Hybrid Quantum Convolutional Neural Network-Aided Pilot Assignment in Cell-Free Massive MIMO Systems

Doan Hieu Nguyen*, Xuan Tung Nguyen*, Seon-Geun Jeong, Trinh Van Chien, Lajos Hanzo, *Life Fellow, IEEE*, Won-Joo Hwang, *Senior Member, IEEE*

***Abstract*—A sophisticated hybrid quantum convolutional neural network (HQCNN) is conceived for handling the pilot assignment task in cell-free massive MIMO systems, while maximizing the total ergodic sum throughput. The existing model-based solutions found in the literature are inefficient and/or computationally demanding. Similarly, conventional deep neural networks may struggle in the face of high-dimensional inputs, require complex architectures, and their convergence is slow due to training numerous hyperparameters. The proposed HQCNN leverages parameterized quantum circuits (PQCs) relying on superposition for enhanced feature extraction. Specifically, we exploit the same PQC across all the convolutional layers for customizing the neural network and for accelerating the convergence. Our numerical results demonstrate that the proposed HQCNN offers a total network throughput close to that of the excessive-complexity exhaustive search and outperforms the state-of-the-art benchmarks.**

***Index Terms*—Cell-free massive MIMO, Pilot Allocation, Quantum Machine Learning**

I. Introduction

Cell-free massive multiple-input multiple-output (MIMO) leads to an innovative wireless communication architecture eliminating cell-boundaries by allowing multiple access points (APs) to jointly serve multiple users over a wide area [1]. It improves both the spatial diversity, and geographic load-balancing, plus boosts the coverage, throughput, and fairness [2]. The users transmit the pilot signals in the uplink to their APs for channel estimation. However, it would be extremely wasteful to have long user-specific pilots, hence they are reused by the co-channel APs, which leads to interference also known as pilot contamination. This phenomenon reduces the quality of channel estimation and degrades the spectral efficiency [3]. Hence, the pilot assignment schemes must be carefully designed for mitigating pilot contamination. We emphasize that the pilot assignment constitutes a challenging combinatorial problem, making the optimal full search computationally prohibitive. In [4], the authors proposed a greedy pilot assignment scheme for maximizing the minimum data rate by mitigating the pilot contamination. Zaher *et al.* [5] introduced a so-called master-AP pilot assignment scheme, where each user identifies its AP having the highest channel gain as the master AP. The pilot signals are then assigned by minimizing the mutual interference at the master APs. A location-based pilot allocation strategy was introduced in [6], which divides the coverage area into grid-based regions and assigns the pilot signals into disjoint subsets. Explicitly, each region is assigned a subset, so that users sharing the same pilot are geographically distant, hence minimizing interference across the network. However, these heuristic methods focus on interference reduction, rather than directly maximizing the total ergodic throughput. In [7], a convolutional neural network (CNN) was shown to perform well in the testing phase by utilizing an exhaustive search to gather data for training labels. However, utilizing large number of trainable parameters makes CNN becomes burdensome for large-scale systems including multiple users and APs.

Deep neural networks (DNN) having numerous hyperparameters usually require a long time for convergence, hence necessitating a new design for supporting large-scale communication systems. Quantum Machine Learning (QML), leveraging Parameterized Quantum Circuits (PQCs) as computational layers in classical models, presents a promising alternative to former approaches due to its unique quantum advantages. These advantages arise from the fundamental properties of qubits and quantum circuits: (1) Superposition, allowing qubits to exist in both 0 and 1 states simultaneously [8]; (2) Tensor product structure, allowing qubits to encode exponentially more states than classical bits [9]; and (3) Unitary evolution, preserving superposition throughout computation in quantum circuits. By exploiting these properties, QML models can harness the superposition to simultaneously process multiple states, theoretically reducing the runtime below those of classical counterparts [8]. Moreover, quantum states are well-suited for handling large-scale datasets, despite utilizing a reduced number of hyperparameters [10]. Recently, QML has attracted substantial research attention, especially in the classification tasks [8], [11]–[13]. Especially, QML has demonstrated superior performance over classical CNNs in multi-label image classification tasks [8], [13]. However, prior studies often overlook critical factors such as the number of trainable parameters in both classical and quantum

This work was supported in part by the Quantum Computing based on Quantum Advantage challenge research(RS-2024-00408613) through the National Research Foundation of Korea(NRF) funded by the Korean government (Ministry of Science and ICT(MSIT)); in part by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT)(RS-2024-00336962); in part by Institute of Information & communications Technology Planning & Evaluation(IITP) under the Artificial Intelligence Convergence Innovation Human Resources Development(IITP-2025-RS-2023-00254177) grant funded by the Korea government(MSIT); and in part by the IITP(Institute of Information & Communications Technology Planning & Evaluation)-ITRC(Information Technology Research Center) grant funded by the Korea government(Ministry of Science and ICT)(IITP-2025-RS-2023-00260098).

