

Enhancing Physics-Informed Neural Networks for the Nonlinear Schrödinger Equation via Residual-Based Refinement and Conservation Laws

Damir Kim & Rozalina Alkeyeva

Supervisor: Amin Esfahani

Second Reader: Adilbek Kairzhan

Department of Mathematics

Nazarbayev University

Abstract

Physics-informed neural networks (PINNs) offer a powerful alternative for solving partial differential equations, yet their efficacy is highly sensitive to collocation point distribution and the formulation of the loss function. This research evaluates the performance of the PINNsTF2-DeepXDE framework against enhanced configurations to solve the focusing cubic Nonlinear Schrödinger Equation (NLSE). Specifically, it investigates three methods deployed to counteract limitations of original PINNs, namely, Residual-based Adaptive Refinement (RAR), an L_2 energy conservation constraint, and the weighting of the PDE residual loss (w_f). Across all tested configurations, RAR emerged as the most effective catalyst of error reduction. The model that employed RAR with $w_f = 4$ achieved the best overall accuracy, with relative ℓ_2 error for the solution h at 7.566×10^{-4} and the lowest validation errors for both components u and v . The L_2 conservation law showed a mixed effect on aggregate error, but integrated physically consistent solutions. Results also show that no single w_f value is universally optimal, as the best weight depends on whether refinement and/or conservation constraints are active. The hybrid configuration (RAR + L_2) provides a practical compromise between numerical accuracy and physical guidance, though it does not surpass the best RAR-only setting in this study (relative ℓ_2 error for h reaching 8.689×10^{-4}). Overall, the findings indicate that robust PINN performance for nonlinear wave dynamics requires coordinated tuning of adaptive collocation and loss composition.

April 21, 2026

Contents

1	Introduction	1
2	Problem Formulation and Exact Solution	2
2.1	Non-linear Schrödinger Equation	3
2.2	Exact & Analytical Solution	3
3	Physics-Informed Neural Networks	4
3.1	PINN Formulation	4
3.2	Training Procedure	6
3.3	Configuration	6
3.4	Collocation Points	7
3.5	Error Metrics	8
4	Baseline	8
4.1	Numerical Results	9
5	Residual-Based Adaptive Refinement	11
5.1	Concept and Theory	11
5.2	Numerical Results	12
6	Conservation Laws in PINNs	13
6.1	Concept and Theory	13
6.2	Numerical Results	14
7	Combined Approach	15
7.1	Numerical Results	16
8	Discussion	17
9	Conclusion	20

List of Figures

1	Distribution of 10,000 collocation points at first epoch	7
2	Plot of the ground truth solution of one-soliton focusing NLSE. The choice of the plots was inspired by Serkin & Belyaeva [8]. <i>Left</i> : 3D surface of the baseline magnitude $ h(x, t) $ over (x, t) . <i>Center</i> : Waterfall of spatial slices $ h(x, t_i) $ at selected times t_i showing temporal evolution. <i>Right</i> : 2D heatmap of $ h(x, t) $ in the (x, t) plane with color indicating magnitude.	8

3	Baseline Hybrid PINNsTF2-DeepXDE: (a) shows the predictions made after the training, while (b) illustrates the logging error plots from W&B system.	10
4	Hybrid PINNsTF2-DeepXDE with RAR enabled: (a) shows the predictions made after the training, while (b) illustrates the logging error plots from W&B system. . .	12
5	Hybrid PINNsTF2-DeepXDE with $L2$ Conservation Law: (a) shows the predictions made after the training, while (b) illustrates the logging error plots from W&B system.	15
6	Hybrid PINNsTF2-DeepXDE with RAR and $L2$ Conservation Law: (a) shows the predictions made after the training, while (b) illustrates the logging error plots from W&B system.	17

List of Tables

1	Performance comparison of PDE loss weight values affecting the performance of Baseline Hybrid PINNsTF2-DeepXDE.	11
2	Performance comparison of PDE loss weight values affecting the performance of Hybrid PINNsTF2-DeepXDE with RAR.	13
3	Performance comparison of PDE loss weight values affecting the performance of Hybrid PINNsTF2-DeepXDE with $L2$	15
4	Performance comparison of PDE loss weight values affecting the performance of Hybrid PINNsTF2-DeepXDE with $L2$ and RAR	17
5	Performance comparison of various PINN configurations.	18

1 Introduction

In recent years, neural network-based approaches have emerged as an alternative framework for solving Partial differential equations (PDEs). These techniques are especially effective for exploring and characterizing nonlinear, high-dimensional, and parametric PDEs where initial conditions are subject to variation. This Capstone investigates a Machine Learning (ML) model to solve PDEs – Physics-Informed Neural Network (PINN) [1], first introduced in 2019 with several demonstrated applications for solving various PDEs. PINNs have gained significant attention due to their ability to incorporate physical laws directly into the learning process. They approximate the solution of a PDE by training a neural network to minimize a loss function that encodes the governing equation, as well as initial and boundary conditions. Moreover, unlike standard black-box deep learning models, PINNs leverage physical laws as a regularization agent to achieve exceptional data efficiency and interpretability, while their mesh-free nature allows them to solve high-dimensional problems without the prohibitive discretization costs and heavy preparation required by traditional numerical solvers [1, 2].

Despite their transformative potential, PINNs do face several downsides such as high training costs and slow convergence speeds when compared to established numerical solvers [3, 4]. Another issue with PINNs arises from the composite nature of the loss function, which frequently leads to imbalanced or biased gradients where the network fails to simultaneously minimize the losses over the governing PDE, initial conditions, and boundary conditions [3, 5]. Furthermore, standard fully-connected architectures often suffer from spectral bias, which is an inability to learn high-frequency functions, and are susceptible to getting stuck in local minima or vanishing gradients [3, 5]. For time-dependent PDEs, standard PINN models can also violate temporal causality, leading to sampling confusion and convergence to non-physical or trivial solutions [2]. Such disadvantages make it difficult for PINNs to have significant practicality for real-time applications, making traditional numerical methods an optimal choice.

