Published: 24 January 2026

Citation in BibTeX format

ICICA 2025: The 14th International Conference on Information Communication and Applications
September 14 - 16, 2025
Oxford, United Kingdom

RESEARCH-ARTICLE

A Comparative Analysis of Adaptive and Scheduled Dynamic Loss Weighting Strategies in Quantum Physics-Informed Neural Networks (QPINNs) for Solving the 1D TimeIndependent Schrodinger Equation

JOHN EDOH ONUH, University of Sunderland, Sunderland, Tyne and Wear, U.K.

MING JIANG, University of Sunderland, Sunderland, Tyne and Wear, U.K.

Open Access Support provided by:

University of Sunderland

ICICA '25: Proceedings of the 14th International Conference on Information Communication and Applications (September 2025)
https://doi.org/10.1145/3743158.3783855
ISBN: 9798400721151

A Comparative Analysis of Adaptive and Scheduled Dynamic Loss Weighting Strategies in Quantum Physics-Informed Neural Networks (QPINNs) for Solving the 1D TimeIndependent Schrodinger Equation

John Edoh Onuh
School of Computer Science & Engineering
University of Sunderland
Sunderland, United Kingdom
bi51ma@student.sunderland.ac.uk

Ming Jiang
School of Computer Science & Engineering
University of Sunderland
Sunderland, United Kingdom
Ming.Jiang@sunderland.ac.uk

Abstract

In recent years, Quantum Computing has revolutionized the field of physics by enabling researchers to tackle complex problems that were previously unsolvable. The ability to simulate and analyze phenomena at a quantum level has led to significant breakthroughs in our understanding of the universe. In this article, we describe and discuss how quantum computers are being used for cutting-edge research in quantum physics. The time-independent Schrödinger equation is a cornerstone of quantum mechanics, yet its solution for complex systems is a formidable computational challenge. Quantum Physics-Informed Neural Networks (QPINNs) have emerged as a promising paradigm, leveraging Quantum Neural Networks (QNNs) as a wavefunction ansatz within a physics-informed machine learning framework. A critical and often overlooked aspect of training these models is the management of the multi-term loss function, which balances adherence to the governing Partial Differential Equation (PDE) with physical constraints such as normalization. This paper presents a comprehensive implementation and comparative analysis of two distinct Dynamic Loss Weighting (DLW) strategies for training QPINNs to solve the 1D time-independent Schrödinger equation. The first strategy is an adaptive, gradient-based method that dynamically balances loss components based on their real-time impact on model parameters. The second is a pre-scheduled annealing method that follows a curriculum-like approach, prioritizing different physical constraints at different stages of training. We apply these methods to two benchmark systems: the Quantum Harmonic Oscillator (QHO) and the Finite Square Well. Our results demonstrate that the adaptive, gradient-based DLW, when paired with a physic informed wavefunction ansatz for the QHO, achieves remarkable accuracy, converging to the exact ground state energy of $E = 0.5$. Conversely, the scheduled annealing strategy applied to Square Well, using a more generic ansatz, converges to a plausible energy but with a higher residual loss. This comparative analysis reveals a crucial insight: the effectiveness of a DLW strategy is deeply intertwined with the physical-informedness of the underlying wavefunction ansatz. This suggests that a synergistic co-design philosophy, which considers the interplay between the quantum model's architecture and the adaptive training algorithm, is essential for developing robust and accurate QPINN-based solvers for quantum systems.

ICICA 2025, Oxford, United Kingdom
© 2025 Copyright held by the owner/author(s).
ACM ISBN /2025/09
https://doi.org/10.1145/3743158.3783855

CCS Concepts

• **Machine Learning**; • **Emerging Technologies**; • **Models of Computation**;

Keywords

Quantum Physics, Quantum Computing, Physics-Informed Neural Networks

ACM Reference Format:
John Edoh Onuh and Ming Jiang. 2025. A Comparative Analysis of Adaptive and Scheduled Dynamic Loss Weighting Strategies in Quantum Physics-Informed Neural Networks (QPINNs) for Solving the 1D TimeIndependent Schrodinger Equation. In *The 14th International Conference on Information Communication and Applications (ICICA 2025), September 14–16, 2025, Oxford, United Kingdom.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3743158.3783855

1 Introduction

1.1 The Schrödinger Equation: A Cornerstone with Computational Hurdles

The Schrödinger equation stands as a fundamental postulate in non-relativistic quantum mechanics, providing a mathematical description of how the quantum state of a physical system evolves [1]. Despite its central importance, obtaining analytical solutions to this equation is feasible only for a limited set of simple systems, such as the hydrogen atom or the quantum harmonic oscillator. For most systems encountered in physics and chemistry, particularly those involving multiple interacting particles like multi-electron atoms, molecules, and condensed matter systems, the complexity of interaction potentials renders analytical solutions generally impossible. The introduction of electron-electron interaction terms, for instance, makes the partial differential equation (PDE) non-separable, a primary mathematical hurdle that prevents exact analytical treatment.

