Introduction to Quantum Centralized-Critic and Multiple-Actor Networks

Joongheon Kim
School of Electrical Engineering, Korea University, Seoul, Republic of Korea
E-mail: joongheon@korea.ac.kr

***Abstract*—This paper introduces quantum centralized-critic and multiple-actor network architectures and organization.**

I. Introduction

Qunatum Centralized-Critic and Multiple-Actor Network. We propose a novel quantum centralized critic and multiple actor network. Here, there is one *centralized critic* and multiple *actor* networks where the number of actor networks is commensurate with the number of agents. The agents make the sequential decision dispersively and train their multiple *actor* networks corresponding to the policy by evaluating the value of *centralized critic* network, which can be expressed as [1],

$$Q(o,a;\boldsymbol{\theta}) = \beta_a \langle O_a \rangle_{o,\boldsymbol{\theta}} = \beta_a \mathrm{Tr}(U^{a\dagger}(o;\boldsymbol{\theta}) M_a U^a(o;\boldsymbol{\theta})) \quad (1)$$

$$V(s;\boldsymbol{\phi}) = \beta_c \langle O \rangle_{s,\boldsymbol{\phi}} \ = \beta_c \mathrm{Tr}(U^{c\dagger}(s;\boldsymbol{\phi}) M_c U^c(s;\boldsymbol{\phi})) \quad (2)$$

where operators $\mathrm{Tr}(\cdot)$, $U(\cdot)$, and $(\cdot)^\dagger$ represent trace operator, the unitary operation for qubit rotation, and the entanglement of multiple qubits and complex conjugate, respectively. When the quantum state is measured, the output (known as observable) exists between -1 and 1, *i.e.*, $\forall \langle O \rangle \in [-1, 1]$, we utilize hyper-parameters (β_a, β_c) for *actors* and *critic* networks to be well-trained. Note that M_a and M_c are Hermitian matrices. With Eqs. (1)–(2), and the hyper-parameters, the *actor-critic* networks approximate the value function.

Quantum Actor. At every time step t, the m-th quantum *actor* chooses the action with the most significant probability among the currently possible actions based on its state and observation information, which is represented as, $\mathrm{a}_{\mathrm{m,real}} = \arg\max D_{\mathrm{a}} \pi_{\boldsymbol{\theta}_m}(\mathrm{a}_{\mathrm{ideal}} | s_{\mathrm{ideal}} + n_s, o_m) + n_a$, subject to $\pi_{\boldsymbol{\theta}_m}(\mathrm{a}_{\mathrm{ideal}} | s_{\mathrm{ideal}} + n_s, o_m) \triangleq \textit{softmax}(Q(o,a;\boldsymbol{\theta}_m))$ and $\textit{softmax}(\mathbf{x}) \triangleq \left[\frac{e^{x_1}}{\sum_{i=1}^N e^{x_i}}, \cdots, \frac{e^{x_N}}{\sum_{i=1}^N e^{x_i}}\right]$, where the $\textit{softmax}(\cdot)$ is an activation function to normalize the inputs. By using it, we extract all actions' probabilities of the *actor* with the observable $\langle O_a \rangle_{o,\boldsymbol{\theta}}$ in (1).

Quantum Centralized Critic. The CTDE has a *centralized critic* responsible for valuing the current state with a state-value function as follows, $V_{\boldsymbol{\phi}}(s) = \langle O_c \rangle_{s,\boldsymbol{\phi}} \simeq \mathbb{E}_{s_{\mathrm{real}} \sim E,\, \mathrm{a}_{\mathrm{real}} \sim \pi_{\boldsymbol{\theta}}} \left[\sum_{t'=t}^{T} \gamma^{t'-t} \cdot r(s_{\mathrm{real}}, \mathbf{a}_{\mathrm{real}}, s'_{\mathrm{real}}) \right]$, where s_t is the measured state at the current state at t. We also use the *critic* network's observable to evaluate the current state's value.

