Quantum Quandaries: Unraveling Encoding Vulnerabilities in Quantum Neural Networks

***Abstract*—Quantum computing (QC) has the potential to revolutionize fields like machine learning, security, and healthcare. Quantum machine learning (QML) has emerged as a promising area, enhancing learning algorithms using quantum computers. However, QML models are lucrative targets due to their high training costs and extensive training times. The scarcity of quantum resources and long wait times further exacerbate the challenge. Additionally, QML providers may rely on third-party quantum clouds for hosting models, exposing them and their training data to potential threats. As QML-as-a-Service (QML-LaaS) becomes more prevalent, reliance on third-party quantum clouds poses a significant security risk. This work demonstrates that adversaries in quantum cloud environments can exploit white-box access to QML models to infer the user's encoding scheme by analyzing circuit transpilation artifacts. The extracted data can be reused for training clone models or sold for profit. We validate the proposed attack through simulations, achieving high accuracy in distinguishing between encoding schemes. We report that ≈95% of the time, the encoding can be predicted correctly. To mitigate this threat, we propose a transient obfuscation layer that masks encoding fingerprints using randomized rotations and entanglement, reducing adversarial detection to near-random chance ≈42%, with a depth overhead of ≈8.5% for a 5-layer QNN design.**

***Index Terms*—QML Security, Untrusted Quantum Cloud, Encoding, Transpilation artifacts**

I. Introduction

Quantum computing (QC) is gaining attention for its potential to revolutionize problem-solving across various fields. By utilizing quantum properties like superposition, entanglement, and interference, QC offers significant speedups for certain tasks, surpassing classical computing capabilities. With potential applications in machine learning [1], [2], security [3], drug discovery [4], optimization [5], [6], finance [7], and healthcare [8], quantum computing is becoming increasingly important in both academia and industry.

Noisy Intermediate-Scale Quantum (NISQ) devices, characterized by a limited number of qubits and susceptibility to noise, face challenges such as restricted qubit connectivity, gate errors, and decoherence, leading to inaccuracies in computation. These limitations hinder the direct implementation of large-scale quantum algorithms, prompting the exploration of hybrid approaches like the Quantum Approximate Optimization Algorithm (QAOA) and Variational Quantum Eigensolver (VQE), which combine classical optimization with quantum subroutines to mitigate noise effects. In this emergent field, quantum machine learning (QML) has gained considerable attention, aiming to improve learning algorithms by leveraging quantum capabilities. Various QML models have been explored, including quantum support vector machines (QSVMs) [9], quantum convolutional neural networks (QCNNs) [1], and quantum generative adversarial networks (QGANs) [10]. Among these, quantum neural networks (QNNs) [11] are notable for replicating the structure and function of classical neural networks within a quantum framework. QNNs utilize parameterized quantum circuits (PQCs) composed of trainable single-qubit and two-qubit gates, with their parameters optimized using classical optimizers. By encoding data into quantum states through methods such as amplitude, angle, or basis encoding, QNNs leverage superposition and entanglement to process information in ways that classical systems cannot replicate. However, optimizing PQCs poses significant challenges due to high computational complexity, resource constraints, and lengthy execution times.

Motivation: The practical realization of this transformative potential of quantum neural networks (QNNs) largely depends on cloud-based quantum services, where users submit circuits to remote hardware—a dependency that introduces critical, yet understudied, security vulnerabilities. Leading platforms such as IBM [12], Google [13], and AWS Braket [14] provide scalable and accessible quantum computing services; however, they face challenges like job submission latency, queue congestion, and high operational costs. The reliance on cloud infrastructure is driven by the exorbitant costs and specialized requirements of quantum hardware. Training QML models incurs significant expenses, with leading platforms like IBM and IonQ charging up to $1.6 per second for superconducting qubits and $0.01 per shot for trapped-ion systems—orders of magnitude costlier than classical alternatives. Compounding this, the iterative nature of hybrid quantum-classical algorithms leads to prolonged training times, often spanning months due to scarce quantum resources and queue congestion on cloud platforms. These factors render trained QML models exceptionally valuable intellectual property (IP), incentivizing adversarial attacks aimed at stealing circuit architectures, optimized parameters, or embedded training data. As the quantum ecosystem expands, third-party providers offering "Quantum Machine Learning as a Service" (QMLaaS) [15] further exacerbate these risks. Services like Orquestra [16] and tKet [17] facilitate multi-hardware integration, while Baidu's "Liang Xi" [18] offers flexible quantum access via mobile, desktop, and cloud interfaces. While trusted hardware remains the preferred choice for applications with high economic or social stakes, hybrid quantum-classical algorithms often incur significant expenses and delays due to the numerous iterations required. Although governments and large enterprises may possess dedicated quantum resources, they are costly and ge-