This intractability necessitates a reliance on numerical approximation methods. However, even these approaches face a formidable obstacle known as the ”exponential wall” or the ”curse of dimensionality” [4]. The Hilbert space, which encompasses all possible states of a quantum system, grows exponentially with the number of constituent particles. To illustrate, describing the spin state of N electrons requires 2^N complex coefficients, while discretizing space into M grid points for each of N particles means the number of coefficients needed to specify the wavefunction scales as M^N. This exponential scaling ensures that the computational resources (memory and processing time) required for a direct numerical solution become prohibitive for all but the smallest systems. Traditional numerical techniques, such as the Finite Difference Method (FDM) or Finite Element Method (FEM), are directly confronted by this scaling problem. This limitation underscores that the intractability of the Schrödinger equation for many-body systems is not merely a question of insufficient computational power; it is a fundamental mathematical challenge rooted in non-separability and the exponential scaling of the solution space, compelling the exploration of new paradigms.

1.2 Foundations of Physics-Informed Neural Networks

The fundamental principle of PINNs is to approximate the solutions of PDEs using neural networks that are trained by minimizing a loss function incorporating the governing physical laws. This composite loss function typically includes terms penalizing the PDE residual (how much the network's output violates the PDE), deviations from specified boundary and initial conditions, and, if available, mismatches with observational data. A key enabling technology for PINNs is automatic differentiation [14, 23], which is used to compute the derivatives of the network's output with respect to its input coordinates as required by the PDE terms.

PINNs offer several advantages, including being mesh-free and capable of handling complex geometries. However, their training presents notable challenges. The loss landscapes are often highly complex and non-convex, and models can suffer from "gradient pathologies," where imbalances between different loss terms or vanishing/exploding gradients hinder effective optimization. This multi-objective nature can lead to "stiffness," where different components of the loss landscape have vastly different scales, making it difficult for optimizers to find a good minimum. These challenges have spurred the development of numerous variants and advanced training strategies, including the dynamic weighting schemes that are the focus of this work.

1.3 The Critical Role of Dynamic Loss Weighting (DLW) in Training PINNs

A pivotal challenge in training any PINN, whether classical or quantum, is the management of its composite loss function. To solve the Schrödinger equation, the loss function must enforce multiple physical constraints simultaneously: minimizing the PDE residual, ensuring the wavefunction is normalized, and satisfying boundary conditions. This creates a multi-objective optimization problem where different loss components can have conflicting gradients or operate at vastly different numerical scales. Using static, fixed weights for these loss terms often requires extensive manual tuning and can lead to training instability, where one term dominates the optimization process at the expense of others. This complex interplay motivates the use of Dynamic Loss Weighting (DLW) strategies, which adaptively adjust the weights of loss components during training to guide the optimizer toward a physically valid solution more smoothly and reliably.

1.4 The Quantum Ansatz: A Case for Quantum Physics-Informed Neural Networks (QPINNs)

Concurrently, the field of Quantum Machine Learning (QML) has emerged, aiming to leverage the principles of quantum computation to enhance machine learning algorithms. The synergy between these fields has given rise to Quantum Physics-Informed Neural Networks (QPINNs) [6, 10, 16]. In a QPINN framework, a Quantum Neural Network (QNN), also known as a Parameterized Quantum Circuit (PQC) [17, 24], is employed to learn and represent the solution to a differential equation, such as the wavefunction in the Schrödinger equation.

The central hypothesis is that QNNs might offer a more "natural" or parametrically efficient ansatz for representing quantum wavefunctions compared to classical neural networks. This could translate into improved accuracy or a superior ability to capture inherently quantum phenomena like entanglement and superposition, which are native to the QNN's operational principles. The motivation extends beyond merely finding a numerical solution; it aims to find a quantum-native solution, where the quantum nature of the ansatz aligns with the quantum nature of the problem, guided and constrained by the governing physical laws.

1.5 Contributions and Structure of This Paper

This paper makes several key contributions to the burgeoning field of QPINNs. First, we detail the implementation of a flexible QPINN framework for solving the 1D time-independent Schrödinger equation, built upon the hybrid quantum-classical computing capabilities of PennyLane [14, 24] and PyTorch [23]. Second, we conduct a practical, comparative study of two major classes of DLW: an adaptive, gradient-based strategy that reacts to the instantaneous training dynamics, and a pre-scheduled, annealing-based strategy that follows a pre-defined curriculum. Third, through a detailed analysis of results for two benchmark systems—the Quantum Harmonic Oscillator and the Finite Square Well—we reveal a deep and crucial interplay between the design of the wavefunction ansatz and the performance of the DLW strategy. Our findings suggest that a holistic co-design principle is essential for the future development of robust QPINN solvers.

The remainder of this paper is structured as follows. Section 2 provides a review of related work on PINNs, QNNs, and DLW techniques. Section 3 details our proposed methodology, including the QPINN formulation, the quantum wavefunction ansatz, and the implementation of the two DLW strategies. Section 4 presents a thorough analysis of our experimental results. Section 5 discusses the broader implications of our findings, situates them within the context of key challenges in the field, and outlines directions for future work. Finally, Section 6 provides concluding remarks.

2 Related Work

2.1 Quantum Neural Networks as Differentiable Function Approximators

Quantum Neural Networks (QNNs), or Parameterized Quantum Circuits (PQCs), are quantum circuits where some of the constituent quantum gates are parameterized by classical variables, analogous to the weights and biases in classical neural networks. The structure of a typical QNN involves three key stages:

1. Data Encoding (Embedding): Classical input data, such as a spatial coordinate x, must be encoded into the quantum state of the QNN. Common strategies include angle encoding, where input values determine the rotation angles of specific gates.
2. Variational Circuit (Ansatz): This is the core of the QNN, consisting of layers of quantum gates with tunable parameters. These layers typically include single-qubit rotations and multi-qubit entangling gates (e.g., CNOTs), which together define the "ansatz" of the QNN.
3. Measurement: After the circuit has processed the encoded input, measurements are performed to extract classical information. This usually involves measuring the expectation value of one or more quantum observables (e.g., Pauli operators) on specific qubits, which form the output of the QNN.

