Quantum Machine Learning: Bridging Quantum Computing and Artificial Intelligence

QC+AI Workshop @ AAAI 2025

Samuel Yen-Chi Chen
Lead Research Scientist@Wells Fargo Bank
ycchen1989@ieee.org

3 Mar 2025

- Fundamentals of Quantum Computing
- Hybrid Quantum-Classical Paradigm
- Variational Quantum Circuits (a.k.a Parameterized Quantum Circuits)
- Applications
- Machine Learning for Quantum Machine Learning Model Design
- Challenges in Quantum Machine Learning
- Conclusion and Outlook

- Applications
    - Quantum Classification
    - Privacy-Preserving Quantum Machine Learning (Federated Learning, Differential Privacy)
    - Quantum Recurrent Neural Network
    - Quantum Reinforcement Learning
    - Quantum Natural Language Processing
    - Quantum Neural Networks for Model Compression

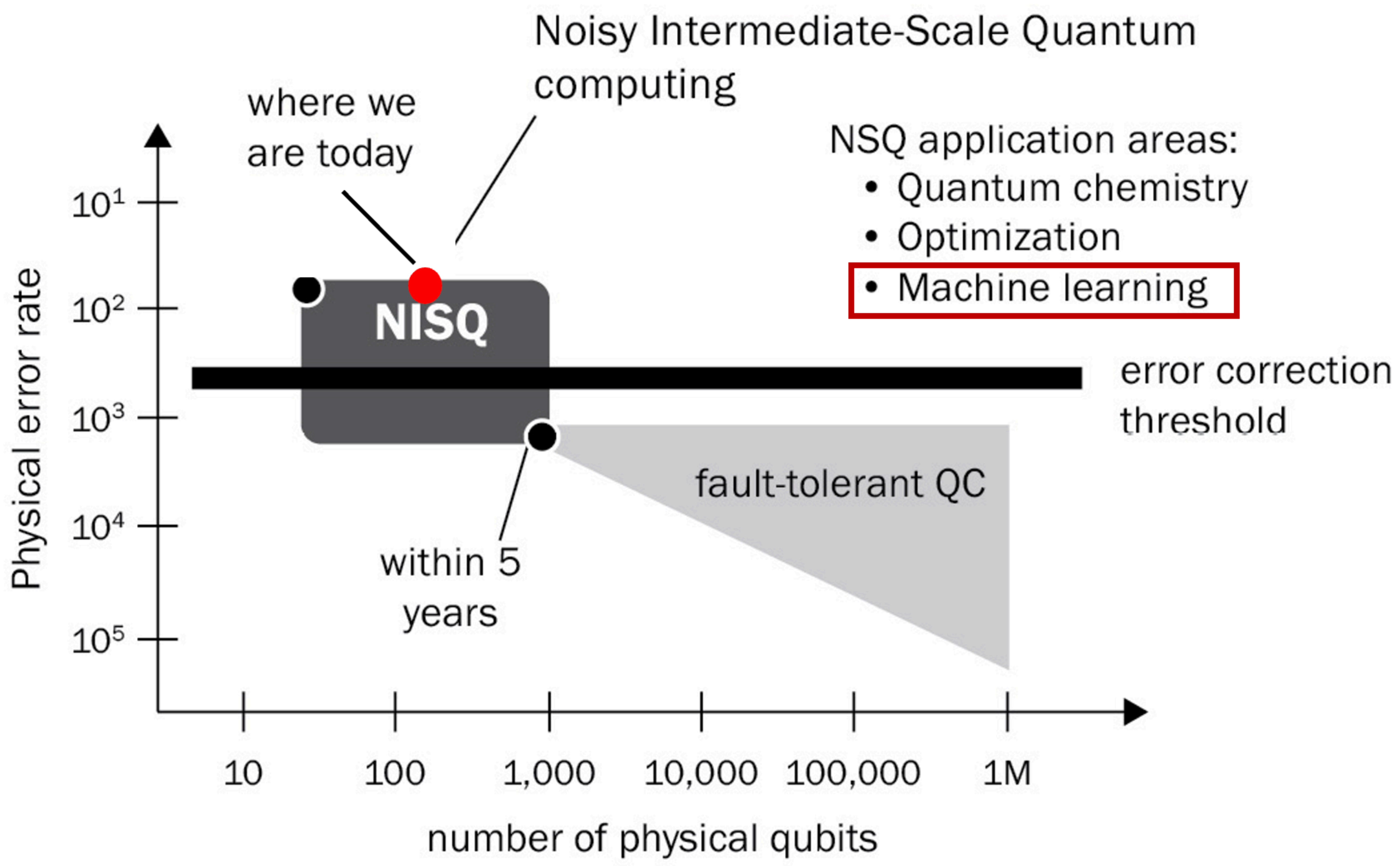
- **Fundamentals of Quantum Computing**
- Hybrid Quantum-Classical Paradigm
- Variational Quantum Circuits (a.k.a Parameterized Quantum Circuits)
- Applications
- Machine Learning for Quantum Machine Learning Model Design
- Challenges in Quantum Machine Learning
- Conclusion and Outlook

Quantum Computing

- Classical computers: Classical bits 0 vs 1
- Quantum computers: Quantum bits (qubit)
  $|\Psi\rangle = \alpha|0\rangle + \beta|1\rangle$ where α and β are complex numbers
- Quantum **entanglements**: A unique property of quantum physics
  —> No analog in the classical computer
- Famous algorithms:
  - **Shor's algorithm**: Can be used to break the state-of-the-art public key cryptography systems such as RSA
  - **Grover's algorithm**: Quadratic speedup in unstructured search
- Designing a quantum algorithm is non-trivial task
- Even harder in the noisy quantum machines

Schrödinger's cat from AI's imagination!

Quantum Computing



Quantum computing in the NISQ era [1]

Quantum computers from ChatGPT's imagination!

[1] SAXENA, Anshul, et al. Financial Modeling Using Quantum Computing: Design and manage quantum machine learning solutions for financial analysis and decision making. Packt Publishing Ltd, 2023.

Quantum States

Single Qubit State

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$|1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Two Qubit State

$$|0\rangle \otimes |0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

N Qubit State

$$\underbrace{|0\rangle \otimes |0\rangle \otimes \cdots \otimes |0\rangle}_{N} = \underbrace{\begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \cdots \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix}}_{N}$$

Quantum Operations

$\boxed{X}$ $$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$\boxed{Y}$ $$\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$$

$\boxed{Z}$ $$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

$\boxed{H}$ $$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

Example:

$$|0\rangle \;\text{———}\; \boxed{X}$$

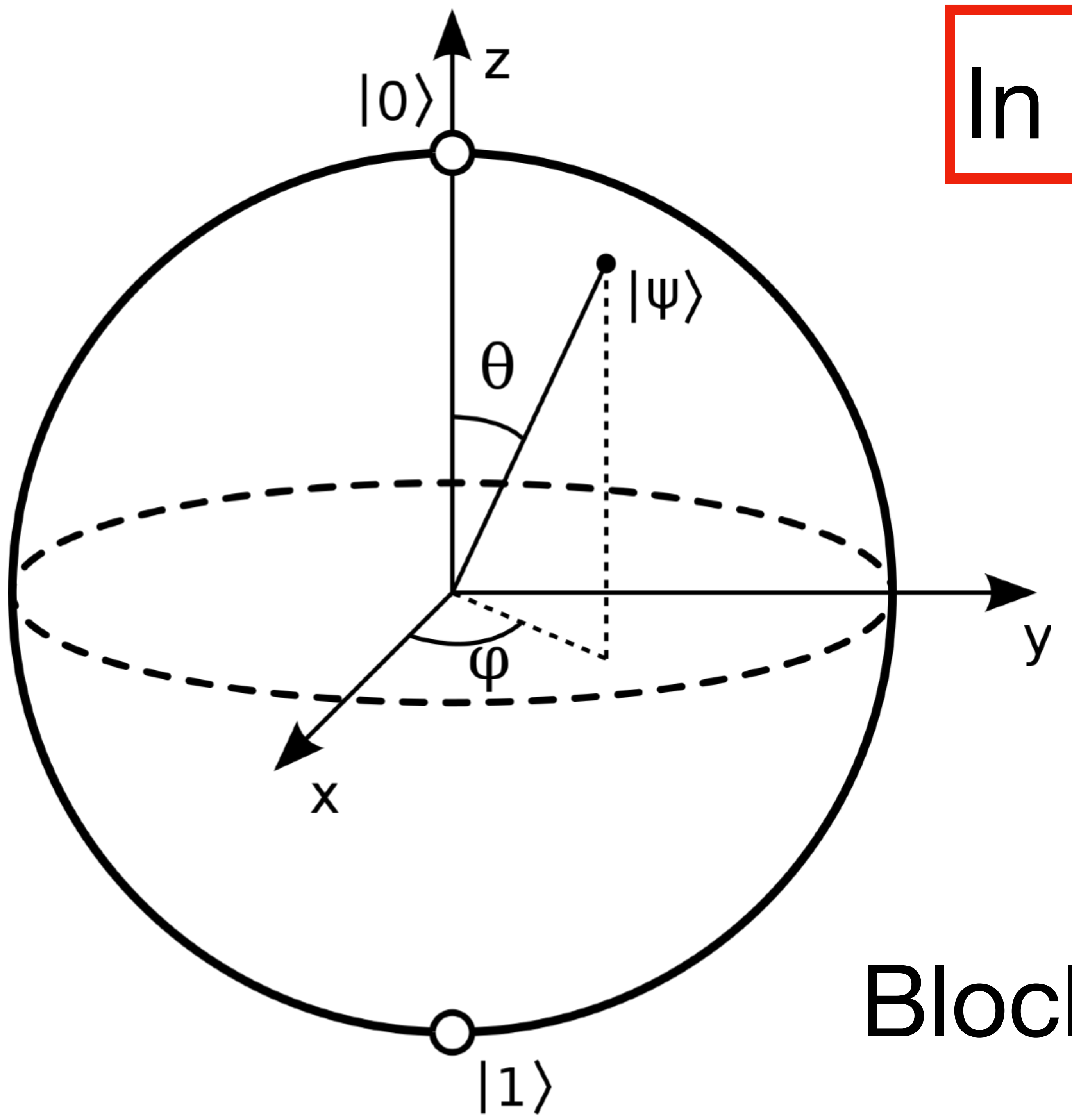
$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} = |1\rangle$$

Quantum Operations

$$\boxed{R(\phi, \theta, \omega)}$$

$$\begin{bmatrix} e^{-i(\phi+\omega)/2}\cos(\theta/2) & e^{-i(\phi-\omega)/2}\sin(\theta/2) \\ e^{-i(\phi-\omega)/2}\sin(\theta/2) & e^{i(\phi+\omega)/2}\cos(\theta/2) \end{bmatrix}$$

In QML, the angles ϕ, θ, ω are learnable.



Bloch sphere

Quantum Operations

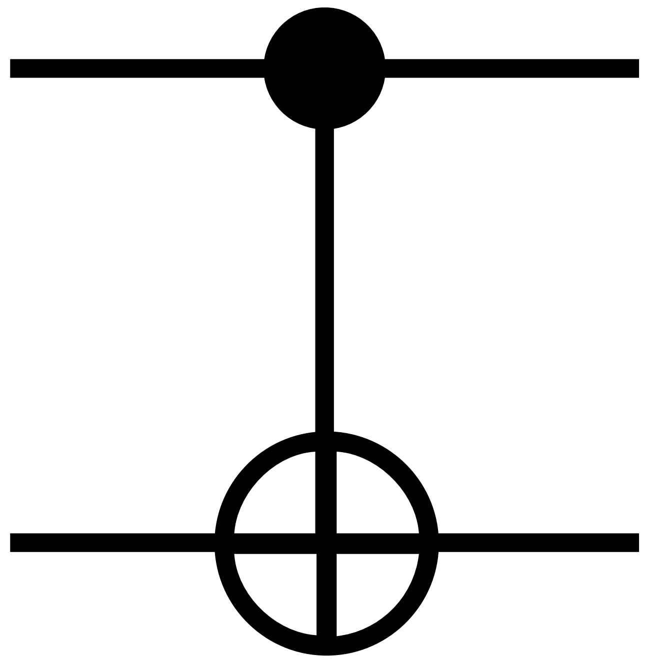
$$\boxed{R_x(\phi)} = e^{-i\phi\sigma_x/2} = \begin{bmatrix} \cos(\phi/2) & -i\sin(\phi/2) \\ -i\sin(\phi/2) & \cos(\phi/2) \end{bmatrix}$$

$$\boxed{R_y(\phi)} = e^{-i\phi\sigma_y/2} = \begin{bmatrix} \cos(\phi/2) & -\sin(\phi/2) \\ \sin(\phi/2) & \cos(\phi/2) \end{bmatrix}$$

$$\boxed{R_z(\phi)} = e^{-i\phi\sigma_z/2} = \begin{bmatrix} e^{-i\phi/2} & 0 \\ 0 & e^{i\phi/2} \end{bmatrix}$$

Quantum Operations

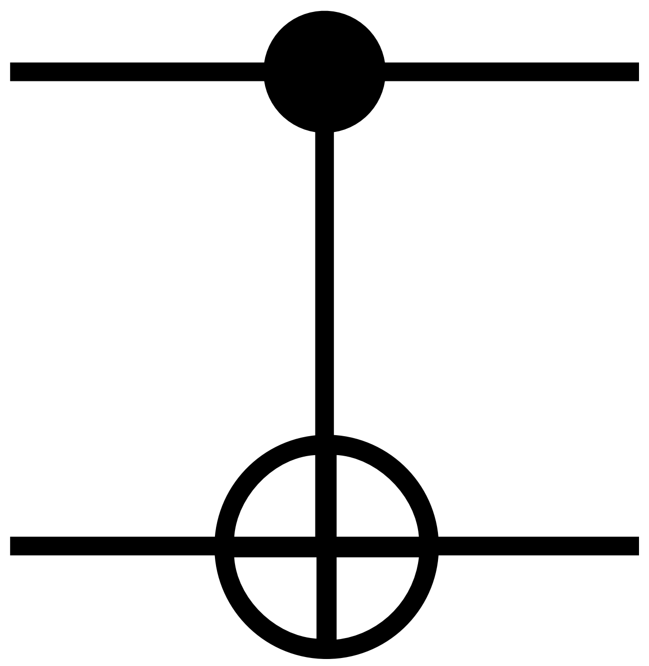
CNOT



$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

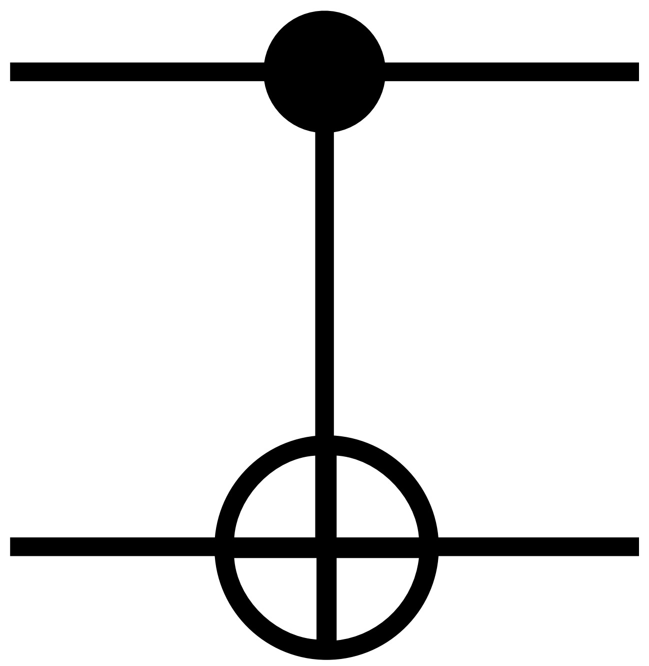
Result

$|0\rangle$ $|0\rangle$



$|0\rangle$ $|0\rangle$

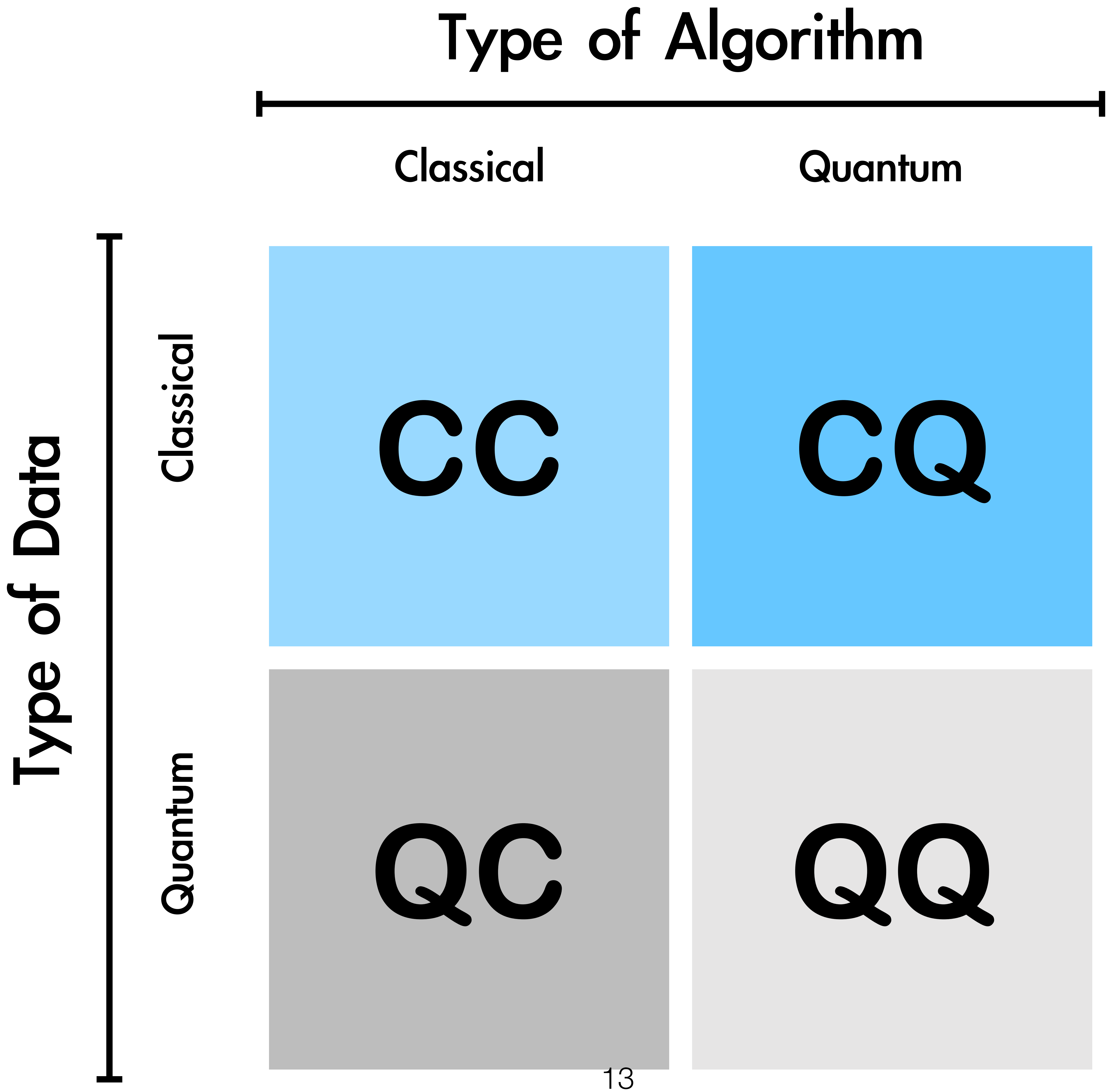
$|1\rangle$ $|0\rangle$



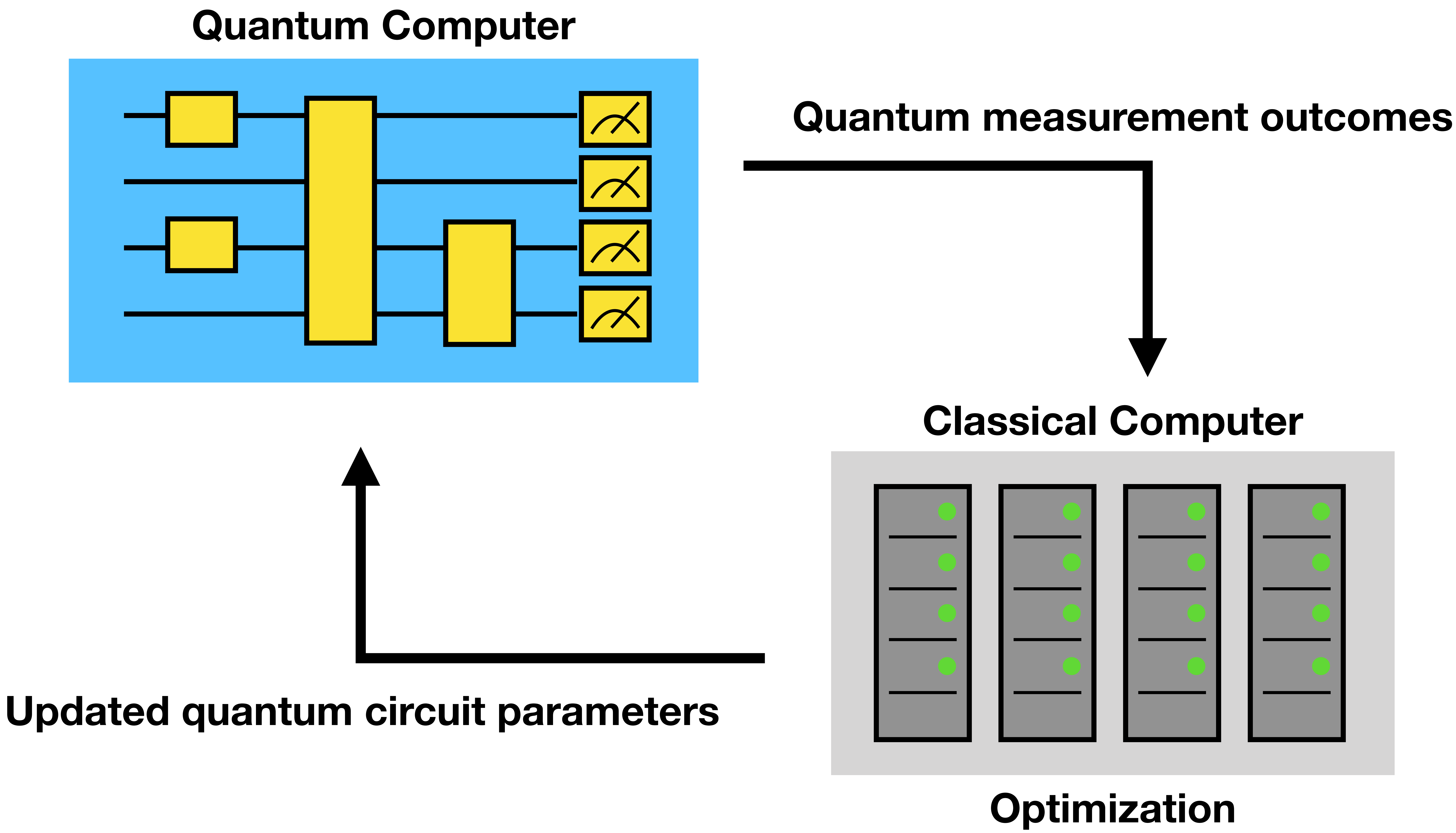
$|1\rangle$ $|1\rangle$

- Fundamentals of Quantum Computing
- **Hybrid Quantum-Classical Paradigm**
- Variational Quantum Circuits (a.k.a Parameterized Quantum Circuits)
- Applications
- Machine Learning for Quantum Machine Learning Model Design
- Challenges in Quantum Machine Learning
- Conclusion and Outlook

Quantum Machine Learning

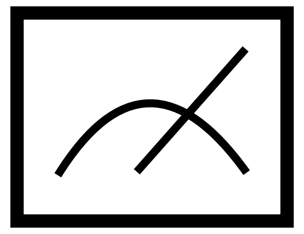


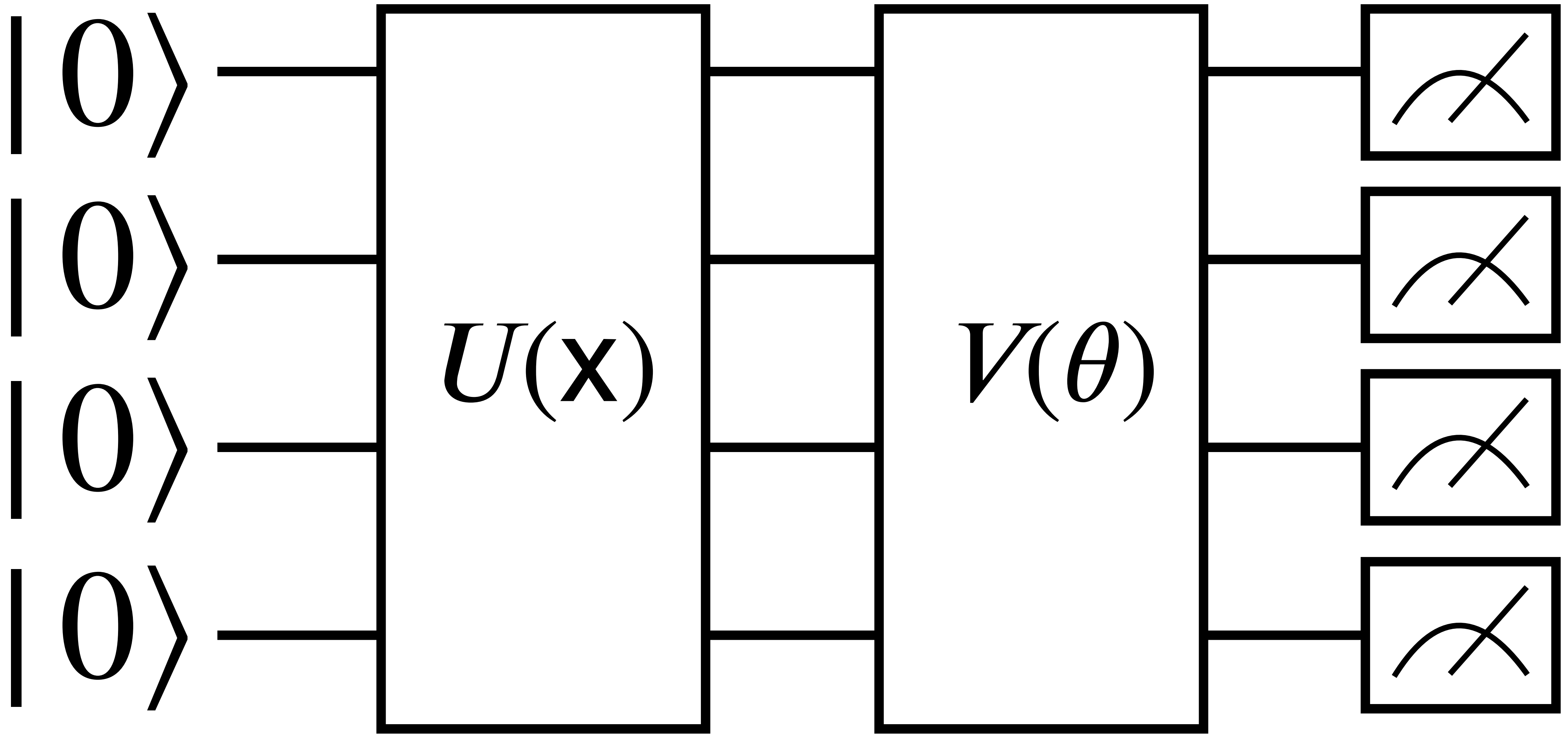
Hybrid Quantum-Classical Paradigm



- Fundamentals of Quantum Computing
- Hybrid Quantum-Classical Paradigm
- **Variational Quantum Circuits (a.k.a Parameterized Quantum Circuits)**
- Applications
- Machine Learning for Quantum Machine Learning Model Design
- Challenges in Quantum Machine Learning
- Conclusion and Outlook

Variational Quantum Circuits

- Also known as **parameterized quantum circuits (PQC)**.
- Quantum circuits with **tunable parameters**.
- Subject to iterative optimization procedures.
- $U(\mathbf{x})$: encoding circuit.
- $V(\theta)$: variational circuit.
-  : measurement.



Variational Quantum Circuits

- Choosing some observables (e.g. Pauli-X, Y or Z)

- Expectation value from a particular qubit: $\left\langle \hat{B}_k \right\rangle = \left\langle 0 \left| U^\dagger(\vec{x}) V^\dagger(\vec{\theta}) \hat{B}_k V(\vec{\theta}) U(\vec{x}) \right| 0 \right\rangle$

- Quantum function (output from the VQC): $\overrightarrow{f(\vec{x};\vec{\theta})} = \left(\left\langle \hat{B}_1 \right\rangle, \cdots, \left\langle \hat{B}_n \right\rangle \right)$

- Gradient calculation by **parameter-shift** rule.

Quantum Encoding and State Preparation

A general N qubit quantum state can be represented as:

$$|\psi\rangle = \sum_{(q_1,q_2,\cdots,q_N)\in\{0,1\}} c_{q_1,q_2,\cdots,q_N} \left|q_1\right\rangle \otimes \left|q_2\right\rangle \otimes \cdots \otimes \left|q_N\right\rangle$$

where $c_{q_1,\cdots,q_N} \in \mathbb{C}$ is the complex amplitude for each basis state and each $q_i \in \{0,1\}$

The total probability is equal to 1: $$\sum_{(q_1,\cdots,q_N)\in\{0,1\}} \| c_{q_1,\cdots,q_N} \|^2 = 1$$

Quantum Encoding and State Preparation

Amplitude Encoding

Encode a vector $(\alpha_0, \cdots, \alpha_{2^n-1})$ into a n-qubit quantum state:

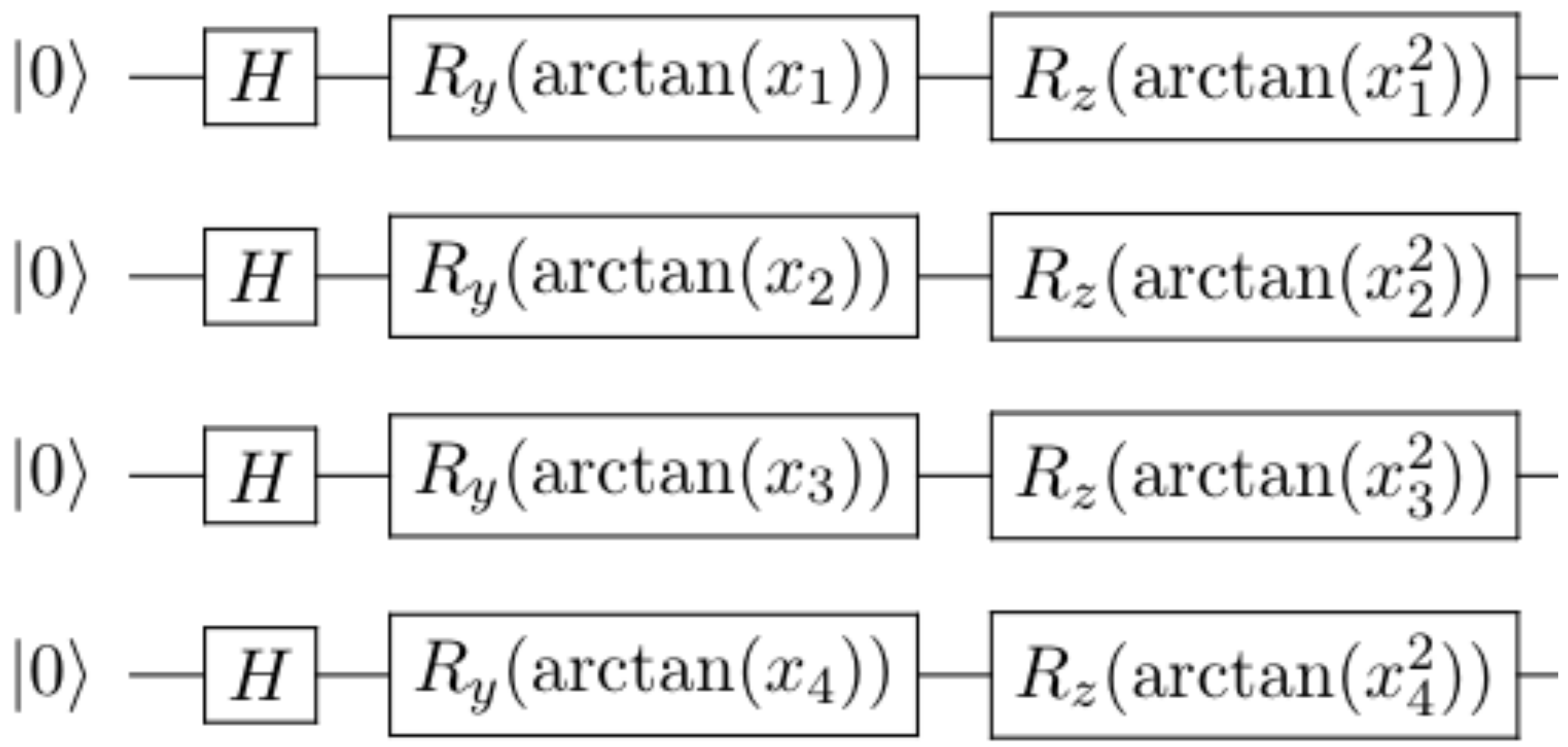
$$|\Psi\rangle = \alpha_0|00\cdots0\rangle + \cdots + \alpha_{2^n-1}|11\cdots1\rangle$$

where α_i are real numbers and $(\alpha_0, \cdots, \alpha_{2^n-1})$ is normalized

N-dimensional vector will require only $\log_2(N)$ qubits to encode

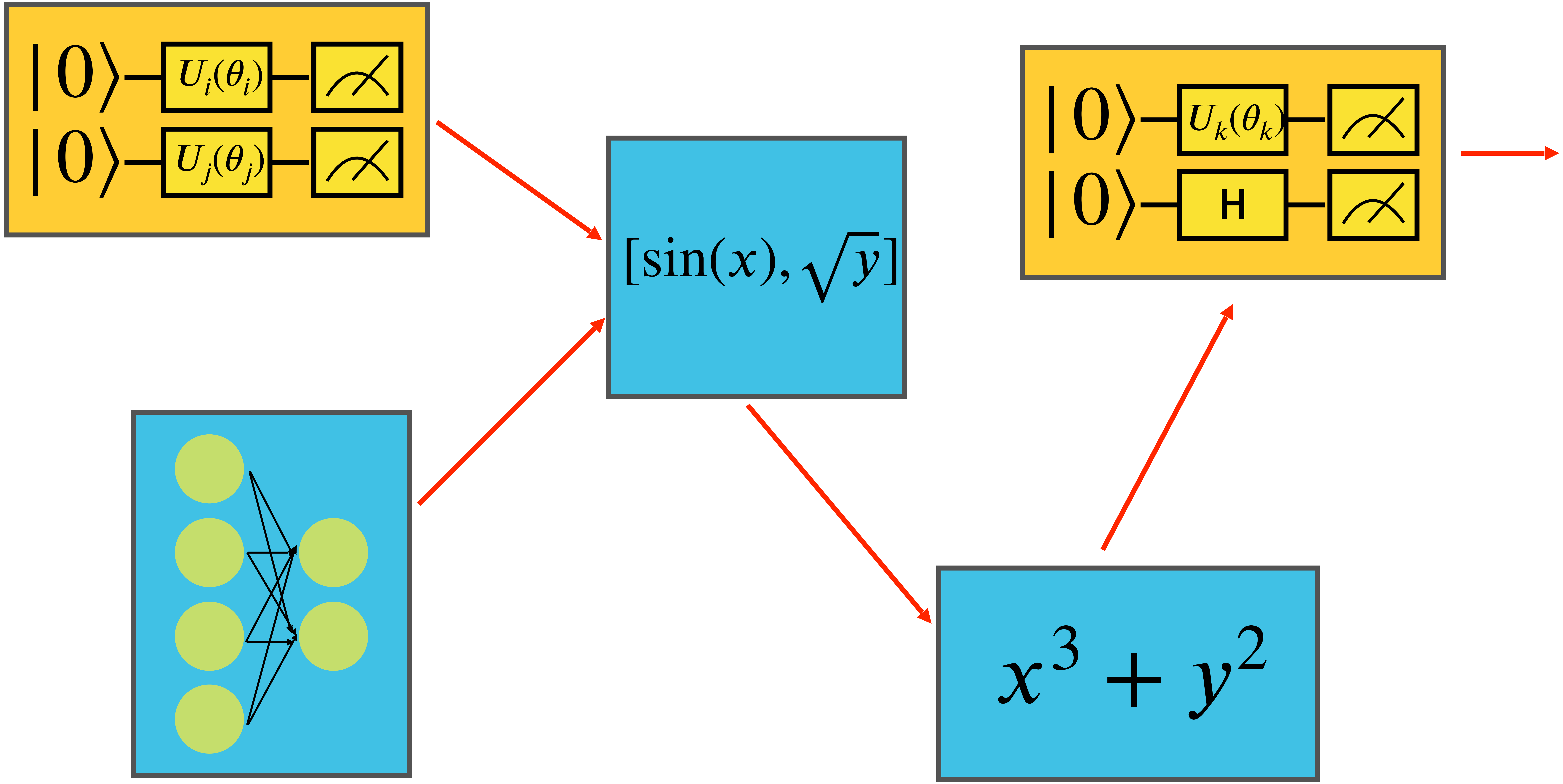
Variational Encoding (Angle Encoding)

Input numbers $x_1 \cdots x_n$ are used as quantum rotation angles



Simpler implementation than amplitude encoding

Interfacing with Classical ML



Interfacing with Classical ML

1. Mixing classical and quantum computing components.

2. These classical and quantum nodes are arranged in a **directed acyclic graph (DAG)**.

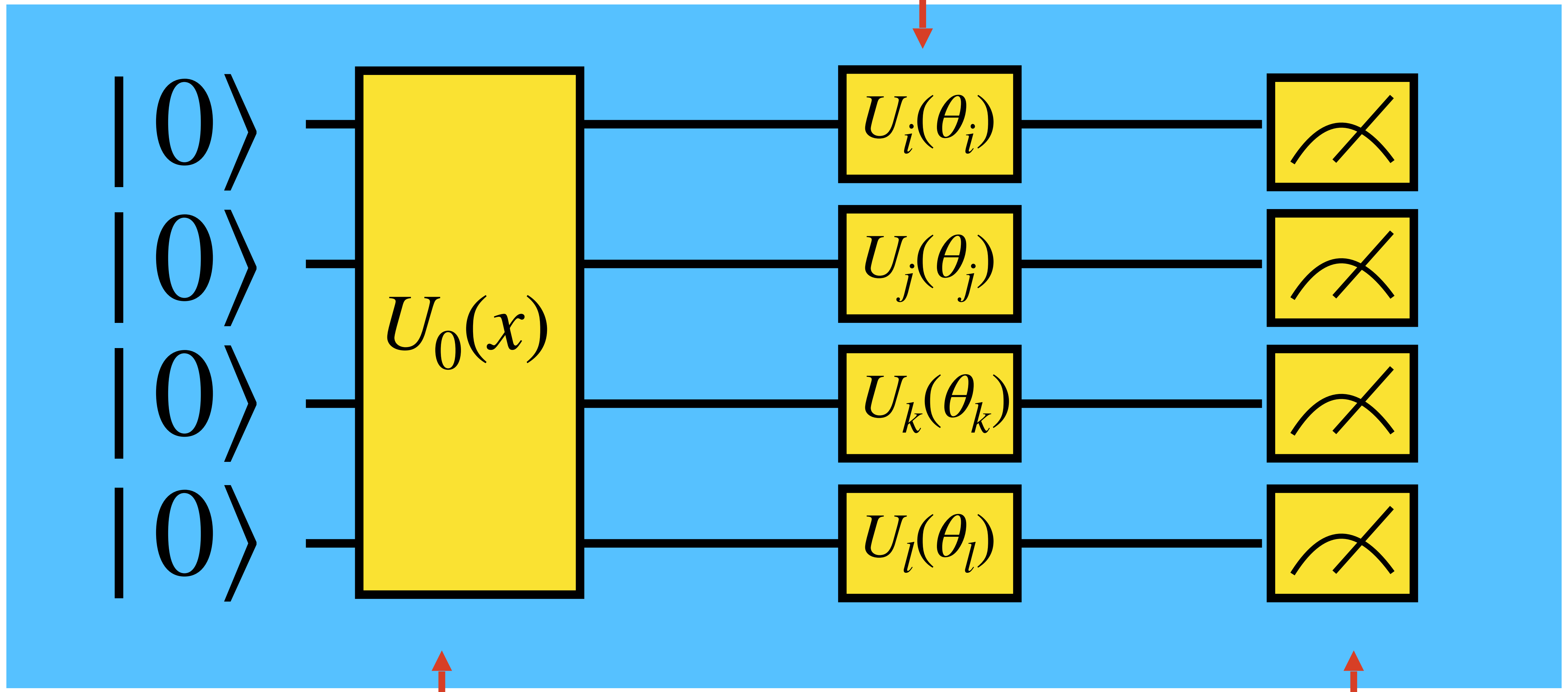
3. The hybrid architecture is similar to the one in deep learning models.

4. The whole model can be trained with backpropagation method or other gradient-free methods, such as evolutionary optimization.

5. The next question is “**How to calculate the gradient of a quantum node?**”

Quantum Gradients

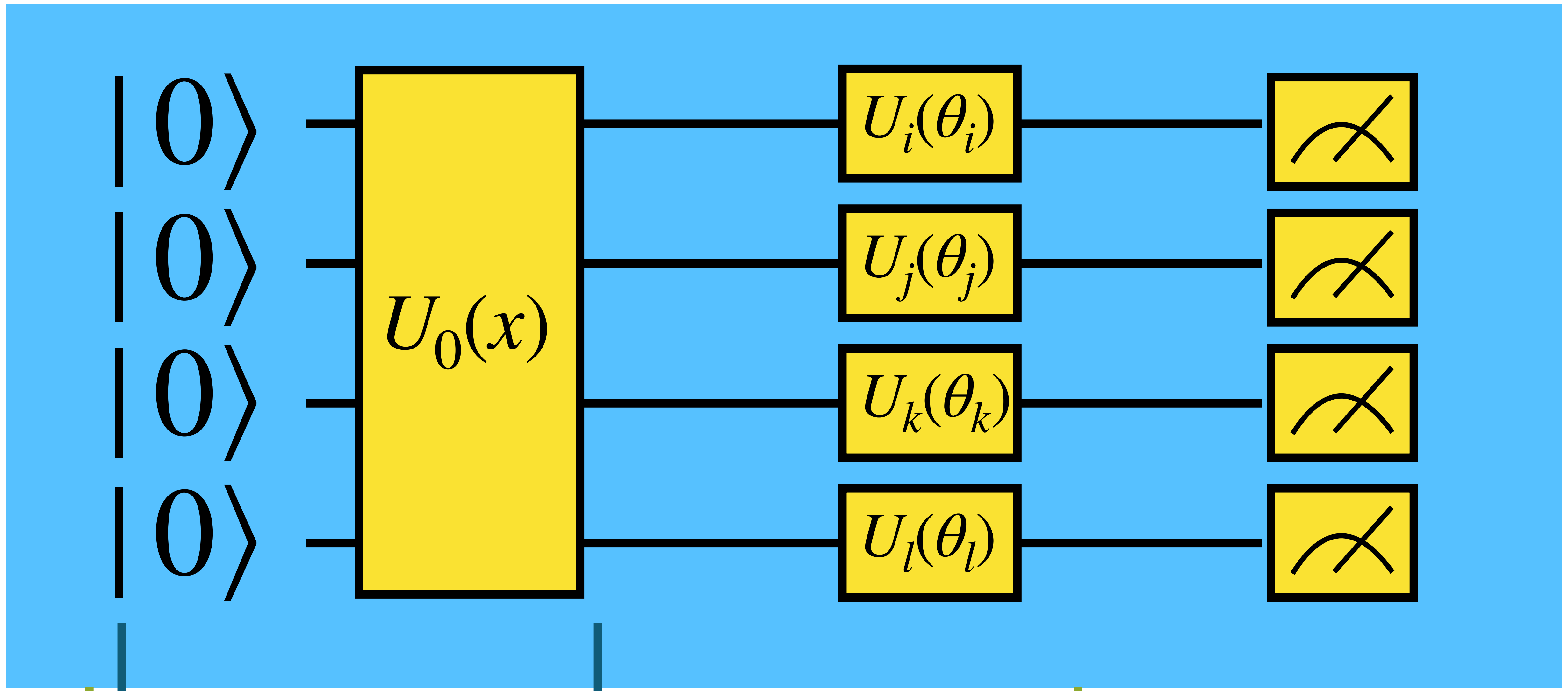
Learnable quantum circuit parameters



Quantum encoding / state preparation circuit

Quantum measurements

Quantum Gradients



$$U_0(x)|0\rangle$$

$$U_i\left(\theta_i\right) U_0(x)|0\rangle$$

$$\left\langle 0 \middle| U_0^\dagger(x) U_i^\dagger\left(\theta_i\right) \hat{B} U_i\left(\theta_i\right) U_0(x) \middle| 0 \right\rangle$$

Quantum Gradients

$$f\left(x;\theta_i\right) = \left\langle 0 \left| U_0^\dagger(x) U_i^\dagger\left(\theta_i\right) \hat{B} U_i\left(\theta_i\right) U_0(x) \right| 0 \right\rangle = \left\langle x \left| U_i^\dagger\left(\theta_i\right) \hat{B} U_i\left(\theta_i\right) \right| x \right\rangle$$

x: input value

$U_0(x)$: encoding circuit

i: circuit parameter index

$U_i(x_i)$: single-qubit rotation generated by the Pauli operators

Mitarai, K., Negoro, M., Kitagawa, M., & Fujii, K. (2018). Quantum circuit learning. *Physical Review A*, *98*(3), 032309.
Schuld, M., Bergholm, V., Gogolin, C., Izaac, J., & Killoran, N. (2019). Evaluating analytic gradients on quantum hardware. *Physical Review A*, *99*(3), 032331.

Quantum Gradients

The gradient of f with respect to the parameter θ_i is:

$$\nabla_{\theta_i} f\left(x;\theta_i\right) = \frac{1}{2}\left[f\left(x;\theta_i + \frac{\pi}{2}\right) - f\left(x;\theta_i - \frac{\pi}{2}\right)\right]$$

This value can be calculated via running two quantum circuits with shifted parameters, the so-called *parameter-shift* rule.

Mitarai, K., Negoro, M., Kitagawa, M., & Fujii, K. (2018). Quantum circuit learning. *Physical Review A*, *98*(3), 032309.
Schuld, M., Bergholm, V., Gogolin, C., Izaac, J., & Killoran, N. (2019). Evaluating analytic gradients on quantum hardware. *Physical Review A*, *99*(3), 032331.

Automatic Differentiation

1. **Chain rule!**

2. Directed acyclic graphs (DAG)

3. Using known gradient calculation

4. Workhorse of modern deep learning.

5. Quantum node is a **black-box**

6. Backpropagate through the **computational graph,** not the quantum node itself!

Open Source

- Quantum Computing/QML platforms: Qiskit, PennyLane, TorchQuantum, TensorFlow Quantum…
- Simulation backends: Qulacs, cuQuantum...

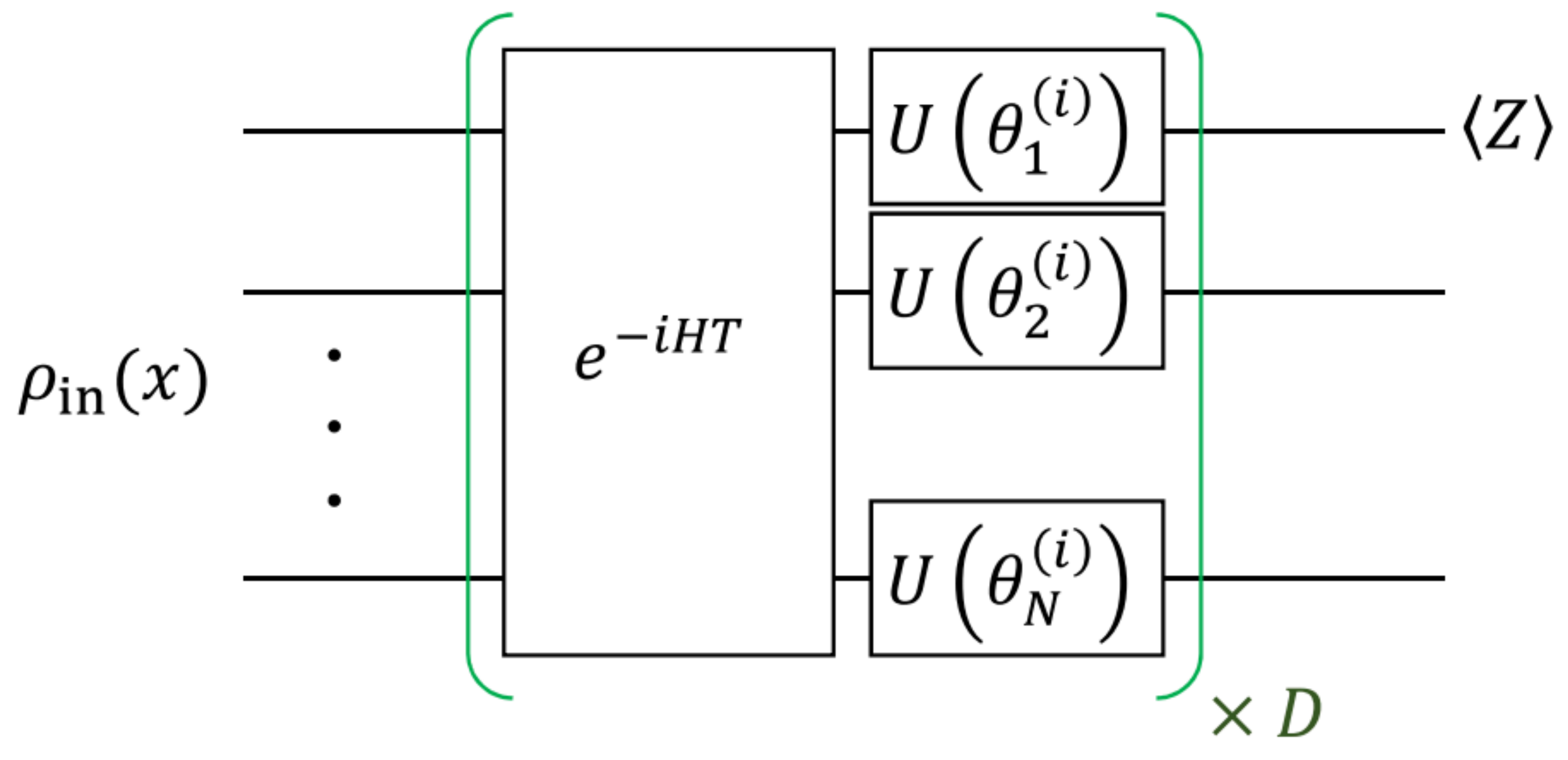
- Fundamentals of Quantum Computing
- Hybrid Quantum-Classical Paradigm
- Variational Quantum Circuits (a.k.a Parameterized Quantum Circuits)
- **Applications**
- Machine Learning for Quantum Machine Learning Model Design
- Challenges in Quantum Machine Learning
- Conclusion and Outlook

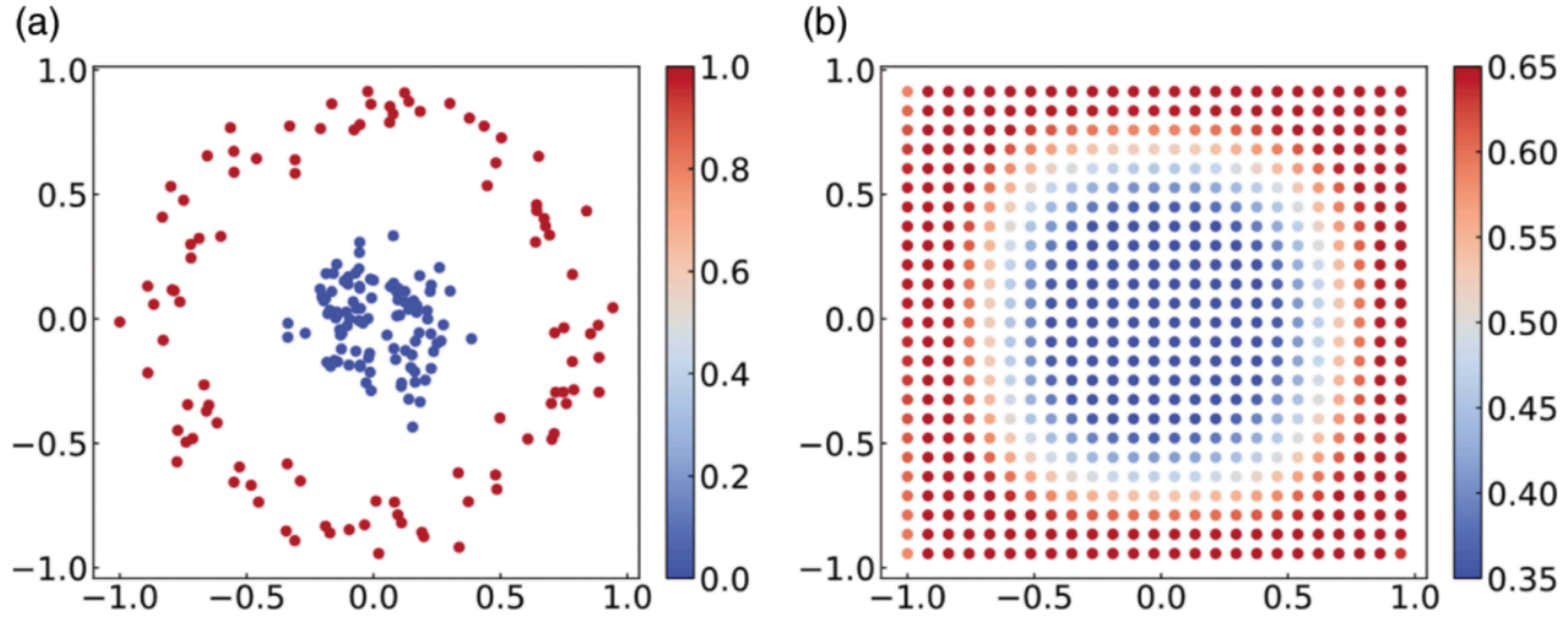
- **Applications**
    - **Quantum Classification**
    - **Privacy-Preserving Quantum Machine Learning (Federated Learning, Differential Privacy)**
    - **Quantum Recurrent Neural Network**
    - **Quantum Reinforcement Learning**
    - **Quantum Natural Language Processing**
    - **Quantum Neural Networks for Model Compression**

