ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-6327-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICASSP49357.2023.10094634

OPTIMIZING QUANTUM FEDERATED LEARNING BASED ON FEDERATED QUANTUM NATURAL GRADIENT DESCENT

Jun Qi[1,2], *Xiao-Lei Zhang*[3], *Javier Tejedor*[4]

[1] Electronic Engineering, Fudan University, Shanghai, China
[2] Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA
[3] Marine Science and Technology, Northwestern Polytechnical University, Xian, China
[4] Institute of Technology, Universidad San Pablo-CEU, CEU Universities, Urb. Montepríncipe, Madrid, Spain

ABSTRACT

Quantum federated learning (QFL) is a quantum extension of the classical federated learning model across multiple local quantum devices. An efficient optimization algorithm is always expected to minimize the communication overhead among different quantum participants. In this work, we propose an efficient optimization algorithm, namely federated quantum natural gradient descent (FQNGD), and further, apply it to a QFL framework that is composed of a variational quantum circuit (VQC)-based quantum neural networks (QNN). Compared with stochastic gradient descent methods like Adam and Adagrad, the FQNGD algorithm admits much fewer training iterations for the QFL to get converged. Moreover, it can significantly reduce the total communication overhead among local quantum devices. Our experiments on a handwritten digit classification dataset justify the effectiveness of the FQNGD for the QFL framework in terms of a faster convergence rate on the training set and higher accuracy on the test set.

Index Terms— Quantum neural network, variational quantum circuit, quantum federated learning, federated quantum natural gradient descent

1. INTRODUCTION

Deep learning (DL) technologies have been successfully applied in many machine learning tasks such as speech recognition (ASR) [1], natural language processing (NLP) [2], and computer vision [3]. The bedrock of DL applications highly relies on the hardware breakthrough of the graphic processing unit (GPU) and the availability of a large amount of training data [4, 5]. However, the advantages of large-size DL models, such as GPT-3 [6] and BERT [7], are faithfully attributed to the significantly powerful computing capabilities that are only privileged to big companies equipped with numerous costly and industrial-level GPUs. With the rapid development of noisy intermediate-scale quantum (NISQ) devices [8, 9, 10], the quantum computing hardware is expected to speed up the classical DL algorithms by creating novel quantum machine learning (QML) approaches like quantum neural networks (QNN) [11, 12, 13, 14] and quantum kernel learning (QKL) [14]. The VQC-based QNN seeks to parameterize a distribution through some set of adjustable model parameters, and the QKL methods utilize quantum computers to define a feature map that projects classical data into the quantum Hilbert space. Both QML methods have advantages and disadvantages in dealing with different machine learning tasks and it could not be simply claimed which one is the most suitable choice. However, two obstacles prevent the NISQ devices from applying to QML in practice. The first challenge is that the classical DL models cannot be deployed on NISQ devices without model conversion to quantum tensor formats [15, 16]. For the second challenge, the NISQ devices admit a few physical qubits such that insufficient qubits could be spared for the quantum error correction [17, 9, 10]. More significantly, the representation power of QML is quite limited to the small number of currently available qubits [18] and the increase of qubits may lead to the problem of Barren Plateaus [19].

To deal with the first challenge, in this work, we introduce a variational quantum algorithm, namely a variational quantum circuit (VQC), to enable QNN to be simulated on the currently available NISQ devices. The VQC-based QNNs have attained even exponential advantages over the DL counterparts on exclusively many tasks like ASR [20, 21], NLP [22], and reinforcement learning [23]. As for the second challenge, distributed QML systems, which consist of local quantum machines, can be set up to enhance the quantum computing power. One particular distributed QML architecture is called quantum federated learning (QFL), which aims to build a decentralized computing model derived from a classical FL [24].

