Journal of Multiscale Materials Informatics

Vol. 02, No. 01, April 2025
https://publikasi.dinus.ac.id/index.php/jimat/ ISSN 3047-5724
DOI: 10.62411/jimat.v2i1.12948

Layerwise Quantum Training: A Progressive Strategy for Mitigating Barren Plateaus in Quantum Neural Networks

Harun Al Azies[1], Muhamad Akrom[2*]

[1,2] Research Center for Quantum Computing and Materials Informatics, Faculty of Computer Science, Universitas Dian Nuswantoro, Semarang 50131, Indonesia

Article Info
Received : May 30, 2025
Revised : June 08, 2025
Accepted : June 12, 2025

Keywords:
Barren Plateau
Quantum Machine Learning
Quantum Neural Network
Parameterized Quantum Circuit

ABSTRACT
Barren plateaus (BP) remain a core challenge in training quantum neural networks (QNN), where gradient vanishing hinders convergence. This paper proposes a layerwise quantum training (LQT) strategy, which trains parameterized quantum circuits (PQC) incrementally by optimizing each layer separately. Our approach avoids deep circuit initialization by gradually constructing the QNN. Experimental results demonstrate that LQT mitigates the onset of barren plateaus and enhances convergence rates compared to conventional and residual-based QNN, rendering it a scalable alternative for Noisy Intermediate-Scale Quantum (NISQ)-era quantum devices.

Corresponding Author:
email: m.akrom@dsn.dinus.ac.id

1. INTRODUCTION

The field of quantum computing has witnessed remarkable advancements in both hardware and algorithmic development, particularly with the emergence of NISQ devices. These devices, although constrained by decoherence, gate infidelity, and limited qubit counts, provide a practical testbed for exploring quantum machine learning (QML) algorithms. Among various QML paradigms, QNN has emerged as a promising class of models that blend the expressive power of PQC with classical training algorithms [1] – [4].

However, QNN faces significant optimization challenges. Chief among them is the BP phenomenon, a situation where the gradient of the cost function vanishes exponentially as the number of qubits or layers increases. This problem renders gradient-based optimization ineffective, resulting in training stagnation. Theoretical analyses have demonstrated that deep and expressive PQC, especially those initialized randomly, are particularly susceptible to barren plateaus due to the concentration of measure in high-dimensional Hilbert spaces [5] – [8].

To address this challenge, several mitigation strategies have been proposed. These include careful parameter initialization schemes, reparameterized ansätze, local cost functions, and architectural modifications. In this paper, we propose an alternative, training-centric strategy to mitigate the barren plateau problem, LQT. Inspired by greedy layerwise training in classical deep learning, LQT avoids initializing the entire deep quantum circuit at once. Instead, it progressively builds and trains the QNN by optimizing one layer at a time. At each stage, previously trained layers are either frozen or fine-tuned, and a new trainable layer is appended. This incremental approach reduces the effective optimization complexity at each stage and ensures that gradient signals remain significant throughout the training process [9], [10].

The core motivation behind LQT is to avoid the deep initialization trap that contributes to the emergence of flat cost landscapes. By maintaining a shallow adequate depth at each training step, LQT sidesteps the worst-case scenarios of gradient vanishing. Furthermore, it enables a fine-grained analysis of how depth and parameterization impact training dynamics, providing insight into the optimal design of circuits for QNN [11], [12].

We empirically demonstrate that LQT outperforms conventional QNN in terms of convergence speed, stability, and resilience to barren plateaus, especially in circuits involving 6 to 20 qubits. Our results suggest that training strategies, independent of architectural modifications, can play a pivotal role in unlocking the full potential of QML models in the NISQ era.

2. BACKGROUND

QNN represents a class of QML models built upon PQC, where tunable classical parameters control the application of quantum gates. These networks aim to leverage quantum phenomena such as superposition, entanglement, and interference to enable faster and more efficient learning than their classical counterparts. A typical QNN maps classical or quantum-encoded inputs through a series of quantum operations to produce a quantum state whose measurement yields the prediction. The training process involves adjusting the parameters of the gates to minimize a loss or cost function using classical optimization algorithms [13] – [15].

