Characterizing Scaling Trends of Post-Compilation Circuit Resources for NISQ-era QML Models

Rupayan Bhattacharjee, Pau Escofet, Santiago Rodrigo, Sergi Abadal, Carmen G. Almudéver[1], Eduard Alarcón

NaNoNetworking Center in Catalonia (N3Cat), Universitat Politècnica de Catalunya, Spain
[1]*Universitat Politècnica de Valencia, Spain*
Email: rupayan.bhattacharjee@upc.edu

Abstract—This work investigates the scaling characteristics of post-compilation circuit resources for Quantum Machine Learning (QML) models on connectivity-constrained NISQ processors. We analyze Quantum Kernel Methods and Quantum Neural Networks across processor topologies (linear, ring, grid, star), focusing on SWAP overhead, circuit depth, and two-qubit gate count. Our findings reveal that entangling strategy significantly impacts resource scaling, with circular and shifted circular alternating strategies showing steepest scaling. Ring topology demonstrates slowest resource scaling for most QML models, while Tree Tensor Networks lose their logarithmic depth advantage after compilation. Through fidelity analysis under realistic noise models, we establish quantitative relationships between hardware improvements and maximum reliable qubit counts, providing crucial insights for hardware-aware QML model design across the full-stack architecture.

Index Terms—Quantum Machine Learning, Quantum Computing Systems, Quantum Kernels, Quantum Neural Networks.

I. INTRODUCTION

Quantum Machine Learning (QML) has emerged as a promising approach to leverage the unique properties of quantum systems for learning complex data distributions [1]. As quantum hardware continues to improve in qubit count and fidelity, there is growing interest in developing models that can demonstrate practical quantum advantage for real-world machine learning tasks, particularly in the Noisy Intermediate-Scale Quantum (NISQ) era and beyond [2].

Among the most studied paradigms are Quantum Kernel Methods (QKMs) [3] and Quantum Neural Networks (QNNs) [4], [5], where quantum circuits are used to implicitly map classical data into high-dimensional Hilbert spaces to learn complex patterns while also being suitable for NISQ devices. Although several works have demonstrated that these methods can outperform classical ones in specific regimes [6]–[9], particularly in small-scale learning tasks [10], [11], these studies provide no insight into how these methods scale, especially in terms of their implementability on near-term quantum processors with limited qubit connectivity and resources.

Several previous approaches have attempted to address the optimal design of QML models by leveraging expressive

power of Parametrized Quantum Circuits (PQCs) [12] and proposed robust circuit design techniques via gate quantization and pruning [13], error-tolerant embeddings [14] and quantum architecture search [15]–[17]. However, most lack either their adaptability to connectivity and resource-constrained quantum processors, or an understanding of how circuit-level resources scale, particularly under realistic noise channels, impacting QML performance.

In this work, we characterize the compilation-aware resource consumption of a wide variety of quantum circuits used in QKMs and QNNs, quantifying the resulting overhead when compiling on different processor topologies and assessing their scalability with increasing qubit counts. Unlike prior methods, our approach allows a system architecture-aware analysis of the scalability of QML models, building groundwork for developing QML systems optimized across multiple layers of the full-stack. We further analyze the fidelity of QML models on near-term hardware while drawing insights into the optimal design of these models from a hardware-aware perspective. Our main contributions are as follows:

- A systematic characterization of post-compilation resource scaling for QML models across diverse quantum processor topologies.
- A quantitative study relating gate error and gate time improvements to maximum reliable qubit counts for QML models.

II. BACKGROUND

A. NISQ-era QML Models

QKMs use quantum circuits $U(\vec{x})$ acting on the state $|0\rangle^{\otimes N}$ to encode classical data $\vec{x}$ into a quantum state $|\Psi(\vec{x})\rangle$, inducing an implicit high-dimensional feature map. The resulting kernel between two data points $\vec{x_i}$ and $\vec{x_j}$ is defined as $K(\vec{x_i}, \vec{x_j}) = |\langle\Psi(\vec{x_i})|\Psi(\vec{x_j})\rangle|^2 = |\langle 0|^{\otimes N} U^\dagger(\vec{x_j})U(\vec{x_i})|0\rangle^{\otimes N}|^2$, and can be used in standard classifiers such as Support Vector Machines (SVMs).