Training QNNs typically involves a hybrid quantum-classical optimization loop. The quantum circuit is executed, the measurement outcomes are used to compute a classical loss function, and a classical optimization algorithm updates the circuit parameters. Gradients of the QNN output with respect to its parameters can be computed using techniques like the parameter-shift rule, which enables the use of standard gradient-based optimizers. While QNNs possess the theoretical capacity to approximate complex functions by navigating high-dimensional Hilbert spaces, their practical application is hindered by significant challenges, most notably the "barren plateau" phenomenon [21], where gradients can vanish exponentially as the number of qubits or circuit depth increases, making optimization extremely difficult.

2.2 Dynamic Loss Weighting Strategies for Multi-Term Loss Functions

The multi-objective optimization problem posed by the PINN loss function has motivated the development of a wide array of Dynamic Loss Weighting (DLW) techniques. These strategies aim to adaptively adjust the weights of different loss components during training to improve stability, accelerate convergence, and achieve better overall performance. They can be broadly categorized as follows:

- Annealing-Based Schedules: Inspired by curriculum learning, these methods vary the loss weights according to a pre-defined schedule as a function of the training epoch. This allows the network to first learn the basic form of the solution before being subjected to stricter constraints. While simple to implement, the schedule itself is heuristic and not responsive to the live training dynamics.
- Gradient-Based Balancing: These methods adjust weights based on the statistics of the gradients of individual loss terms. The core idea is to ensure all tasks contribute meaningfully to learning by balancing the magnitudes of their gradients. Prominent examples include GradNorm [15], which aims to equalize gradient norms, and more recent, sophisticated methods like Dual-Balanced PINN (DB-PINN) [7] and SOAP [13], which address both gradient magnitude and directional conflicts. These approaches are more adaptive but can add complexity and computational overhead.
- Uncertainty-Based Weighting: These strategies weight loss terms based on some measure of the model's uncertainty associated with them. For instance, a loss term corresponding to a constraint where the model is highly uncertain might be down-weighted. This provides a principled way to handle varying levels of confidence but requires a mechanism to quantify uncertainty, which can be computationally expensive.
- Learning-Based Approaches: This category includes methods that use other forms of learning to determine loss weights, such as soft attention mechanisms that can adjust weights for each individual sampling point (SA-PINN) or even dedicated neural networks that learn to adjust the weights of their respective loss components (LA-PINN).

The two methods implemented in this paper are representative of the two most foundational categories: the pre-defined "Annealing" approach and the reactive "Gradient-Based" approach. This allows for a direct empirical comparison of these distinct philosophies for managing the multi-objective loss in a QPINN context.

3 Proposed Methods

3.1 QPINN Formulation for the Time-Independent Schrödinger Equation

The core of our investigation is the 1D time-independent Schrödinger equation, which, in natural units (m= 1, h= 1), is given by:

$$-\frac{1}{2}\frac{d^2\psi(x)}{dx^2} + V(x)\psi(x) = E\psi(x) \tag{1}$$

Here, ψ(x) is the complex-valued wavefunction, V(x) is the potential energy, and E is the energy eigenvalue, which we treat as a trainable parameter in our model.

To train the QPINN, we formulate a physics-informed loss function, Ltotal, composed of two primary components. The first is the PDE residual loss, LPDE, which measures the extent to which the network's output violates the Schrödinger equation. It is defined as the mean squared error of the PDE residual over a set of collocation points {xi} sampled from the problem domain:

$$L_{PDE} = \langle| -\frac{1}{2}\frac{d^2\psi(x)}{dx^2} + V(x)\psi(x) - E\psi(x)\ |^2\rangle_x \tag{2}$$

The second component is the normalization loss, Lnorm, which enforces the physical constraint that the total probability of finding the particle must be unity. This is formulated as the squared error from the ideal normalization condition:

$$L_{norm} = \left(\int_{domain} |\psi(x)|^2 dx - 1.0\right)^2 \tag{3}$$

The integral is approximated numerically using the trapezoidal rule over the collocation points. The total loss function is then a dynamically weighted sum of these two components:

$$L_{total} = w_{PDE} L_{PDE} + w_{norm} L_{norm} \tag{4}$$

where w_{PDE} and w_{norm} are the dynamic weights determined by the specific DLW strategy being employed.

3.2 The Quantum Wavefunction Ansatz

The wavefunction $\psi(x)$ is represented by a hybrid quantum-classical ansatz. This construction involves a quantum core that generates a base complex function, which is then shaped by a classical envelope function to incorporate physical inductive biases.

3.2.1 The Quantum Core (PQC). The quantum component of our model is a Parameterized Quantum Circuit (PQC) [17, 24] implemented using PennyLane [14, 24].