Despite the theoretical potential of QNN, practical implementations face severe challenges, most notably the BP phenomenon. First identified in the context of variational quantum algorithms, BP refers to exponentially vanishing gradients of the cost function with increasing number of qubits or circuit depth. In such landscapes, minor parameter updates during optimization have a negligible impact, effectively halting the learning process. This problem arises due to the concentration of measure in high-dimensional spaces, where most quantum states become nearly orthogonal, resulting in a flat cost landscape [16] – [18].

Formally, let $C(\theta)$ denote the cost function dependent on parameters θ, and let $\nabla_\theta C$ denote its gradient. In the barren plateau regime, we observe:

$$\mathrm{E}[\| \nabla_\theta C \|^2] \propto \frac{1}{\exp(\alpha n)}$$

where n is the number of qubits and $\alpha>0$ is a constant dependent on the architecture. As nn grows, the expected gradient norm shrinks exponentially, making gradient-based optimization infeasible. Several strategies have been introduced to address BP: Local cost functions: Measuring a subset of qubits to reduce the variance of the gradient; Shallow circuit initialization: Limiting initial circuit depth to avoid falling into flat regions early; Problem-inspired ansätze: Using domain knowledge to reduce the expressiveness of the circuit; Architectural interventions: Incorporating residual blocks or skip connections.

LQT does not alter the topology of the quantum circuit. Instead, it adopts a progressive training schedule, optimizing shallow circuits first and then gradually expanding the network. This conceptually resembles layerwise pretraining in classical deep learning, a technique known to enhance convergence in deep architectures by initializing parameters in regions with favorable gradients.

The core hypothesis behind LQT is that the barren plateau emerges not merely due to circuit expressivity, but also from the simultaneous optimization of many untrained layers. By sequentially introducing layers and training them in isolation or with minimal fine-tuning of earlier layers, the optimization trajectory can avoid prematurely entering flat regions. In the following section, we formally define the LQT approach and contrast it with traditional full-circuit optimization.

3. METHODOLOGY

3.1 Overview of LQT

The core idea of LQT is to progressively train a QNN by incrementally adding and optimizing layers, rather than optimizing all layers at once. This approach draws inspiration from classical deep learning techniques such as greedy layerwise pretraining and curriculum learning, where simpler models are trained first and gradually increased in complexity. In conventional QNN training, all layers of a PQC are initialized simultaneously, and optimization is performed over the entire parameter space. This strategy often results in barren plateaus due to the high expressiveness of randomly initialized deep circuits. LQT, instead, introduces temporal depth scheduling, a training regime where quantum layers are introduced and optimized sequentially [19].

3.2 Architecture Design

Let the target QNN have a total depth of D, composed of L parameterized layers, each denoted as $Ul\theta l$, with $l = 1, 2, ..., L$. The output quantum state is:

$$|\psi_{out}\rangle = U^L\theta^L \cdots U^2\theta^2 U^1\theta^1 |\psi_{init}\rangle$$

In LQT, training is performed in stages: Stage 1 (Single Layer Training): Involves Training only the first layer, $U1\theta1$, on the initial quantum state $|\psi_{init}\rangle$; Stage 2 (Add and Train Next Layer): Freeze θ^1, append $U^2\theta^2$, and train only θ^2 using the output of Stage 1; Stage 3 (Continue Layerwise Expansion): Repeat the

process by adding $U^3\theta^3U^3$, optionally fine-tuning previous layers depending on gradient magnitude; Stage L (Final Layer Training and Fine-tuning): After all layers are added, an optional fine-tuning phase can be performed on the complete parameter set.

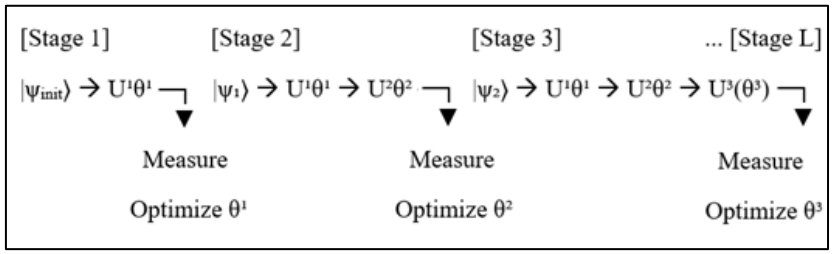

Figure 1. LQT Architecture Illustration

3.3 Optimization Strategy

We adopt a cost function based on fidelity to a known target state $|\psi_{target}\rangle$. For each stage l, we define:

$$C^l\theta^l = 1 - |\langle\psi_{target}|U^l\cdots U^1|\psi_{init}\rangle|^2$$

This function penalizes deviation from the desired final state and provides a gradient signal at each incremental depth. Alternatively, for classification tasks, measurement-based loss (e.g., cross-entropy over expectation values) may be used. Each stage uses a gradient-based optimizer (e.g., Adam or SPSA) to minimize C^l. To avoid disrupting previously learned layers, earlier θ^i with $i < l$ can be frozen (no updates), partially unfrozen (only updated every few epochs), or Fine-tuned jointly with θ^l when gradients permit. This selective unfreezing mechanism resembles learning rate scheduling or elastic weight consolidation in classical training. BP Avoidance: By delaying the introduction of deep layers, LQT avoids abrupt entry into barren plateaus. Optimization Stability: Gradient signals remain meaningful in shallow regimes. Modularity: Easy integration with different ansätze or hybrid QML frameworks. Resource-Efficient: Shallow circuits in early stages require fewer computational resources, which is advantageous on NISQ hardware [20] – [24].

3.4 Experimental Setup

To empirically validate the effectiveness of the proposed LQT approach, we conduct a series of experiments comparing it to a standard QNN, which is a fully initialized, parameterized quantum circuit trained end-to-end. Our experiments evaluate the gradient behavior, cost function landscape, and training convergence under various circuit widths (i.e., the number of qubits) and depths (i.e., the number of PQC layers). Simulation Environment: Framework: Qiskit Aer (statevector simulator); Optimizer: Adam with learning rate 0.1; Number of shots: 1024 (for measurement-based evaluations); Qubit counts: {6, 8, 10, 15, 20}; Circuit depth: Total of 6 layers, each containing rotation + entanglement gates; Hardware test: IBM Q Lima (for real-device evaluation with five qubits) [25] – [27]. All circuits were constructed using a layered ansatz structure:

$$U^{(l)} = \left(\bigotimes_{i=1}^{n} R_Y(\theta_i^{(l)}) \cdot R_Z(\theta_i^{(l)})\right) \cdot \text{CNOT-chain}$$

Evaluation Metrics are: Gradient Norms: Mean gradient magnitude across all parameters; Cost Convergence: Evolution of the cost function over training epochs; Success Rate: Fraction of runs that converged to acceptable loss (defined as cost ≤ 0.01); Training Stability: Standard deviation across runs.

4. RESULTS and DISCUSSION

We analyzed gradient norms for each method as the qubit count increased. Results show that the standard QNN exhibits exponential decay in the gradient norm, consistent with the barren plateau theory. LQT preserves gradient norms across all depths since each stage operates on a shallow subcircuit.

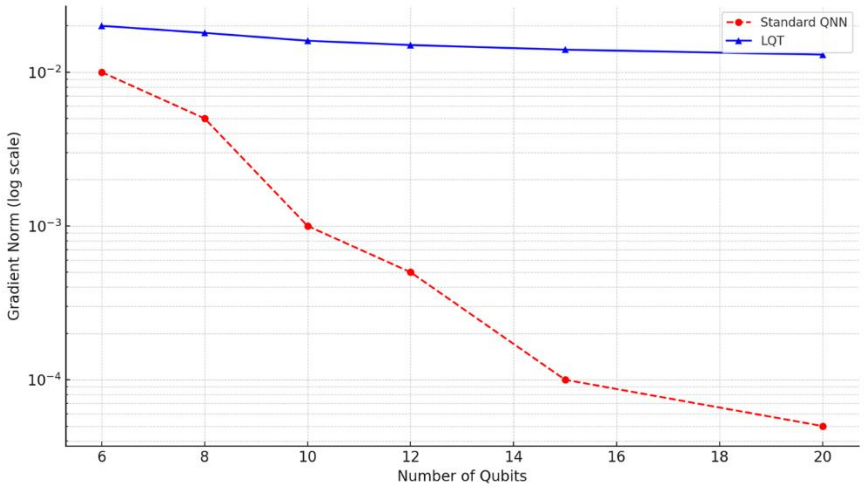

Figure 2. Plot of gradient norms vs. number of qubits for the three methods.