QNNs implement feature maps $U(\vec{x})$ to encode the data, followed by trainable PQCs (ansatz) $V(\vec{\theta})$ constructed via layered circuits of single-qubit rotations and entangling gates. The variational parameters $\vec{\theta}$ are varied to optimize the expectation value of an observable $\hat{O}$ which serves as the cost function $C(\vec{x}, \vec{\theta}) = \langle\Psi(\vec{x}, \vec{\theta})|\hat{O}|\Psi(\vec{x}, \vec{\theta})\rangle = \langle 0|^{\otimes N} U^\dagger(\vec{x})V^\dagger(\vec{\theta})\hat{O}V(\vec{\theta})U(\vec{x})|0\rangle^{\otimes N}$ of the QNN. The model

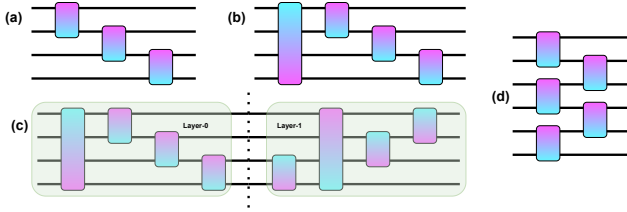arXiv:2509.11980v1 [quant-ph] 15 Sep 2025

Fig. 1. Entangling strategies with gate direction shown via color gradients: (a) Linear, each qubit connects to the next; (b) Circular, linear with an added gate between first and last qubits; (c) SCA (Shifted Circular Alternating), circular layout where, across layers, the first–last qubit connection shifts position, gate directions reverse, and application order alternates; (d) Pairwise, alternating layers of gates between odd–even and even–odd qubit pairs.

is trained by evaluating gradients of the cost function via the parameter-shift rule [18], i.e. $\frac{\partial C(\vec{x}, \vec{\theta})}{\partial \theta_i} = \frac{1}{2}(C(\theta_i + \pi/2) - C(\theta_i - \pi/2))$.

The structure of the ansatz plays a key role in determining the impact of barren plateaus on the models [19]. The distribution of entangling two-qubit gates in the circuit, often colloquially called entangling strategy, is known to determine the expressibility and entangling capacity of QNNs [12]. Our work builds on this by assessing various entanglement strategies (shown in Figure 1) and one logarithmic-depth ansatz under realistic hardware constraints.

B. Hardware Topologies and Compilation

Implementing quantum circuits on processors requires a three-stage compilation process: (1) mapping virtual qubits of the quantum circuit to physical qubits on the processor (called mapping/ layout stage), (2) inserting SWAP gates when two-qubit operations are needed between unconnected qubits (routing stage), and (3) decomposition into the processor's native gate set, and circuit optimization through gate commutations and cancellations. A visual depiction of quantum circuit compilation, particularly qubit layout and routing stages, is given in Figure 3.

The actual cost of executing any quantum circuit on a given processor topology depends on the number of two-qubit gates (i.e. qubit interactions) and their distribution in the quantum circuit. In this work, we explore the implementation cost of QML models with different two-qubit gate distributions on several hardware topologies, since the compilation overhead directly affects circuit fidelity and execution reliability.

III. METHODOLOGY

We evaluated two classes of QML models: QKMs and QNNs as they are the most NISQ-compatible. We build quantum kernel estimation circuits using the ZZFeatureMap to encode the data followed by the inverse of the same. This particular feature map is known to impart advantage in certain classification tasks [3]. We set the number of layers of ZZFeatureMap to 1, i.e. `ZZFeatureMap(reps=1)`, and consider various circuit architectures by varying the entangling

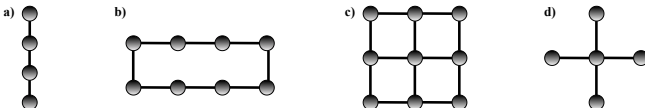Fig. 2. Processor topologies: (a) Linear, (b) Ring, (c) Grid, and (d) Star.

Parameter	Value
Quantum Circuits	QKM (ZZFeatureMap), QNN(ZZFeatureMap+TwoLocal), TTN, GHZ.
Entangling strategies	'linear','circular','sca','pairwise'
Processor topologies	Linear, Ring, Grid, Star
Compiler	Qiskit [20]
Mapping (Layout)	TrivialLayout
Routing method	SABRE [21]
Basis Gate Set	[$U_3(\theta, \phi, \lambda)$, CNOT]
Single-qubit error rate	7.42×10^{-5} [22]
Two-qubit error rate	7×10^{-4} [23]
Single qubit gate time	7.9 ns [24]
Two-qubit gate time	30 ns [25]
T_1 and T_2	1.2 ms [26] and 1.16 ms [26]
Qubit count ranges	TTN: [8, 16, 32,..., 1024], others: [100, 200, 300,...1000]

strategy. For QNN circuits, we use ZZFeatureMap with a fixed entangling strategy ('linear') followed by the TwoLocal ansatz with varying entanglement strategies. The number of layers in the feature map and the ansatz are kept at 1 and 2 respectively, so that the number of two-qubit gates in QKM circuits is comparable to that in the QNNs, as ZZFeatureMap has twice as many CNOTs as TwoLocal ansatz. In the following section, circuits are labeled by their type and entanglement strategy (e.g., 'Kernel Circular' refers to a kernel circuit of ZZFeatureMap with circular entanglement, 'QNN Pairwise' refers to ZZFeatureMap followed by TwoLocal with pairwise entanglement).