To address PINNs’ limitations such as imbalanced gradients, slow training, and the difficulty in resolving local features, researchers have introduced adaptive sampling strategies such as Residual-based Adaptive Refinement (RAR) and Predictive Dynamic Monitoring sampling (PDM) [2, 6, 7, 8]. Implementing RAR and PDM during training allows for dynamical gathering of collocation points in regions where PDE residuals are the largest or solution variations are the most rapid [2, 7]. In their research, Guo et al. [2] implement novel adaptive causal sampling methods that emphasize respecting the temporal order of physical evolution to prevent the network from converging to incorrect solutions. It can be seen that various options for selecting collocation points help counteract several shortcomings of the basic PINN setup. Beyond sampling, Wu et al. [9] proposed the integration of conservation laws into the loss function has emerged as a strong method that significantly improves the calculation accuracy for complex wave structures like rogue waves and soliton molecules. Scalability and generalization have been further enhanced through domain de-

composition and segmentation approaches, including Conservative PINNs (cPINNs) [10], Extended PINNs (XPINNs) [11], and Mixture-of-Experts PINNs (MoE-PINNs) [12], which train specialized sub-networks on distinct subdomains to handle high-dimensional or discontinuous regimes. Theoretical analysis through the lens of the Neural Tangent Kernel (NTK), performed by Wang et al. [5], has enabled the development of adaptive algorithms that calibrate the imbalanced convergence rates of disparate loss components, thereby stabilizing the overall global optimization process.

Initially, Raissi et al. [1] provided the sample scripts to train PINNs on several PDEs, such as Burgers' and Navier-Stokes equations. Recently, advanced methodologies have been successfully applied to a diverse array of complex, nonlinear partial differential equations across multiple physical domains. The Nonlinear Schrödinger Equation (NLSE) hierarchy, including its standard cubic form and generalized quintic and higher-order variants up to the fifth order, has been a primary focus for modeling optical solitons, rogue waves, and soliton molecules [6, 8, 9]. Furthermore, these methods have proven effective for dispersive wave models like the Korteweg-de Vries (KdV) and Boussinesq equations, as well as fundamental systems such as the Poisson and wave equations [2, 5, 6]. This broad spectrum of successful applications underscores the versatility of PINNs as a general-purpose framework for scientific computing.

Building on these advancements, this Capstone project focuses on applying two strategies to enhance the robustness of the standard PINNs for NLSE: Residual-Based Adaptive Refinement (RAR) [7] and physical Conservation Law $L2$ [9]. These methods are employed to address limitations in original PINNs, such as the inefficiency of fixed collocation sampling in resolving sharp gradients. By attuning the model's capacity through RAR and enforcing global physical constraints through conservation laws, this study aims to evaluate the performance of altered PINN-based solvers and compare it to the baseline. On top of that, within the frame of this work, the weights on PDE residuals during training, employed from DeepXDE library [13], are also altered for expansion of study cases. Every model is trained on identical configurations to guarantee fairness of comparisons.

2 Problem Formulation and Exact Solution

Partial differential equations (PDEs) play a central role in modeling a wide range of physical phenomena, including quantum mechanics, fluid dynamics, and wave propagation. This work revolves around one of the classical examples of wave equations, which describe the evolution of physical systems in space and time, is the Non-linear Schrödinger Equation (NLSE), which was extensively covered in the original paper on PINNs [1].

2.1 Non-linear Schrödinger Equation

Beyond its role in quantum mechanics, the Schrödinger equation is a vital model for wave propagation in nonlinear media. In the study by Raissi et al. [1], the (1+1)-dimensional focusing NLSE, distinguished by the positive sign of the nonlinear term $+|h|^2h$, is employed as a benchmark to test the ability of PINNs to resolve complex-valued fields. In the focusing regime, the nonlinearity and the group velocity dispersion balance each other to produce stable, non-dispersive wave structures known as bright solitons. The system is defined as:

$$\begin{cases} ih_t + \frac{1}{2}h_{xx} + |h|^2h = 0, & x \in [-5, 5], t \in [0, \pi/2], \\ h(0, x) = 2 \operatorname{sech}(x) \\ h(t, -5) = h(t, 5), \\ h_x(t, -5) = h_x(t, 5). \end{cases} \quad (1)$$

In this system, $h(t, x)$ represents a complex-valued wave function, which the neural network must approximate by learning its real and imaginary components. The governing equation, $ih_t + \frac{1}{2}h_{xx} + |h|^2h = 0$, describes the evolution of the wave through a combination of a temporal derivative h_t , a second-order dispersive term h_{xx} , and a cubic nonlinear term $|h|^2h$ that models the self-interaction of the field.

Regarding the boundary conditions, there is a mathematical inconsistency between a theoretically non-periodic sech-type soliton, which exists on an infinite domain, and the periodic boundary conditions applied here. To resolve this, the periodicity is simulated by truncating the domain to the finite interval $x \in [-5, 5]$. The initial condition $h(0, x) = 2 \operatorname{sech}(x)$ is specifically chosen because the hyperbolic secant profile is an exact analytical eigensolution for the focusing NLSE. With an initial amplitude of $A = 2$, the solution becomes a second-order bound-state soliton, which does not disperse and undergoes periodic oscillations instead, reproducing its initial shape at intervals of $t = \pi/2$. Furthermore, periodic boundary conditions are enforced at $x = \pm 5$, requiring that both the wave function h and its spatial derivative h_x remain identical at the boundaries ($h(t, -5) = h(t, 5)$ and $h_x(t, -5) = h_x(t, 5)$) to ensure a continuous and physically consistent wrap-around of the wave structure.

