad based on TD errors.

$$y^j = r_i^j + \gamma Q_i^{\mu'}\left(x'^{\,j}, a_1', \ldots, a_N'\right)\Big|_{a_k' = \mu_k'(o_k^j)}$$

$$\mathcal{L}(\theta_i) = \frac{1}{S}\Sigma_j\left(y^j - Q_i^{\mu}\left(x^j, a_1^j, \ldots, a_N^j\right)\right)^2$$

Here, r_i represents the global reward. Compared to classical neural networks, quantum neural networks possess better expressive power and require fewer parameters for training.

3. EXPERIMENTAL ANALYSIS AND DISCUSSION

This section focuses on demonstrating the effectiveness and superiority of the quantum-based multi-agent reinforcement learning approach through concrete experiments.

3.1. Experimental Setup

Based on the Multi-Agent Particle Environment (MPE) provided by OpenAI, this study constructs the Cooperative Navigation with Obstacles (CNO) environment to investigate the quantum-based multi-agent reinforcement learning problem [13]. The CNO experimental environment is illustrated in Figure 5. The experiment is set up with three agents (circular entities) and ten obstacles (square entities). At the beginning of each episode, the coordinates of all elements in the environment are randomly initialized. The agents must collaborate and navigate to reach the landmarks (pentagon symbols) while avoiding obstacles. Agents need to observe the relative positions of other agents and landmarks to learn cooperation and receive rewards based on their proximity to the landmarks.

3.2. Experimental Results

The initial training iteration for the agents is set to 15,000 rounds, with each iteration consisting of 25 steps. Once the buffer accumulates 1,000 interaction rounds, the agents' Actor-Critic (AC) network begins training. Training concludes when the maximum iteration limit is reached, and the trained model is saved.

To validate the superiority of the quantum-based multi-agent reinforcement learning algorithm, a comprehensive comparison was conducted with five baseline algorithms in the paper, namely MADDPG [13], VDN [14], COMA [15], QMIX [16], and QTRAN [17]. The paper compared the

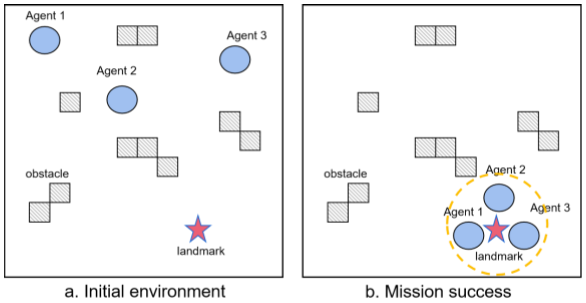

Figure 5. The CNO environment with three agents.

rewards obtained during the training process by the quantum-based multi-agent reinforcement learning method and the five baseline algorithms, as shown in Figure 6. After reaching convergence, the paper further compared the quantum-based method with the baselines in terms of average episode reward, distance to the target, average episode length, and average collision count with obstacles. The statistical results are presented in Table 1.

Table 1. Comparison of performance metrics of QMARL and different algorithms in the CNO environment.

Parameter index	**QMARL**	MADDPG	COMA	QTRAN	VDN	QMIX
Average round reward	286.558	266.177	267.107	267.682	269.592	269.309
Distance from landmarks	0.459	0.578	0.581	0.589	0.588	0.591
Average round collisions	3.877	6.336	5.5742	5.1594	4.676	4.278
Average path length	24.451	31.723	31.605	32.774	32.632	31.581

These results demonstrate that QMARL outperforms the baselines in terms of average episode reward, distance to the target, collision count, and episode path length. It showcases the ability of the quantum-based approach to enable agents to navigate to the target more efficiently in the cooperative navigation with obstacles scenario.

To validate the effectiveness of the proposed quantum module for solving multi-agent reinforcement learning problems, this paper conducts ablation experiments on the QMARL method from the following aspects. Firstly, the quantum module, as the differing factor between MADDPG and QMARL, provides a baseline for ablation (with or without the quantum module) to verify the effectiveness of the quantum module. Secondly, to validate the effectiveness of the quantum experience replay module in the quantum-based multi-agent reinforcement learning framework, this paper compares the methods with and without the quantum experience replay module. The comparison results are shown in Figure 7. In the CNO environment, the method with the quantum experience replay module converges faster than the method without the quantum experience replay module and MADDPG.

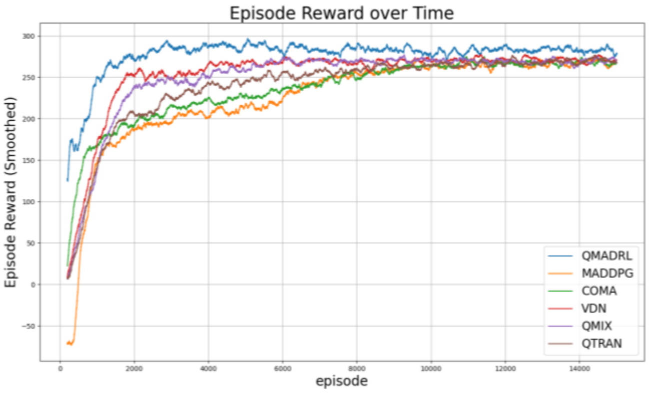

Figure 6. Reward variation of QMARL and multiple baseline algorithms.

