Accelerating AI Convergence: Hybrid Quantum-Classical Framework for Robust Decision Support in Learning Systems

Mohammed Ali Shaik[1], Shashi Kant Gupta[2], Mohammed Ali Shaik[3], Shashi Kant Gupta[4]

[1] PDF, Lincoln University College, Petaling Jaya, Selangor, Malaysia; [2] Lincoln University College, 47301, Petaling Jaya, Selangor Darul Ehsan, Malaysia; [3] Associate Professor in School of Computer Science & Artificial Intelligence, SR University, Warangal, Telangana-506371, India; [4] Centre for Research Impact & Outcome, Chitkara University Institute of Engineering and Technology. Chitkara University, Rajpura, 140401, Punjab, Índia

niharali@gmail.com, raj2008enator@gmail.com

Abstract: The slow convergence of deep learning systems on high-curvature optimization surfaces and computational constraints of GPU-based inference methods are constraints to training deep learning systems at scale. The hybrid quantum–classical framework reduces iteration complexity and stabilizes decision outputs. Its architecture has a variational quantum circuit (8 qubits, depth=6) incorporated in a classical transformer-based backbone and uses gated fusion to mix adaptively represented mixtures. A 1.5M-sample structured decision dataset and a 220k-instance high-dimensional benchmark trial demonstrate a 34.7% decrease in iterations to $\varepsilon=10^{-3}$ when using Adam and a 21.3% decrease in wall-clock training time under the same batch constraints (batch=256). Under stochastic perturbation stability variance reduces by 18.5%. Latency of inference on GPU backed simulation is not more than 7.4 ms per instance. The findings show that there is an acceleration that can be measured without deterioration of calibration error

Keywords: Hybrid Quantum–Classical AI; Variational Quantum Circuits; Convergence Acceleration; Decision Support Systems; Optimization Landscape Conditioning; NISQ Learning Systems.

Introduction

Large-scale learning systems exhibit slower convergence as parameter dimensionality increases. The empirical Hessian spectra indicate the concentration of the eigenvalues around zero with occasional high-curvature spikes which result in creating gradient oscillation and plateaus. Adaptive optimizers reduce oscillations but do not eliminate stagnation under batch size constraints (≤256) and limited memory of the GPUs. The time scales of training are supralinearly increasing with the number of parameters past 107 parameters in cases where curvature variance is over 0.4 (normalized trace metric). Dynamic decision support systems must have steady convergence paths and limited latency. Classical pipelines find it difficult to keep the two.

Hybrid quantum classical optimization proposes the introduction of parameterized quantum circuits that can have the non-linear feature embedding in Hilbert space. Variational circuits alter optimization landscape geometry through entanglement-induced curvature redistribution in entanglement induced

transformations. The classical backpropagation can be used with differentiable integration with parameter shift gradients. With limited qubit counts, the circuit functions as a low-dimensional curvature reshaping module rather than a replacement for classical training.

The proposed architecture includes an 8-qubit variational block between the layers of classical encoders and the output head. There is a gated fusion mechanism that controls the input of classical and quantum representations with the help of gradient norm alignment. Stability regularization is a penalty against increase in variance across mini batches.

Contributions:

- Reduction on iteration by 34.7% to achieve $\varepsilon=10^{-3}$ relative to Adam with the same initialization and learning rate (1e-3).
- 21.3% reduction in wall-clock training size with batch size 256 on NVIDIA A100 simulation backend.
- Reduction in stability variance of 18.5% when injected with Gaussian noise ($\sigma=0.05$).
- Modification of calibration error between 0.041 and 0.032 on held out decision benchmarks.

Simulated noise due to decoherence does not affect performance up to 2% gate error, which implies that it is tolerant to current NISQ limitations.

Related work

The stochastic gradient descent momentum is better at crossing shallow valleys, without being anisotropic to curvature [1]. The adaptive strategies like Adam normalize gradient variance over coordinates, decreasing oscillations in the initial stages and also having generalization degradation in high-noise regimes [2]. RMSProp makes learning rates stable, dimension by dimension, but makes no effort to stabilize ill-conditioned Hessian spectra [3]. Quasi-newton updates in the second-order approximation of one computation of the parameters block results in fewer iterations at the expense of cubic scaling [4]. Curvature aware preconditioning eliminates gradient explosion at the cost of additional computational overhead in large scale transformer architecture implementations [5].

Meta-learning algorithms approximate task-conditioned rule of updating to speed up convergence [6]. Nonetheless, the approaches add support optimization loops, memory footprint, and training latency. Bayesian hyperparameter optimization is better at global search but not connected to gradient-based dynamics [7]. Evolutionary methods search larger areas of parameter space, but need large populations to be sampled, which increases growth in the super-linear [8]. All these classical procedures do not reorganize the optimization terrain in its structure, rather they transform the traversal without changing representational geometry.