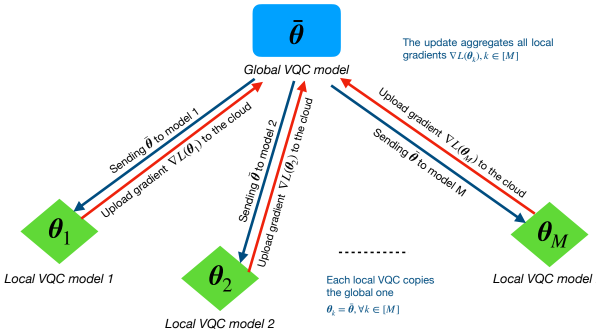

$\bar{\boldsymbol{\theta}}$
Global VQC model
The update aggregates all local gradients $\nabla L(\boldsymbol{\theta}_k), k \in [M]$
Sending $\bar{\boldsymbol{\theta}}$ to model 1
Upload gradient $\nabla L(\boldsymbol{\theta}_1)$ to the cloud
Sending $\bar{\boldsymbol{\theta}}$ to model 2
Upload gradient $\nabla L(\boldsymbol{\theta}_2)$ to the cloud
Upload gradient $\nabla L(\boldsymbol{\theta}_M)$ to the cloud
Sending $\bar{\boldsymbol{\theta}}$ to model M
$\boldsymbol{\theta}_1$
Local VQC model 1
$\boldsymbol{\theta}_2$
Local VQC model 2
Each local VQC copies the global one $\boldsymbol{\theta}_k = \bar{\boldsymbol{\theta}}, \forall k \in [M]$
$\boldsymbol{\theta}_M$
Local VQC model M

Fig. 1. *An illustration of quantum federated learning. The global VQC parameter $\bar{\boldsymbol{\theta}}$ is first transmitted to local VQCs $\boldsymbol{\theta}_k$. Then, the updated gradients $\nabla\mathcal{L}(\boldsymbol{\theta}_k)$ based on the participants' local data are sent back to the centralized server and then they are aggregated to update the parameters of the global VQC.*

Konečnỳ *et al.* [25] first proposed the FL strategies to improve the communication efficiency of a distributed computing system, and McMahan *et al.* [26] set up the FL systems with the concerns in the use of big data and a large-scale cloud-based DL [27]. The FL framework depends on the advances in hardware progress, making tiny DL systems practically powerful. For example, an ASR system on the cloud can transmit a global acoustic model to a user's cell phone and then send the updated information back to the cloud without collecting the user's private data on the centralized computing server. As shown in Figure 1, the QFL system is similar to a classical FL system and differs from distributed learning in several ways as follows: (a) the datasets in the framework of QFL are not necessarily balanced; (b) the data in QFL are not assumed to be generated from an independent and identical (i.i.d.) distribution.

Chen *et al.* [28] demonstrates the QFL architecture that is built upon the classical FL paradigm, where the central node holds a global VQC and receives the trained VQC parameters from participants' local quantum devices. Therefore, the QFL model, which inherits the advantages of the FL framework, can unite tiny local quantum devices to generate a powerful global one. This methodology helps to build a privacy-preserving QML system and leverages quantum computing to further boost the computing power of the classical FL. As shown in Figure 1, our proposed QFL and FL differ in the models utilized in federated learning systems, where QFL employs VQC models instead of their classical DL counterparts for FL. More specifically, the QFL comprises a global VQC model deployed on the cloud, and there are M local VQC models assigned to users' devices. The training process of QFL involves three key procedures: (1) the parameters of global VQC model $\bar{\boldsymbol{\theta}}$ are transmitted to K local participants' devices; (2) each local VQC first adaptively trains its own model based on the local users' data, and then separately sends the model gradients $\nabla\mathcal{L}(\boldsymbol{\theta}_k)$ back to the centralized platform; (3) the uploaded gradients from local participants are averagely aggregated to create a global gradient to update further the global model parameters $\bar{\boldsymbol{\theta}}$.

Despite the advantages of QFL in practice, an inherent bottleneck of QFL is the communication overhead among different VQC models, which bounds up with the performance of QFL. To reduce the cost of communication overhead, we expect a more efficient training algorithm to speed up the convergence rate such that fewer counts of global model updates can be attained. Based on the above analysis, in this work, we put forth a federated quantum learning algorithm, namely federated quantum natural gradient descent (FQNGD), for the training of QFL. The FQNGD algorithm, developed from the quantum natural gradient descent (QNGD) algorithm, admits a more efficient training process for a single VQC [29]. In particular, Stokes *et al.* [29] first claimed that the Fubini-Study metric tensor could be employed for the QNGD. Besides, compared with the work [28], the gradients of VQC are uploaded to a global model rather than the VQC parameters of local devices such that the updated gradients can be collected without being accessed to the VQC parameters as shown in [28].