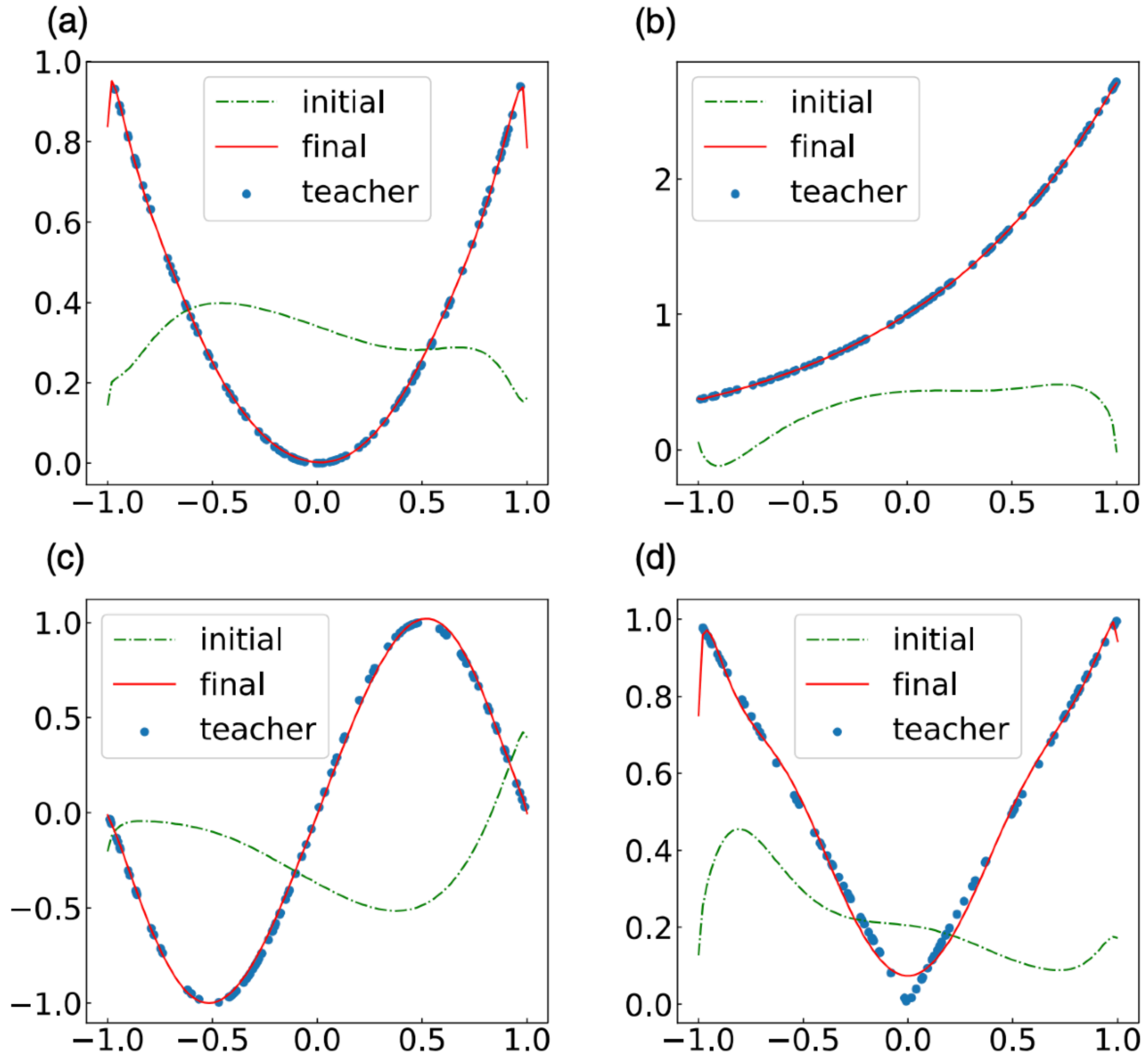
- **Applications**
  - **Quantum Classification**
  - **Privacy-Preserving Quantum Machine Learning (Federated Learning, Differential Privacy)**
  - **Quantum Recurrent Neural Network**
  - **Quantum Reinforcement Learning**
  - **Quantum Natural Language Processing**
  - **Quantum Neural Networks for Model Compression**

Quantum Circuit Learning

- First VQC-based QML model.
- Can perform simple “classification” ans “function approximation”

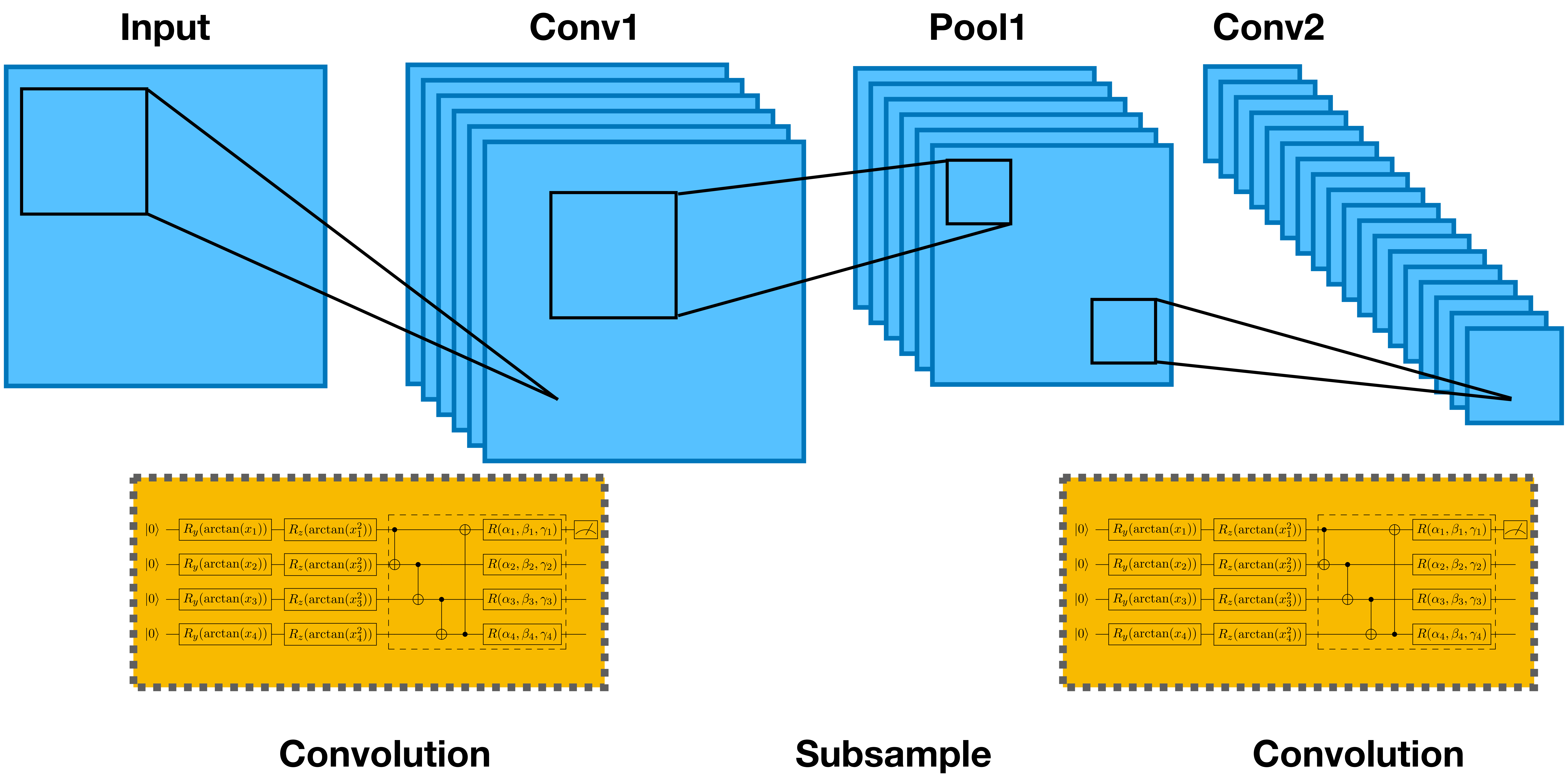






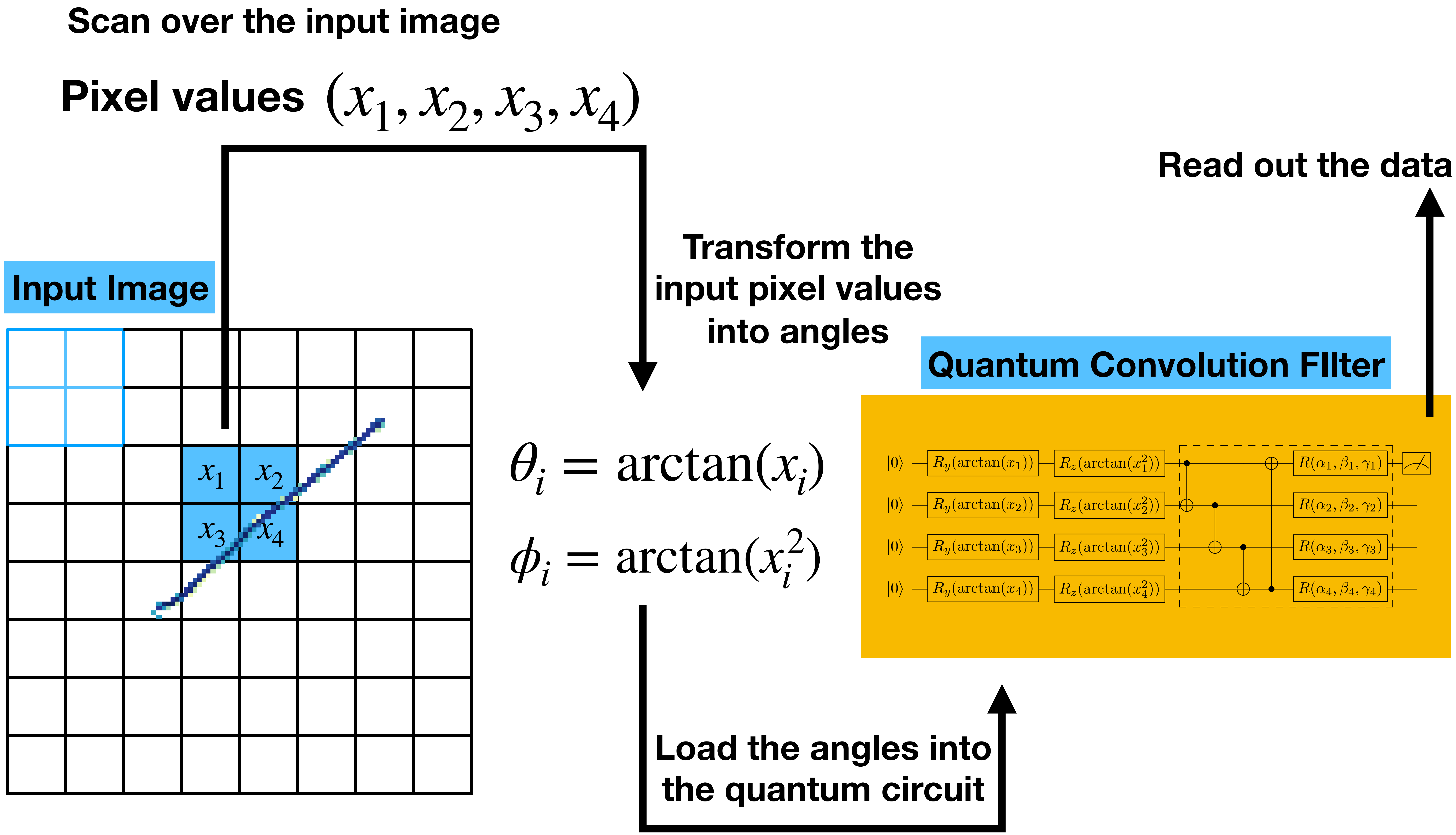
Mitarai, Kosuke, Makoto Negoro, Masahiro Kitagawa, and Keisuke Fujii. "Quantum circuit learning." *Physical Review A* 98, no. 3 (2018): 032309.

Quantum CNN



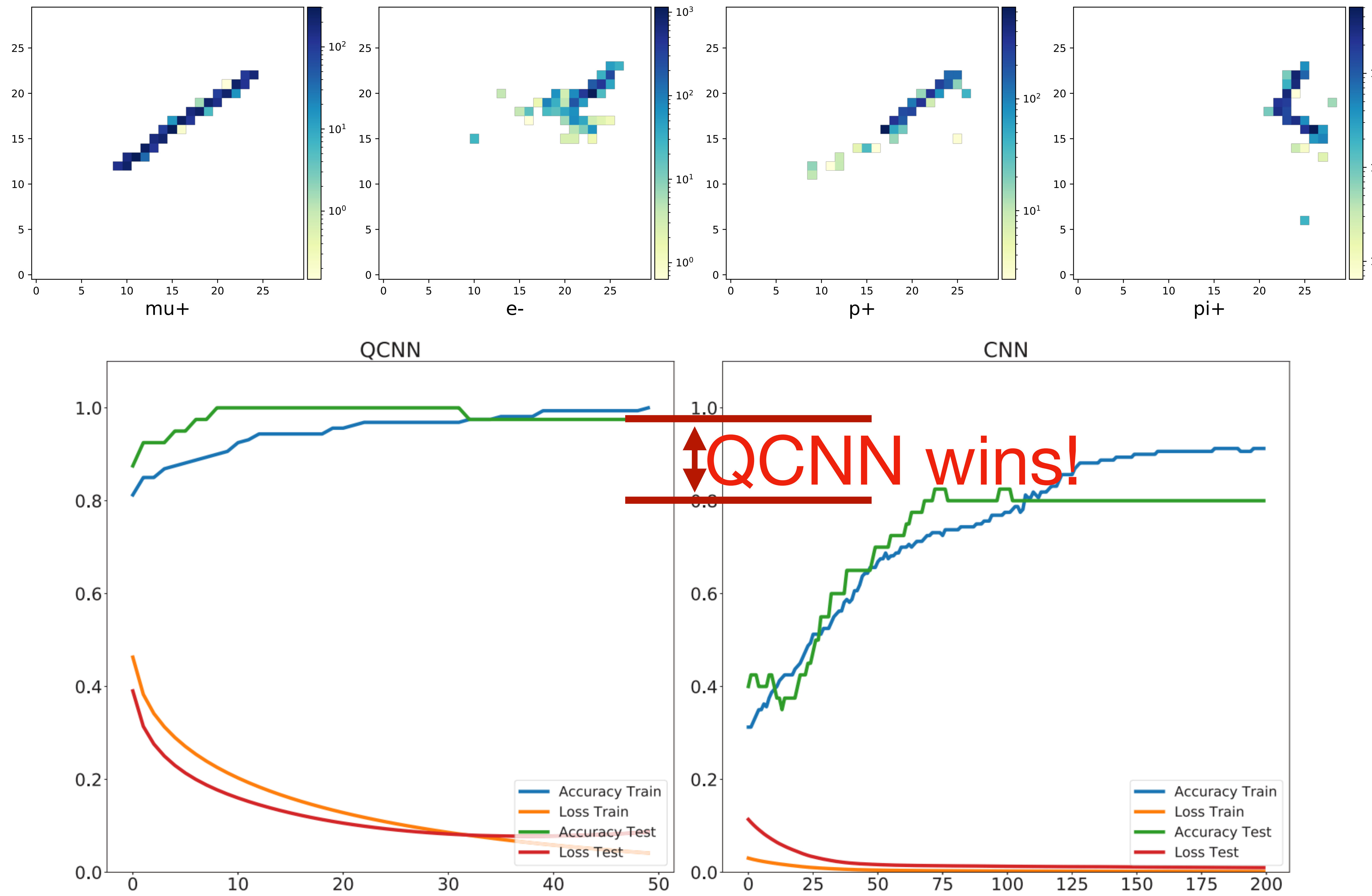
Chen, S. Y. C., Wei, T. C., Zhang, C., Yu, H., & Yoo, S. (2022). **Quantum convolutional neural networks for high energy physics data analysis.** *Physical Review Research*, *4*(1), 013231.

Quantum CNN



Chen, S. Y. C., Wei, T. C., Zhang, C., Yu, H., & Yoo, S. (2022). **Quantum convolutional neural networks for high energy physics data analysis.** *Physical Review Research*, *4*(1), 013231.

Quantum CNN

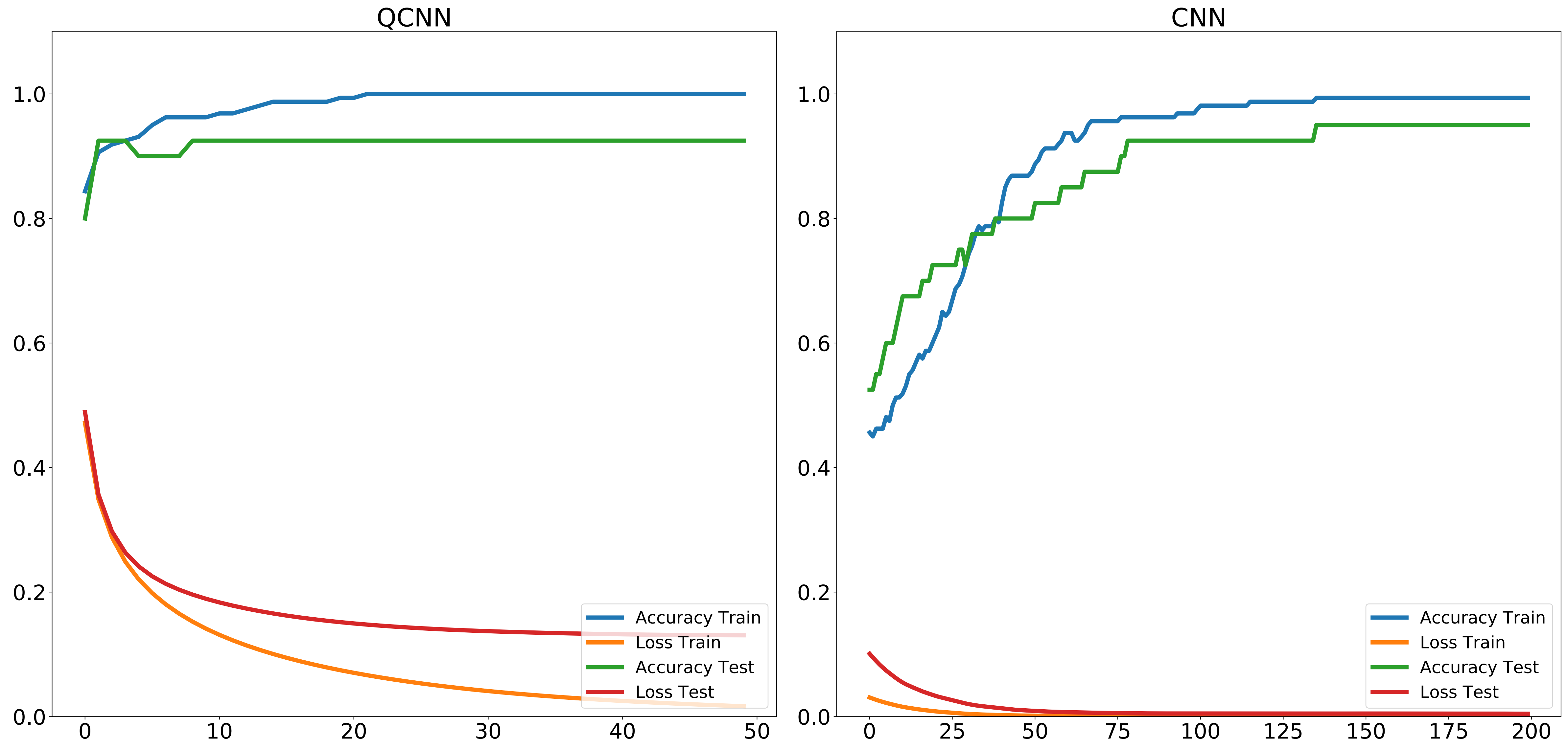


QCNN wins!

FIG. 8. QCNN on binary classification of muon vs proton. Training the QCNN for the classification of μ^+ and p. The filter size is 3 in the first convolutional layer and 2 in the second convolutional layer. There is 1 channel in both convolutional layers. The numbers of parameters in this setting are $9 \times 3 \times 2 = 54$ in the first convolutional layer, $4 \times 3 \times 2 = 24$ in the second convolutional layer, and $14 \times 14 \times 1 \times 2 + 2 = 394$ in the fully connected layer. The total number of parameters is $54 + 24 + 394 = 472$.

mu+ vs proton

QCNN reaches high accuracy with smaller number of epochs!



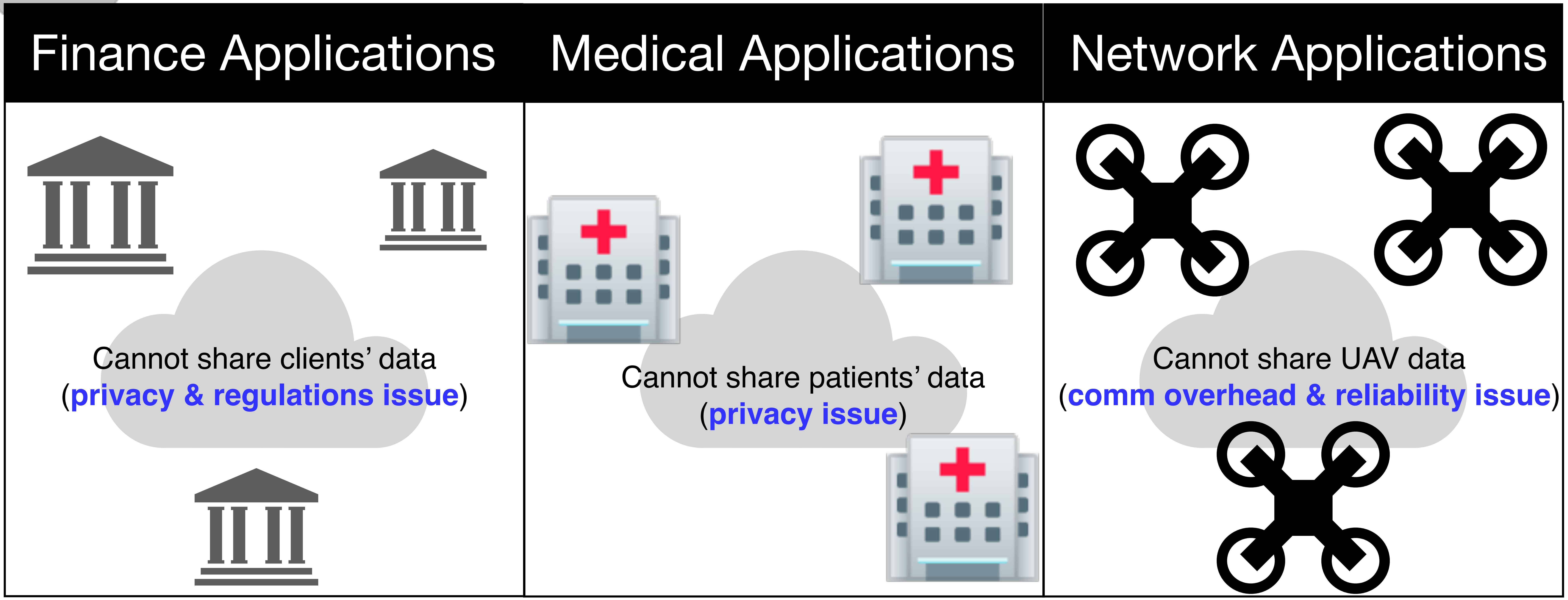
mu+ vs e-

Chen, S. Y. C., Wei, T. C., Zhang, C., Yu, H., & Yoo, S. (2022). **Quantum convolutional neural networks for high energy physics data analysis.** *Physical Review Research*, *4*(1), 013231.

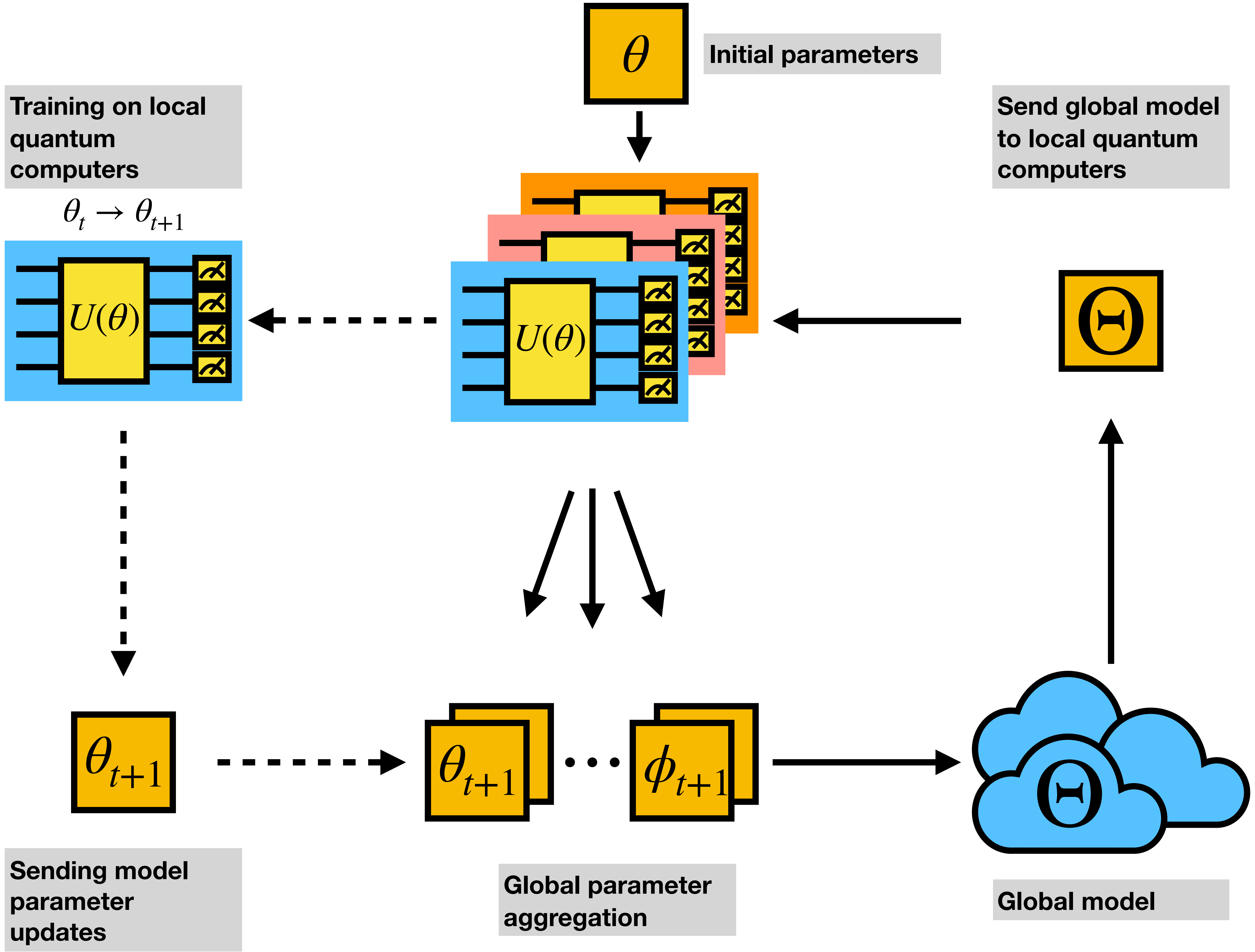
- Applications
    - Quantum Classification
    - Privacy-Preserving Quantum Machine Learning (Federated Learning, Differential Privacy)
    - Quantum Recurrent Neural Network
    - Quantum Reinforcement Learning
    - Quantum Natural Language Processing
    - Quantum Neural Networks for Model Compression

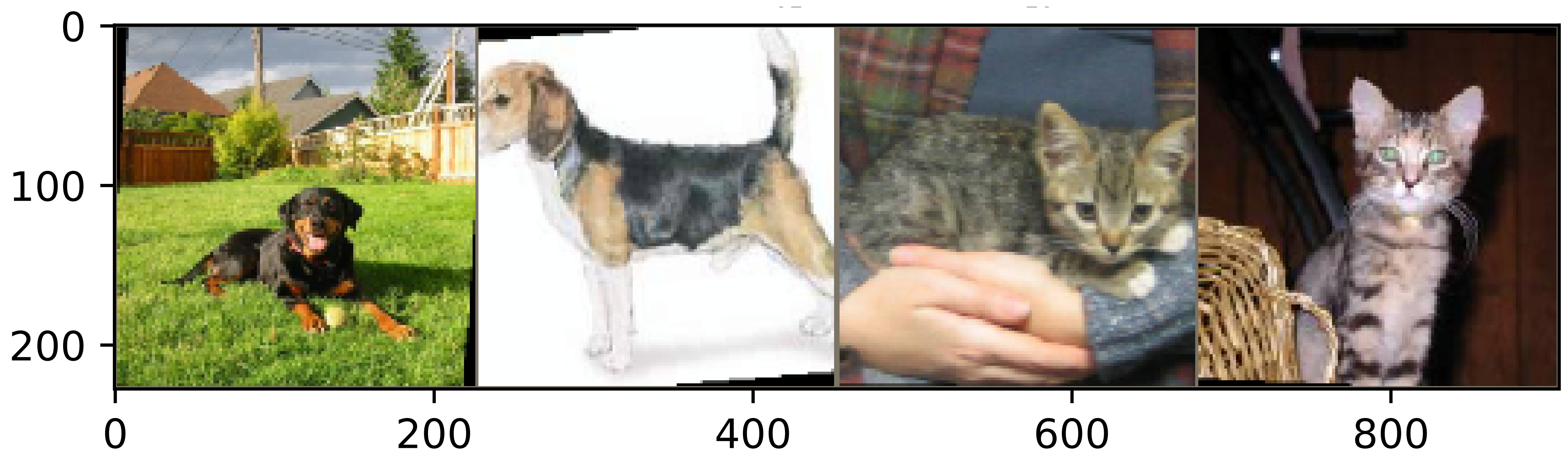
Why Federated Learning?

Having all data in a single storage is very hard in real-world applications!



Quantum Federated Learning





Chen, S. Y. C., & Yoo, S. (2021). **Federated quantum machine learning.** *Entropy*, *23*(4), 460.

Quantum Federated Learning

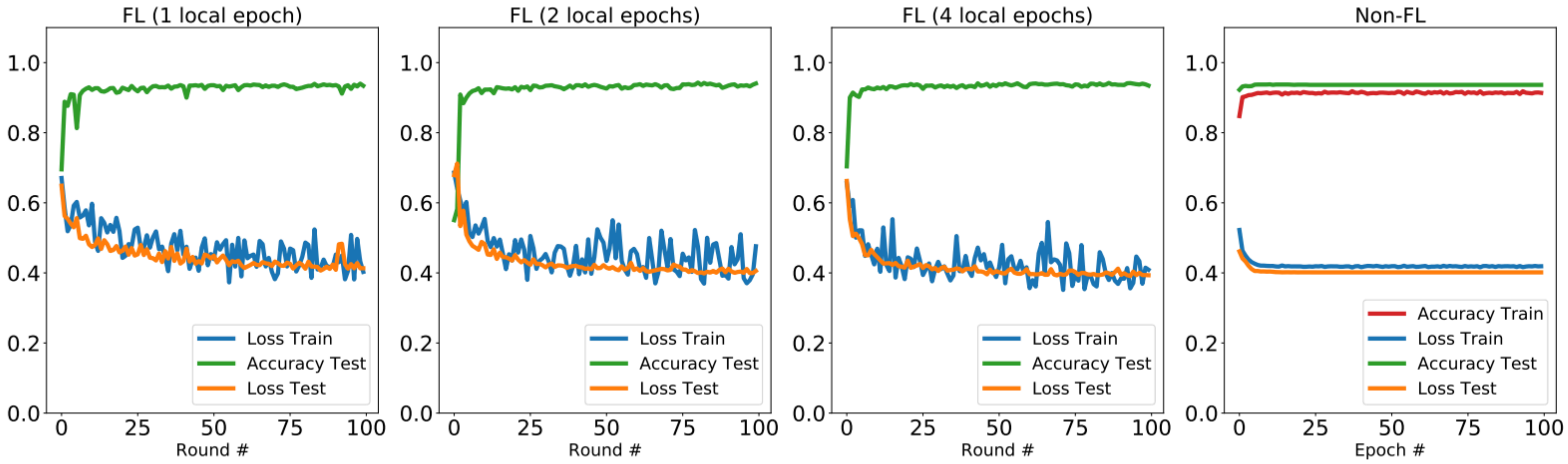


Figure 8. Results: Planes vs. Cars.

Table 3. Comparison of performance in different training schemes with CIFAR (Planes vs. Cars) dataset.

	Training Loss	**Testing Loss**	**Testing Accuracy**
Federated Training (1 local epoch)	0.4029	0.4133	93.40%
Federated Training (2 local epochs)	0.4760	0.4056	94.05%
Federated Training (4 local epochs)	0.4090	0.3934	93.45%
Non-Federated Training	0.4190	0.4016	93.65%

Chen, S. Y. C., & Yoo, S. (2021). **Federated quantum machine learning.** *Entropy*, *23*(4), 460.

Quantum Federated Learning with Quantum Data

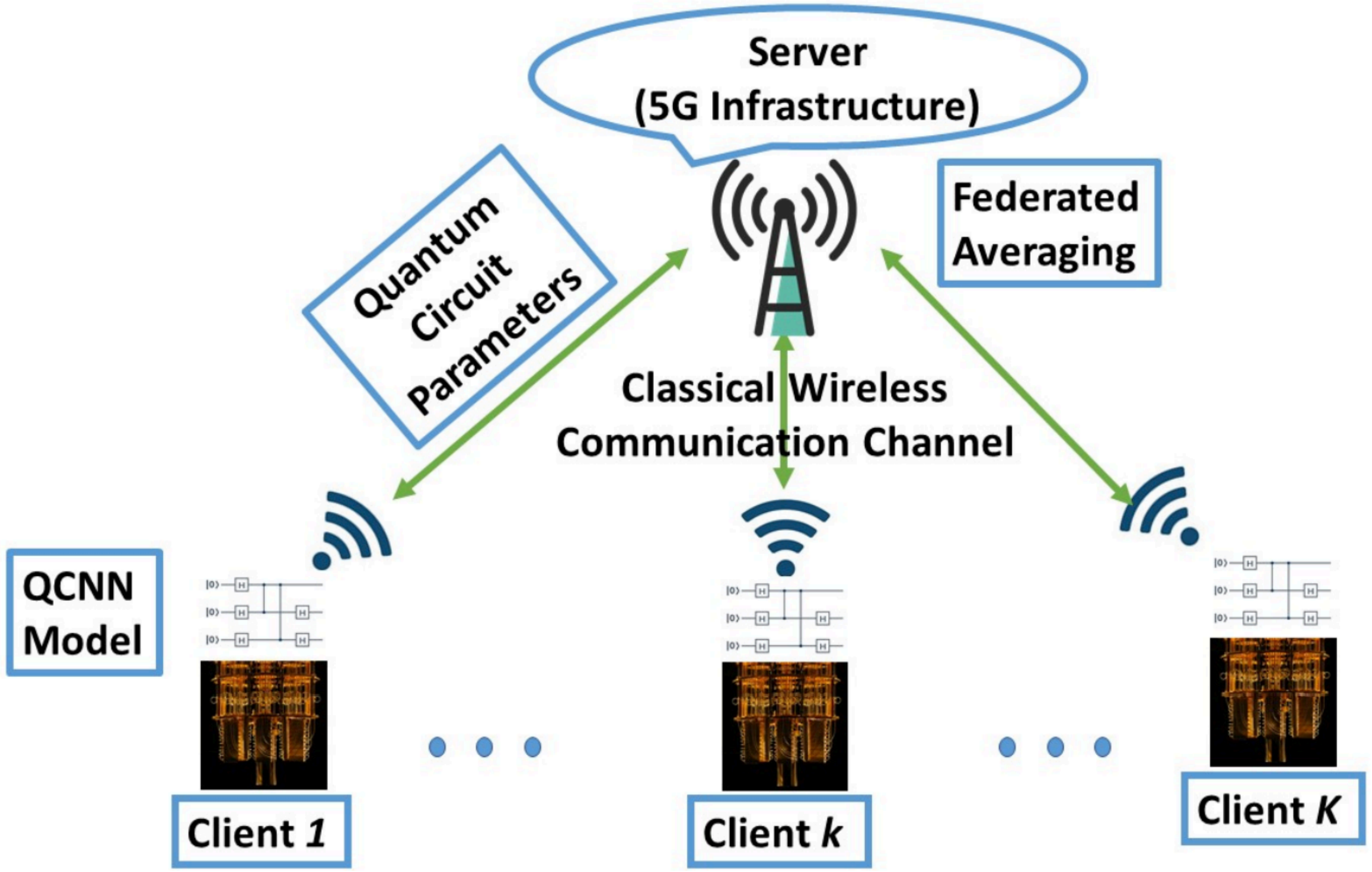


Fig. 1: Proposed general QFL setup.

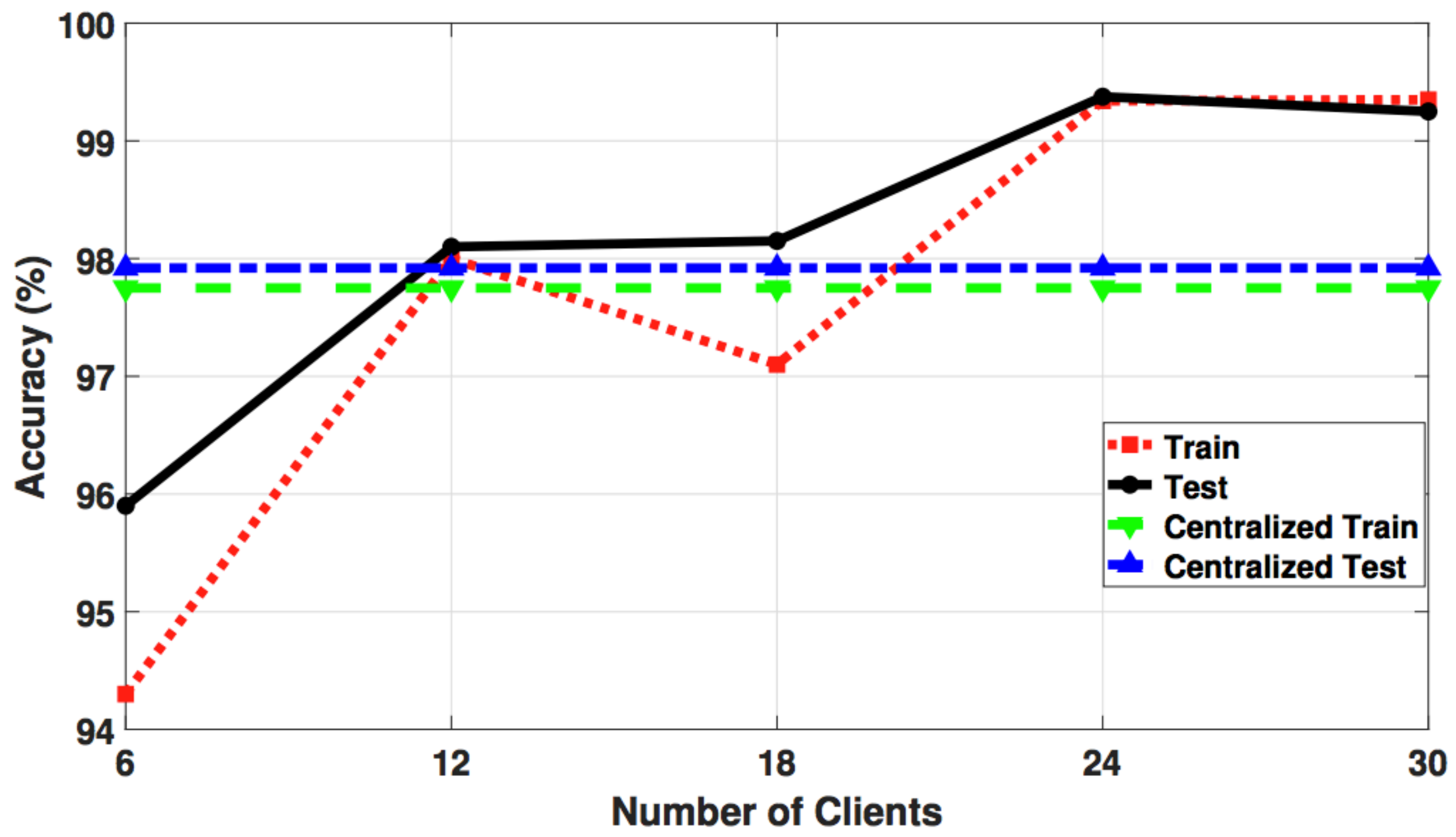


Fig. 2: Evaluation of QFL accuracy vs number of clients.

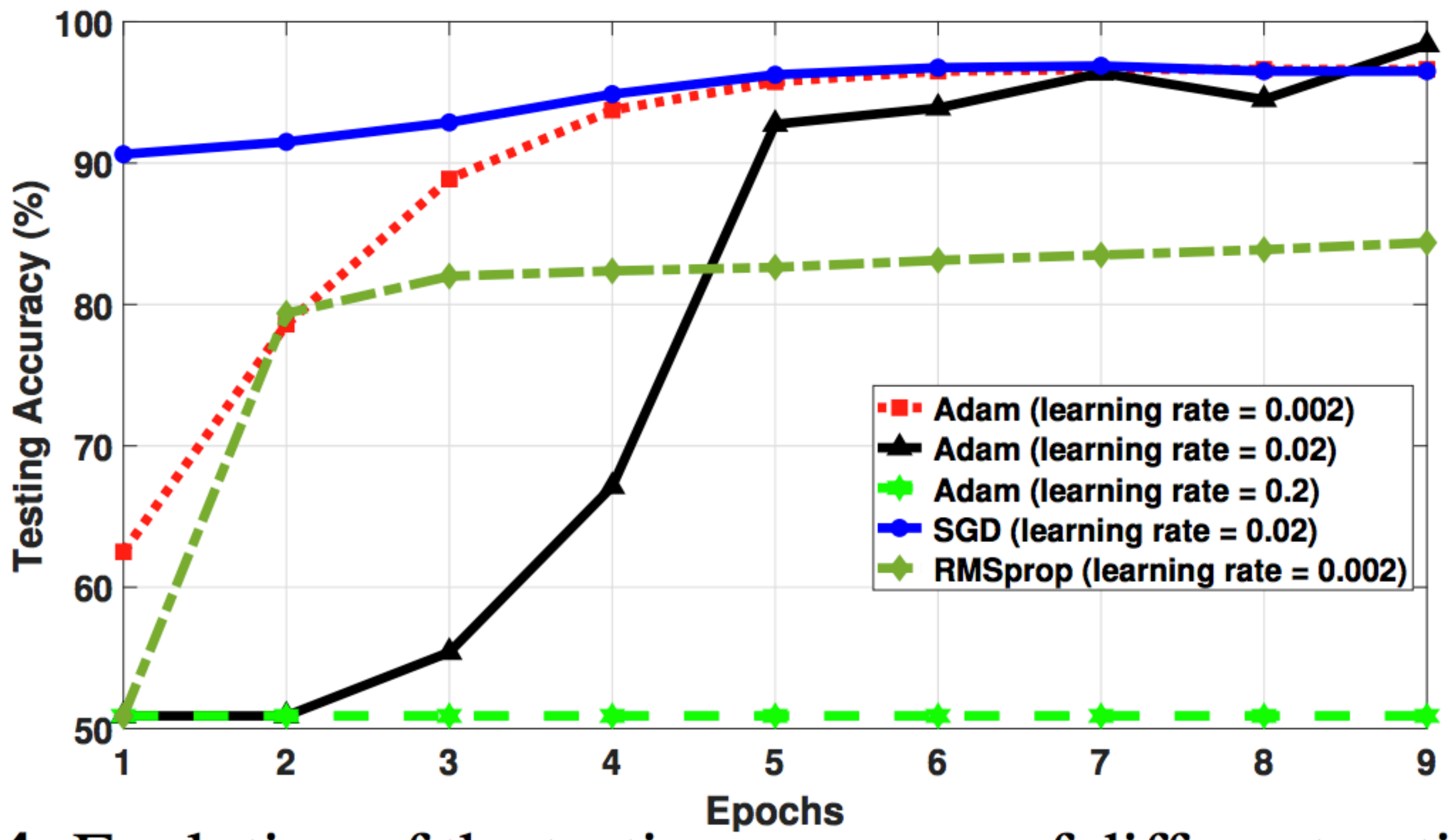


Fig. 4: Evolution of the testing accuracy of different optimizers over the training epochs.

Chehimi, M., & Saad, W. (2022, May). Quantum federated learning with quantum data. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 8617-8621). IEEE.

Quantum Federated ML with Differential Privacy

Algorithm 1 QFL-DP

Input: Examples $\{x_1, \ldots, x_M\}$, loss function $\mathcal{L}(\theta) = \frac{1}{N}\sum_i \mathcal{L}(\theta, x_i)$.

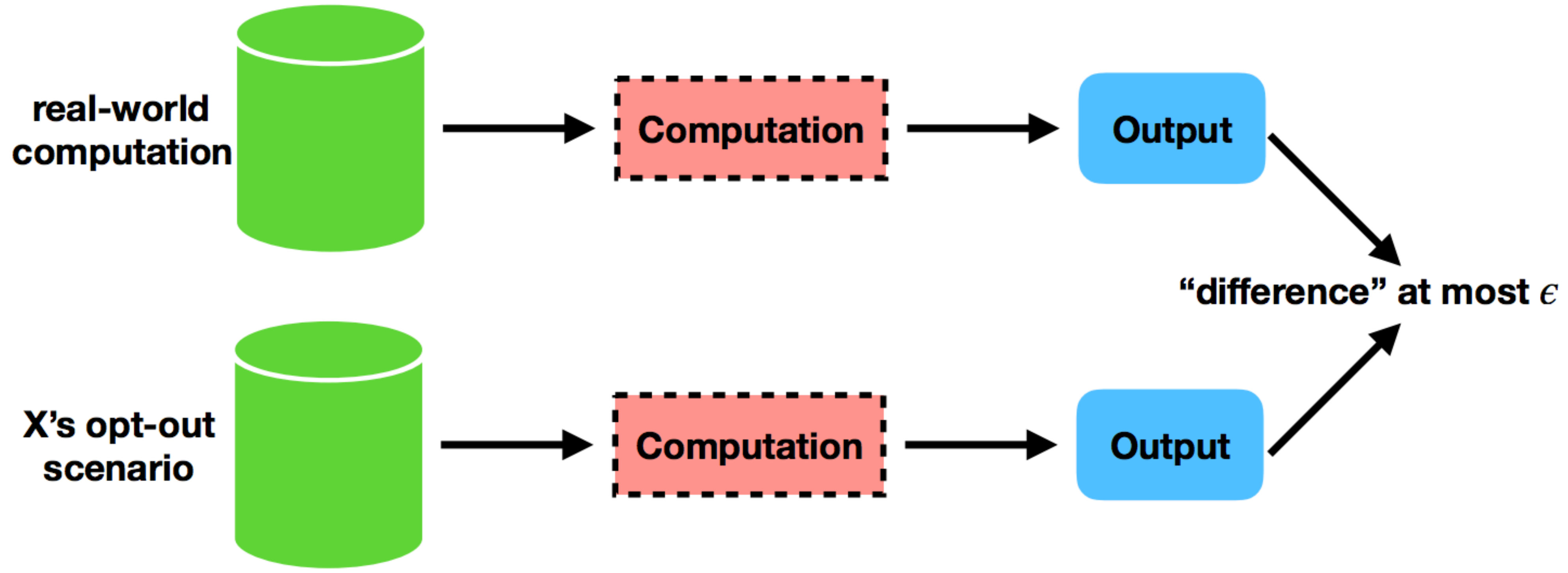
Parameters: Clients K, selected J, local epochs T, rounds R, learning rate η_t, noise scale σ, group size L, gradient norm bound C.

Partition: From M examples, construct $\mathcal{D}_1, \ldots, \mathcal{D}_K$ among K clients randomly, $|\mathcal{D}_i| = N = M/K$

Initialize: Quantum global model $\Theta_0 \in \mathbb{R}^n$

```
for r ∈ [R] do
    Model distribution:
    Make K identical copies of Θ_r for local set
    {Φ_r1, ..., Φ_rK} and send Φ_rk to client k
    Take random sample J from K clients
    for j ∈ [J] do
        for t ∈ [T] do
            DP client update:
            Perform DP-SGD(N, L, η_t, σ, L, C) on
            Φ_rj ← Φ̃_rj ≠ Φ_rj
        end for
    end for
    Model aggregation:
    Θ_{r+1} = averaging the parameters across
    each model in {Φ̃_rj}_{j=1}^J
end for
```

Output: Θ_R and compute the overall privacy cost (ϵ, δ) using a privacy accounting method.



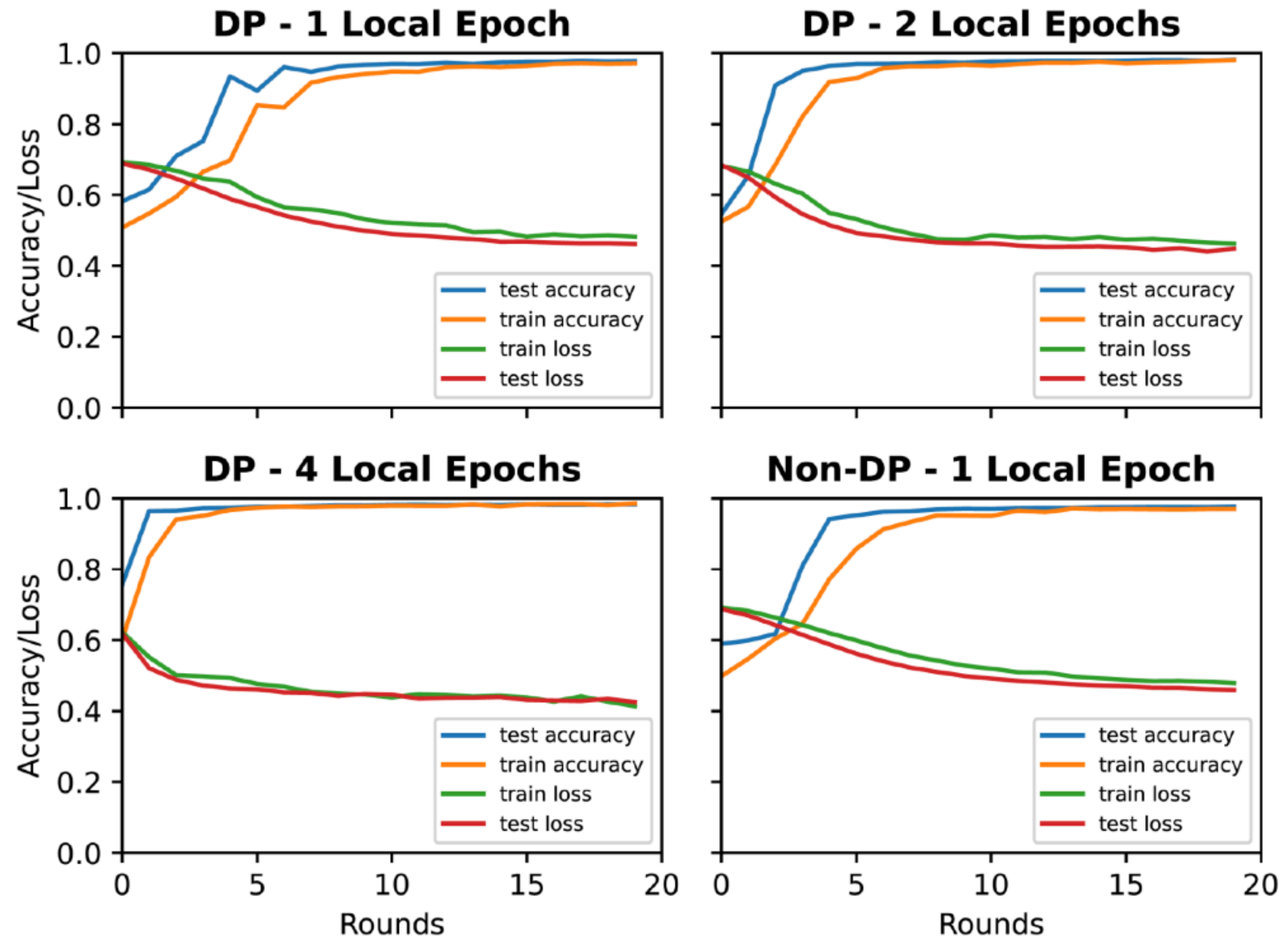


Fig. 6. All DP plots are $(\epsilon = 1.24, \delta = 10^{-5})$-DP and acquire test accuracy converging at approximately 0.98.

Rofougaran, R., Yoo, S., Tseng, H. H., & Chen, S. Y. C. (2023). **Federated Quantum Machine Learning with Differential Privacy.** *ICASSP 2024*

Watkins, W. M., Chen, S. Y. C., & Yoo, S. (2023). **Quantum machine learning with differential privacy.** *Scientific Reports*, *13*(1), 2453.