We choose the ZZFeatureMap and TwoLocal ansatz as they are the most commonly used quantum circuits in QML. Besides these, we also use the Tree Tensor Network (TTN) ansatz [27], as they have logarithmic circuit depth and hence unlikely to be affected by barren plateaus [28]. Additionally, we compare them with the GHZ state preparation circuit, as a non-QML circuit with comparable two-qubit gate complexity before compilation. For circuit resource analysis, qubit counts ranged from 100-1000 (intervals of 100) for QKM, QNN, and GHZ circuits, while TTN used 8-1024 (powers of 2) due to its binary tree structure.

A. Circuit Compilation and Processor Topologies

All quantum circuits were compiled on various quantum processor topologies using the Qiskit framework [20]. The circuits were decomposed to a basis gate set consisting of $U_3(\theta, \phi, \lambda)$ and $CNOT$ gates. The qubit mapping technique is TrivialLayout, which means that all the virtual qubits (q_i) of the circuit were mapped to the corresponding physical qubit (Q_i) on the processor. The qubit routing method used was SABRE [21]. The processor topologies considered were linear, ring, grid and star as shown in Figure 2. The coupling map is bidirectional, allowing the use of CNOT gates in both directions.

B. Noise Model for Fidelity Estimation

For fidelity calculations, we consider an analytical model for depolarizing noise proposed by [29]. This particular model
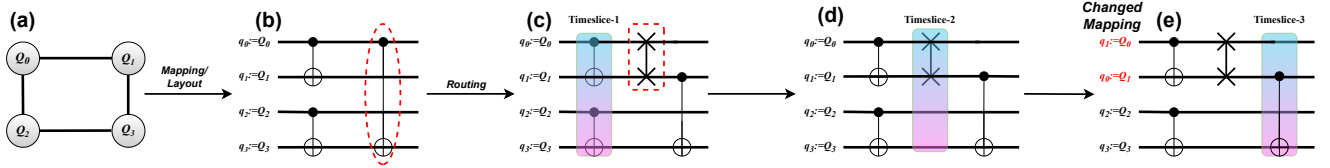
Fig. 3. Qubit mapping and routing stages of the quantum circuit compilation process: (a) The qubit connectivity of a quantum processor (coupling map); (b) the quantum circuit to be implemented: mapping q_i of the quantum circuit to Q_i of the processor. It can be observed that the CNOT between q_0 and q_3 (in red) cannot be implemented as Q_0 and Q_3 are not connected; (c) routing stage: a SWAP gate is applied between Q_0 and Q_1 (in red) so that the CNOT gate can be applied between q_0 and q_3, the first layer of CNOTs is implemented in timeslice-1, (d) the added SWAP gate is implemented in timeslice-2, (e) the last CNOT gate is applied between Q_1 and Q_3 (equivalently between virtual qubits q_0 and q_3) in timeslice-3.
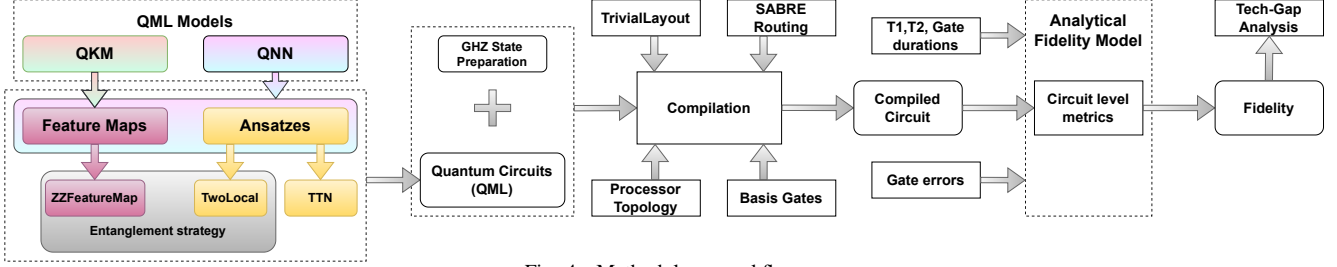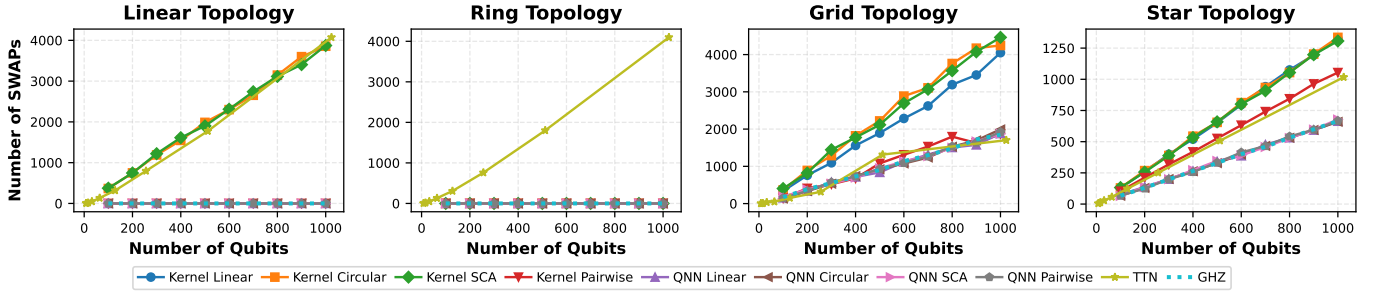

Fig. 4. Methodology workflow.