2.2 Exact & Analytical Solution

The (1+1)-dimensional focusing NLSE is an integrable system, meaning it admits exact analytical solutions through the Inverse Scattering Transform (IST). For the specific initial condition $h(0, x) = 2 \operatorname{sech}(x)$, the solution is a higher-order soliton that exhibits periodic behavior in time. The exact analytical expression for the wave function $h(t, x)$ is given by:

$$h(t, x) = \frac{4 (\cosh(3x) + 3e^{4it} \cosh(x)) e^{it/2}}{\cosh(4x) + 4 \cosh(2x) + 3 \cos(4t)} \quad (2)$$

In this Capstone, this analytical expression serves as the "ground truth" or reference solution. By evaluating this function across the spatio-temporal domain $(t, x) \in [0, \pi/2] \times [-5, 5]$, we can directly quantify the accuracy of the Physics-Informed Neural Network (PINN). The performance of the model is typically measured by the relative ℓ_2 error between the predicted solution $\hat{h}(t, x)$ and this exact baseline, specifically testing the network's ability to capture the periodic breathing motion and the phase shifts of the soliton pulse without numerical dissipation.

3 Physics-Informed Neural Networks

3.1 PINN Formulation

In the PINN framework, the complex-valued solution $h(t, x)$ is approximated by a deep neural network $h_\theta(t, x)$ with parameters θ . Because standard neural network architectures are real-valued, we define the network to output two distinct real-valued components $u(t, x)$ and $v(t, x)$:

$$h_\theta(t, x) = \begin{bmatrix} u(t, x) \\ v(t, x) \end{bmatrix} \approx \begin{bmatrix} \text{Re}(h) \\ \text{Im}(h) \end{bmatrix}. \quad (3)$$

To ensure the network respects the physics laws, a complex-valued PDE residual $f(t, x)$ is defined by substituting the network outputs into Equation 1:

$$f(t, x) := ih_t + \frac{1}{2}h_{xx} + |h|^2h. \quad (4)$$

Separating $f(t, x)$ into its real, f_u , and imaginary, f_v , components using the network outputs u and v , yields:

$$f_u := -v_t + \frac{1}{2}u_{xx} + (u^2 + v^2)u, \quad (5)$$

$$f_v := u_t + \frac{1}{2}v_{xx} + (u^2 + v^2)v. \quad (6)$$

The parameters θ are optimized by minimizing a composite loss function $\mathcal{L}(\theta)$ that penalizes deviations from the governing PDE, initial conditions, and boundary constraints as below:

$$\mathcal{L}_1(\theta) = w_f \mathcal{L}_{\text{PDE}} + w_{ic} \mathcal{L}_{\text{IC}} + w_{bc} \mathcal{L}_{\text{BC}}. \quad (7)$$

where w_f , w_{ic} , and w_{bc} are scalar loss weights that control the relative importance of satisfying the PDE residual, the initial condition, and the boundary condition, respectively, during optimization. Since the individual loss terms can differ significantly in scale, these weights are used to balance

their contributions to the total objective and its gradient, thereby influencing the balance between interior physics enforcement and constraint matching. In fact, Equation 7 is later referred to as $L1$, while $L2$ conservation law loss function is defined as:

$$\mathcal{L}_2(\theta) = w_f \mathcal{L}_{\text{PDE+en}} + w_{ic} \mathcal{L}_{\text{IC}} + w_{bc} \mathcal{L}_{\text{BC}}, \quad (8)$$

where $\mathcal{L}_{\text{PDE+en}}$ denotes the mean-squared residual over the NLS equations (f_u, f_v) and the energy conservation residual f_{en} , all weighted together by w_f as a single interior block in the training objective [9]. Essentially, the structural change between $L1$ (Equation 7) and $L2$ (Equation 8) is the expansion of the interior residual block to include the energy conservation law, effectively treating the violation of physical invariants as an additional governing constraint that is optimized simultaneously with the fundamental differential equations. While $L1$ focuses purely on the differential form of the physics, $L2$ integrates these conservation structures into the same weighted component to provide a constraint that ensures global physical consistency across the collocation set.

The individual loss components are defined as follows:

- **PDE Residual Loss:** Enforces the governing equation at N_f collocation points throughout the domain:

$$\mathcal{L}_{\text{PDE}} = \frac{1}{N_f} \sum_{i=1}^{N_f} (|f_u(t_i, x_i)|^2 + |f_v(t_i, x_i)|^2). \quad (9)$$

- **Initial Condition Loss:** Matches u and v at $t = 0$ on N_0 points to the reference fields from the dataset $(u^{\text{ref}}, v^{\text{ref}}; \text{ for the single-soliton case } u^{\text{ref}}(0, x) = 2 \text{sech}(x), v^{\text{ref}}(0, x) = 0)$:

$$\mathcal{L}_{\text{IC}} = \frac{1}{N_0} \sum_{i=1}^{N_0} (|u_\theta(0, x_i) - u^{\text{ref}}(0, x_i)|^2 + |v_\theta(0, x_i) - v^{\text{ref}}(0, x_i)|^2). \quad (10)$$

- **Periodic Boundary Loss:** Enforces periodicity of u , v , and $\partial u/\partial x$, $\partial v/\partial x$ at the spatial endpoints $x_{\min}, x_{\max}$ (paired at the same t_i) for N_b samples:

$$\begin{aligned} \mathcal{L}_{\text{BC}} = \frac{1}{N_b} \sum_{i=1}^{N_b} & \left(|u_\theta(t_i, x_{\min}) - u_\theta(t_i, x_{\max})|^2 + |v_\theta(t_i, x_{\min}) - v_\theta(t_i, x_{\max})|^2 \right. \\ & \left. + \left| \frac{\partial u_\theta}{\partial x}(t_i, x_{\min}) - \frac{\partial u_\theta}{\partial x}(t_i, x_{\max}) \right|^2 + \left| \frac{\partial v_\theta}{\partial x}(t_i, x_{\min}) - \frac{\partial v_\theta}{\partial x}(t_i, x_{\max}) \right|^2 \right). \end{aligned} \quad (11)$$

The derivatives required for these loss terms (such as u_t, u_{xx}, v_t, v_{xx}) are computed efficiently using reverse-mode automatic differentiation (TensorFlow), allowing the PINN to be trained via gradient-based optimization algorithms such as Adam or L-BFGS.