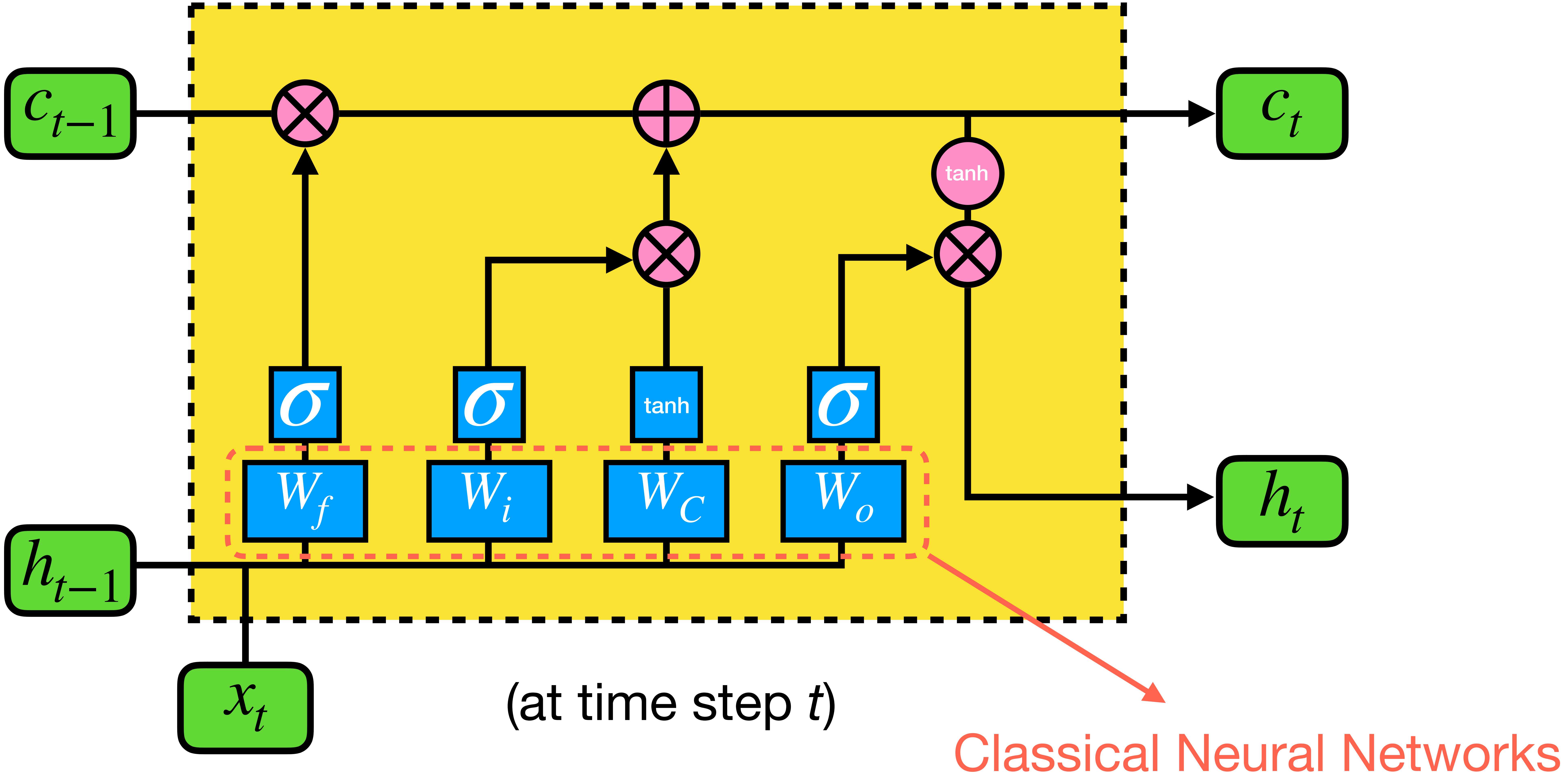
- Applications
  - Quantum Classification
  - Privacy-Preserving Quantum Machine Learning (Federated Learning, Differential Privacy)
  - Quantum Recurrent Neural Network
  - Quantum Reinforcement Learning
  - Quantum Natural Language Processing
  - Quantum Neural Networks for Model Compression

Quantum LSTM

Recurrent neural networks (RNN)

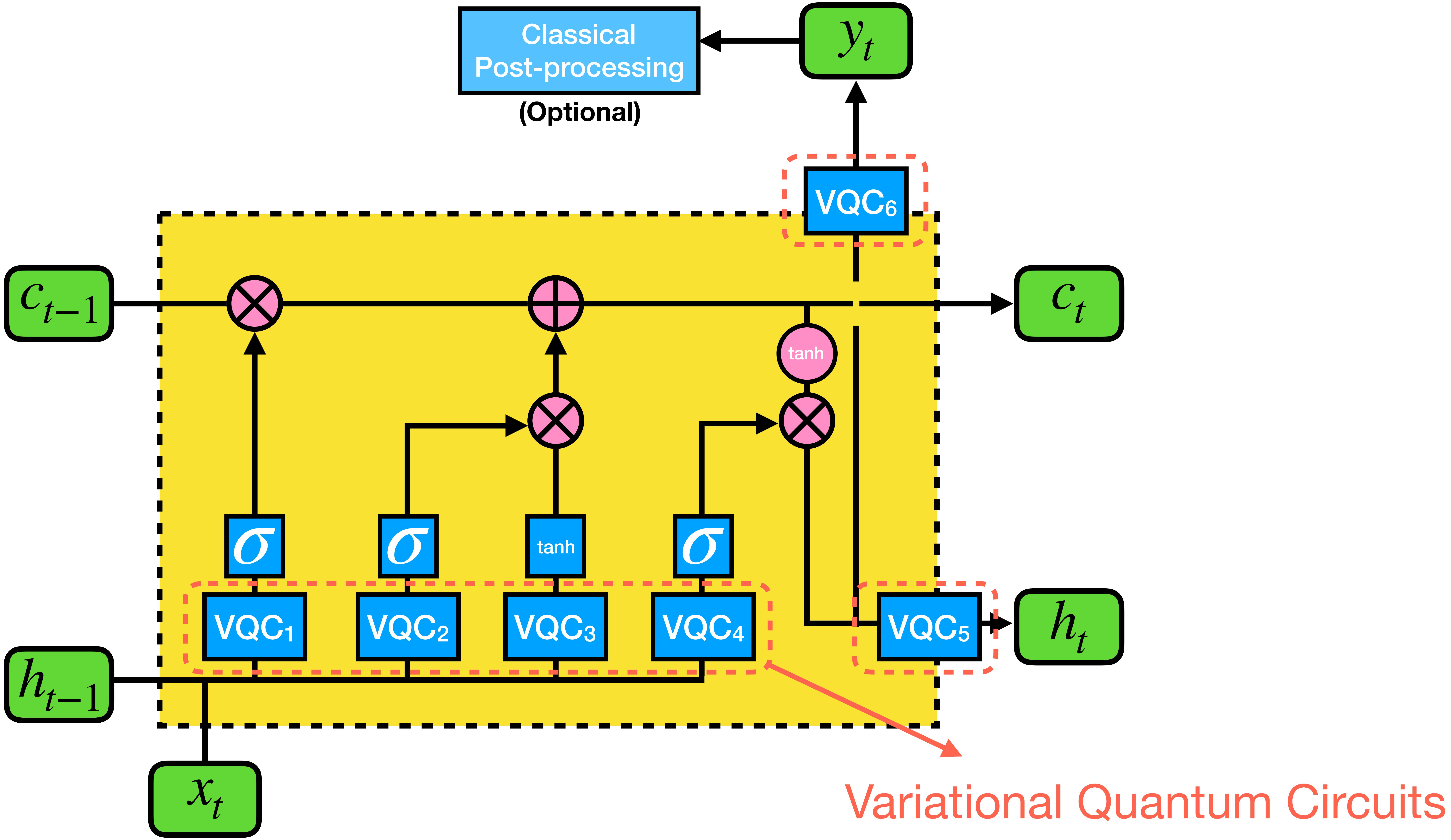
Quantum LSTM

(Classical) Long short-term memory (LSTM)



(at time step t)

Quantum LSTM

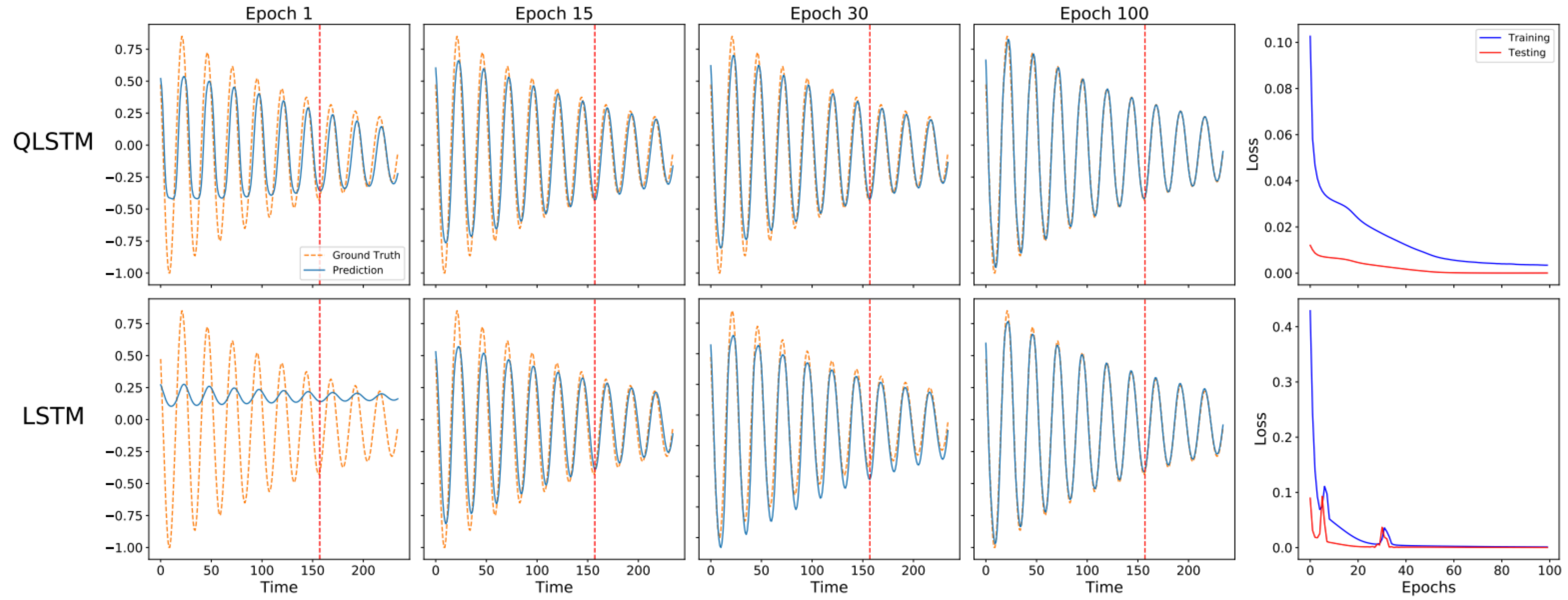


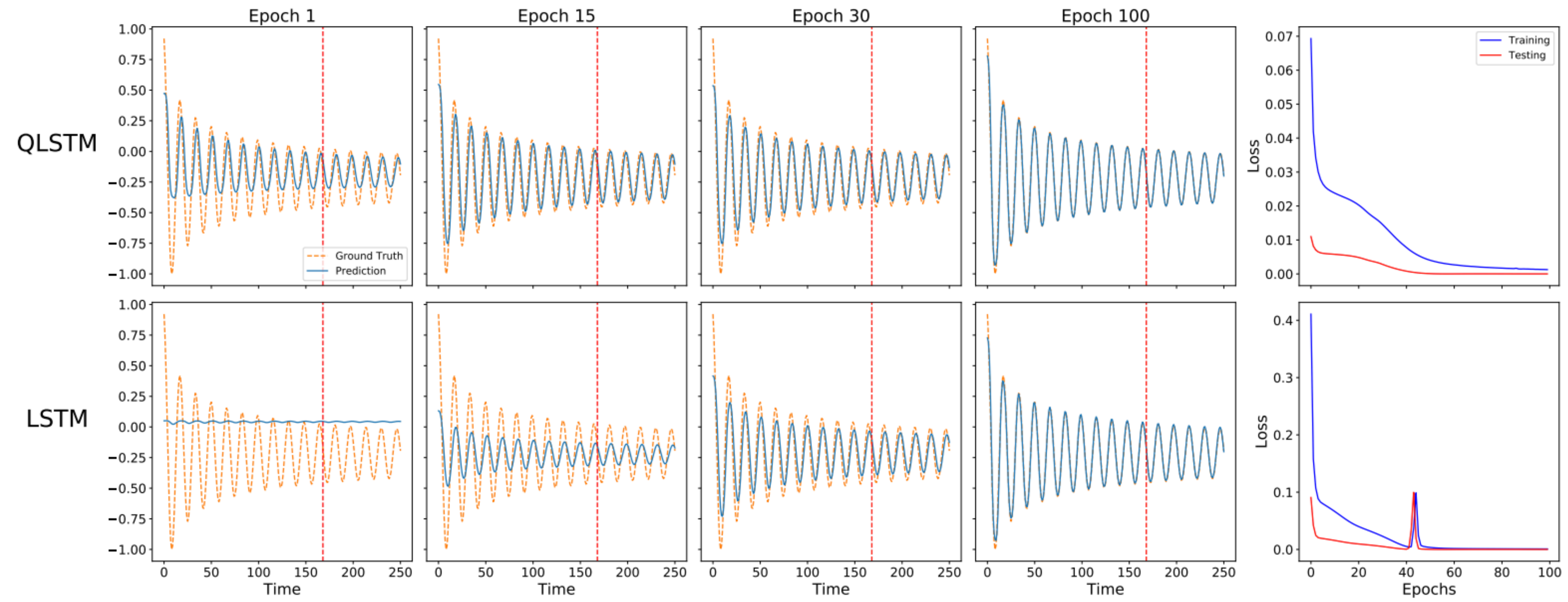
Quantum LSTM

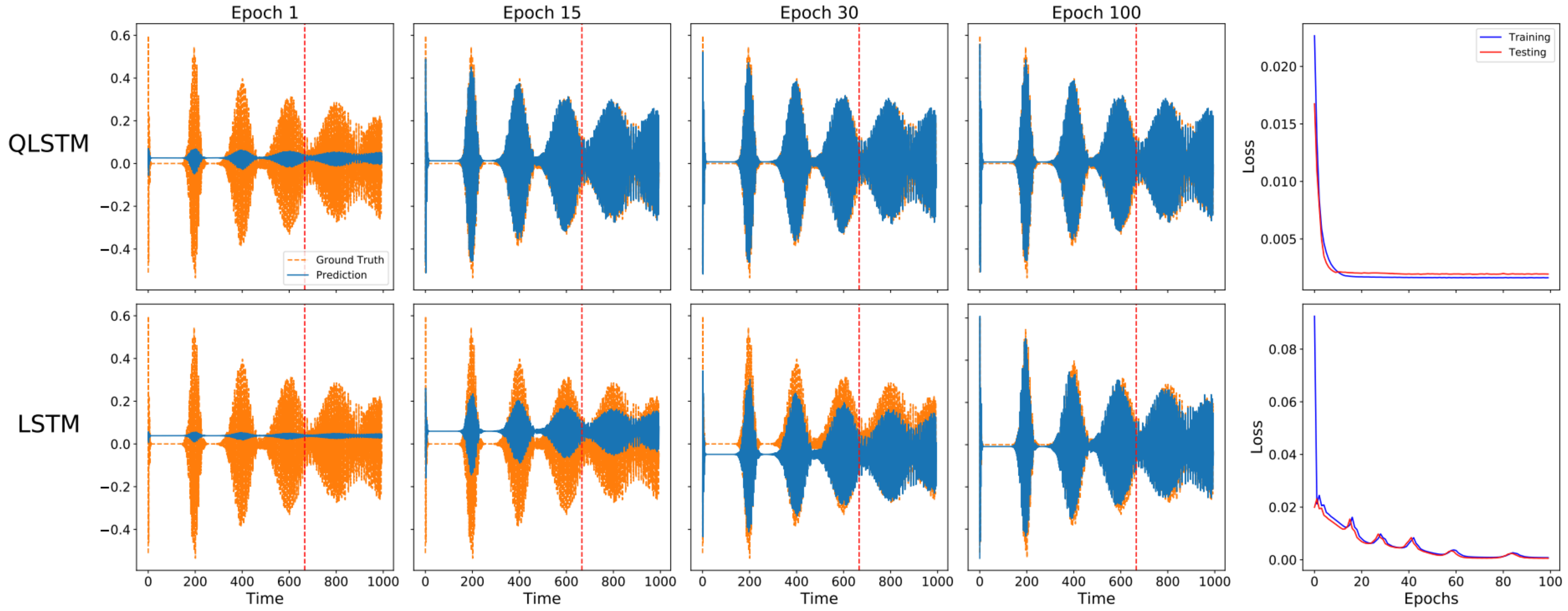
predict

time series $= \boxed{x_1, x_2, x_3, x_4}, x_5, x_6, x_7, \cdots$

given

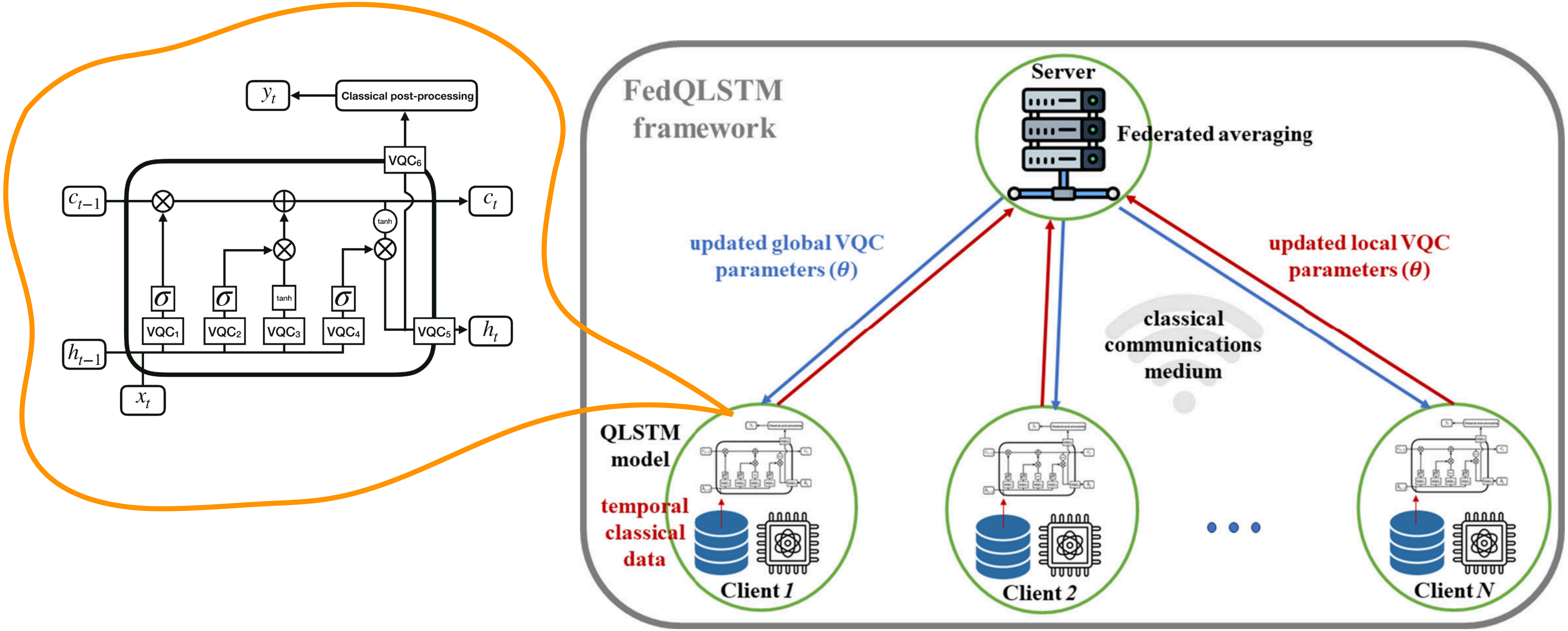






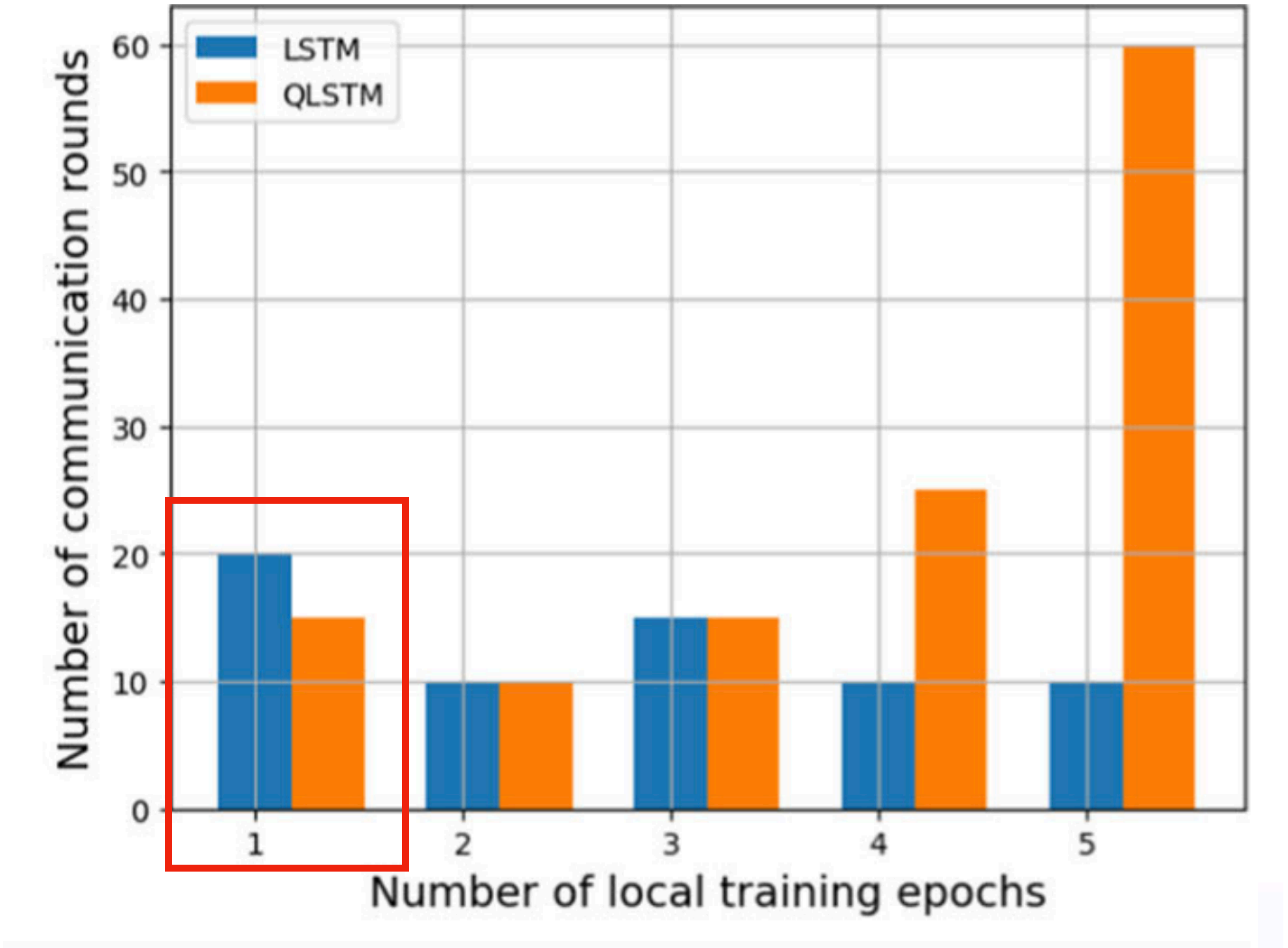
Chen, S. Y. C., Yoo, S., & Fang, Y. L. L. (2022, May). **Quantum long short-term memory.** In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 8622-8626). IEEE.

Federated QLSTM

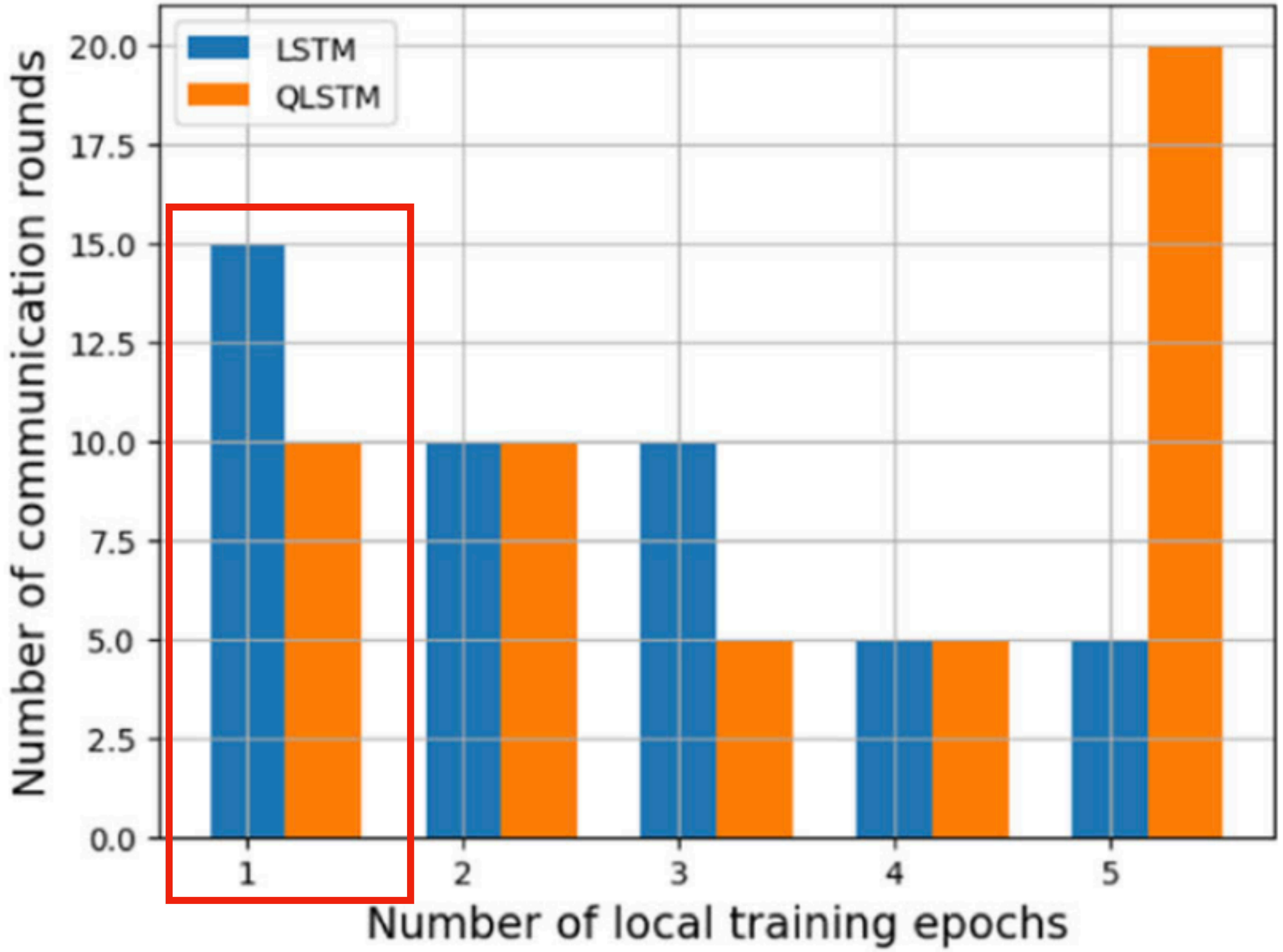


Chehimi, M., Chen, S. Y. C., Saad, W., & Yoo, S. (2024). Federated quantum long short-term memory (FedQLSTM). *Quantum Machine Intelligence*, *6*(2), 43.

Federated QLSTM



Bessel functions

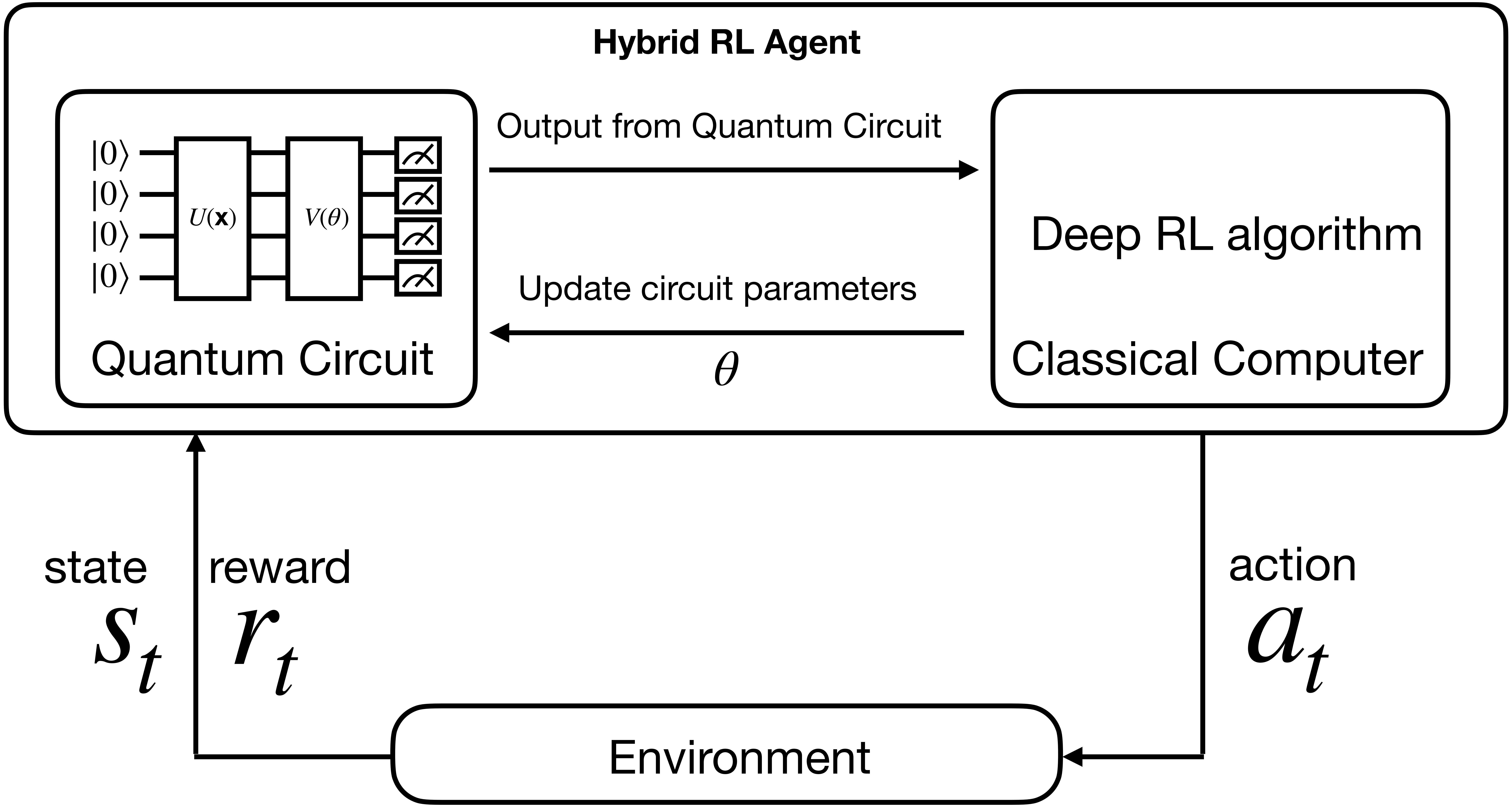


Delayed Quantum Control functions

Chehimi, M., Chen, S. Y. C., Saad, W., & Yoo, S. (2024). Federated quantum long short-term memory (FedQLSTM). *Quantum Machine Intelligence*, *6*(2), 43.

- **Applications**
  - Quantum Classification
  - Privacy-Preserving Quantum Machine Learning (Federated Learning, Differential Privacy)
  - Quantum Recurrent Neural Network
  - **Quantum Reinforcement Learning**
  - Quantum Natural Language Processing
  - Quantum Neural Networks for Model Compression

Quantum RL



Hybrid RL Agent
$|0\rangle$
$|0\rangle$
$|0\rangle$
$|0\rangle$
$U(\mathbf{x})$
$V(\theta)$
Quantum Circuit
Output from Quantum Circuit
Update circuit parameters
θ
Deep RL algorithm
Classical Computer
state
s_t
reward
r_t
action
a_t
Environment

Quantum RL

Variational Quantum Circuits for Deep Reinforcement Learning

SAMUEL YEN-CHI CHEN[1,2], CHAO-HAN HUCK YANG[3], JUN QI[3], (Member, IEEE), PIN-YU CHEN[4], (Member, IEEE), XIAOLI MA[3], (Fellow, IEEE), AND HSI-SHENG GOAN[1,2,5]

Quantum agents in the Gym: a variational quantum algorithm for deep Q-learning

Andrea Skolik[1,2], Sofiene Jerbi[3], and Vedran Dunjko[1]

Parametrized Quantum Policies for Reinforcement Learning

Sofiene Jerbi
Institute for Theoretical Physics,
University of Innsbruck
sofiene.jerbi@uibk.ac.at

Casper Gyurik
LIACS,
Leiden University

Simon C. Marshall
LIACS,
Leiden University

Hans J. Briegel
Institute for Theoretical Physics,
University of Innsbruck

Vedran Dunjko
LIACS,
Leiden University

A Survey on Quantum Reinforcement Learning

Nico Meyer, Christian Ufrecht, Maniraman Periyasamy, Daniel D. Scherer, Axel Plinge, and Christopher Mutschler

Fraunhofer IIS, Fraunhofer Institute for Integrated Circuits IIS, Nuremberg, Germany
{firstname.lastname|daniel.scherer2}@iis.fraunhofer.de

An Introduction to Quantum Reinforcement Learning (QRL)

Samuel Yen-Chi Chen
Wells Fargo
New York, NY, USA
yen-chi.chen@wellsfargo.com

Quantum Multi-Agent Reinforcement Learning via Variational Quantum Circuit Design

[†]Won Joon Yun, [†]Yunseok Kwak, [†]Jae Pyoung Kim, [§]Hyunhee Cho, [‡]Soyi Jung, [°]Jihong Park, and [†]Joongheon Kim
[†]School of Electrical Engineering, Korea University, Seoul, Republic of Korea
[§]School of Electronic and Electrical Engineering, Sungkyunkwan University, Suwon, Republic of Korea
[‡]School of Software, Hallym University, Chuncheon, Republic of Korea
[°]School of Information Technology, Deakin University, Geelong, Victoria, Australia

Quantum RL

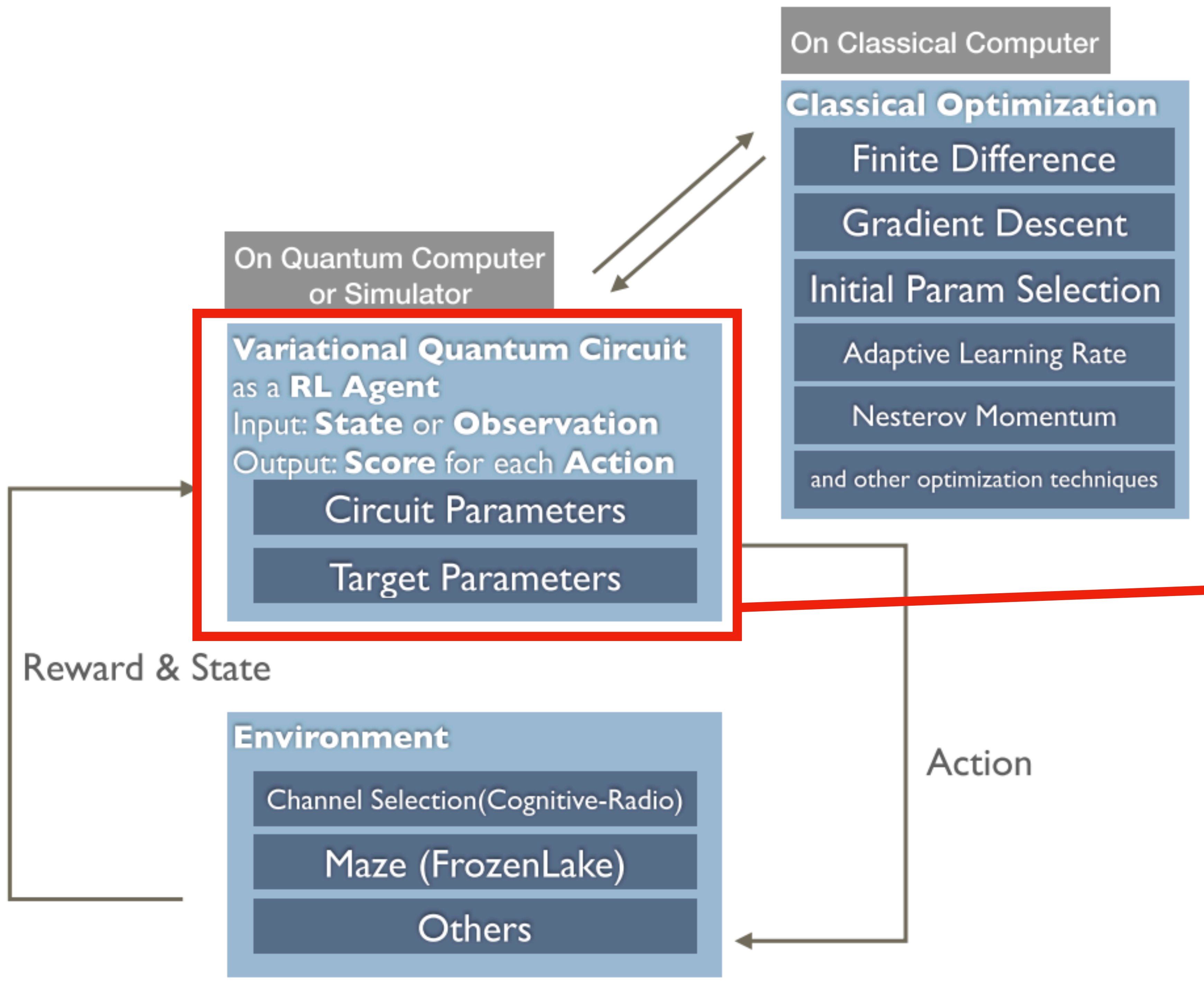
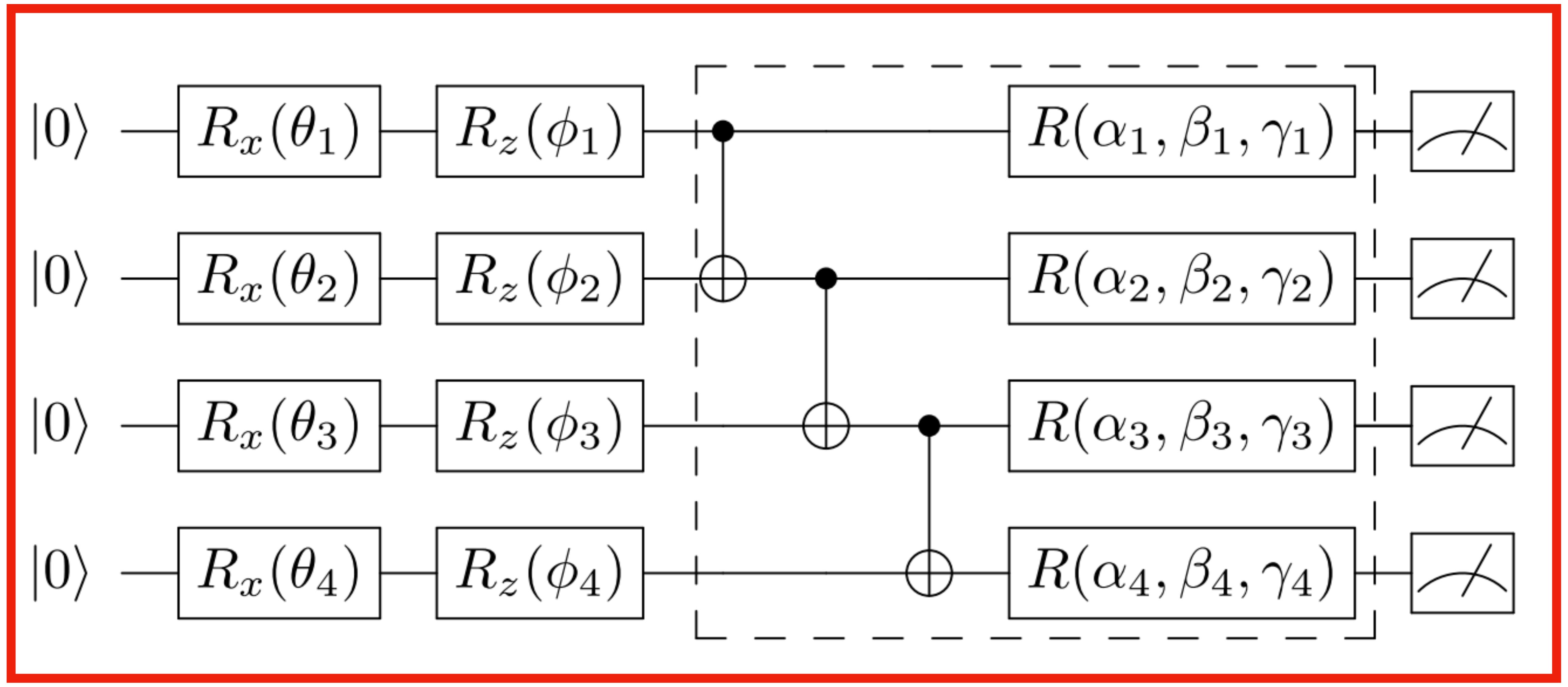


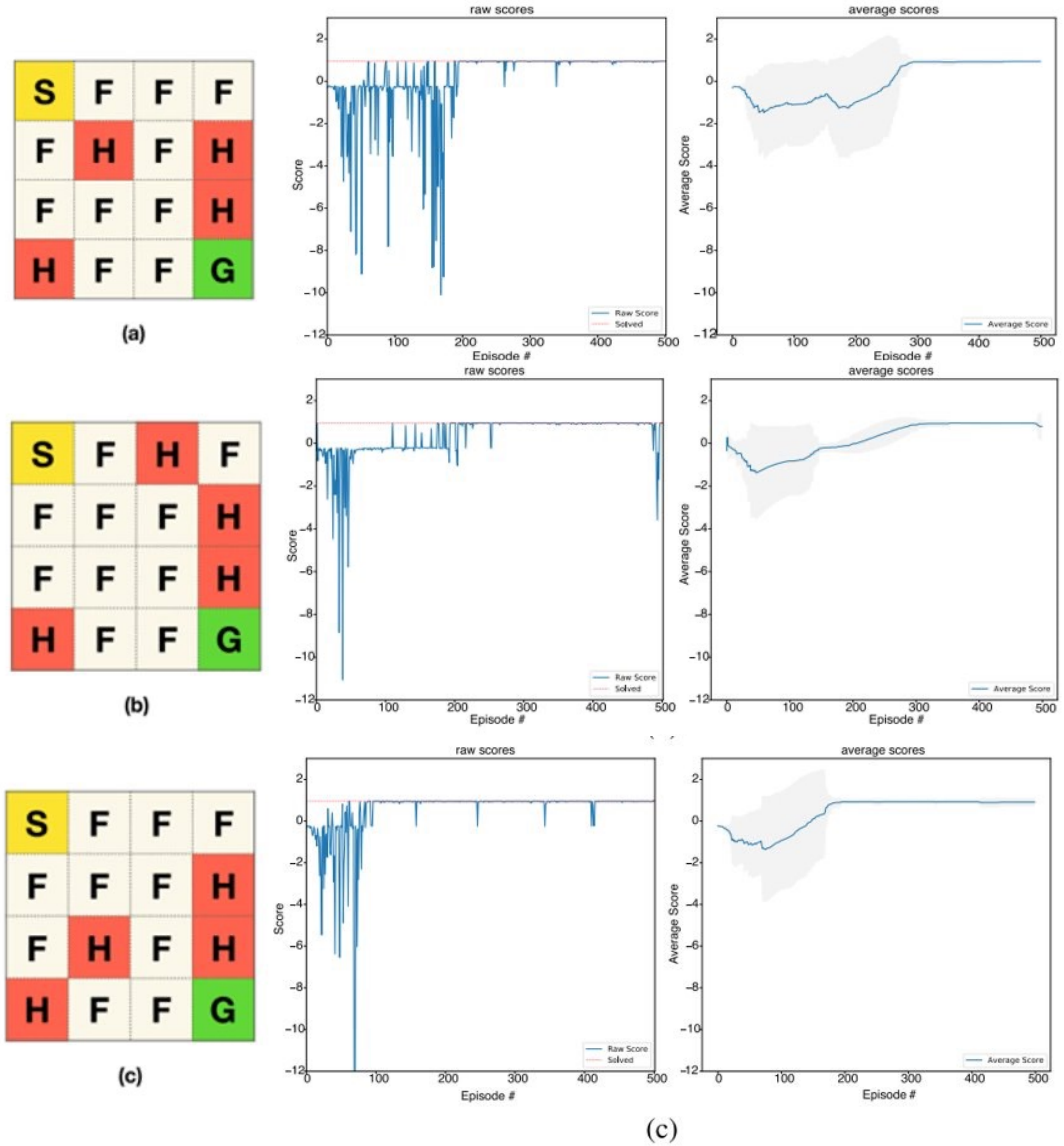
FIGURE 4. Overview of variational quantum circuits for DRL. In this work, we study the capability of variational quantum circuits in performing DRL tasks. This DRL agent includes a quantum part and a classical part. Under current limitations on the scale of quantum machines and the capabilities of quantum simulations, we select frozen-lake and cognitive-radio environments for the proof-of-principle study. The proposed framework is rather general and is expected to solve complicated tasks when larger-scale quantum machines are available.



IEEE Access, *8*, 141007-141024.

Quantum RL

- Environment with 16 states
- States numbered as 0-15
- Example:
  - State 12: 1100 -> 1,1,0,0
    - Rotation: $$\theta_i = \pi \times b_i$$ $$\phi_i = \pi \times b_i$$
    - Result: $|1\rangle \otimes |1\rangle \otimes |0\rangle \otimes |0\rangle$



Quantum Policy Gradient

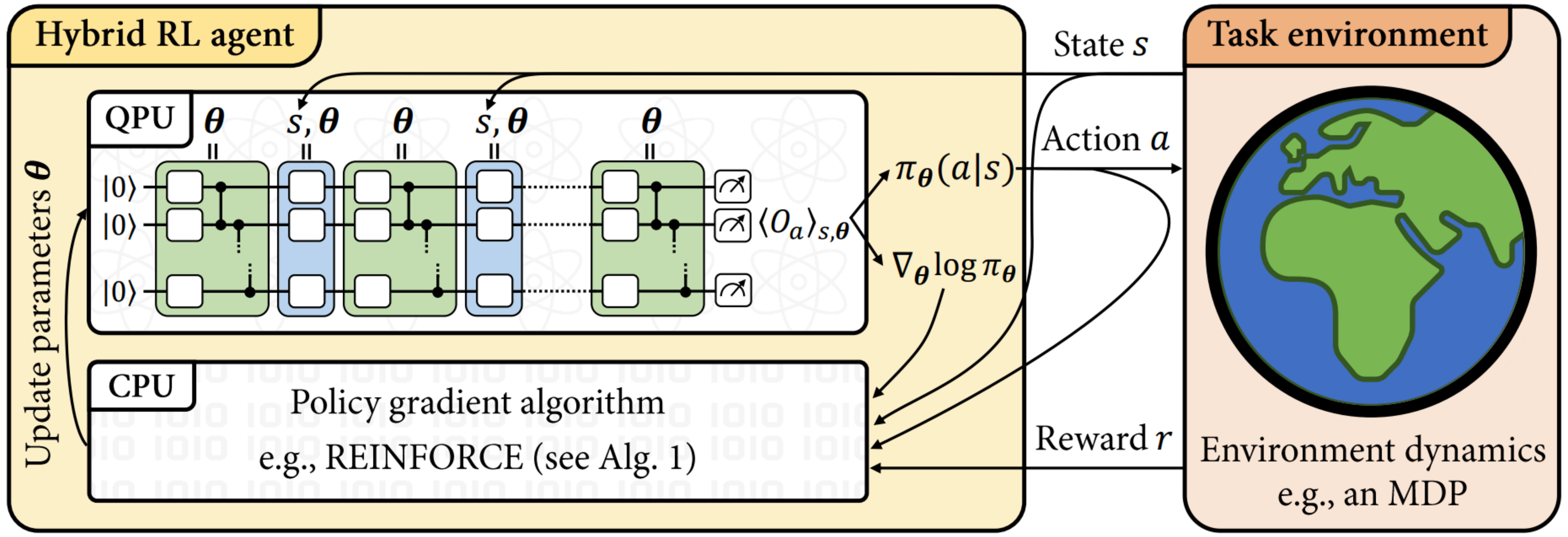
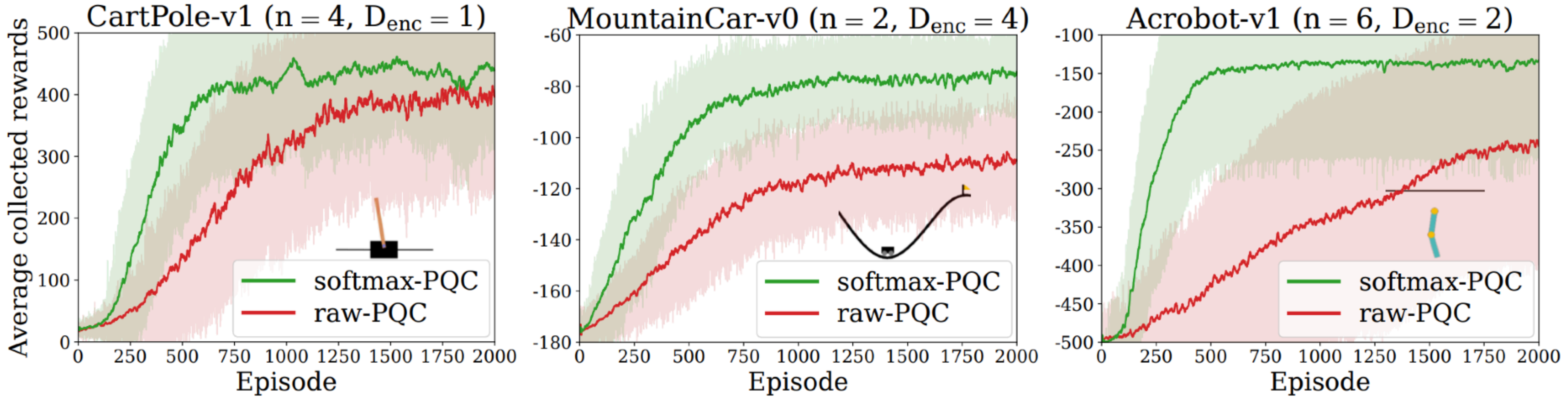


Figure 1: **Training parametrized quantum policies for reinforcement learning.** We consider a quantum-enhanced RL scenario where a hybrid quantum-classical agent learns by interacting with a classical environment. For each state s it perceives, the agent samples its next action a from its policy $\pi_{\boldsymbol{\theta}}(a|s)$ and perceives feedback on its behavior in the form of a reward r. For our hybrid agents, the policy $\pi_{\boldsymbol{\theta}}$ is specified by a PQC (see Def. 1) evaluated (along with the gradient $\nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}$) on a quantum processing unit (QPU). The training of this policy is performed by a classical learning algorithm, such as the REINFORCE algorithm (see Alg. 1), which uses sample interactions and policy gradients to update the policy parameters $\boldsymbol{\theta}$.



Algorithm 1: REINFORCE with PQC policies and value-function baselines

Input: a PQC policy $\pi_{\boldsymbol{\theta}}$ from Def. 1; a value-function approximator $\widetilde{V}_{\boldsymbol{\omega}}$

Initialize parameters $\boldsymbol{\theta}$ and $\boldsymbol{\omega}$;

while *True* **do**

  Generate N episodes $\{(s_0, a_0, r_1, \ldots, s_{H-1}, a_{H-1}, r_H)\}_i$ following $\pi_{\boldsymbol{\theta}}$;

  for *episode i in batch* **do**

    Compute the returns $G_{i,t} \leftarrow \sum_{t'=1}^{H-t} \gamma^{t'} r_{t+t'}^{(i)}$;

    Compute the gradients $\nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a_t^{(i)}|s_t^{(i)})$ using Lemma 1;

  Fit $\left\{\widetilde{V}_{\boldsymbol{\omega}}(s_t^{(i)})\right\}_{i,t}$ to the returns $\{G_{i,t}\}_{i,t}$;

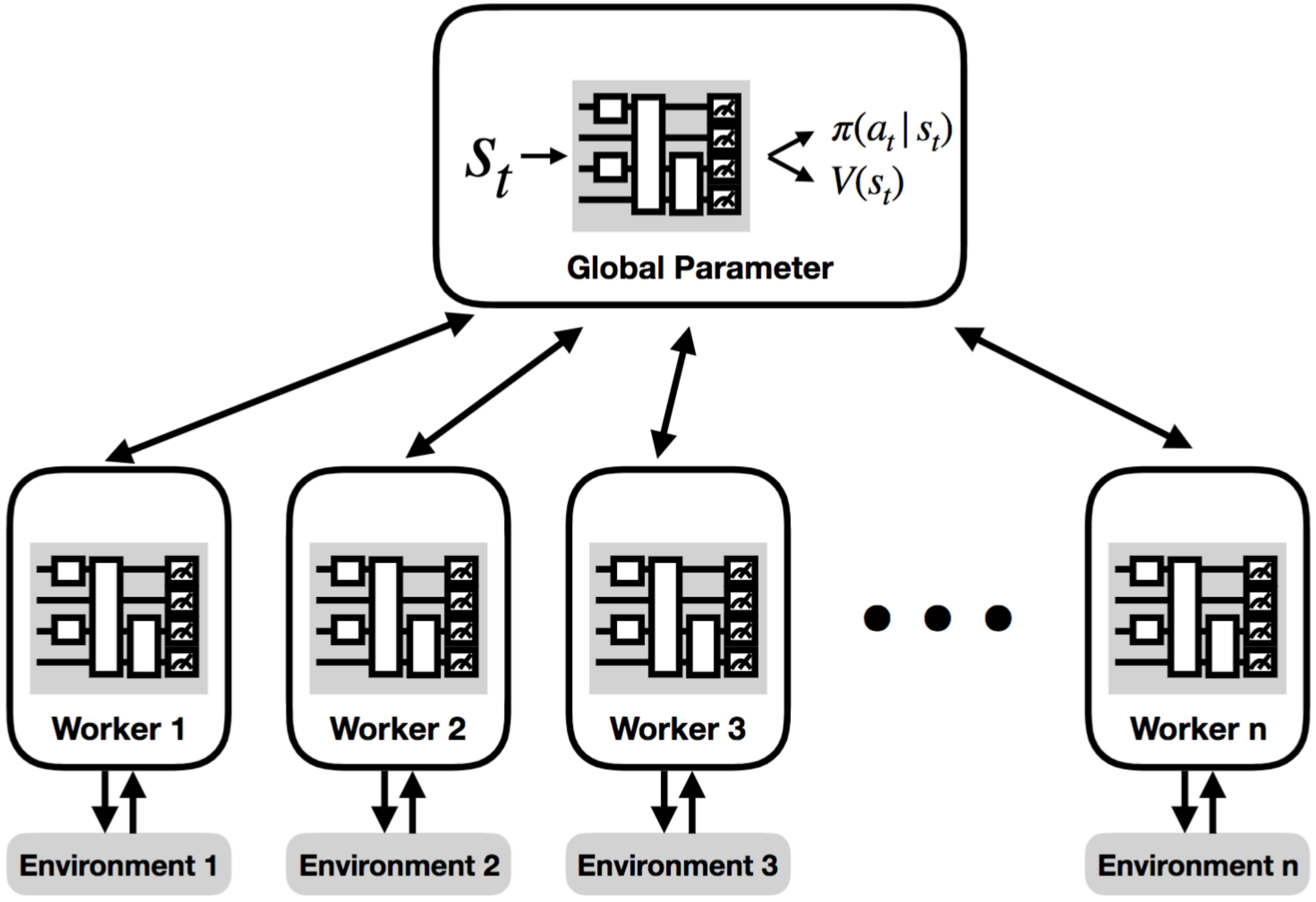
  Compute $\Delta\boldsymbol{\theta} = \dfrac{1}{N} \sum_{i=1}^{N} \sum_{t=0}^{H-1} \nabla_{\boldsymbol{\theta}} \log \pi_{\boldsymbol{\theta}}(a_t^{(i)}|s_t^{(i)}) \left(G_{i,t} - \widetilde{V}_{\boldsymbol{\omega}}(s_t^{(i)})\right)$;

  Update $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \alpha \Delta\boldsymbol{\theta}$;

Jerbi, S., Gyurik, C., Marshall, S., Briegel, H., & Dunjko, V. (2021). Parametrized quantum policies for reinforcement learning. *Advances in Neural Information Processing Systems*, *34*, 28362-28375.

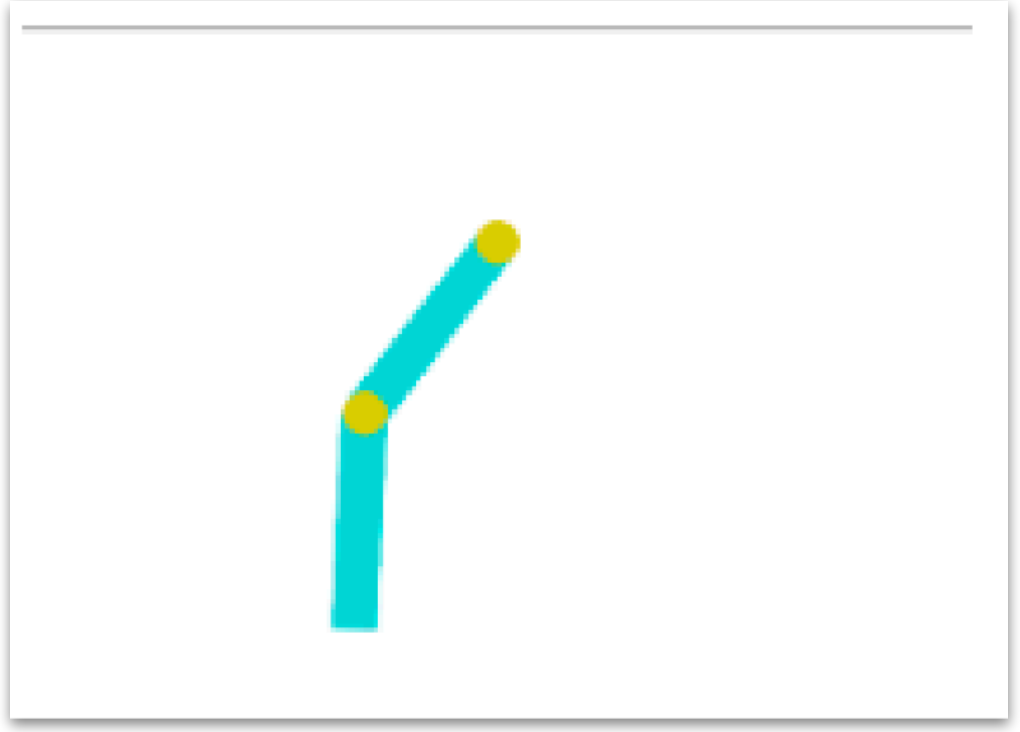
Asynchronous QRL

- **Multiple concurrent actors** learning the policy through parallelization.
- Executing multiple agents on multiple instances of the environments.
- Allowing the agents to encounter diverse states at on-policy RL such as actor-critic.
- No need of replay memory.

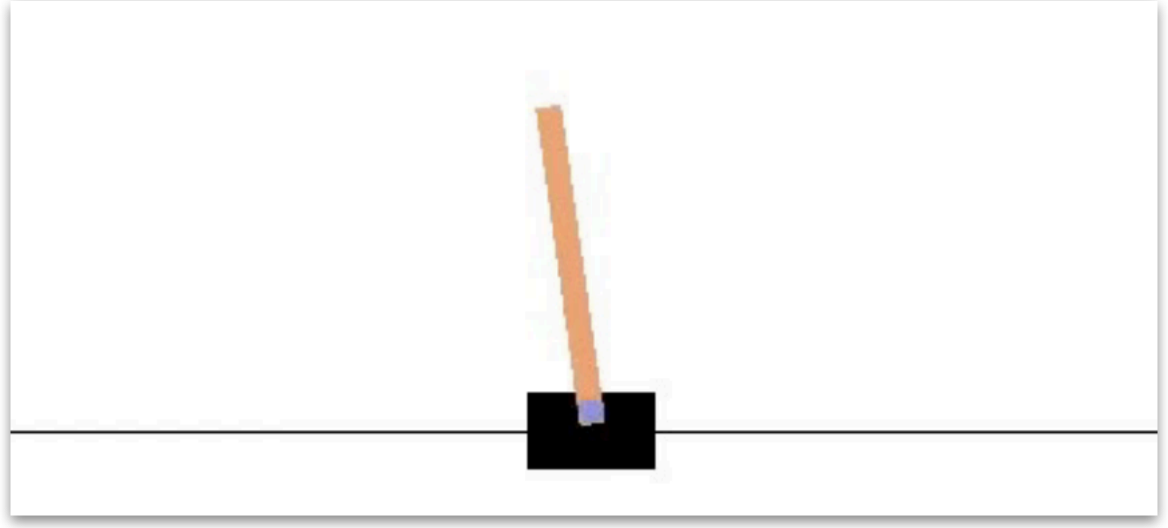


Chen, S. Y. C. (2023). **Asynchronous training of quantum reinforcement learning**. *Procedia Computer Science*, *222*, 321-330.