Doan Hieu Nguyen, and Seon-Geun Jeong are with the Department of Information Convergence Engineering, Pusan National University, South Korea (e-mail: hieu.nguyendoan@pusan.ac.kr, wjdtjsrms11@pusan.ac.kr). Nguyen Xuan Tung is with Faculty of Interdisciplinary Digital Technology, PHENIKAA University, Yen Nghia, Hanoi, Viet Nam (e-mail: tung.nguyenxuan@phenikaa-uni.edu.vn). Trinh Van Chien is with the School of Information and Communications Technology, Hanoi University of Science and Technology, Vietnam (e-mail: chientv@soict.hust.edu.vn). Lajos Hanzo is with the Department of Electronics and Computer Science, University of Southampton, U.K. (e-mail: lh@ecs.soton.ac.uk). Won-Joo Hwang is with School of Computer Science and Engineering, Digital-X AIoT Research Center, School of Computer Science and Engineering, Pusan National University, South Korea (e-mail: wjhwang@pusan.ac.kr). *(Corresponding author: Won-Joo Hwang)*

*Equal contribution.

components, as well as the influence of noise in quantum environments [13]. These limitations necessitate a carefully designed hybrid quantum-CNN model that accounts for both model complexity and quantum noise.

These finding inspired us to propose an HQCNN for handling with the pilot assignment task of cell-free massive MIMO communication, with the objective of maximizing the total network throughput. In contrast to the layer-specific quantum circuit designs used in [11], we propose to use the same parameterized quantum circuit across all the convolutional layers. This allow us to significantly reduce the number of training parameters and accelerates the convergence during the training phase. The proposed HQCNN is trained using both supervised and unsupervised techniques to analyze models performance from different perspectives. To the best of our knowledge, this is the first study harnessing QML for solving the pilot assignment problem of cell-free massive MIMO communication as a multi-class classification task. Our numerical results demonstrate that the proposed HQCNN converges faster than classical deep learning with a margin of 2% of the global optimum. We also estimate the impacts of noisy environment and scalability of HQCNN in this work.

II. System Model and Problem Formulation

A. System Model

We consider a cell-free massive MIMO system operating in the time division duplexing model. Expliciting, M APs - each equipped with L antennas - jointly serve K single-antenna users randomly distributed in the coverage area. In each coherence interval, the system assigns τ_p orthogonal pilot signals for estimating the propagation channels in the uplink training phase. The binary variable $p_{kt} \in \{0,1\}$ indicates the pilot assignment: $p_{kt} = 1$ if user k occupies the t-th pilot $\boldsymbol{\varphi}_t \in \mathbb{C}^{\tau_p}$, where we have $\|\boldsymbol{\varphi}_t\|^2 = 1$. Otherwise, $p_{kt} = 0$. Let us denote the channel between AP m and user k by $\mathbf{g}_{mk} \in \mathbb{C}^L$. All the elements of $\mathbf{g}_{mk}$ are independent and identically distributed (i.i.d) as $\mathcal{CN}(0, \beta_{mk})$, with β_{mk} representing the large-scale fading (LSF).

1) Uplink Pilot Training

All the users simultaneously transmit the assigned pilot signals to the APs. The signal received at AP m is $\mathbf{Y}_m = \sqrt{\tau_p \rho_p} \sum_{k=1}^{K} \sum_{t=1}^{\tau_p} p_{kt} \mathbf{g}_{mk} \boldsymbol{\varphi}_t^H + \mathbf{W}_m$, where ρ_p is the normalized signal-to-noise ratio (SNR) of each pilot symbol. Furthermore, $\mathbf{W}_m \in \mathbb{C}^{L \times \tau_p}$ represents the additive white Gaussian noise (AWGN) whose elements are i.i.d $\mathcal{CN}(0,1)$. Similar to [1], the channel estimate $\hat{\mathbf{g}}_{mk}$ is defined by exploiting the linear minimum mean square error (LMMSE) estimator at the core network as

$$\hat{\mathbf{g}}_{mk} = \sqrt{\tau_p \rho_p} c_{mk} \sum_{j=1}^{K} \sum_{t'=1}^{\tau_p} \sum_{t=1}^{\tau_p} \mathbf{g}_{mk} p_{jt'} \boldsymbol{\varphi}_{t'}^H p_{kt} \boldsymbol{\varphi}_t + \mathbf{W}_m c_{mk} \sum_{t=1}^{\tau_p} p_{kt} \boldsymbol{\varphi}_t, \quad (1)$$

where $c_{mk} \triangleq \frac{\sqrt{\tau_p \rho_p} \beta_{mk}}{\tau_p \rho_p \sum_{j=1}^{K} \sum_{t'=1}^{\tau_p} \sum_{t=1}^{\tau_p} \beta_{mj} |p_{jt'} \boldsymbol{\varphi}_{t'}^H p_{kt} \boldsymbol{\varphi}_t| + 1}$. The second moment of the channel estimate $\hat{\mathbf{g}}_{mk}$ is expressed in closed form as $\gamma_{mk} \triangleq \mathbb{E}\left\{|\hat{\mathbf{g}}_{mk}|^2\right\} = \sqrt{\tau_p \rho_p} \beta_{mk} c_{mk}$.

2) Downlink Data Transmission

The APs apply the classic conjugate beamforming technique for the transmit precoding (TPC) of signals by exploiting the channel estimates. In particular, the signal transmitted from AP m to user k is $\mathbf{x}_m = \sqrt{\rho_d} \sum_{k=1}^{K} \sqrt{\eta_{mk}} \hat{\mathbf{g}}_{mk} q_k$, where ρ_d is the maximum normalized transmit power at each AP; q_k is the symbol intended for user k with $\mathbb{E}\{|q_k|^2\} = 1$; and η_{mk} is the control coefficient satisfying the power budget constraint of $L \sum_{k=1}^{K} \eta_{mk} \gamma_{mk} \leq 1$. The signal received by user k is

$$\begin{aligned} r_k &= \sum_{m=1}^{M} \mathbf{g}_{mk}^H \mathbf{x}_m + w_k \\ &= \sqrt{\rho_d} \sum_{m=1}^{M} \sum_{j=1}^{K} \sqrt{\eta_{mk}} \mathbf{g}_{mk} \hat{\mathbf{g}}_{mj}^* q_j + w_k, \end{aligned} \quad (2)$$

where w_k is the AWGN at user k distributed as $\mathcal{CN}(0,1)$. From (2), one can obtain the achievable downlink ergodic throughput of user k as formulated in Lemma 1.