3.2 Training Procedure

The PINN is trained to solve the focusing NLSE over a fixed space-time domain. Training data are sampled from interior collocation points (for PDE enforcement), initial-condition points (for the starting wave profile), and boundary points (for periodic behavior). A fully connected neural network takes (t, x) as input and predicts the real and imaginary parts of the solution. Training minimizes a combined mean-squared-error loss consisting of PDE, initial-condition, and boundary-condition terms. All derivatives are obtained through reverse-mode automatic differentiation. To make the setup reproducible and measurable, training uses regular validation checks and experiment logging, which also allows for convenient estimation on how certain parameters affected the training and overall performance. The logging was executed using Weights & Biases (W&B) [14] platform, which stores information as well as displays training process in real time.

3.3 Configuration

The following configuration choices directly shape training behavior and contain the general configuration chosen for models evaluation:

- **N_f (number of collocation points):** Controls how strongly the PDE is enforced in the interior; increasing N_f usually improves physics consistency but raises training cost. The general N_f is 20,000.
- **N_0 (initial-condition points):** Controls fidelity to the initial wave profile; too few points can weaken early-time accuracy. For all model trainings, N_0 is set at 100.
- **N_b (boundary-condition points):** Controls how well periodic boundary constraints are learned; larger values typically reduce boundary mismatch. In the general configuration, the N_b is equal 200.
- **Sampling Method:** We use a scrambled Sobol sequence to draw collocation and RAR candidate points because it gives more uniform coverage of the (x, t) domain than plain random sampling, assists clustering that can waste points in smooth regions, and remains reproducible when a fixed seed is set-while still benefiting from scrambling to avoid systematic alignment.
- **Network depth/width:** Controls how much the MLP can fit after the input is mapped; bigger nets are more expressive but harder to train. For all model, including the baseline, there are three hidden layers composed of 40 neurons.
- **Maximum epochs:** Sets total optimization budget; more epochs can improve convergence, but may increase runtime and risk overfitting to sampled points. Overall, the models in this

study were given 20,000, with up to 10,000 epochs for first stage of Adam optimizer, and the remaining 10,000 epoch trained using L-BFGS optimizer.

- **Validation frequency:** Determines how often generalization is monitored; frequent checks improve visibility into training progress. The model performance is reported every 100 epochs for this work.
- **Loss function / weighting strategy:** Weights w_f , w_{ic} , and w_{bc} control the balance between PDE, IC, and BC objectives, respectively. Changing this shifts what the model prioritizes during training. By default, $w_{ic} = w_{bc} = 100$, while $w_f \in \{1, 2, 4\}$, for this study. For models under the additional conservation law constraints, configurations switch to mode $L2$, otherwise it is $L1$ by default.
- **Random seed:** Affects data sampling and initialization; fixed seeds improve reproducibility across runs. The random seed in general configuration is 1234.

3.4 Collocation Points

Figure 1 displays the distribution of the sampled collocation points during the initial training epoch. The points were selected using Sobol sequence [15], which allowed for consistent and reproducible distribution of points, and better performance of the model [8]. Serkin & Belyaeva [8] compared Sobol sequence sampling to other known sampling methods such as LHS (2.04×10^{-3}), random (2.39×10^{-3}), equidistance grid 3.16×10^{-3} , where it achieved 1.77×10^{-3} .

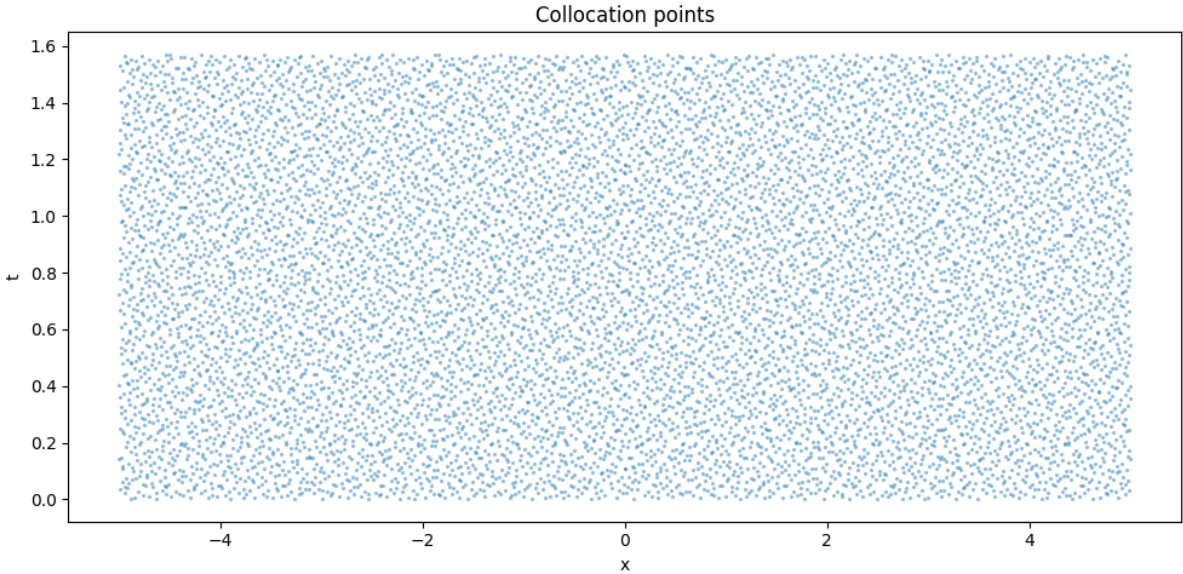


Figure 1: Distribution of 10,000 collocation points at first epoch

3.5 Error Metrics

Performance is mainly quantified using a discrete relative ℓ_2 error metric, also used in the original [1, 16] and many other studies assessing variations of PINNs [2, 5, 6, 8, 9]. For each scalar field (i.e. u , v , or $h = \sqrt{u^2 + v^2}$), predictions and references are compared on the same validation grid points. The reported relative error is

$$e_{\text{rel}} := \sqrt{\frac{\sum_{i=1}^N (\hat{y}_i - y_i)^2}{\sum_{i=1}^N (y_i)^2}} \quad (12)$$

where $\hat{y}_i$ is the model prediction and y_i the reference value at grid point i . Thus the numerator is the mean squared error, $\frac{1}{N} \sum_{i=1}^N (\hat{y}_i - y_i)^2$, and the denominator is the mean squared reference level, $\frac{1}{N} \sum_{i=1}^N y_i^2$; equivalently, the metric is the ratio of the discrete ℓ_2 norms, $\|\hat{\mathbf{y}} - \mathbf{y}\|_2 / \|\mathbf{y}\|_2$, on the evaluation grid. The rationale is to report the root-mean-square error of the prediction relative to the typical magnitude (RMS) of the reference field, so that errors are measured in a scale-invariant way with respect to the overall amplitude of y on the grid.