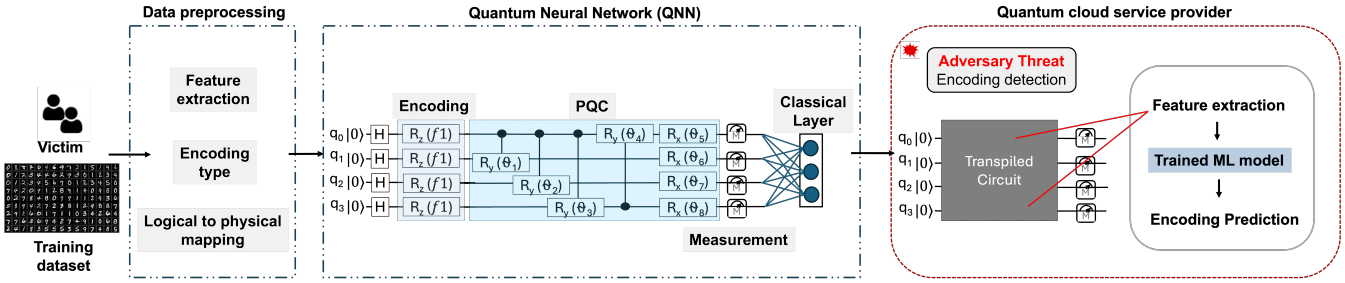

Fig. 1. Proposed attack model where the adversary, posing as a reliable quantum cloud service provider, uses the white box access to the QNN submitted to a quantum cloud to identify the type of encoding.

ographically constrained. This growing reliance on third-party compilers, hardware ecosystems, and cloud services introduces pressing concerns regarding the reliability and security of quantum computations.

A QNN comprises of a data encoding circuit that transforms data into quantum states, a parameterized quantum circuit (PQC) with tunable parameters, and measurement operations for extracting information. Encoding methods, such as basis, amplitude, or angle encoding, are crucial as they determine how input data is represented in the quantum system. These methods imprint distinct structural signatures in transpiled circuits, such as characteristic gate sequences, rotation angle distributions, and entanglement patterns. Malicious cloud providers with white-box access, acting as adversaries, can exploit these artifacts to reverse-engineer proprietary encoding schemes, enabling them to infer sensitive details about the input data, model architecture, and even training objectives. Such breaches compromise not only intellectual property but also data privacy, as encoding methods often embed domain-specific preprocessing critical to a model's functionality. Several studies have addressed protecting quantum circuits from untrusted clouds [19], [20], however they focus on protecting computation outputs or mitigating hardware-level attacks, overlooking the risks posed by encoding side channels. This gap leaves users vulnerable to adversaries who can simply inspect transpiled circuits and steal proprietary encoding techniques or reconstruct training data. *This paper to our knowledge is the very first attempt which aims to understand and mitigate this security vulnerability.*

Contributions: In this work, we: (a) propose a novel framework for classifying popular quantum encoding schemes (i.e., angle, amplitude, and basis encodings) through transpilation patterns, (b) demonstrate the effectiveness of our approach using a comprehensive dataset of circuits generated with varied encoding types and parameterized quantum circuits (PQCs), (c) introduce a multi-level feature extraction pipeline that captures discriminative patterns from transpiled circuits, including structural signatures, rotation analysis, and entanglement metrics, (d) validate our methods through simulations, achieving high accuracy in distinguishing between encoding schemes, (e) propose a lightweight, transient obfuscation layer that masks encoding fingerprints using randomized rotations and entanglement, reducing adversarial detection to near-random chance.