Fig. 5. SWAP gate overhead of several QML models and GHZ state circuit upon compilation onto different processor topologies. The non-QML circuit ('GHZ') is shown as a dotted line.

has been shown to outperform all existing methods for circuit fidelity evaluations and is computationally lightweight due to its analytical nature. The compiled quantum circuit is divided into timeslices, where each timeslice represents a layer of gates that execute in parallel on the quantum processor. The fidelity of all qubits start with 1, and, for each single-qubit gate applied, the fidelity F_{q_i} of qubit q_i reduces to:

$$F_{q_i} \leftarrow (1-p)F_{q_i} + (1-p_{ent})\frac{p}{2} \qquad (1)$$

where p is a depolarization parameter and p_{ent} is the entanglement hyperparameter which decides the degree of correlation between the depolarizing errors in qubits. For each two-qubit gate, the fidelity of both qubits reduce to

$$F_{q_i,q_j} \leftarrow \sqrt{(1-p)}F_{q_i,q_j} + (1-p_{ent})\eta, \qquad (2)$$

where η is given by

$$\eta = \frac{1}{2}(\sqrt{(1-p)(F_{q_i}+F_{q_j})^2 + p} - \sqrt{1-p}(F_{q_i}+F_{q_j})). \qquad (3)$$

For every timeslice, the fidelity of each qubit is updated (including qubits with no gates applied) to capture the effects of decay and decoherence, as follows,

$$F_{q_i} \leftarrow F_{q_i}.e^{-t_{layer}/T_1}.\frac{1}{2}(e^{-t_{layer}/T_2}+1), \qquad (4)$$

where t_{layer} is the timeslice duration, set by the slowest gate time in that layer. Finally, the fidelity values of each qubit, are multiplied to yield total circuit fidelity. The single and two-qubit gate error rates, coherence times (T_1, T_2), and gate execution times are assumed to be uniform across all qubits and connections.

For technology gap analysis, we define the threshold qubit count ($N_{threshold}$) as the maximum number of qubits achievable while maintaining circuit fidelity above 0.99. Qubit ranges of $10-100$ (interval of 10) were used for most circuits, with TTN using $4-64$ (powers of 2) qubits. $N_{threshold}$ values were determined by fitting stretched exponential functions to fidelity curves.

The entire experimental configuration is listed in Table I and the methodology workflow is depicted in Figure 4.

IV. RESULTS AND DISCUSSION

A. Circuit Resource Overhead

Understanding post-compilation resource overhead is critical for assessing the practical scalability of QML models on near-term quantum devices with limited connectivity and for determining the reliability of circuit implementations. Figure 5 illustrates how the SWAP overhead scales with qubit count
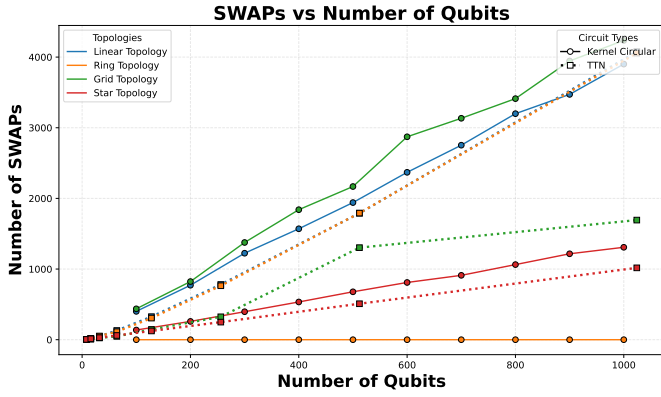
Fig. 6. SWAP overhead scaling for Kernel Circular and TTN circuits on all processor topologies to facilitate the comparison between multiple topologies.

for various QML circuits and the GHZ state preparation circuit on different processor topologies. As shown, all circuits follow linear trends except TTN which demonstrates a steeper scaling on linear and ring topologies. Kernel Circular and SCA are the most expensive in terms of SWAP overhead, across all topologies except the ring topology where TTN shows the steepest scaling and all other circuits have no SWAP overhead as ring topology satisfies the qubit interactions required to implement them all. On grid and star topologies, none of the circuits can be implemented without adding SWAP gates. All QNNs and the GHZ circuit are among the least expensive to implement on all processor topologies. Overall, the optimal topology with the best SWAP overhead scaling for all models other than TTN is ring followed closely by a linear topology. For a more convenient comparison between processor topologies, we show in Figure 6 how the SWAP overhead scales with the number of qubits for Kernel Circular and TTN circuits. It shows that the most optimal configuration for Kernel Circular and TTN circuits are ring and star topologies respectively.

