

Resource-efficient and robust quantum federated learning via loss aggregation

Cheng-En Tsai, and Hao-Chung Cheng, *Member, IEEE*

Abstract—Federated learning across quantum devices offers a pathway to scalable and privacy-preserving quantum machine learning. However, existing frameworks exchange large gradients or model parameters, creating severe communication and execution overheads that scale with system size and often destabilize training on heterogeneous client data. We introduce Loss Aggregation Quantum Federated Learning (LossAgg-QFL), a framework that transmits only scalar loss values from each client to a central server. Using Simultaneous Perturbation Stochastic Approximation (SPSA), the server infers global parameter updates through constant-cost evaluations, eliminating dependence on qubit number, circuit depth, and parameter dimension. This design reduces per-round communication and circuit executions by up to 10^6 times and improves runtime scaling from $\mathcal{O}(n^3)$ to $\mathcal{O}(n)$ with respect to qubit count. A multi-directional SPSA strategy further stabilizes optimization, lowering convergence variance from 3.6% to 0.8% across stochastic, gradient-free updates. Evaluations on the MNIST and Fashion-MNIST benchmarks show that LossAgg-QFL achieves up to +59% and +33.8% improvements in accuracy under label- and quantity-skew conditions, and also outperforms baseline methods under balanced data. By aligning learning dynamics with quantum-hardware constraints and federated scalability, LossAgg-QFL demonstrates that loss-based aggregation can elevate quantum federated learning from primarily simulation-driven studies to a more practical and resource-efficient paradigm for distributed quantum intelligence.

Index Terms—Quantum federated learning, Resource-efficient quantum training, Gradient-free optimization, Simultaneous perturbation stochastic approximation, Non-iid robustness

I. INTRODUCTION

Quantum Federated Learning (QFL) [1]–[4] combines quantum computing [5], [6] with the decentralized architecture of Federated Learning (FL) [7], [8], offering a promising path toward scalable, privacy-preserving distributed quantum intelligence. As hardware roadmaps progress toward modular, distributed, and multi-chip quantum processors [9], [10], QFL naturally aligns with these interconnected resources. However, practical deployment is severely hindered by several critical bottlenecks.

A central barrier is *communication overhead* [11], [12]. Conventional frameworks exchange large parameter vectors

or gradient sets with costs that grow with qubit count and model size, creating prohibitive bandwidth usage and complicating privacy-preserving protocols [13], [14]. Another fundamental challenge is the *non-IID quagmire* [15]–[17], where heterogeneous, imbalanced, or label-skewed client data cause high-dimensional local gradient updates to magnify local discrepancies, resulting in training divergence and reduced accuracy [18], [19].

These obstacles are further exacerbated by a gap between simulation and hardware feasibility. Most QFL studies evaluate with classical simulators using exact-gradient optimizers such as Adam [20]. On real quantum hardware, however, back-propagation [21] is infeasible due to wavefunction collapse [22], [23]. While the parameter-shift rule [24], [25] provides accurate gradient estimates, its circuit executions scale extensively with model size and depth, making repeated evaluations under device noise unstable and resource-prohibitive [26]. Taken together, these limitations highlight the need for a new framework that is both *resource-efficient* in hardware cost and *robust* to heterogeneous data.

To this end, we propose *Loss Aggregation Quantum Federated Learning (LossAgg-QFL)*, a resource-efficient and robust framework designed for practical quantum processors. Instead of transmitting large parameter vectors, clients report only scalar losses, and the central server estimates gradients using a multi-directional variant of Simultaneous Perturbation Stochastic Approximation (SPSA) [27], [28]. This design reduces communication and execution costs by orders of magnitude, while enhancing robustness against non-IID data and optimization stochasticity. Our core contributions are:

- **Resource efficiency in quantum federated learning:** LossAgg-QFL reduces communication (uplink data) and executions (circuit evaluations multiplied by shots) to a constant overhead $\mathcal{O}(1)$ per client per training round, independent of qubit count, circuit depth, or model architecture. It lowers device runtime scaling from $\mathcal{O}(n^3)$ to $\mathcal{O}(n)$ with respect to qubit number n [29], [30], yielding up to a $10^6 \times$ total reduction.
- **Robustness to data heterogeneity and stochastic updates:** By aggregating scalar losses rather than high-dimensional gradients, LossAgg-QFL mitigates bias from heterogeneous client updates under non-IID distributions, achieving up to +59% accuracy gains over FedAvg [8] and FedAdam [18] in label-skew scenarios. Furthermore, its multi-directional SPSA strategy reduces gradient-free estimation variance from 3.6% to 0.8%, ensuring stable convergence.

Cheng-En Tsai is with the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan.

Hao-Chung Cheng is with the Department of Electrical Engineering, the Department of Mathematics, and the Center for Quantum Science and Engineering, National Taiwan University, Taipei, Taiwan; he is also with the Hon Hai (Foxconn) Quantum Computing Center, New Taipei City, Taiwan, and the Physics Division, National Center for Theoretical Sciences, Taipei, Taiwan.

*Corresponding author. ORCID: <https://orcid.org/0000-0003-4499-4679>. A full standalone version of this manuscript is available for reference.

II. METHODOLOGY

A. Framework Design

We introduce Loss Aggregation Quantum Federated Learning (LossAgg-QFL), a resource-efficient and robust framework that addresses two fundamental challenges in quantum federated learning: *limited hardware resources* and *non-IID data heterogeneity*. Conventional federated learning frameworks exchange large gradient or parameter tensors between clients and the central server—an approach that is both communication-intensive and incompatible with quantum hardware, where back-propagation is infeasible and circuit execution time dominates overall cost. LossAgg-QFL changes this process by transmitting only *scalar loss values* from each client to the server, aligning optimization directly with hardware realities.

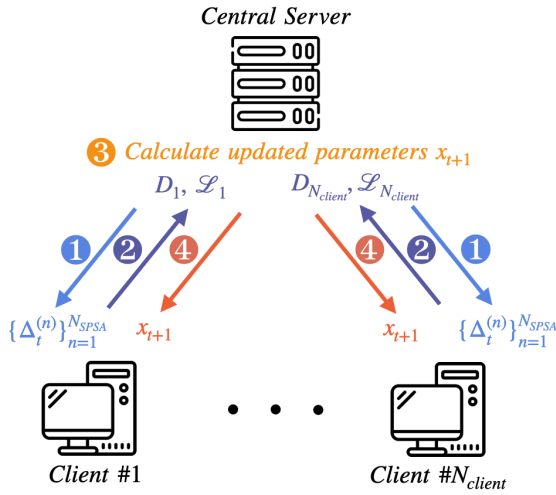


Fig. 1. **Overview of the LossAgg-QFL framework.** (1) The central server samples N_{SPSA} random perturbation directions $\{\Delta_t^{(n)}\}$ and broadcasts them to clients; (2) Each client evaluates perturbed circuit losses $L_i^{(n)\pm} = L(x_t \pm c_t \Delta_t^{(n)})$ strictly on private local datasets D_i , avoiding any local gradient or parameter computations; (3) The server aggregates these into statistically unbiased global averages $\bar{L}_n^{\pm}$ weighted by dataset size; and (4) Operating in a purely gradient-free regime, the server reconstructs the global descent trajectory from the aggregated scalars, executing parameter updates via multi-directional stochastic approximation (**LossAgg**) or tracking dynamic momentum trajectories via adaptive scaling (**LossAggAdam**).