As a supportive complement, the visualization of the comparison of ground truth labels to the model predictions is also yielded after each training session. This helps to observe which (t,x) points the model performed better or worse. The Figure 2 below is the ground truth plot of what is expected to be displayed by model predictions.

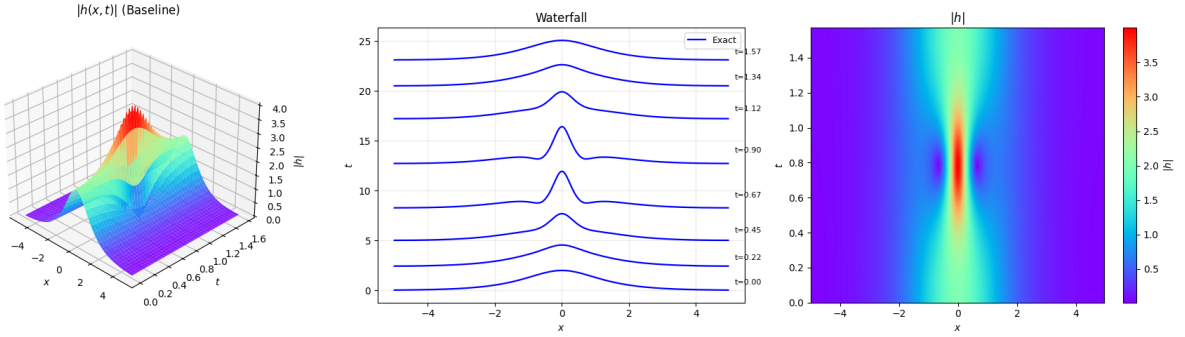


Figure 2: Plot of the ground truth solution of one-soliton focusing NLSE. The choice of the plots was inspired by Serkin & Belyaeva [8]. *Left*: 3D surface of the baseline magnitude $|h(x,t)|$ over (x,t) . *Center*: Waterfall of spatial slices $|h(x,t_i)|$ at selected times t_i showing temporal evolution. *Right*: 2D heatmap of $|h(x,t)|$ in the (x,t) plane with color indicating magnitude.

4 Baseline

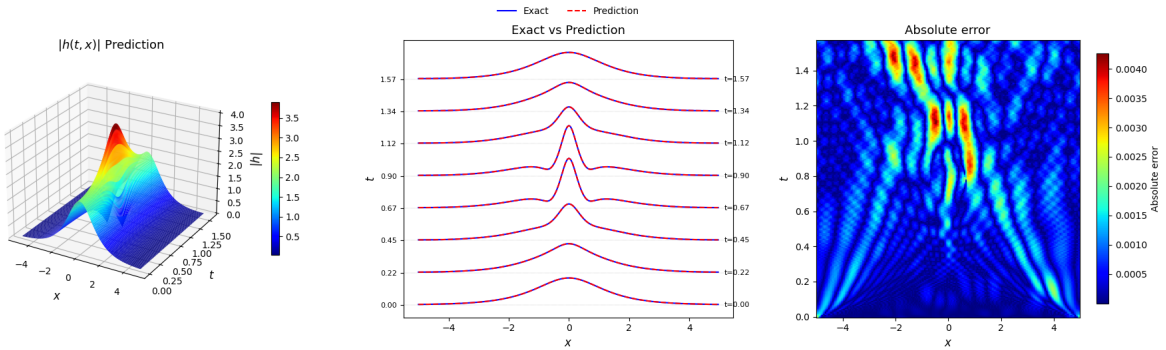
In this study, a hybrid PINNsTF2–DeepXDE model is treated as a baseline. It extends the original TensorFlow 2 PINNsTF2 pipeline [16] for the nonlinear Schrödinger equation. It is com-

pared to models that integrated Conservation Law, RAR, or both. The baseline configuration uses the enhanced network, weighted losses, learning-rate decay, and Adam optimizer followed by L-BFGS-B, with L1 (PDE residual only). Moreover, this study adopted Sobol Sequence Sampling [8] that ensure reproducibility of results and minimize randomness behind the model performances. Conservation-law variants switch to L2, which augment the collocation loss with energy residuals while keeping the same data, domain, and training schedule for a fair comparison. For the other models, RAR is toggled via the configuration settings so that runs differ only in whether collocation points are fixed or periodically enriched by residual magnitude runs. Model quality is assessed by relative ℓ_2 errors on u , v , and $h = \sqrt{u^2 + v^2}$ over the validation grid.