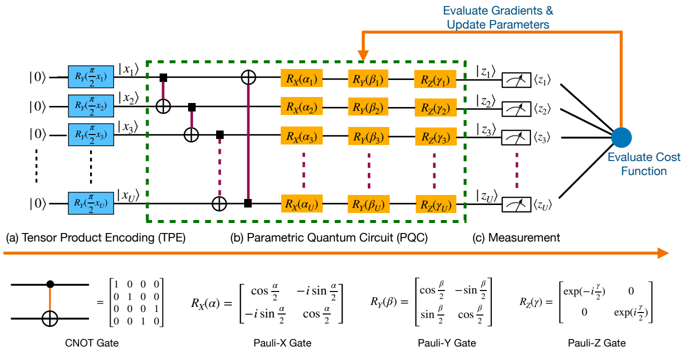


Fig. 2. *The VQC is composed of three components: (a) TPE; (b) PQC; (c) Measurement. The TPE utilizes a series of $R_Y(\frac{\pi}{2}x_i)$ to transform classical inputs into quantum states. The PQC consists of CNOT gates and single-qubit rotation gates R_X, R_Y, R_Z with trainable parameters $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, and $\boldsymbol{\gamma}$. The CNOT gates are non-parametric and impose the property of quantum entanglement among qubits, and R_X, R_Y and R_Z are parametric gates and can be adjustable during the training stage. The PQC model in the green dash square is repeatably copied to build a deep model. The measurement converts the quantum states $|z_1\rangle, |z_2\rangle, ..., |z_U\rangle$ into the corresponding expectation values $\langle z_1\rangle, \langle z_2\rangle, ..., \langle z_U\rangle$. The outputs $\langle z_1\rangle, \langle z_2\rangle, ..., \langle z_U\rangle$ is connected to a loss function and the gradient descent algorithms can be used to update the VQC model parameters. Besides, both CNOT gates and R_X, R_Y and R_Z correspond to unitary matrices as shown below the VQC framework.*

2. VARIATIONAL QUANTUM CIRCUIT

An illustration of VQC is shown in Figure 2, where the VQC model consists of three components: (a) tensor product encoding (TPE); (b) parametric quantum circuit (PQC); (c) measurement. The TPE initializes the input quantum states $|x_1\rangle$, $|x_2\rangle$, ..., $|x_U\rangle$ from the classical inputs x_1, x_2, ..., x_U, and the PQC operator transforms the quantum states $|x_1\rangle$, $|x_2\rangle$, ..., $|x_U\rangle$ into the output quantum states $|z_1\rangle$, $|z_2\rangle$, ..., $|z_U\rangle$. The outputs correspond to the expected observations $\langle z_1\rangle$, $\langle z_2\rangle$, ..., $\langle z_U\rangle$ arised from the measurement of the Pauli-Z operators. We present the three components in detail next.

The TPE model was first proposed in [30]. It aims to convert a classical vector $\mathbf{x}$ into a quantum state $|\mathbf{x}\rangle$ by setting up a one-to-one mapping as Eq. (1).

$$\begin{aligned} |\mathbf{x}\rangle &= \left(\otimes_{i=1}^{U} R_Y(\frac{\pi}{2}x_i)\right)|0\rangle^{\otimes U} \\ &= \begin{bmatrix}\cos(\frac{\pi}{2}x_1)\\ \sin(\frac{\pi}{2}x_1)\end{bmatrix} \otimes \begin{bmatrix}\cos(\frac{\pi}{2}x_2)\\ \sin(\frac{\pi}{2}x_2)\end{bmatrix} \otimes \cdots \otimes \begin{bmatrix}\cos(\frac{\pi}{2}x_U)\\ \sin(\frac{\pi}{2}x_U)\end{bmatrix}, \end{aligned} \tag{1}$$

where $R_Y(\cdot)$ refers to a single-qubit quantum gate rotated across Y-axis and each x_i is constrained to the domain of $[0, 1]$, which results in a reversely one-to-one conversion between $\mathbf{x}$ and $|\mathbf{x}\rangle$.

Moreover, the PQC is equipped with the CNOT gates for quantum entanglement and learnable quantum gates, i.e., $R_X(\alpha_i)$, $R_Y(\beta_i)$, and $R_Z(\gamma_i)$, where the qubit angles α_i, β_i, and γ_i are tuned in the training process. The PQC framework in the green dash square is repeatedly copied to set up a deep model, and the number of the PQC frameworks is called the depth of the VQC. The operation of the measurement outputs the classical expected observations $|z_1\rangle$, $|z_2\rangle$, ..., $|z_U\rangle$ from the quantum output states. The expected outcomes are used to calculate the loss value and the gradient descents [31], which are used to update the VQC model parameters by applying the back-propagation algorithm [32] based on the stochastic gradient descent (SGD) optimizer.