Lemma 1. *The ergodic throughput of user k is expressed in closed form in* (3) *with B being the system bandwidth.*

Proof. First, a lower bound on the downlink channel capacity is attained by assuming that the APs apply linear precoding and then forget the TPC vectors when computing the throughput. Then, the closed-form expression is obtained by formulating the moments of Gaussian random variables. □

The ergodic throughput expressed in (3) only depends on the channel statistics so that the formulated may rely upon a specific pilot assignment policy for a long period of time.

B. Problem Formulation

We aim for maximizing the total ergodic rate by optimizing the pilot assignment, which is mathematically formulated as

$$\begin{aligned} \underset{\{p_{kt}\}}{\text{maximize}} \quad & \sum_{k=1}^{K} R_{dk} && \text{(4a)} \\ \text{subject to} \quad & \sum_{t=1}^{\tau_p} p_{kt} = 1, \forall k, && \text{(4b)} \\ & p_{kt} \in \{0,1\}, \forall k, t, && \text{(4c)} \end{aligned}$$

where the constraint (4b) ensures that each user only has a single pilot signal. The constraint (4c) makes Problem (4) combinatorial. It is computationally challenging to obtain the optimal solution as the system serves many users. Even though an exhaustive search would indeed find the globally optimal solution, it is excessively complex for large-scale systems. By contrast, heuristic methods are capable of reducing the complexity at the cost of performance reduction. Against this backdrop, we design a hybrid quantum neural network to solve Problem (4) by maximizing the total ergodic throughput.

III. Proposed Model

This section presents the proposed HQCNN consisting of three main parts: pre-processing, quantum convolutional neural network (QCNN), and post-processing as illustrated in Fig. 1.

A. Design of HQCNN Model

1) Pre-processing layer

To utilize the QML, we must transform the classical data into the quantum state in a higher-dimensional Hilbert space. A fully connected (FC) and a quantum embedding layer (QEL) are first deployed. Therein, the linear layer adjusts the dimension of classical input data to the n_{in}-dimensional aggregated vectors, where n_{in} is the predefined number of qubits in the quantum circuits. In this work, we adopt the angle embedding method over basis and amplitude embedding techniques. The

$$R_{dk} = B\log_2\left(1+\frac{\rho_d L^2\left(\sum_{m=1}^{M}\sqrt{\eta_{mk}}\gamma_{mk}\right)^2}{\rho_d L^2\sum_{j=1,j\neq k}^{K}\left(\sum_{m=1}^{M}\sqrt{\eta_{mj}}\gamma_{mj}\frac{\beta_{mk}}{\beta_{mj}}\right)^2\left|\sum_{t=1}^{\tau_p}\sum_{t'=1}^{\tau_p}p_{jt}\boldsymbol{\varphi}_t^H p_{kt'}\boldsymbol{\varphi}_{t'}\right|+\rho_d L\sum_{j=1}^{K}\sum_{m\in\mathcal{M}(k)}^{M}\eta_{mj}\gamma_{mj}\beta_{mk}+1}\right) \quad (3)$$

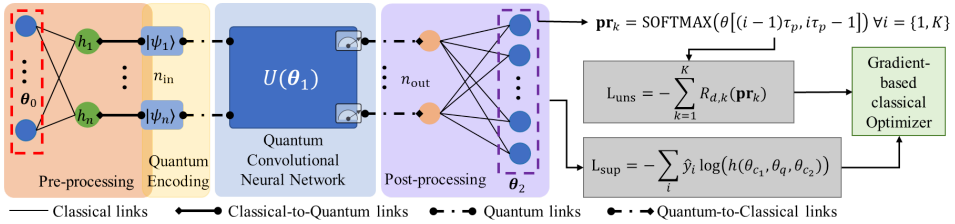

Fig. 1: Architecture of the proposed HQCNN model: the pre-processing layer for embedding classical data to quantum state; QCNN for processing quantum states; the post-processing layer for converting outcomes of QCNN.

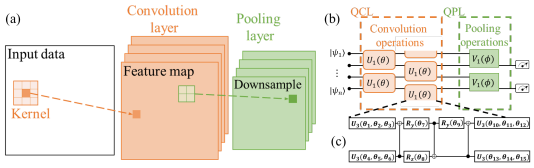

Fig. 2: (a) Illustration of a CNN layer; (b) QCNN layer; and (c) PQC design.

rationale is that basis embedding is limited to binary inputs, while amplitude embedding demands deeper quantum circuits, thereby increasing complexity and vulnerability to quantum noise [14]. In contrast, angle embedding produces shorter output vectors, which helps reduce the number of trainable parameters in the pre-processing layer. These advantages contribute to lower circuit complexity and a faster convergence rate during training [14]. Note that the initial quantum state $|\psi\rangle$ is represented as $|\psi\rangle = \bigotimes_{i=1}^{n_{\text{in}}}(\cos(h_i)|0\rangle + \sin(h_i)|1\rangle)$, where h_i is the i-th element of $\mathbf{h} = \boldsymbol{\theta_1}^T\mathbf{x} + \mathbf{b}$ in which $\mathbf{x}$ is the classical input vector; θ_1, and $\mathbf{b}$ are weighting matrix and bias vector of the linear layer, respectively.