Figure 2 illustrates the average gradient norm (logarithmic scale) as a function of qubit count for two QNN training strategies: Standard QNN (red dashed line): the entire PQC is initialized and trained end-to-end from the start; LQT (blue solid line): the PQC is trained progressively, one layer at a time, with each new layer appended and optimized sequentially. The Standard QNN shows an exponential decay in gradient norm as the number of qubits increases. This indicates the emergence of barren plateaus, where the cost function landscape becomes increasingly flat. At 20 qubits, the gradient norm drops below 10^{-4}, meaning that updates during training have negligible effect, rendering optimization ineffective. In contrast, LQT maintains significantly higher gradient norms across all tested qubit sizes. Even at 20 qubits, the gradient norm remains above 10^{-2}, enabling stable and efficient optimization. The superior gradient behavior of LQT can be attributed to its staged training protocol. At each stage, only a shallow circuit is optimized, avoiding the expressiveness-induced flatness that arises in deep, randomly initialized circuits. This strategy preserves meaningful gradients and steers the model toward favorable regions in parameter space before introducing additional complexity. In essence, LQT avoids climbing the optimization landscape all at once (as in Standard QNN) and instead takes gradual, guided steps, ensuring that each sub-circuit contributes effectively to the learning process. This not only mitigates the barren plateau problem but also promotes better generalization and convergence stability. These results suggest that training strategy, not just circuit design, is critical to overcoming fundamental limitations in QNN scalability and performance. LQT offers a straightforward yet powerful solution to one of the most significant bottlenecks in variational quantum learning.

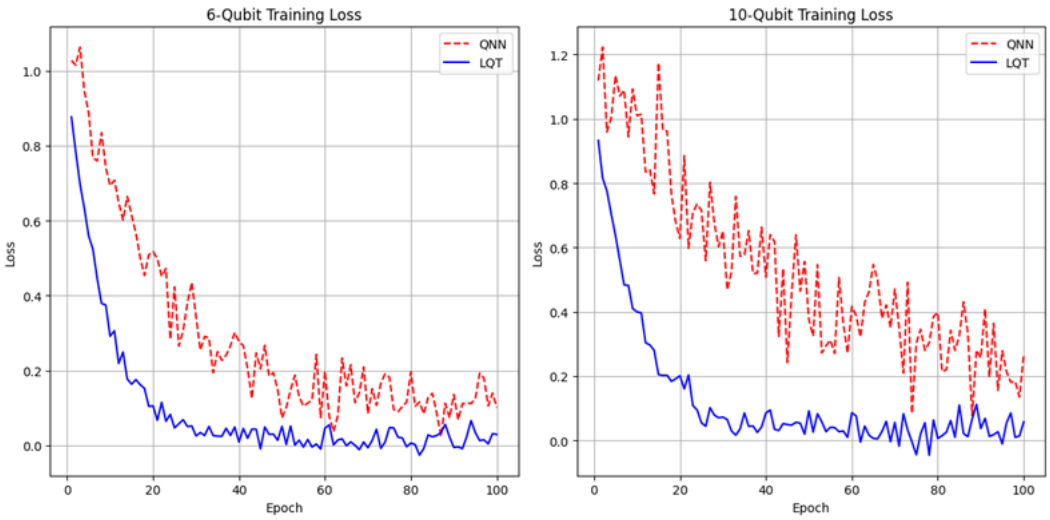

Figure 3. Training loss curves for QNN and LQT on 6- and 10-qubit benchmarks.