3. QUANTUM NATURAL GRADIENT DESCENT

As shown in Eq. (2), at step t, the standard gradient descent minimizes a loss function $\mathcal{L}(\boldsymbol{\theta})$ with respect to the parameters $\boldsymbol{\theta}$ in a Euclidean space.

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta\nabla\mathcal{L}(\boldsymbol{\theta}_t), \tag{2}$$

where η is the learning rate.

The standard gradient descent algorithm conducts each optimization step in a Euclidean geometry on the parameter space. However, since the form of parameterization is not unique, different compositions of parameterizations are likely to distort the distance geometry within the optimization landscape. A better alternative method is to perform the gradient descent in the distribution space, namely natural gradient descent [33], which is dimension-free and invariant for different parameterization forms. Each optimization step of the natural gradient descent chooses the optimum step size for the update of parameter $\boldsymbol{\theta}_t$, regardless of the choice of parameterization. Mathematically, the standard gradient descent is modified as Eq. 3.

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta F^{-1} \nabla \mathcal{L}(\boldsymbol{\theta}_t), \tag{3}$$

F denotes the Fisher information matrix, which acts as a metric tensor that transforms the steepest gradient descent in the Euclidean parameter space to the steepest descent in the distribution space.

Since the standard Euclidean geometry is sub-optimal for the optimization of quantum variational algorithms, a quantum analog has the following form as Eq. (4).

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta g^{+}(\boldsymbol{\theta}_t) \nabla \mathcal{L}(\boldsymbol{\theta}_t), \tag{4}$$

where $g^{+}(\boldsymbol{\theta}_t)$ refers to the pseudo-inverse and is associated with the specific architecture of the quantum circuit. The coefficient $g^{+}(\boldsymbol{\theta}_t)$ can be calculated using the Fubini-Study metric tensor, which it then reduces to the Fisher information matrix in the classical limit [34].

4. QUANTUM NATURAL GRADIENT DESCENT FOR VQC

Before employing the QFNGD for a quantum federated learning system, we concentrate on the use of QNGD for a single VQC. For simplicity, we leverage a block-diagonal approximation to the Fubini-Study metric tensor for composing QNGD into the VQC training on the NISQ quantum hardware.

We set an initial quantum state $|\psi_0\rangle$ and a PQC with L layers. For $l \in [L]$, we separately denote $\mathbf{W}_l$ and $\mathbf{V}_l(\boldsymbol{\theta}_l)$ as the unitary matrices associated with non-parameterized quantum gates and parameterized quantum ones, respectively.

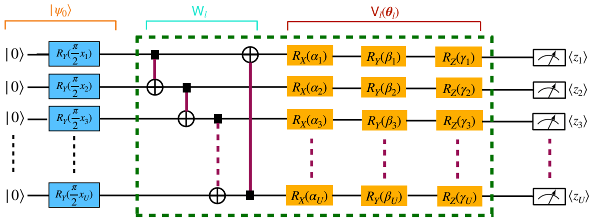


Fig. 3. *An illustration of unitary matrices associated with the non-parametric and parametric gates. $\forall l \in [L]$, the matrices $\mathbf{W}_l$ correspond to the non-parametric gates, the matrices $\mathbf{V}_l(\boldsymbol{\theta}_l)$ are associated with the parametric ones, and $|\psi_0\rangle$ refers to the initial quantum state that is derived from the operation of the TPE.*

Let's consider a variational quantum circuit as Eq. (5).

$$U(\boldsymbol{\theta})|\psi_0\rangle = \mathbf{V}_L(\boldsymbol{\theta}_L)\mathbf{W}_L \cdots \mathbf{V}_l(\boldsymbol{\theta}_l)\mathbf{W}_l \cdots \mathbf{V}_1(\boldsymbol{\theta}_1)\mathbf{W}_1|\psi_0\rangle \tag{5}$$

Furthermore, any unitary quantum parametric gates can be rewritten as $\mathbf{V}_l(\boldsymbol{\theta}_l) = \exp(i\boldsymbol{\theta}_l H_l)$, where H_l refers to the Hermitian generator of the gate $\mathbf{V}_L$. The approximation to the Fubini-Study metric tensor admits that for each parametric layer l in the variational quantum circuit, the $n_l \times n_l$ block-diagonal submatrix of the Fubini-Study metric tensor $g^{+}_{l,i,j}$ is calculated by Eq. (6).