Paper organization: Section II provides background information. Section III outlines the threat model and section IV presents the proposed defense. Section V covers results and Section VI concludes the paper.

II. Background

A. Quantum Neural Network (QNN)

A QNN consists of three main components (Fig. 1): (i) a data encoding circuit that maps classical data into quantum states, (ii) a parameterized quantum circuit (PQC) with tunable parameters, and (iii) measurement operations to extract useful information from the quantum system. Due to the limited number of qubits in current quantum devices, classical preprocessing techniques such as Principal Component Analysis (PCA) are often employed to reduce the dimensionality of input data. The reduced data is then encoded into quantum states using methods like basis encoding, amplitude encoding, or angle encoding. The PQC is the core trainable component of a QNN. It consists of a sequence of quantum gates with adjustable parameters designed to recognize patterns in data and solve specific problems. Finally, measurement operations collapse the qubit states to either 0 or 1. The expectation value of Pauli-Z is used to determine the average state of the qubits. These measured values are typically fed into a classical neuron layer, with the number of neurons equal to the number of classes in the dataset. This layer performs the final classification task. A classical optimizer optimizes the parameters iteratively to achieve the desired input-output relationship.

B. Quantum Cloud Services

Quantum computers require extensive and expensive infrastructure, such as cryogenic coolers and superconducting wires, making direct access challenging. Quantum cloud services like IBM [12], Google [13], and AWS Braket [14] provide remote access, simplifying system management. Users send their quantum circuits to these services, specifying target hardware and metadata.

C. Data Encoding

Data encoding is a critical step in quantum machine learning (QML) and quantum neural networks (QNNs), as it transforms classical data into quantum states, enabling quantum algorithms to process information efficiently. The choice of encoding method significantly impacts the performance, expressivity, and scalability of quantum models. Some widely used encoding schemes are:

1) Basis Encoding: Maps classical data directly to the computational basis states of qubits. In this approach, each classical data point is represented as a binary string, where each bit in the string corresponds to a qubit state. For example, consider a classical data vector $[1, 0, 1]$, which can be encoded into the quantum state $|101\rangle$. This encoding method is particularly advantageous for representing discrete or categorical data, as it provides a straightforward and efficient mapping from classical to quantum representation. However, a major drawback is that the number of qubits required grows linearly with the size of the dataset, which poses scalability challenges when dealing with high-dimensional data.

2) Angle Encoding: Embeds classical data into the rotational angles of qubits on the Bloch sphere. In this approach, a classical feature x_i is represented as a rotation along a specific axis using quantum gates such as $R_x(x_i)$, $R_y(x_i)$, or $R_z(x_i)$. This method allows for the encoding of continuous data into quantum states efficiently. Additionally, dense angle encoding can encode multiple features per qubit by incorporating additional phase gates, enabling more compact representations. A key advantage of angle encoding is its efficiency in terms of gate depth and the reduced number of qubits required compared to basis encoding, making it particularly suitable for continuous-valued data.

3) Amplitude Encoding: Utilizes the amplitudes of quantum states to represent normalized classical data. Given a classical vector $x = [x_1, x_2, \dots]$, it is transformed into the quantum state $|\psi\rangle = \sum_i x_i |i\rangle$, where each amplitude x_i corresponds to a feature in the data. This encoding scheme is highly space-efficient, as it can represent 2^n features using only n qubits, making it particularly advantageous for large datasets. Amplitude encoding is primarily limited by its high circuit complexity, requiring intricate quantum operations for precise state preparation.

4) Hybrid and Advanced Encoding Techniques: Recent research has explored hybrid approaches that combine angle and amplitude encoding to leverage their respective strengths. These hybrid encoding methods aim to enhance information representation while addressing the limitations inherent to individual techniques.