- Architecture: The circuit operates on a 4-qubit register, simulated using PennyLane's default.qubit device [14, 24].
- Input Encoding: The classical input coordinate x is encoded into the initial state of the qubits using qml.AngleEmbedding [24]. This operation applies a rotation around the Y-axis to each qubit, with the rotation angle being a function of x. This makes the subsequent quantum state dependent on the spatial coordinate.
- Variational Layers: The core of the ansatz is a sequence of 8 qml.StronglyEntanglingLayers [24]. Each layer consists of single-qubit rotations around the X, Y, and Z axes, followed by a cascade of CNOT gates that entangle neighboring qubits. This deep, highly entangling structure provides the QNN with significant expressive power. The trainable parameters of the model, analogous to weights in a classical network, are the parameters of these rotation gates, with a total shape of (8 layers, 4 qubits, 3 angles per rotation).
- Measurement: To produce a classical, complex-valued output, we measure the expectation value of the Pauli Z operator on the first two qubits. The output of the first qubit is interpreted as the real part of a base function, and the output of the second qubit is interpreted as its imaginary part.

3.2.2 The Full Wavefunction Ansatz. The raw output from the QNN is transformed into an intermediate complex function. To ensure the real part of the function is centered around a non-zero value, we shift the real component from the QNN, which has an expectation value in [-1, 1], to be in:

$$\phi(x) = (1.0 + \boldsymbol{Re}[\boldsymbol{QNN}(\boldsymbol{x})]) + \boldsymbol{i}.\boldsymbol{Im}[\boldsymbol{QNN}(\boldsymbol{x})] \tag{5}$$

where $Re[QNN(x)]$ and $Im[QNN(x)]$ are the expectation values from the first and second qubits, respectively.
A crucial element of our methodology is the application of a classical envelope function to this base quantum function. The complete wavefunction ansatz is constructed as:

$$\boldsymbol{\psi}(\boldsymbol{x}) = \phi(\boldsymbol{x}) \cdot \boldsymbol{e}^{-x^2/\sigma_{gauss}^2} \tag{6}$$

This Gaussian envelope serves as a strong inductive bias. By its nature, it ensures that the wavefunction smoothly approaches zero as $x \rightarrow \pm\infty$. This architecturally enforces the boundary conditions for bound state problems, eliminating the need for an explicit boundary condition loss term, LBC, in our total loss function. The parameter σ_{gauss} controls the width of this envelope and is a key hyperparameter that can be tuned to match the expected spatial extent of the ground state wavefunction for a given potential. This co-design of a quantum core with a classical, physics-informed structure is a central aspect of our proposed method.

3.3 Implementation of Dynamic Loss Weighting Strategies

We implement and compare two distinct DLW strategies, representing two different philosophies of loss balancing.

3.3.1 Strategy 1: Adaptive Gradient-Based Balancing. This strategy is a reactive approach that dynamically adjusts loss weights based on the real-time gradient information during training. It is inspired by the GradNorm [7] family of algorithms and aims to balance the learning rates of different tasks.

Strategy 2: Scheduled Annealing
This strategy is a proactive, curriculum-based approach where the weights follow a pre-defined, non-adaptive schedule. The goal is to guide the learning process through distinct phases. Such scheduled, curriculum-like methods are a common heuristic in machine learning [2, 8].

4 Experiments and Data Analysis

4.1 Experimental Setup: Benchmark Potentials and Hyperparameters

To evaluate and compare the two DLW strategies, we apply our QPINN framework to two canonical 1D quantum mechanics problems: the Quantum Harmonic Oscillator (QHO) and the Finite Square Well [4, 17]. The QHO, with its smooth potential $V(x) = 0.5x^2$, provides an ideal testbed for validation as its ground state has a well-known analytical solution [5, 17]. The Finite Square Well, with its sharp, discontinuous potential, presents a different type of challenge for the neural network approximator [6, 17]. The specific hyperparameters for each experiment were empirically chosen to yield the best performance for the selected DLW strategy and are detailed in [7, 17]. Both experiments utilize the Adam optimizer and a ReduceLROnPlateau learning rate scheduler [7, 17]. A fixed random seed was used for reproducibility [8, 17].

4.1.1 QPINN Model Architecture. A single, consistent Quantum Neural Network (QNN) architecture serves as the core for both case studies. The architecture, implemented in PennyLane [8, 11], is detailed below and visualized in Figure 1.

- Qubits: The circuit operates on a 4-qubit register [9, 11].
- Input Encoding: A classical input coordinate x is encoded using qml.AngleEmbedding, which applies a rotation to each qubit based on the input value .
- Variational Layers: The heart of the network consists of 8 layers of qml.StronglyEntanglingLayers [3, 12]. This provides a highly expressive ansatz with trainable parameters for the rotation gates within each layer [5, 12].
- Measurement: The real and imaginary components of the base wavefunction are derived from the expectation values

Table 1: Hyperparameter Configuration for Experimental Cases

Hyperparameter	Harmonic Oscillator	Square Well
Potential Function	$V(x) = 0.5x^2$	$V(x) = 0\ for\ \|x\| < 2$, 10 otherwise
Domain	[-7, 7]	[-5, 5]
Collocation Points	120	100
Max Epochs	1500	600
Optimizer	Adam	Adam
Initial Learning Rate	0.001	0.001
LR Scheduler	ReduceLROnPlateau (patience=50)	ReduceLROnPlateau (patience=50)
DLW Strategy	gradient_based	annealing
Ansatz sigma_gauss	$\sqrt{2.0} \approx 1.414$	2.5
DLW: grad_norm_beta	0.9	N/A
DLW: w_pde_anneal	N/A	1.0
DLW: w_norm_initial_anneal	N/A	1.0
DLW: w_norm_final_anneal	N/A	20.0
DLW: delay_fraction_anneal	N/A	0.1
DLW: ramp_duration_fraction_anneal	N/A	0.2