2) Quantum convolution layer

A QCNN containts parameterized quantum circuits, each similar to a filter which process a quantum state as its input and outputs a vector of expectation values. As reported in [11], a QCNN is capable of adopting the core principles of convolutional neural networks, including the convolutional and pooling layers for feature mapping and dimension reduction, as shown in Fig. 2a. By harnessing this, a kernel is slid over the input data, performing element-wise operations in order to transform data from the input to the output in CNN. By exploiting parameter-sharing across the layers, the proposed HQCNN is capable of significantly reducing the number of training parameters with the aid of a series of quantum convolution layers (QCLs) and quantum pooling layers (QPLs). Fig. 2b illustrates example of one QCNN layer. The QCLs employ PQCs to aggregate quantum information from the adjacent qubits, while the QPLs reduce the network size by measuring half of the qubits. These layers are repeated until the main features of information are successfully extracted.

Again, in the previous QCNN design of [11], each QCL exploited identical PQC structures but along with different parameter sets. This approach required a total of $T\sum_{i=1}^{\log_2(n)}2^i$ parameters if only a single outcome is measured, where T is the number of parameters per PQC. We therefore propose a novel QCL design inspired by the concept of CNN kernel parameter-sharing [7] for reducing the number of learnable parameters. Using the same set of parameters for all the PQCs at each layer reduces the total number of parameters to $T\log_2(n_0)$, where n_0 is the initial number of qubits. The exponential parameter reduction capability of our design significantly reduces the computational cost, making the quantum system more efficient. The PQC structure of our neural network design is adopted from the structure of [11], relying on the detailed illustration seen in Fig. 2c.

3) Post-processing layer

Finally, a fully connected layer is used at the output for processing the collapsed classical outcomes of the QCNN. The post-processing layer takes the n^{out}-outcome expectation values of the QCNN as its input. We view the pilot assignment problem in the scope of a multi-class classification problem with the output of a $(K \times \tau_p)$ matrix.

B. Training Procedure

1) Inference process

The proposed HQCNN model performs the pilot assignment by only considering the LSF coefficients hosted by the matrix $\boldsymbol{\beta} \in \mathbb{C}^{M\times K}$, since the achievable data rate of each user predominantly relies on its LSF with respect to other APs. The input layer of the proposed hybrid neural network maps the matrix $\boldsymbol{\beta}$ to the initial quantum state. In QCNN layer, the output state of the l-th layer is

$$|\psi_i(\boldsymbol{\theta}_i)\rangle\langle\psi_i(\boldsymbol{\theta}_i)| = \text{Tr}_{O_i}\left(U_i(\boldsymbol{\theta}_i)|\psi_{i-1}\rangle\langle\psi_{i-1}|U_i(\boldsymbol{\theta}_i)^\dagger\right), \quad (5)$$

where $\text{Tr}_{O_i}(\cdot)$ represents the partial trace operation over the subsystem O_i; $U_i(\cdot)$ is the parameterized unitary operation combining the quantum convolution and pooling operations; and $\boldsymbol{\theta}_i$ denotes the parameters of the PQC in the i-th quantum convolutional layer. After processing by all the QCNN layers, the output is calculated as in [15], given by

$$\langle\mathcal{M}_i\rangle = \langle\boldsymbol{\psi}|U(\theta)^\dagger A_i U(\theta)|\boldsymbol{\psi}\rangle, \quad (6)$$

where $|\boldsymbol{\psi}\rangle$ is the final quantum state, $U(\boldsymbol{\theta})$ is the overall unitary operation composed of all the quantum layers, and O_i are the observables, typically chosen as the Pauli operators. In this study, we use the Pauli-Z operators for the measurement layer. The final outcomes are estimated as expectation values $\langle\mathcal{M}_i\rangle$ to avoid probabilistic measure noise, serving as the input of the classical post-processing layer. In the post-processing

$$\mathrm{LF}_{\mathrm{uns}}^{(k)} = -B\sum_{k=1}^{K}\mathbb{E}\left\{\log_2\left(1+\frac{\rho_d L^2\left(\sum_{m=1}^{M}\sqrt{\eta_{mk}}\gamma_{mk}\right)^2}{\rho_d L^2\sum_{j=1,j\neq k}\left(\sum_{m=1}^{M}\sqrt{\eta_{mj}}\gamma_{mj}\frac{\beta_{mk}}{\beta_{mj}}\mathbf{q}_k^H\mathbf{q}_j\right)^2+\rho_d L\sum_{j=1}^{K}\sum_{m=1}^{M}\eta_{mj}\gamma_{mj}\beta_{mk}+1}\right)\right\}. \tag{8}$$

layer, we apply a softmax function to each row of the $(K\times\tau_p)$ output matrix to compute the pilot selection probabilities.

For supervised learning, the pilot assignment orders obtained from [5] are exploited as labels. Users select the specific pilot signals with the highest probability. In this framework, the cross-entropy loss function is employed as

$$\mathrm{LF}_{\mathrm{sup}} = -\left\|\sum_{k=1}^{K}\sum_{i=1}^{\tau_p}\mathbf{y}_{ki}\log(h(\boldsymbol{\beta};\boldsymbol{\theta})_{ki})\right\|_1, \tag{7}$$

where $h(\boldsymbol{\beta};\boldsymbol{\theta})$ is the prediction probability and $\boldsymbol{\theta}$ represents the trainable parameters. Since both the gradient-based optimizers such as the classical backpropagation and quantum parameter shift rule [10] are based on the gradient descent method, both classical and quantum machine learning models minimize the loss function iteratively. Hence, if the unsupervised loss function is reformulated as the negative version of the sum-rate function influenced by the pilot probabilities, as in (8), where $\mathbf{q}_k \in \mathbb{R}^{\tau_p}$ denotes the probability of the pilot selection for user k, then minimizing the loss function process is similar to maximizing the total throughput function, in line with the objective of Problem (4).