D. Related works

The security of machine learning systems has been extensively studied in classical computing, with attacks ranging from side-channel exploits on hardware accelerators to black-box model extraction via adversarial queries [21] [22]. Recent work has extended reverse engineering (RE) to quantum circuits, using lookup tables (LUTs) to map transpiled gate sequences to original QML architectures [23]. By analyzing rotation gate ordering and entanglement patterns, adversaries can infer circuit parameters, exposing proprietary model designs. Building on this, [24] demonstrates how adversaries can extract state preparation circuits and training labels from QML models, directly stealing training data by reverse-engineering the encoding process. This underscores the criticality of encoding schemes as attack surfaces, as they bridge raw data to quantum computation. Quantum homomorphic encryption (QHE) [25], while theoretically viable, imposes prohibitive overheads incompatible with near-term devices. Recent quantum-specific defenses address model theft through strategies like distributed execution (QuMoS [26]) and output obfuscation (STIQ [27]), but these focus on protecting trained parameters rather than preventing encoding detection. Furthermore, while strategies like circuit partitioning [19], [20] aim to distribute trust across providers, they fail to protect QNNs and related IPs because any untrusted provider with access to transpiled circuits can recover encoding schemes. Our work diverges by addressing encoding-specific transpilation artifacts. Unlike [24], which focuses on training data extraction, we demonstrate that adversaries can preemptively identify encoding methods to streamline subsequent attacks.

III. Threat model

A. Basic Idea

The assets in QNNs include proprietary algorithms, training data, and the resulting trained models. Identifying the encoding scheme used in a QNN can facilitate the extraction of embedded features, posing a security risk. While quantum encoding methods such as angle, amplitude, and basis encoding provide distinct data representations, their implementation on near-term hardware introduces artifacts due to transpilation. We propose that these artifacts inherently imprint encoding-specific signatures during transpilation, which can be exploited by a malicious quantum service provider. In the proposed attack model, the adversary operates as an untrusted cloud provider while posing as a legitimate and reliable hardware vendor. With access to transpiled QNNs submitted for training, the adversary can analyze the state preparation (encoding) circuits (Fig. 1). The primary objective is to extract critical details about the victim's encoding scheme and embedded features, which can then be monetized.

B. Adversary Capabilities and Assumption

We assume that: (a) The adversary has access to the transpiled circuit submitted by users and the results produced by the quantum computer. This is justified because the cloud provider, by design, has access to both the QNN and the measurement results; (b) The victim employs one of the three widely used encoding schemes: angle encoding, amplitude encoding, or basis encoding, and encodes only one feature per qubit. This assumption is reasonable as these encodings are the foundation of many near-term QML algorithms and are widely adopted due to their compatibility with current NISQ devices. By restricting our scope to these encoding methods,

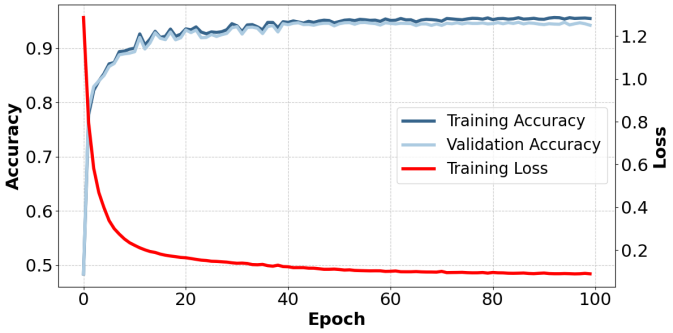

Fig. 2. Training, validation accuracy and loss for classifying the encoding scheme in a 3-qubit circuit with 3 encoded features.

we establish a realistic and practical baseline for analyzing encoding detection under existing hardware constraints.

C. Attack Process

This section presents a hybrid quantum-classical approach for encoding classification, combining quantum circuit transpilation patterns with classical feature engineering. The methodology focuses on identifying quantum circuit signatures through systematic analysis of gate-level implementations across different encoding schemes.

*1) **Data Set Generation**:* The dataset is constructed using PQCs with varied encoding schemes and rotational configurations:

Encoding Layer Construction: We generate distinct quantum state representations using three fundamental state preparations: *a) Pauli Rotations*: maps classical features x_i to qubit rotations $R_j(x_i)$, where $j \in X, Y, Z$,with $x_i \sim \mathcal{U}(0, 2\pi)$. *b) Amplitude Encoding*: represents 2^n features in n-qubit state using haar-random 2^n-dimensional statevector initialization $\|\psi\rangle = \frac{1}{\|\vec{v}\|}\sum_k v_k|k\rangle$. *c) Basis Encoding*: direct bit-to-qubit mapping via X gates.