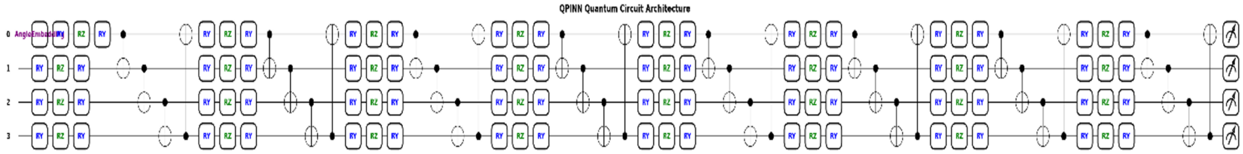

Figure 1: Quantum circuit architecture of the QNN used as the wavefunction ansatz.

of the Pauli Z operator on the first two qubits, respectively [7, 12].

The circuit takes a single classical input (x), encodes it via AngleEmbedding, processes it through 8 StronglyEntanglingLayers with trainable weights (θ), and measures the expectation values of PauliZ on the first two qubits to produce the real and imaginary components of the output.

4.2 Case Study 1: The Quantum Harmonic Oscillator with Gradient-Based DLW

4.2.1 Energy Convergence and Accuracy. The primary goal for the QHO is to find the ground state energy. The training successfully converged to a final learned energy of $E = 0.5$ [4, 18]. This result is in perfect agreement with the well-known analytical ground state energy for this potential, $E_0 = 0.5$, demonstrating the capability of the QPINN framework to achieve extremely high accuracy on this problem [5, 18]. The model with the lowest total loss was found at the final epoch, 1499 [6, 18].

4.2.2 Analysis of Loss Dynamics and Weight Adaptation. An analysis of the training process reveals a fascinating dynamic between the loss components and the adaptive weights, shown in Figure 2. The training logs show that after an initial period of fluctuation, the normalization loss weight, w_{norm}, is driven rapidly towards zero by the gradient-based DLW algorithm, while the PDE loss weight, w_{PDE}, is maintained at its maximum value of 2.0 [9, 18]. Consequently, the total loss becomes almost entirely dominated by the PDE residual.

This behavior is not a failure of the DLW but rather a sign of its effectiveness. The key lies in the choice of the Gaussian envelope [22] hyperparameter, $\sigma_{gauss} = \sqrt{2.0}$. For the QHO ground state, the analytical probability density is a Gaussian, $|\psi(x)|^2 \propto e^{-x^2}$. The normalization integral of our ansatz's envelope, $\int e^{-2x^2/\sigma_{gauss}^2} dx$, becomes $\int e^{-x^2} dx$ dx $when\ \sigma_{gauss} = \sqrt{2.0}$. This means our ansatz was constructed with a classical envelope that is already extremely close to the correct, normalized shape of the true solution. The adaptive DLW algorithm correctly identifies this. It senses that the raw normalization loss, L_{norm}, is already small and that its gradients are negligible compared to the gradients of the PDE loss, L_{PDE}. To find the most efficient path to a global minimum, the optimizer determines that forcing the already-near-perfect normalization would introduce conflicting gradients and hinder the minimization of the more substantial PDE residual. Therefore, it intelligently learns to effectively ”turn off” the normalization constraint and focus all its capacity on satisfying the Schrödinger equation itself. This demonstrates a powerful synergy: a physically-informed ansatz combined with an adaptive DLW creates an efficient and accurate learning path [9, 19].

4.2.3 Comparison of Learned and Analytical Wavefunctions. The quality of the solution is further confirmed by visual inspection of the learned wavefunction. The learned probability density, $|\psi(x)^2|$, shows an excellent match with the analytical Gaussian probability

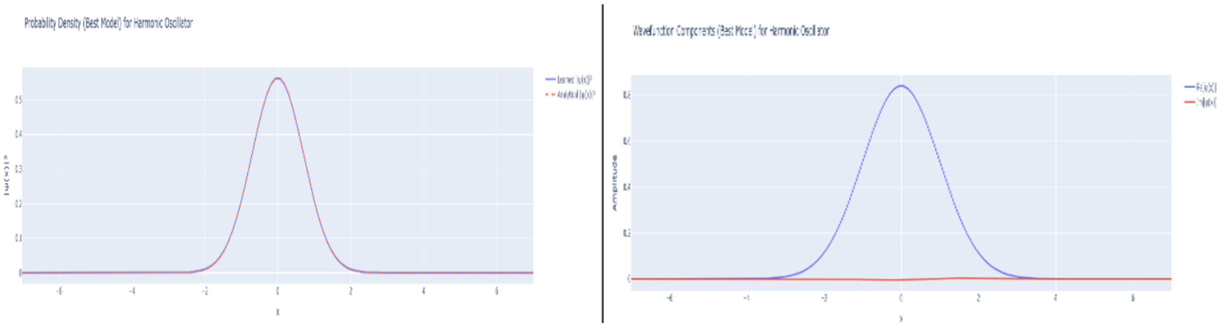

Figure 2: Comparison of the learned and analytical probability densities for the Harmonic Oscillator ground state (Feft) and the real and imaginary components of the learned wavefunction for the Harmonic Oscillator (Right).

density (figure 2) [2, 20]. Similarly, the real and imaginary parts of the learned wavefunction, $\psi(x)$, align perfectly with the shape of the analytical solution (a purely real Gaussian function), with the imaginary part correctly converging to zero across the entire domain (figure 2) [3, 20]. This confirms that the QPINN has not just found the correct energy but has also learned a highly accurate representation of the true quantum state.