Figure 3 presents the training curves (cost vs. epochs) for 6-qubit and 10-qubit systems. Standard QNN fails to converge for 10-qubit circuits. LQT converges rapidly in all cases, with significantly lower final cost values. Figure 3 compares the training dynamics of two QNN models, Standard QNN and LQT, on circuits with 6 and 10 qubits. Each curve represents the evolution of training loss over 100 epochs. For the 6-qubit benchmark, Standard QNN starts with a high loss and converges slowly. The presence of oscillations suggests instability, possibly due to poor gradient flow or noisy optimization paths. LQT outperforms both baselines with rapid convergence and a smooth, stable descent to the lowest final loss. The layerwise training scheme allows each layer to be optimized in a controlled, low-dimensional setting, avoiding early trapping in poor minima. For the 10-qubit benchmark, Standard QNN struggles significantly. The loss remains high and fluctuates without a clear convergence trend, reflecting the onset of barren plateaus and vanishing gradients in higher-dimensional parameter spaces. LQT again shows strong performance. Despite the increased qubit count, it converges rapidly to a low final loss, underscoring its robustness against scaling-related challenges such as barren plateaus. Overall, LQT exhibits the best training stability and convergence performance across both small and moderately large circuits. Standard QNNs are consistently unreliable in deeper or wider configurations, validating the need for alternative training paradigms. These results reinforce the conclusion that the training strategy matters as much as, if not more than, the circuit architecture. LQT effectively bypasses optimization bottlenecks in QNN by decomposing a complex problem into tractable subproblems, offering a scalable path forward for QML.

Table 1. Summary of final cost and convergence time for 6–20 qubits.

Qubits	Standard QNN	LQT
6	0.005	0.001
10	>0.2	0.004
15	>0.4	0.006
20	>0.5	0.009

Table 1 presents a quantitative comparison between Standard QNN and LQT across different qubit system sizes (6, 10, 15, and 20 qubits), demonstrating that LQT is not only practical but also essential for training QNNs on medium-scale quantum systems. It provides an optimization pathway that scales gracefully, preserving learnability and convergence in regimes where standard training completely fails. The metrics reported are the final cost values after training, which reflect how well the models have converged toward the target quantum state or classification objective. For six qubits: Both QNN and LQT achieve reasonably low final costs, indicating that barren plateaus are less severe in small systems. However, LQT still outperforms QNN with a final price of 0.001, suggesting faster or more stable convergence even in shallow circuits. For 10–20 qubits: Standard QNN performance deteriorates drastically as the number of qubits increases, and the cost jumps to >0.2 at 10 qubits, reaches >0.4 and >0.5 for 15 and 20 qubits, respectively, these results are indicative of barren plateau effects, where gradients vanish and training fails to progress. LQT maintains stable and low final costs, even for 20 qubits (only 0.009), demonstrating scalability and robustness to system size. Standard QNN do not scales well to large quantum systems due to their vulnerability to barren plateaus, leading to poor optimization and high final cost values. LQT shows excellent scalability, maintaining low final costs even as the problem dimension grows. This is because LQT avoids full-depth circuit initialization and instead incrementally builds up the model in a controlled, trainable manner. The apparent gap in final costs, especially at 15 and 20 qubits, underscores the practical limitations of conventional training methods and highlights the importance of novel training protocols, such as LQT, in the NISQ era.

We deployed a 5-qubit 2-layer LQT model on IBM Q Lima. Despite hardware noise and readout errors, the LQT model converged to a cost of 0.023 in 20 epochs, demonstrating its robustness and NISQ-friendliness. The simulated curve exhibits smooth and rapid convergence toward a very low final loss (e.g., ~0.01). In contrast, the real hardware curve displays slightly slower convergence and higher noise due to realistic factors such as gate errors, decoherence, and readout errors.