2) Updating process

In the CNN model, the backpropagation method updates the weights in kernel by computing gradients of the loss function with respect to the parameters and adjusting them using the gradient descent method. Unlike the CNN model having visible activations in hidden layers, the quantum states in QCNNs are not measurable until they collapse, preventing QCNNs from computing gradients via the chain rule. Hence, the QCNNs utilize another gradient-based approach, namely the parameter shift method of [10] to estimate the natural quantum gradient, given as

$$\frac{\partial\langle\mathcal{M}\rangle(\theta)}{\partial\theta} = \frac{\langle\mathcal{M}\rangle(\theta+s)+\langle\mathcal{M}\rangle(\theta-s)}{2\sin(s)}, \tag{9}$$

where s is the parameter shift value. For an arbitrary quantum state $|\psi\rangle = a|0\rangle + b|1\rangle$, the outcome of a QCNN is expressed by a single-qubit unitary matrix as

$$\begin{pmatrix} e^{-i\alpha}\cos(\varphi) & -e^{i\beta}\sin(\varphi) \\ e^{-i\beta}\sin(\varphi) & e^{i\alpha}\cos(\varphi)\end{pmatrix}, \tag{10}$$

where α is the phase, while β and φ define the angle of general rotation. Due to the periodic nature of rotation functions, the shift value is selected as $(\pi/2)$ for simplicity. Hence, the gradient is formulated as

$$\frac{\partial\langle\mathcal{M}\rangle(\theta)}{\partial\theta} = \frac{1}{2}\left(\langle\mathcal{M}\rangle\left(\theta+\frac{\pi}{2}\right)+\langle\mathcal{M}\rangle\left(\theta-\frac{\pi}{2}\right)\right). \tag{11}$$

After getting the gradients, new parameters are calculated similarly to the popular gradient-descent methods, as

$$\theta_{\mathrm{new}} = \theta_{\mathrm{old}} - \epsilon\frac{\partial\langle\mathcal{M}\rangle(\theta)}{\partial\theta}. \tag{12}$$

Instead of requiring backpropagation, the quantum circuit has to be estimated twice along with $\theta\pm\pi/2$. Hence, by significantly reducing the number of trainable parameters, the computational complexity in our design of the updating process is exponentially lower than those of the benchmark design in [11]. Based on, the construction and training process conceived, the proposed HQCNN can perform the pilot assignment, as formulated in Lemma 2.

TABLE I: Parameters analysis.

Model	Number of parameters
HQCNN	$O\left(nK(M+\tau_p)+15\right)$ (quantum parameters))
HQCNN [11]	$O\left(nK(M+\tau_p)+15n\right)$ (quantum parameters))
CNN	$O\left(k^2\left(\sum_{i=1}^{F}C_{\mathrm{in},i}C_{\mathrm{out},i}+M\tau_pK^2\right)\right)$
MLP	$O\left(nK(M+\tau_p)+n^2\right)$

Lemma 2. *The proposed HQCNN model relying on $n\in\mathbb{N}$ number of qubits in the PQC, and denoted as $\boldsymbol{\theta}^*$, is capable of solving Problem* (4a) *for cell-free massive MIMO systems configured by the triplet $\{M,K,\tau_p\}$ via approximating the mapping from the large-scale fading coefficients to the pilot selection probabilities.*

Proof. The proof is available in the appendix. □

IV. Numerical Results

Let us now evaluate the performance of the proposed HQCNN under both supervised and unsupervised training frameworks. For a data-driven scenario, the proposed HQCNN model is compared to both the MLP as well as to the light- and heavy-CNN models, which are inspired by the CNN of [7], regarding the convergence rate of the supervised learning and computational efficiency of the unsupervised framework. Table I analyses the complexity order of each model. While the architecture of both our model and of the MLP rely on M, K, τ_p and n, the structure of CNN depends on other factors like the kernel size (in pixels) k^2, the number of convolutional layers F, and the number of input and output channels in each convolution layer C_{in}, C_{out}. Both CNNs use 3×3 kernels in two convolutional layers, but the number of output channels are $(8,16)$ and $(32,64)$ in light-CNN and heavy-CNN, respectively. In this work, 8 qubits are utilized in the quantum model, the number of trainable parameters in HQCNN approximately equal to those of MLP and light-CNN, while much lower than those of heavy-CNN, as a benefit of quantum parameter-sharing harnessed for reducing the computational complexity. The model-based approaches used for our comparison involve the random (RPA), Greedy (GPA) [4], Master-AP (MPA) [5], location-based (LPA) [6] and exhaustive search (EPAS).