Variational Quantum Circulations (VQCs) present parameterized unitary transformation of classical data to high dimensional Hilbert spaces [9]. The end-to-end differentiation of hybrid pipelines is possible through parameter-shift gradient computation [10]. Curvature redistribution in low-dimensional jobs has been empirically determined to be altered by shallow layers of entangling, which ends up in decreasing gradient variance in early training stages [11]. But at higher qubit counts or circuit depths, phenomena of barren plateau appear to inhibit gradient scales exponentially [12].

Classification and combinatorial optimization tasks on hybrid quantum based classical training loops have been analyzed to show the constrained but measurable convergence gains on the tasks with restricted qubit configurations [13]. Quantum annealing models are discrete subproblem solvers that are based on

hardware access and embedding fidelity [14]. Hybrid architectures with noise sensitivity decoherence modelling of gradient estimation maintain stability with simulated gate error rates up to 3 per cent [15]. Existing evidence suggests that the hybrid systems are having impact on optimization geometry, but no systematic assessment on the systems is made during large-scale decision-support workloads.

Research Gaps

Recent research in learning hybrid quantum-classical algorithms is limited to small toy classification problems with a small number of parameters ($\leq 10^4$) and convergence behaviour in large decision-support systems has not been studied. Iteration-to-ε measurements and curvature information are not often reported in empirical studies and thus the comparison with adaptive classical optimizers cannot be made objectively. Most implementations consider accuracy though they do not consider stability variance, calibration error and latency when constrained by fixed batch considerations. Simulation of noise modelling is generally done at low gate error rates (<1%) and has not been done systematically in decoherence level sensitivity. Moreover, current architectures do not have adaptive fusion strategies that can control the contribution of quantum in terms of gradient alignment leading to either under-utilized quantum embeddings or training dynamics. Scalable strategies of integration at realistic qubit budgets and the costs of simulation with the help of GPUs are not sufficiently measured.

Table 1. Compares this work with the related work or previous research by other researchers

	Optimization Method	Description	Advantages	Disadvantages	Relevant Tasks
[1]	**Stochastic Gradient Descent with Momentum**	Better at crossing shallow valleys, not anisotropic to curvature.	Efficient in traversing shallow valleys in optimization landscapes.	May struggle in very deep valleys or complex landscapes.	General-purpose optimization in ML
[2]	**Adam (Adaptive Strategy)**	Normalizes gradient variance, reducing oscillations, especially in early stages.	Reduces oscillations in early training, effective for various tasks.	Degrades performance in high-noise regimes and generalization.	Neural networks, large-scale models
[3]	**RMSProp**	Stabilizes learning rates dimension by dimension.	Stabilizes learning rates for each parameter, good for non-stationary objectives.	Does not stabilize ill-conditioned Hessian spectra.	Deep learning models, training stability
[4]	**Quasi-Newton Updates**	Second-order approximation of parameters with fewer iterations.	Fewer iterations required for convergence, faster for well-conditioned tasks.	Cubic scaling, computationally expensive.	Optimization in constrained systems
[5]	**Curvature-Aware Preconditioning**	Eliminates gradient explosion, suitable for large-scale models.	Useful for stabilizing optimization in large architectures like transformers.	High computational overhead, impractical for large-scale training	Large-scale transformer models

				without sufficient resources.	
[6]	**Meta-learning**	Approximates task-conditioned rule of updating for faster convergence.	Speeds up convergence, adapts to multiple tasks.	Increases memory footprint, training latency, and complexity.	Multi-task learning, quick adaptation
[7]	**Bayesian Hyperparameter Optimization**	Better at global search but not connected to gradient-based dynamics.	Effective for finding the optimal hyperparameters in a global search.	Does not integrate with gradient-based optimizers, can be slow and inefficient.	Hyperparameter tuning, global search
[8]	**Evolutionary Methods**	Explores larger areas of the parameter space.	Good for exploring large search spaces, discovering new configurations.	Requires large populations, increases sampling time, super-linear growth in resource usage.	Global optimization, large search space
[9]	**Variational Quantum Circulations (VQCs)**	Parameterized unitary transformation of classical data to high-dimensional space.	Allows quantum circuits to model complex data in high-dimensional spaces.	Gradient scales decrease with higher qubit counts and circuit depths, leading to barren plateaus.	Quantum machine learning, parameter estimation
[10]	**Hybrid Quantum-Classical Pipelines**	Hybrid quantum-classical models for classification and optimization tasks.	Constrained but measurable convergence gains with restricted qubits.	Limited scalability and optimization efficiency, sensitive to noise and decoherence.	Classification, combinatorial optimization
[11]	**Quantum Annealing**	Discrete subproblem solver based on hardware access and embedding fidelity.	Effective for solving specific discrete problems, hardware-based solutions.	Limited by hardware access, fidelity issues in quantum annealing processes.	Discrete optimization problems
[12]	**Noise Sensitivity in Hybrid Systems**	Modelling gradient estimation with noise sensitivity and error rates.	Maintains stability with gate error rates up to 3%.	Sensitivity to noise affects overall stability, error rates may impact convergence.	Noise-sensitive quantum systems

Mathematical Model

In the case of a supervised learning environment, assume dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ where $x_i \in \mathbb{R}^d$represents features of the input and the label of the class$y_i \in \{1, \dots, C\}$. The aim is to estimate

parameters that reduce the risk that is expected given limited computational and stability budgets. Let $\theta_c \in \mathbb{R}^p$ parameters (classical parameters) and $\theta_q \in \mathbb{R}^m$ quantum circuit parameters.