B. Loss Aggregation Mechanism

As illustrated in Fig. 1, each training round in LossAgg-QFL follows a structured client–server protocol:

- 1) **Perturbation sampling:** The server generates N_{SPSA} random perturbation vectors $\{\Delta_t^{(n)}\}$ and broadcasts them to participating clients.
- 2) **Local evaluation:** Each client i , with local dataset size D_i , evaluates perturbed circuit losses at parameter settings $x_t \pm c_t \Delta_t^{(n)}$, returning only the scalar results $\{L_i^{(n)+}, L_i^{(n)-}\}$ and D_i to the server.
- 3) **Loss aggregation:** The server aggregates client responses into global weighted averages,

$$\bar{L}_n^{\pm} = \frac{1}{\sum_i D_i} \sum_i D_i L_i^{(n)\pm},$$

which serve as unbiased estimators of the total loss across all clients.

- 4) **Gradient reconstruction and update:** Using the aggregated results, the server approximates the global gradient as

$$g_t^{(n)} \approx \frac{\bar{L}_n^+ - \bar{L}_n^-}{2c_t} \Delta_t^{(n)-1},$$

and updates parameters x_{t+1} via averaging (LossAgg) or Adam-style momentum (LossAggAdam).

This loss-centric aggregation paradigm eliminates the transmission of high-dimensional gradient or parameter tensors, reducing uplink bandwidth from $\mathcal{O}(p)$ to $\mathcal{O}(1)$ per round. By communicating only scalar quantities, it inherently limits information exposure, mitigating vulnerabilities such as gradient inversion attacks [31], [32] while maintaining statistical correctness and scalability.

C. Multi-SPSA Strategy

To enable efficient and hardware-compatible optimization, LossAgg-QFL adopts a gradient-free approach based on Simultaneous Perturbation Stochastic Approximation (SPSA) [27], [28]. SPSA estimates gradients using only two evaluations of the loss function, irrespective of the number of parameters, providing constant-cost optimization ideally suited to quantum circuits where parameter-shift rules scale as $\mathcal{O}(p)$.

At each iteration, the server samples a random perturbation vector Δ_t whose entries are drawn from a Rademacher distribution (± 1) and approximates the gradient as

$$\hat{g}_{t,i} = \frac{L(x_t + c_t \Delta_t) - L(x_t - c_t \Delta_t)}{2c_t \Delta_{t,i}}, \quad (1)$$

where $L(\cdot)$ denotes the aggregated global loss, $x_t \in \mathbb{R}^p$ the parameter vector, c_t the perturbation magnitude, and $\Delta_{t,i}$ the i -th component of the perturbation vector. This constant-cost property—requiring only two circuit evaluations per update—makes SPSA particularly suitable for large-scale variational quantum models.

To enhance stability, we extend this scheme into a *multi-directional* variant, termed **multi-SPSA**. In each iteration, the server samples N_{SPSA} random perturbation directions,

$$\{\Delta_t^{(n)}\}_{n=1}^{N_{\text{SPSA}}}, \quad (2)$$

collects the corresponding client loss values, and averages the resulting gradient estimates. This simple modification yields three key benefits:

- **Variance reduction:** Averaging over multiple perturbations suppresses stochastic noise in gradient estimation.
- **Accelerated convergence:** Lower gradient variance enables larger effective learning rates and smoother descent trajectories.
- **Noise robustness:** Aggregation across perturbations mitigates instability from quantum sampling fluctuations or transient device noise.

Because each client transmits only constant-sized loss values, adding extra perturbation directions increases the workload by a tunable yet constant factor, preserving the $\mathcal{O}(1)$ communication and execution scaling of LossAgg-QFL. This

extension, therefore, balances the trade-off between efficiency and stability, enabling reliable convergence under realistic quantum-noise conditions.

D. Server-Side Aggregation Strategies

After obtaining multiple SPSA-based updates, the server integrates them using one of two aggregation rules designed to balance simplicity and adaptivity:

- **LossAgg (averaging)** The server averages all SPSA-based updates across perturbation directions, following the FedAvg principle to produce a stable global parameter estimate. This variant prioritizes minimal computation and communication overhead.
- **LossAggAdam (adaptive)** The server incorporates Adam-style momentum and variance correction to dynamically adjust learning rates, accelerating convergence and improving smoothness in noisy regimes.

Both variants consistently outperform baseline QFL optimizers. While adaptive smoothing in LossAggAdam can further stabilize convergence, the plain averaging rule often achieves comparable or superior accuracy—highlighting the intrinsic robustness of the loss-aggregation formulation itself.

Because the framework is modular by design, additional aggregation or optimization algorithms (e.g., second-order methods or trust-region updates) can be readily incorporated. This extensibility ensures that the integration of multiple SPSA directions can be tailored to a wide range of model architectures, training objectives, and hardware configurations.

E. Implementation and Evaluation Protocols

LossAgg-QFL was implemented in Python using Qiskit [33] for quantum circuit simulation and federated orchestration. Experiments were conducted on both small- and large-scale benchmarks: the **Iris** dataset for proof-of-concept testing, and **MNIST** [34] and **Fashion-MNIST (FMNIST)** [35] for large-scale evaluation. For MNIST/FM-NIST, we examined two configurations—**10-MNIST/10-FMNIST** with ten clients and **8-MNIST/8-FMNIST** with eight clients. Each client was trained on mini-batches of size 16 using the full training sets (60k and 48k samples, respectively), and results were reported on a held-out validation set of 1,024 samples.

Data partitioning: We assessed three canonical client-partitioning strategies [17]: (i) *Label distribution skew*, where each client accesses only $\#C \in \{1, 2\}$ classes; (ii) *Quantity skew*, generated from a Dirichlet distribution with concentration parameter $\alpha = 0.5$; and (iii) *IID*, where samples are evenly distributed across clients. These settings emulate real-world heterogeneity in federated systems, ranging from domain specialization to unbalanced client participation.