We consider a one-square-kilometer area having the system configurations as follows: pilot transmit power 100 [mW]; access point downlink transmit power 200 [mW]; bandwidth 20 [MHz]; carrier frequency $f = 1.9$ [GHz]; noise figure of 9 [dB]. The LSF coefficient β_{mk} models the path-loss and shadow fading as follows $\beta_{mk} = \mathrm{PL}_{mk}\cdot10^{\frac{\sigma_{sh}z_{mk}}{10}}$, where PL_{mk} represents the path-loss, and $10^{\frac{\sigma_{sh}z_{mk}}{10}}$ represents the shadow fading having the standard deviation σ_{sh}, and $z_{mk}\sim\mathcal{N}(0,1)$. The path-loss is represented by the three-

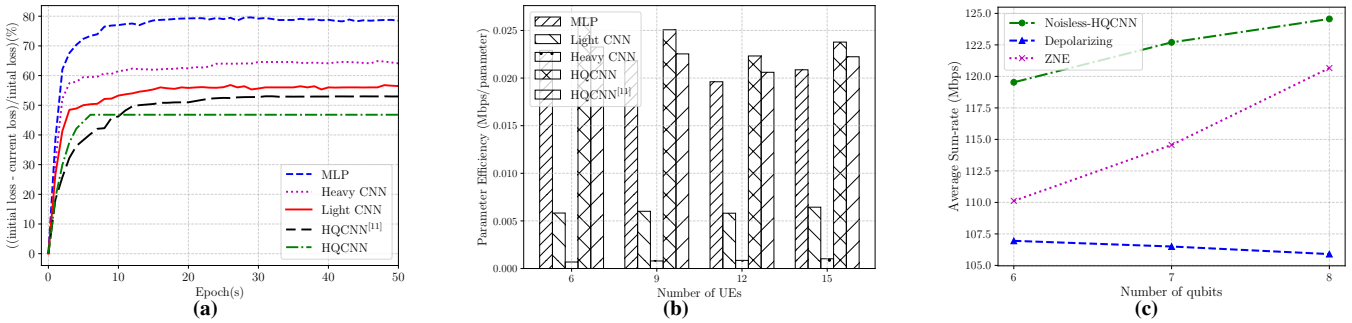

Fig. 3: (a) Proportion of training loss reduction relative to the initial loss value versus the training epochs (supervised); (b) Parameter efficiency of learning-based benchmarks (unsupervised); and (c) the total ergodic throughput with $(M, L, K, \tau_p) = (40, 2, 20, 10)$ in noiseless- and noise-simulation (unsupervised).

slope model as

$$\mathrm{PL}_{mk} = \begin{cases} -L - 35\log_{10}(d_{mk}), & \text{if } d_{mk} > d_1 \\ -L - 15\log_{10}(d_1) - 20\log_{10}(d_{mk}), & \text{if } d_0 < d_{mk} \leq d_1 \\ -L - 15\log_{10}(d_1) - 20\log_{10}(d_0), & \text{if } d_{mk} \leq d_0. \end{cases}$$

This setting was used in [4], along with similar selections of $L = 141$, $d_1 = 50$m, $d_0 = 10$m and $\sigma_{sh} = 8$dB. The power control coefficients are uniformly set as $\eta_{mk} = 1/L\sum_{k=1}^{K}\gamma_{mk}$ for all the APs.

A. Performance of Supervised Learning

We select the *Master-AP* method as the label generator algorithm and use the cross-entropy loss function of (7). The models are trained iteratively to minimize the gap between the predictions and the ground truth labels. Fig. 3a, which displays the proportion of training loss reduction relative to the initial loss value versus the training epochs. Both the DNN-based models and conventional HQCNN [11] exhibit slower convergence due to their increased computational overhead and more complex optimization landscapes, requiring more than 20 epochs to converge. By contrast, the proposed HQCNN converges significantly faster, stabilizing within approximately 7 epochs, showing the advantage of our design. Moreover, according to the Google Quantum AI document [16], the estimated quantum simulation time of a 16-qubit model is 0.009 seconds and 0.023 seconds for a classical c2-standard-60 CPU processor in case of noiseless and noisy scenarios, respectively. This shows that QML has the potential to speed up the computational process even at the simulation level. Note that the processing speed of the quantum system significantly depends on the hardware configuration.

B. Performance of Unsupervised Learning

In Table II, we compare the performance of the learning-based benchmarks and EPAS, scaling up the system from small to medium size. In a small system, the EPAS provides the optimal solution at the highest cost, which becomes excessively complex if $K > 15$. Both our proposed HQCNN and the HQCNN of [11] achieve approximately 98% of the global optimum, becoming the second-best benchmark. However, the hybrid quantum models still remain effective as the system-scale is expanding, demonstrating robustness in scenarios where the EPAS fails to find the solution, clearly outperforming other methods in all tested scenarios. The DNN-based models' performance is consistently lower than that of ours, with approximately 5%, 10%, and 13% lower for heavy-CNN, light-CNN, and MLP, respectively. We also observe that heuristic methods such as RPA, GPA, MPA, and LPA, consistently yield lower average sum-rates than the proposed HQCNN model. The former becomes even more inferior when the system scales up. For example, in small-scale settings of $(M, L, K, \tau_p) = (10, 2, 6, 3)$, the best-performing heuristic method (MPA) achieves 17.21 (Mbps), 4.8% lower than our model (18.08%). As the system scales to medium sized configurations (e.g., $M = 30 \sim 45$), the limitations of heuristic methods become more pronounced. Quantitatively, the exhibit performance gaps up to 20.4%. For instance, at $M = 45$, $K = 20$, LPA yields 111.05 Mbps, compared to our HQCNN's 133.75 Mbps, while GPA and MPA are outperformed by about 17%. These methods rely on localized optimization criteria, such as minimizing interference for individual UEs, specific APs, or within spatially bounded regions. These strategies falter in complex multi-user environments. For example, GPA prioritizes low-performance users and overlooks high-rate UEs; MPA restricts decisions to the single strongest AP per user; and LPA fails when UEs are located densely in specific areas. These observations have shown the effectiveness of our model in attaining superior sum-rate over other methods.