PQC Augmentation: PQCs with varying configurations (e.g., entanglement patterns and gate types) are added to enhance the diversity of the dataset. We used different PQCs with different starting gates (P):

$$\mathcal{P} = \{\texttt{rx}, \texttt{ry}, \texttt{rz}, \texttt{cx}, \texttt{x}, \texttt{sx}, \texttt{crx}, \texttt{cry}, \texttt{crz}\}$$

Hardware Simulation: Quantum circuits are transpiled using a noisy fake back-end (GenericBackendV2), ensuring features reflect hardware-aware implementations.

Labeling: Each sample is labeled based on its encoding type to facilitate supervised learning.

Example: For a sample 3-qubit system with single-feature encoding, we generate a comprehensive dataset of 18,000 circuits.

*2) **Feature Extraction and Engineering**:* Feature extraction focuses on converting the characteristics of quantum circuits into a numerical feature space suitable for classical machine learning. Our approach starts with an intuitive analysis of transpiled circuits across various encoding schemes to identify recognizable patterns. Insights from this exploratory analysis guided the creation of a systematic feature extraction process, designed to capture key quantum circuit patterns crucial for differentiating between various encoding types.

$$\text{Features} = 27 + 2 \times \text{Qubits} \tag{1}$$

The feature engineering pipeline refines the process by identifying discriminative circuit features (Equation 1) through three levels of analysis:

Structural Signatures: Provide essential insights into the encoding mechanisms used within quantum circuits. Key markers include: *a) Basis encoding markers*: Ratios of X and SX gates, along with binary pre-CNOT patterns; *b) Amplitude encoding indicators*: Diversity in rotation gate parameters indicating high variability in amplitude encoding; *c) Transpilation artifacts*: Frequency of RZ-SX sequences and statistical distributions of rotation angles.

Rotation Analysis: Rotation-based features capture the statistical properties of gate parameters, including: a) Mean, standard deviation, and entropy of rotation angles for each gate type; b) Correlation of consecutive gate parameters across the circuit; c) Modular distributions of rotation angles $(\theta \mod 2\pi)$ to identify characteristic transformations.

Entanglement Signatures: Entanglement metrics characterize multi-qubit interactions, which are crucial for understanding circuit complexity.

The extracted feature space captures both global circuit properties and local qubit-wise patterns.

*3) **Training Model**:* The classification model integrates quantum-aware preprocessing with classical deep learning to effectively differentiate encoding schemes.

Preprocessing Pipeline: The preprocessing pipeline ensures standardized input representation and maintains encoding balance. A stratified data splitting approach is employed, using a 60-20-20 split to preserve encoding ratios across training, validation, and test sets. Additionally, Z-score normalization is applied to standardize the features to have zero mean and unit variance.

Network Architecture: The neural network employs a multilayer perceptron (MLP) classifier designed to capture both local and global circuit characteristics. The input layer consists of a number of neurons equal to the total extracted features, ensuring a direct mapping from the feature space to the network. The first hidden layer, comprising 25 neurons, while the second hidden layer, with 10 neurons. Finally, the output layer applies a softmax activation function to classify inputs into five encoding types. The network is trained using the ReLU activation function, the Adam optimizer, and a maximum of 100 training iterations.

IV. Proposed Defense

The adversarial detection of quantum encoding scheme relies on identifying predictable patterns in the transpiled circuit structure. To counter this, we introduce a temporary scrambling mechanism that strategically alters the transpiled circuit's transient properties while preserving its final output. Our proposed defense mechanism acts as a transient

TABLE I
CLASSIFICATION PERFORMANCE METRICS (3-QUBIT QNN)

Encoding Type		Precision	Recall	F1-Score	Support
Amplitude		1.00	1.00	1.00	900
Basis		0.93	0.81	0.87	900
Angle	Rx	0.92	0.96	0.94	900
	Ry	0.93	0.97	0.95	900
	Rz	0.95	0.98	0.96	900
Accuracy		-	-	0.94	4500
Macro Avg		0.94	0.94	0.94	4500
Weighted Avg		0.94	0.94	0.94	4500