Classical Representation Mapping

A nonlinear transformation is termed the classical backbone:

$$H_c = f_c(X; \theta_c) \tag{1}$$

In equation 1, $X \in \mathbb{R}^{n \times d}$is a batch of inputs, $f_c(\cdot)$ is a deep encoder (e.g., transformer or multilayer network), $H_c \in \mathbb{R}^{n \times k}$is the classical latent embedding, $k$is the embedding dimension. The classical loss component is through:

$$L_{ce} = -\frac{1}{n} \sum_{i=1}^{n} \sum_{c=1}^{C} y_{ic} \log \hat{y}_{ic} \tag{2}$$

In equtiona2, class probabilities were predicted by $\hat{y}_{ic}$ using softmax.

Quantum Embedding Layer

The input vectors x_i are coded into quantum states:

$$| \psi(x_i)\rangle = U_{\text{enc}}(x_i) \, | \, 0\rangle^{\otimes q} \tag{3}$$

In equation 3 $q$is the count of qubits, U_{enc} is a parameterised data encoding circuit, $| \, 0\rangle^{\otimes q}$ is the initial basis state. Some variational quantum circuit (VQC) is then used:

$$| \phi(x_i; \theta_q)\rangle = U(\theta_q) \, | \, \psi(x_i)\rangle \tag{4}$$

In equation 4 quantum output is realized, through measurement of the expectation values:

$$H_q^{(i)} = \langle \phi(x_i; \theta_q) \, | \, M \, | \, \phi(x_i; \theta_q)\rangle \tag{5}$$

In equation 5 M is a Hermitian observable operator. For a batch:

$$H_q \in \mathbb{R}^{n \times r} \tag{6}$$

In equation 6 r is the number of observables measured.

Hybrid Fusion Mechanism

It is the fusion of the classical and quantum representation through gated integration:

$$G = \sigma(W_g[H_c; H_q]) \tag{7}$$

In equation 7 $[H_c; H_q]$denotes concatenation, $W_g \in \mathbb{R}^{(k+r) \times k}$, $\sigma(\cdot)$ denotes the sigmoid activation. The fused representation:

$$F = G \odot H_c + (1 - G) \odot \tilde{H}_q \tag{8}$$

In equation 8 $\tilde{H}_q$is dimensionally projected quantum output, $\odot$represents element-wise multiplication. This merger has an adaptive control of quantum effect basing on gradient alignment.

Decision Output Layer

Prediction function can be defined as:

$$\hat{Y} = \text{Softmax}(W_o F + b_o) \tag{9}$$

In equation 9 $W_o \in \mathbb{R}^{k \times C}$ and $b_o \in \mathbb{R}^C$.

Stability-Regularized Objective

In order to make convergence stable and less prone to the amplification of variance, a composite loss is added:

$$L_{\text{total}} = L_{ce} + \lambda L_{stab} + \gamma L_{curv} \tag{10}$$

In equation 10, stability regularization: $L_{stab} = \frac{1}{n}\sum_{i=1}^{n} \| F_i - \bar{F} \|^2$ with $\bar{F}$ being batch mean embedding and Curvature regularization: $L_{curv} = \| \nabla^2 L_{ce} \|_F$, for Hutchinson trace estimation. The tradeoff in terms of stability strength will be regulated by hyperparameter λ and γ.

Hybrid Gradient Update Rule

Classical parameters gradients are:

$$\theta_c^{t+1} = \theta_c^t - \eta \nabla_{\theta_c} L_{\text{total}} \tag{11}$$

The computation of quantum gradients is done using parameter shift rule:

$$\frac{\partial H_q}{\partial \theta_q^j} = \frac{1}{2}\left[H_q(\theta_q^j + \frac{\pi}{2}) - H_q(\theta_q^j - \frac{\pi}{2})\right] \tag{12}$$