Encoding and classifier circuits: Classical data were embedded into quantum states via amplitude encoding [36], [37]. For 10-class tasks, images were zero-padded to 32×32 and mapped to 10 qubits; for 8-class tasks, images were resized to 16×16 and mapped to 8 qubits. The quantum classifier circuit, adapted from [38], comprised 30 layers of alternating entangling and rotation blocks—nearest-neighbor CNOTs for

entanglement, followed by parameterized R_X – R_Z – R_X rotations on each qubit. This structure balances expressivity and hardware realism, making it representative of near-term QPU capabilities.

Hyperparameters: The QNN parameter count was defined as $p = cnL$, with n qubits, L layers, and c trainable parameters per qubit per layer. SPSA hyperparameters followed the standard decay rule $a_t = a/(A + t)^\alpha$ and $c_t = c/t^\gamma$. For LossAggAdam, we adopted $\beta_1 = 0.9$, $\beta_2 = 0.99$, and learning rate $\eta = 0.01$.

Evaluation metrics: Performance was evaluated along four dimensions: (i) *Communication volume*, measured as uplink payload per round; (ii) *Circuit executions*, defined as the number of quantum circuits $\times$ measurement shots per round; (iii) *Runtime scaling*, computed as wall-clock device occupation time under layer-wise scheduling; and (iv) *Classification accuracy*, reported as top-1 accuracy with standard deviations across five independent runs.

F. Privacy Considerations

Because LossAgg-QFL transmits only scalar losses rather than gradients or model parameters, it inherently mitigates privacy risks such as gradient inversion and parameter reconstruction. Although explicit privacy mechanisms (e.g., differential privacy or secure aggregation) were not applied in this study, the framework is fully compatible with these techniques. It can seamlessly integrate them in future extensions without altering the core loss-based communication protocol.

G. Algorithmic Workflow

For completeness, Algorithm 1 summarizes the complete training workflow of LossAgg-QFL, from perturbation sampling and client-side loss evaluation to server-side aggregation and parameter updates. This workflow formalizes the principles of constant-cost communication and gradient-free optimization that underpin the framework’s scalability.

H. Data, Materials, and Software Availability

All code used to implement the LossAgg-QFL framework and to reproduce the simulation results reported in this study is publicly available at <https://github.com/xNTsai/LossAgg-QFL>. The MNIST, Fashion-MNIST, and IRIS datasets used in this work are publicly available benchmark datasets and were accessed from their original sources.

III. RESULTS

A. Loss aggregation achieves constant communication cost

We first evaluate uplink communication, a significant bottleneck in federated learning. Consider a quantum neural network (QNN) with n qubits, L layers, and c trainable parameters per qubit per layer, giving a total parameter count of $p = cnL$. In the parameter-shift (PS) framework, each client must transmit gradients for all parameters, leading to a per-round uplink cost of