Figure 7 shows various circuit-level metrics of the compiled quantum circuit on different topologies. Just as with SWAPs all circuits exhibit a linear scaling for circuit depth and two-qubit gate count, except TTN which follows a more aggressive two-qubit gate count scaling on linear and ring topologies. The Kernel Circular and SCA have the steepest scaling for compiled circuit depth and two-qubit gate overhead across all processor topologies except ring, where the worst scaling of two-qubit gates is displayed by TTN circuits. Additionally, TTN circuit that exhibits logarithmic depth scaling with the number of qubits no longer remains logarithmic but becomes linear after compilation. This behavior might affect its resilience to noise and barren plateaus [28]. This stresses the importance of designing processor topologies and compilation-aware circuits for greater resilience to noise and barren plateaus. The highest percentage increase in depth and two-qubit gate count is noted in TTN circuits for all but star topology. The GHZ state circuit has the lowest number of two-qubit gates in all processor topologies, making it the least affected by depolarizing gate errors. In terms of circuit depth, the Kernel Pairwise circuit is the most optimal with fixed compiled circuit depth, making it the least likely to be affected by decoherence across all

topologies except star.

B. Fidelity Estimation

While circuit resource analysis provides important insights into compilation overhead, the ultimate measure of practical utility is the resulting circuit fidelity, which determines the reliability of QML computations on real quantum hardware. Figure 8 shows the fidelity of the quantum circuits for various QML models as the number of qubits increases for different quantum processor topologies. It is observed that all circuits except GHZ and Kernel Pairwise (in linear and ring topologies) encounter a significant loss of fidelity (below 0.6) at a scale of 100 qubits, rendering them highly unreliable for kernel or gradient computations. As expected, the fidelity decreases the fastest for Kernel Circular and SCA circuits. The ring topology exhibits the lowest decay in fidelity for all models except TTN which decays much slower in grid topology. In linear and ring topologies, we notice a crossover point for Kernel Pairwise and GHZ state circuits between 60 and 70 qubits after which the fidelity of Kernel Pairwise remains higher than that of the GHZ state circuit. This crossover is observed between 20 and 30 qubits in the grid topology, whereas the GHZ state circuit shows the highest fidelity in star topology where Kernel Pairwise is among the least robust circuits. All QNNs show very low fidelity (less than 0.4) at a scale of 100 qubits in all topologies. TTN circuits seem to be the most reliable in grid topology.

C. Technological Gap Analysis

To understand how technological advances impact QML model performance, we examine how improvements in gate error rates and gate times (equivalently coherence times) affect the fidelity of various QML models and the GHZ circuit. We assume that the error rates and gate times are reduced by an 'Improvement Factor' (Δ_{improv}) and assess the fidelity of the quantum circuits with a fixed qubit count of 256 qubits for improvement factors ranging from 1 to 100. These results are illustrated in Figure 9. For every topology, the fidelity increases rapidly with the improvement factor and then saturates. In linear and ring processor topologies, the fidelity of the Kernel Pairwise circuit saturates much faster than GHZ and other QML circuits, owing to its constant depth scaling. The exception is the grid topology, where TTN scaling is slightly faster than Kernel Pairwise. QNN circuits also converge much faster than kernel circuits, with Kernel Linear and Kernel Circular being the slowest to converge across all processor architectures, supporting the results from previous figures. All QNNs and TTN show the same scaling trend except on grid topology. The star topology exhibits the least variance between saturation curves of all circuits, but appears to be the least optimal configuration.