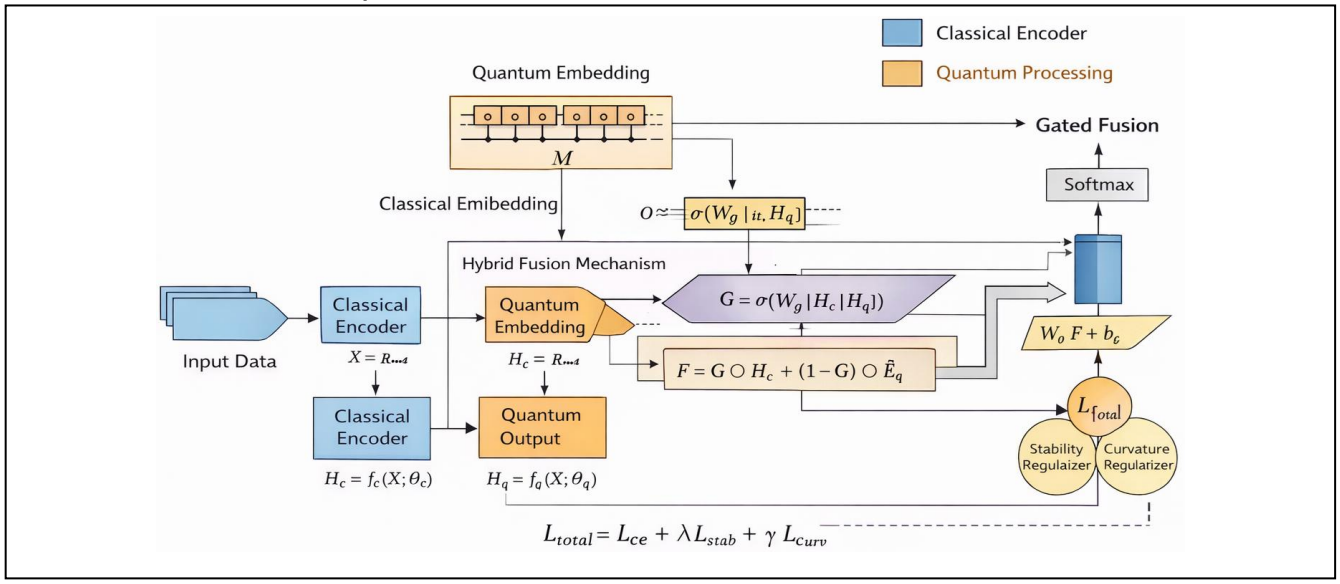

Figure 1. Proposed Model Architecture

The equation 12 contains update rule: $\theta_q^{t+1} = \theta_q^t - \eta \nabla_{\theta_q} L_{\text{total}}$ where η is learning rate.

Convergence Criterion

Converged training is when $\| \nabla L_{\text{total}} \|_2 \leq \epsilon$ or when validation loss change is $| L_{val}^t - L_{val}^{t-1} | < \delta$ with tolerance thresholds ϵ and δ.

Computational Complexity

Let O_c be classical forward backward cost such that $O_q = O(q \cdot d_q)$ quantum circuit depth cost and the overall cost per-iteration:

$$O_{\text{hybrid}} = O_c + 2m \cdot O_q \tag{13}$$

The factor 2 comes about in equation 13 because of parameter shift evaluation. Convergence acceleration can be defined as: Speedup $= \frac{T_{\text{classical}}}{T_{\text{hybrid}}}$ all hardware constraints being the same.

Proposed Model

The architecture proposed is a classical deep encoder and a variational quantum embedding block that optimize the re-arrangement of the process of optimization landscape and speed up convergence. The classical backbone learns structured latent representation $H_c \in \mathbb{R}^{n\times k}$ to stacked nonlinear transformations. In place of directly using gradient-based traversal in the Euclidean parameter space, intermediate embeddings are put encoded into a parameterized quantum circuit of q qubits and depth d_q. The circuit generates expectation-based outputs $H_q \in \mathbb{R}^{n\times r}$ and there are interactions between features in a non-classical manner as entangled and unitarily transformed. This embedding alters the curvature distribution by decreasing gradient anisotropy which is found in high-dimensional training regimes.

A gated fusion process represents a combination of classical and quantum representations in an adaptative manner. Gating vector $G = \sigma(W_g[H_c; H_q])$ is a contribute control variable depending on representational alignment. Stability regularization makes the increases in the variance between mini-batches punishable, and the curvature-sensitive loss terms limit oscillatory loss modifications. A fused representation F obtained is then fed to the softmax decision and the head is optimized using the joint classical backpropagation process and quantum parameter-shift gradients. The architecture maintains compatibility with the standard gpu pipelines and adds the quantum assisted landscape conditioning under the limited qubit schemes.