$$g^{+}_{l,i,j} = \langle\psi_l|H_l(i)H_l(j)|\psi_l\rangle - \langle\psi_l|H_l(i)|\psi_l\rangle\langle\psi_l|H_l(j)|\psi_l\rangle, \tag{6}$$

where

$$|\psi_l\rangle = \mathbf{V}_l(\boldsymbol{\theta}_l)\mathbf{W}_l \cdots \mathbf{V}_1(\boldsymbol{\theta}_1)\mathbf{W}_1|\psi_0\rangle. \tag{7}$$

In Eq. (7), $|\psi_l\rangle$ denotes the quantum state before the application of the parameterized layer l. Figure 4 illustrates a simplified version of a VQC, where $\mathbf{W}_1$ and $\mathbf{W}_2$ are related to non-parametric gates, and $\mathbf{V}_1(\theta_0, \theta_1)$ and $\mathbf{V}_2(\theta_2, \theta_3)$ correspond to the parametric gates with adjustable parameters, respectively. Since there are two layers, each of which owns two free parameters, the block-diagonal approximation is composed of two 2×2 matrices, g^{+}_1 and g^{+}_2, which can be separately expressed as Eq. (8) and (9).

$$g^{+}_1 = \begin{bmatrix} \langle z_0^2\rangle - \langle z_0\rangle^2 & \langle z_0 z_1\rangle - \langle z_0\rangle\langle z_1\rangle \\ \langle z_0 z_1\rangle - \langle z_0\rangle\langle z_1\rangle & \langle z_1^2\rangle - \langle z_1\rangle^2 \end{bmatrix}, \tag{8}$$

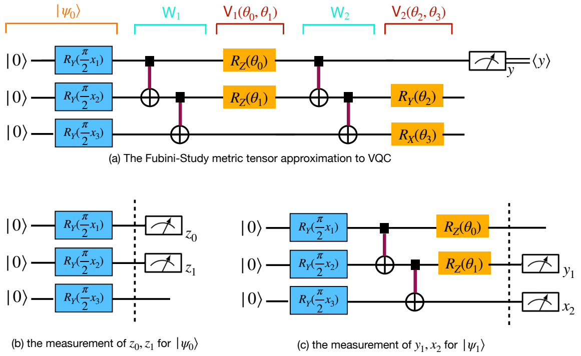
and

$$g^{+}_2 = \begin{bmatrix} \langle y_1^2\rangle - \langle y_1\rangle^2 & \langle y_1 x_2\rangle - \langle y_1\rangle\langle x_2\rangle \\ \langle y_1 x_2\rangle - \langle y_1\rangle\langle x_2\rangle & \langle x_2^2\rangle - \langle x_2\rangle^2 \end{bmatrix}. \tag{9}$$

The elements of g^{+}_1 and g^{+}_2 compose $g^{+}(\boldsymbol{\theta})$ as Eq. (10).

$$g^{+}(\boldsymbol{\theta}) = \begin{bmatrix} g^{+}_1 & 0 \\ 0 & g^{+}_2 \end{bmatrix}. \tag{10}$$

Then, we employ Eq. (4) to update the VQC parameter $\boldsymbol{\theta}$.



(a) The Fubini-Study metric tensor approximation to VQC

(b) the measurement of z_0, z_1 for $|\psi_0\rangle$

(c) the measurement of y_1, x_2 for $|\psi_1\rangle$

Fig. 4. *A demonstration of the VQC approximation method based on the Fubini-Study metric tensor: (a) A block-diagonal approximation to VQC based on the Fubini-Study metric tensor; (b) a measurement of z_0, z_1 for $|\psi_0\rangle$; (c) measurement of y_1, x_2 for $|\psi_1\rangle$.*

5. FEDERATED QUANTUM NATURAL GRADIENT DESCENT

A QFL system can be built by setting up VQC models in an FL manner, given the dataset S composed of subsets $S_1, S_2, ..., S_K$, the objective of QFL can be formulated as:

$$\min_{\boldsymbol{\theta}} \sum_{k=1}^{K} w_k g^{+}_k(\boldsymbol{\theta}) \mathcal{L}(\boldsymbol{\theta}; S_k), \tag{11}$$