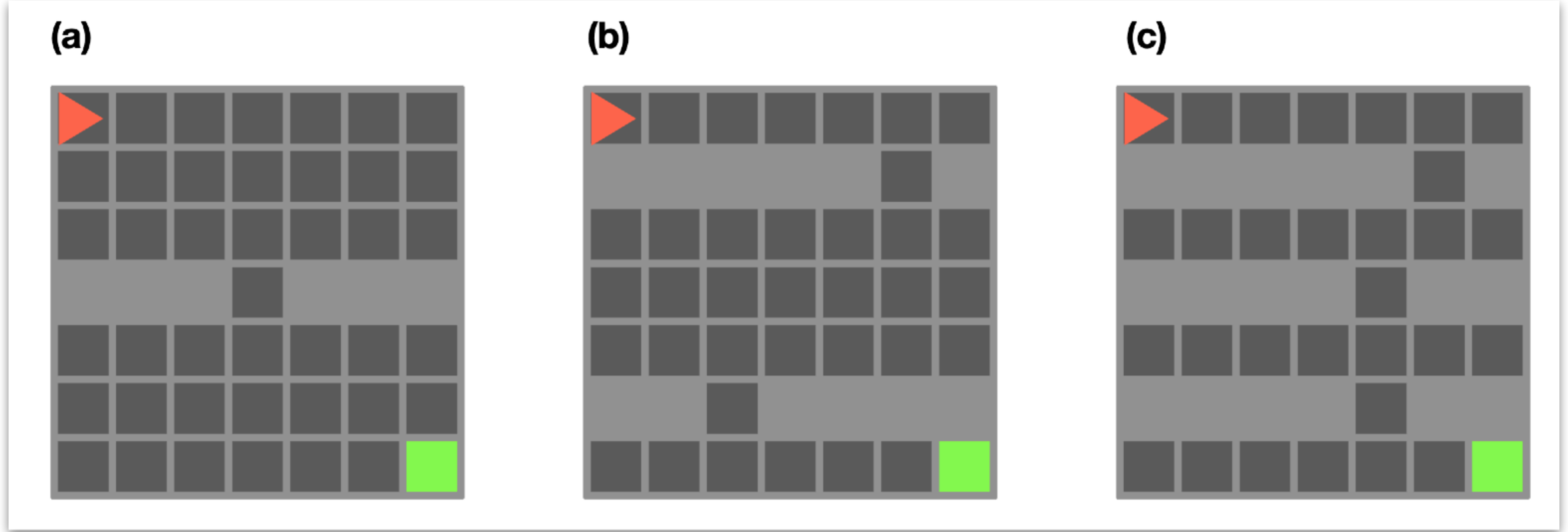
Asynchronous QRL



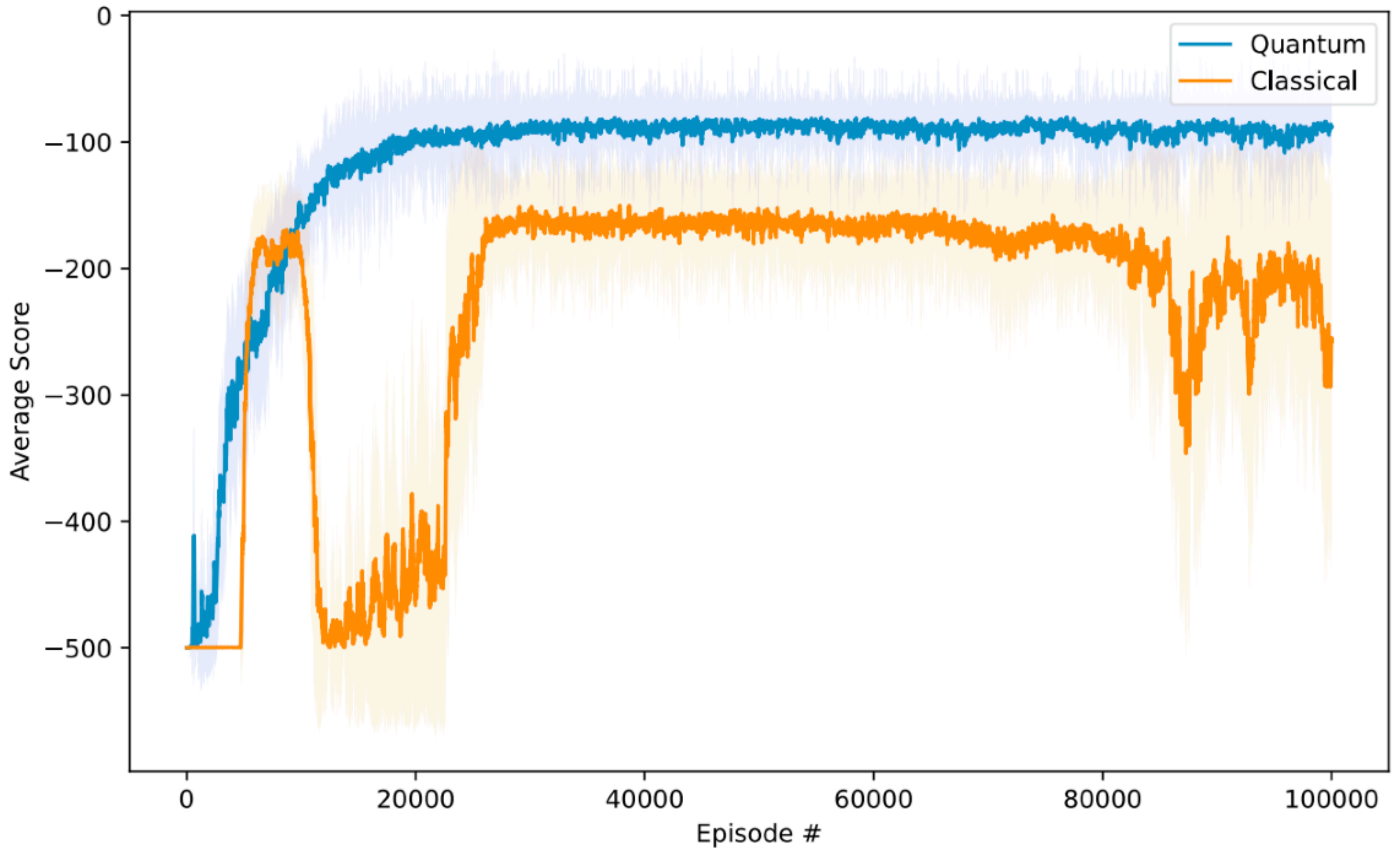
Acrobot



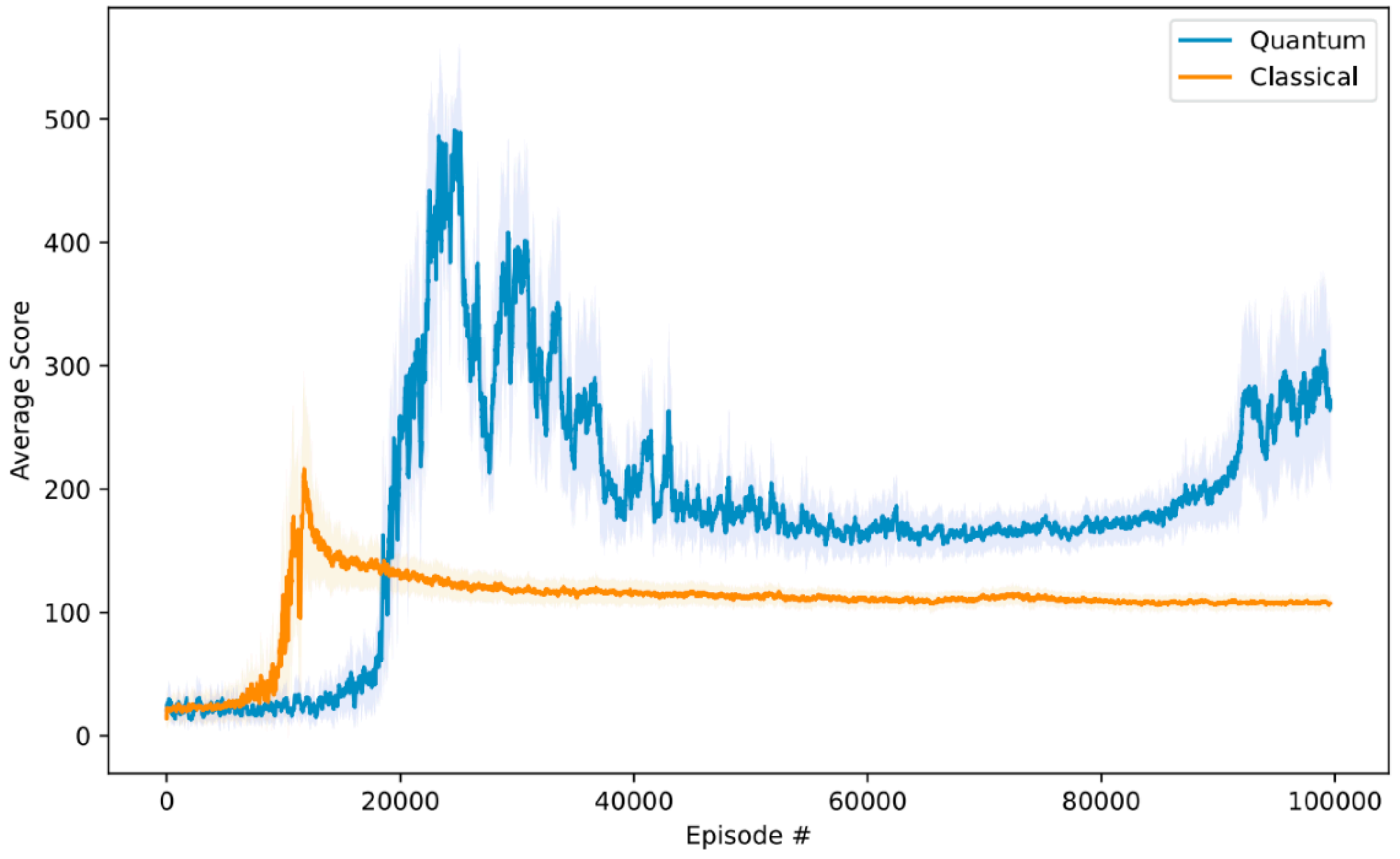
CartPole



S9N1 S9N2 S9N3



(a) **Results: Quantum A3C in the Acrobot environment.**



(b) **Results: Quantum A3C in the CartPole environment.**

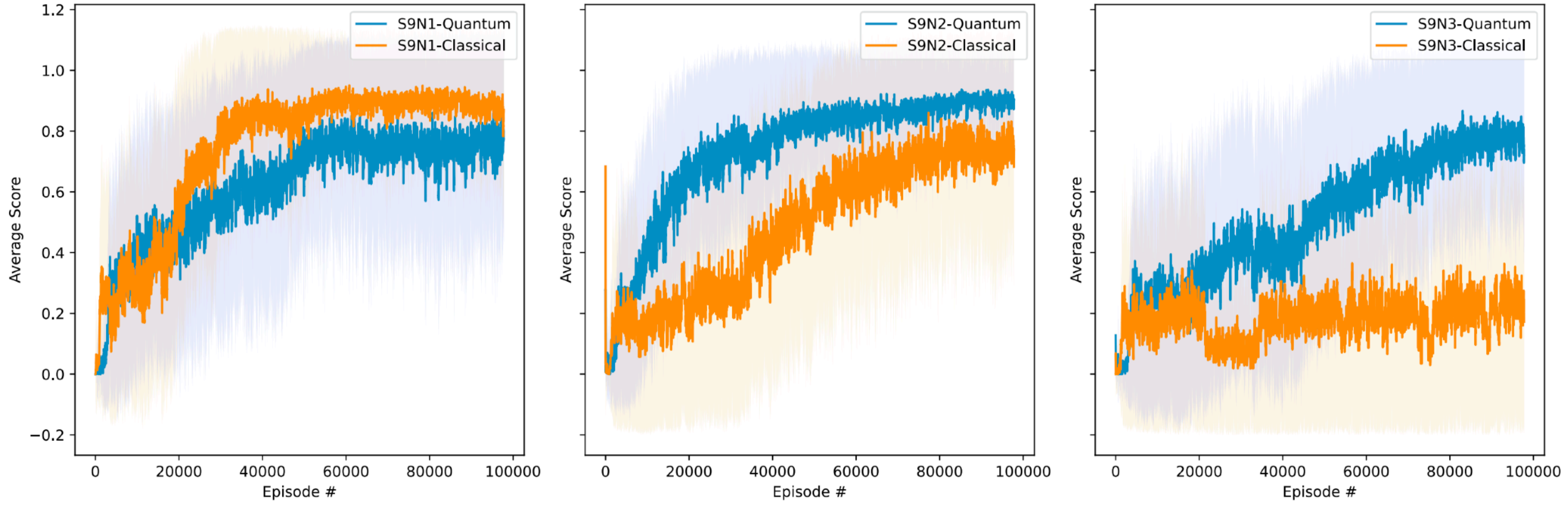
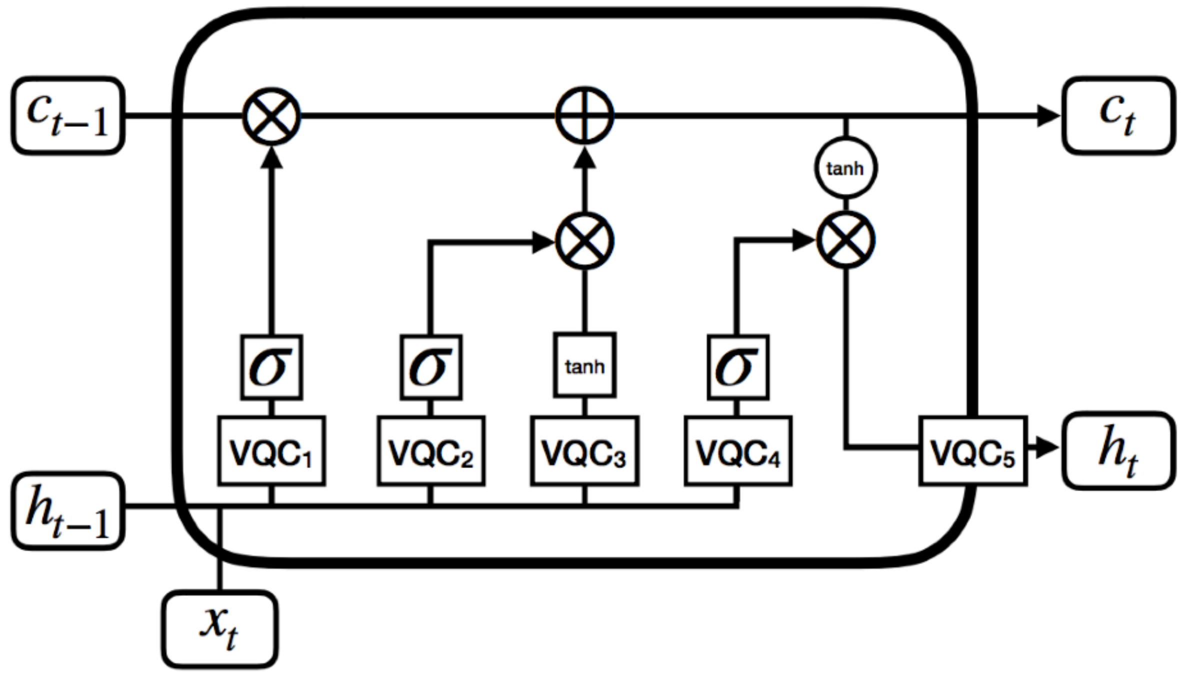


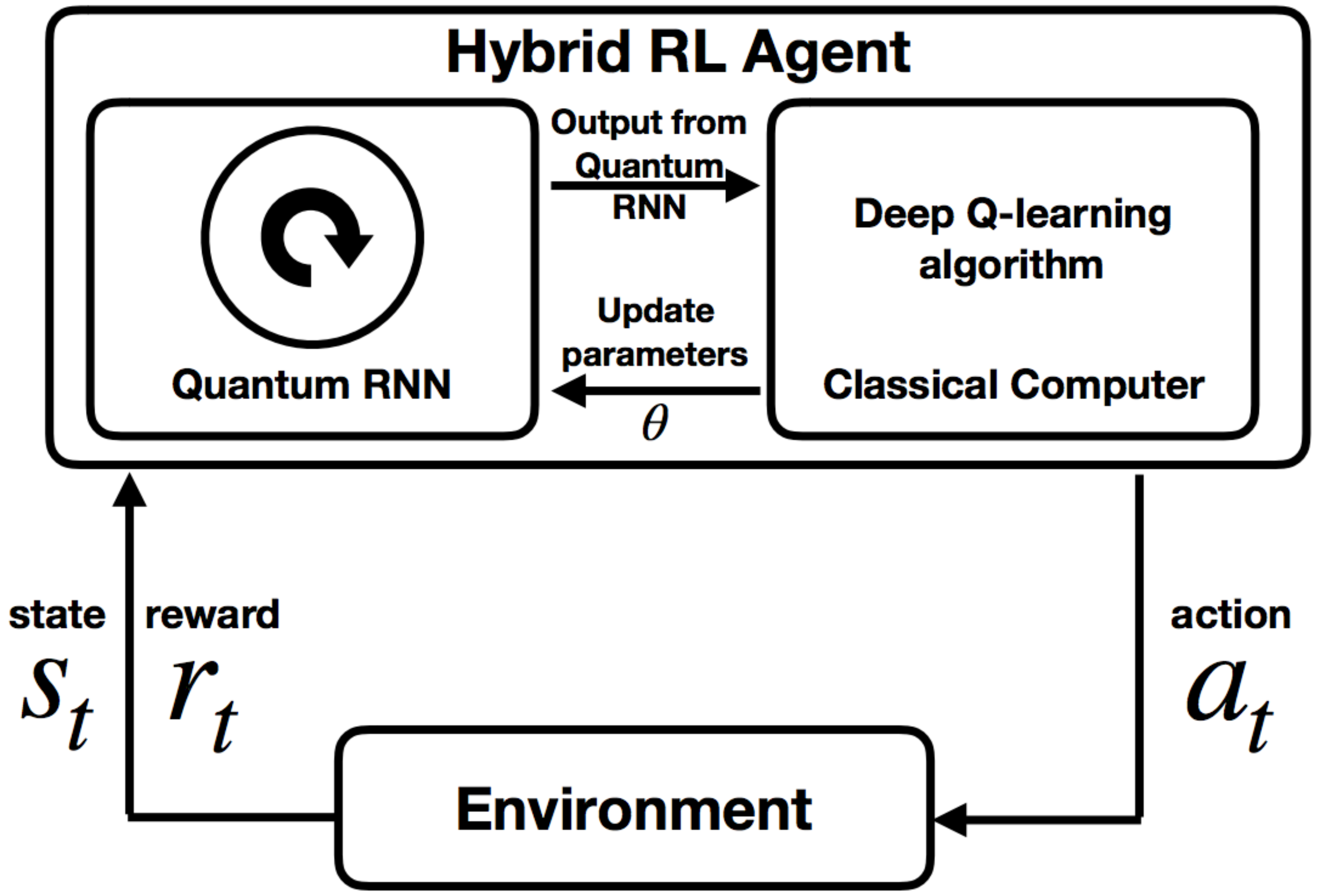
Fig. 7: **Results: Quantum A3C in the MiniGrid-SimpleCrossing environment.**

Chen, S. Y. C. (2023). **Asynchronous training of quantum reinforcement learning**. *Procedia Computer Science*, *222*, 321-330.

Quantum RL with QLSTM

- **Motivation**: Many real-world environments are only partially observable. The AI can only receive partial information of the world.
- **Challenges**: Existing QRL architectures do not have the capabilities to memorize previous time steps.
- **Approach**: Could quantum recurrent neural nets (QRNN) be helpful in QRL?



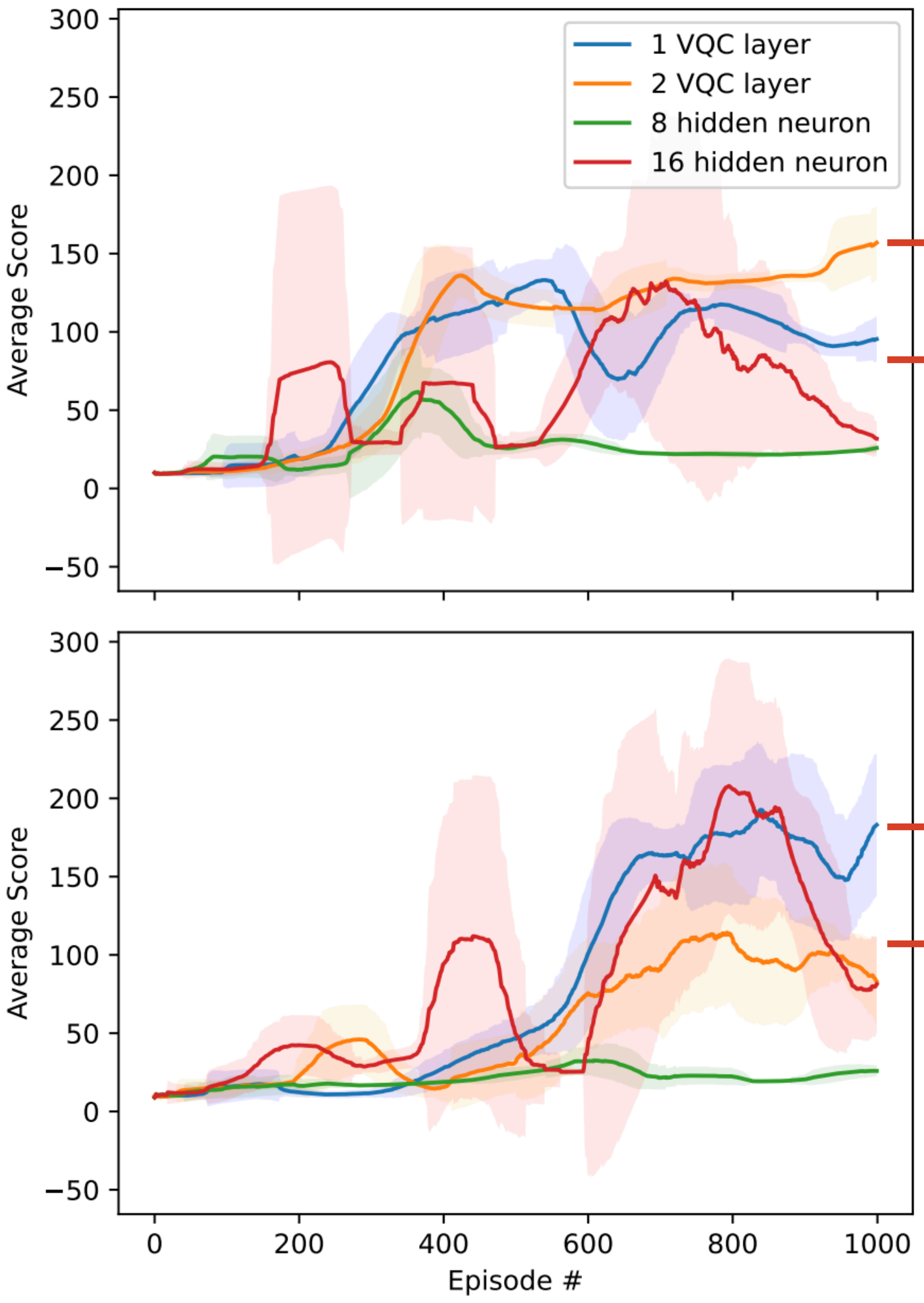


Chen, S. Y. C. (2023, June). **Quantum deep recurrent reinforcement learning.** In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 1-5). IEEE.

Quantum RL with QLSTM

Algorithm 1 Quantum deep recurrent Q-learning

Initialize replay memory $\mathcal{D}$ to capacity N
Initialize action-value function dressed QLSTM Q with random parameters θ
Initialize target dressed QLSTM Q with $\theta^{-} = \theta$
for episode $= 1, 2, \ldots, M$ **do**
  Initialize the episode record buffer $\mathcal{M}$
  Initialise state s_1 and encode into the quantum state
  Initialize h_1 and c_1 for the QLSTM
  for $t = 1, 2, \ldots, T$ **do**
    With probability ϵ select a random action a_t otherwise select $a_t = \max_a Q^*(s_t, a; \theta)$ from the output of the QLSTM
    Execute action a_t in emulator and observe reward r_t and next state s_{t+1}
    Store transition (s_t, a_t, r_t, s_{t+1}) in $\mathcal{M}$
    Sample random batch of trajectories $\mathcal{T}$ from $\mathcal{D}$
    Set $y_j = \begin{cases} r_j & \text{for terminal } s_{j+1} \\ r_j + \gamma \max_{a'} Q(s_{j+1}, a'; \theta) & \text{for non-terminal } s_{j+1} \end{cases}$
    Perform a gradient descent step on $(y_j - Q(s_j, a_j; \theta^{-}))^2$
    Update the target network θ^{-} every S steps.
  end for
  Store episode record $\mathcal{M}$ to $\mathcal{D}$
  Update ϵ
end for

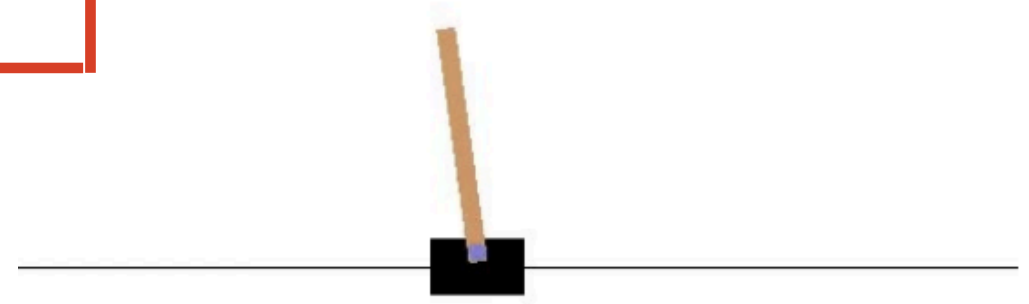


Quantum models use **smaller number of parameters**

	QLSTM-1	QLSTM-2	LSTM-8	LSTM-16
Full	150	270	634	2290
Partial	146	266	626	2274

Table 1. Number of parameters.

Quantum models show **higher** or **more stable** scores



Env: CartPole

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

QLSTM without training?

- **Motivation**: Time-series modeling is an important task in machine learning. Recurrent neural network (quantum or classical) is one of the framework to model time-series.
- **Challenges**: Quantum RNN (e.g. QLSTM) training are computationally expensive, requiring gradient calculation of deep quantum circuit models. (**Backpropagation-Through-Time (BPTT) is slow!**)

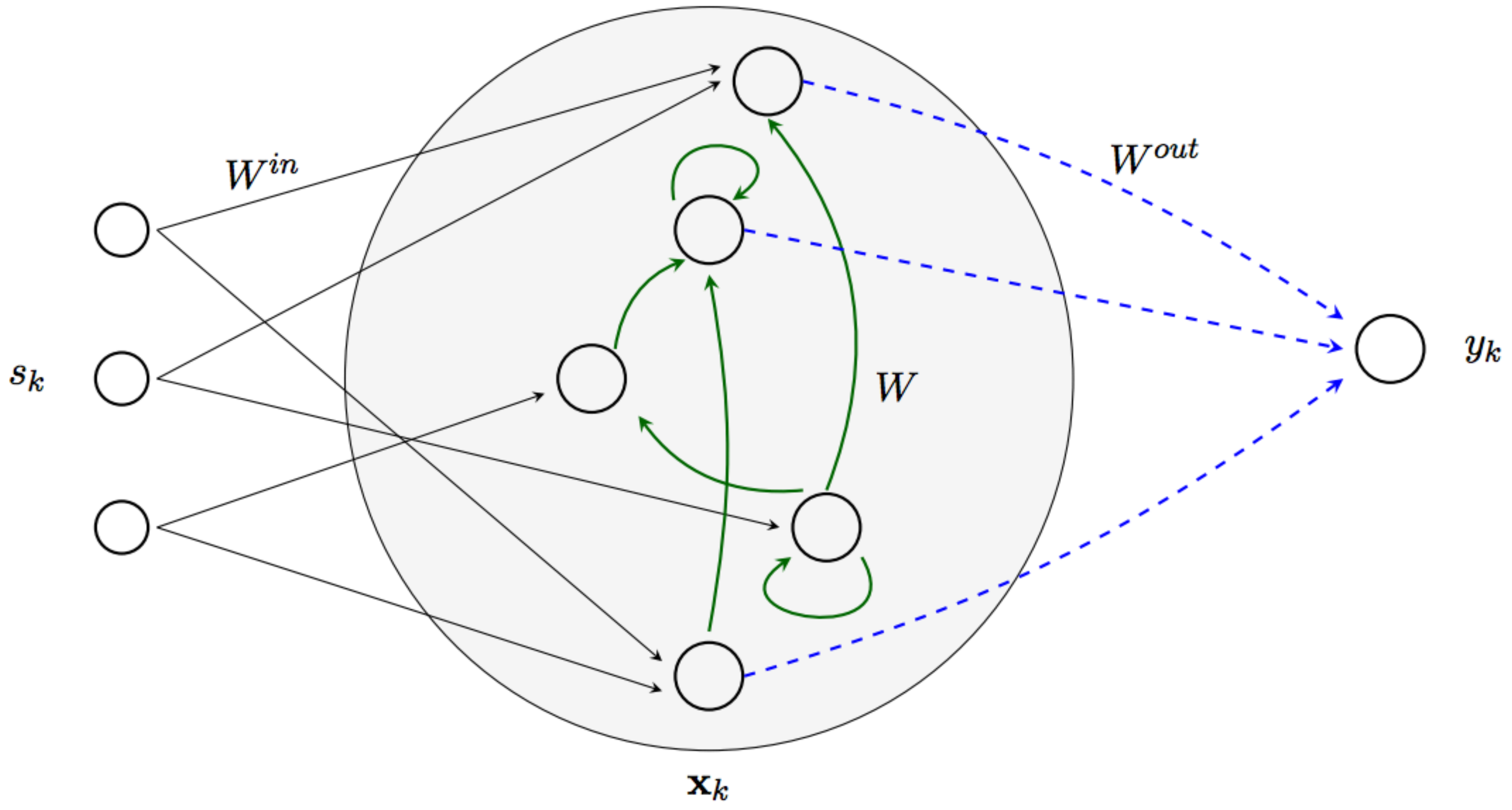
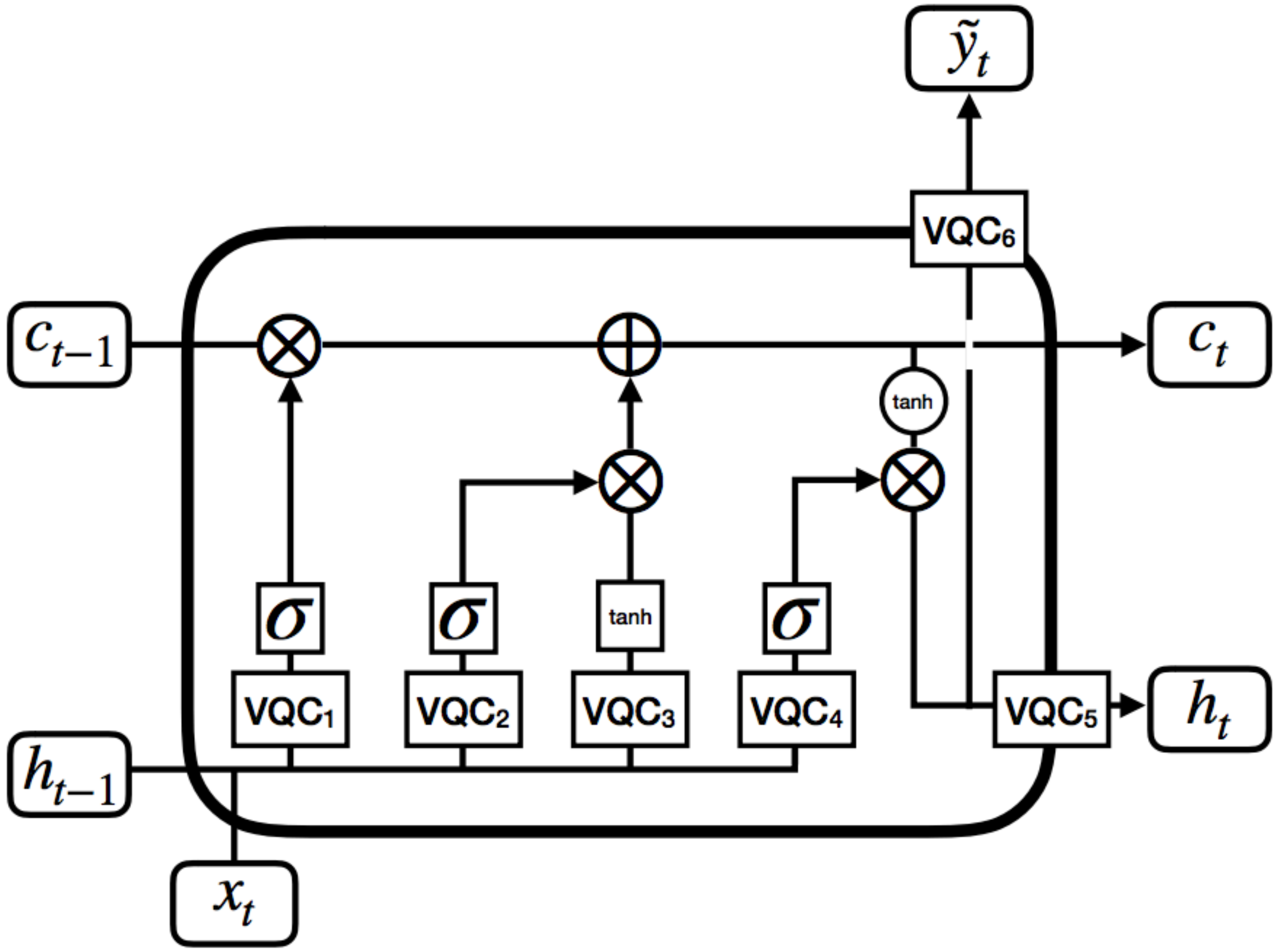


FIG. 1. **Reservoir computing (RC).**

Chen, S. Y. C. (2024, April). **Efficient quantum recurrent reinforcement learning via quantum reservoir computing**. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 13186-13190). IEEE.

- **Approach**: Adopt the classical idea of reservoir computing in the quantum regime: treating the quantum RNN as a reservoir. (The quantum parameters are randomly initialized and fixed. Only the final classical layers are trained.)

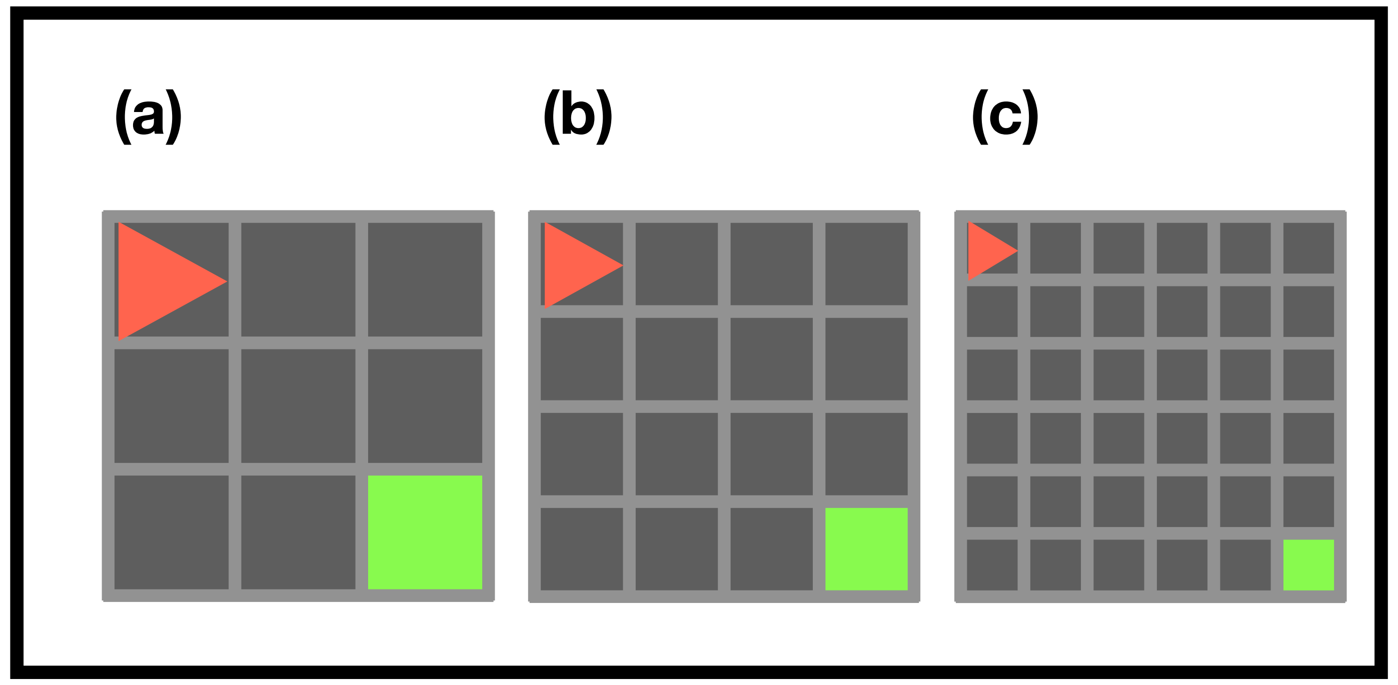
- **Results**: Previous works show that the QRNN within the reservoir computing framework can reach comparable performance to fully trained ones.



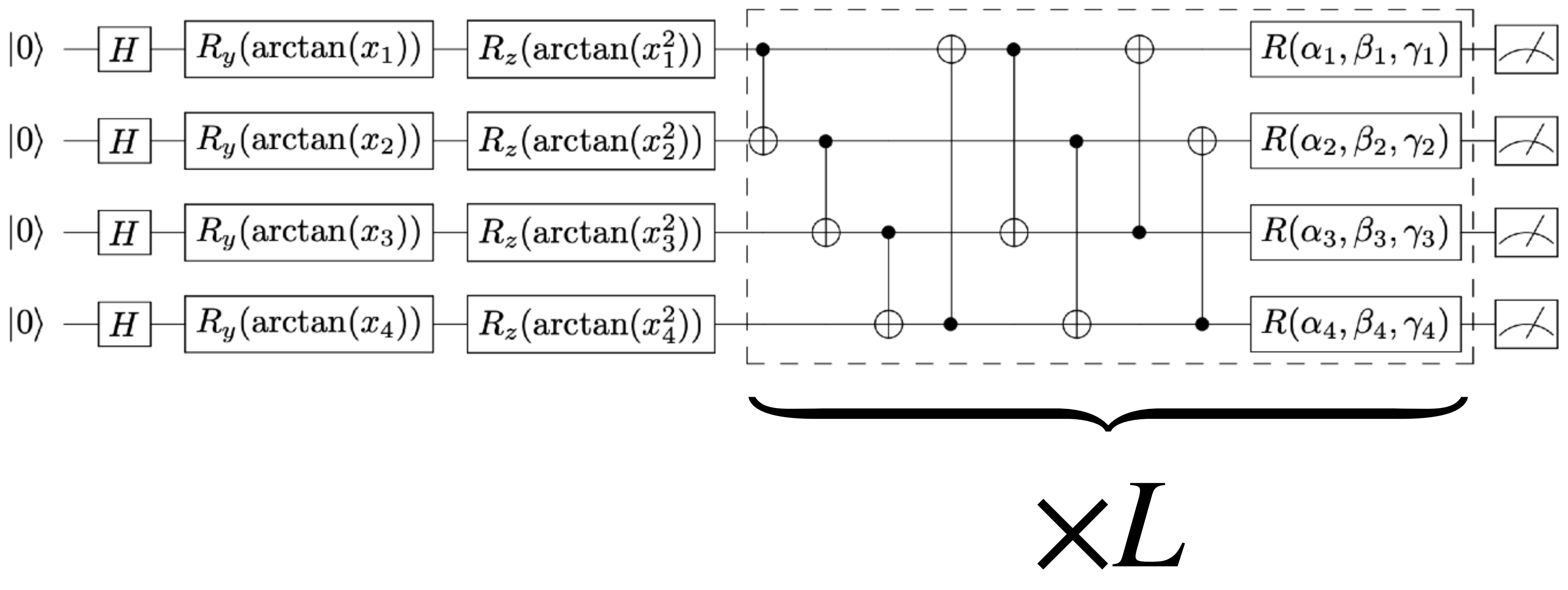
VQCs are **NOT** trained

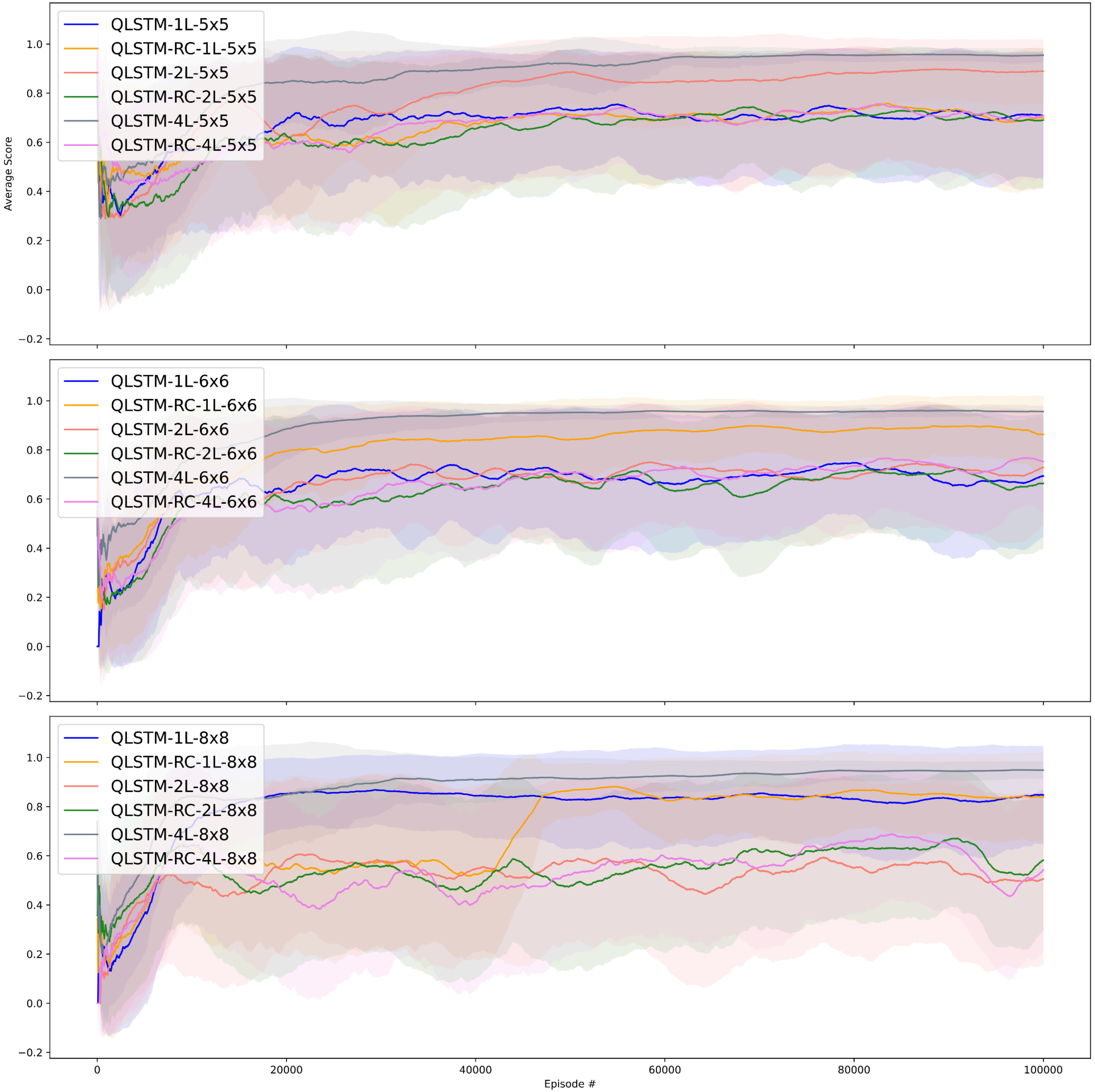
Chen, S. Y. C. (2024, April). **Efficient quantum recurrent reinforcement learning via quantum reservoir computing**. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 13186-13190). IEEE.

Environment:



VQC in QLSTM:

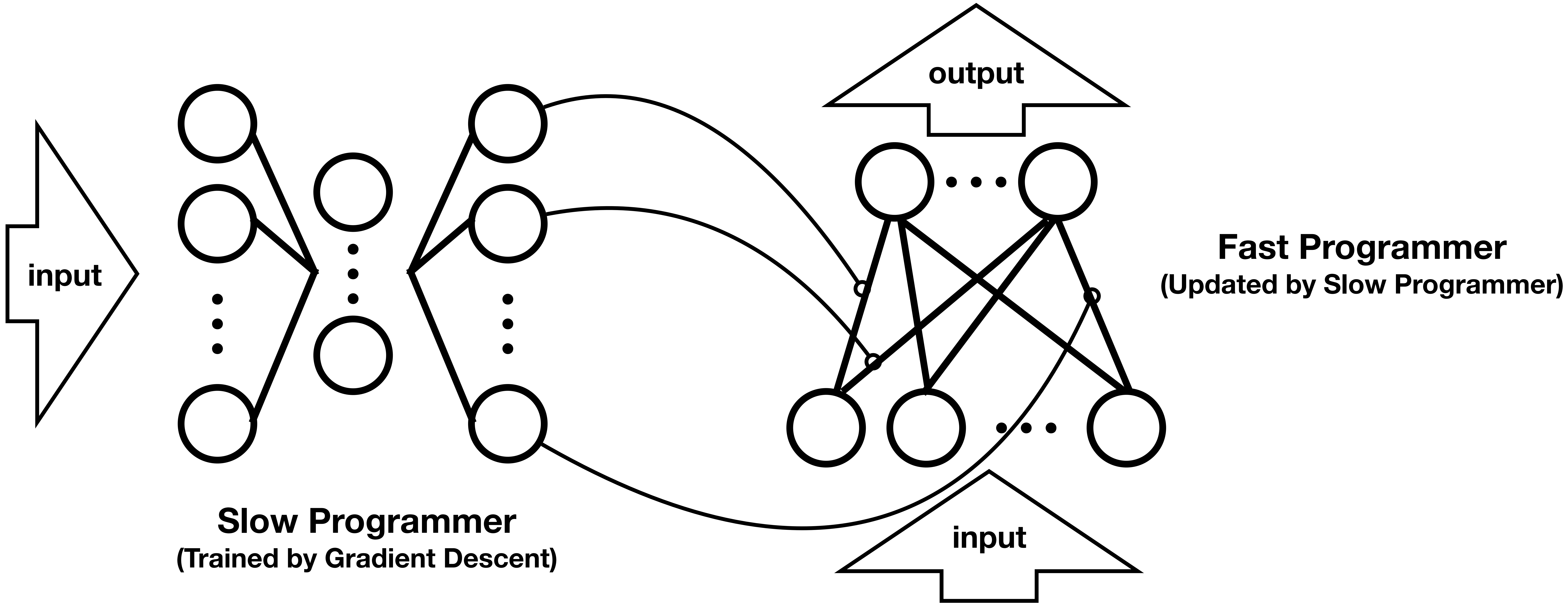




Chen, S. Y. C. (2024, April). **Efficient quantum recurrent reinforcement learning via quantum reservoir computing**. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 13186-13190). IEEE.

Don’t want ANY quantum RNN?

Classical FWP



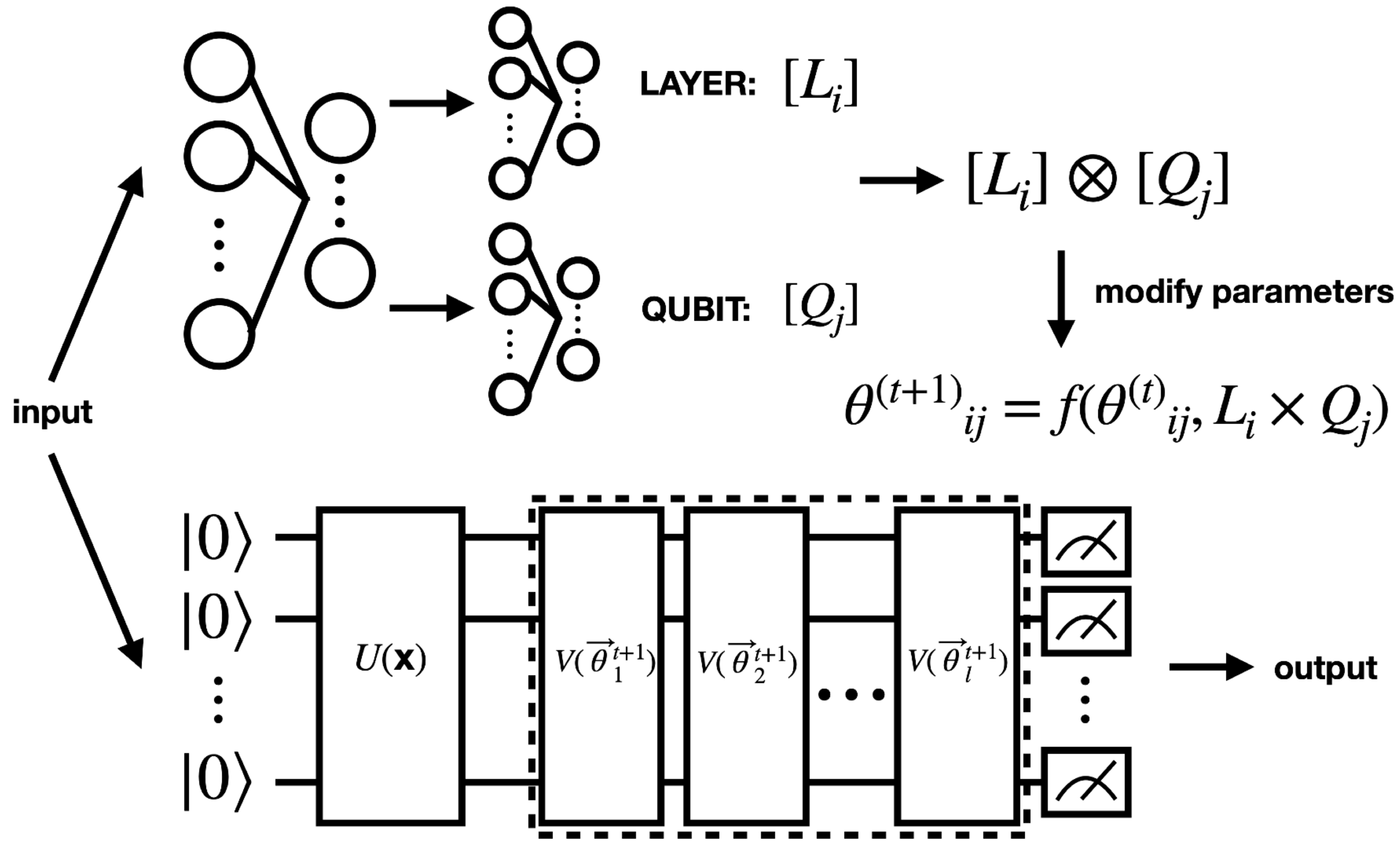
Schmidhuber, J. (1992). Learning to control fast-weight memories: An alternative to dynamic recurrent networks. *Neural Computation*, *4*(1), 131-139.

Learning to Program a VQC

- Classical NN generates circuit parameter updates for each "**layer**" and "**qubit**".

- Use tensor product to generate parameter <u>*updates*</u> for all parameterized gates.