4.3 Case Study 2: The Finite Square Well with Annealing-Based DLW

4.3.1 Energy Convergence. For the Finite Square Well potential, the QPINN trained with the scheduled annealing DLW converged to a final ground state energy of $E = 0.3760$ [7, 20]. Unlike the QHO, a simple closed-form analytical solution for this potential's energy levels does not exist, but the result is physically plausible, representing a low-lying bound state energy within the well [8, 20]. The training converged to its best state at the final epoch, 599, with a final total loss of 0.0386 [9, 20].

4.3.2 Analysis of Scheduled Weight Adaptation. The loss dynamics for this case clearly reflect the pre-defined annealing schedule. For the first 10% of epochs (up to epoch 60), $w_{norm}$is held at 1.0. It then ramps up linearly over the next 20% of epochs (up to epoch 180), reaching its final value of 20.0, where it remains for the rest of the training. The training logs show that the raw normalization loss drops to near zero very quickly, and the total loss becomes dominated by the PDE residual.

However, the final total loss (0.0386) and PDE loss are significantly higher than in the QHO case (which was effectively zero). This suggests a less optimal solution. The underlying reason appears to be a mismatch between the ansatz and the DLW strategy. The ground state wavefunction of a finite square well is cosine-like inside the well and has exponential tails outside. The Gaussian envelope [22], $\psi(x)\ \alpha\ e^{-x^2/\sigma_{gauss}^2}$, is a poorer structural approximation for this shape compared to its perfect fit for the QHO. The rigid, pre-scheduled DLW is not responsive to this mismatch. It proceeds to aggressively ramp up the normalization weight to 20.0 according to its fixed curriculum. This creates a strong three-way tension in the optimization: the optimizer must simultaneously (1) satisfy the PDE, (2) satisfy the strong normalization constraint imposed by $w_{norm} = 20$, and (3) work within the structural confines of the ill-suited Gaussian envelope [22]. This conflict prevents the model from finding a truly low-loss solution, resulting in a compromise that has a higher residual error.

4.3.3 Learned Wavefunction and Physical Plausibility. The plot of the learned probability density, $|\psi(x)^2|$, shows a function that is correctly concentrated within the potential well (between $x = -2\ and\ x = 2$) and decays outside of it, which is physically correct for a bound state (Figure 3). The shape is bell-like, consistent with the Gaussian envelope used [22]. While physically plausible, it does not capture the flatter top and sharper shoulders of the true cosine-like solution, a direct consequence of the structural bias imposed by the ansatz.

4.4 Summary of Experimental Findings

The quantitative outcomes of the two case studies are summarized in Table 2. The stark difference in final loss values and the perfect accuracy achieved for the QHO highlight the profound impact of the interplay between ansatz design and DLW strategy.
Figure 4 provides a compelling visual summary of our findings:

- Loss Comparison: The Harmonic Oscillator (HO) case, using the adaptive DLW and a well-suited ansatz, drives its total loss and PDE loss to virtually zero. The Square Well (SW) case, with its scheduled DLW and mismatched ansatz, sees its loss plateau at a much higher value.
- Dynamic Weights: The difference between the DLW philosophies is clear. The HO's adaptive weights react to the training state, down-weighting the easy normalization task. The SW's weights follow a rigid, pre-defined schedule, ramping up the normalization weight regardless of the underlying optimization landscape.
- Energy Convergence: Both models converge to stable energy values. However, the HO model's convergence to the exact analytical energy of 0.5000 is a direct result of its ability to find a near-perfect solution with minimal loss. The SW model finds a plausible, but less accurate, energy as a compromise, constrained by the higher residual loss.

This direct comparison reinforces the central thesis that the effectiveness of a DLW strategy is deeply intertwined with the physical-informed underlying wavefunction ansatz [13].

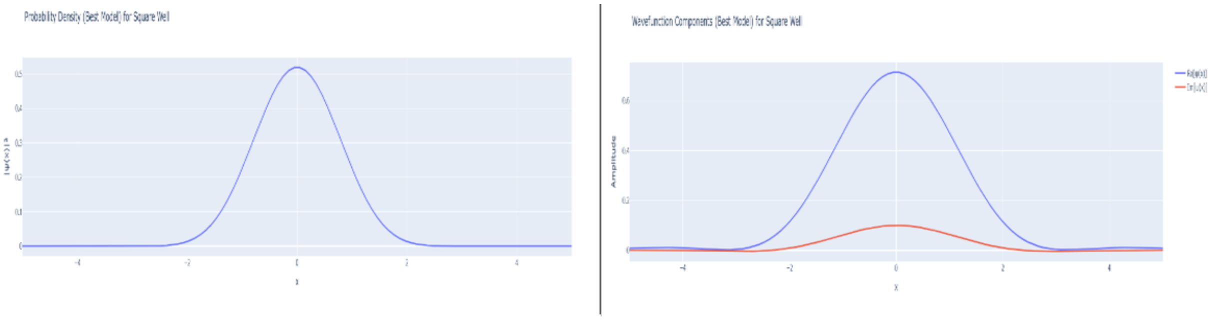

Figure 3: The learned probability density for the Finite Square Well (left) and the real and imaginary components of the learned wavefunction for the Square Well (Right).