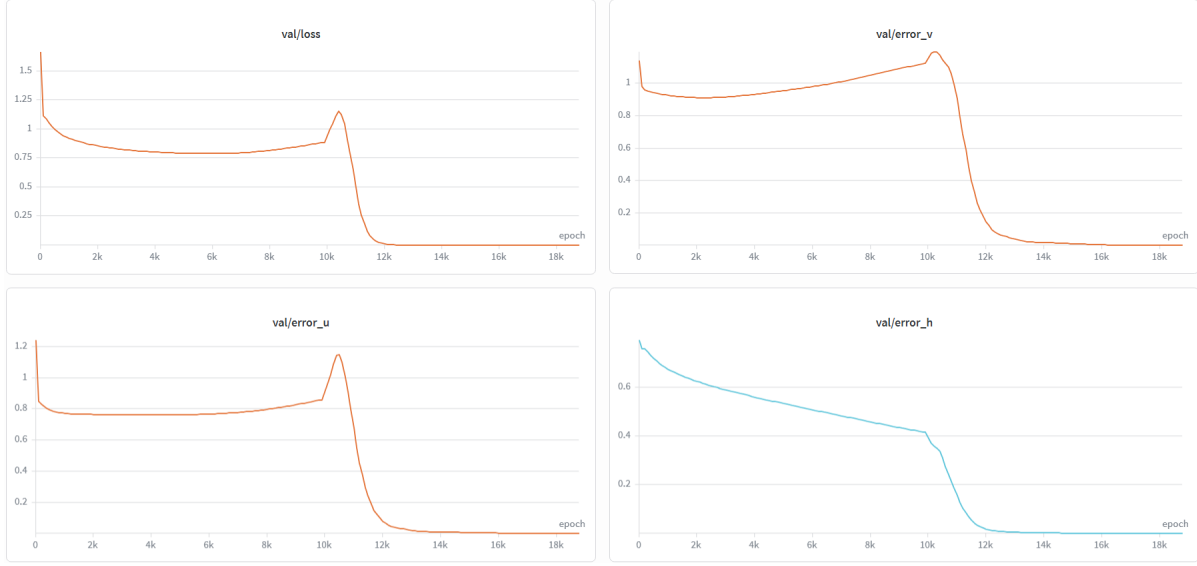
4.1 Numerical Results

The baseline hybrid PINNsTF2–DeepXDE model that used $L1$ without RAR completed 18,780 epochs and has delivered the results in the expected fashion. This run serves as a deliberate baseline, since neither the L2 conservation-law augmentation nor residual-based adaptive refinement was activated, so any improvement observed in later configurations can be attributed to those mechanisms rather than to changes in network architecture, or the Sobol-based static sampling strategy illustrated in Figure 1.

Quantitatively, the trained network achieved low relative ℓ_2 errors on the validation grid for the real and imaginary parts and for the modulus $h = \sqrt{u^2 + v^2}$ (1.041×10^{-3} for h , 2.089×10^{-3} for u , 3.115×10^{-3} for v), indicating close agreement with the reference solution over the full (x, t) domain.



(a) Hybrid PINNsTF2–DeepXDE baseline predictions



(b) Hybrid PINNsTF2-DeepXDE baseline training logs of errors for functions h, u, v , and $loss$

Figure 3: Baseline Hybrid PINNsTF2-DeepXDE: (a) shows the predictions made after the training, while (b) illustrates the logging error plots from W&B system.

The training logs in Figure 3b illustrate a characteristic behavior across all functions (h, u, v). The Adam optimizer stage (first 10,000 epochs) successfully captures the global structure but reaches a plateau. The transition to the L-BFGS optimizer (visible as a spike in the loss plots at the 10k mark) triggers a rapid convergence phase, refining the solution and reducing the error. The absolute error map in Figure 3a shows that the maximum error is concentrated at the soliton peak, remaining below 4×10^{-3} .

Another comparison is to the model that used original PINNsTF2 with the same configuration [8] that achieved 4.78×10^{-3} . Baseline PINNsTF2-DeepXDE has decreased the relative ℓ_2 error on h by roughly 78.2% relative to that. This sets up the ground for general investigation of the effect that conservation laws and refinement have on model performance.

As shown in Table 1, the PDE loss weight (w_f) significantly influences the approximation accuracy. Increasing the weight from $w_f = 1$ to $w_f = 2$ resulted in a 24.8% reduction in the relative ℓ_2 error (1.041×10^{-3} to 7.827×10^{-4}). However, further increasing the weight to $w_f = 4$ led to a slight degradation in performance for solution h (9.259×10^{-4}), suggesting an optimization trade-off between the PDE residual and the initial/boundary constraints.

Table 1: Performance comparison of PDE loss weight values affecting the performance of Baseline Hybrid PINNsTF2-DeepXDE.

PDE Loss Weight w_f	Epochs completed	Relative ℓ_2 Error	Validation Error on u	Validation Error on v
1	Adam 10k, L-BFGS 8.78k	1.041×10^{-3}	2.089×10^{-3}	3.115×10^{-3}
2	Adam 9.4k, L-BFGS 8.3k	7.827×10^{-4}	1.168×10^{-3}	1.641×10^{-3}
4	Adam 8.2k, L-BFGS 7.38k	9.259×10^{-4}	1.167×10^{-3}	1.521×10^{-3}

5 Residual-Based Adaptive Refinement

Residual-based adaptive refinement (RAR) [7, 13] addresses a common weakness of static collocation in physics-informed learning which is when a fixed set of interior points may undersample regions where the learned field violates the governing equations the most.