$$C_{\text{PS}} = \mathcal{O}(p) = \mathcal{O}(cnL). \quad (3)$$

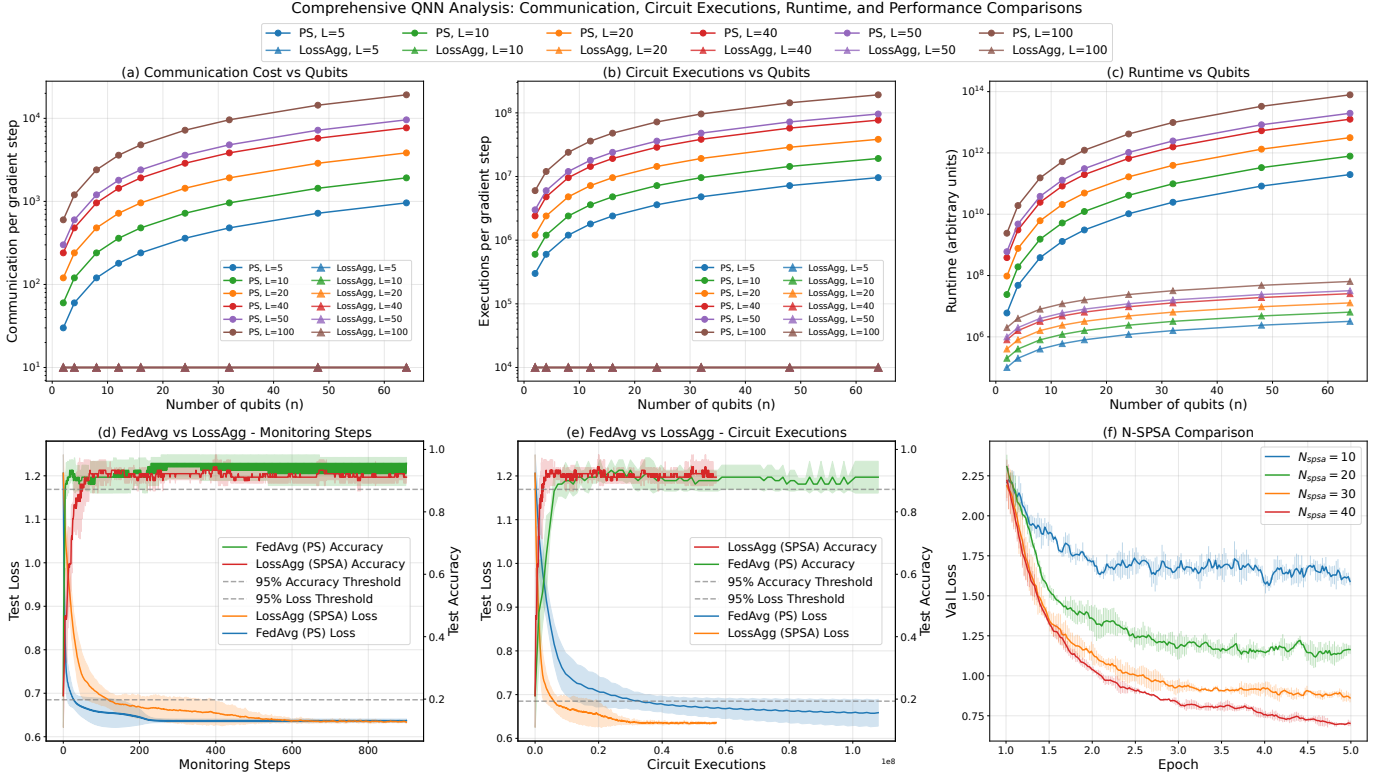


Fig. 2. **Resource efficiency and robustness of LossAgg-QFL.** (a) Uplink communication cost per round (C_{PS} vs. C_{LossAgg}) as a function of circuit width (n) and depth (L), showing that LossAgg maintains constant communication $\mathcal{O}(1)$ compared with $\mathcal{O}(nL)$ in parameter-shift (PS) methods. (b) Total circuit executions N_{PS} and N_{LossAgg} versus model size, demonstrating constant execution cost for LossAgg. (c) Runtime per round (T_{PS} vs. T_{LossAgg}) under layer-wise scheduling, where LossAgg reduces device occupation time by over $10^5\times$ at $(n, L) = (32, 40)$. (d) Convergence trajectories on the 3-class Iris task [39], comparing global steps required to reach 95% accuracy and loss thresholds; SPSA’s slower step-wise convergence is compensated by lower per-step cost. (e) Hardware-normalized efficiency comparison measured in total circuit executions to reach equivalent accuracy and loss, showing $3.2\times$ – $5.7\times$ overall savings for LossAgg. (f) Effect of the multi-directional SPSA strategy ($N_{\text{SPSA}} \in \{10, 20, 30, 40\}$): increasing perturbation directions enhances convergence smoothness and improves mean accuracy from 48.9% to 83.5% while reducing variance from 3.6% to 0.8%. All shaded regions denote $\pm$ one standard deviation over five runs.

In practice, realistic quantum neural networks incur substantially higher communication overheads. To achieve sufficient *expressibility* and global *entanglement*, practical QNN ansätze include parameterized two-qubit gates whose number scales quadratically with the qubit count. Sim *et al.* [40] showed that expressibility grows with both connectivity and depth, requiring parametrized two-qubit gates whose number increases with the qubit count. This implies that the number of trainable parameters per qubit per layer scales as $c = \mathcal{O}(n)$, giving a total parameter budget $p = cnL = \mathcal{O}(n^2L)$. Combined with empirical findings that layer depth L typically increases from logarithmic to linear in n [41], [42], PS-based QFL can further approach $\mathcal{O}(cn^3)$ scaling for deep, densely entangled models.

By contrast, LossAgg transmits only two scalar losses per SPSA direction. With N_{spsa} perturbation directions, the per-round uplink becomes

$$C_{\text{LossAgg}} = 2N_{\text{spsa}} = \mathcal{O}(1), \quad (4)$$

fully independent of n, L . This constant-cost communication decouples network traffic from model complexity, substantially reducing uplink bandwidth.

LossAgg also benefits from reduced precision requirements. While gradient-based schemes must transmit full-precision parameter vectors to ensure stable global aggregation, Los-

sAgg communicates scalar loss values that are inherently tolerant to quantization noise. When averaged over multiple perturbation directions in the multi-SPSA strategy, these values can be aggressively compressed without accuracy degradation, reducing both the number of transmitted entries and their bitwidth.

Figure 2(a) visualizes uplink communication versus qubit count and circuit depth. Each PS curve rises sharply with n , forming parallel slopes that reveal its $\mathcal{O}(nL)$ dependence. For shallow circuits ($L = 5$ – 10), the uplink cost grows from $C_{\text{PS}} = 30$ at $(n, L) = (2, 5)$ to 960 at $(64, 5)$, while deeper models ($L = 50$ – 100) reach 9.6×10^3 – 1.9×10^4 transmitted units, spanning three orders of magnitude. In striking contrast, the LossAgg curve remains perfectly flat at $C_{\text{LossAgg}} = 10$ for $R = 5$, independent of circuit size. At $(n, L) = (64, 50)$, LossAgg reduces total communication by approximately $10^3\times$ compared with PS-based QFL.

These savings directly relieve bandwidth limitations and, more importantly, make LossAgg-QFL practical for deployment on quantum-secured or hybrid quantum-classical networks, where low transmission volume is critical for maintaining fidelity and minimizing decoherence.

Algorithm 1 Loss Aggregation Quantum Federated Learning (LossAgg-QFL and LossAggAdam)

```

1: Initialize: global parameters  $x_0$ ; SPSA parameters
   ( $a, A, c, \alpha, \gamma$ ); number of perturbations  $N_{\text{SPSA}}$ .
2: For LossAggAdam, also initialize  $m_0 = 0, v_0 = 0,$ 
    $\beta_1, \beta_2 \in [0, 1], \eta, \tau$ .
3: for round  $t = 0, \dots, T - 1$  do
4:   Server samples  $\{\Delta_t^{(n)}\}_{n=1}^{N_{\text{SPSA}}}$  (Rademacher  $\pm 1$  entries)
   and broadcasts to clients.
5:   for each client  $i \in \mathcal{S}$  in parallel do
6:     for  $n = 1, \dots, N_{\text{SPSA}}$  do
7:       Client computes perturbed losses  $L_i^{(n)\pm} =$ 
        $L(x_t \pm c_t \Delta_t^{(n)})$ .
8:     end for
9:     Client sends  $\{L_i^{(n)\pm}\}$  and dataset size  $D_i$  to server.
10:   end for
11:   for  $n = 1, \dots, N_{\text{SPSA}}$  do
12:     Server aggregates weighted losses:

$$\bar{L}_n^\pm = \frac{1}{\sum_{i \in \mathcal{S}} D_i} \sum_{i \in \mathcal{S}} D_i L_i^{(n)\pm}.$$

13:     Estimate SPSA update:

$$g_t^{(n)} = \frac{\bar{L}_n^+ - \bar{L}_n^-}{2c_t} (\Delta_t^{(n)})^{-1}, \quad x_t^{(n)} = x_t - a_t g_t^{(n)}.$$

14:   end for
15:   LossAgg update:  $x_{t+1} = \frac{1}{N_{\text{SPSA}}} \sum_{n=1}^{N_{\text{SPSA}}} x_t^{(n)}.$ 
16:   LossAggAdam update:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1)(x_{t+1} - x_t),$$


$$v_t = \beta_2 v_{t-1} + (1 - \beta_2)(x_{t+1} - x_t)^2,$$


$$\hat{m}_t = m_t / (1 - \beta_1^t), \quad \hat{v}_t = v_t / (1 - \beta_2^t),$$


$$x_{t+1} = x_t + \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \tau}.$$

17: end for

```

B. Execution and runtime savings on quantum hardware

Beyond communication, we next examine circuit executions and quantum runtime—the dominant contributors to physical hardware cost [29], [30]. Let S denote the number of measurement shots per circuit evaluation and R the number of local update directions per round (for PS, these align with local epochs; for SPSA, they correspond to perturbation directions).