Additionally, Figure 10 shows the highest number of qubits that can be used in a given quantum circuit ($N_{threshold}$) or a QML model so that the fidelity of the overall circuit is greater than 0.99. We assess this threshold qubit count for a wide range of values of the improvement factor up to 100. It is clear
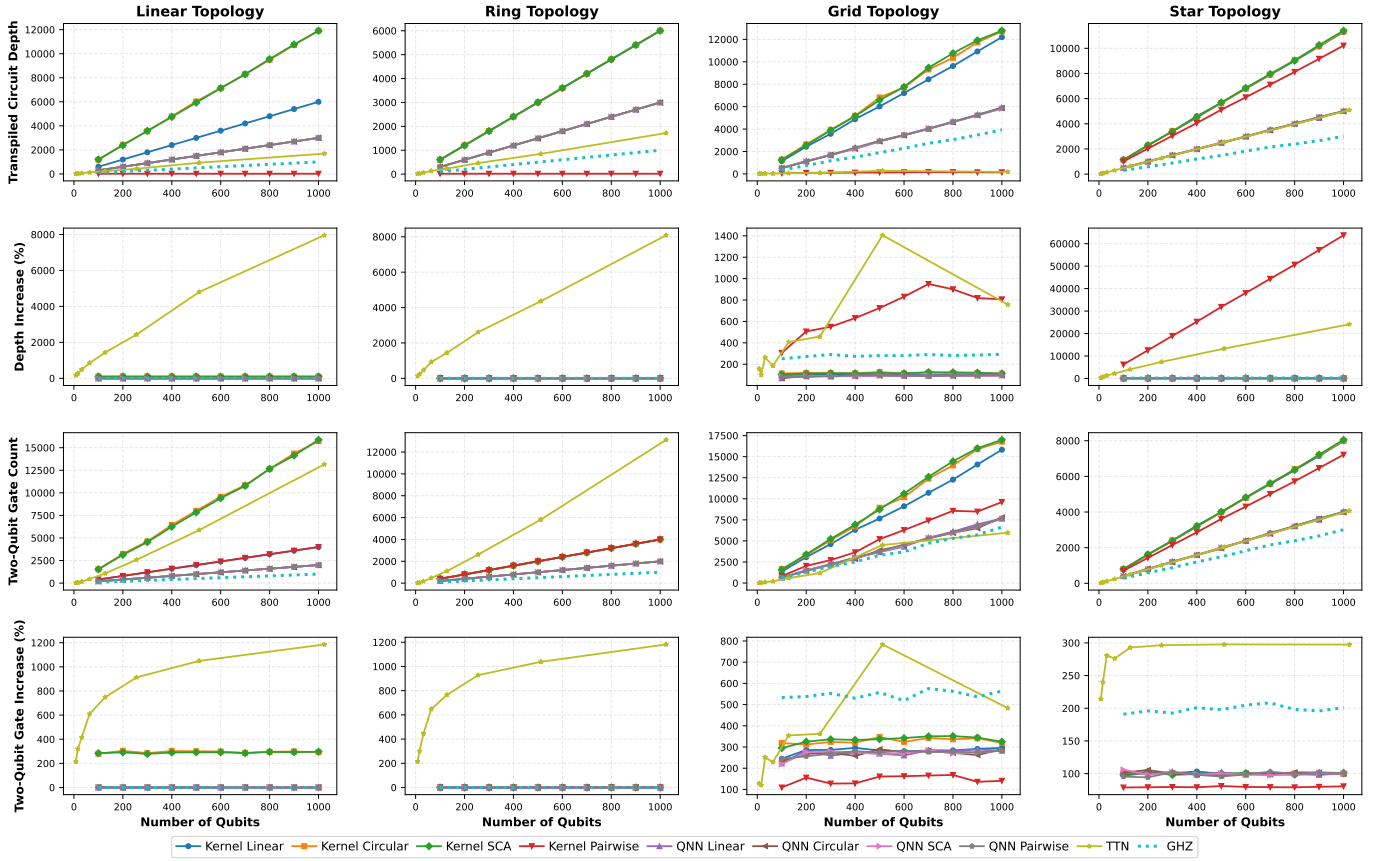
Fig. 7. Resource scaling trends for various post-compilation circuit resources and metrics for all the circuits on all the topologies. The topologies are shown in columns and the metrics (compiled circuit depth, percentage increase in depth post compilation, two-qubit gate count after compilation and two-qubit gate overhead) are shown in rows.
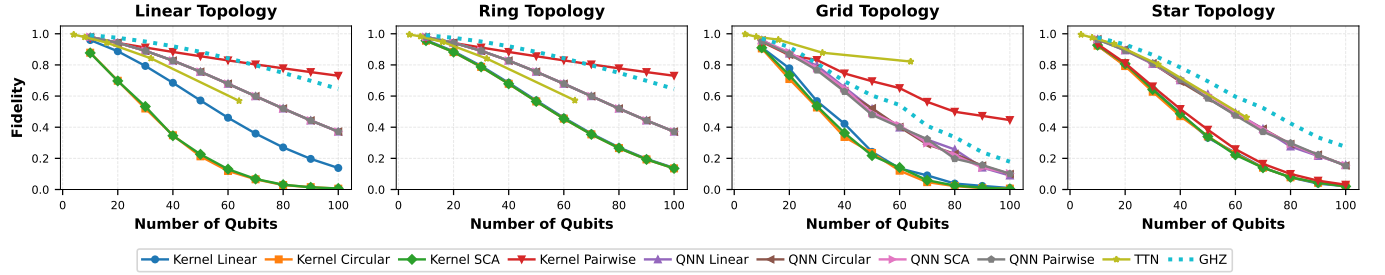