Table 2: Summary of Final Experimental Results

Metric	Harmonic Oscillator	Square Well
DLW Strategy	gradient_based (Adaptive)	annealing (Scheduled)
Final Learned Energy	0.5000	0.3760
Analytical/Target Energy	0.5	N/A
% Error	0.0%	N/A
Final Total Loss	0.0000	0.0386
Final PDE Loss (raw)	0.0000	0.0386
Final Norm Loss (raw)	0.0654	0.0000

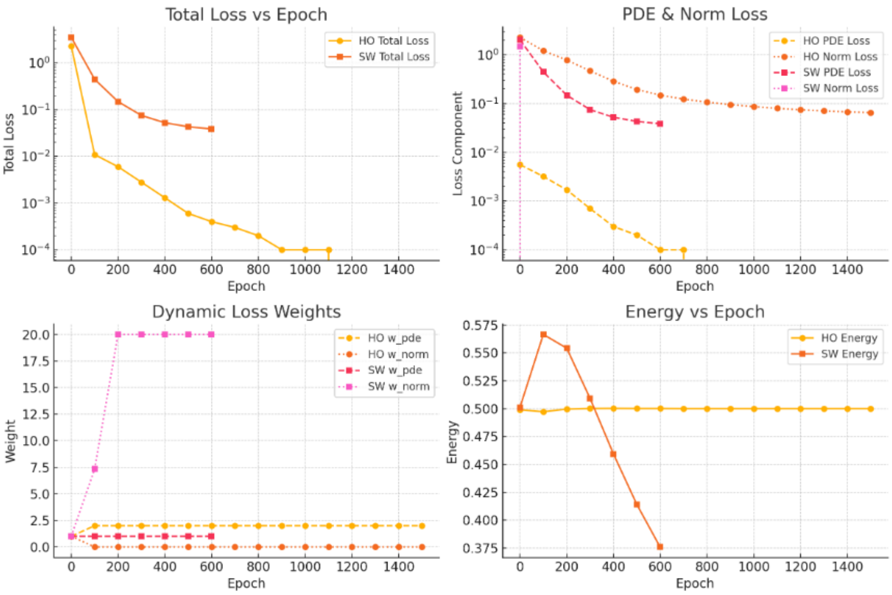

Figure 4: Comparative analysis of training dynamics. (Top Left) Total loss vs. epoch. (Top Right) Individual raw loss components. (Bottom Left) Dynamic loss weights. (Bottom Right) Energy vs. epoch. The plots clearly show the QHO case achieving near-zero loss, while the SW case stalls at a higher loss. The DLW plot highlights the difference between the adaptive (HO) and scheduled (SW) strategies.

5 Discussions and Future Work

The experimental results provide a clear comparative analysis of the two DLW strategies. On the surface, the adaptive, gradient-based method used for the QHO appears superior, driving the loss to zero and achieving the exact analytical energy. However, a deeper analysis reveals a more nuanced conclusion. The success of the adaptive DLW was not solely due to the algorithm's intrinsic properties but was critically enabled by the choice of a highly effective, physically-informed ansatz. The Gaussian envelope [22] was so well-suited to the QHO problem that the adaptive DLW had the flexibility to recognize this and dynamically simplify the optimization problem by down-weighting the nearly-satisfied normalization constraint.

In contrast, the scheduled annealing strategy for the Square Well was paired with a less suitable ansatz. The rigid, pre-defined curriculum of the annealing schedule could not adapt to this mismatch. It inflexibly enforced a strong normalization penalty, creating a conflict with the structural bias of the ansatz and the demands of the PDE, ultimately leading to a higher-loss, compromised solution.

This leads to the central thesis of this work: the co-design principle. The optimal training of a QPINN does not arise from selecting the best QNN or the best DLW strategy in isolation. Rather, it emerges from the synergistic co-design of a quantum ansatz that incorporates as much physical knowledge as possible and a DLW strategy that is adaptive and flexible enough to understand and exploit the strengths of that ansatz. A powerful adaptive DLW can find an efficient optimization path if the ansatz provides a good starting point, while a rigid DLW can hinder performance by enforcing constraints that conflict with a mismatched ansatz. Based on the findings and discussion, several concrete directions for future research emerge in testing DLW flexibility to better manage the ansatz-potential mismatch and find a lower-loss solution compared to the rigid annealing schedule; incorporating Noise Models by using a noisy quantum simulator or, eventually, on actual quantum hardware [2].

6 Conclusion

This paper has presented a detailed implementation and comparative analysis of a Quantum Physics-Informed Neural Network framework for solving the 1D time-independent Schrödinger equation, with a specific focus on the role of Dynamic Loss Weighting. By applying two distinct DLW strategies, one adaptive and gradient-based, the other pre-scheduled and curriculum-based—to the Quantum Harmonic Oscillator and Finite Square Well problems, we have uncovered a critical relationship between the model's architecture and its training algorithm.