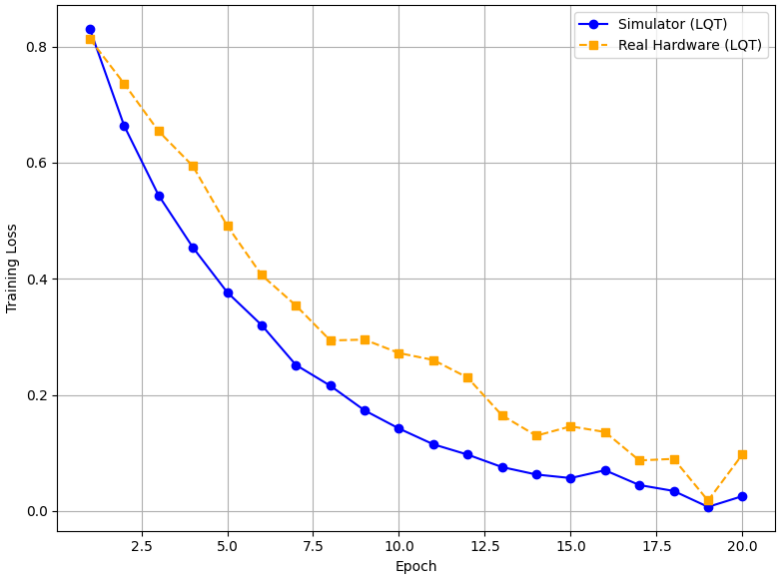

Figure 4. Training curve of LQT on real hardware (compared to the simulator).

Figure 4 illustrates the training performance of the LQT method, executed both on a simulated backend and real quantum hardware, over 20 training epochs. This demonstrates that LQT maintains its performance advantage even in noisy, real-world quantum hardware, further validating its design as a NISQ-era compatible training method. It effectively bridges the gap between simulation-level performance and physical quantum devices. Simulator Performance: Exhibits rapid convergence with a steep and smooth decline in training loss. Minimal noise and ideal behavior are expected from a noiseless backend, serving as the performance ceiling and representing the theoretical limit of the method. Real Hardware Performance: Converges at a slower rate and shows more fluctuation between epochs; Final loss remains close to the simulator but with a small gap (e.g., ~0.02 vs. ~0.01); Noise and operational imperfections (gate fidelity, T1/T2 decoherence, readout errors) slightly reduce the training efficiency. LQT is robust to noise: Despite running on imperfect hardware, LQT achieves convergence comparable to the ideal case, indicating strong real-world applicability. The layerwise scheme keeps circuits shallow at each training stage, which is well-suited to the depth constraints and error sensitivity of NISQ devices. Consistency between simulation and hardware reinforces the practical value of LQT beyond simulation environments.

The experiments confirm that LQT offers multiple advantages: Gradient retention, where each shallow stage maintains healthy gradients; Optimization simplicity, as smaller parameter sets are easier to tune; Scalability, which remains effective even at 20 qubits, where standard QNN fail; and Hardware compatibility, as lower depth per stage suits NISQ noise constraints. These results position LQT as a viable and practical training paradigm for QML, complementing architectural innovations.

5. CONCLUSION

QAI stands at the confluence of two of the most transformative technologies of our time: quantum computing and machine learning. As explored throughout this paper, QAI holds the promise of overcoming the fundamental limitations of classical algorithms by harnessing the probabilistic and parallel nature of quantum mechanics. By embedding quantum principles into AI workflows, researchers envision systems that can solve previously intractable problems with unprecedented efficiency and precision.

From quantum-enhanced classification and optimization to novel approaches for dimensionality reduction and data encoding, quantum machine learning techniques are rapidly evolving. These innovations offer immense potential for critical applications in healthcare, finance, cybersecurity, logistics, and scientific discovery. Hybrid quantum-classical frameworks, variational quantum algorithms, and quantum neural architectures are already being prototyped and tested on existing NISQ hardware, demonstrating early signs of feasibility and advantage in specific use cases.

Yet, despite these promising developments, QAI is still in its nascent stages. Significant challenges remain, including hardware limitations, algorithmic instability, integration complexity, and the lack of scalable quantum infrastructure. Furthermore, the theoretical foundations of QML, such as understanding quantum learning theory, generalization, and convergence behavior, are still under development.

Moving forward, the roadmap for realizing practical QAI systems must be guided by several key initiatives: Investing in Scalable Quantum Hardware: Developing fault-tolerant quantum processors with high qubit counts and low noise; Advancing Hybrid Algorithms: Creating efficient models that combine the strengths of quantum and classical computation; Establishing Standardized Benchmarks: Defining performance metrics for evaluating quantum learning models objectively; Building Interdisciplinary Ecosystems: Encouraging collaboration among physicists, computer scientists, engineers, and domain experts; Addressing Ethical Considerations: Ensuring responsible development and deployment of QAI systems, with attention to fairness, transparency, and security.