Algorithm 1 Iterative Approximation

```
Given the dataset $S = S_1 \cup S_2 \cup \cdots \cup S_K$.
Initialize global parameter $\bar{\boldsymbol{\theta}}$ and broadcast it to participants $\boldsymbol{\theta}_0^{(k)}$.
Assign each participant with the subset $S_k$.
For each global model update at epoch $t = [T]$ do
    For each participant $k \in [K]$ **in parallel** do
        Attain $g^+(\boldsymbol{\theta}_t^{(k)}; S_k)\nabla\mathcal{L}(\boldsymbol{\theta}_t^{(k)})$ for the $k^{th}$ VQC.
        Send gradient $g^+(\boldsymbol{\theta}_t^{(k)})\nabla\mathcal{L}(\boldsymbol{\theta}_t^{(k)}; S_k)$ to the coordinator.
    End for
    The coordinator aggregates the received gradients.
    The coordinator updates the global model by Eq. (13).
    Broadcast the updated global $\bar{\boldsymbol{\theta}}_{t+1}$ to all participants.
End for
```

where w_k refers to the coefficient assigned to the k-th gradient participant, and each w_k can be estimated as:

$$w_k = \frac{|S_k|}{|S|} = \frac{|S_k|}{\sum_{k=1}^{K} |S_k|}. \tag{12}$$

The QNGD algorithm is applied for each VQC and the uploaded gradients of all VQCs are aggregated to update the model parameters of the global VQC. The FQNGD can be mathematically summarized as:

$$\bar{\boldsymbol{\theta}}_{t+1} = \bar{\boldsymbol{\theta}}_t - \eta \sum_{k=1}^{K} \frac{|S_k|}{|S|} g_k^+(\boldsymbol{\theta}_t^{(k)})\nabla\mathcal{L}(\boldsymbol{\theta}_t^{(k)}; S_k), \tag{13}$$

where $\bar{\boldsymbol{\theta}}_t$ and $\boldsymbol{\theta}_t^{(k)}$ separately correspond to the model parameters of the global VQC and the k-th VQC model at epoch t, and N_k represents the amount of training data stored in the participant k, and the sum of K participants' data is equivalent to N.

Compared with the SGD counterparts used for QFL, the FQNGD algorithm admits adaptive learning rates for the gradients such that the convergence rate could be accelerated according to the VQC model status.

5.0.1. *Empirical Results*

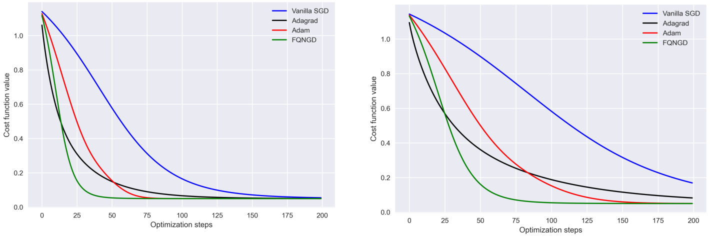
To demonstrate the FQNGD algorithm for QFL, we perform the binary and ternary classification tasks on the standard MNIST dataset [35], with digits $\{2, 5\}$ for the binary task and $\{1, 3, 7\}$ for the ternary one. There are 11379 training data and 1924 test data for the binary classification, and 19138 training data and 3173 test data are assigned for the ternary classification. As for the setup of QFL in our experiments, the QFL system consists of 6 identically local VQC participants, each of which owns the same amount of training data. The test data are stored in the global part and are used to evaluate the classification performance.

Table 1. The simulation results of a binary classification.

Methods	Vanilla SGD	Adagrad	Adam	FQNGD
Accuracy	98.48	98.81	98.87	**99.32**

Table 2. The simulation results of a ternary classification.

Methods	Vanilla SGD	Adagrad	Adam	FQNGD
Accuracy	97.86	98.63	98.71	**99.12**



(a) Learning curves on the binary classification

(b) Learning curves on the ternary classification

Fig. 5. *Simulation results of binary and ternary classifications on the training set of the MNIST database. (a) The learning curves of various optimization methods for the binary classification; (b) the learning curves of various optimization methods for the ternary classification.*

We compare our proposed FQNGD algorithm with other three optimizers: the naive SGD optimizer, the Adagrad optimizer [36], and the Adam optimizer [37]. The Adagrad optimizer is a gradient descent optimizer with a past-gradient-dependent learning rate in each dimension. The Adam optimizer refers to the gradient descent method with an adaptive learning rate as well as adaptive first and second moments.

As shown in Figure 5, our simulation results suggest that our proposed FQNGD method is capable of achieving the fastest convergence rate among the optimization approaches. It means that the FQNGD method can reduce the communication overhead cost and maintain the baseline performance of binary and ternary classifications on the MNIST dataset. Moreover, we evaluate the QFL performance in terms of classification accuracy. The FQNGD method outperforms the other counterparts with the highest accuracy values. In particular, the FQNGD is designed for the VQC model and can attain better empirical results than the Adam and Adagrad methods with adaptive learning rates over epochs.