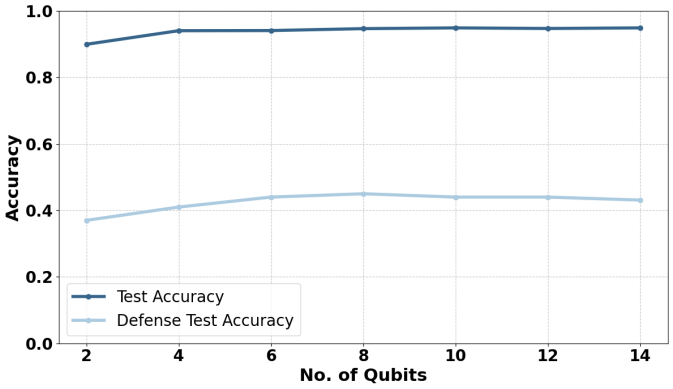

Fig. 3. Test accuracy for circuits with varying qubit counts, where each qubit encodes a single feature.

"cloaking" layer strategically positioned between the quantum encoding circuit and the parameterized quantum circuit (PQC). This layer temporarily obfuscates the encoded state's structure, ensuring adversaries cannot identify the encoding methods. The defense operates in three stages:

*1) **Obfuscation Phase**:* Immediately after the encoding circuit, we apply a sequence of randomized quantum gates:

Basis randomization: Hadamard (H) gates place all qubits into superposition states.

Phase randomization: Parameterized RX(θ) rotations with angles θ sampled uniformly from [-π, π] introduce qubit-specific phase shifts.

Entanglement creation: CNOT gates entangle adjacent qubit pairs (e.g., qubit 0→1, 2→3), generating artificial correlations.

After the initial state encoding , we apply:

$$U_{\text{obf}} = \left(\prod_{i=0}^{n-1} H_i RX(\theta_i) \right) \left(\prod_{j=0}^{\lfloor n/2 \rfloor} \text{CNOT}_{2j,2j+1} \right)$$

*2) **Isolation Barrier**:* A hardware-enforced barrier prevents quantum compilers from optimizing or rearranging gates across the defense structure, preserving the intentional obfuscation pattern. The barrier is not observable at the cloud end during execution thereby eliminating any adversarial clue.

*3) **Inversion Phase**:* Before the PQC circuit, we systematically undo the obfuscation:

Entanglement removal: Apply inverse CNOT gates in reverse order (e.g., qubit 2→3 first, then 0→1).

TABLE II
AVERAGE CIRCUIT DEPTH (ORIGINAL VS OBFUSCATED)

Encoding Type		Original	Obfuscated	Δ Abs	Δ↑%
Amplitude		95.2	101.1	+5.9	6.2%
Basis		67.6	74.1	+6.5	9.7%
Angle	Rx	70.2	76.1	+5.9	8.4%
	Ry	69.2	76.1	+6.9	10%
	Rz	67.7	74.1	+6.4	9.5%

Phase cancellation: Implement RX(θ) rotations to negate the initial random phases.

Basis restoration: Reapply H gates to return qubits to their original basis.

This architecture preserves encoding anonymity while ensuring correct functionality, as the parameterized quantum circuit (PQC) operates directly on the originally encoded quantum state $|\psi_{\text{enc}}\rangle$.

V. RESULTS AND EVALUATION

Due to the long queue times and limited availability of real quantum devices, we used Qiskit's fake provider module that mimics IBM's system and includes real hardware-calibrated data. Training was performed on an Intel Core-i7-12700H CPU with 40GB of RAM. We evaluated the efficacy of the proposed encoding detection attack through three critical dimensions: classification accuracy across QNN architectures with varying qubit counts and different PQCs, feature scalability, and training convergence.

A. Training Convergence and Attack Reliability

The training dynamics Fig.2 confirms the reliability of the classifier under the design feature space. The validation accuracy reaches 80% in epoch 40 and stabilizes at 90% training accuracy by epoch 100. Concurrently, training loss decreases monotonically from 1.2 to 0.2, indicating stable optimization without overfitting.