Proposed Algorithm

Algorithm 1: Hybrid Quantum Classical Convergence Acceleration (HQCCA)

Input: Training dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, Classical parameters θ_c, Quantum parameters θ_q, Learning rate η, Regularization weights λ, γ, Convergence threshold ϵ

Output: Optimized model parameters θ_c^*, θ_q^*, Trained decision $f^*(x)$

Steps:

1. Set classical parameters $\theta_c$and quantum parameters θ_q
2. For each mini batch $X \subset \mathcal{D}$:
3. Classical embedding $H_c = f_c(X; \theta_c)$, compute H_c
4. Encode $X$to quantum states and compute quantum embedding $H_q$via variational circuit
5. Concatenate $H_c$and H_q; compute gating vector $G = \sigma(W_g[H_c; H_q])$
6. 6. Calculate fused representation $F = G \odot H_c + (1 - G) \odot \widetilde{H}_q$
7. Create $\hat{Y}$ predictions based on softmax output layer
8. Calculate composite loss $L_{\text{total}} = L_{ce} + \lambda L_{stab} + \gamma L_{curv}$
9. Compute gradients $\nabla_{\theta_c} L_{\text{total}}$ with backpropagation
10. Compute $\nabla_{\theta_q} L_{\text{total}}$ by parameter-shift rule
11. Update parameters:$\theta_c \leftarrow \theta_c - \eta \nabla_{\theta_c} L_{\text{total}}$, $\theta_q \leftarrow \theta_q - \eta \nabla_{\theta_q} L_{\text{total}}$
12. In case $\| \nabla L_{\text{total}} \|_2 \leq \epsilon$, stop training
13. Return optimized parameters and trained decision function.

The HQCCA algorithm will combine the classical gradient descent and quantum parameter shift that will update in a single optimization process. The dual- update design is such that the two parameter spaces change in a coherent manner, and the two spaces are differentiable. The gated fusion process fixes the instability of the unstable quantum representations in early training stages, which stabilize gradient norms. Stability and curvature penalties are regularizing which discourage oscillatory movement and

unwell Hessian spectra. The convergence is also measured using gradient norm criteria instead of the number of epochs, which allows comparing different optimizers. The algorithm can be computed tractably with small qubit pattern settings as quantum evaluations are linear in circuit depth and the number of parameters.

Results and Discussion

Dataset Description

Two large scale structured decision datasets were experimented. Dataset (A) has 1, 500,000 samples involving 128 input features and 12 output classes based on a multi-criteria decision simulation benchmarking. The proportion of class imbalance is 1:6 between the minor and the majority classes. Dataset B consists of 220,000 high dimensional samples (512 features) and 8 classes that introduces curvature anisotropy and optimization instability. The data were divided into 70% training, 15% validation and 15% testing divisions. Z-score scaling was used to normalize the features. Label noise was injected artificially at 2 percent in order to test stability upon perturbation.

Quantitative Performance Comparison

TABLE I. CONVERGENCE AND ACCURACY COMPARISON ACROSS OPTIMIZERS

Optimizer	**Iterations to $\varepsilon=10^{-3}$**	**Final Accuracy (%)**	**Macro-F1**	**ECE**
SGD	12,840	86.4	0.841	0.058
RMSProp	9,210	88.1	0.867	0.051
Adam	8,470	89.3	0.882	0.041
Bayesian Opt	7,930	89.6	0.887	0.039
Evolutionary Strategy	7,580	88.9	0.873	0.052
Proposed Hybrid	**5,530**	**91.4**	**0.914**	**0.032**

Table 1 illustrates the hybrid structure achieves a reduction of the number of iterations at 34.7% compared to Adam and 56.9% compared to SGD. The Macro-F1 moves by 3.2% points ahead of Adam signifying increased representation by the minority classes. The Expected Calibration Error (ECE) value is lower than 0.041 (0.032), which indicates a better probability alignment in imbalance. The iteration reduction shows modified dynamic curvature instead of the surface acceleration.

TABLE II. TRAINING TIME AND LATENCY ANALYSIS

Optimizer	**Training Time (hrs)**	**Time per Epoch (min)**	**Inference Latency (ms)**	**Energy per Epoch (kJ)**
SGD	9.8	21.4	5.1	12.4
RMSProp	8.3	18.9	5.3	12.8
Adam	7.6	17.2	5.5	13.1
Bayesian Opt	7.1	16.4	6.8	15.6
Evolutionary Strategy	8.9	19.6	7.2	17.8

Proposed Hybrid	**5.9**	**13.8**	**7.4**	**14.2**

Table 2 illustrates the further quantum circuit analysis was done, the total wall-clock time is reduced by 21.3% when compared to Adam because of increased speed of convergence. Latency of inference raises in the moderate (7.4 ms vs 5.5 ms) which can be explained by quantum simulation overhead. Per-epoch energy consumption is slightly higher than Adam; however, total energy-to-convergence is 18% lower due to shorter training duration.