In the parameter-shift (PS) framework, every trainable parameter requires two circuit evaluations per direction, yielding a per-client, per-round execution count of

$$N_{\text{PS}} = 2cnLR S = \mathcal{O}(cnL). \quad (5)$$

As discussed earlier, practical QNNs designed for sufficient expressibility and entanglement often employ parameterized two-qubit gates whose number scales quadratically with qubit count, resulting in $p = \mathcal{O}(cn^2L)$. In contrast, LossAgg performs only two circuit evaluations per direction, entirely

independent of parameter count:

$$N_{\text{LossAgg}} = 2RS = \mathcal{O}(1). \quad (6)$$

Figure 2(b) clearly illustrates this difference. Each PS curve, corresponding to a fixed layer depth, rises sharply with qubit number, forming a fan-shaped bundle of nearly parallel trajectories that reveal its $\mathcal{O}(nL)$ scaling. The LossAgg curve, by comparison, appears as a perfectly flat line at 10,000 executions across all (n, L) configurations, positioned several orders of magnitude below the PS surfaces. At moderate circuit complexity $(n, L) = (32, 40)$, PS already exceeds 10^7 total executions, while LossAgg remains fixed—representing a dramatic $10^6\times$ reduction in total circuit workload.

Here, *executions* count the number of distinct circuit instances multiplied by the number of measurement shots per instance, a standard metric in cloud-based QPU benchmarking. Although we assume a constant S for clarity, in practice the shot number often increases with n to preserve estimation precision as measurement variance scales with Hilbert-space dimension [43], [44], further improving LossAgg’s efficiency advantage.

Runtime differences are even more striking. Let T_{PS} and T_{LossAgg} denote the total quantum execution time per round, the duration which the processor performs gate operations and measurements, excluding classical scheduling overhead. Given that the gate depth per layer D typically scales with n for nearest-neighbor entangling layouts, we have:

$$T_{\text{PS}} = \mathcal{O}(nL^2D^2) = \mathcal{O}(n^3L^2), \quad (7)$$

$$T_{\text{LossAgg}} = \mathcal{O}(LD) = \mathcal{O}(nL). \quad (8)$$

Figure 2(c) clearly visualizes these trends. The PS runtime surfaces rise steeply with both qubit number and layer depth, approaching 10^{12} to 10^{13} quantum-time units for large-scale circuits. In contrast, the LossAgg trace remains nearly horizontal near 10^7 , forming a flat baseline far below the escalating PS region. At $(n, L) = (32, 40)$, PS-based QFL consumes over 10^{12} time units, compared to only 1.28×10^7 for LossAgg—over $10^5\times$ shorter active runtime.

Such reductions directly translate to dramatically shorter queue delays and higher throughput on shared quantum hardware, enabling efficient scaling to large federated systems.

C. Resource efficiency at equal accuracy

The preceding analyses quantify per-round savings; we now examine whether these advantages persist when both frameworks reach comparable model performance. Experiments were conducted on the 3-class Iris classification task, with target thresholds defined at 95% of the best mean accuracy (0.872) and validation loss (0.685).

Figure 2(d) traces the global training progress. FedAvg (PS) attains the accuracy and loss thresholds within 6 and 28 global rounds, respectively, while LossAgg (SPSA) requires 45 and 118 rounds. This slower round-wise convergence reflects the stochastic nature of SPSA, which deliberately trades per-step precision for constant-cost scalability. The curves in Fig. 2(d) illustrate this contrast: the FedAvg trajectory rises sharply and stabilizes early, whereas the LossAgg trajectory advances

more gradually, with steady improvement toward the same asymptote.

However, global round count is not an accurate measure of hardware cost. Federated quantum workloads are dominated by circuit executions and on-device runtime [29], [30], not by the number of optimization rounds. Thus, a fair comparison must consider the total number of circuit executions rather than the iteration count. When evaluated under this hardware-relevant metric, the comparison reverses decisively.

As shown in Fig. 2(e), FedAvg (PS) requires 7.20×10^6 executions to reach the accuracy threshold and 3.36×10^7 for the loss threshold. LossAgg, by contrast, achieves the same results using only 2.25×10^6 and 5.9×10^6 executions, reducing hardware usage by approximately 69% and 82%, respectively. Equivalently, for matched final accuracy and loss, LossAgg completes training with less than one-third the number of quantum executions and in a fraction of the runtime budget.

Together, these findings confirm that LossAgg’s constant-cost principle extends beyond theoretical scaling to practical end-to-end efficiency. Even when accounting for slower round-wise convergence, LossAgg delivers substantially lower total resource consumption, establishing it as a hardware-efficient and scalable paradigm for realistic quantum training workloads.

D. Multi-directional SPSA stabilizes convergence

LossAgg employs Simultaneous Perturbation Stochastic Approximation (SPSA) to estimate gradients with only two loss evaluations per update, enabling constant-cost optimization independent of parameter count. However, single-direction SPSA can exhibit high stochastic variance, slowing and destabilizing convergence under quantum sampling noise. To mitigate this, we introduce a *multi-directional* extension that samples and averages across N_{SPSA} random perturbation directions per round. This averaging smooths gradient estimates, supports larger learning rates, and enhances robustness to measurement fluctuations.

Table I quantifies these gains: increasing N_{SPSA} from 10 to 40 improves final accuracy from 48.9% to 83.5% while reducing the standard deviation from 3.6% to 0.8%. Figure 2(f) mirrors this trend visually. At low N_{SPSA} , the loss trajectories oscillate heavily, with sharp fluctuations and erratic minima; as N_{SPSA} increases, these curves become progressively smoother and converge monotonically toward a stable optimum, forming well-damped, overlapping traces. This clear visual transition underscores the stabilizing role of multi-directional averaging.

Importantly, increasing N_{SPSA} scales the workload only by a constant factor in both N_{LossAgg} and C_{LossAgg} , preserving the framework’s constant-cost property. By combining constant-cost optimization with tunable variance suppression, multi-directional SPSA transforms LossAgg-QFL from a merely efficient algorithm into one that is both *stable* and *hardware-resilient*, maintaining scalability without sacrificing convergence quality.

E. Robustness across non-IID distributions

Federated learning systems must remain reliable even when client data are highly imbalanced or non-identically dis-

TABLE I
CLASSIFICATION ACCURACY COMPARISON FOR DIFFERENT N_{SPSA} VALUES

N_{SPSA}	10	20	30	40
Accuracy (%)	48.9 ± 3.6	70.6 ± 1.6	78.9 ± 1.1	83.5 ± 0.8

Values indicate mean classification accuracy with standard deviation.

tributed—a challenge known as *non-IID heterogeneity*. Such heterogeneity often induces gradient bias and destabilizes convergence across clients. LossAgg-QFL inherently mitigates these effects by aggregating scalar losses instead of client-specific gradients, thereby eliminating dependence on parameter dimensionality and reducing bias from uneven data distributions. The integrated multi-directional SPSA further suppresses stochastic fluctuations, reinforcing stability under both heterogeneous and noisy training conditions.