6. CONCLUSION AND FUTURE WORK

This work focuses on the design of the FQNGD algorithm for the QFL system in which multiple local VQC models are applied. The FQNGD is derived from training a single VQC based on QNGD, which relies on the block-diagonal approximation of the Fubini-Study metric tensor to the VQC architecture. We put forth the FQNGD method to train the QFL system. Compared with other SGD methods such as Adagrad and Adam optimizers, our experiments of the classification tasks on the MNIST dataset demonstrate that the FQNGD method attains better empirical results than other SGD methods, while the FQNGD exhibits a faster convergence rate than the others, which implies that our FQNGD method suggests that it is capable of reducing the communication cost and can maintain the baseline empirical results.

Although this work focuses on the optimization methods for the QFL system, the decentralized deployment of a high-performance QFL system for adapting to the large-scale dataset is left for our future investigation. In particular, it is essential to consider how to defend against malicious attacks from adversaries and also boost the robustness and integrity of the shared information among local participants. Besides, the deployment of other quantum neural networks like quantum convolutional neural networks (QCNN) [38] are worth further attempts to compose a QFL system.

7. REFERENCES

[1] Xuedong Huang, James Baker, and Raj Reddy, "A Historical Perspective of Speech Recognition," *Communications of the ACM*, vol. 57, no. 1, pp. 94–103, 2014.

[2] Julia Hirschberg and Christopher D Manning, "Advances in Natural Language Processing," *Science*, vol. 349, no. 6245, pp. 261–266, 2015.

[3] Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, and Eftychios Protopapadakis, "Deep Learning for Computer Vision: A Brief Review," *Computational Intelligence and Neuroscience*, vol. 2018, 2018.

[4] Jun Qi, Jun Du, Sabato Marco Siniscalchi, Xiaoli Ma, and Chin-Hui Lee, "Analyzing Upper Bounds on Mean Absolute Errors for Deep Neural Network-based Vector-to-Vector Regression," *IEEE Transactions on Signal Processing*, vol. 68, pp. 3411–3422, 2020.

[5] Jun Qi, Jun Du, Sabato Marco Siniscalchi, and Chin-Hui Lee, "A Theory on Deep Neural Network Based Vector-to-Vector Regression with An Illustration of Its Expressive Power in Speech Enhancement," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 12, pp. 1932–1943, 2019.

[6] Tom Brown and et al., "Language Models Are Few-Shot Learners," in *Proc. Advances in Neural Information Processing Systems*, 2020, vol. 33, pp. 1877–1901.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, "Bert: Pre-Training of Deep Bidirectional Transformers for Language Understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[8] John Preskill, "Quantum Computing in the NISQ Era and Beyond," *Quantum*, vol. 2, pp. 79, August 2018.

[9] Laird Egan and et al., "Fault-Tolerant Control of An Error-Corrected Qubit," *Nature*, vol. 598, no. 7880, pp. 281–286, 2021.

[10] Qihao Guo, Yuan-Yuan Zhao, Markus Grassl, Xinfang Nie, Guo-Yong Xiang, Tao Xin, Zhang-Qi Yin, and Bei Zeng, "Testing A Quantum Error-Correcting Code on Various Platforms," *Science Bulletin*, vol. 66, no. 1, pp. 29–35, 2021.

[11] Marco Cerezo and et al., "Variational Quantum Algorithms," *Nature Reviews Physics*, vol. 3, no. 9, pp. 625–644, 2021.

[12] Hsin-Yuan Huang, Michael Broughton, Masoud Mohseni, Ryan Babbush, Sergio Boixo, Hartmut Neven, and Jarrod R. McClean, "Power of Data in Quantum Machine Learning," *Nature Communications*, vol. 12, no. 1, pp. 1–9, 2021.

[13] Yuxuan Du, Min-Hsiu Hsieh, Tongliang Liu, Shan You, and Dacheng Tao, "Learnability of Quantum Neural Networks," *PRX Quantum*, vol. 2, no. 4, pp. 040337, 2021.

[14] Hsin-Yuan Huang and et al., "Quantum Advantage in Learning from Experiments," *Science*, vol. 376, no. 6598, pp. 1182–1186, 2022.

[15] Jun Qi, Chao-Han Huck Yang, and Pin-Yu Chen, "QTN-VQC: An End-to-End Learning Framework for Quantum Neural Networks," in *NeurIPS 2021 Workshop on Quantum Tensor Networks in Machine Learning*, 2022.