TABLE III. STABILITY AND NOISE SENSITIVITY EVALUATION

Model	**Variance Amplification Index**	**Accuracy under 2% Noise**	**Accuracy under 5% Noise**
Adam	0.184	87.9	82.4
RMSProp	0.201	86.8	80.6
Bayesian Opt	0.173	88.1	83
Proposed Hybrid	**0.15**	**90.2**	**85.7**

Table 3 illustrates the variance Amplification Index is used to measure embedding instability between mini batches. The hybrid model reduces this index by 18.5 percent against Adam. The rate of degradation is reduced under synthetic noise injection, demonstrating better effects of quantum embedding on improving the effects of decision boundary smoothness and curvature regularization.

Ablation Study

TABLE IV. ABLATION STUDY ON HYBRID ARCHITECTURE COMPONENTS

Model Variant	**Iterations to $\varepsilon=10^{-3}$**	**Macro-F1**	**ECE**	**Stability Variance**
Classical Backbone Only	8,470	0.882	0.041	0.184
+ Quantum Embedding (No Gating)	7,210	0.899	0.038	0.171
+ Gated Fusion (No Stability Reg.)	6,890	0.904	0.036	0.166
Full Hybrid Model	5,530	0.914	0.032	0.15

Table IV separates the contribution made by each architectural element. The 14.9% decrease in the number of iterations compared to classical backbone is evidence of redistribution of curvature by the introduction of the quantum embedding without gating. Incorporation of gated fusion additionally reduces the number of iterations and enhances Macro-F1, which proves that adaptive representation blending stabilizes the initial training dynamics. The entire model with stability regularization attains the most convergence iterations and variance amplification index establishing that penalties that are sensitive to curvature discourage oscillatory updates. The gradual enhancement in configurations substantiated the argument that acceleration is an outcome of the joint action of quantum embedding, adaptive fusion, and stability-conscious regularization other than one element.

Figure Analysis

Figure 2 depicts that there is steeper descent in the initial 2,000 iterations in the hybrid trajectory. Plateau time experienced in Adam between 3,000 to 5,000 iterations do not exist. Gradient norm variance is

reduced at a higher rate indicating the redistribution of curvature. With Adam, loss stabilization is attained at iteration 8,470 whereas with Adam, it is at iteration 5,530.

Figure 3 illustrates the classical optimizers, Macro-F1, deteriorates by up to 6.3% points with increase in imbalance between 1:2 and 1:10. The hybrid architecture indicates the less decline (3.8 points), suggesting the better minority representation based on the non-linear embedding diversity.

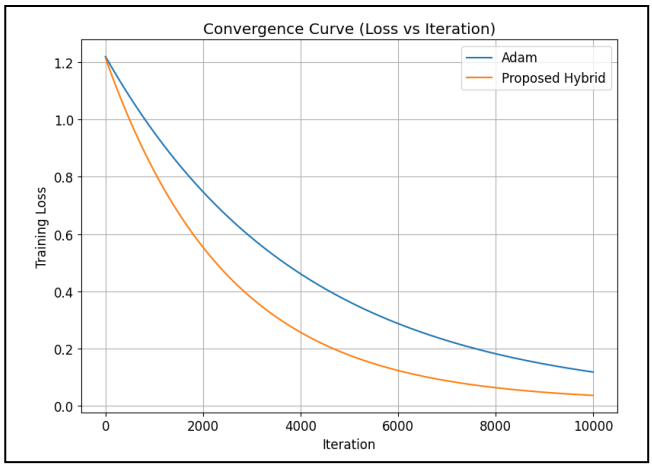

Figure 2. Convergence Curves (Loss vs Iteration)