We assess robustness under three representative data-partitioning schemes [17]:

- **Label Distribution Skew:** Each client holds samples from only one or two classes (e.g., limited MNIST digits), mimicking domain-restricted settings such as hospitals or sensors specialized for specific categories.
- **Quantity Skew:** Client dataset sizes are imbalanced, drawn from a Dirichlet distribution $q \sim \text{Dir}(0.5)$, reflecting uneven participation frequencies.
- **Homogeneous Partition (IID baseline):** Data are uniformly and class-balanced across clients, serving as an upper-bound reference.

Figure 3 visualizes convergence across these regimes. In label-skew cases (Figs. 3a,b), FedAvg and FedAdam exhibit pronounced oscillations and plateau early, while both LossAgg variants show steady, monotonic decline toward lower loss values. Under quantity skew (Fig. 3c), the same pattern persists—LossAgg curves remain smooth and stable with minimal fluctuation, whereas gradient-based baselines continue to waver. Even in the IID baseline (Fig. 3d), where all methods eventually converge, LossAgg reaches the lowest final loss, confirming its consistent optimization behavior. The visual separation between the erratic FedAvg/FedAdam traces and the damped, descending LossAgg trajectories underscores its robustness across heterogeneous data settings.

Quantitatively, Table II reports classification accuracies under each partition. In label-distribution skew, LossAgg surpasses FedAvg by up to **+59%** and FedAdam by **+48.1%**. For quantity skew, it delivers a **+33.8%** accuracy gain with markedly smoother convergence. Even under balanced IID conditions, LossAgg matches or slightly exceeds baseline performance, demonstrating robustness without trade-offs.

Between the two internal variants, both LossAgg and LossAggAdam outperform all gradient-based baselines. While the Adam extension provides marginally steadier curves, its overall improvement is limited—showing that LossAgg alone captures the essential stabilizing mechanism while remaining compatible with further optimizer integration.

Overall, LossAgg-QFL maintains high accuracy and stable convergence across the full spectrum of federated het-

TABLE II
TOP-1 ACCURACY (%) COMPARISON OF DIFFERENT OPTIMIZATION METHODS ACROSS DATASETS AND PARTITIONING STRATEGIES. VALUES ARE MEAN $\pm$ STANDARD DEVIATION.

Category	Dataset	Partitioning	FedAvg	FedAdam	LossAgg	LossAggAdam
Label distribution skew	8-MNIST	$\#C = 1$	22.4 ± 2.8	33.3 ± 7.7	81.4 ± 2.4	81.4 ± 1.8
		$\#C = 2$	26.3 ± 8.6	46.4 ± 5.4	83.5 ± 0.8	80.8 ± 1.6
	10-MNIST	$\#C = 1$	18.6 ± 3.8	30.9 ± 2.7	68.8 ± 0.9	70.1 ± 3.1
		$\#C = 2$	33.9 ± 6.2	47.8 ± 2.5	71.8 ± 0.4	65.9 ± 1.7
	8-FMNIST	$\#C = 1$	21.5 ± 3.2	27.8 ± 12.3	66.8 ± 1.6	66.1 ± 0.6
		$\#C = 2$	23.9 ± 6.1	41.3 ± 3.6	64.8 ± 0.4	64.8 ± 1.0
	10-FMNIST	$\#C = 1$	20.5 ± 6.0	31.4 ± 4.9	63.3 ± 1.7	62.0 ± 1.7
		$\#C = 2$	28.3 ± 3.4	41.2 ± 5.2	64.3 ± 0.8	61.3 ± 3.9
Quantity skew	8-MNIST	$q \sim \text{Dir}(0.5)$	46.5 ± 2.2	69.0 ± 2.9	80.3 ± 0.8	78.8 ± 1.1
	10-MNIST	$q \sim \text{Dir}(0.5)$	55.4 ± 2.5	61.4 ± 0.5	67.5 ± 0.6	67.7 ± 3.0
	8-FMNIST	$q \sim \text{Dir}(0.5)$	45.5 ± 4.8	59.4 ± 1.1	64.4 ± 0.6	64.9 ± 1.1
	10-FMNIST	$q \sim \text{Dir}(0.5)$	58.0 ± 3.2	59.0 ± 2.1	63.0 ± 1.5	61.3 ± 1.8
Homogeneous (IID)	8-MNIST	IID	47.0 ± 7.3	72.1 ± 2.0	80.2 ± 0.7	80.5 ± 1.8
	10-MNIST	IID	54.1 ± 2.7	61.4 ± 4.3	68.5 ± 1.6	67.3 ± 3.5
	8-FMNIST	IID	44.6 ± 1.7	59.4 ± 0.5	63.8 ± 0.7	65.3 ± 0.7
	10-FMNIST	IID	56.1 ± 3.4	58.9 ± 5.9	63.3 ± 0.9	59.0 ± 3.2