[16] Samuel Yen-Chi Chen, Chih-Min Huang, Chia-Wei Hsing, and Ying-Jer Kao, "An End-to-End Trainable Hybrid Classical-Quantum Classifier," *Machine Learning: Science and Technology*, vol. 2, no. 4, pp. 045021, 2021.

[17] Philip Ball, "Real-Time Error Correction for Quantum Computing," *Physics*, vol. 14, pp. 184, 2021.

[18] Jun Qi, Chao-Han Huck Yang, Pin-Yu Chen, and Min-Hsiu Hsieh, "Theoretical error performance analysis for variational quantum circuit based functional regression," *npj Quantum Information*, vol. 9, no. 1, pp. 4, 2023.

[19] Jarrod R McClean, Sergio Boixo, Vadim N Smelyanskiy, Ryan Babbush, and Hartmut Neven, "Barren plateaus in quantum neural network training landscapes," *Nature communications*, vol. 9, no. 1, pp. 4812, 2018.

[20] Chao-Han Huck Yang, Jun Qi, Samuel Yen-Chi Chen, Pin-Yu Chen, Sabato Marco Siniscalchi, Xiaoli Ma, and Chin-Hui Lee, "Decentralizing Feature Extraction with Quantum Convolutional Neural Network for Automatic Speech Recognition," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021, pp. 6523–6527.

[21] Jun Qi and Javier Tejedor, "Classical-to-Quantum Transfer Learning for Spoken Command Recognition Based on Quantum Neural Networks," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, 2021.

[22] Chao-Han Huck Yang, Jun Qi, Samuel Yen-Chi Chen, Yu Tsao, and Pin-Yu Chen, "When BERT Meets Quantum Temporal Convolution Learning for Text Classification in Heterogeneous Computing," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, 2022.

[23] Samuel Yen-Chi Chen, Chao-Han Huck Yang, Jun Qi, Pin-Yu Chen, Xiaoli Ma, and Hsi-Sheng Goan, "Variational Quantum Circuits for Deep Reinforcement Learning," *IEEE Access*, vol. 8, pp. 141007–141024, 2020.

[24] Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 50–60, 2020.

[25] Jakub Konečnỳ, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon, "Federated Learning: Strategies for Improving Communication Efficiency," *arXiv preprint arXiv:1610.05492*, 2016.

[26] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," in *Proc. Artificial Intelligence and Statistics*, 2017, pp. 1273–1282.

[27] Reza Shokri and Vitaly Shmatikov, "Privacy-Preserving Deep Learning," in *Proc. ACM SIGSAC Conference on Computer and Communications Security*, 2015, pp. 1310–1321.

[28] Samuel Yen-Chi Chen and Shinjae Yoo, "Federated Quantum Machine Learning," *Entropy*, vol. 23, no. 4, pp. 460, 2021.

[29] James Stokes, Josh Izaac, Nathan Killoran, and Giuseppe Carleo, "Quantum Natural Gradient," *Quantum*, vol. 4, pp. 269, 2020.

[30] Edwin Stoudenmire and David J Schwab, "Supervised Learning with Tensor Networks," in *Proc. Advances in Neural Information Processing Systems*, 2016, vol. 29.

[31] Sebastian Ruder, "An Overview of Gradient Descent Optimization Algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[32] Paul J Werbos, "Backpropagation Through Time: What It Does and How to Do It?," *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990.

[33] Shun-Ichi Amari, "Natural Gradient Works Efficiently in Learning," *Neural Computation*, vol. 10, no. 2, pp. 251–276, 1998.

[34] Sam McArdle, Tyson Jones, Suguru Endo, Ying Li, Simon C Benjamin, and Xiao Yuan, "Variational Ansatz-Based Quantum Simulation of Imaginary Time Evolution," *NPJ Quantum Information*, vol. 5, no. 1, pp. 1–6, 2019.

[35] Li Deng, "The MNIST Database of Handwritten Digit Images for Machine Learning Research," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 141–142, 2012.

[36] Agnes Lydia and Sagayaraj Francis, "Adagrad—An Optimizer for Stochastic Gradient Descent," *Int. J. Inf. Comput. Sci*, vol. 6, no. 5, pp. 566–568, 2019.

[37] Diederik P Kingma and Jimmy Ba, "Adam: A Method for Stochastic Optimization," in *Proc. International Conference on Representation Learning*, 2015.

[38] Iris Cong, Soonwon Choi, and Mikhail D Lukin, "Quantum Convolutional Neural Networks," *Nature Physics*, vol. 15, no. 12, pp. 1273–1278, 2019.