1) Training and Inference: MARL agents try to maximize the expected return. We utilize the congruent state-value function V_ϕ of the *centralized critic* network to derive the gradients from maximizing the common goal. With the parameters of *actor* and *critic* networks, which correspond to $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$, we configure a multi-agent policy gradient (MAPG) based on the temporal difference *actor-critic* model by *Bellman optimality equation*, as follows,

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \mathbb{E}_{s_{\mathrm{real}}, \mathrm{o} \sim E} \left[\sum_{t=1}^{T} \sum_{m=1}^{M} \delta_{\boldsymbol{\phi}}^t \cdot \nabla_{\boldsymbol{\theta}_m} \log \pi_{\boldsymbol{\theta}_m}(\mathrm{a}_m^t | s^t, o_m^t) \right], \quad (3)$$

and

$$\nabla_{\boldsymbol{\phi}} \mathcal{L}(\boldsymbol{\phi}) = \sum_{t=1}^{T} \nabla_{\boldsymbol{\phi}} \left\| \delta_{\boldsymbol{\phi}}^t \right\|^2, \quad (4)$$

subject to

$$\delta_{\boldsymbol{\phi}}^t = r\left(s^t, \mathbf{a}^t, s^{t+1}\right) + \gamma V_{\boldsymbol{\phi}}(s^{t+1}) - V_{\boldsymbol{\phi}}(s^t). \quad (5)$$

Among the three equations presented above, (3) is the objective function for *actor* networks, and the neural network parameters used in the equation are updated to be maximized by gradient ascent as, $\boldsymbol{\theta}_m^{t+1} \approx \boldsymbol{\theta}_m^t + \alpha_{\mathrm{actor}} \times [\, \delta_{\boldsymbol{\phi}}^t \cdot \nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}_m}(\mathrm{a}_m^t \,|\, s^t, \mathrm{o}_m^t)\,]$, where α_{actor} stands for a learning rate of *actor* networks. (4) corresponds to the *centralized critic* network's loss function which should be minimized by gradient descent as follows, $\boldsymbol{\phi}^{t+1} \approx \boldsymbol{\phi}^t + \alpha_{\mathrm{critic}} \times [\, \delta_{\boldsymbol{\phi}}^t \cdot \nabla_{\boldsymbol{\phi}} V_{\boldsymbol{\phi}}(s^t)\,]$, where α_{critic} is *centralized critic* network's learning rate. We describe how to obtain loss gradients with quantum and classical computing. Hereafter, we denote an *actor*-network and *critic* network as $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ for mathematical amenability. Their loss values are calculated with the temporal difference error of *centralized critic* $\delta_{\boldsymbol{\phi}}$ in (5), where the derivative of *actor*/*critic*'s i-th parameters is expressed as follows,

$$\frac{\partial J(\boldsymbol{\theta})}{\partial \theta_i} = \frac{\partial J(\boldsymbol{\theta})}{\partial \pi_{\boldsymbol{\theta}}} \cdot \frac{\partial \pi_{\boldsymbol{\theta}}}{\partial \langle O \rangle_{o,\boldsymbol{\theta}}} \cdot \frac{\partial \langle O \rangle_{o,\boldsymbol{\theta}}}{\partial \theta_i}, \quad (6)$$

$$\frac{\partial \mathcal{L}(\boldsymbol{\phi})}{\partial \phi_i} = \frac{\partial \mathcal{L}(\boldsymbol{\phi})}{\partial V_{\boldsymbol{\phi}}} \cdot \frac{\partial V_{\boldsymbol{\phi}}}{\partial \langle O \rangle_{s,\boldsymbol{\phi}}} \cdot \frac{\partial \langle O \rangle_{s,\boldsymbol{\phi}}}{\partial \phi_i}, \quad (7)$$

where the first and second derivatives of RHS can be calculated by classical computing. However, the latter derivative cannot be calculated because the quantum state is unknown before its measurement, i.e., parameter-shift rule [1] is used.

References

[1] C. Park, W. J. Yun, J. P. Kim, T. K. Rodrigues, S. Park, S. Jung, and J. Kim, "Quantum multi-agent actor-critic networks for cooperative mobile access in multi-uav systems," *IEEE Internet of Things Journal*, pp. 1–1, 2023 (Early Access).