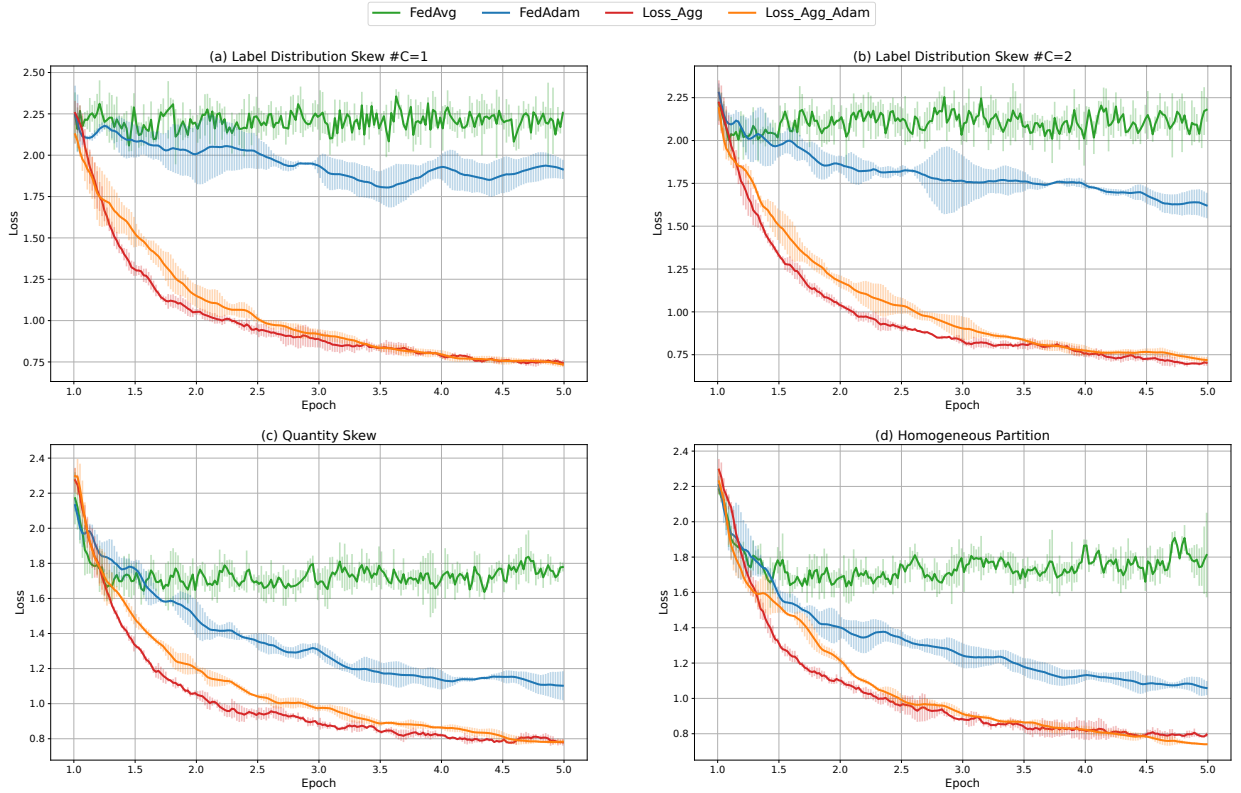


Fig. 3. **Validation loss comparison across partitioning strategies in quantum federated learning.** (a,b) Loss trajectories under label distribution skew with $\#C = 1$ and $\#C = 2$ classes per client, (c) quantity skew simulated via Dirichlet distribution with $\alpha = 0.5$, and (d) homogeneous (IID) partition. Across all settings, LossAgg and LossAggAdam converge faster and reach lower validation loss than FedAvg and FedAdam, particularly under heterogeneous non-IID conditions. Adam smoothing provides additional stability but comparable final loss to pure LossAgg, confirming that loss aggregation inherently mitigates divergence and improves optimization consistency. Shaded regions denote $\pm$ one standard deviation over five independent runs.

erogeneity. Its resilience arises from two complementary principles: (i) scalar-loss aggregation that eliminates gradient bias from unbalanced updates, and (ii) multi-directional SPSA averaging that suppresses stochastic variance. Together, these enable uniform, reliable training across diverse client distributions—demonstrating that LossAgg-QFL is not only communication- and computation-efficient, but also fundamen-

tally robust to real-world non-IID challenges.

IV. CONCLUSION

This study tested the hypothesis that reframing federated learning around *loss aggregation* can simultaneously improve resource efficiency and robustness for quantum models. Across benchmarks, LossAgg reduces communication from $\mathcal{O}(p)$ to

$\mathcal{O}(1)$, lowers circuit executions from $\mathcal{O}(p)$ to $\mathcal{O}(1)$, and compresses runtime scaling from $\mathcal{O}(nL^2)$ to $\mathcal{O}(L)$ —while maintaining stability on non-IID data through multi-directional SPSA. The central insight is that the *unit of communication* matters: transmitting scalar losses aligns algorithmic updates with hardware realities in which circuit evaluations and device time dominate cost. Notably, these advantages hold even at equal model accuracy, where LossAgg reduces total circuit executions by 69 to 82% despite requiring more global rounds, confirming that the end-to-end hardware cost remains substantially lower.

These properties make LossAgg well-suited to emerging modular and distributed quantum systems, where multiple processors operate in parallel within shared cryogenic modules or across network links. By minimizing uplink payloads and QPU occupation time, our approach eases orchestration across heterogeneous nodes and aligns naturally with secure aggregation or differential privacy techniques that operate on scalar statistics.

There are, however, important avenues for future work. First, expanded hardware trials will help characterize behavior under device noise, calibration drift, and time-varying error rates. Second, integrating formal privacy guarantees (e.g., differential privacy with loss clipping and noise addition) and secure aggregation tailored to quantum measurement statistics remains an open direction. Third, extending loss-based aggregation to hybrid quantum–classical models and studying entanglement-assisted variants of federated optimization could further broaden applicability.

Overall, these results suggest that loss-centric communication, coupled with stabilized gradient-free updates, offers a principled route to scalable, reliable quantum federated learning—well matched to the constraints and opportunities of contemporary quantum hardware.

ACKNOWLEDGMENTS

This project is supported by the Emerging Young Scholars Program of the National Science and Technology Council, Taiwan (R.O.C.) under Grants No. NSTC 114-2628-E-002-006, NSTC 114-2119-M-001-002, and NSTC 114-2124-M-002-003, by the Yushan Young Scholar Program of the Ministry of Education, Taiwan (R.O.C.) under Grants No. NTU-114V2016-1 and by the research project ‘Forefront Quantum Computing, Learning, and Engineering in Noisy Intermediate-Scale Quantum Era’ of National Taiwan University under Grant NTU-114L895005. H.-C. Cheng acknowledges the support from the ‘Center for Advanced Computing and Imaging in Biomedicine (NTU-114L900702)’ through The Featured Areas Research Center Program within the framework of the Higher Education Sprout Project by the Ministry of Education (MOE) in Taiwan.