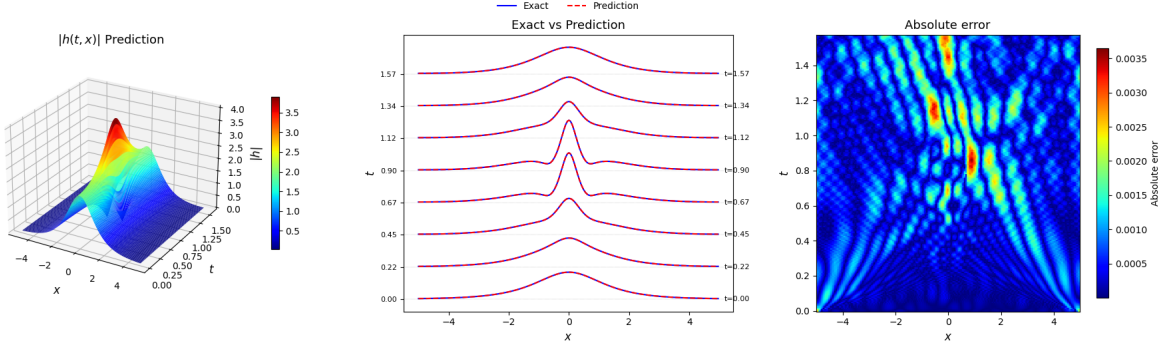
5.1 Concept and Theory

The idea is to periodically evaluate the physics residual on many candidate locations, rank them by magnitude, and expand the training set with points where the residual is largest [13]. In the present hybrid implementation, RAR operates on the *MeshSampler* collocation set used for the interior loss. After an initial space-filling draw (Latin hypercube sampling, LHS), training proceeds for a fixed number of Adam epochs. Whenever the RAR period elapses, a batch of random candidate points is generated in the same (x, t) domain, the forward map and *pde_fn* are applied. The squared residuals summed over the active *collection_points* (e.g. PDE components f_{er}, f_{em} in mode *L1*, or additional conservation residuals in *L2*) define a scalar errors per candidate. The top- k candidates by this score are appended to the existing collocation tensors (*additive* mode), or, as an alternative, the full collocation set can be redrawn (*full* mode) [13]. The training dataloader is then rebuilt so subsequent epochs optimize on the updated point cloud. The computational overhead for resampling is negligible ($O(N)$) compared to gradient backpropagation [7, 17]. By focusing resources on "problematic areas," RAR achieves high accuracy with fewer points, potentially accelerating calculations by nearly 50% and significantly reducing the total training time and resource consumption [6, 17]

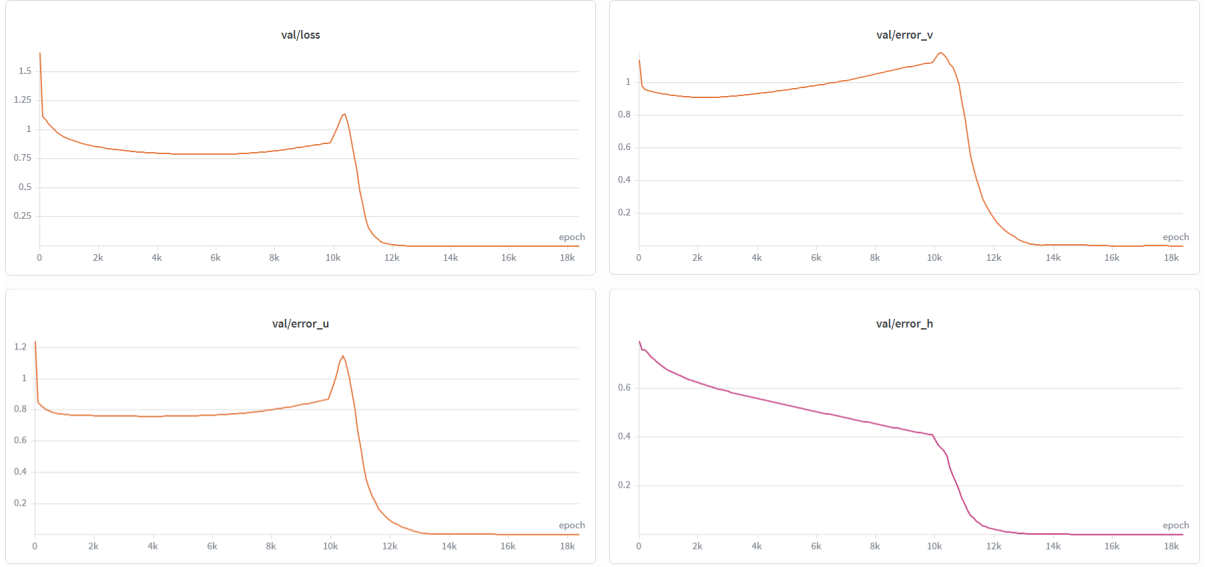
Hyperparameters control the trade-off between cost and adaptivity, defining how often RAR runs, *n_candidates* sets how many random points are scored each time, and *n_add* sets how many high-residual points enter the set per step. When RAR is disabled (*period* = 0), the baseline behaves like standard PINNs with a single LHS collocation draw.

5.2 Numerical Results

The introduction of RAR aimed to improve the resolution of high-gradient areas by dynamically adding collocation points where the PDE residual was highest. Implementing RAR led to improved accuracy compared to the baseline hybrid PINNsTF2-DeepXDE model under $w_f = 1$, achieving relative ℓ_2 errors of 8.700×10^{-4} for h , 1.332×10^{-3} for u , 1.699×10^{-3} for v . Comparing with the baseline PINNsTF2-DeepXDE, a relative improvement of 16.4% can be observed (a total of 81.8% relative to original PINNsTF2 in the same configuration [8]). Convergence behavior is only slightly enhanced.



(a) Hybrid PINNsTF2-DeepXDE + RAR predictions



(b) Hybrid PINNsTF2-DeepXDE + RAR training logs of errors for functions h, u, v , and $loss$

Figure 4: Hybrid PINNsTF2-DeepXDE with RAR enabled: (a) shows the predictions made after the training, while (b) illustrates the logging error plots from W&B system.

Figure 4a highlights the visual improvements of the RAR predictions. Compared to the baseline (see Figure 3), the absolute error map in Figure 4a exhibits a more structured distribution, yet

maintains a low maximum error. The training logs (Figure 4b) show that the validation error on h follows a smoother decay path during the L-BFGS stage than the baseline, indicating that the adaptively placed points helped stabilize the convergence toward the exact analytical solution.

The results in Table 2 demonstrate that RAR consistently improves accuracy within a similar or slightly lower training budget (total epochs). Notably, the configuration with $w_f = 4$ achieved a relative ℓ_2 error of 7.566×10^{-4} (improvement by 27.3% relative to baseline at $w_f = 1$, and by 3.33% to baseline at $w_f = 2$), the highest accuracy among all models tested without additional physical constraints. This suggests a "sweet spot" where the weights are balanced enough to allow the optimizer to reach a deep global minimum without becoming stuck, since standard PINNs often suffer from imbalanced gradients where the network prioritizes fitting initial or boundary conditions over the governing PDE. Increasing w_f helps to calibrate these convergence rates, forcing the model to satisfy the spatiotemporal physics more rigorously [5].

Table 2: Performance comparison of PDE loss weight values affecting the performance of Hybrid PINNsTF2-DeepXDE with RAR.