Table I details the attack's precision, recall, and F1-scores for distinguishing encoding schemes in a 3-qubit system. The amplitude encoding achieves perfect classification (F1=1.00), attributed to its unique transpilation artifacts. In contrast, basis encoding shows reduced recall (0.81) and F1-score (0.87), likely due to its reliance on X and SX gates—patterns that may overlap with angle-encoded circuits post-transpilation. The adversary can predict the encoding scheme used by the user with a $\approx$ 95% success rate. *These results validate the feature engineering pipeline's ability to isolate encoding-specific signatures, even for structurally similar angle variants. The adversary can identify the type of encoding used and, in the case of angle encoding, determine the specific rotation gate employed for data encoding.*

B. Feature Scalability and Attack Robustness

The linear relationship between feature count and the number of qubits (Equation.1) demonstrates the attack's adaptability to circuit complexity. At the 3-qubit baseline, 33 features capture structural patterns, rotation statistics, and entanglement properties. Expanding to 14 qubits increases the feature

space to 55, improving the ability to distinguish encoding-specific transpilation artifacts (e.g., RZ-SX sequences in basis encoding). While extrapolation to 100 qubits (227 features) raises classical computational costs, *the linear growth ensures practical viability for near-term attacks targeting* $\leq 14-qubit$ *QNNs—consistent with current QML benchmarks.*

C. Encoding Classification Accuracy vs. Qubit Count

The adversary can accurately classify encoding types even as the number of qubits increases (Fig.3). For a 3-qubit system, the attack achieves 94% accuracy, demonstrating its effectiveness at small scales. As qubit count scales to 14, test accuracy improves from 90% to 95%, indicating that the model effectively captures and differentiates encoding-specific transpilation artifacts. Furthermore, the model not only identifies the encoding type but also determines the specific rotation gates used in angle encoding, even as circuit complexity grows. *The transpilation artifacts and circuit structures remain sufficiently distinct at higher qubit counts, allowing the model to generalize effectively across varying circuit sizes.*

D. Defense Efficacy and Overhead

To assess the effectiveness of the proposed defense, we generated a test dataset incorporating the defense strategy and evaluated the trained classifier's accuracy. The results indicate a significant reduction in adversarial encoding detection accuracy, dropping from 95% to an average of 42% across all encoding types (Fig.3). However, this obfuscation introduces an average depth increase of ≈8.5% compared to baseline transpiled circuits. Table II presents a comparison of the average circuit depth across 800 instances for each encoding type, considering a 5-layers, of a low-depth PQC. Notably, for deeper QNNs and multi-layer PQC architectures, the relative % increase in depth is expected to be negligible.

VI. Conclusion

This work identifies a critical vulnerability of QNN's white-box access to adversaries in untrusted quantum cloud by demonstrating that quantum encoding schemes can be reliably detected with ≈95% accuracy. This is due to transpilation artifacts which can aid in subsequent state preparation circuit (an IP) theft. To mitigate this risk, we propose to strategically insert transient obfuscation layers—randomized rotations and entanglement—to obscure encoding patterns. This approach reduces adversarial detection accuracy to near-random levels (≈42%) while introducing a minimal circuit depth overhead of ≈8.5% for a 5-layer QNN design.

References

[1] I. Cong, S. Choi, and M. D. Lukin, "Quantum convolutional neural networks," *Nature Physics*, vol. 15, no. 12, pp. 1273–1278, 2019.

[2] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, "Quantum machine learning," *Nature*, vol. 549, no. 7671, pp. 195–202, 2017.

[3] S. Ghosh, S. Upadhyay, and A. A. Saki, "A primer on security of quantum computing," *arXiv preprint arXiv:2305.02505*, 2023.

[4] Y. Cao, J. Romero, and A. Aspuru-Guzik, "Potential of quantum computing for drug discovery," *IBM Journal of Research and Development*, vol. 62, no. 6, pp. 6–1, 2018.

[5] E. Farhi, J. Goldstone, and S. Gutmann, "A quantum approximate optimization algorithm," *arXiv preprint arXiv:1411.4028*, 2014.

[6] J. Tilly, H. Chen, S. Cao, D. Picozzi, K. Setia, Y. Li, E. Grant, L. Wossnig, I. Rungger, G. H. Booth *et al.*, "The variational quantum eigensolver: a review of methods and best practices," *Physics Reports*, vol. 986, pp. 1–128, 2022.