REFERENCES

- [1] Rui Huang, Xiaoqing Tan, and Qingshan Xu. Quantum federated learning with decentralized data. *IEEE Journal of Selected Topics in Quantum Electronics*, 28(4): Mach. Learn. in Photon. Commun. and Meas. Syst.):1–10, 2022.
- [2] Qi Xia and Qun Li. Quantumfede: A federated learning framework for collaborative quantum training. In *2021 IEEE Global Communications Conference (GLOBECOM)*, pages 1–6, 2021.
- [3] Chao Ren, Rudai Yan, Huihui Zhu, Han Yu, Minrui Xu, Yuan Shen, Yan Xu, Ming Xiao, Zhao Yang Dong, Mikael Skoglund, Dusit Niyato, and Leong Chuan Kwek. Toward quantum federated learning. *IEEE Transactions on Neural Networks and Learning Systems*, 36(9):15580–15600, 2025.
- [4] Mahdi Chehimi and Walid Saad. Quantum federated learning with quantum data. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8617–8621, 2022.
- [5] John Preskill. Quantum Computing in the NISQ era and beyond. *Quantum*, 2:79, August 2018.
- [6] Shubham, Prachi Sajwan, and N. Jayapandian. Challenges and opportunities: Quantum computing in machine learning. In *2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, pages 598–602, 2019.
- [7] Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3):50–60, 2020.
- [8] H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data, 2023.
- [9] Muhammad AbuGhanem. Ibm quantum computers: evolution, performance, and future directions. *The Journal of Supercomputing*, 81(5), April 2025.
- [10] D. Main, P. Drmota, D. P. Nadlinger, E. M. Ainley, A. Agrawal, B. C. Nichol, R. Srinivas, G. Araneda, and D. M. Lucas. Distributed quantum computing across an optical network link. *Nature*, 638(8050):383–388, February 2025.
- [11] Osama Shahid, Seyedamin Pouriyeh, Reza M. Parizi, Quan Z. Sheng, Gautam Srivastava, and Liang Zhao. Communication efficiency in federated learning: Achievements and challenges, 2021.
- [12] Harashta Tatimma Larasati, Muhammad Firdaus, and Howon Kim. Quantum federated learning: Remarks and challenges. In *2022 IEEE 9th International Conference on Cyber Security and Cloud Computing (CSCloud)/2022 IEEE 8th International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, pages 1–5, 2022.
- [13] Changhao Li, Niraj Kumar, Zhixin Song, Shouvanik Chakrabarti, and Marco Pistoia. Privacy-preserving quantum federated learning via gradient hiding. *Quantum Science and Technology*, 9(3):035028, May 2024.
- [14] Zhonghui Li, Kaiping Xue, Jian Li, Lutong Chen, Ruidong Li, Zhaoying Wang, Nenghai Yu, David S. L. Wei, Qibin Sun, and Jun Lu. Entanglement-assisted quantum networks: Mechanics, enabling technologies, challenges, and research directions. *IEEE Communications Surveys & Tutorials*, 25(4):2133–2189, 2023.
- [15] Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Cavin, and Vikas Chandra. Federated learning with non-iid data, 2018.
- [16] Kevin Hsieh, Amar Phanishayee, Onur Mutlu, and Phillip B. Gibbons. The non-iid data quagmire of decentralized machine learning, 2020.
- [17] Qibin Li, Yiqun Diao, Quan Chen, and Bingsheng He. Federated learning on non-iid data silos: An experimental study. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*, pages 965–978, 2022.
- [18] Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečný, Sanjiv Kumar, and H. Brendan McMahan. Adaptive federated optimization, 2021.
- [19] Xiaodong Ma, Jia Zhu, Zhihao Lin, Shanxuan Chen, and Yangjie Qin. A state-of-the-art survey on solving non-iid data in federated learning. *Future Generation Computer Systems*, 135:244–258, 2022.
- [20] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.
- [21] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986.
- [22] Benjamin Schumacher and Michael Westmoreland. *Quantum processes systems, and information*. Cambridge University Press, 2010.
- [23] W. K. Wootters and W. H. Zurek. A single quantum cannot be cloned. *Nature*, 299(5886):802–803, 1982.
- [24] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii. Quantum circuit learning. *Physical Review A*, 98(3), September 2018.
- [25] Maria Schuld, Ville Bergholm, Christian Gogolin, Josh Izaac, and Nathan Killoran. Evaluating analytic gradients on quantum hardware. *Physical Review A*, 99(3), March 2019.

- [26] Maria Schuld, Ville Bergholm, Christian Gogolin, Josh Izaac, and Nathan Killoran. Evaluating analytic gradients on quantum hardware. *Physical Review A*, 99(3), March 2019.
- [27] James C. Spall. An overview of the simultaneous perturbation method for efficient optimization. *Johns Hopkins Apl Technical Digest*, 19:482–492, 1998.
- [28] James C. Spall. Implementation of the simultaneous perturbation algorithm for stochastic optimization. *IEEE Transactions on Aerospace and Electronic Systems*, 34:817–823, 1998.
- [29] Gokul Subramanian Ravi, Tan Nguyen, Robert Rines, Gideon Chern, Thomas O’Brien, and Jay M. Gambetta. Quantum computing in the cloud: Analyzing job and machine characteristics. *arXiv preprint arXiv:2203.13121*, 2022. Analyzes large-scale IBM Quantum workloads and execution time metrics in cloud settings.
- [30] Darshana Lall, Stephen Kelly, Neil Bingham, Alexander Cowtan, Jaspreet Singh, and Ross Duncan. A review and collection of metrics and benchmarks for quantum computers: Definitions, methodologies, and software. Technical report, University of Strathclyde, 2025. Provides standardized runtime and performance metrics for quantum processors.
- [31] Ligeng Zhu, Zhijian Liu, and Song Han. Deep leakage from gradients. *Advances in neural information processing systems*, 32, 2019.
- [32] Jonas Geiping, Hartmut Bauermeister, Hannah Dröge, and Michael Moeller. Inverting gradients-how easy is it to break privacy in federated learning? *Advances in neural information processing systems*, 33:16937–16947, 2020.
- [33] Ali Javadi-Abhari, Matthew Treinish, Kevin Krsulich, Christopher J. Wood, Jake Lishman, Julien Gacon, Simon Martiel, Paul D. Nation, Lev S. Bishop, Andrew W. Cross, Blake R. Johnson, and Jay M. Gambetta. Quantum computing with Qiskit, 2024.
- [34] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- [35] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.
- [36] Maria Schuld and Francesco Petruccione. Supervised learning with quantum computers. *Quantum science and technology (Springer, 2018)*, 2018.
- [37] Manuela Weigold, Johanna Barzen, Frank Leymann, and Marie Salm. Data encoding patterns for quantum computing. In *Proceedings of the 27th Conference on Pattern Languages of Programs*, pages 1–11, 2020.
- [38] Haimeng Zhao. Non-iid quantum federated learning with one-shot communication complexity. *Quantum Machine Intelligence*, 5(1), January 2023.
- [39] R. A. Fisher. Iris. UCI Machine Learning Repository, 1936. DOI: <https://doi.org/10.24432/C56C76>.
- [40] Sukin Sim, Peter D. Johnson, and Alán Aspuru-Guzik. Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms. *Advanced Quantum Technologies*, 2(12), October 2019.
- [41] Maria Schuld, Ryan Sweke, and Johannes Jakob Meyer. Effect of data encoding on the expressive power of variational quantum-machine-learning models. *Physical Review A*, 103(3), March 2021.
- [42] M. Cerezo, Akira Sone, Tyler Volkoff, Lukasz Cincio, and Patrick J. Coles. Cost function dependent barren plateaus in shallow parametrized quantum circuits. *Nature Communications*, 12(1), March 2021.
- [43] Ryan Shaffer, Lucas Kocia, and Mohan Sarovar. Surrogate-based optimization for variational quantum algorithms. *Physical Review A*, 107(3), March 2023.
- [44] Masaya Kohda, Ryosuke Imai, Keita Kanno, Kosuke Mitarai, Wataru Mizukami, and Yuya O. Nakagawa. Quantum expectation-value estimation by computational basis sampling. *Physical Review Research*, 4(3), September 2022.