Fig. 8. Fidelity scaling trends of QML models on various topologies with qubit counts upto a 100 qubits with state-of-the-art values for single and two-qubit gate error rates shown in Table I.
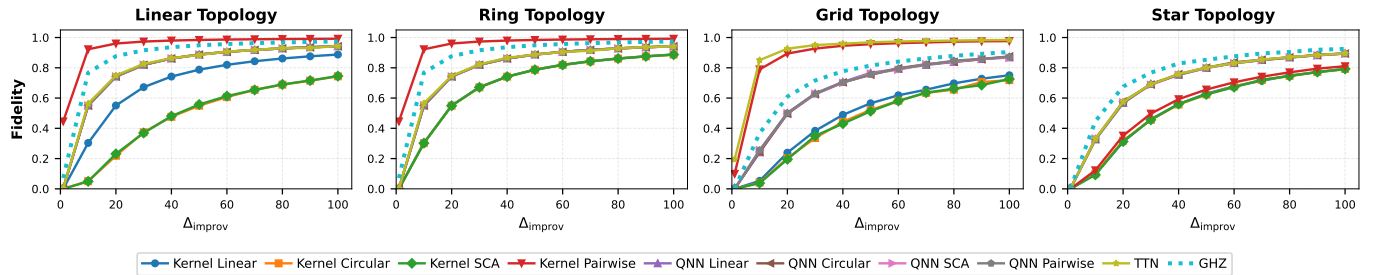

Fig. 9. Technological Gap Analysis: fidelity vs improvement factor (Δ_{improv}), for various QML models and GHZ state preparation circuit at a fixed qubit count of 256 on different processor topologies.

that $N_{threshold}$ scales very slowly with improvement factors except for Kernel Pairwise, which demonstrates linear scaling across most processor architectures. However, even Kernel Pairwise follows slow sub-linear scaling on star topology and
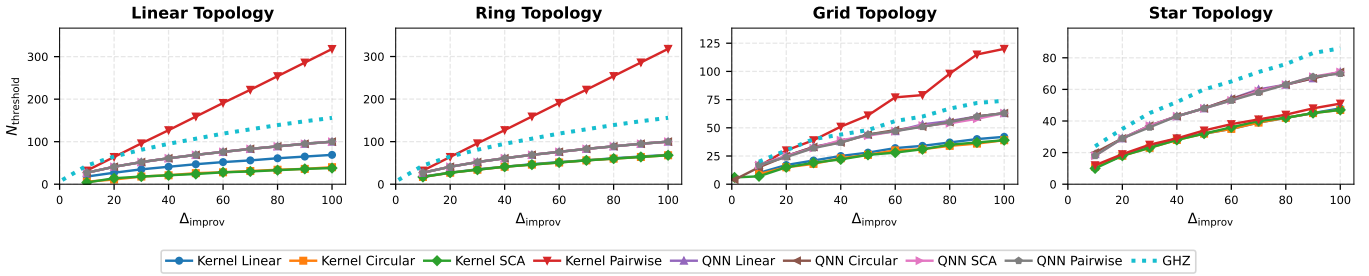
Fig. 10. Highest qubit count with overall circuit fidelity greater than 0.99 ($N_{threshold}$) vs improvement factor (Δ_{improv}), for all the circuits and topologies. TTN results omitted due to sparse and unevenly spaced sampling causing unreliable curve fitting.

is among the worst performing circuits. Across all processor configurations, we notice that even with drastic technological enhancements, most models cannot reliably implement even 100 qubits, which underscores the importance of fault tolerance. However, these scaling trends would show a much higher value for $N_{threshold}$ if we set a lower target fidelity as opposed to a highly demanding value of 0.99. The non-QML GHZ circuit outperforms all QML circuits in the star topology complementing Figures 8 and 9.

V. CONCLUSION

Our comprehensive analysis of post-compilation resource scaling for QML models reveals critical insights for NISQ implementations with qubit connectivity constraints. We demonstrate that while resource requirements mostly scale linearly with qubit count across all processor topologies, ring topology exhibits the most favorable scaling characteristics for most QML models, requiring substantially fewer SWAP gates than other topologies. Importantly, entanglement strategies yield dramatically different resource requirements after compilation, with circular and SCA strategies showing the steepest scaling, while Kernel Pairwise and QNNs offer more modest requirements. A particularly significant finding is that TTN circuits lose their theoretical logarithmic depth advantage after compilation, challenging assumptions about their NISQ advantage. Our fidelity analysis shows that Kernel Pairwise circuits typically maintain higher fidelity than most circuits, with threshold qubit counts scaling linearly with gate error and gate time improvements, as opposed to other models that cannot reliably implement 100 qubits even with significant improvements. These findings allow hardware designers and algorithm developers to make informed decisions about technology requirements to scale QML implementations, while emphasizing the need for full-stack co-design approaches that consider hardware constraints during QML model design rather than relying solely on algorithm-level optimization.