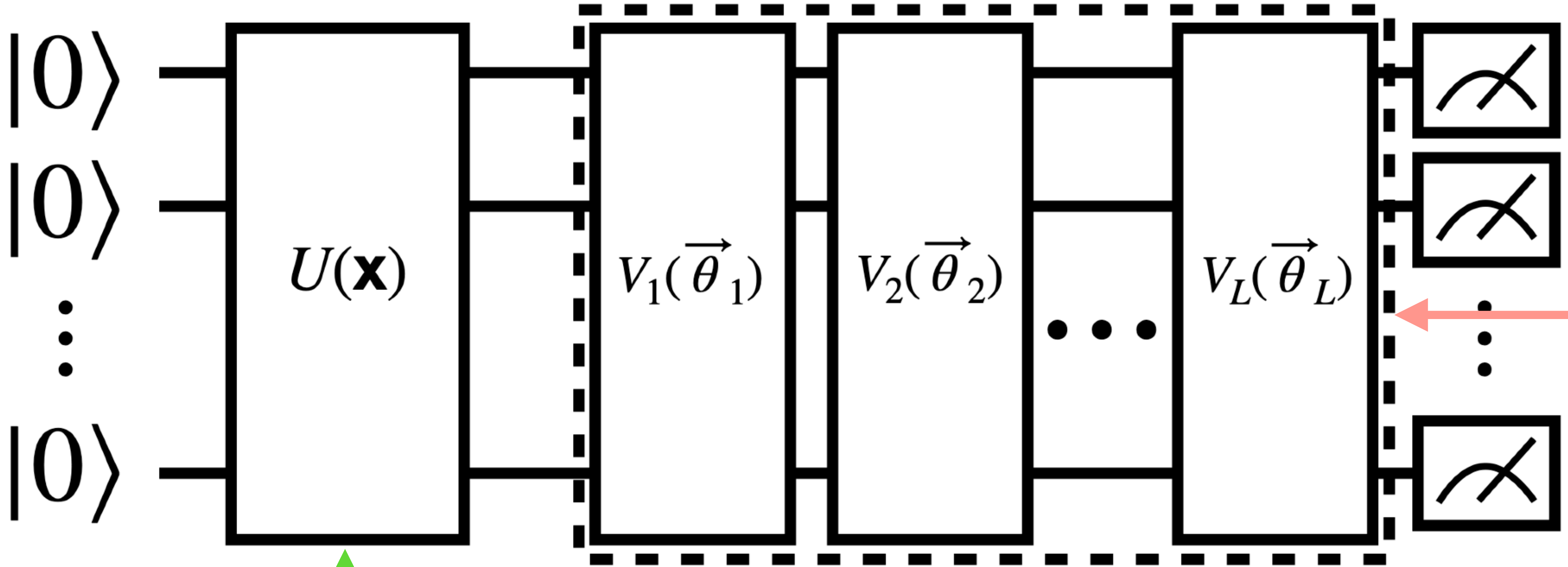
$$
\begin{aligned}
[L_i] \otimes [Q_j] &= [M_{ij}] \\
&= [L_i \times Q_j] \\
&= \begin{bmatrix} L_1 \times Q_1 & L_1 \times Q_2 & \cdots & L_1 \times Q_n \\ L_2 \times Q_1 & L_2 \times Q_2 & \cdots & L_2 \times Q_n \\ \vdots & \ddots & & \vdots \\ L_l \times Q_1 & L_l \times Q_2 & \cdots & L_l \times Q_n \end{bmatrix}
\end{aligned}
$$



Codes:

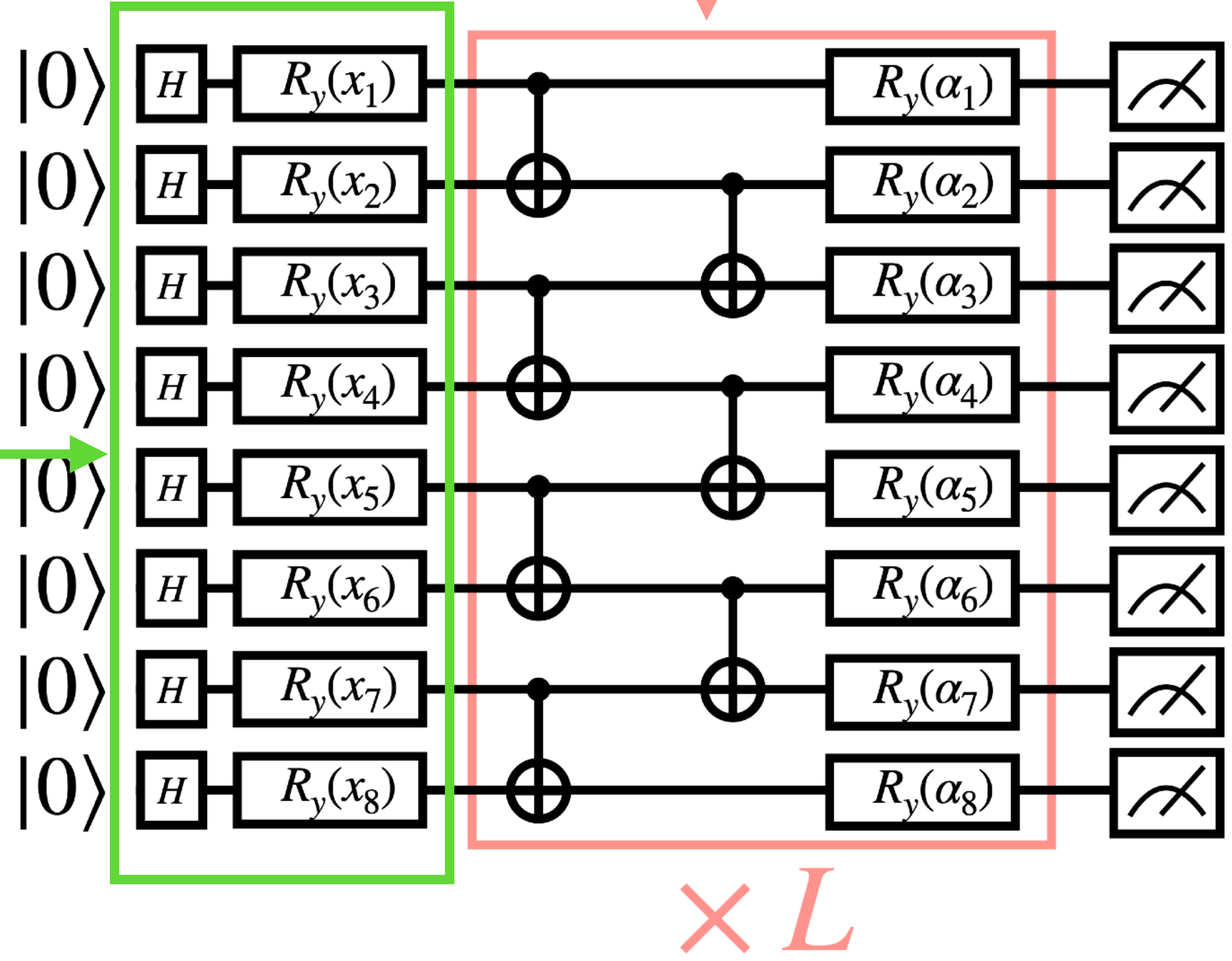
Paper:

Learning to Program a VQC



Quantum Circuits Parameters updated by classical NN

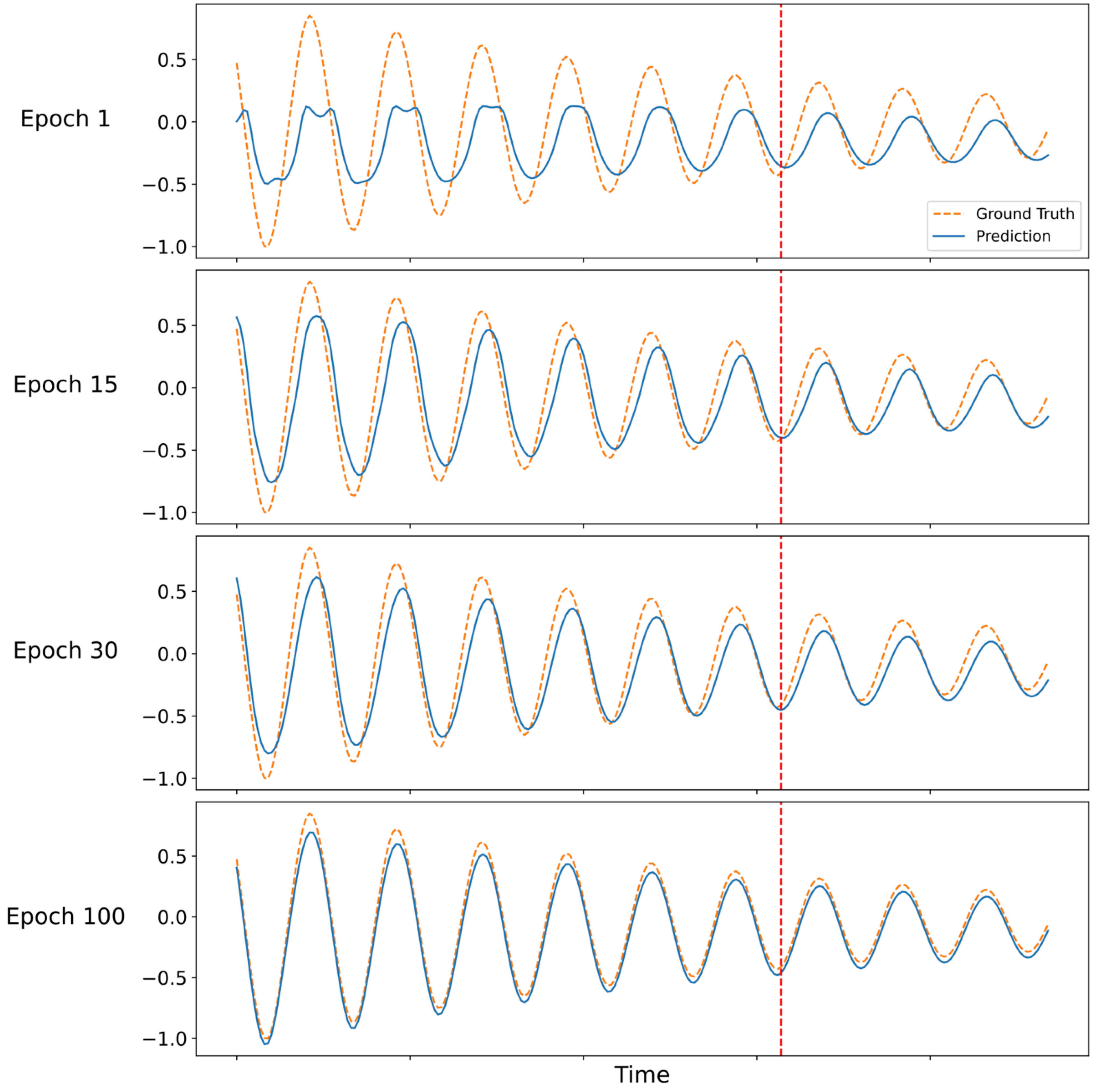
Data encoding circuit to transform classical data into a quantum state



arXiv:2402.17760, IJCNN 2024

RESULTS: TIME-SERIES MODELING - DAMPED SHM

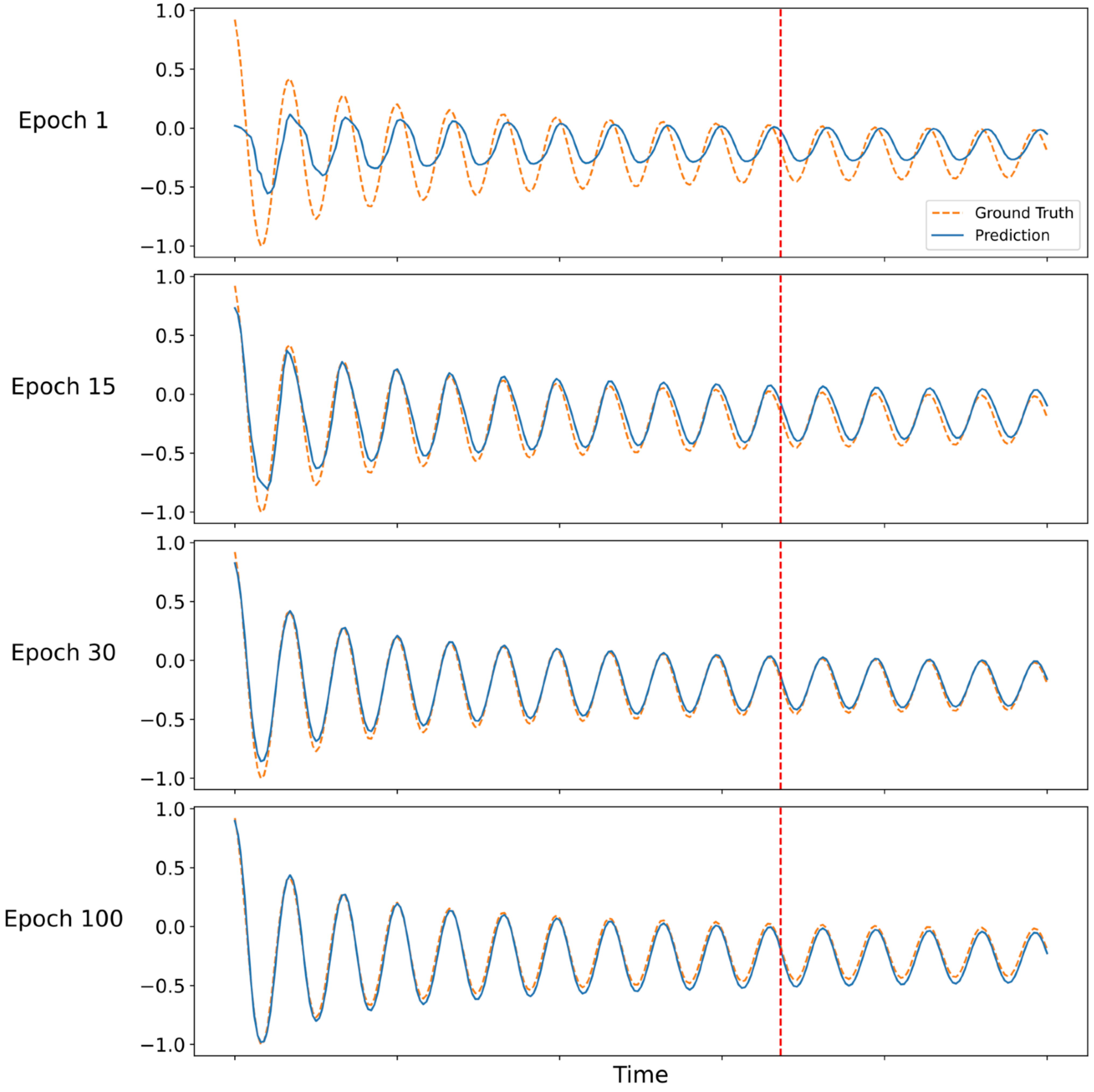
	QLSTM [30]	QFWP
Epoch 1	$1.66 \times 10^{-1}/1.35 \times 10^{-2}$	$3.33 \times 10^{-1}/3.26 \times 10^{-2}$
Epoch 15	$2.89 \times 10^{-2}/5.53 \times 10^{-3}$	$7.21 \times 10^{-2}/1.65 \times 10^{-2}$
Epoch 30	$9.06 \times 10^{-3}/3.41 \times 10^{-4}$	$5.96 \times 10^{-2}/1.34 \times 10^{-2}$
Epoch 100	$2.86 \times 10^{-3}/1.94 \times 10^{-4}$	$1.09 \times 10^{-2}/2.70 \times 10^{-3}$



Quantum FWP for damped SHM

RESULTS: TIME-SERIES MODELING - BESSEL FUNCTION J_2

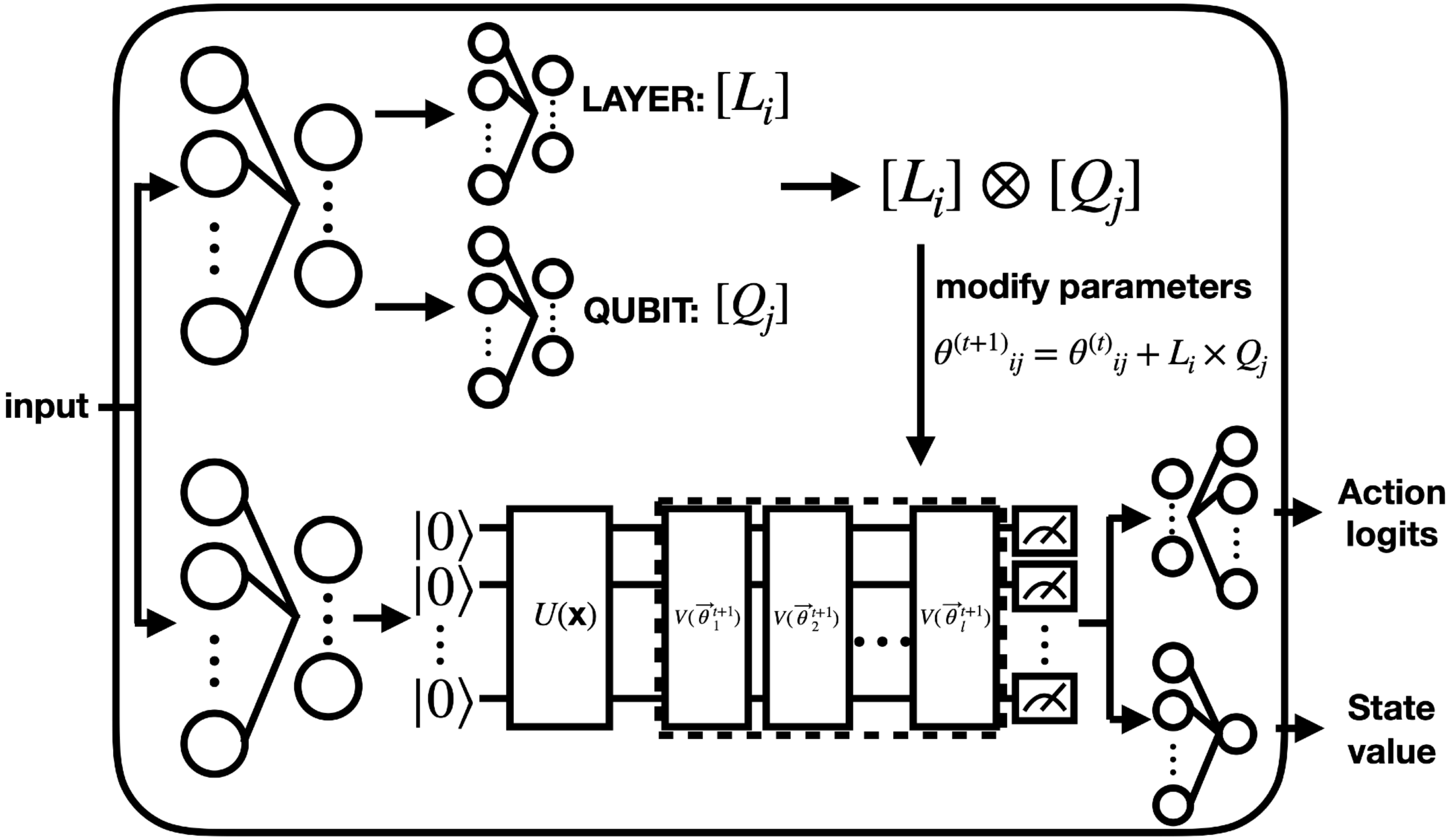
	QLSTM [30]	QFWP
Epoch 1	$1.04 \times 10^{-1}/1.66 \times 10^{-2}$	$1.17 \times 10^{-1}/1.58 \times 10^{-2}$
Epoch 15	$2.30 \times 10^{-2}/5.35 \times 10^{-3}$	$1.22 \times 10^{-2}/4.56 \times 10^{-3}$
Epoch 30	$1.27 \times 10^{-2}/2.42 \times 10^{-3}$	$5.52 \times 10^{-3}/7.80 \times 10^{-4}$
Epoch 100	$6.97 \times 10^{-4}/1.21 \times 10^{-5}$	$8.57 \times 10^{-4}/2.30 \times 10^{-3}$



Quantum FWP for Bessel function

Learning to Program a VQC for RL

- Slow programmer:
  - Encoder
  - NN for quantum layers L_i
  - NN for qubit index Q_j
- Fast programmer:
  - 8-qubit VQC
  - $L = 2$ or $L = 4$ VQC layers



arXiv:2402.17760, IJCNN 2024

Learning to Program a VQC for RL

- QLSTM baseline
- 8-qubit VQC
  - 4 qubits for input
  - 4 qubits for hidden dimension
- Classical NN for dimensional reduction, actor and critic outputs.

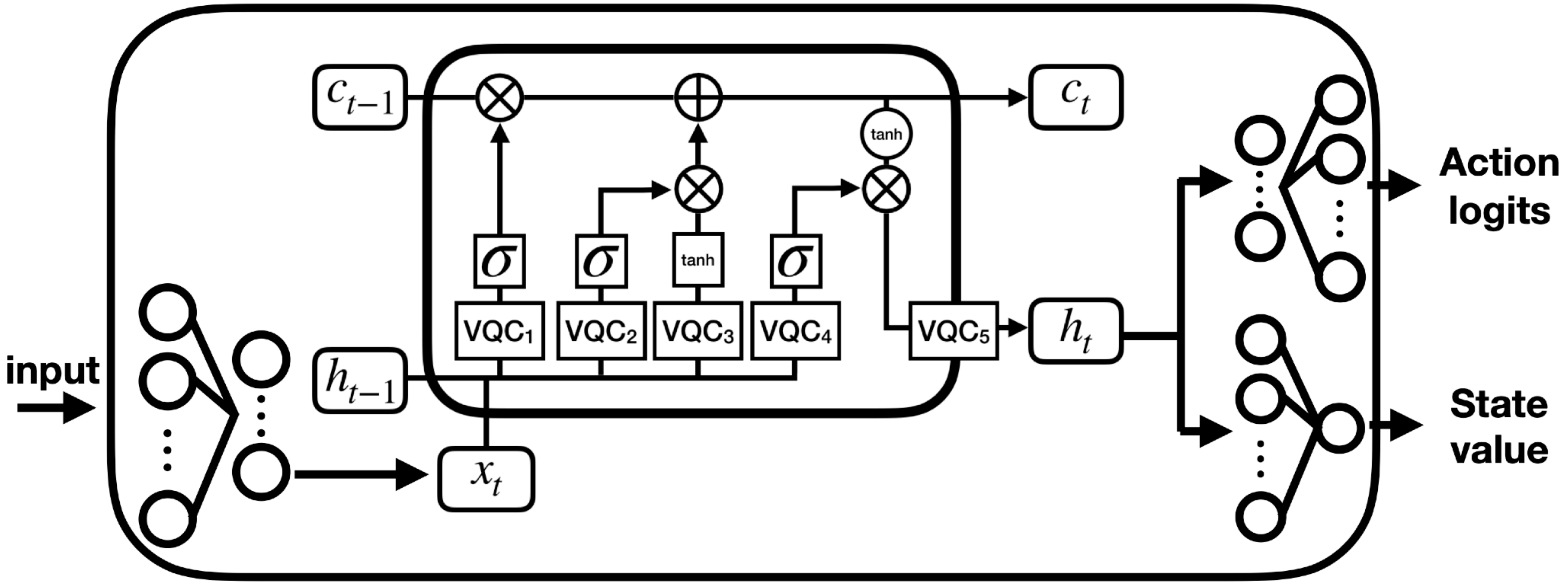


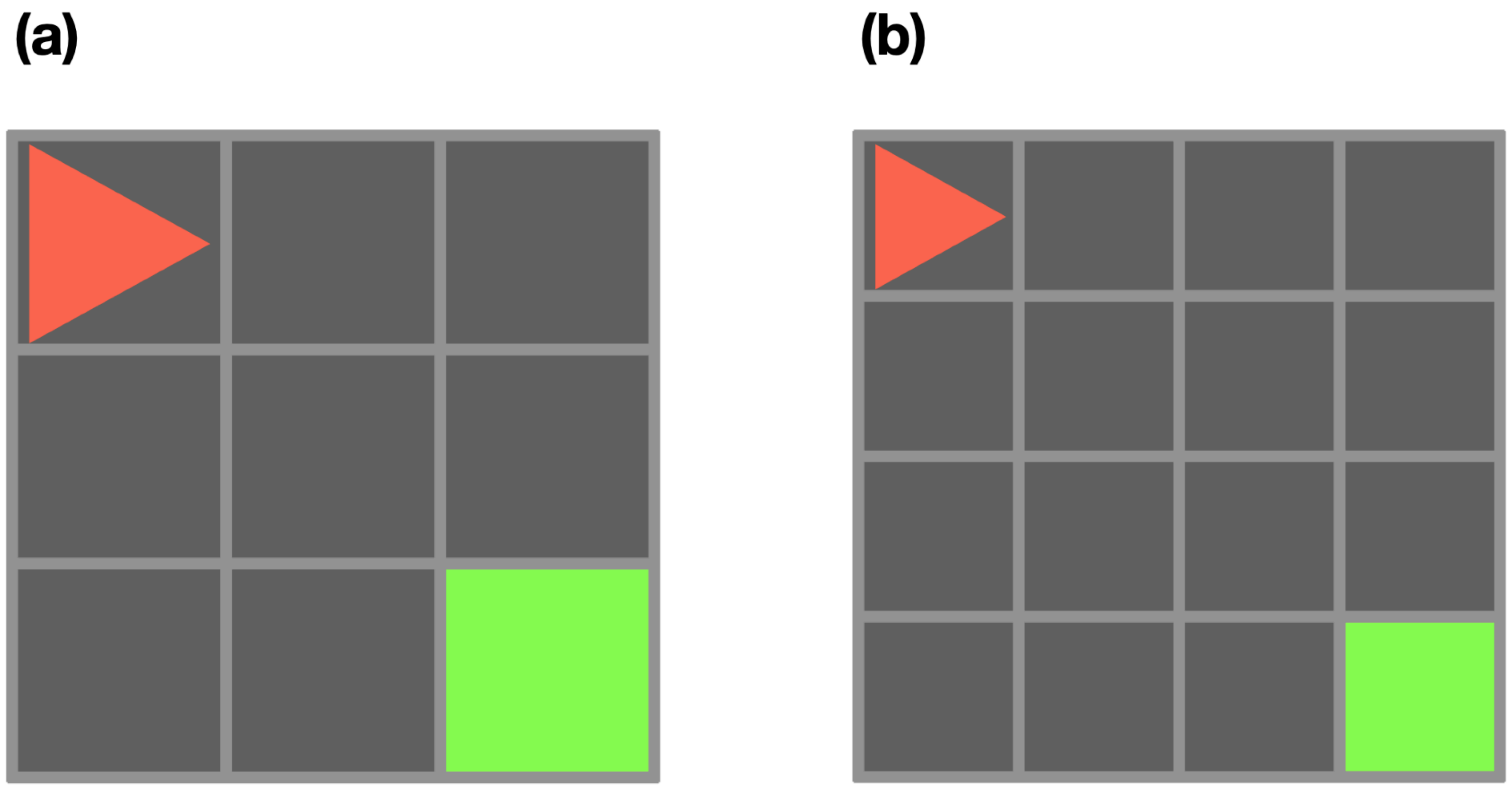
TABLE VI
NUMBER OF PARAMETERS IN QFWP AND QLSTM MODELS IN QRL EXPERIMENTS.

	Classical	Quantum
QLSTM-2 VQC Layer	627	240
QLSTM-4 VQC Layer	627	480
QLSTM-6 VQC Layer	627	720
QLSTM-8 VQC Layer	627	960
QLSTM-10 VQC Layer	627	1200
Quantum FWP-2 VQC Layer	2521	16
Quantum FWP-4 VQC Layer	2539	32

arXiv:2402.17760, IJCNN 2024

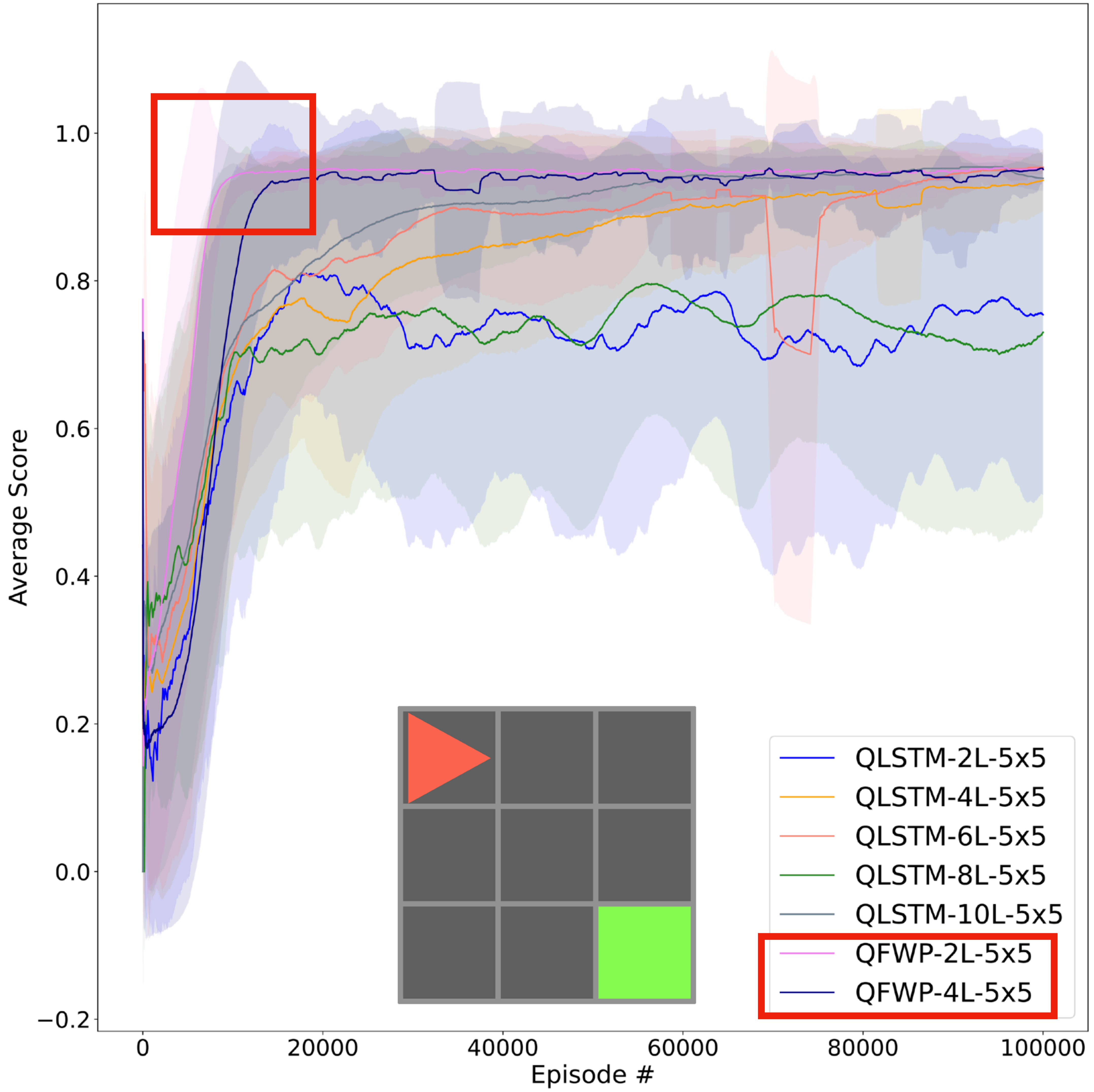
Learning to Program a VQC for RL

- **Observation:** 147-dimensional vector.
- **Action:** There are six actions: *turn left*, *turn right*, *move forward*, *pick up an object*, *drop the object being carried* and *toggle*. Only the first three of them are having actual effects in this case. The agent is expected to learn this fact.
- **Reward:** The agent receives a reward of 1 upon reaching the goal. A penalty is subtracted from this reward based on the formula

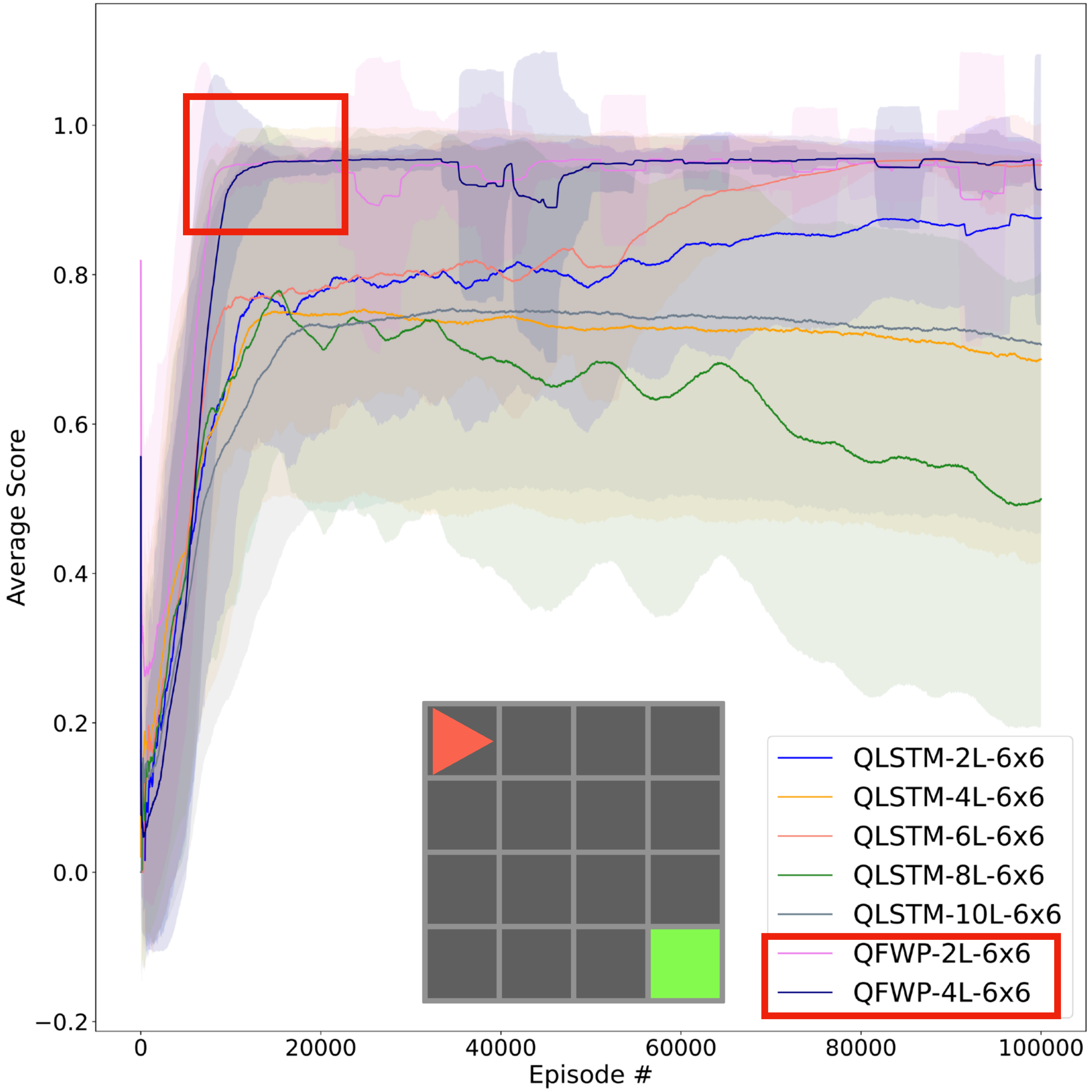


arXiv:2402.17760, IJCNN 2024

Learning to Program a VQC for RL



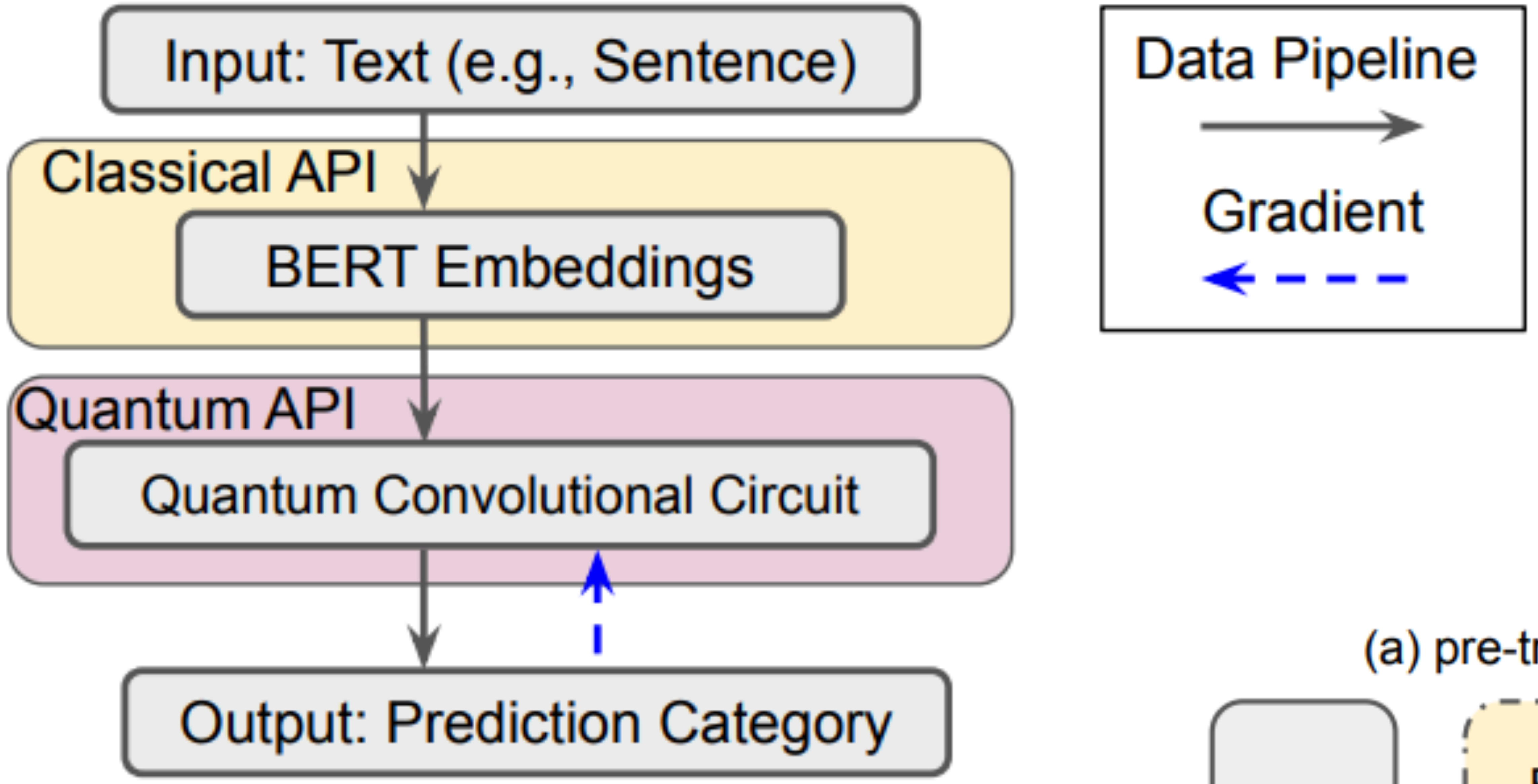
MiniGrid-Empty-5x5

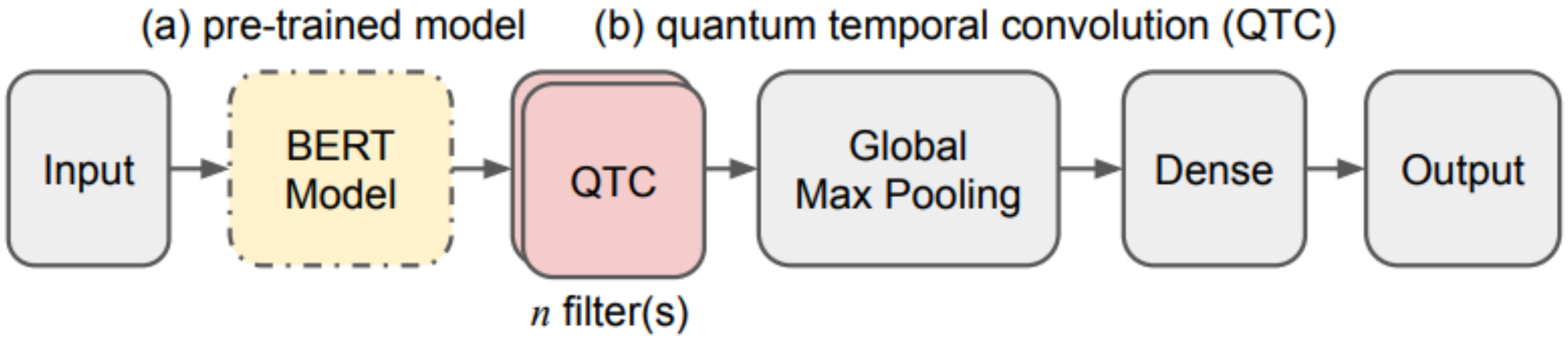


MiniGrid-Empty-6x6

- **Applications**
  - Quantum Classification
  - Privacy-Preserving Quantum Machine Learning (Federated Learning, Differential Privacy)
  - Quantum Recurrent Neural Network
  - Quantum Reinforcement Learning
  - **Quantum Natural Language Processing**
  - Quantum Neural Networks for Model Compression

BERT with Quantum Temporal Convolution Learning





Yang, C. H. H., Qi, J., Chen, S. Y. C., Tsao, Y., & Chen, P. Y. (2022). **When BERT Meets Quantum Temporal Convolution Learning for Text Classification in Heterogeneous Computing**. *arXiv preprint arXiv:2203.03550*. *ICASSP 2022*

BERT with Quantum Temporal Cone Learning

Table 3: Average accuracy on intent classification for Snips with a set of different number (n) of convolutional filter and kernel size (k).

Embedding	word2vec				BERT			
(n,k)	(1,4)	(2,2)	(2,3)	(2,4)	(1,4)	(2,2)	(2,3)	(2,4)
TCN	82.02	83.37	82.90	83.15	95.48	95.23	95.12	95.27
QTC	**83.32**	**83.94**	**83.61**	**84.64**	**96.41**	**96.42**	**96.62**	**96.44**

Table 4: Average accuracy on intent classification for $ATIS_7$ with a set of different number (n) of convolutional filter and kernel size (k).

Embedding	word2vec				BERT			
(n,k)	(1,4)	(2,2)	(2,3)	(2,4)	(1,4)	(2,2)	(2,3)	(2,4)
TCN	80.09	80.22	80.91	82.34	95.18	95.03	94.95	95.23
QTC	**81.42**	**82.49**	**83.82**	**83.95**	**96.69**	**96.92**	**96.32**	**96.98**

Yang, C. H. H., Qi, J., Chen, S. Y. C., Tsao, Y., & Chen, P. Y. (2022). **When BERT Meets Quantum Temporal Convolution Learning for Text Classification in Heterogeneous Computing**. *arXiv preprint arXiv:2203.03550*. *ICASSP 2022*

Quantum Language Models

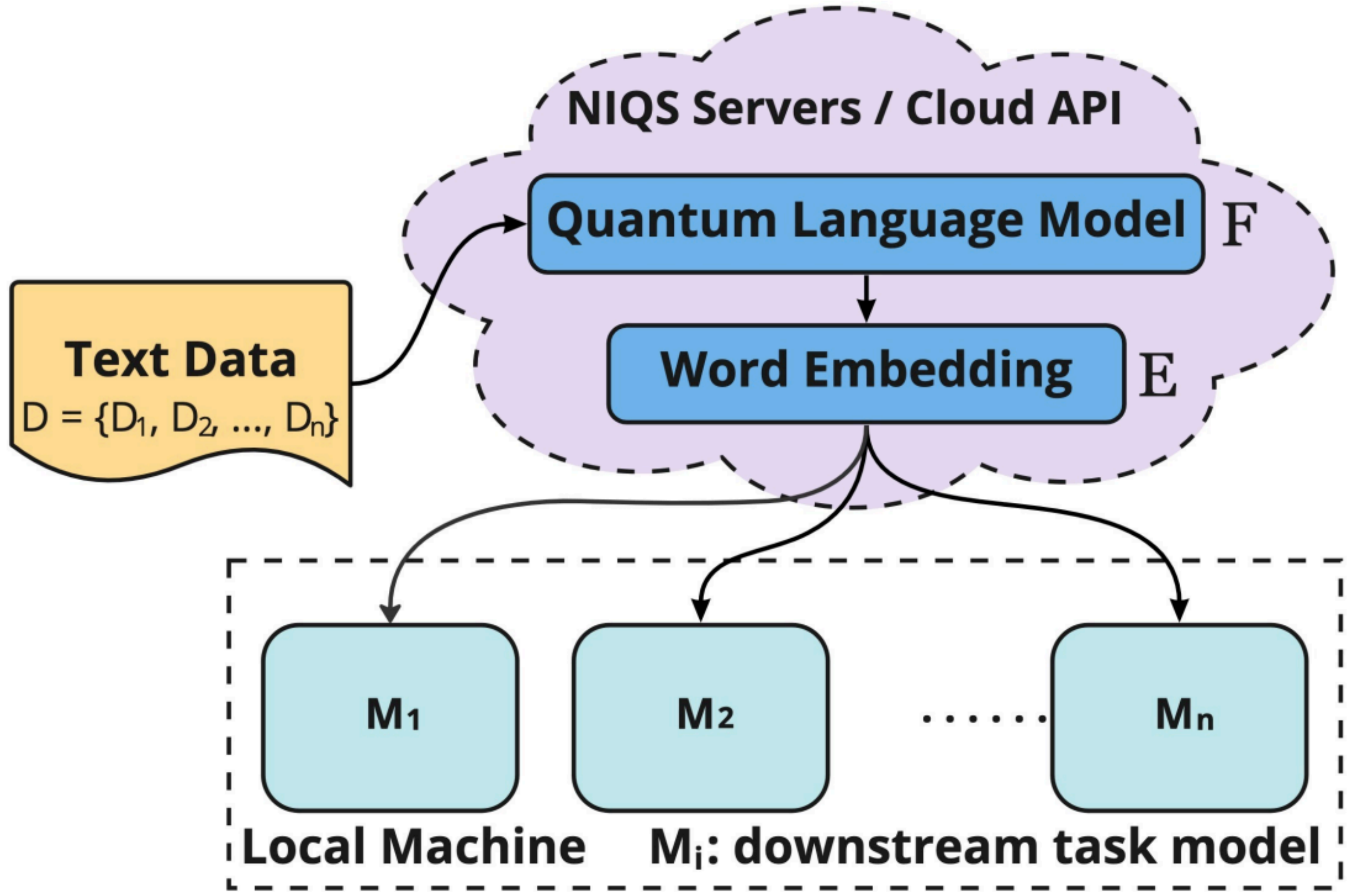
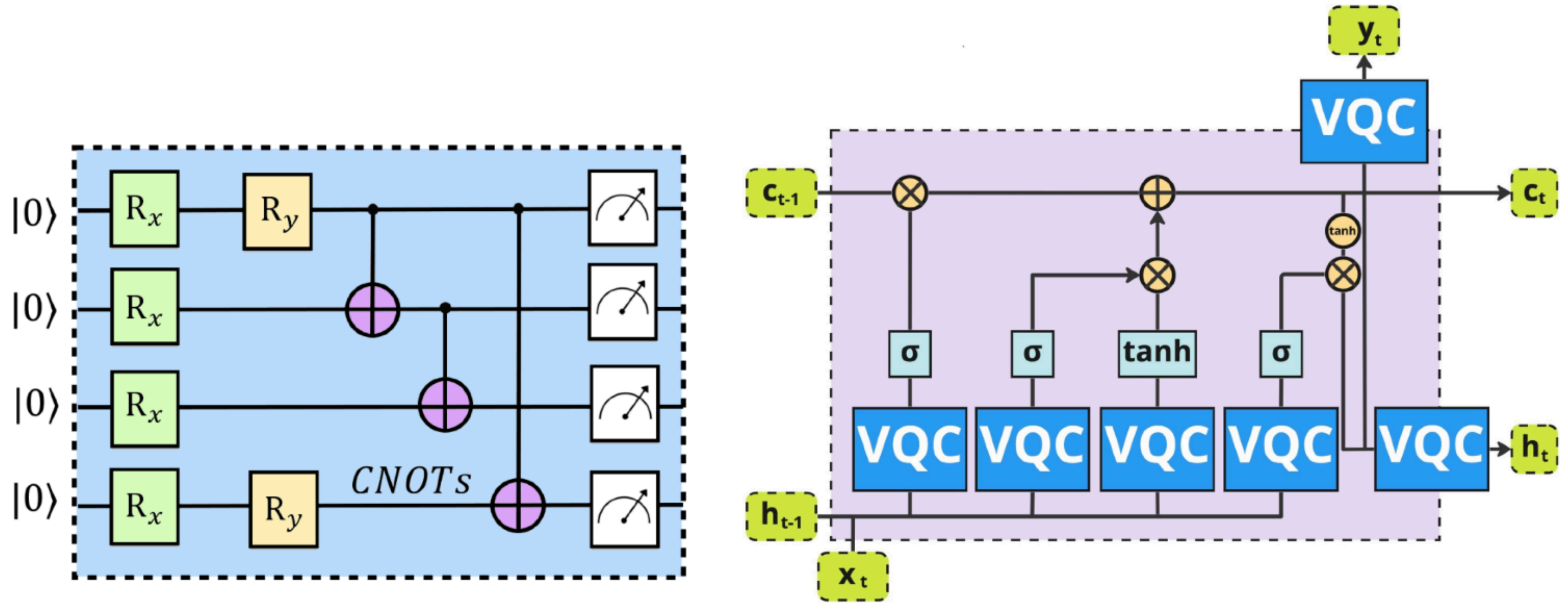


Fig. 1: Decentralized Quantum Language Model Pipeline. Text data is trained on language model on NISQ servers, the word embeddings are transferred to downstream models $\mathcal{M}_i$



(a) Quantum Circuit [28] (b) Q-LSTM

Fig. 2: Model Architecture

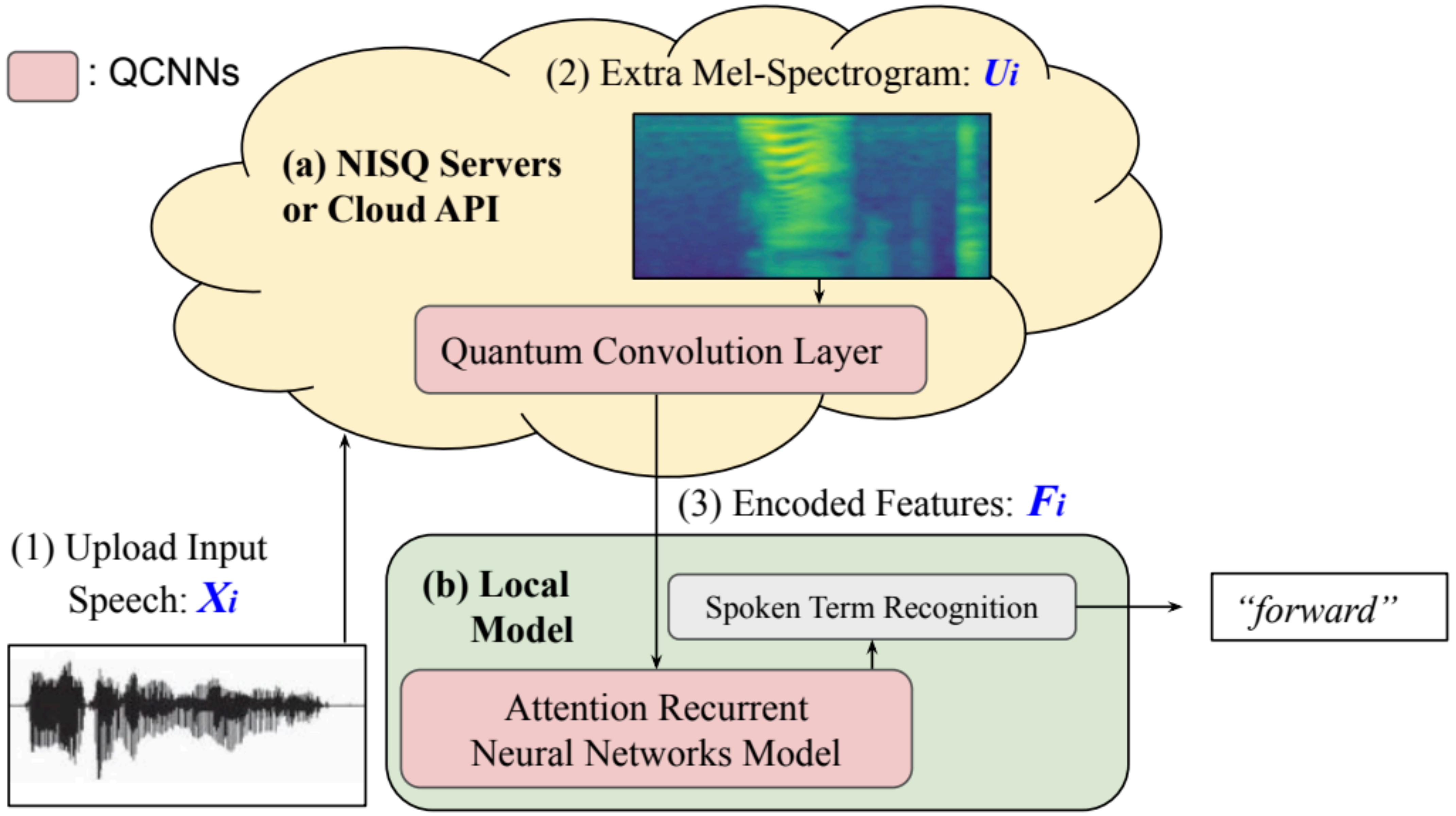
PLM	LSTM	Q-LSTM (4q)
accuracy	0.928	0.934
weighted f1	0.93	0.93

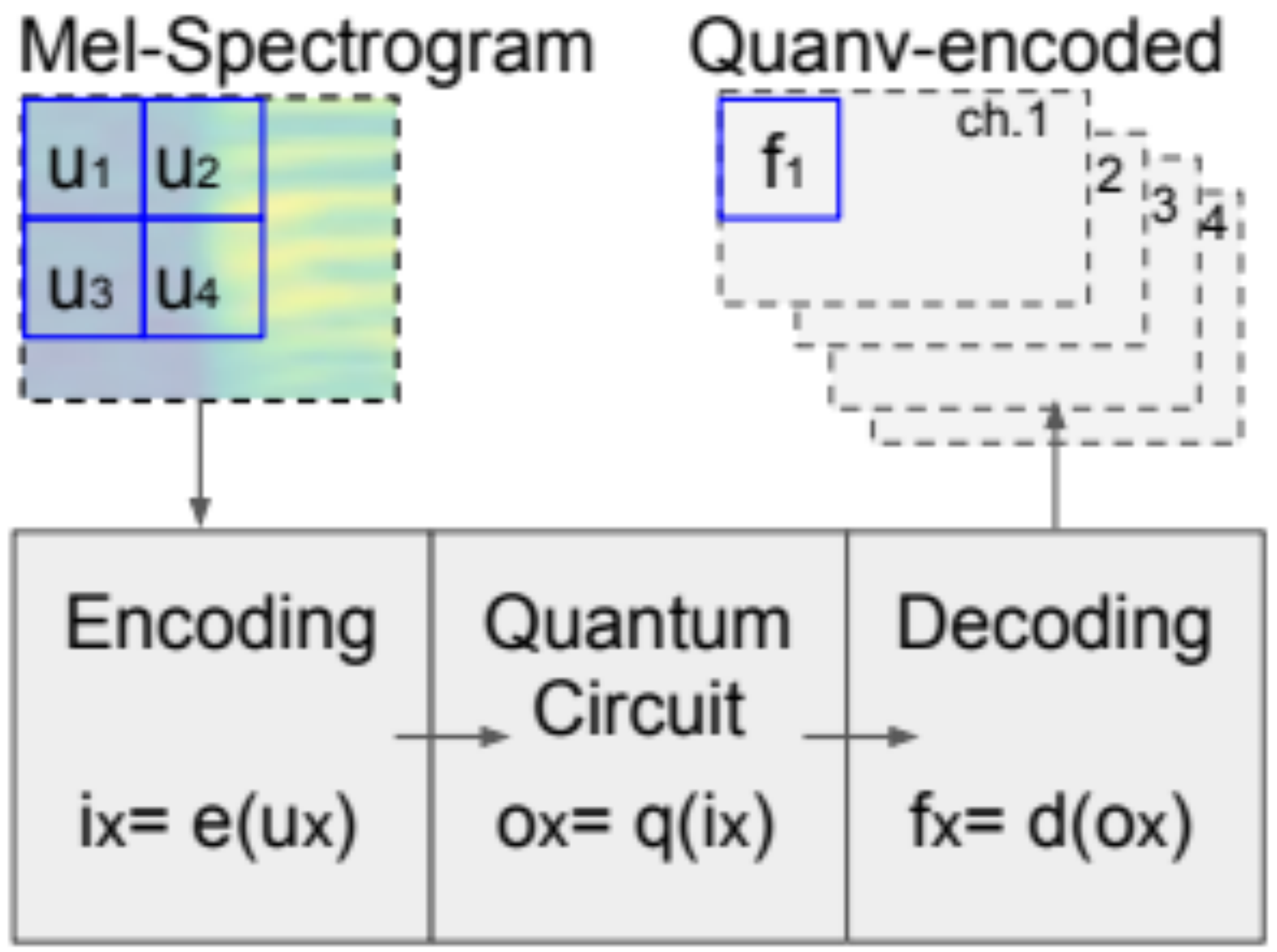
Table 2: SA Performance on Multilingual Twitter Dataset

Li, S. S., Zhang, X., Zhou, S., Shu, H., Liang, R., Liu, H., & Garcia, L. P. (2023, June). PQLM-Multilingual Decentralized Portable Quantum Language Model. In *ICASSP 2023*

Quantum Speech Recognition

- Vertical federated learning
- Speech input are first processed into Mel spectrogram and then sent into a quantum layer for encoding (on the cloud).
- The encoded features are used to train the acoustic model (on user devices).
- Can reduce model parameter leakage.





Yang, C. H. H., Qi, J., Chen, S. Y. C., Chen, P. Y., Siniscalchi, S. M., Ma, X., & Lee, C. H. (2021, June). **Decentralizing feature extraction with quantum convolutional neural network for automatic speech recognition**. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 6523-6527). IEEE.

- **Applications**
  - Quantum Classification
  - Privacy-Preserving Quantum Machine Learning (Federated Learning, Differential Privacy)
  - Quantum Recurrent Neural Network
  - Quantum Reinforcement Learning
  - Quantum Natural Language Processing
  - **Quantum Neural Networks for Model Compression**

- Challenges of training a QNN:
  - Challenges of data encoding
  - Quantum hardware requirement during inference

- Is there a way of leveraging the best part from both the quantum and classical NN?

Hilbert space is a BIG place!

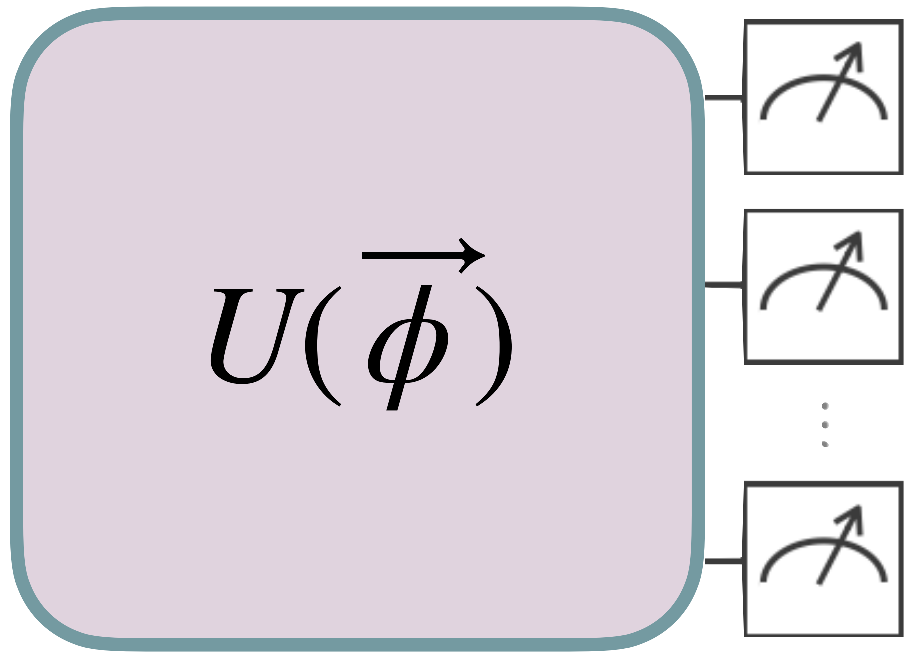
- Instead of preparing M initial parameters, we attempt to generate these M parameters using a QNN $U(\overrightarrow{\phi})$ with $N = \lceil \log_2 M \rceil$ qubits.