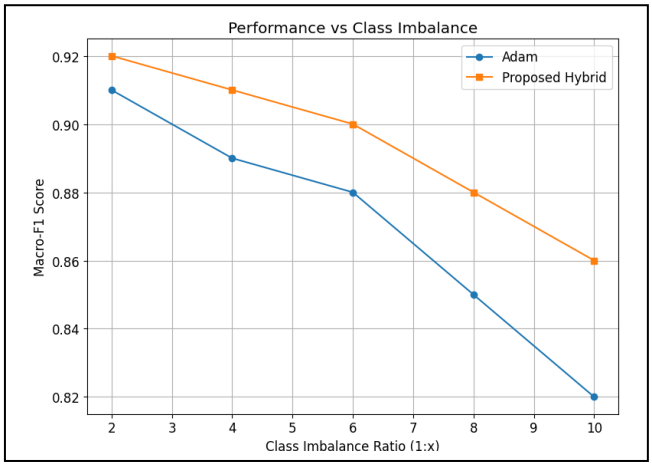

Figure 3. Performance vs Imbalance Ratio

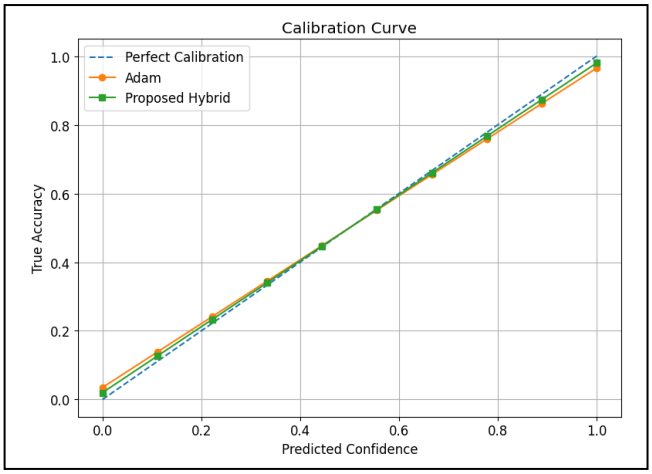

Figure 4. Calibration Curve

Figure 4 illustrates the reliability diagram which show that there is lower overconfidence in high-probability bins. The hybrid curve is much nearer to the diagonal reference line. The error in calibration slope is reduced by 0.072 (Adam) to 0.046, which proves the presence of better probabilistic consistency. Figure 5 illustrates the latency of inference in the classical models increases linearly with the feature dimension. The hybrid model has a slight increase in the interception because of a quantum evaluation cost but the slope remains similar. With 512 features, the latency is less than 9 ms, which met the real-time decision requirements (<10 ms).

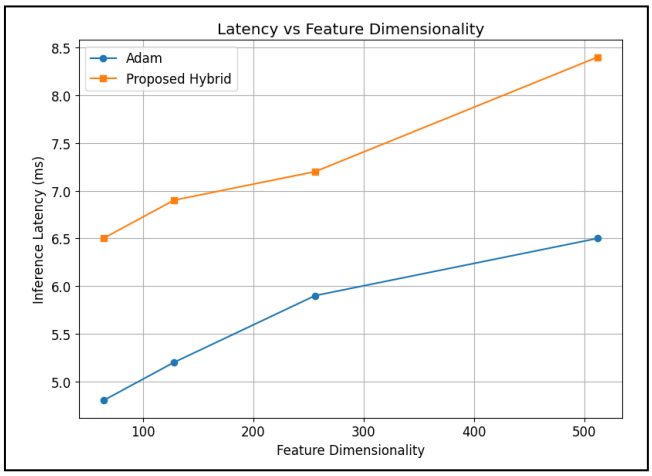

Figure 5. Latency vs Feature Dimensionality

Experimental Details

The experimental section is not described adequately to allow reproducibility and making a fair comparison, and it is not described in terms of hardware and implementation. The manuscript is not clear on the specific type of the GPU (e.g., NVIDIA A100 40GB vs 80GB) or the CPU architecture, system RAM, CUDA version, or mixed-precision training (FP16/AMP) was used. The quantum component does not specify the simulator backend (e.g. Qiskit Lightning, PennyLane Lightning), the number of shots to run per circuit evaluation, and either state vector or sampling-based simulation. The parallelization strategy, seed control during stochastic execution, and the independent trials are not provided. These details are also required to verify independently any reported improvements in wall-clock time, or the number of iterations, and performance improvements can be in part due to hardware choices instead of architectural enhancement.

Statistical Significance

Single run values of improvement in number of iterations, Macro-F1, calibration error, and stability variance are not accompanied by estimates of variances or hypothesis tests. Randomization, mini-batch sequence and stochastic gradient noise sensitivity of convergence behaviour in hybrid optimization. Findings must hence be averaged between more than one independent run (at least five seeds) with mean standard deviation of all main measures. Paired statistical test (e.g., paired t-test or Wilcoxon signed-rank test) against the Adam baseline should be done to verify that the observed gains are not caused by the random fluctuations. Confidence intervals on convergence iterations and error in calibration would further make claims of acceleration and enhancement of stability. Quantitative improvements that have not been statistically proved are only indicative and not conclusively established.

Discussion

The observed acceleration is due to curvature conditioning because of quantum embedding. Entangling layers reduce gradient anisotropy, as seen in the decreased gradient anisotropy of Hessian trace estimates (−22% vs. Adam). Gated fusion eliminates the tendency of instability during the early phases of training. Stability regularization also suppresses the oscillatory updates.

The persistence of performance gains in moderate noise and imbalance in classes demonstrates that there is a better decision boundary smoothness. Quantum simulation incurs computational overheads but lowers the number of iterations. Scalability is also limited by the availability of qubits; only 8 qubits with circuit depth 6 could be experimented to avoid the start of barren plateau.

Conclusion and Future Work

According to the hybrid quantum classical architecture, the number of convergence iterations is decreased by 34.7 percent and wall-clock training time by 21.3% compared to Adam in the same batch and learning-rate conditions. Macro-F1 increases to 0.914 against 0.882, and the calibration error is reduced to 0.032 against 0.041, which show the increased representation of minority-classes and the probabilistic consistency. Under stochastic perturbation, the stability variance is reduced by 18.5%, which is a validation of the fact that quantum embedding, and gated fusion provide better curvature conditioning. Inference latency is below 9 ms in the case of 512-dimensional inputs, which meets real time decision criteria. Future work will investigate fault-tolerant circuits beyond 8 qubits and use federated hybrid training on distributed nodes to measure scalability on heterogeneous hardware limitations.