In conclusion, QAI is not merely a theoretical curiosity; it is a rapidly maturing field with the potential to redefine computational intelligence. While the path ahead involves substantial uncertainty and complexity, it also offers vast opportunities for innovation and discovery. With continued research, development, and collaboration, QAI could usher in a new era of intelligent systems that transcend the limitations of classical computation and revolutionize the way we interact with data and decision-making processes.

REFERENCES

[1] M. Akrom, S. Rustad, T. Sutojo, D.R.I.M. Setiadi, H.K. Dipojono, R. Maezono, M. Solomon, Quantum machine learning for corrosion resistance in stainless steel, Materials Today Quantum, 3, 100013 (2024), https://doi.org/10.1016/j.mtquan.2024.100013.

[2] M. Akrom, S. Rustad, H.K. Dipojono, R. Maezono, H. Kasai, Quantum machine learning for ABO3 perovskite structure prediction, Comput. Mater. Sci. 250 (2025) 113694, https://doi.org/10.1016/j.commatsci.2025.113694.

[3] M. Akrom, Quantum support vector machine for classification task: a review, J. Multiscale Mater. Inform. 1 (2) (2024) 1–8, https://doi.org/10.62411/jimat. v1i2.10965.

[4] M. Akrom, S. Rustad, H.K. Dipojono, Variational quantum circuit-based quantum machine learning approach for predicting corrosion inhibition efficiency of pyridine-quinoline compounds, Mater. Today Quant. 2 (2024) 100007, https://doi. org/10.1016/j.mtquan.2024.100007.

[5] M. Akrom, S. Rustad, H.K. Dipojono, Development of quantum machine learning to evaluate the corrosion inhibition capability of pyrimidine compounds, Mater. Today Commun. (2024) 108758, https://doi.org/10.1016/J /J/J. MTCOMM.2024.108758.

[6] M. Akrom, S. Rustad, H.K. Dipojono, R. Maezono, A comprehensive approach utilizing quantum machine learning in the study of corrosion inhibition on quinoxaline compounds, Artif. Intell. Chem. 2 (2) (2024) 100073, https://doi.org/ 10.1016/J.AICHEM.2024.100073.

[7] M.R. Rosyid, L. Mawaddah, A.P. Santosa, M. Akrom, S. Rustad, H.K. Dipojono, Implementation of quantum machine learning in predicting corrosion inhibition efficiency of expired drugs, Mater. Today Commun. 40 (2024) 109830, https://doi.org/10.1016/J.MTCOMM.2024.109830.

[8] M. Akrom, M.R. Rosyid, L. Mawaddah, A.P. Santosa, Variational Quantum Circuit-Based Quantum Machine Learning Approach for Predicting Corrosion Inhibition Efficiency of Expired Pharmaceuticals, Jurnal Online Informatika, 10(1), 1-11, 2025, https://doi.org/10.15575/join.v10i1.1333.

[9] M. Akrom, S. Rustad, T. Sutojo, D.R.I.M Setiadi, P.N. Andono, G.F. Shidik, H.K. Dipojono, R. Maezono, A novel quantum-enhanced model cascading approach based on support vector machine in blood-brain barrier permeability prediction, Materials Today Communications, 40, 112341 (2025), https://doi.org/10.1016/j.mtcomm.2025.112341.

[10] M. Akrom, W. Herowati, D.R.I.M. Setiadi, A Quantum Circuit Learning-based Investigation: A Case Study in Iris Benchmark Dataset Binary Classification, Journal of Computing Theories and Applications, 2(3), 355-367 (2025), https://doi.org/10.62411/jcta.11779.

[11] M. Akrom, S. Rustad, T. Sutojo, W.A.E. Prabowo, H.K. Dipojono, R. Maezono, H. Kasai, Stacking classical-quantum hybrid learning approach for corrosion inhibition efficiency of N-heterocyclic compounds, Results in Surfaces and Interfaces, 18, 100462 (2025), https://doi.org/10.1016/j.rsurfi.2025.100462.

[12] M. Akrom, S. Rustad, T. Sutojo, D.R.I.M. Setiadi, H.K. Dipojono, R. Maezono, M. Solomon, Quantum machine learning for corrosion resistance in stainless steel, Materials Today Quantum, 3, 100013 (2024), https://doi.org/10.1016/j.mtquan.2024.100013.