- The size of the Hilbert space is $2^N = 2^{\lceil \log_2 M \rceil} \geq M$ such that each probability $|\langle i | U(\overrightarrow{\phi}) \rangle|^2$ of a computational basis $|i\rangle$ could correspond to one of the parameters in $\vec{\theta}$.

- Assuming the QNN has a polynomial depth of layers, the number of parameters is $polylog(M)$.

N-qubit quantum neural network with parameters $\overrightarrow{\phi}$

Required qubits : $N = \lceil \log_2 M \rceil$



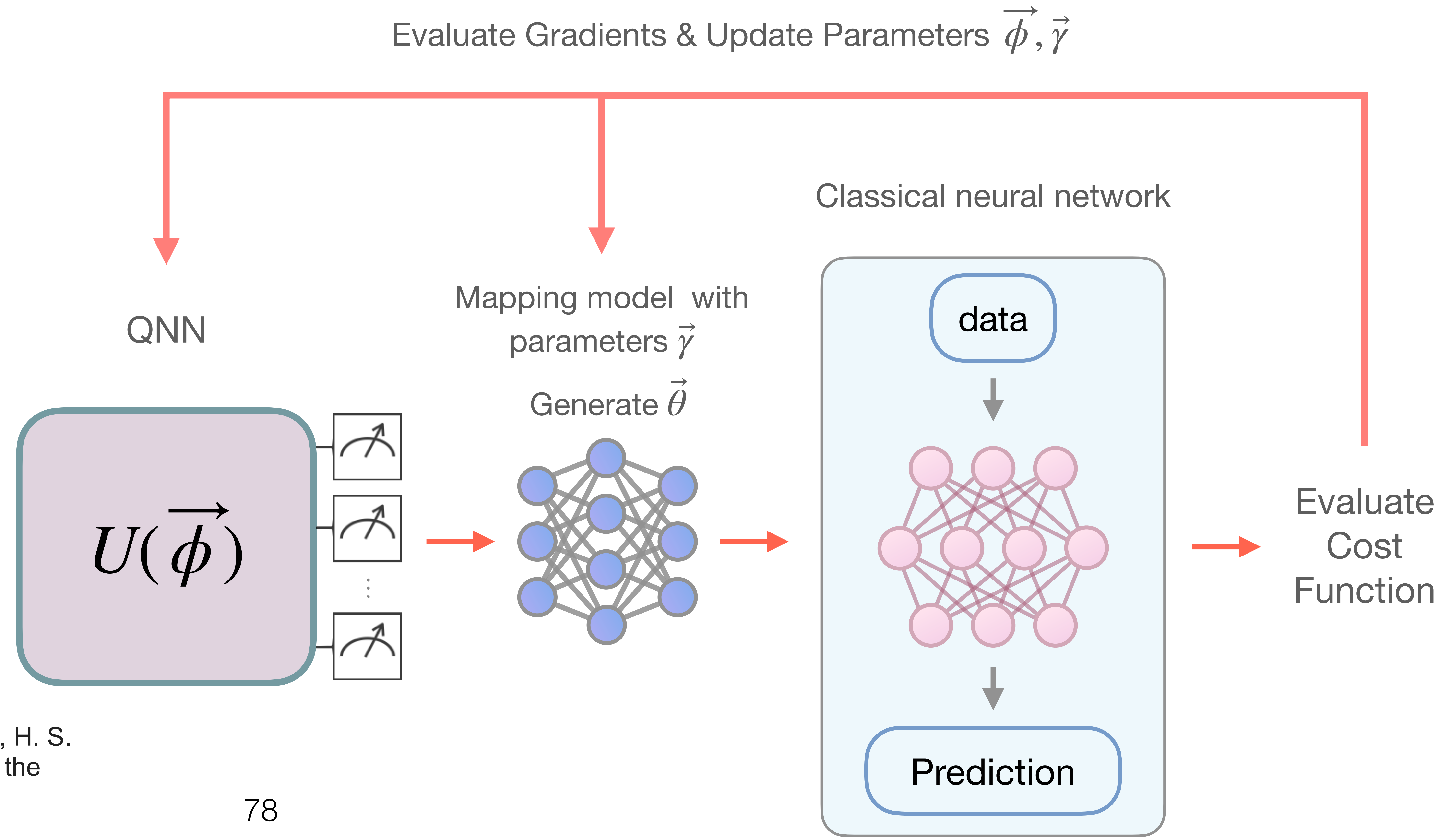
# of parameters: $polylog(M)$

$$(|i\rangle, |\langle i | U(\overrightarrow{\phi}) \rangle|^2) \xrightarrow{?} \theta_i$$

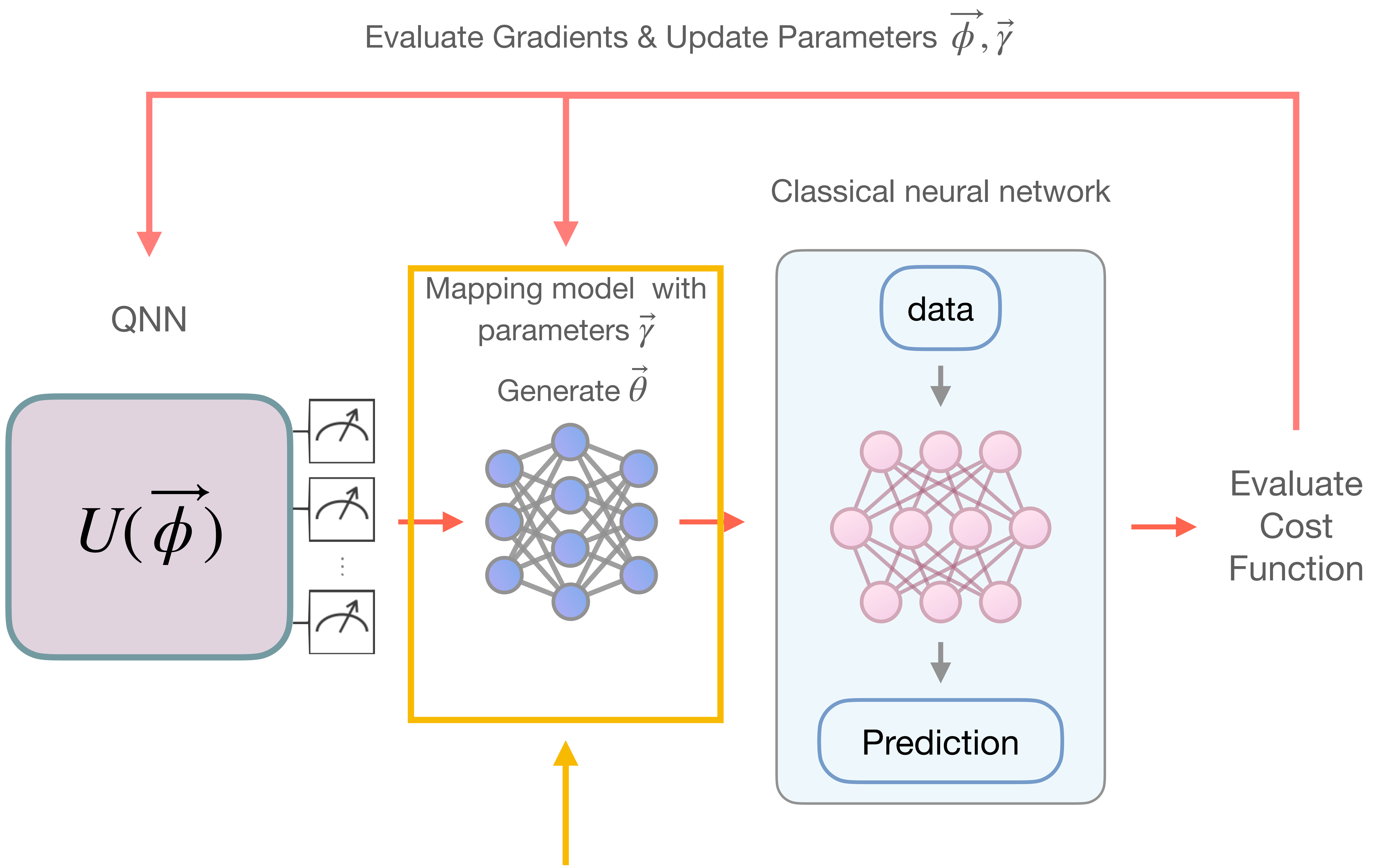
$$\forall i \in \{1,2,\ldots,M\}$$

Liu, C. Y., Kuo, E. J., Lin, C. H. A., Young, J. G., Chang, Y. J., Hsieh, M. H., & Goan, H. S. (2024). Quantum-Train: Rethinking Hybrid Quantum-Classical Machine Learning in the Model Compression Perspective. *arXiv preprint arXiv:2405.11304*.

- “Generate” the classical NN parameters by QNN
- The “trained” result is a classical NN

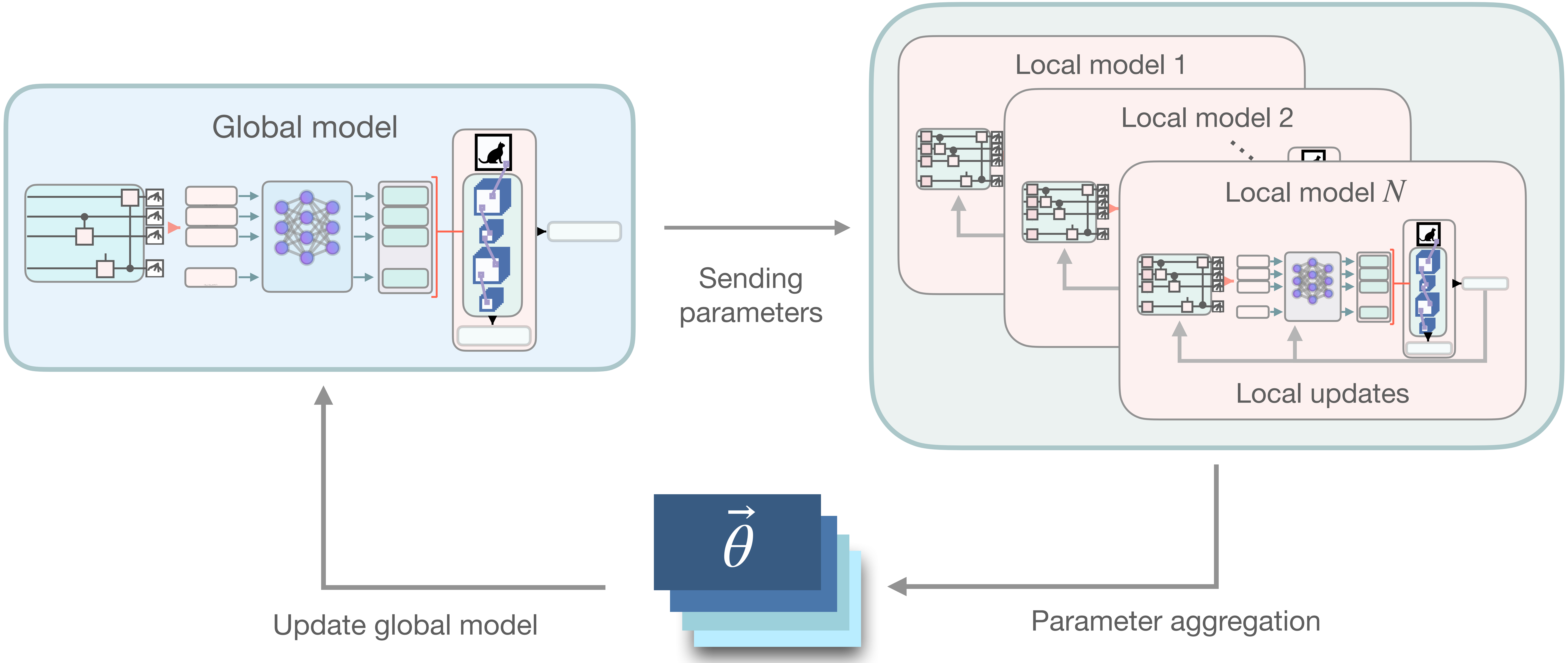


Liu, C. Y., Kuo, E. J., Lin, C. H. A., Young, J. G., Chang, Y. J., Hsieh, M. H., & Goan, H. S. (2024). Quantum-Train: Rethinking Hybrid Quantum-Classical Machine Learning in the Model Compression Perspective. *arXiv preprint arXiv:2405.11304*.



Mapping model is required to transform (rescale) the expectation values.

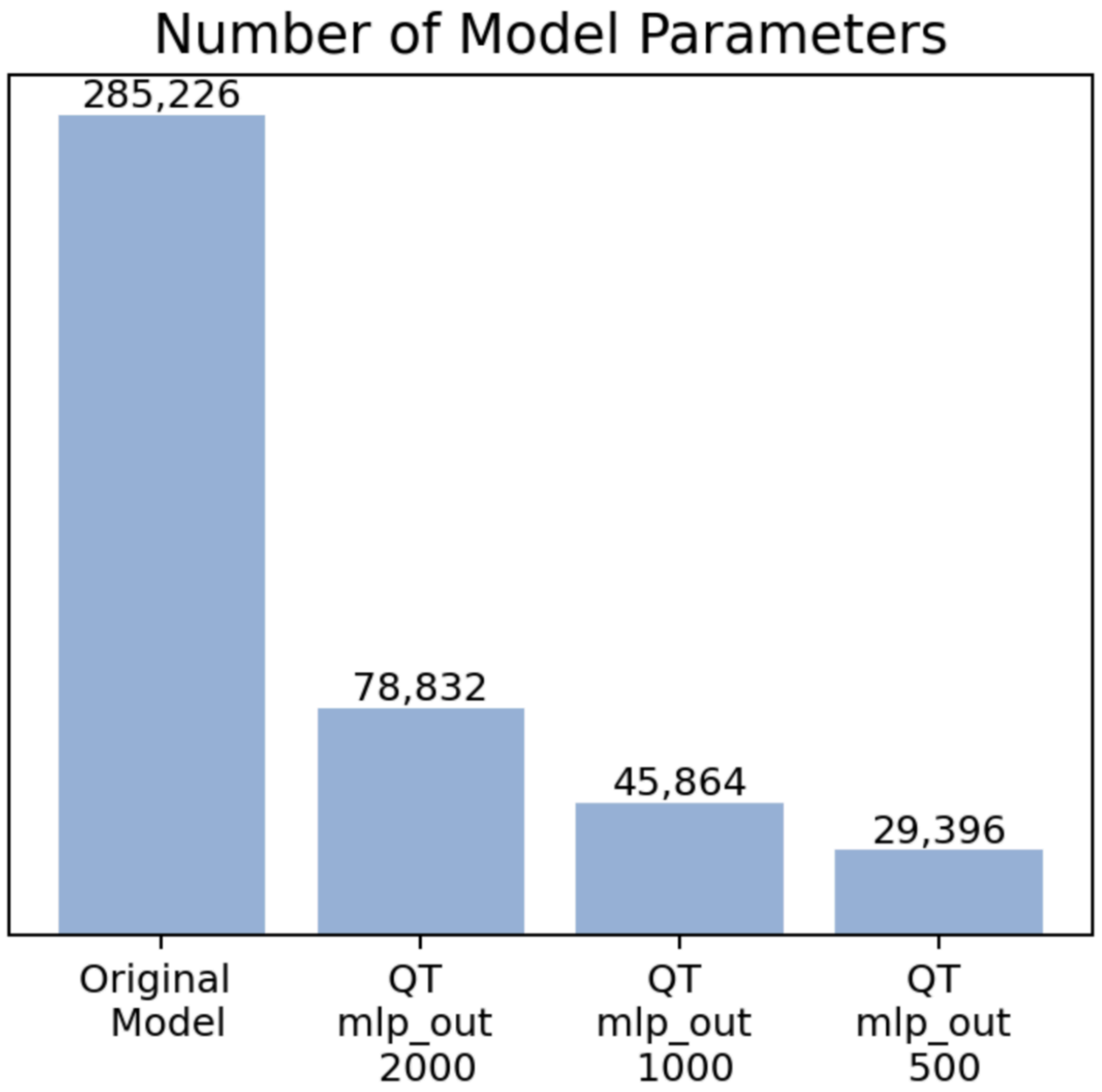
Liu, C. Y., Kuo, E. J., Lin, C. H. A., Young, J. G., Chang, Y. J., Hsieh, M. H., & Goan, H. S. (2024). Quantum-Train: Rethinking Hybrid Quantum-Classical Machine Learning in the Model Compression Perspective. *arXiv preprint arXiv:2405.11304*.



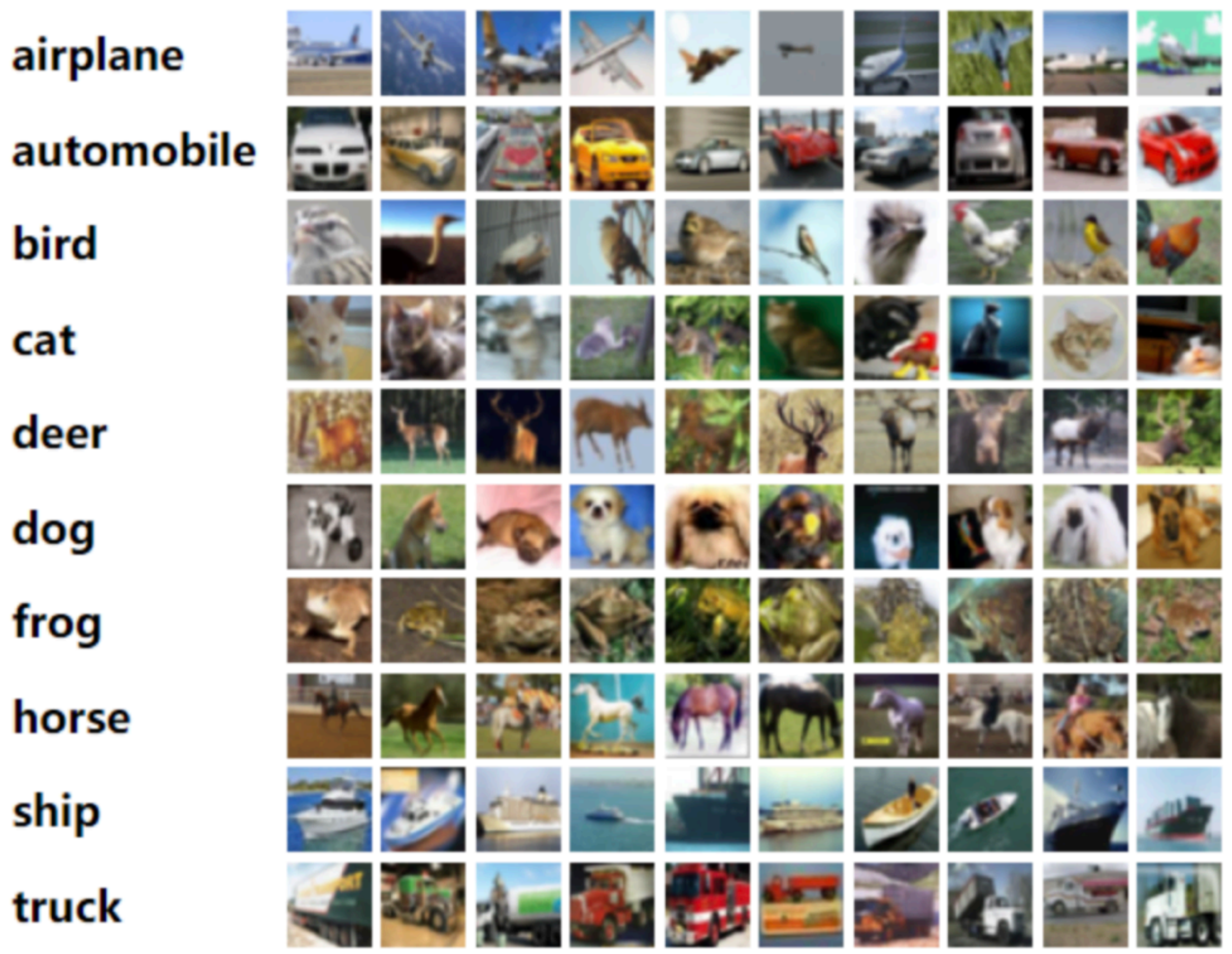
Liu, C. Y., & Chen, S. Y. C. (2024). Federated quantum-train with batched parameter generation. *arXiv preprint arXiv:2409.02763*.

Use less training parameters by QT

- VGG-like CNN with 285226 parameters
- QT-BG2000 with 78832 parameters
- QT-BG1000 with 45864 parameters
- QT-BG500 with 29396 parameters

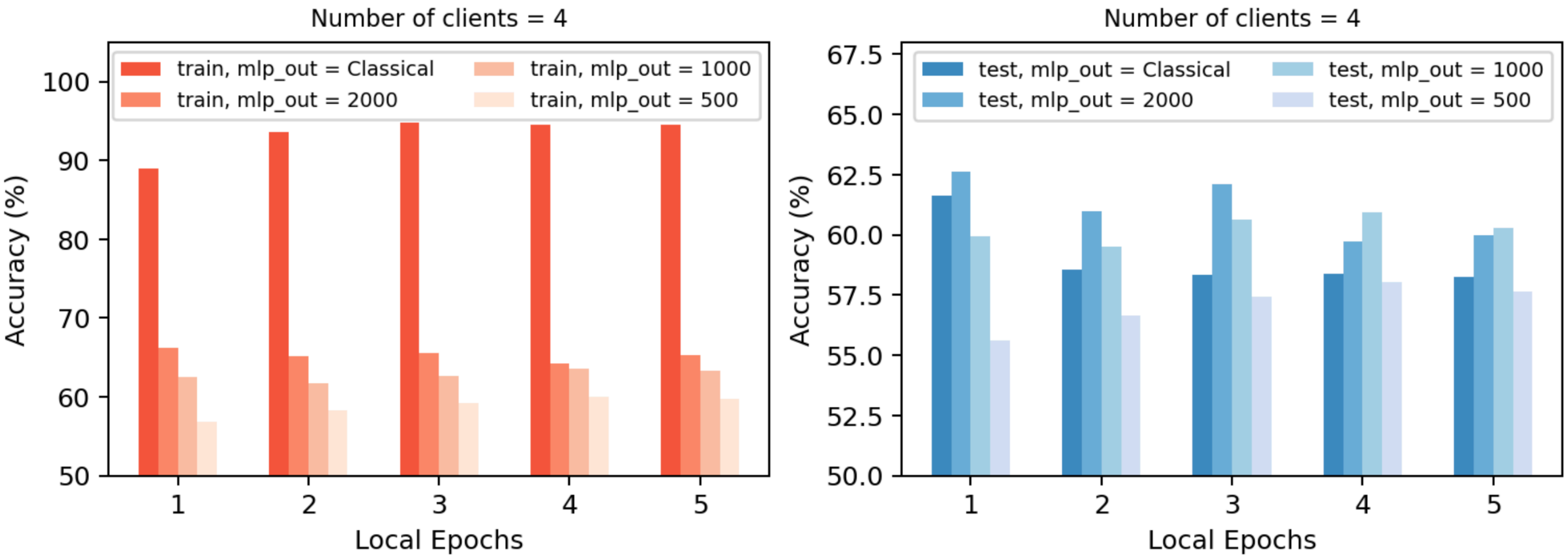


CIFAR-10 dataset



Liu, C. Y., & Chen, S. Y. C. (2024). Federated quantum-train with batched parameter generation. *arXiv preprint arXiv:2409.02763*.

Quantum-Train closing the gap between training acc and testing acc, the so called *generalization error*! (arXiv:2405.11304)

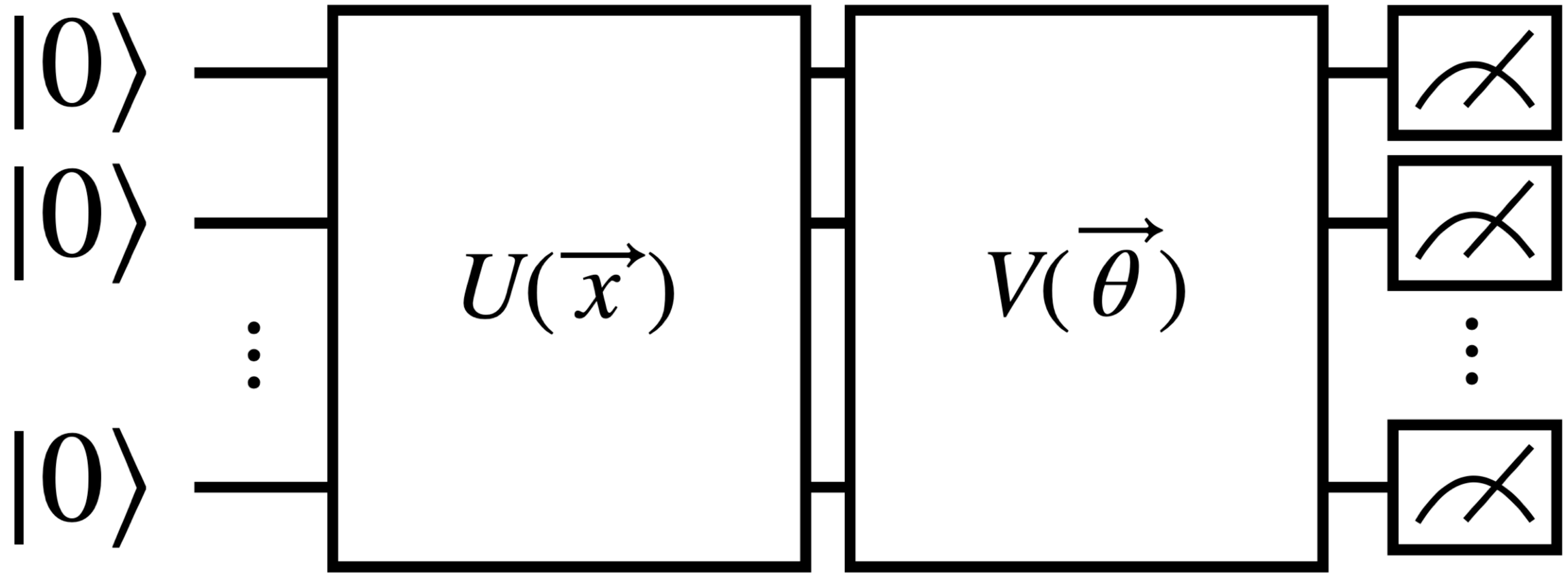


Liu, C. Y., & Chen, S. Y. C. (2024). Federated quantum-train with batched parameter generation. *arXiv preprint arXiv:2409.02763*.

- Fundamentals of Quantum Computing
- Hybrid Quantum-Classical Paradigm
- Variational Quantum Circuits (a.k.a Parameterized Quantum Circuits)
- Applications
- **Machine Learning for Quantum Machine Learning Model Design**
- Challenges in Quantum Machine Learning
- Conclusion and Outlook

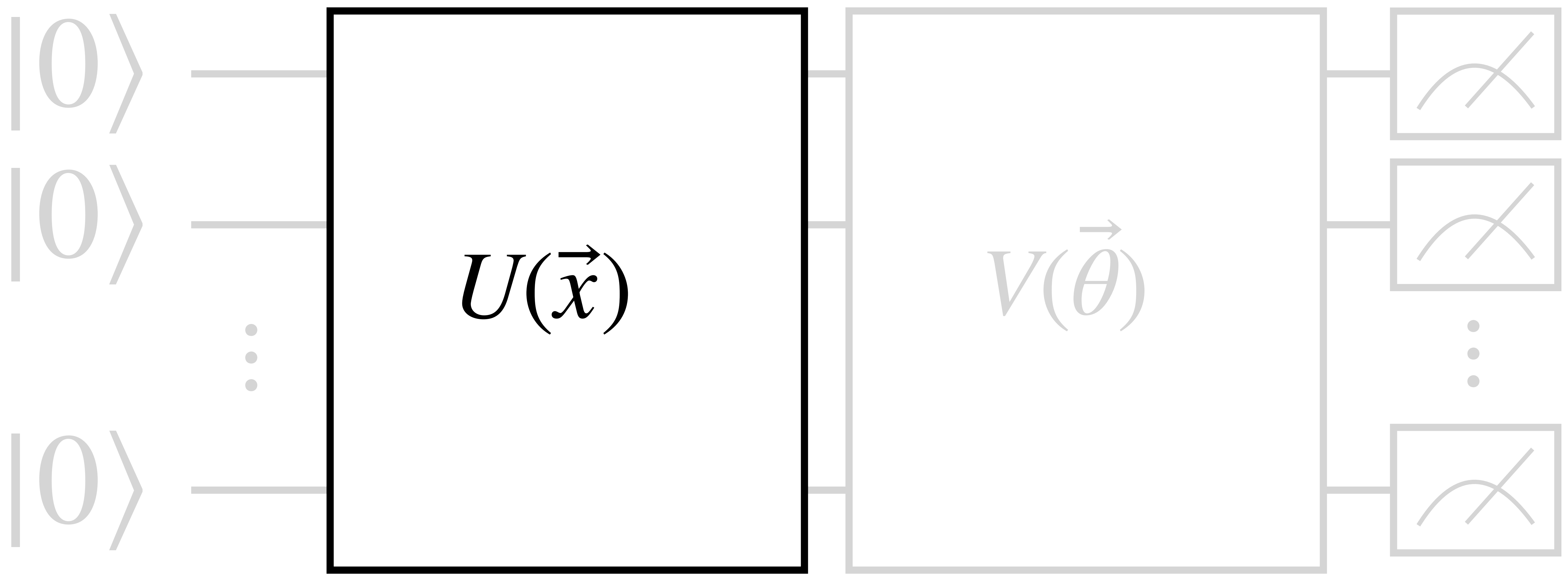
Quantum Circuit Design Challenges

Given a problem, we want to build something like this:



Quantum Circuit Design Challenges

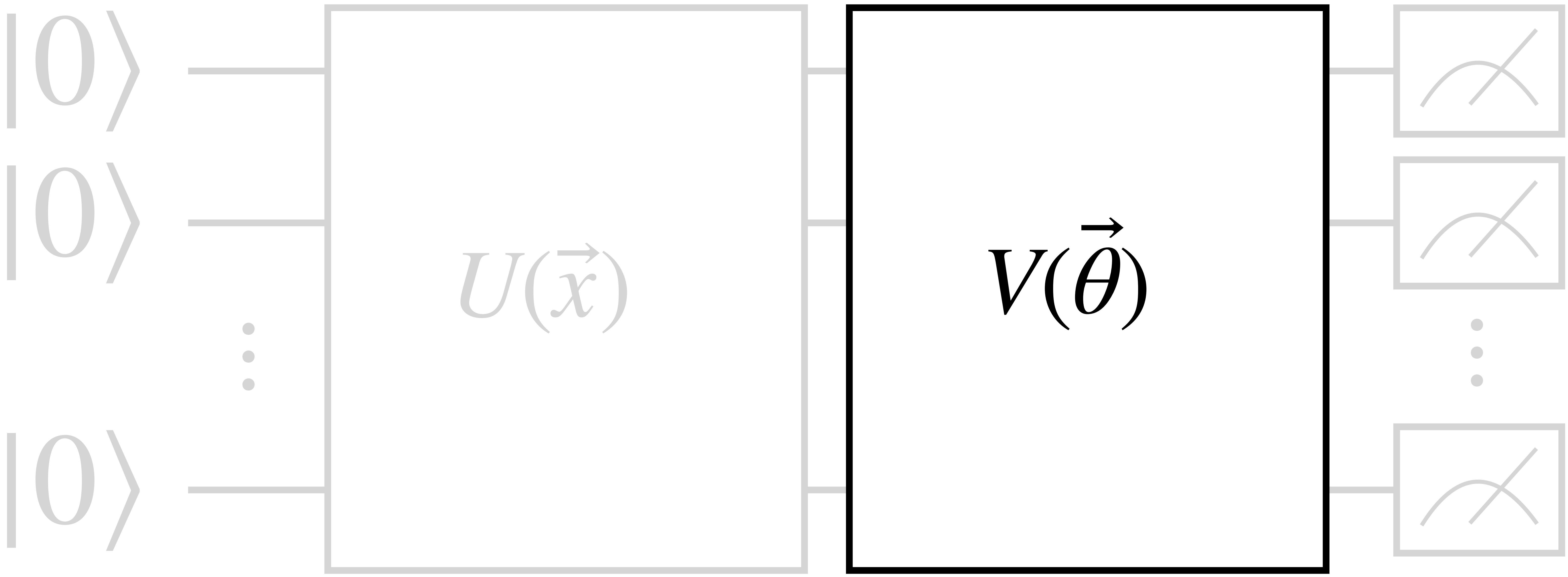
What should be those components?



How to design the **“encoding circuit”**?

Quantum Circuit Design Challenges

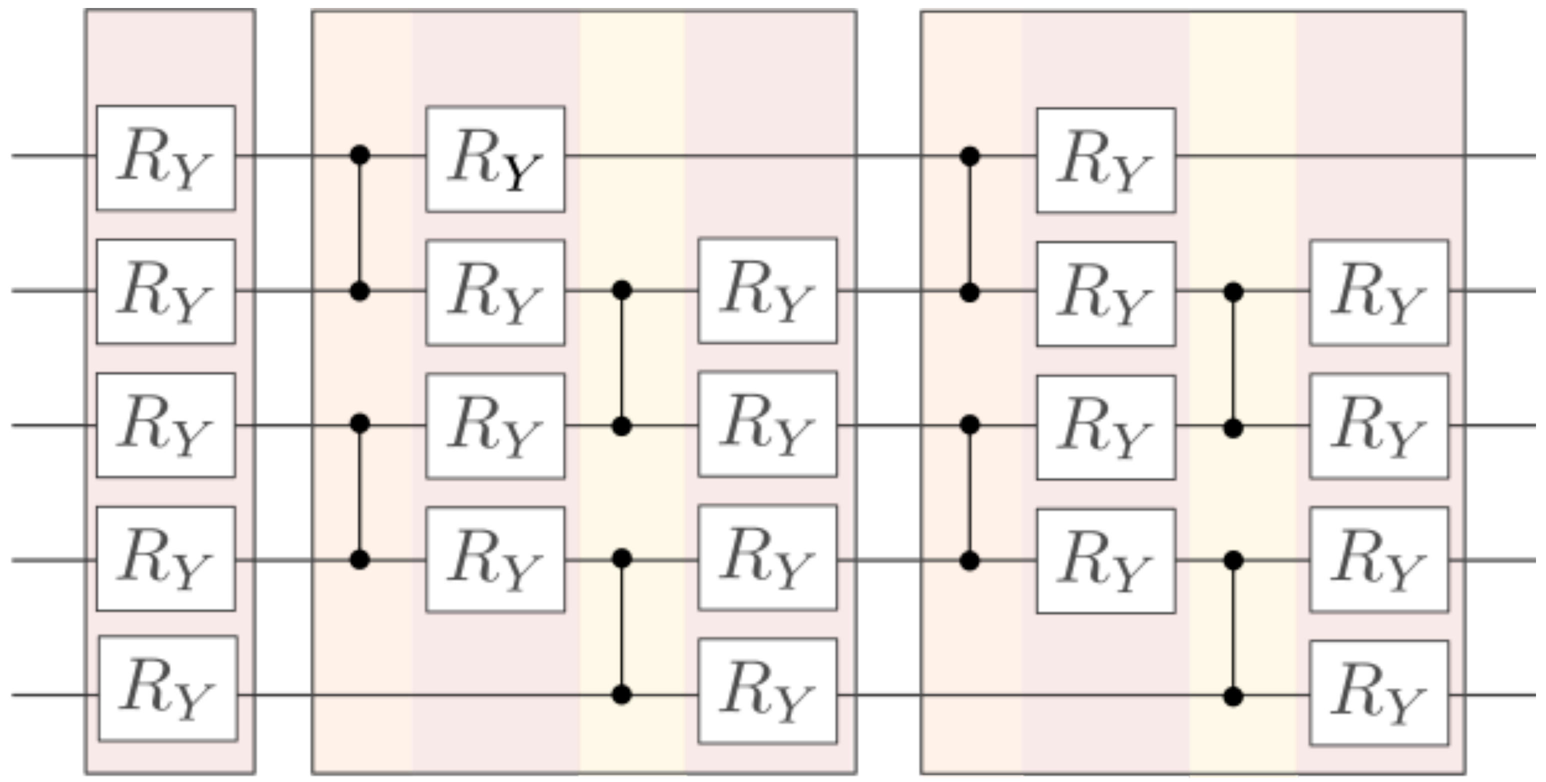
What should be those components?

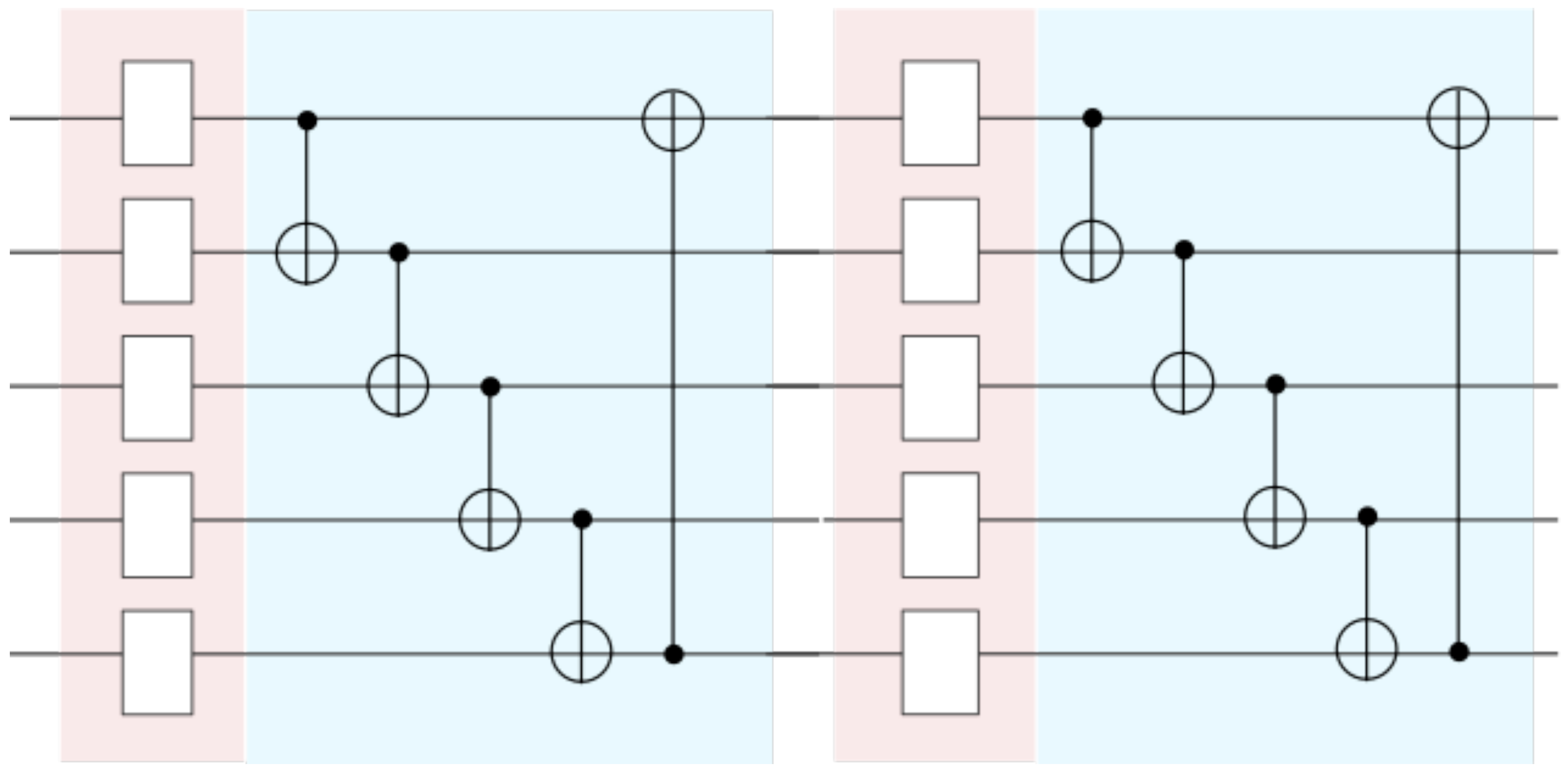


How to design the **“variational circuit”**?

Quantum Circuit Design Challenges

- There are many options for both **encoding circuit** and **variational circuit**.
- Different **initial circuit**, **entanglement structures, rotation gates (**R_X,R_Y, R_Z**)**





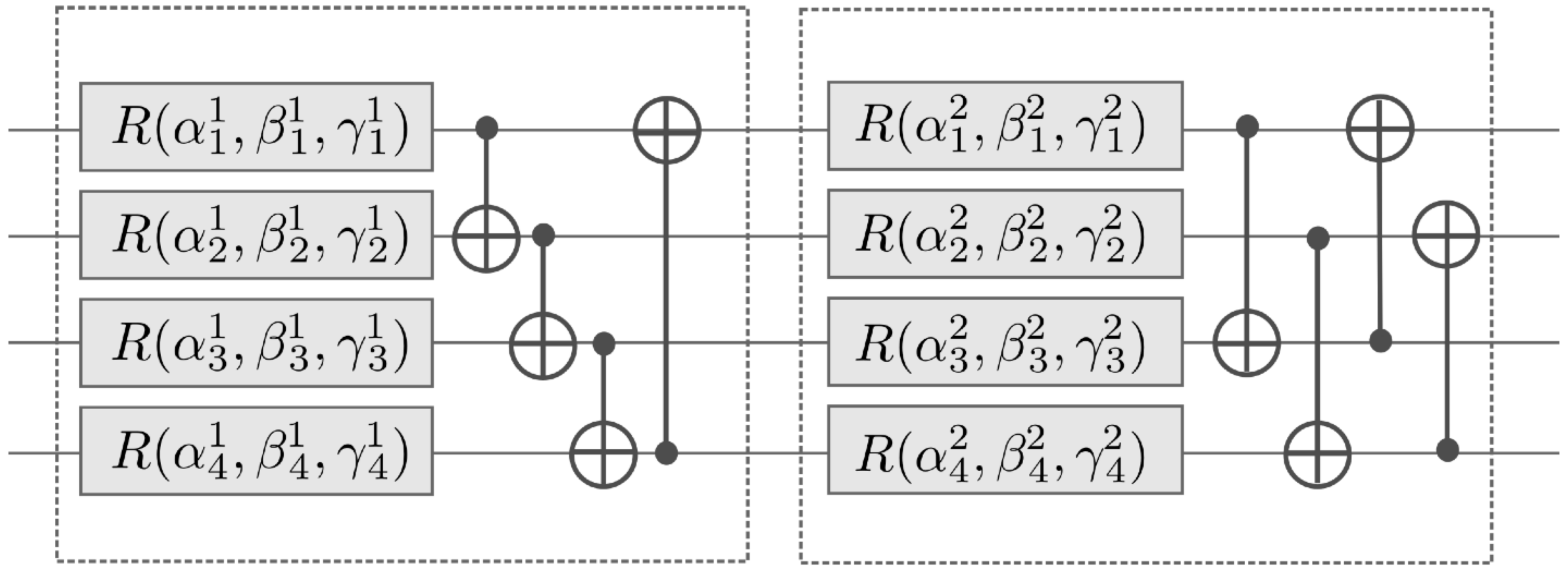


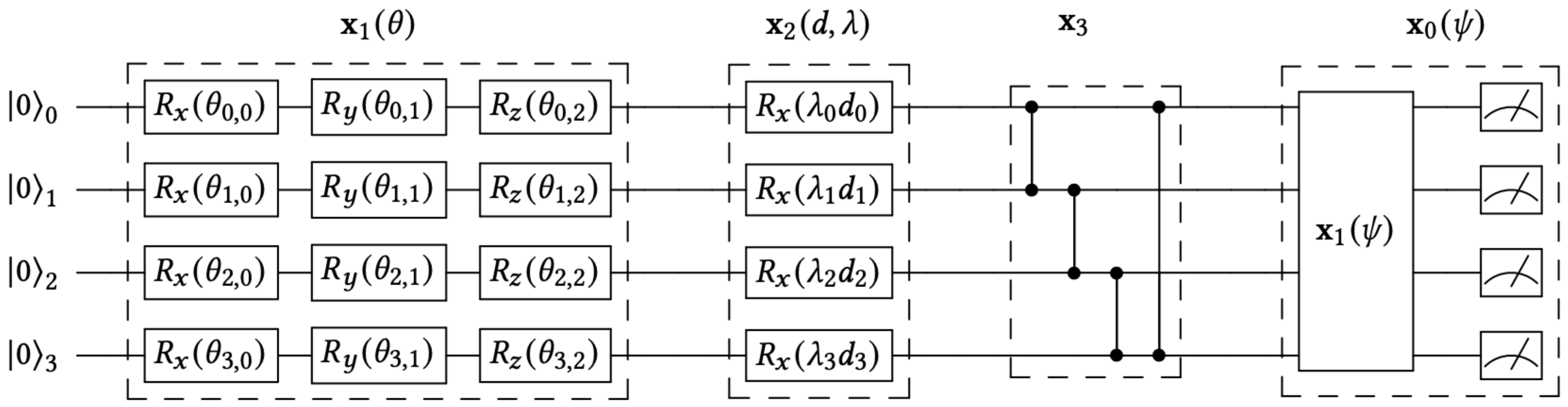
Image credit: PennyLane.ai

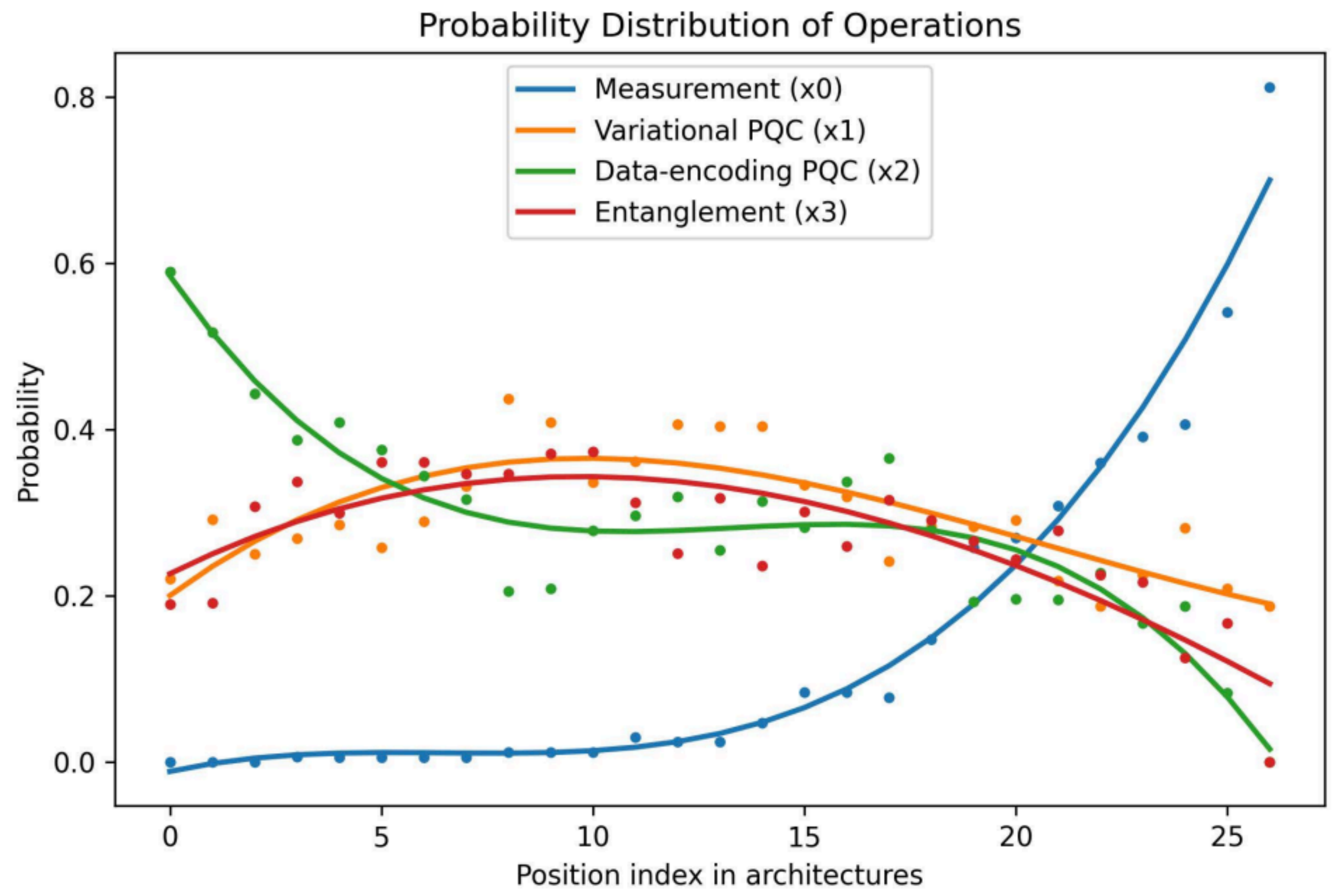
Quantum Architecture Search

- Evolutionary Optimization
- Reinforcement Learning
- Differentiable Search

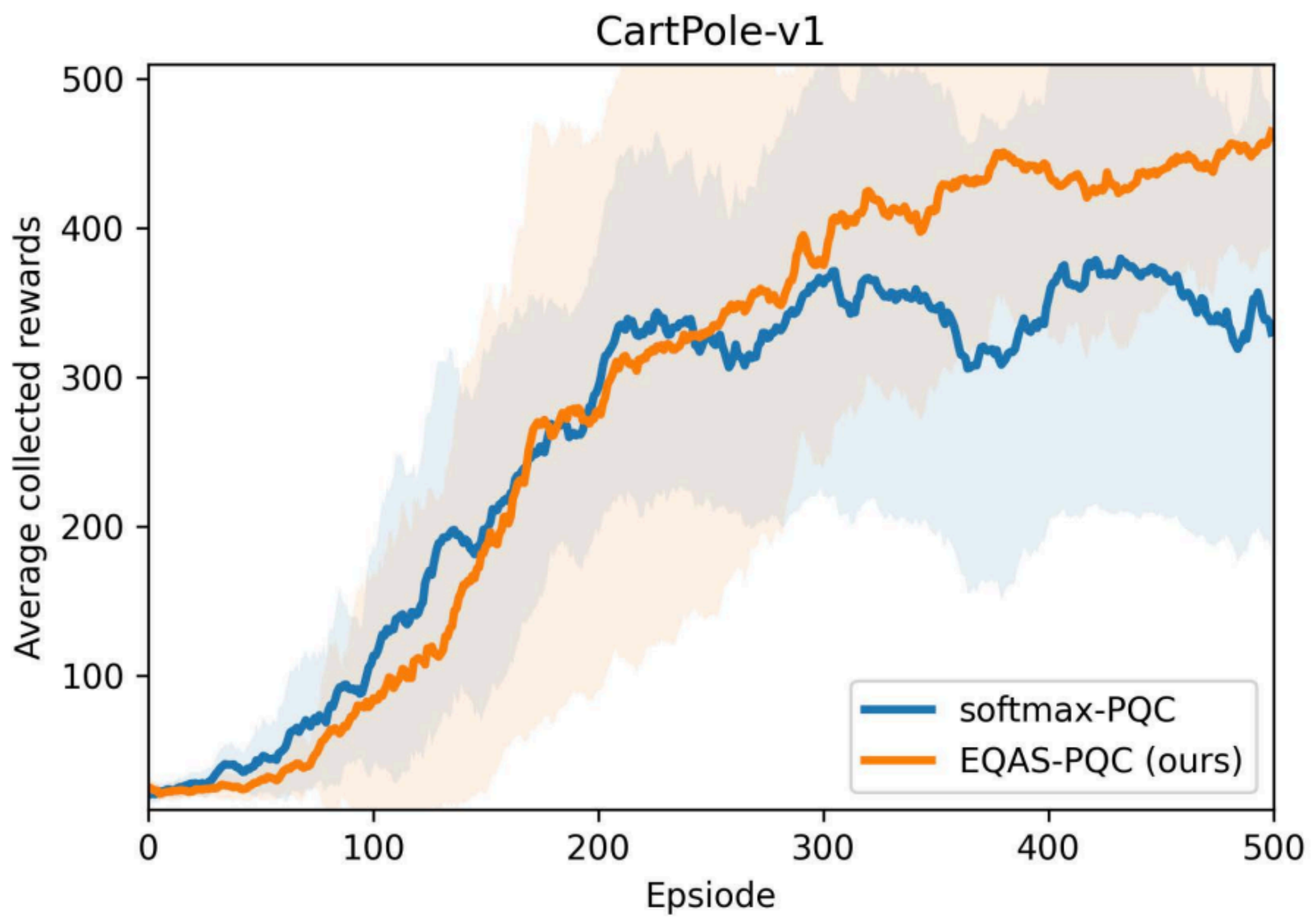
Evolutionary QAS

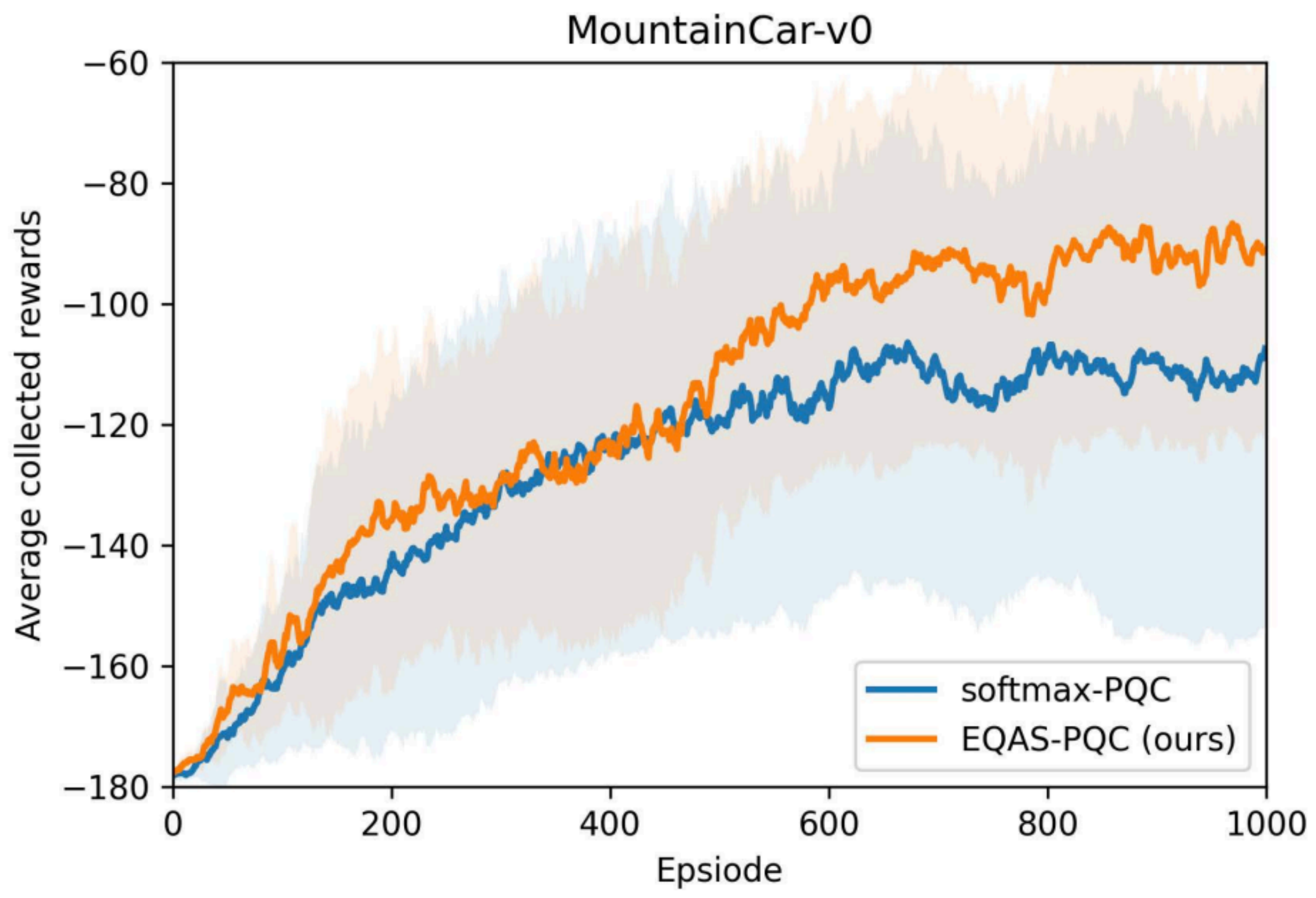
- Evolutionary Optimization





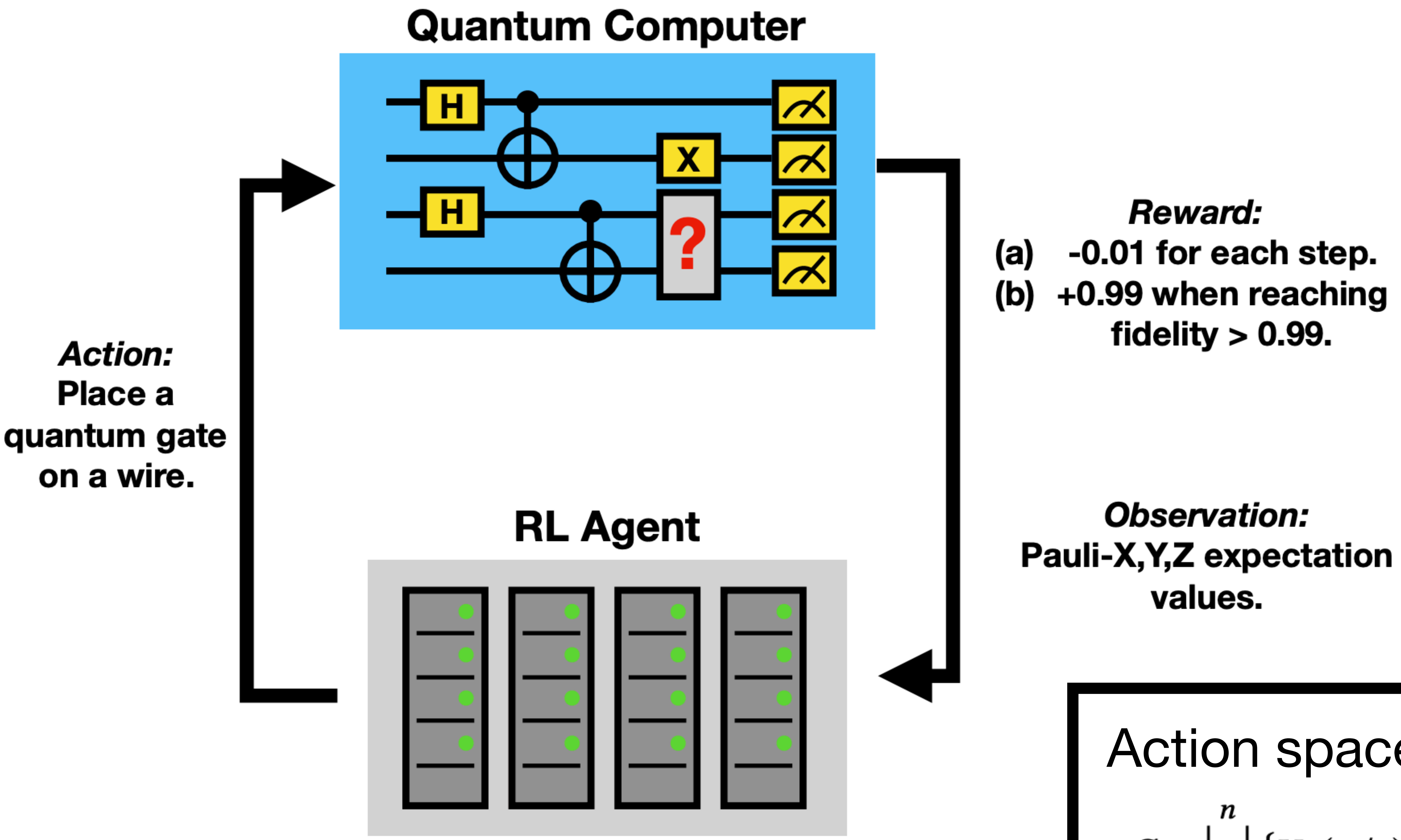
- $\mathbf{x}_1$: Variational PQC - A circuit with single-qubit rotations R_x, R_y, R_z performed on each qubit, with the rotation angles as trainable parameters.
- $\mathbf{x}_2$: Data-encoding PQC - A circuit with single-qubit rotations R_x performed on each qubit, with the rotation angles is the input scaled by trainable parameters.
- $\mathbf{x}_3$: Entanglement - A circuit that performs circular entanglement to all the qubits by applying one or multiple controlled-Z gates.
- $\mathbf{x}_0$: Measurement - A Variational PQC followed by measurement.