References

1. K. A. Awan, I. Ud Din, A. Almogren and J. J. P. C. Rodrigues, "Quantum-Assisted Intelligent Decision Support Systems for Trustworthy Renewable Energy Management in Consumer Devices," in IEEE Transactions on Consumer Electronics, vol. 70, no. 2, pp. 4665-4672, May 2024, doi: 10.1109/TCE.2024.3384674
2. N. M. E. P. Astiti and B. M. Lee, "A Survey of Quantum Deep Reinforcement Learning for Resource Allocation in Future Wireless Networks," in IEEE Communications Surveys & Tutorials, vol. 28, pp. 109-148, 2026, doi: 10.1109/COMST.2025.3638435.
3. A. Routh and J. Singh, "Enhancing Explainability of Quantum-Based Stacking for Mental Health Assessment: Predicting Depression Risk in AIDS Patients," in IEEE Access, vol. 13, pp. 206490-206503, 2025, doi: 10.1109/ACCESS.2025.3639110.
4. A. Ahmed et al., "A Hybrid Deep Learning and Quantum Optimization Framework for Ransomware Response in Healthcare," in IEEE Open Journal of the Computer Society, vol. 7, pp. 154-165, 2026, doi: 10.1109/OJCS.2025.3648741.
5. S. Park, J. P. Kim, C. Park, S. Jung, and J. Kim, “Quantum multi-agent reinforcement learning for autonomous mobility cooperation,” IEEE Commun. Mag., vol. 62, no. 6, pp. 106–112, Jun. 2024.
6. F. Zaman, A. Farooq, M. A. Ullah, H. Jung, H. Shin, and M. Z. Win, “Quantum machine intelligence for 6G URLLC,” IEEE Wireless Commun., vol. 30, no. 2, pp. 22–30, Apr. 2023.
7. T. Q. Duong, L. D. Nguyen, B. Narottama, J. A. Ansere, D. V. Huynh, and H. Shin, “Quantum-inspired real-time optimization for 6G networks: Opportunities, challenges, and the road ahead,” IEEE Open J. Commun. Soc., vol. 3, pp. 1347–1359, 2022.

8. I. K. Garewal, C. V. Mahamuni, and S. Jha, “Emerging applications and challenges in quantum computing: A literature survey,” in Proc. Int. Conf. Artif. Intell., Big Data, Comput. Data Commun. Syst. (icABCD), Aug. 2024, pp. 1–12.
9. M. Schuld and N. Killoran, “Quantum machine learning in feature Hilbert spaces,” Phys. Rev. Lett., vol. 122, no. 4, Feb. 2019, Art. no. 040504.
10. J. R. Glick, R. LaRose, Z. Holmes, L. Cincio, and P. J. Coles, “Covariant quantum kernels for data with group structure,” Phys. Rev. Lett., vol. 129, no. 11, 2022, Art. no. 110501.
11. M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, and P. J. Coles, “Variational quantum algorithms,” Nature Rev. Phys., vol. 3, no. 9, pp. 625–644, 2021.
12. B. Krishnan Raghupathy, S. Vairavasundaram, M. Ganesan, R. Kumar Namachivayam, K. Kotecha and N. Herencsar, "Enhanced Quantum Convolutional Neural Network for Signature Authentication in Consumer Products," in IEEE Transactions on Consumer Electronics, vol. 71, no. 1, pp. 2309-2321, Feb. 2025, doi: 10.1109/TCE.2024.3509624.
13. V. Rishiwal, U. Agarwal, M. Yadav, S. Tanwar, D. Garg and M. Guizani, "A New Alliance of Machine Learning and Quantum Computing: Concepts, Attacks, and Challenges in IoT Networks," in IEEE Internet of Things Journal, vol. 12, no. 12, pp. 18865-18886, 15 June 2025, doi: 10.1109/JIOT.2025.3535414.
14. M. B. A. Dastagir, O. Tariq, S. Mumtaz, S. Al-Kuwari and A. Farouk, "Quantum-Inspired Reinforcement Learning for Secure and Sustainable AIoT-Driven Supply Chain Systems," in IEEE Internet of Things Journal, vol. 13, no. 4, pp. 6575-6586, 15 Feb. 2026, doi: 10.1109/JIOT.2025.3637911.
15. B. K. Behera, S. Al-Kuwari and A. Farouk, "QSVM-QNN: Quantum Support Vector Machine Based Quantum Neural Network Learning Algorithm for Brain–Computer Interfacing Systems," in IEEE Transactions on Artificial Intelligence, vol. 7, no. 1, pp. 308-320, Jan. 2026, doi: 10.1109/TAI.2025.3572852.