To characterized the performance versus complexity, we consider an unconventional efficiency metric, the ratio of the sum-rate and the number of parameters as seen in Fig. 3b. Although, the performance of our model and the HQCNN in [11] are equal, our model obtains higher per-parameter-throughput since our design utilizes less quantum trainable parameters. However, the heavy CNN can obtain higher total throughput than the light CNN and MLP, it requires much more hardware resources, leading to the lowest parameter efficiency. These results suggest that our model may indeed be deemed efficient over benchmarks.

C. Limitation

Despite its potential advantages, HQCNN remains constrained by the limitations of the near-term quantum hardware. The model is particularly sensitive to noise sources such as gate noise, shot noise, and decoherence, which accumulate and degrade the performance during training. Depolarizing noise, arising from imperfect gate operations, leads to information loss, and shot noise reflects statistical fluctuations, owing to using a finite number of measurement samples. While shot noise can be reduced by increasing the number of measurement shots, trading between operation time and the accuracy of circuits, depolarizing can be mitigated by zero-noise extrap-

TABLE II: Average sum-rate (Mbps) comparison in small-scale and medium-scale cell-free massive MIMO systems (unsupervised).

System size	M, L	K	τ_p	Random	Greedy [4]	Master-AP [5]	Location-based [6]	MLP	Light-CNN	Heavy-CNN	HQCNN [11]	Ours	EPAS
Small	$(10, 2)$	6	3	15.75	15.76	17.21	16.50	17.14	17.20	17.51	18.05	18.08	**18.43**
	$(10, 2)$	9	3	18.34	19.07	22.13	20.67	21.82	22.93	23.13	23.80	23.84	**24.29**
	$(10, 2)$	12	3	23.20	24.76	26.22	25.54	25.73	27.18	27.79	28.19	28.19	**28.74**
	$(10, 2)$	15	3	27.18	33.68	35.03	34.55	33.89	35.71	36.85	37.36	37.45	–
Medium	$(30, 2)$	20	10	67.67	82.93	78.93	82.16	77.65	89.77	93.30	99.41	99.51	–
	$(35, 2)$	20	10	78.34	96.98	90.98	94.45	89.69	103.34	109.95	115.44	115.44	–
	$(40, 2)$	20	10	88.43	102.31	98.20	100.12	96.51	111.17	118.79	123.72	124.56	–
	$(45, 2)$	20	10	91.22	112.45	110.45	111.05	109.60	122.31	130.20	133.04	133.75	–

olation (ZNE) techniques, which run the circuit at different noise levels and extrapolate back to the zero-noise limit [17]. Decoherence, caused by environmental interactions, results in the erosion of quantum behavior. This can be addressed by adopting low-depth PQCs to keep circuit execution time within coherence limits [9]. Accordingly, HQCNN utilizes depth-6 PQCs to minimize decoherence risk.

The impact of noise at different number of qubits is illustrated in Fig. 3c for $(M, L, K, \tau_p) = (40, 2, 20, 10)$. The proposed model falters in noisy environments, at a depolarization error rate of 0.1, especially upon increasing the number of qubits. However, the ZNE technique can significantly mitigate depolarizing errors. These observations indicate that increasing the number of qubits can theoretically enhance the computational power of the model by providing richer feature maps. However in practical noisy systems, ZNE might inflict additional errors, hence degrading the performance.

V. Conclusion

A powerful hybrid quantum CNN model was conceived to assign the pilot signals in cell-free massive MIMO systems by maximizing the total ergodic throughput. The proposed HQCNN utilizes parameterized quantum circuits for efficient feature extraction, leveraging quantum-domain advantages, including superposition and entanglement. Our hybrid model significantly reduces trained parameters, so resulting in faster convergence and low cost. Numerical results showed that HQCNN obtained a near-optimal ergodic throughput for small-scale systems and competitive performance in large-scale systems, highlighting its efficiency in pilot assignment tasks.

Appendix

Let us view the pilot assignment in Problem (4) as a mapping. The proposed HQCNN model should be interpreted as: 1) the MLP for pre-processing $f_0(\boldsymbol{\beta}; \boldsymbol{\theta}_0) : \mathbb{R}^{M \times K} \to \mathbb{R}^{n_{\text{in}}}$; 2) the QCNN delighted as, $f_1(\mathbf{z}_0; \boldsymbol{\theta}_1) : \mathbb{R}^{n_{\text{in}}} \to \mathbb{R}^{n_{\text{out}}}$, where $\mathbf{z}_0 = f_0(\boldsymbol{\beta}; \boldsymbol{\theta}_0)$; and 3) the MLP for post-processing $f_2(\mathbf{z}_1; \boldsymbol{\theta}_2) : \mathbb{R}^{n_{\text{out}}} \to \mathbb{R}^{K \times \tau_p}$, where $\mathbf{z}_1 = f_1(\mathbf{z}_0; \boldsymbol{\theta}_1)$. The overall mapping of the proposed HQCNN model is

$$f(\boldsymbol{\beta}; \boldsymbol{\theta}) = f_2(f_1[f_0(\boldsymbol{\beta}; \boldsymbol{\theta}_0); \boldsymbol{\theta}_1]; \boldsymbol{\theta}_2) : \mathbb{R}^{M \times K} \to \mathbb{R}^{K \times \tau_p}. \quad (13)$$