Ding, L., & Spector, L. (2022, July). Evolutionary quantum architecture search for parametrized quantum circuits. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion* (pp. 2190-2195).

RL for Quantum Architecture Search

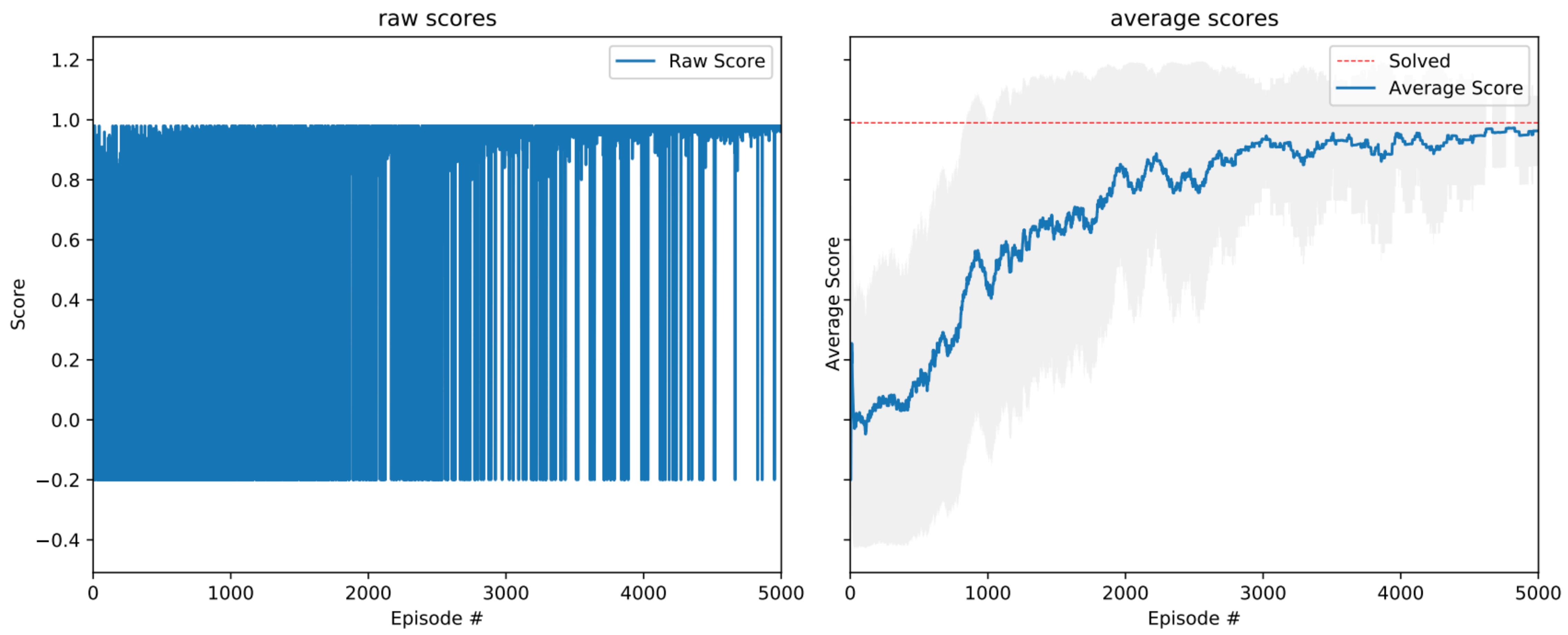


Action space for QAS:

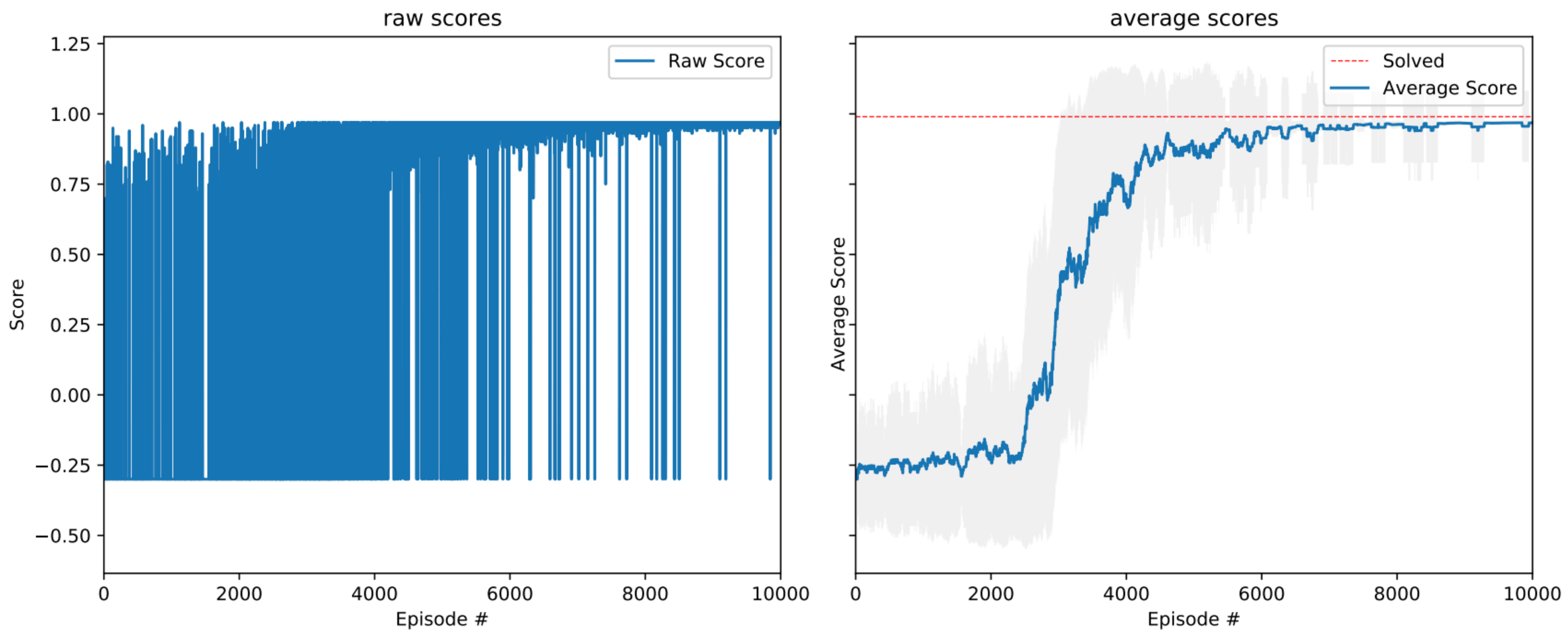
$$\mathbb{G} = \bigcup_{i=1}^{n} \left\{ U_i\left(\pi/4\right), X_i, Y_i, Z_i, H_i, CNOT_{i,(i+1)(mod 2)} \right\}$$

Kuo, E. J., Fang, Y. L. L., & Chen, S. Y. C. (2021). Quantum architecture search via deep reinforcement learning. *arXiv preprint arXiv:2104.07715.*

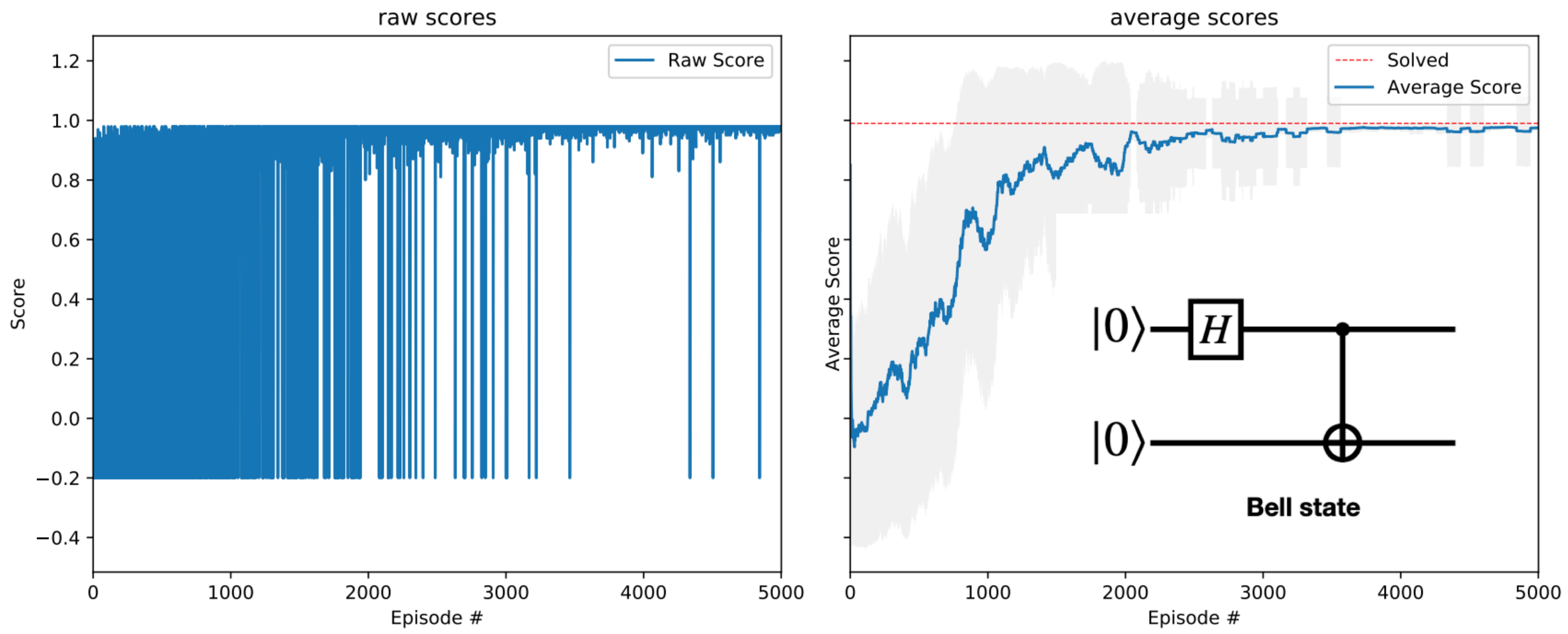
RL for Quantum Architecture Search



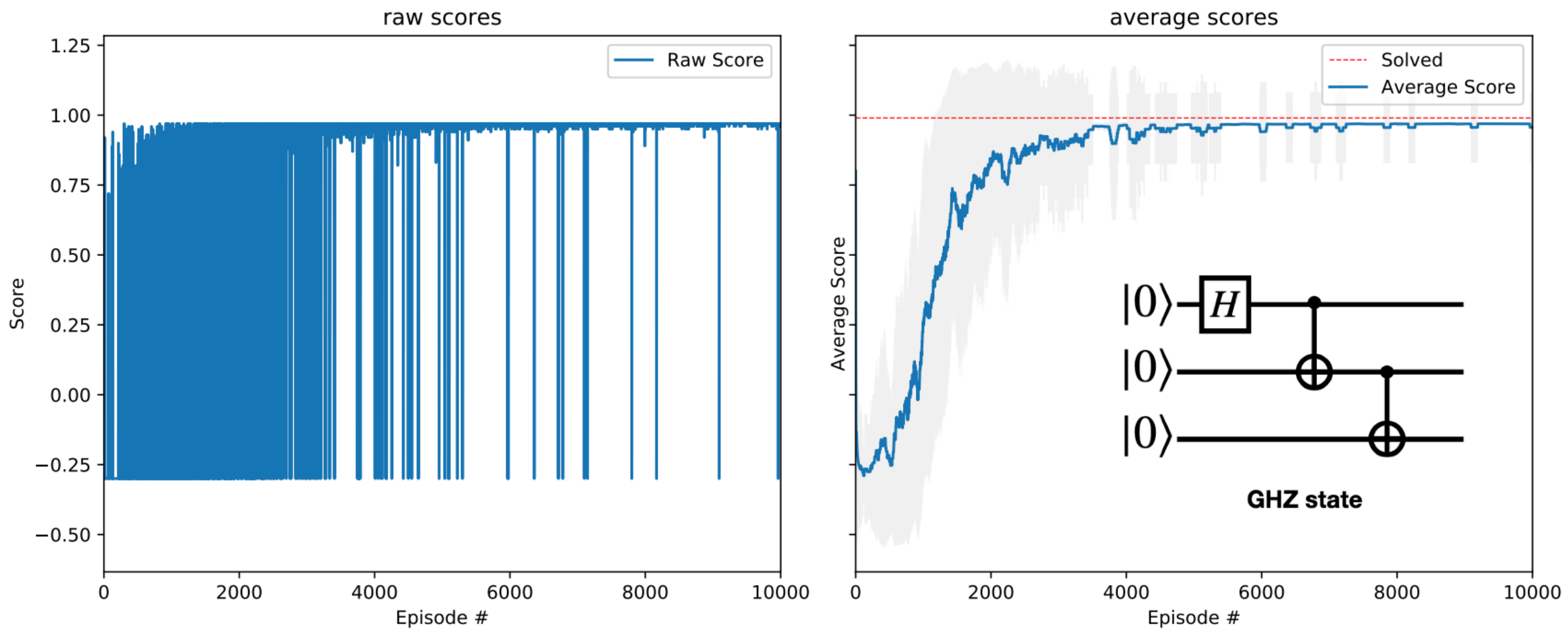
(a) **A2C for noise-free two-qubit system.**



(a) **A2C for noise-free three-qubit system.**



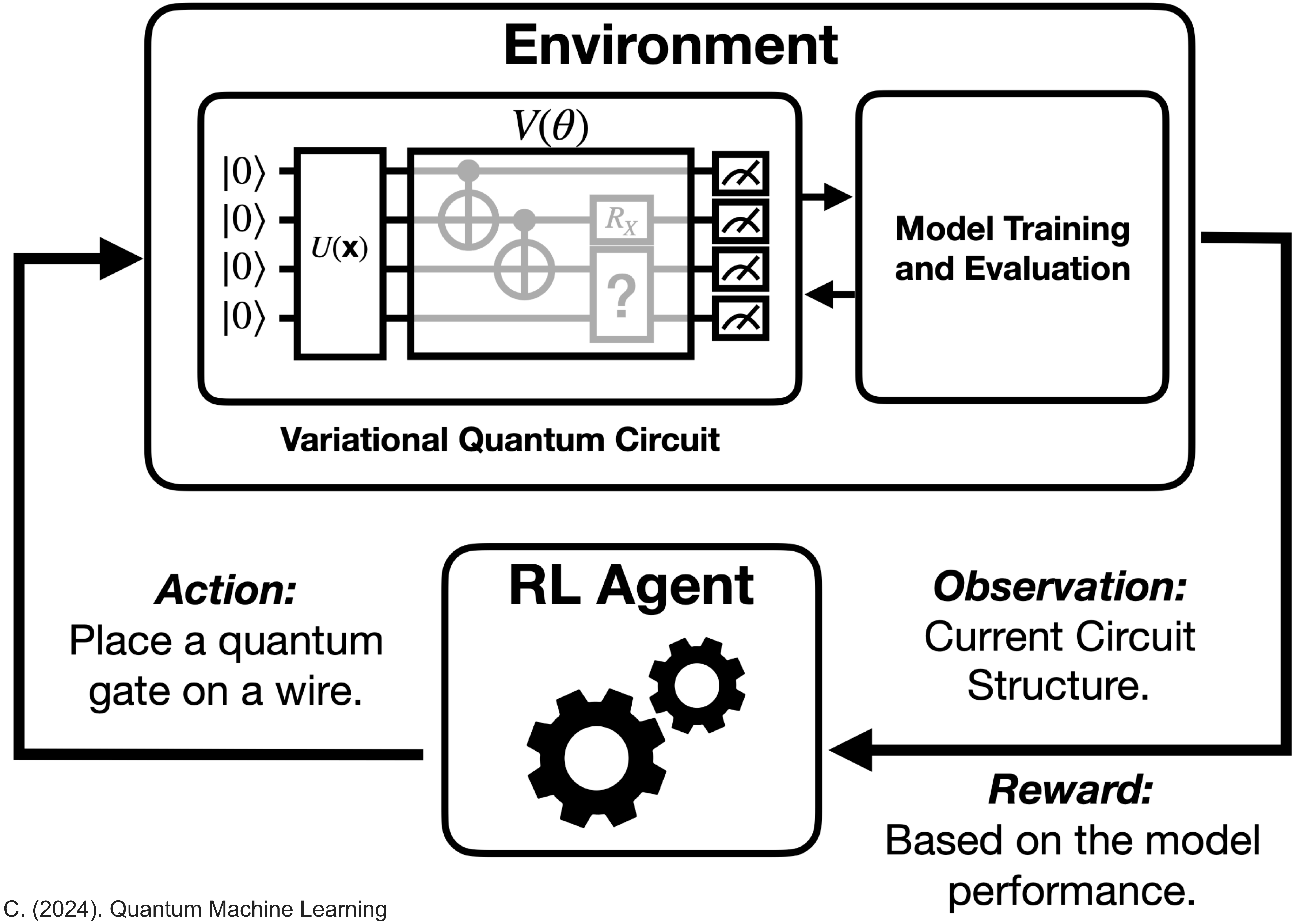
(b) **PPO for noise-free two-qubit system.**



(b) **PPO for noise-free three-qubit system.**

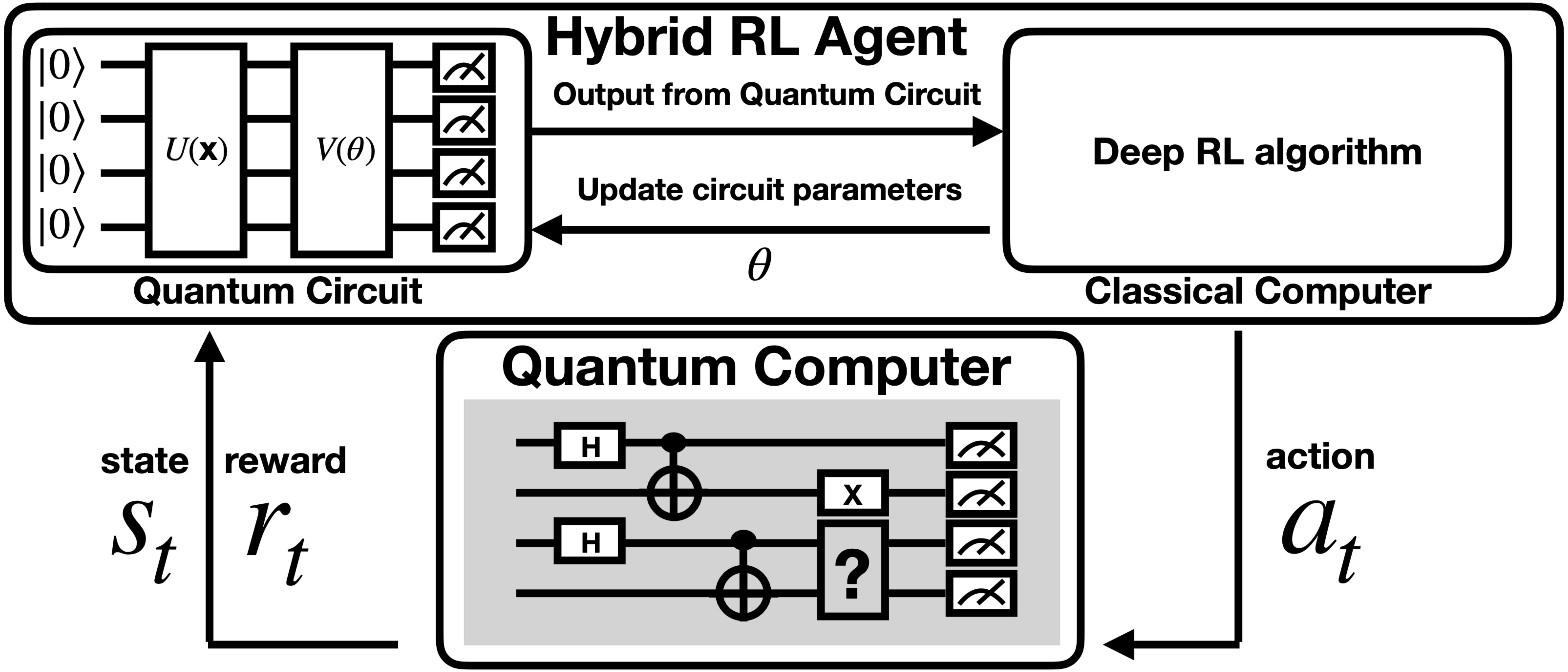
Kuo, E. J., Fang, Y. L. L., & Chen, S. Y. C. (2021). Quantum architecture search via deep reinforcement learning. *arXiv preprint arXiv:2104.07715*.

RL for Quantum Architecture Search



Dai, X., Wei, T. C., Yoo, S., & Chen, S. Y. C. (2024). Quantum Machine Learning Architecture Search via Deep Reinforcement Learning. *arXiv preprint arXiv:2407.20147*.
IEEE QCE 2024

QRL for QAS



Chen, S. Y. C. (2023, August). Quantum reinforcement learning for quantum architecture search. In *Proceedings of the 2023 International Workshop on Quantum Classical Cooperative* (pp. 17-20).

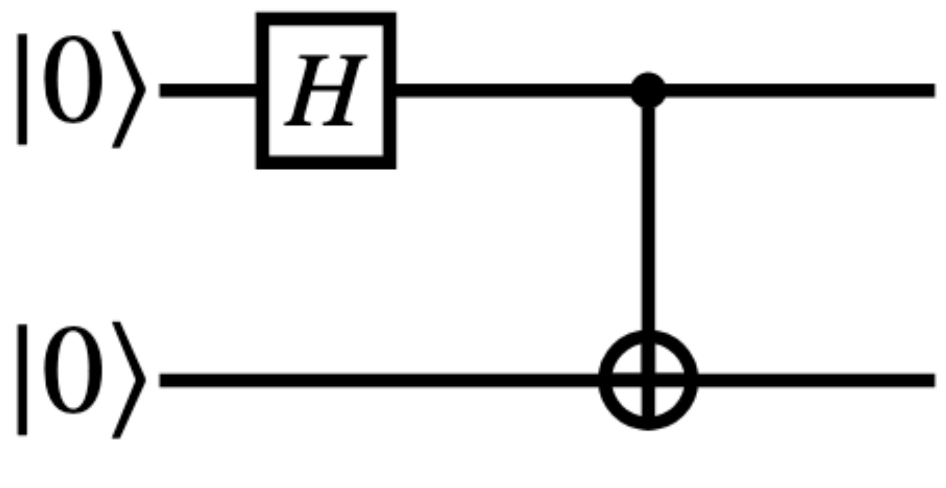
QRL for QAS

- Qiskit simulator with OpenAI Gym wrapper

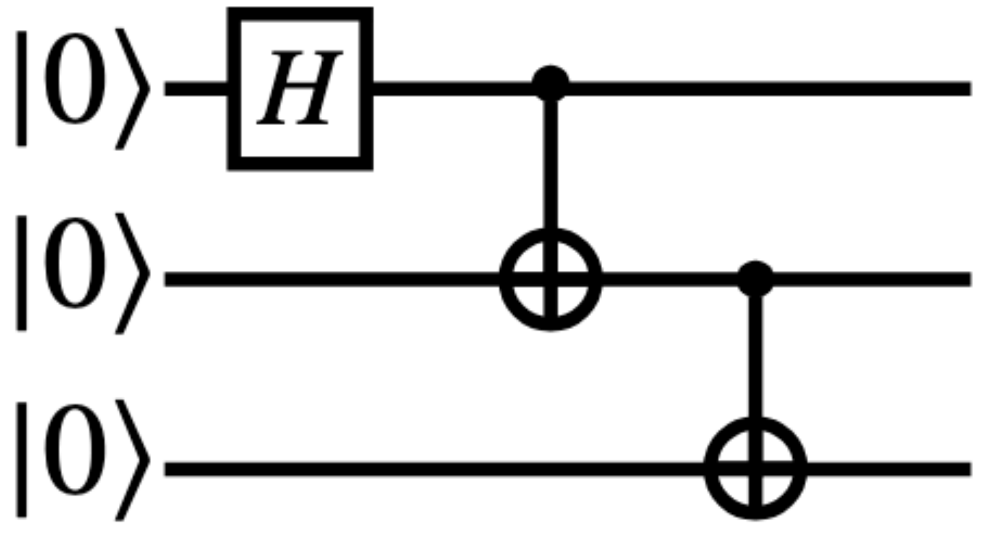
- **State**: Pauli-X, Y, Z expectation values for each qubit. (3n-dimensional vector where n is the number of qubits)

- **Action**: single qubit gates and CNOT gate

- **Reward**: for every step, the environment will feedback a –0.01 reward to **encourage the agent to use smaller number of steps**. If the fidelity of quantum states reach a certain threshold (e.g. 0.95), the reward will be (fidelity – 0.01) and the episode terminates.



Bell state



GHZ state

Action space for QAS:

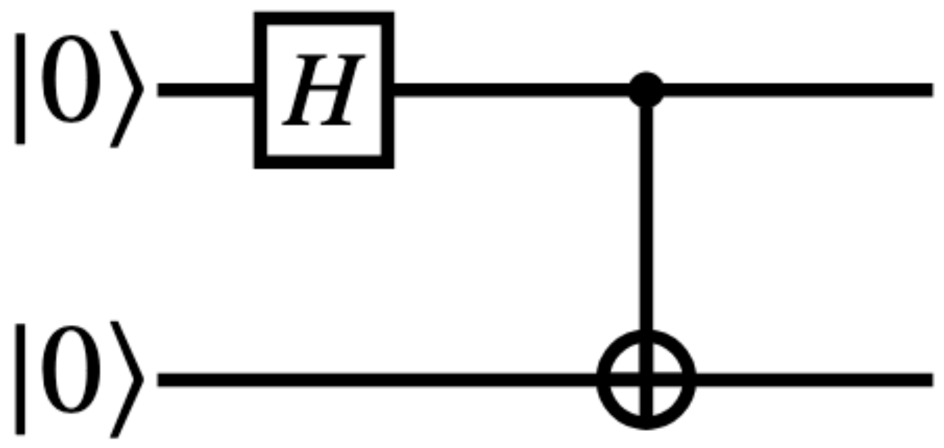
$$\mathbb{G} = \bigcup_{i=1}^{n} \left\{ U_i\left(\pi/4\right), X_i, Y_i, Z_i, H_i, CNOT_{i,(i+1)(mod2)} \right\}$$

Chen, S. Y. C. (2023, August). Quantum reinforcement learning for quantum architecture search. In *Proceedings of the 2023 International Workshop on Quantum Classical Cooperative* (pp. 17-20).

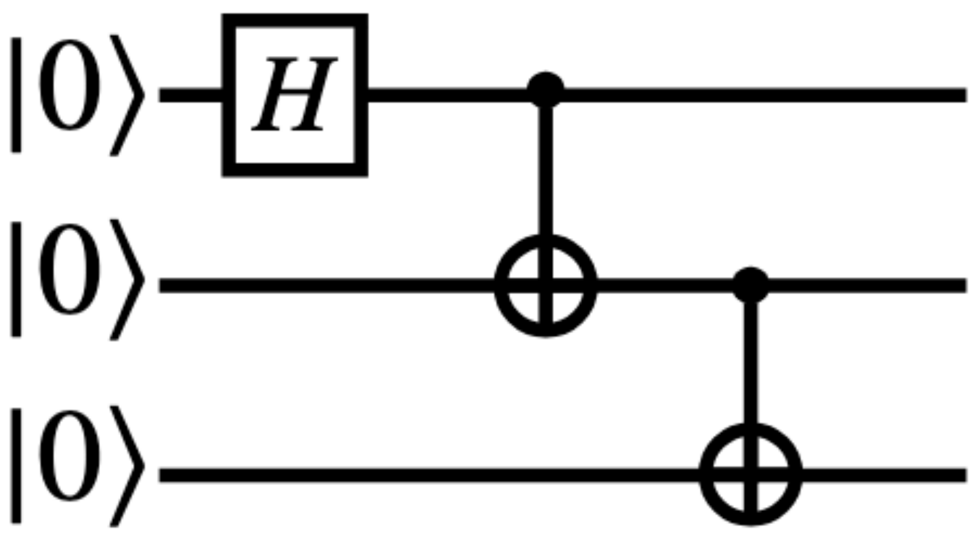
QRL for QAS

- With quantum A3C training algorithms, the hybrid quantum-classical RL agent can find the circuit for *Bell state (two-qubit)* and *GHZ state (three-qubit)*

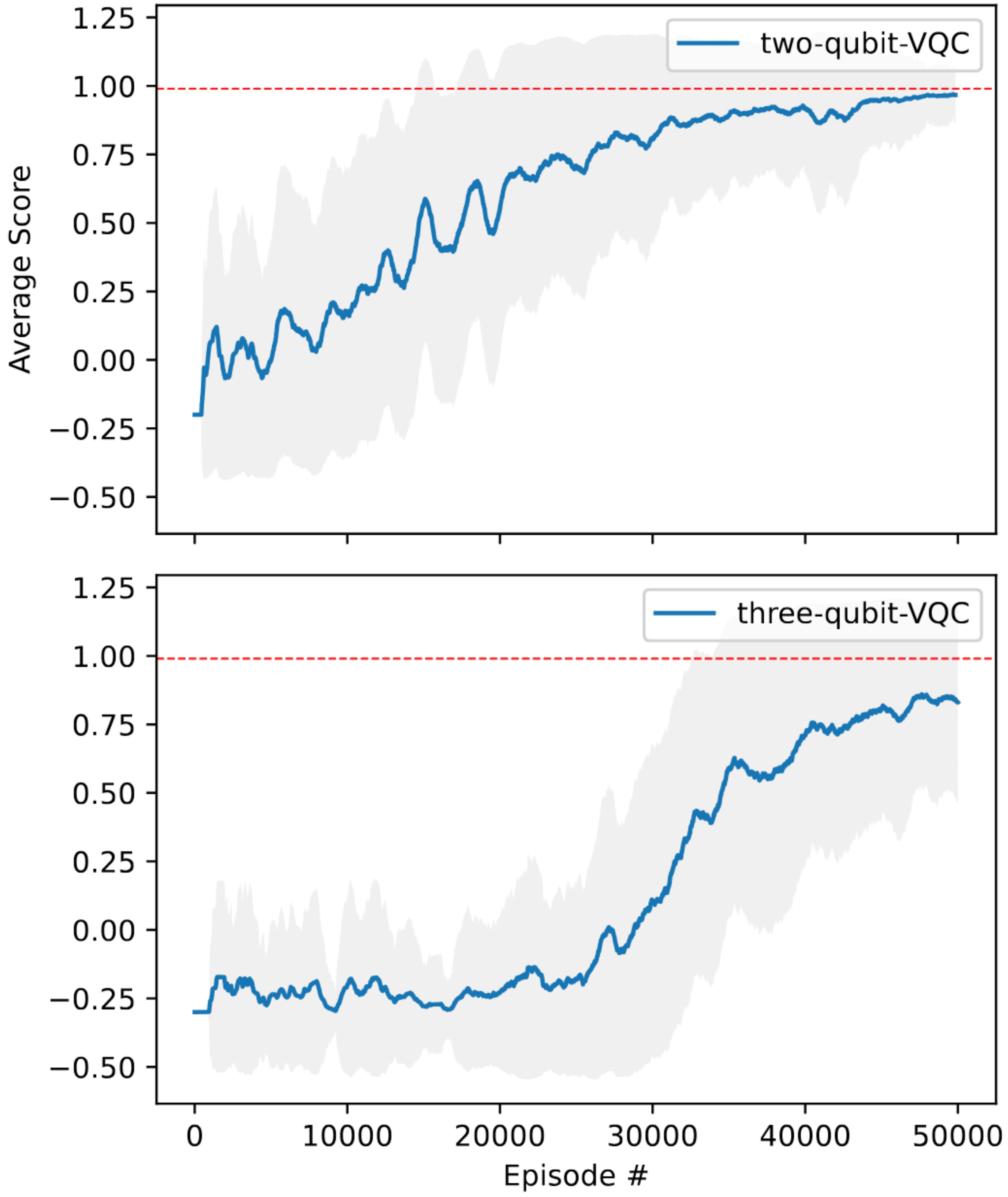
- The three-qubit case requires more training episodes.



Bell state



GHZ state



Chen, S. Y. C. (2023, August). Quantum reinforcement learning for quantum architecture search. In *Proceedings of the 2023 International Workshop on Quantum Classical Cooperative* (pp. 17-20).

Challenges of Evo or RL QAS

- Less sample-efficient, requiring a *large number of interactions* or *iterations* to converge to a good architecture.

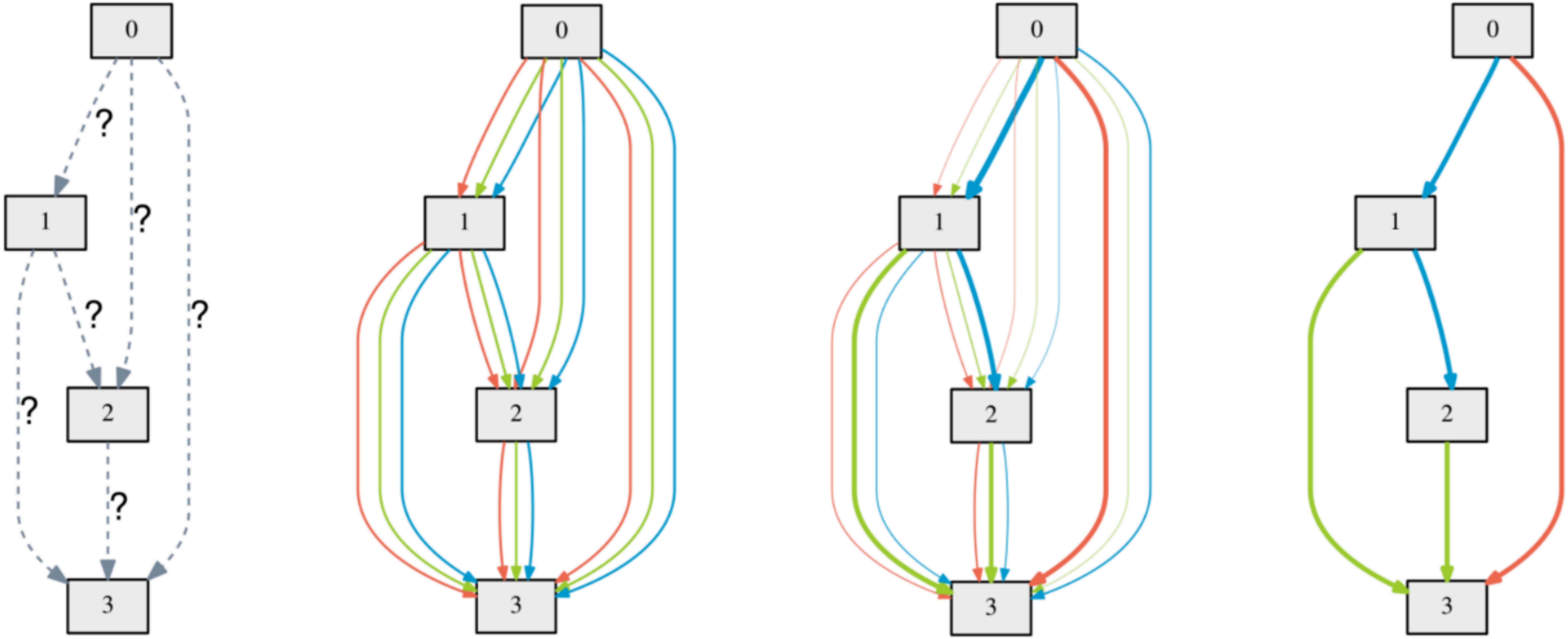
- May *converge slower* because they explore the search space in a more trial-and-error manner.

- More hyperparameters (e.g., mutation rates, crossover probabilities, exploration/exploitation ratios)

- Scalability issues in high-dimensional search spaces (more qubits, deeper quantum circuits).

Differentiable Quantum Architecture Search

Differentiable *Neural* Architecture Search:

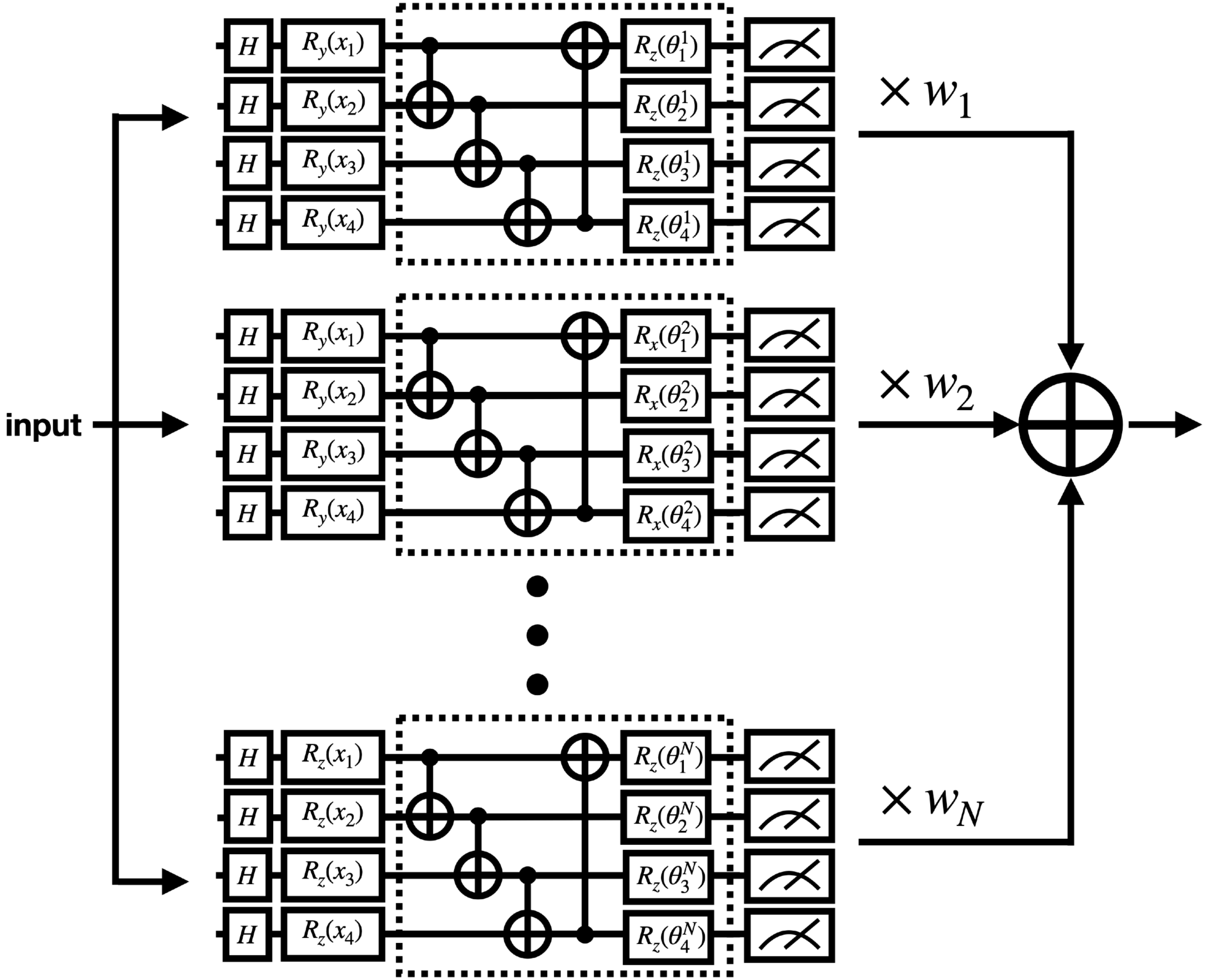


$$\bar{o}^{(i,j)}(x) = \sum_{o \in \mathcal{O}} \frac{\exp(\alpha_o^{(i,j)})}{\sum_{o' \in \mathcal{O}} \exp(\alpha_{o'}^{(i,j)})} o(x)$$

Ref: https://arxiv.org/pdf/1806.09055

Differentiable Quantum Architecture Search

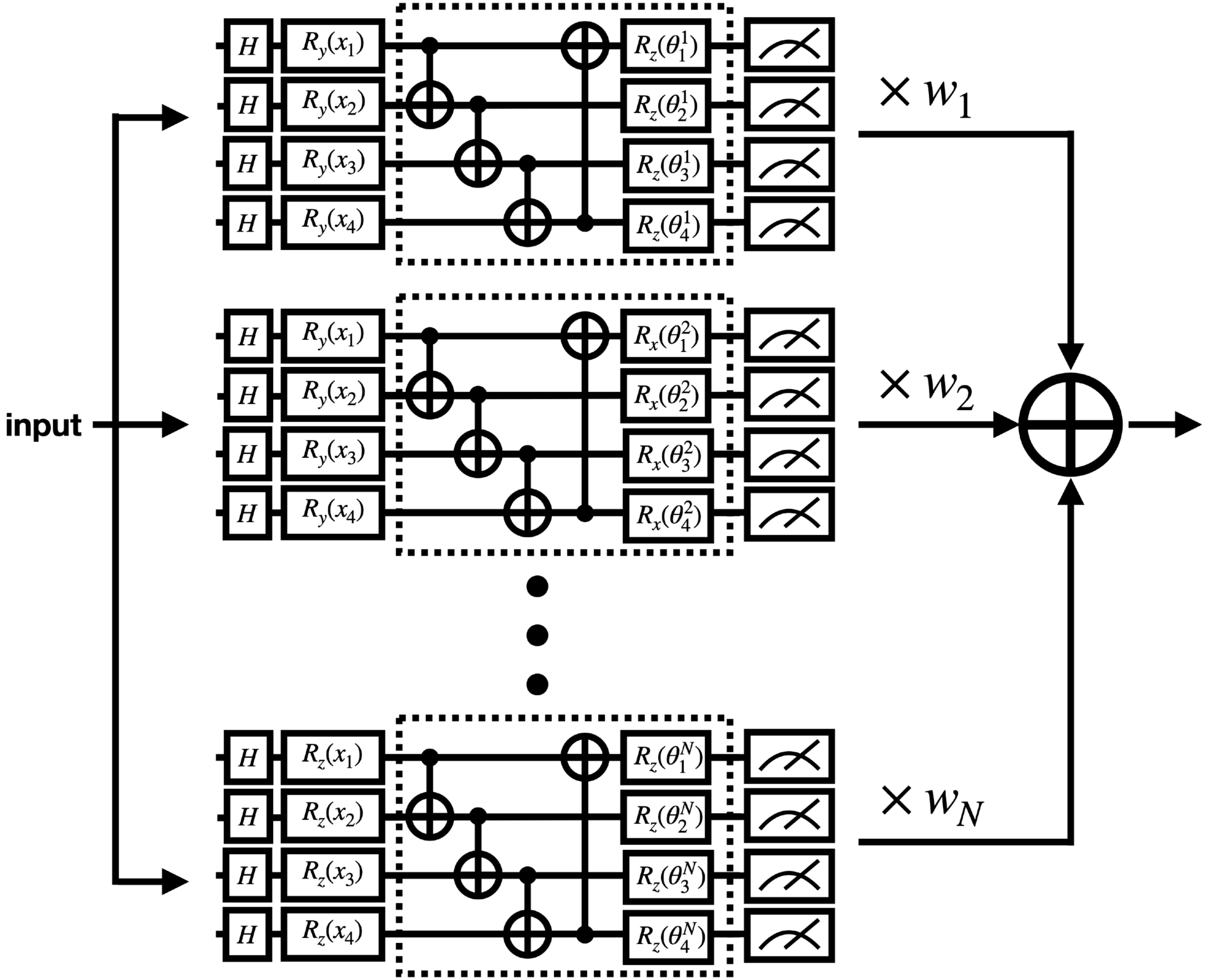
- Goal: Construct quantum circuit $\mathscr{C}$.
- Quantum circuit $\mathscr{C}$ has $\mathscr{S}_1, \mathscr{S}_2, \cdots, \mathscr{S}_n$ sub-components.
- Each $\mathscr{S}_i$ is associated with a corresponding set of allowable circuit choices $\mathscr{B}_i$.
- $|\mathscr{B}_i|$ denotes the number of permissible circuit choices for each sub-component i.
- Number of possible realization $\mathscr{C}$:
$N = |\mathscr{B}_1| \times |\mathscr{B}_2| \times \cdots \times |\mathscr{B}_n|$



Chen, S. Y. C. (2024). Differentiable Quantum Architecture Search in Asynchronous Quantum Reinforcement Learning. *arXiv preprint arXiv:2407.18202*. **IEEE QCE 2024**

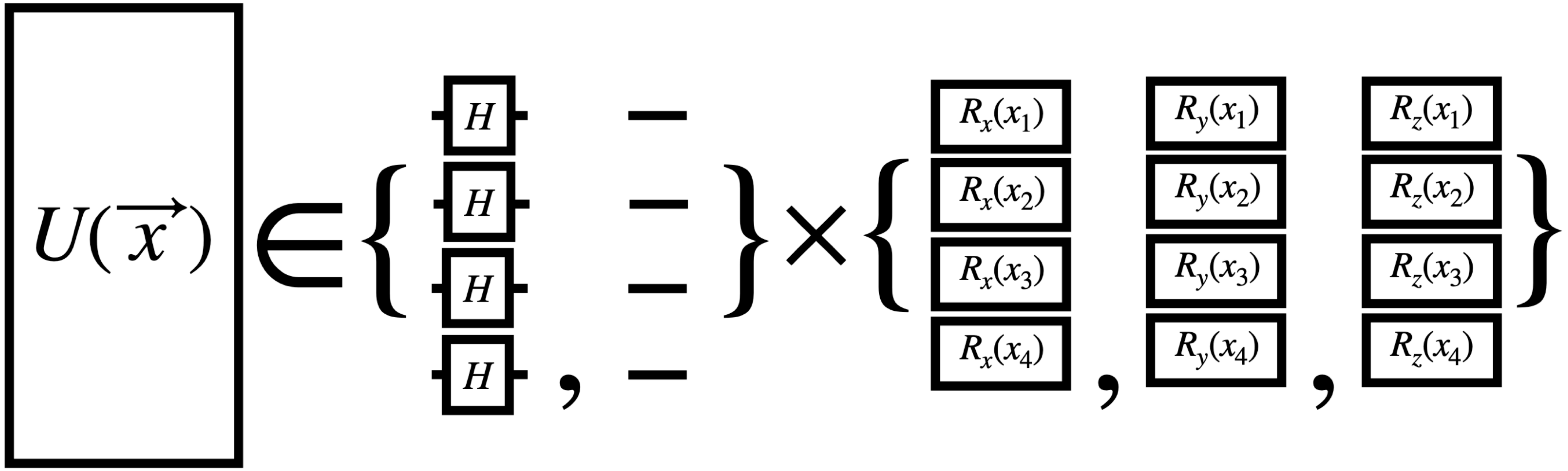
Differentiable Quantum Architecture Search

- Structural weights: w_j
- Each circuit realization $\mathscr{C}_j$ is associated with the trainable parameter θ_j
- Ensemble function $f_{\mathscr{C}} = \sum_{j=1}^{N} w_j f_{\mathscr{C}_j}$
- Loss: $\mathscr{L}(f_{\mathscr{C}})$
- Gradient: $\nabla_{w_j} \mathscr{L}(f_{\mathscr{C}})$

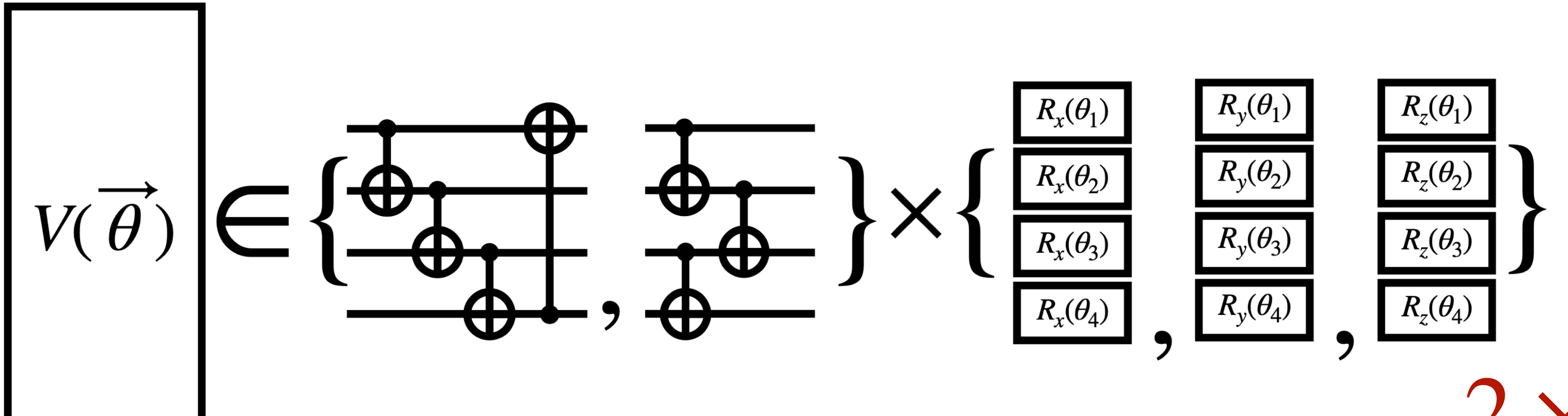


Chen, S. Y. C. (2024). Differentiable Quantum Architecture Search in Asynchronous Quantum Reinforcement Learning. *arXiv preprint arXiv:2407.18202*. **IEEE QCE 2024**

Differentiable Quantum Architecture Search



$2 \times 3 = 6$



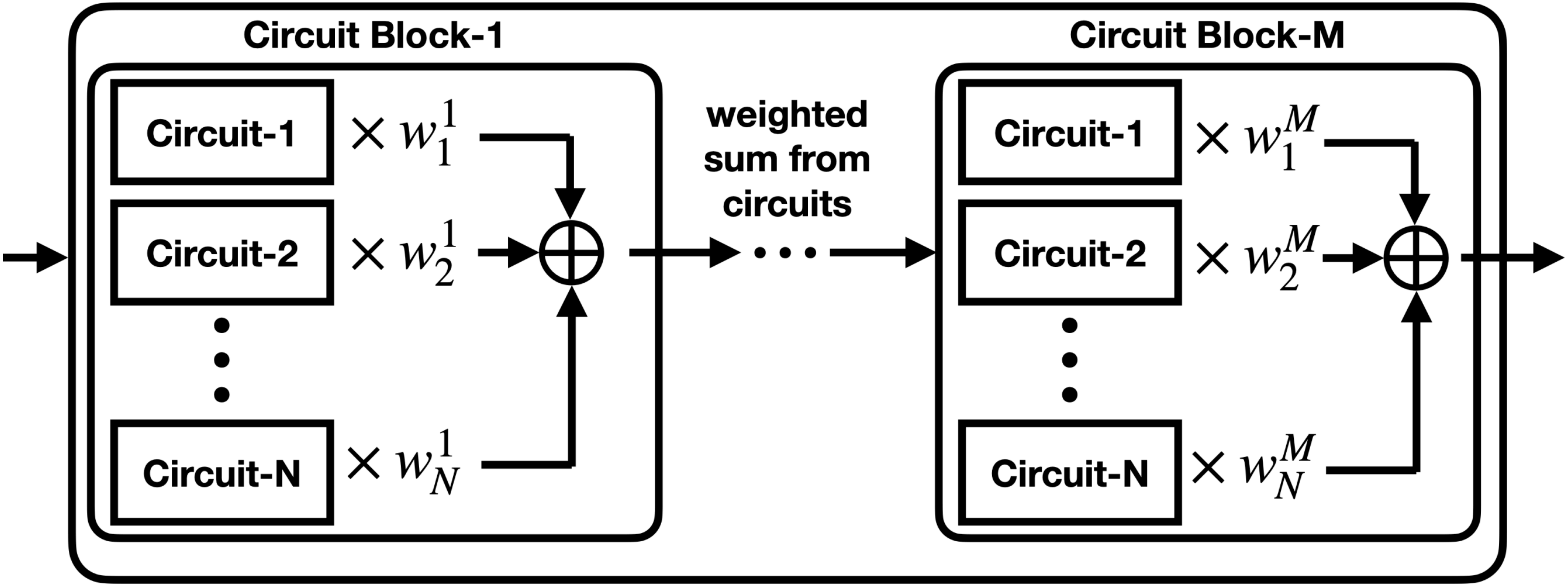
$2 \times 3 = 6$

Chen, S. Y. C. (2024). Differentiable Quantum Architecture Search in Asynchronous Quantum Reinforcement Learning. *arXiv preprint arXiv:2407.18202*. **IEEE QCE 2024**

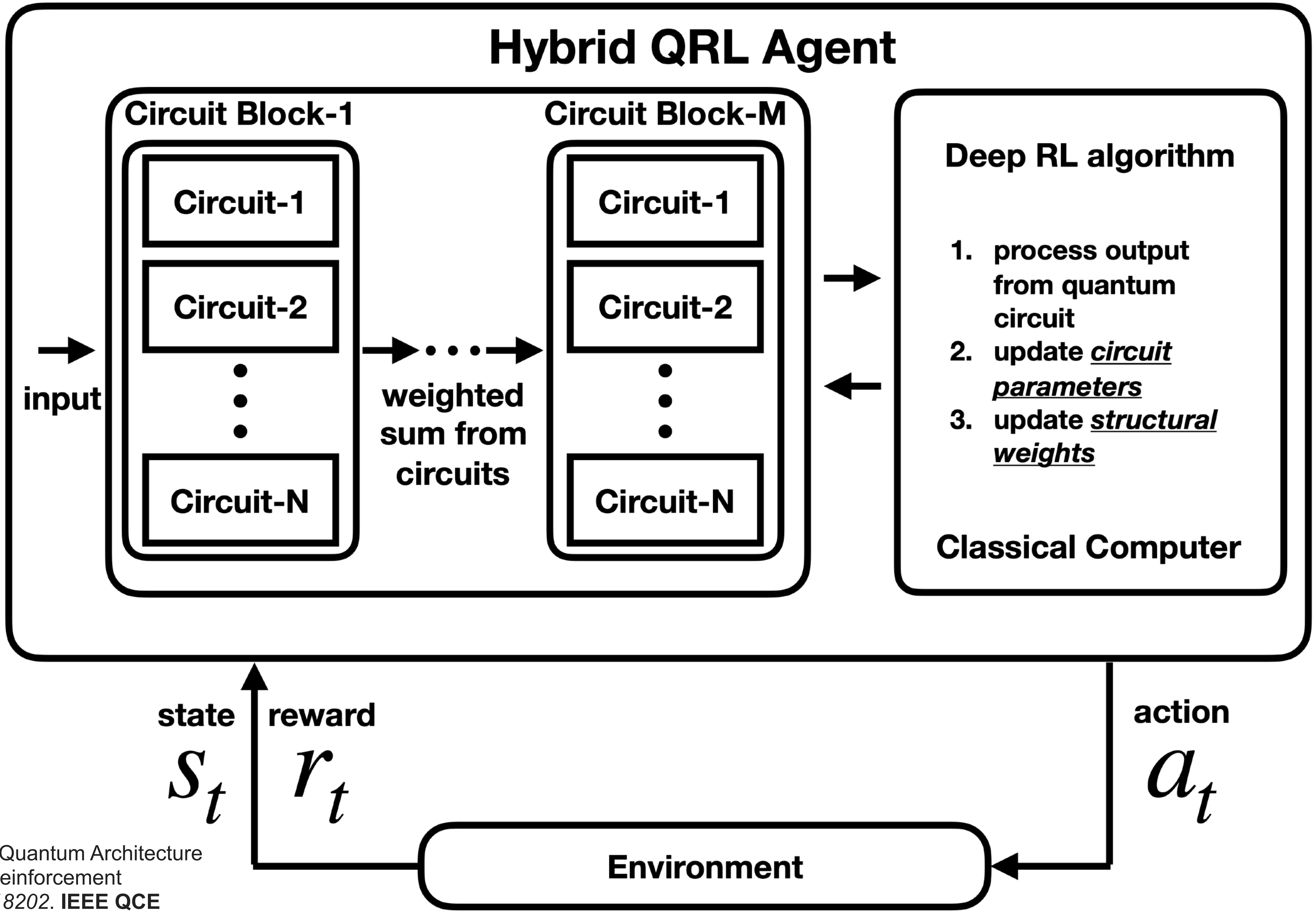
Differentiable Quantum Architecture Search

Connect multiple blocks together!