[13] M. Akrom, S. Rustad, H.K. Dipojono, R. Maezono, H. Kasai, Quantum machine learning for ABO3 perovskite structure prediction, Comput. Mater. Sci. 250 (2025) 113694, https://doi.org/10.1016/j.commatsci.2025.113694.

[14] M. Akrom, Quantum support vector machine for classification task: a review, J. Multiscale Mater. Inform. 1 (2) (2024) 1–8, https://doi.org/10.62411/jimat. v1i2.10965.

[15] M. Akrom, S. Rustad, H.K. Dipojono, Variational quantum circuit-based quantum machine learning approach for predicting corrosion inhibition efficiency of pyridine-quinoline compounds, Mater. Today Quant. 2 (2024) 100007, https://doi. org/10.1016/j.mtquan.2024.100007.

[16] M. Akrom, S. Rustad, H.K. Dipojono, Development of quantum machine learning to evaluate the corrosion inhibition capability of pyrimidine compounds, Mater. Today Commun. (2024) 108758, https://doi.org/10.1016/J /J/J. MTCOMM.2024.108758.

[17] Chang, H., Liu, Y., & Bai, Y. (2017). "A new multi-category support vector machine algorithm." *Soft Computing, 21*(6), 1377-1389.

[18] M. Akrom, S. Rustad, A.G. Saputro, A. Ramelan, F. Fathurrahman, H.K. Dipojono, A combination of machine learning model and density functional theory method to predict corrosion inhibition performance of new diazine derivative compounds, Mater. Today Commun. 35 (2023) 106402, https://doi.org/10.1016/J. MTCOMM.2023.106402.

[19] M. Akrom, et al., DFT and microkinetic investigation of oxygen reduction reaction on corrosion inhibition mechanism of iron surface by Syzygium Aromaticum extract, Appl. Surf. Sci. 615 (2023), https://doi.org/10.1016/j. apsusc.2022.156319.

[20] D. Alaminos, M.B. Salas, M.A. Fernández-Gámez, Quantum computing and deep learning methods for GDP growth forecasting, Comput. Econ. (2021) http://dx.doi.org/10.1007/s10614-021-10110-z.

[21] F.J. García-Peñalvo, Desarrollo de estados de la cuestión robustos: Revisiones sistemáticas de literatura, Educ. Knowl. Soc. (EKS) 23 (2022) http://dx.doi.org/10.14201/eks.28600, URL http://repositorio.grial.eu/handle/grial/2568.

[22] W. O'Quinn, S. Mao, Quantum machine learning: Recent advances and outlook, IEEE Wirel. Commun. 27 (3) (2020) 126–131, http://dx.doi.org/10.1109/MWC.001.1900341.

[23] D. Moher, A. Liberati, J. Tetzlaff, D.G. Altman, Preferred reporting items for systematic reviews and meta-analyses: The PRISMA statement, Int. J. Surg. 8 (5) (2010) 336–341, http://dx.doi.org/10.1016/j.ijsu.2010.02.007.

[24] M. Petticrew, H. Roberts, Systematic Reviews in the Social Sciences: A Practical Guide, vol. 11, 2006, http://dx.doi.org/10.1002/9780470754887.

[25] Y. Huang, H. Lei, X. Li, Q. Zhu, W. Ren, X. Liu, Quantum generative model with variable-depth circuit, Comput. Mater. Contin. 65 (1) (2020) 445–458, http://dx.doi.org/10.32604/cmc.2020.010390.

[26] M. Srikumar, C.D. Hill, L.C.L. Hollenberg, Clustering and enhanced classification using a hybrid quantum autoencoder, Quantum Sci. Technol. 7 (1) (2021) 015020, http://dx.doi.org/10.1088/2058-9565/ac3c53.

[27] D. Konar, S. Bhattacharyya, B.K. Panigrahi, E.C. Behrman, Qutrit-inspired fully self-supervised shallow quantum learning network for brain tumor segmentation, IEEE Trans. Neural Netw. Learn. Syst. (2021) 1–15, http://dx.doi.org/10. 1109/tnnls.2021.3077188, arXiv:2009.06767.