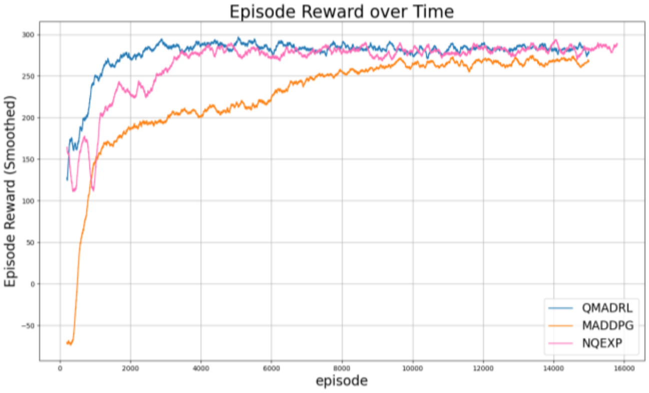

Figure 7. Reward variation before and after the introduction of the quantum experience replay module.

These findings provide strong evidence for the effectiveness of the quantum-based multi-agent reinforcement learning method and its components, including the quantum module and the quantum experience replay module, in addressing multi-agent reinforcement learning problems.

4. CONCLUSION

In this study, we propose a quantum-based multi-agent reinforcement learning algorithm and validate it in a cooperative navigation scenario. The Grover algorithm is employed to make action decisions for the agents, enabling them to learn optimal strategies in the current environment. Additionally, the quantum experience replay method effectively facilitates the training of a centralized critic in multi-agent systems. When compared to various multi-agent reinforcement learning algorithms in the same scenario, the intelligent agent system trained using the quantum-based multi-agent reinforcement learning algorithm achieves higher average round rewards and requires fewer rounds.

12. REFERENCES

[1] Gronauer S, Diepold K. Multi-agent deep reinforcement learning: a survey[J]. Artificial Intelligence Review, 2022: 1-49.

[2] Hernandez-Leal P, Kaisers M, Baarslag T, et al. A survey of learning in multiagent environments: Dealing with non-stationarity[J]. arXiv preprint arXiv:1707.09183, 2017.

[3] Da Silva F L, Taylor M E, Costa A H R. Autonomously Reusing Knowledge in Multiagent Reinforcement Learning[C]//IJCAI. 2018: 5487-5493.

[4] Foerster J, Assael I A, De Freitas N, et al. Learning to communicate with deep multi-agent reinforcement learning[J]. Advances in neural information processing systems, 2016, 29.

[5] Sunehag P, Lever G, Gruslys A, et al. Value-decomposition networks for cooperative multi-agent learning[J]. arXiv preprint arXiv:1706.05296, 2017.

[6] Mnih V, Kavukcuoglu K, Silver D, et al. Human-level control through deep reinforcement learning[J]. nature, 2015, 518(7540): 529-533.

[7] Dong D, Chen C, Li H, et al. Quantum reinforcement learning[J]. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 2008, 38(5): 1207-1220.

[8] Dunjko V, Taylor J M, Briegel H J. Framework for learning agents in quantum environments[J]. arXiv preprint arXiv:1507.08482, 2015.

[9] Dunjko V, Taylor J M, Briegel H J. Quantum-enhanced machine learning[J]. Physical review letters, 2016, 117(13): 130501.

[10] Dunjko V, Taylor J M, Briegel H J. Advances in quantum reinforcement learning[C]//2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC). IEEE, 2017: 282-287.

[11] Cerezo M, Arrasmith A, Babbush R, et al. Variational quantum algorithms[J]. Nature Reviews Physics, 2021, 3(9): 625-644.

[12] Schaul T, Quan J, Antonoglou I, et al. Prioritized experience replay[J]. arXiv preprint arXiv:1511.05952, 2015.

[13] Lowe R, Wu Y I, Tamar A, et al. Multi-agent actor-critic for mixed cooperative-competitive environments[J]. Advances in neural information processing systems, 2017, 30.

[14] Sunehag P, Lever G, Gruslys A, et al. Value-decomposition networks for cooperative multi-agent learning[J]. arXiv preprint arXiv:1706.05296, 2017.

[15] Foerster J, Farquhar G, Afouras T, et al. Counterfactual multi-agent policy gradients//Proceedings of the AAAI conference on artificial intelligence. New Orleans, USA, 2018: 32(1).

[16] Rashid T, Samvelyan M, De Witt C S, et al. Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning[J]. The Journal of Machine Learning Research, 2020, 21(1): 7234-7284.

[17] Son K, Kim D, Kang W J, et al. Qtran: Learning to factorize with transformation for cooperative multi-agent reinforcement learning[C]//International conference on machine learning. PMLR, 2019: 5887-5896.