Our key findings are that the success of a QPINN hinges on a holistic co-design principle. The remarkable accuracy achieved for the QHO was the result of a powerful synergy between a physics informed wavefunction ansatz and an adaptive DLW strategy that was flexible enough to exploit the ansatz's strengths. Conversely, the challenges encountered with the Square Well highlighted how a rigid DLW schedule can conflict with a mismatched ansatz, leading to a compromised solution. This demonstrates that the choice of DLW is not a mere hyperparameter but a fundamental component of the model design that must be considered in concert with the physical properties of the quantum ansatz.

While the field faces significant hurdles on the path to practical quantum advantage—most notably barren plateaus, scalability, and hardware noise—this work underscores the promise of the QPINN paradigm. By continuing to develop more sophisticated, problem-inspired quantum ansatzes and more intelligent, adaptive training strategies, QPINNs could become a powerful new tool in the computational scientist's arsenal. The potential to contribute to scientific discovery in quantum chemistry, materials science, and condensed matter physics by providing a new way to solve the fundamental equations of the quantum world remains a powerful motivator for continued research in this exciting and rapidly evolving field. The simulation-based approach to QPINNs holds great promise for approximating wave functions in complex systems, such as those encountered in quantum critical phenomena. By addressing the challenges and advancing this field, researchers can unlock new insights into the behavior of these systems.

References

[1] Yuanqing Wang, Weitang Liu, and Maziar Raissi. 2023. SchrödingerNet: Solving the Full Electronic-Nuclear Schrödinger Equation by Equalizing Local Energies. arXiv preprint arXiv:2310.20510. https://doi.org/10.48550/arXiv.2310.20510
[2] Samson Wang, Piotr Czarnik, Andrew Arrasmith, M. Cerezo, Lukasz Cincio, Patrick J. Coles, arXiv 2021 Can error mitigation improve trainability of noisy variational quantum algorithms?
[3] Yangfei Liao, Zhongshuo Lin, Jianghao Liu, Qingyuan Sun, Yifan Wang, Teng Wu, Hehu Xie, Mingfeng He, arXiv (2022) Solving Schrödinger Equation Using Tensor Neural Network
[4] Zheyuan Hu, Khemraj Shukla, George Em Karniadakis, Kenji Kawaguchi, arXiv. (2023). Tackling the Curse of Dimensionality with Physics-Informed Neural Networks
[5] Alan Williams, Christopher Leon, Alexander Scheinker, arXiv. (2024). QP Based Constrained Optimization for Reliable PINN Training
[6] Rui Zhang, Liang Li, Stéphane Lanteri, Hao Kang, Jiaqi Li, arXiv. (2025). BO-SA-PINNs: A Bayesian Optimization-Enhanced Self-Adaptive Physics-Informed Neural Network Framework for Solving PDEs.
[7] Chenhong Zhou, Jie Chen, Zaifeng Yang, Ching Eng Png, arXiv. (2025). DB-PINN: Dual-Balanced Physics-Informed Neural Network for Mitigating Gradient and Fitting Imbalances in PDE Solutions.
[8] Kamran Majid, arXiv. (2025). Deep Neural Network Emulation of the Quantum-Classical Transition via Learned Wigner Function Dynamics
[9] Edgar Torres, Jonathan Schiefer, Mathias Niepert, arXiv. (2025). Adaptive Physics-informed Neural Networks: A Survey
[10] Trahan, C., Loveland, M. and Dent, S. Entropy 2024, 26(8), 649; https://doi.org/10.3390/e26080649
[11] Soumyadip Sarkar, arXiv. (2025). Physics-Informed Neural Networks for One-Dimensional Quantum Well Problems
[12] Ding-Dang Yang, arXiv. (2025). Simulating nonlinearity in quantum neural networks while mitigating barren plateaus
[13] Sifan Wang, Ananyae Kumar Bhartari, Bowen Li, Paris Perdikaris, arXiv. (2025) Gradient Alignment in Physics-informed Neural Networks: A Second-Order Optimization Perspective
[14] Bergholm, V., *et al.* (n.d.). PennyLane: Automatic differentiation of hybrid quantum-classical computations. Science Cast.
[15] Chen, Z., *et al.* (2017). GradNorm: Gradient normalization for adaptive loss balancing in deep multitask networks. OpenReview.
[16] Farea, A., Khan, S. and Celebi, M.S. (2025). QCPINN: Quantum-Classical Physics-Informed Neural Networks for Solving PDEs.
[17] Fiveable. (n.d.). Parameterized Quantum Circuits (PQC). Available at: https://fiveable.me [Accessed 24 Jun. 2025].
[18] Ganga, S. and Uddin, Z. (2024). Exploring physics-informed neural networks: From fundamentals to applications in complex systems.
[19] GeeksforGeeks. (n.d.). Graphs, automatic differentiation and autograd in PyTorch. Available at: https://geeksforgeeks.org [Accessed 24 Jun. 2025].
[20] Ingenii. (n.d.). Quantum machine learning fundamentals. Available at: https://ingenii.io [Accessed 24 Jun. 2025]
[21] McClean, J.R., *et al.* (2018). Barren plateaus in quantum neural network training landscapes.
[22] Panichi, G., Corli, S. and Prati, E. (2025). Quantum physics-informed neural networks for multi-variable partial differential equations.
[23] Paszke, A., *et al.* (n.d.). Automatic differentiation in PyTorch. OpenReview.

[24] PennyLane Documentation. (n.d.). qml.StronglyEntanglingLayers. Available at: https://docs.pennylane.ai [Accessed 24 Jun. 2025].