[7] R. Orús, S. Mugel, and E. Lizaso, "Quantum computing for finance: Overview and prospects," *Reviews in Physics*, vol. 4, p. 100028, 2019.

[8] S. Gupta, S. Modgil, P. C. Bhatt, C. J. C. Jabbour, and S. Kamble, "Quantum computing led innovation for achieving a more sustainable covid-19 healthcare industry," *Technovation*, vol. 120, p. 102544, 2023.

[9] P. Rebentrost, M. Mohseni, and S. Lloyd, "Quantum support vector machine for big data classification," *Physical review letters*, vol. 113, no. 13, p. 130503, 2014.

[10] S. Lloyd and C. Weedbrook, "Quantum generative adversarial learning," *Physical review letters*, vol. 121, no. 4, p. 040502, 2018.

[11] A. Abbas, D. Sutter, C. Zoufal, A. Lucchi, A. Figalli, and S. Woerner, "The power of quantum neural networks," *Nature Computational Science*, vol. 1, no. 6, pp. 403–409, 2021.

[12] IBM, "IBM Quantum," 2024. [Online]. Available: https://www.ibm.com/quantum

[13] Google Quantum AI, "Google Quantum Computer," 2024. [Online]. Available: https://quantumai.google/quantumcomputer

[14] Amazon, "Amazon Braket," 2024. [Online]. Available: https://aws.amazon.com/braket/

[15] ——, "Saiwa," 2023. [Online]. Available: Saiwa. [a. n. d.]. Machine Learning as a Service (MLaaS) — Everything you need to know about that. https://saiwa.ai/blog/mlaas-1/

[16] Z. Computing, "Orquestra," May 2021. [Online]. Available: https://www.zapatacomputing.com/orquestra/

[17] C. Q. Computing, "pytket," May 2021. [Online]. Available: https://cqcl.github.io/pytket/build/html/index.html

[18] baidu, "Quantum," 2024. [Online]. Available: https://www.insidequantumtechnology.com/news-archive/chinas-baidu-rolls-beijing-based-quantum- computer-and-access-platform/

[19] S. Upadhyay and S. Ghosh, "Robust and secure hybrid quantum-classical computation on untrusted cloud-based quantum hardware," in *Proceedings of the 11th International Workshop on Hardware and Architectural Support for Security and Privacy*, 2022, pp. 45–52.

[20] S. Upadhyay, R. O. Topaloglu, and S. Ghosh, "Trustworthy computing using untrusted cloud-based quantum hardware," *arXiv preprint arXiv:2305.01826*, 2023.

[21] W. Hua, Z. Zhang, and G. E. Suh, "Reverse engineering convolutional neural networks through side-channel information leaks," in *Proceedings of the 55th Annual Design Automation Conference*, 2018, pp. 1–6.

[22] S. J. Oh, B. Schiele, and M. Fritz, "Towards reverse-engineering black-box neural networks," *Explainable AI: interpreting, explaining and visualizing deep learning*, pp. 121–144, 2019.

[23] A. Ghosh and S. Ghosh, "The quantum imitation game: Reverse engineering of quantum machine learning models," in *Proceedings of the 2024 Workshop on Attacks and Solutions in Hardware Security*, 2024, pp. 48–57.

[24] S. Upadhyay and S. Ghosh, "Quantum data breach: Reusing training dataset by untrusted quantum clouds," 2024. [Online]. Available: https://arxiv.org/abs/2407.14687

[25] K. A. Fisher, A. Broadbent, L. Shalm, Z. Yan, J. Lavoie, R. Prevedel, T. Jennewein, and K. J. Resch, "Quantum computing on encrypted data," *Nature communications*, vol. 5, no. 1, p. 3074, 2014.

[26] Z. Wang, J. Li, Z. Hu, B. Gage, E. Iwasawa, and W. Jiang, "Qumos: A framework for preserving security of quantum machine learning model," in *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, vol. 1. IEEE, 2023, pp. 1089–1097.

[27] S. Kundu and S. Ghosh, "Stiq: Safeguarding training and inferencing of quantum neural networks from untrusted cloud," *arXiv preprint arXiv:2405.18746*, 2024.