REFERENCES

[1] J. Biamonte *et al.*, "Quantum machine learning," *Nature*, vol. 549, no. 7671, pp. 195–202, 2017.
[2] J. Preskill, "Quantum computing in the nisq era and beyond," *Quantum*, vol. 2, p. 79, 2018.
[3] V. Havlíček *et al.*, "Supervised learning with quantum-enhanced feature spaces," *Nature*, vol. 567, no. 7747, pp. 209–212, 2019.
[4] M. Benedetti *et al.*, "Parameterized quantum circuits as machine learning models," *Quantum science and technology*, vol. 4, no. 4, 2019.

[5] K. Beer *et al.*, "Training deep quantum neural networks," *Nature communications*, vol. 11, no. 1, p. 808, 2020.
[6] H.-Y. Huang *et al.*, "Power of data in quantum machine learning," *Nature communications*, vol. 12, no. 1, p. 2631, 2021.
[7] ——, "Quantum advantage in learning from experiments," *Science*, vol. 376, no. 6598, pp. 1182–1186, 2022.
[8] J. R. Glick *et al.*, "Covariant quantum kernels for data with group structure," *Nature Physics*, vol. 20, no. 3, pp. 479–483, 2024.
[9] Y. Liu, S. Arunachalam, and K. Temme, "A rigorous and robust quantum speed-up in supervised machine learning," *Nature Physics*, vol. 17, no. 9, pp. 1013–1017, 2021.
[10] D. Cugini *et al.*, "Comparing quantum and classical machine learning for vector boson scattering background reduction at the large hadron collider," *Quantum Machine Intelligence*, vol. 5, no. 2, p. 35, 2023.
[11] D. Emmanoulopoulos and S. Dimoska, "Quantum machine learning in finance: Time series forecasting," *arXiv:2202.00599*, 2022.
[12] S. Sim, P. D. Johnson, and A. Aspuru-Guzik, "Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms," *Advanced Quantum Technologies*, vol. 2, no. 12, p. 1900070, 2019.
[13] Z. Hu *et al.*, "Quantum neural network compression," in *Proc. IEEE/ACM ICCAD*, 2022, pp. 1–9.
[14] C. Chu *et al.*, "Qmlp: An error-tolerant nonlinear quantum mlp architecture using parameterized two-qubit gates," in *Proc. ACM/IEEE ISLPED*, 2022, pp. 1–6.
[15] Y. Du *et al.*, "Quantum circuit architecture search for variational quantum algorithms," *npj Quantum Information*, vol. 8, no. 1, 2022.
[16] E.-J. Kuo, Y.-L. L. Fang, and S. Y.-C. Chen, "Quantum architecture search via deep reinforcement learning," *arXiv:2104.07715*, 2021.
[17] H. Wang *et al.*, "Quantumnas: Noise-adaptive search for robust quantum circuits," in *Proc. IEEE HPCA*. IEEE, 2022, pp. 692–708.
[18] M. Schuld *et al.*, "Evaluating analytic gradients on quantum hardware," *Physical Review A*, vol. 99, no. 3, p. 032331, 2019.
[19] J. R. McClean *et al.*, "Barren plateaus in quantum neural network training landscapes," *Nature communications*, vol. 9, no. 1, 2018.
[20] A. Javadi-Abhari *et al.*, "Quantum computing with qiskit," *arXiv:2405.08810*, 2024.
[21] G. Li, Y. Ding, and Y. Xie, "Tackling the qubit mapping problem for nisq-era quantum devices," in *Proc. ASPLOS*, 2019, pp. 1001–1014.
[22] Z. Li *et al.*, "Error per single-qubit gate below 10- 4 in a superconducting qubit," *npj Quantum Information*, vol. 9, no. 1, p. 111, 2023.
[23] V. Negîrneac *et al.*, "High-fidelity controlled-z gate with maximal intermediate leakage operating at the speed limit in a superconducting quantum processor," *Physical Review Letters*, vol. 126, no. 22, 2021.
[24] E. Hyyppä *et al.*, "Reducing leakage of single-qubit gates for superconducting quantum processors using analytical control pulse envelopes," *PRX Quantum*, vol. 5, no. 3, p. 030353, 2024.
[25] Y. Xu *et al.*, "High-fidelity, high-scalability two-qubit gate scheme for superconducting qubits," *Physical Review Letters*, vol. 125, no. 24, 2020.
[26] A. Somoroff *et al.*, "Millisecond coherence in a superconducting qubit," *Physical Review Letters*, vol. 130, no. 26, p. 267001, 2023.
[27] W. Huggins *et al.*, "Towards quantum machine learning with tensor networks," *Quantum Science and technology*, vol. 4, no. 2, 2019.
[28] M. Cerezo *et al.*, "Cost function dependent barren plateaus in shallow parametrized quantum circuits," *Nature communications*, vol. 12, no. 1, p. 1791, 2021.
[29] P. Escofet *et al.*, "An accurate and efficient analytic model of fidelity under depolarizing noise oriented to large scale quantum system design," *Quantum Science and Technology*, 2025.