According to the universal approximation theorems of classical ML [18], quantum ML [19], and combining the three main mappings, the error bound of $f(\boldsymbol{\beta}; \boldsymbol{\theta})$ is formulated as

$$\begin{aligned} \|f(\boldsymbol{\beta}; \boldsymbol{\theta}) - g^*(\boldsymbol{\beta})\|_F &= \|f_2(f_1(f_0(\boldsymbol{\beta}; \boldsymbol{\theta}_0); \boldsymbol{\theta}_1); \boldsymbol{\theta}_2) - g^*(\boldsymbol{\beta})\|_F \\ &\overset{(a)}{\leq} \|f_0(\boldsymbol{\beta}; \boldsymbol{\theta}_0) - g_1^*(\boldsymbol{\beta})\|_F + w_0 \|f_1(\boldsymbol{z}_0; \boldsymbol{\theta}_1) - g_2^*(\boldsymbol{z}_0)\|_F \\ &+ w_1 \|f_2(\boldsymbol{z}_1; \boldsymbol{\theta}_2) - g_3^*(\boldsymbol{z}_1)\|_F \overset{(b)}{\leq} \epsilon_0 + w_0 \epsilon_1 n^{-0.5} + w_1 \epsilon_2, \end{aligned} \quad (14)$$

where $g^*(\boldsymbol{\beta})$ is the ideal mapping of the large-scale fading coefficients $\boldsymbol{\beta}$; ϵ_0 and ϵ_2 are the tolerable errors of the MLP; ϵ_1 is the quantum approximation error depending both on the number of qubits and on the design of the PQC; w_0 and w_1 represent the relative error contributions, which demonstrate that both $f_1(\cdot;\cdot)$ and $f_2(\cdot;\cdot)$ are affected by the output of the previous blocks [20]. In (14), (a) is obtained by exploiting the sequential nature of $f(\boldsymbol{\beta}; \boldsymbol{\theta})$ as a composition of the functions that the summation of errors is bounded across each stage [21, Chapter 5] and (b) is derived by the universal approximation theorem. Since $w_0, w_1 \to 1$ as $\epsilon_0, \epsilon_1 \to 0$, the total error can be made arbitrarily small by choosing a sufficiently large number of qubits n and appropriately designing the classical models. Thus, an HQCNN can find the ideal mapping function between the large-scale coefficients and the pilot assignment strategy, which completes the proof.

References

[1] N. X. Tung et al., "Distributed Graph Neural Network Design for Sum Ergodic Spectral Efficiency Maximization in Cell-Free Massive MIMO," *IEEE Trans. Vehicular Tech*, 2024.

[2] D. Yu et al., "Learning decentralized power control in cell-free massive MIMO networks," *IEEE Trans. Vehicular Tech*, 2023.

[3] Z. Wang et al., "Pilot Assignment with Approximation Ratio in Cell-Free Massive MIMO Systems," *IEEE Trans. Vehicular Tech*, 2024.

[4] H. Q. Ngo et al., "Cell-free massive MIMO versus small cells," *IEEE Trans. Wirel. Commun.*, 2017.

[5] M. Zaher et al., "Learning-based downlink power allocation in cell-free massive MIMO systems," *IEEE Trans. Wirel. Commun.*, 2022.

[6] T. H. Nguyen et al., "An efficient location-based pilot assignment in cell-free massive MIMO," *ICT Express*, 2023.

[7] K. Kim et al., "Deep CNN-based pilot allocation scheme in massive mimo systems," *KSII T Internet Info*, 2020.

[8] F. Fan et al., "Hybrid quantum-classical convolutional neural network model for image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, 2023.

[9] Y. Du et al., "Quantum machine learning: A hands-on tutorial for machine learning practitioners and researchers," *arXiv:2502.01146*, 2025.

[10] L. Hanzo et al., "Quantum information processing, sensing, and communications: Their myths, realities, and futures," *Proc. IEEE*, 2025.

[11] T. Hur et al., "Quantum convolutional neural network for classical data classification," *Quantum Mach. Intell.*, 2022.

[12] S.-G. Jeong et al., "Hybrid quantum convolutional neural networks for UWB signal classification," *IEEE Access*, 2023.

[13] S. Oh, J. Choi, and J. Kim, "A tutorial on quantum convolutional neural networks (QCNN)," in *Proc. ICTC*, 2020.

[14] N. Munikote, "Comparing quantum encoding techniques," *arXiv:2410.09121*, 2024.

[15] X. Zhou et al., "Towards quantum-native communication systems: New developments, trends, and challenges," *arXiv:2311.05239*, 2023.

[16] Google Quantum AI, "Choose your quantum hardware," 2024. [Online]. Available: https://quantumai.google/qsim/choose_hw

[17] R. LaRose et al., "Mitiq: A software package for error mitigation on noisy quantum computers," *Quantum*, 2022.

[18] K. Hornik et al., "Multilayer feedforward networks are universal approximators," *Neural Networks*, 1989.

[19] L. Gonon et al., "Universal approximation theorem and error bounds for quantum neural networks and quantum reservoirs," *IEEE Trans. Neural Netw. Learn. Syst.*, 2025.

[20] A. Saltelli, *Global Sensitivity Analysis: the Primer*. John Wiley & Sons, 2008.

[21] S. Shalev-Shwartz et al., *Understanding machine learning: From theory to algorithms*. Cambridge University Press, 2014.