If there are N possible circuit realization, then the number of total possible paths: N^M

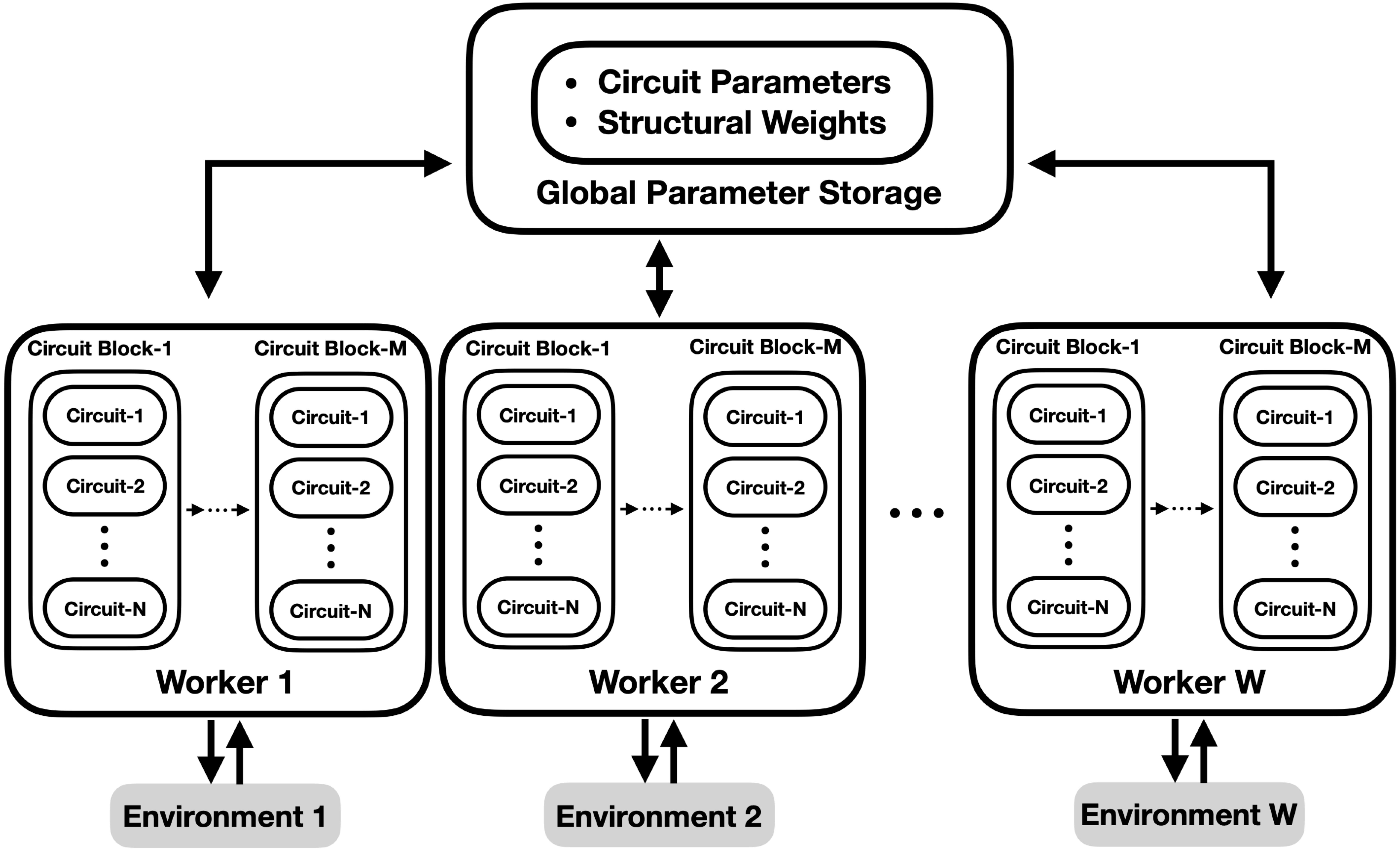


DiffQAS in Quantum RL



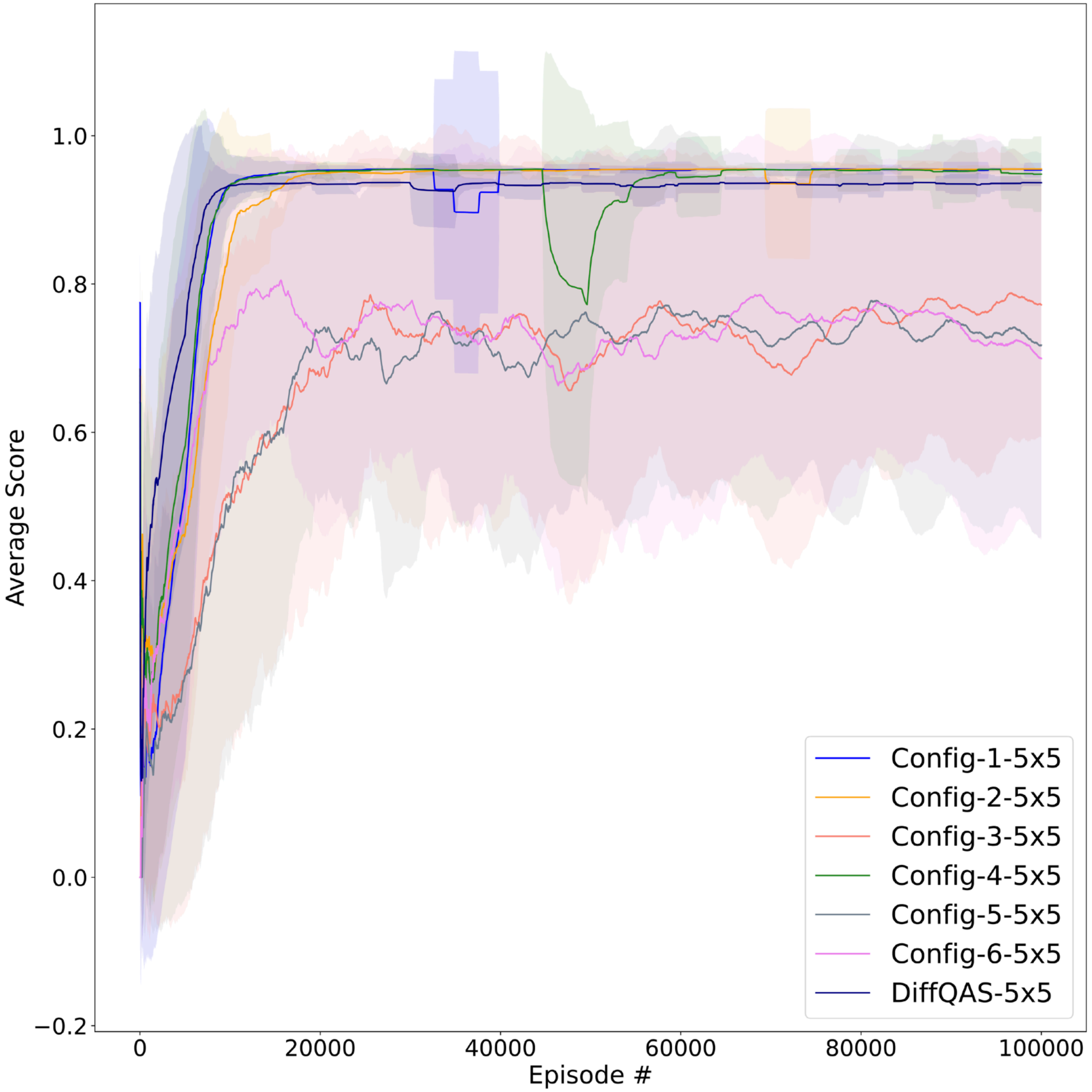
Chen, S. Y. C. (2024). Differentiable Quantum Architecture Search in Asynchronous Quantum Reinforcement Learning. *arXiv preprint arXiv:2407.18202*. **IEEE QCE 2024**

DiffQAS in Asynchronous QRL



Chen, S. Y. C. (2024). Differentiable Quantum Architecture Search in Asynchronous Quantum Reinforcement Learning. *arXiv preprint arXiv:2407.18202*. **IEEE QCE 2024**

Results-MiniGrid-Empty

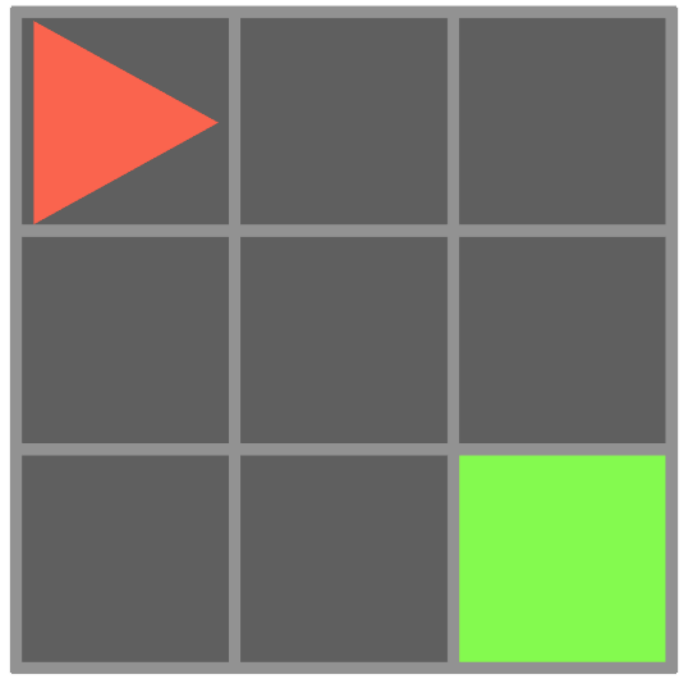


MiniGrid-Empty-5x5

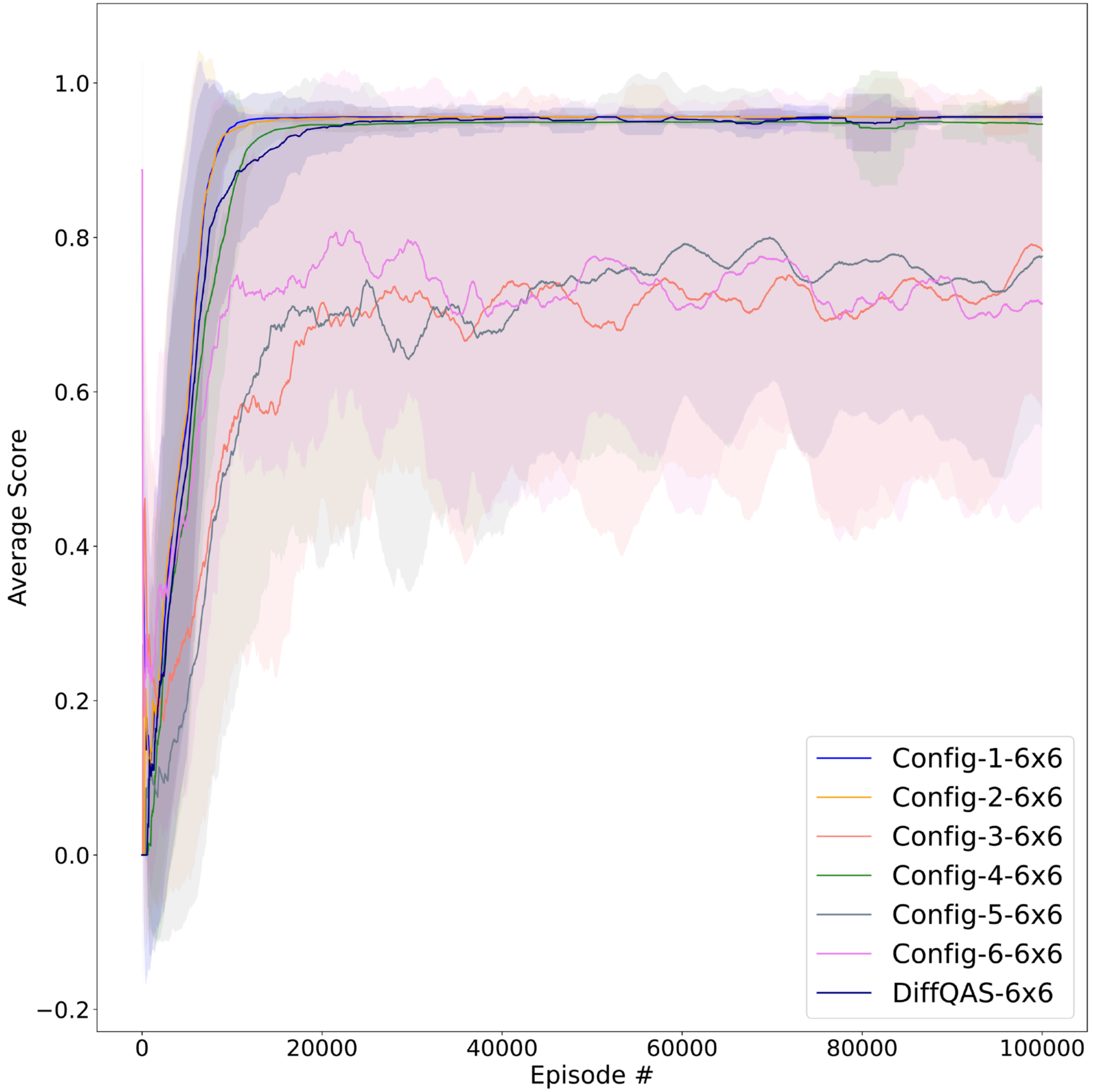
VQC BASELINES.

Component \ VQC config	1	2	3	4	5	6
Encoding	R_y	R_z	R_z	R_y	R_x	R_x
Trainable Rotation Gate	R_y	R_y	R_z	R_z	R_z	R_y

- Performance of DiffQAS is similar to Config-1, 2 and 4.
- Config-3, 5 and 6 fail to reach good performance.



Results-MiniGrid-Empty

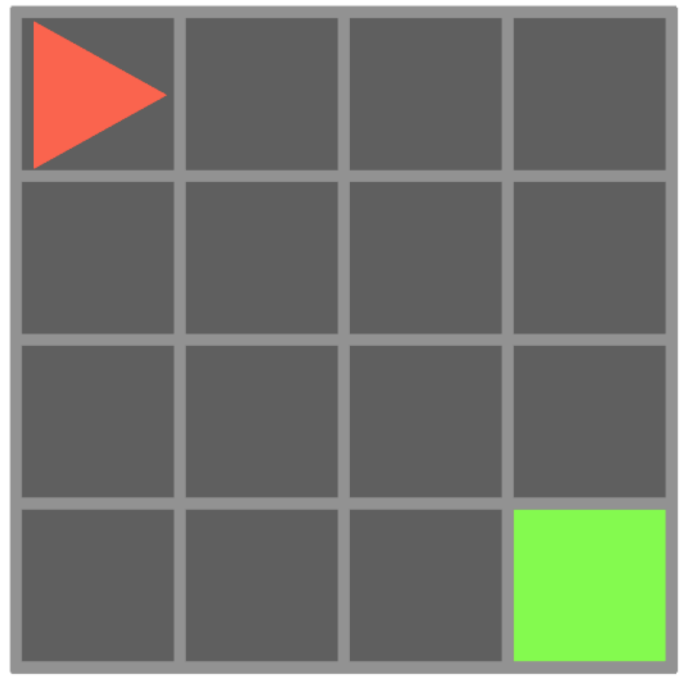


MiniGrid-Empty-6x6

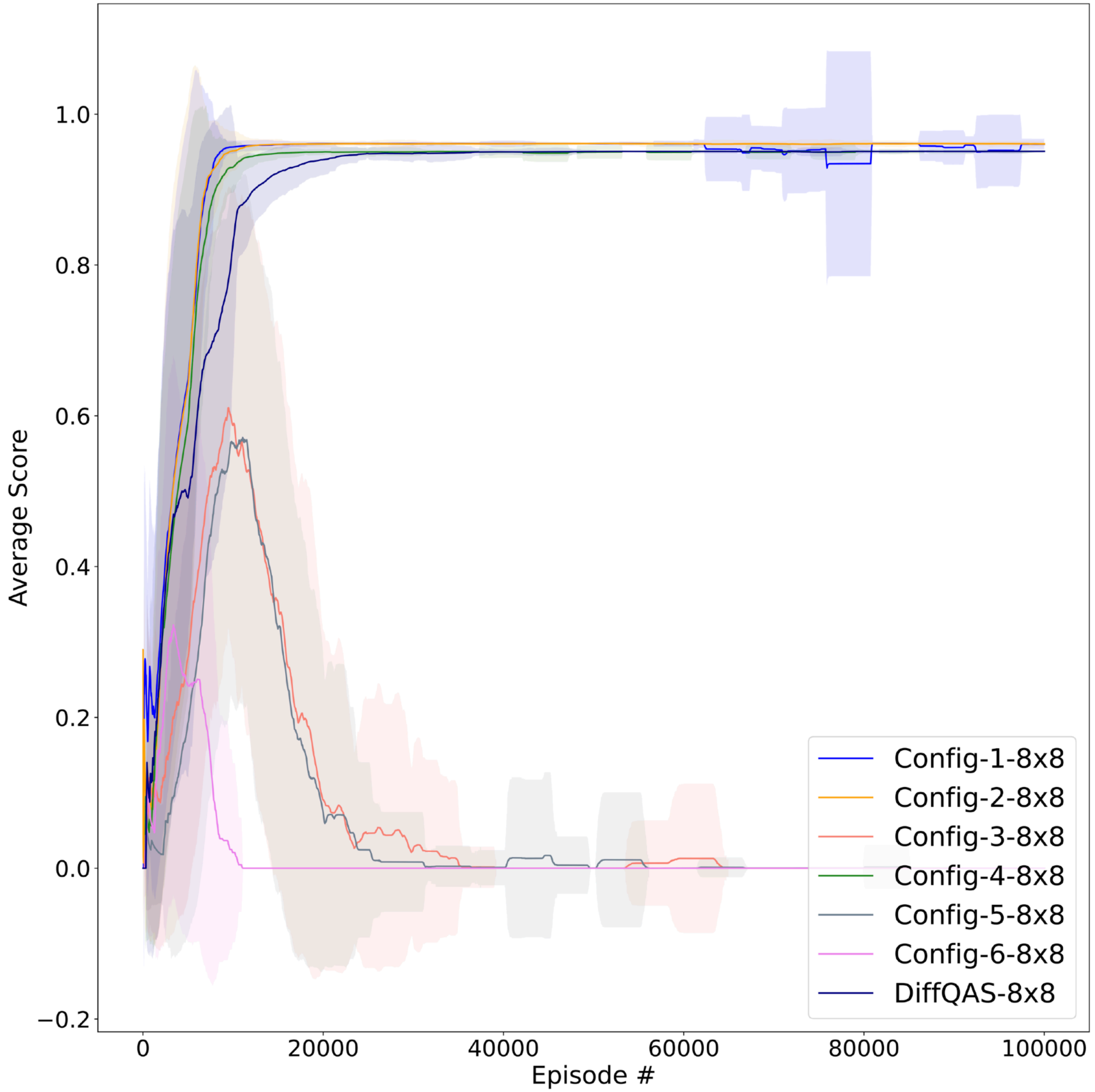
VQC BASELINES.

Component \ VQC config	1	2	3	4	5	6
Encoding	R_y	R_z	R_z	R_y	R_x	R_x
Trainable Rotation Gate	R_y	R_y	R_z	R_z	R_z	R_y

- Performance of DiffQAS is similar to Config-1, 2 and 4.
- Config-3, 5 and 6 fail to reach good performance.



Results-MiniGrid-Empty

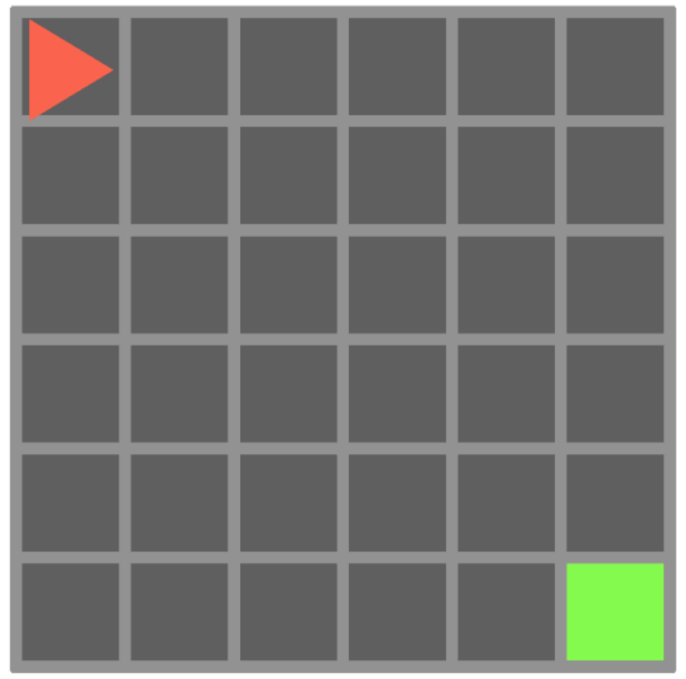


MiniGrid-Empty-8x8

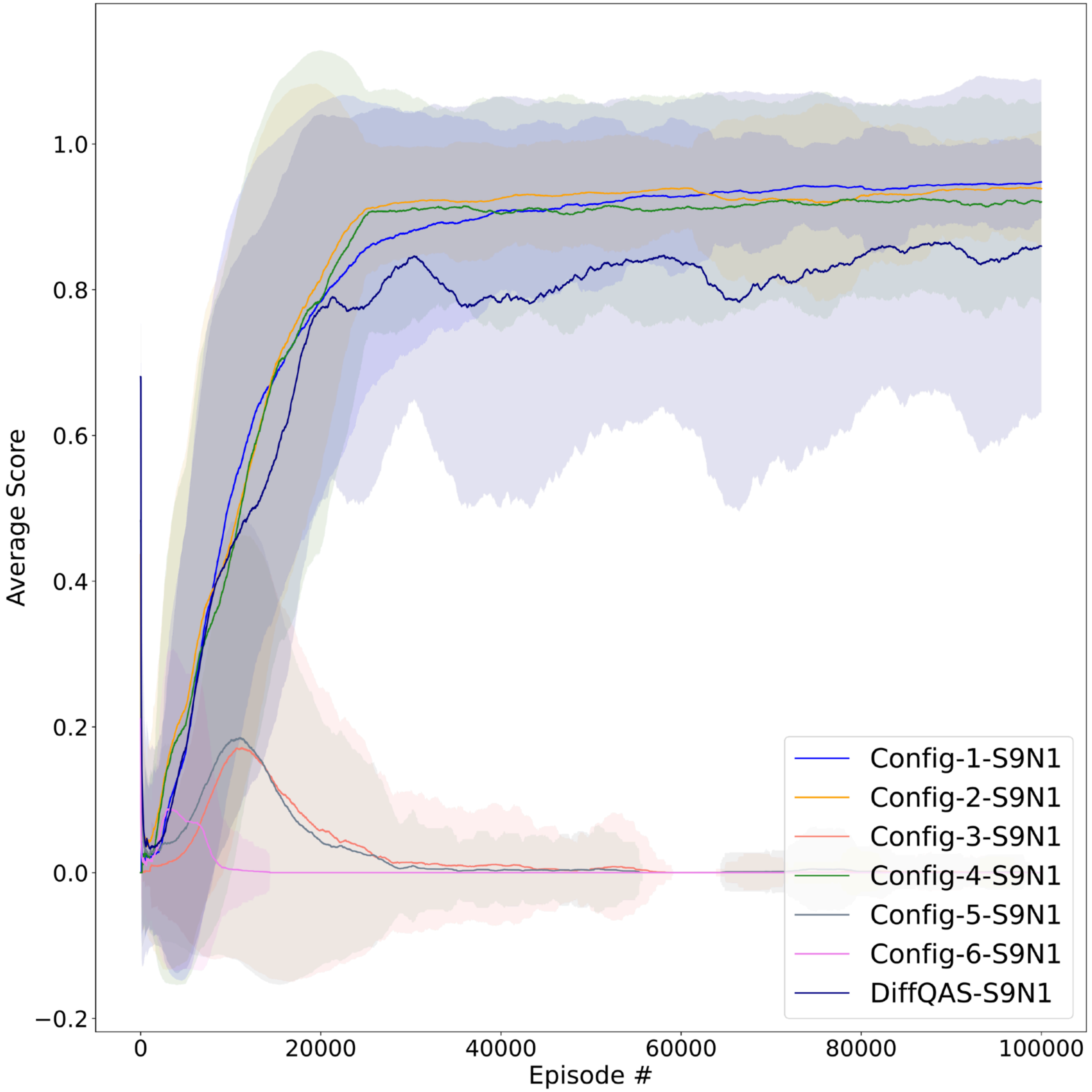
VQC BASELINES.

VQC config / Component	1	2	3	4	5	6
Encoding	R_y	R_z	R_z	R_y	R_x	R_x
Trainable Rotation Gate	R_y	R_y	R_z	R_z	R_z	R_y

- Performance of DiffQAS is similar to Config-1, 2 and 4.
- Config-3, 5 and 6 fail to learn the policy at all.



Results-MiniGrid-SImpleCrossing

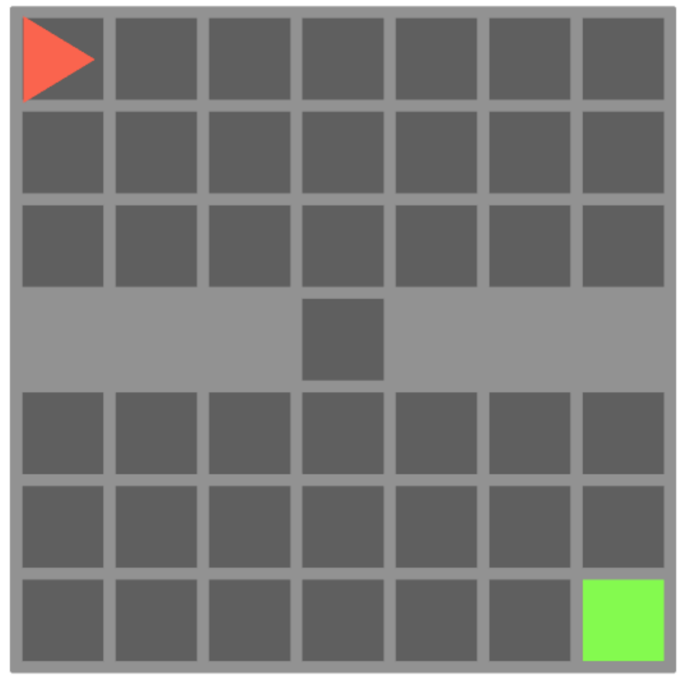


MiniGrid-SimpleCrossing-S9N1

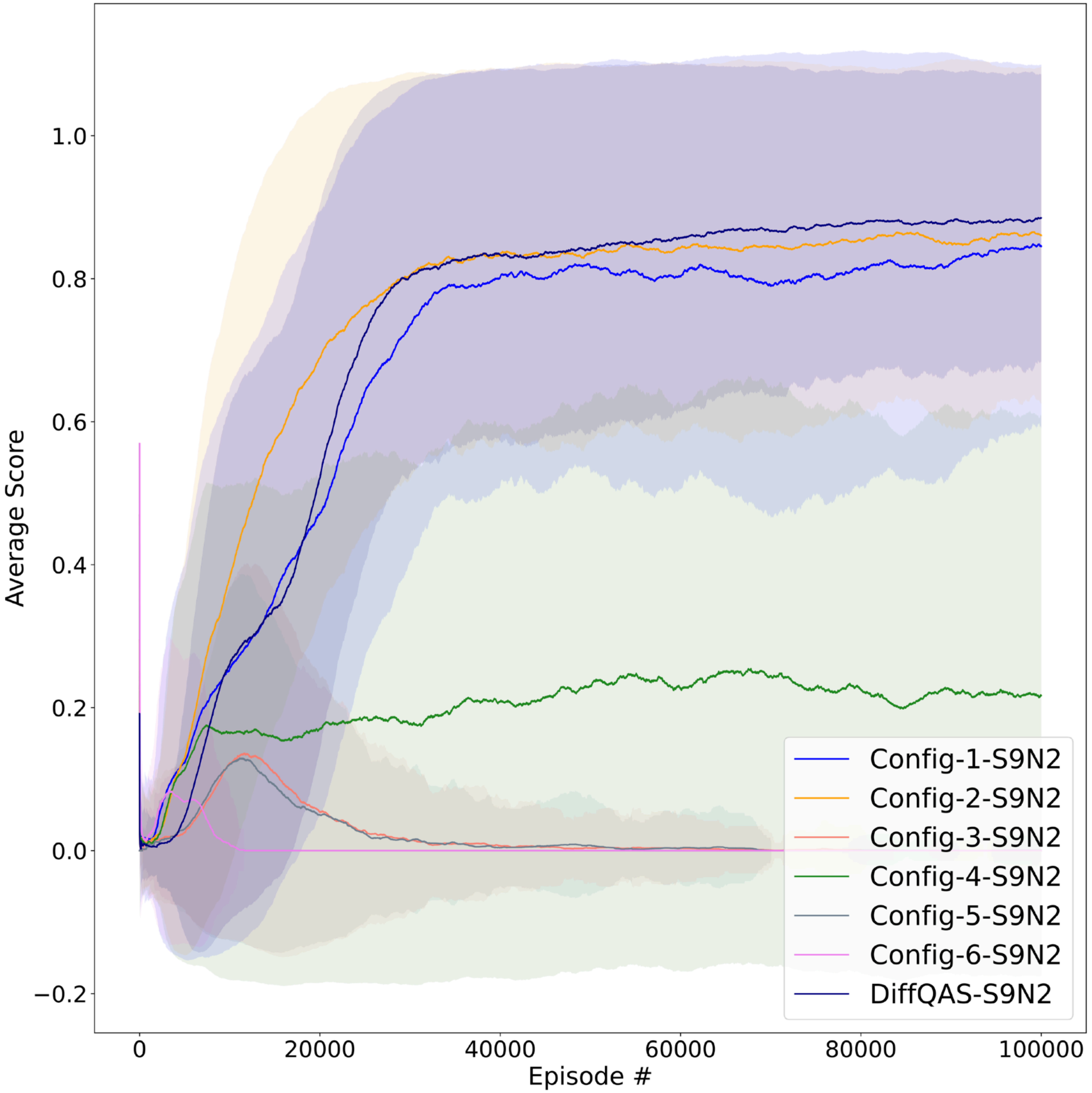
VQC BASELINES.

VQC config / Component	1	2	3	4	5	6
Encoding	R_y	R_z	R_z	R_y	R_x	R_x
Trainable Rotation Gate	R_y	R_y	R_z	R_z	R_z	R_y

- Performance of DiffQAS is close to Config-1, 2 and 4.
- Config-3, 5 and 6 fail to learn the policy at all.



Results-MiniGrid-SImpleCrossing

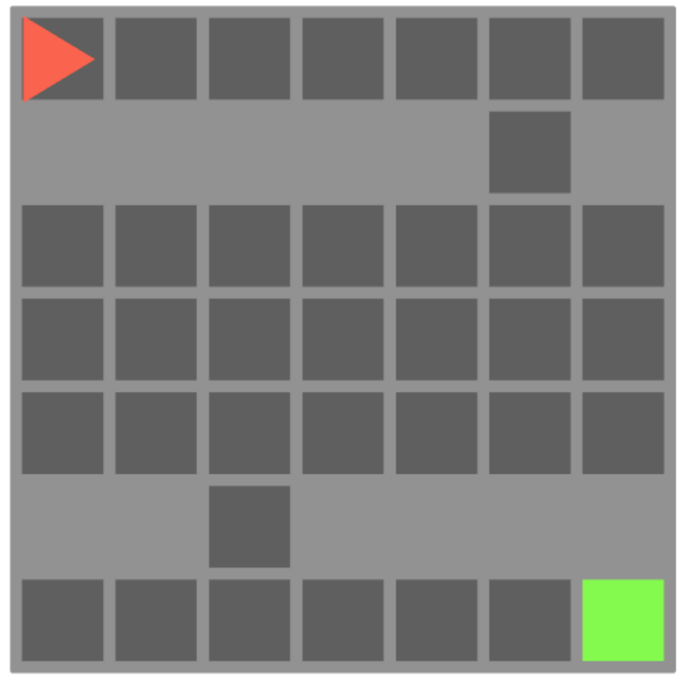


MiniGrid-SimpleCrossing-S9N2

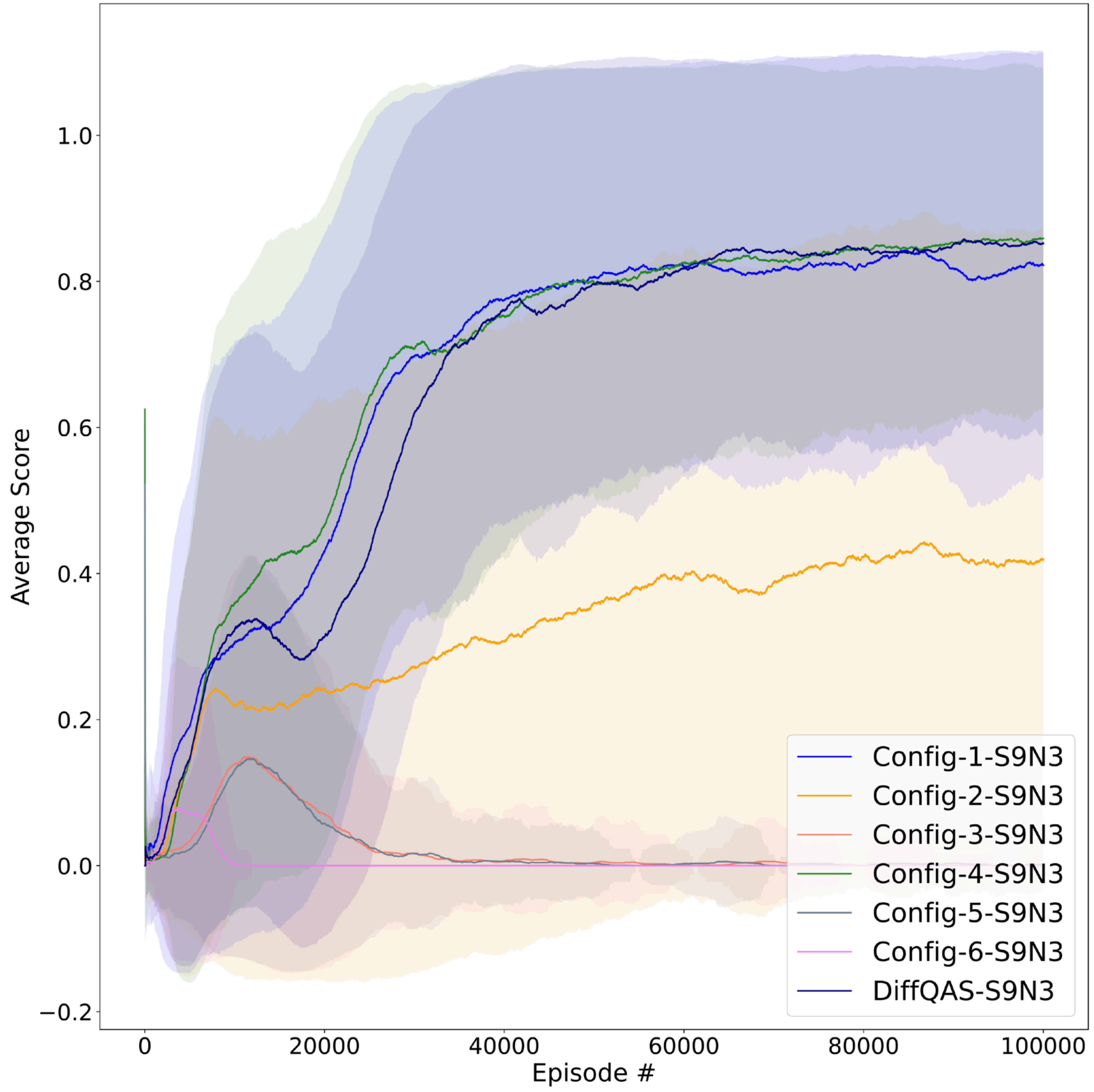
VQC BASELINES.

Component \ VQC config	1	2	3	4	5	6
Encoding	R_y	R_z	R_z	R_y	R_x	R_x
Trainable Rotation Gate	R_y	R_y	R_z	R_z	R_z	R_y

- Performance of DiffQAS is close to Config-1 and 2.
- Config-4 fails to reach the optimal score.
- Config-3, 5 and 6 fail to learn the policy at all.



Results-MiniGrid-SImpleCrossing

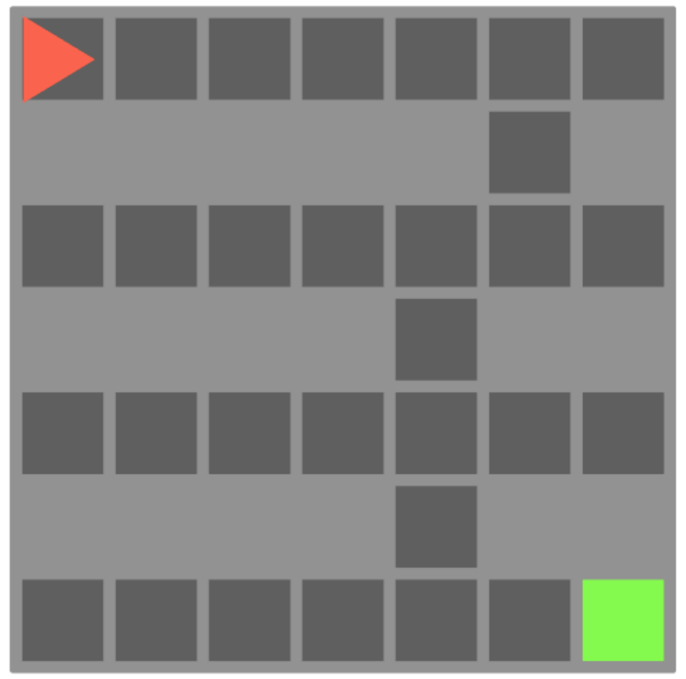


MiniGrid-SimpleCrossing-S9N3

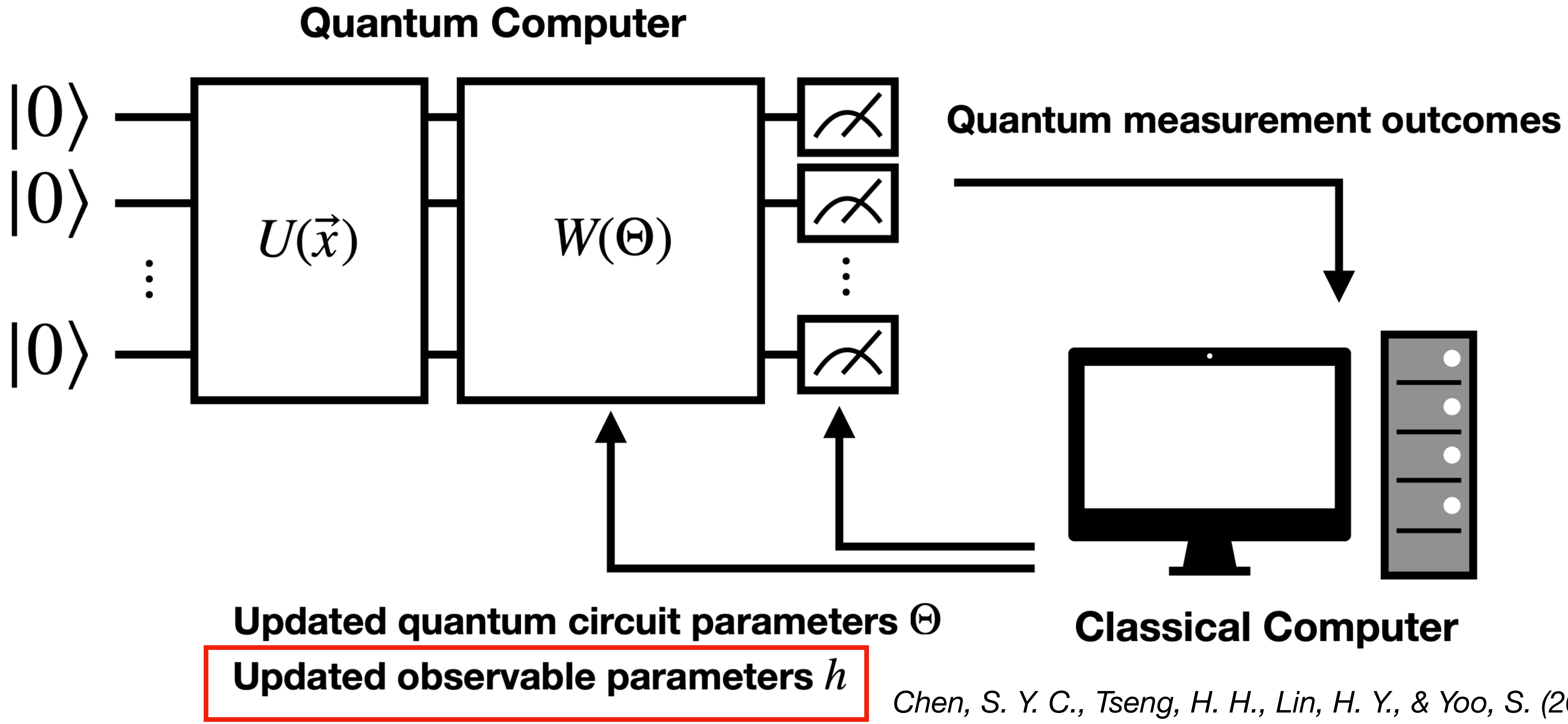
VQC BASELINES.

Component \ VQC config	1	2	3	4	5	6
Encoding	R_y	R_z	R_z	R_y	R_x	R_x
Trainable Rotation Gate	R_y	R_y	R_z	R_z	R_z	R_y

- Performance of DiffQAS is close to Config-1 and 4.
- Config-2 fails to reach the optimal score.
- Config-3, 5 and 6 fail to learn the policy at all.

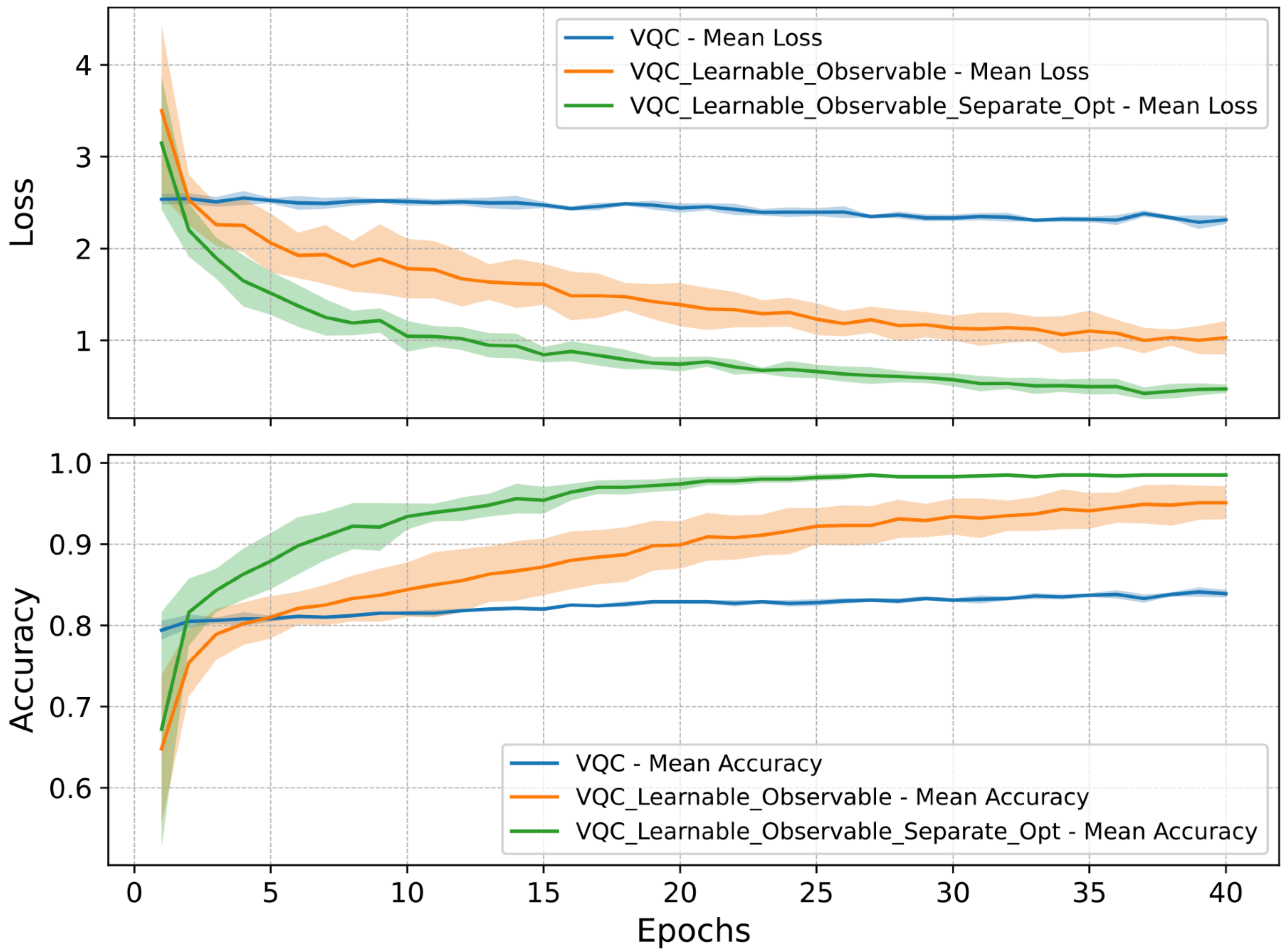


Learning to Measure

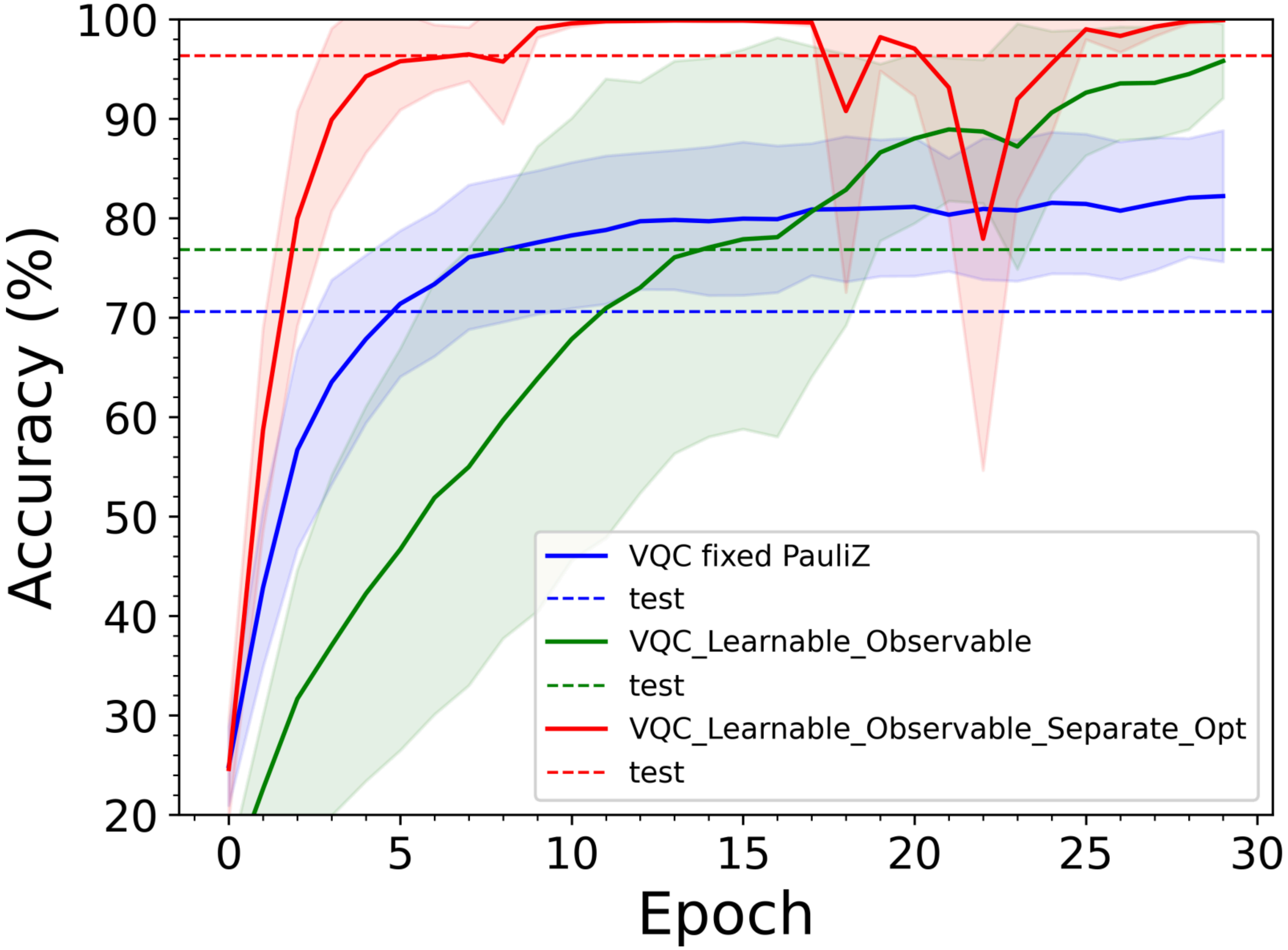


Chen, S. Y. C., Tseng, H. H., Lin, H. Y., & Yoo, S. (2025). Learning to Measure Quantum Neural Networks. arXiv preprint arXiv:2501.05663.

Learning to Measure



Make_Moons Data



VCTK Speaker Recognition Task

Chen, S. Y. C., Tseng, H. H., Lin, H. Y., & Yoo, S. (2025). Learning to Measure Quantum Neural Networks. arXiv preprint arXiv:2501.05663.

- Fundamentals of Quantum Computing
- Hybrid Quantum-Classical Paradigm
- Variational Quantum Circuits (a.k.a Parameterized Quantum Circuits)
- Applications
- Machine Learning for Quantum Machine Learning Model Design
- **Challenges in Quantum Machine Learning**
- Conclusion and Outlook

Challenges in Quantum Machine Learning

Noise and Hardware	Barren Plateau	Operating Conditions	Integration with Classical
• Number of qubits • Qubit decoherence • Gate noise	• Vanishing gradients • Limited model sizes	• Near-zero temperature • Difficult control	• Data transfer between quantum and classical computers

- Fundamentals of Quantum Computing
- Hybrid Quantum-Classical Paradigm
- Variational Quantum Circuits (a.k.a Parameterized Quantum Circuits)
- Applications
- Machine Learning for Quantum Machine Learning Model Design
- Challenges in Quantum Machine Learning
- **Conclusion and Outlook**

Conclusion and Outlook

- Quantum Machine Learning models largely depend on the hybrid quantum-classical framework.

- Variational Quantum Circuits (VQC) a.k.a Parameterized Quantum Circuits (PQC) are the building blocks of QML.

- Quantum and components can be connected as a DAG and backpropogation can be applied to trained the whole model in an end-to-end manner.

Conclusion and Outlook

- Quantum Neural Networks (QNN) can be used to build models such as quantum convolutional neural networks (QCNN), quantum long-short-term memory (QLSTM) and other hybrid quantum-classical models.

- Quantum Neural Networks (QNN) can be used to generate parameters for classical neural networks, reducing a large amount of trainable parameters.

- Quantum Neural Networks (QNN) can learn value functions and policy functions in reinforcement learning (RL).

- Evolutionary, RL and differentiable search can be used to find good QML architectures or good quantum measurement methods.

Thank You!

Feel free to reach out:
ycchen1989@ieee.org