PDE Loss Weight w_f	Epochs completed	Relative ℓ_2 Error	Validation Error on u	Validation Error on v
1	Adam 10k, L-BFGS 8.37k	8.700×10^{-4}	1.332×10^{-3}	1.699×10^{-3}
2	Adam 9.3k, L-BFGS 7.47k	8.755×10^{-4}	1.414×10^{-3}	2.124×10^{-3}
4	Adam 8.1k, L-BFGS 8.97k	7.566×10^{-4}	9.538×10^{-4}	1.215×10^{-3}

By comparing the first block (see Table 1) to the second block (Table 2), it can be seen that adding RAR generally leads to lower relative ℓ_2 and validation errors. This can be explained by the fact that RAR focuses the model’s learning capacity on regions where PDE residuals are largest, such as the sharp peaks of solitons.

6 Conservation Laws in PINNs

Following the idea of constraining PINNs with auxiliary physics in addition to the governing PDE, the hybrid trainer includes conservation laws inspired by recent work on modified PINNs [9].

6.1 Concept and Theory

The interior loss is built from a set of residual fields registered in *collection_points*, and each residual is penalized with the same mean-squared framework as the classical PDE terms. Following

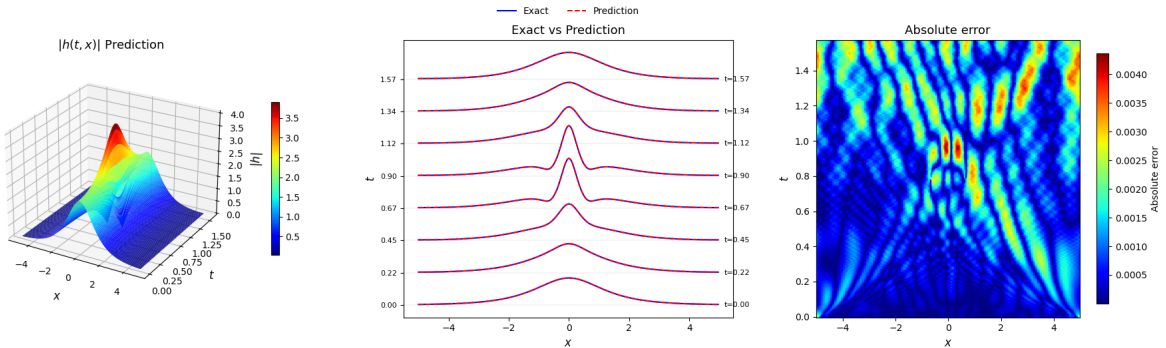
the principles of integrable systems, these residuals incorporate conservation laws, which represent fundamental physical properties, such as energy and momentum, that remain invariant during wave transmission [9].

Mode $L1$ corresponds to the standard formulation, although the complex focusing cubic NLSE split into real and imaginary PDE residuals (denoted here as f_{er} and f_{em}). Mode $L2$ retains those PDE residuals and adds an energy continuity residual (f_{enr}), derived from the energy conservation law which dictates that the power of the light field is conserved over the normalized distance. Thus $L2$ minimizes both equation error and violation of the associated local conservation law on the same collocation set. Mode $L3$ removes the raw PDE block from the collocation loss and instead integrates a pair of momentum-related residuals (f_{mr} , f_{mm}) with the energy residuals. Mode $L4$ keeps only the momentum pair (f_{mr} , f_{mm}) at collocation points. Momentum-based modes involve higher-order derivatives in the graph and can be harder to optimize; the literature also reports mixed benefits of momentum-only or momentum-plus-energy formulations for complex dynamics such as multi-soliton interaction [9].

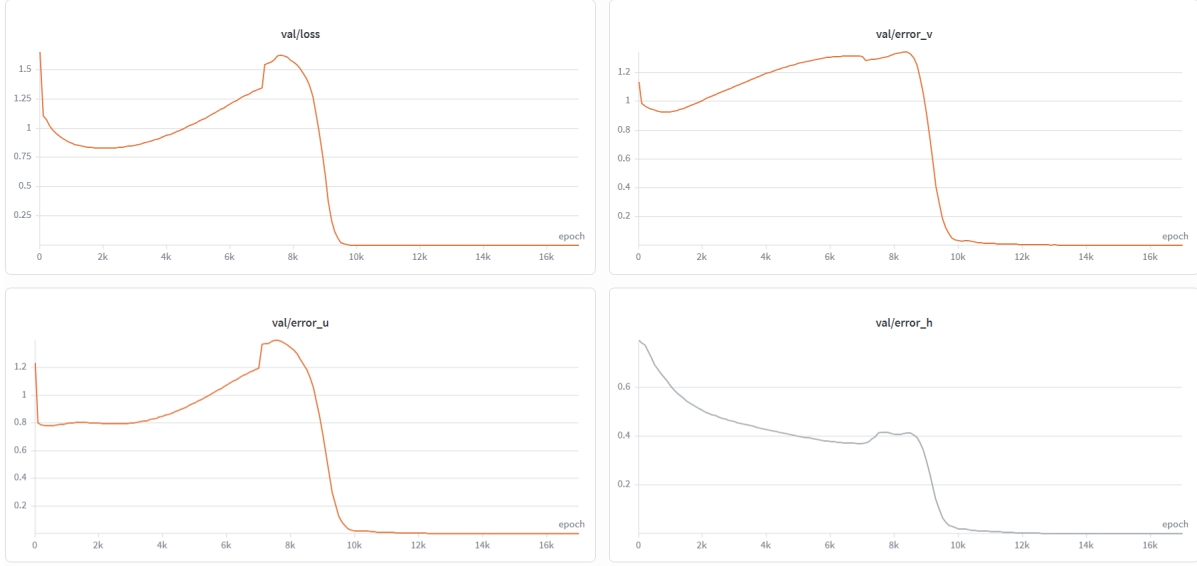
This study reports results only for $L2$, the baseline hybrid model with classical PDE residuals plus the energy (norm-flux) conservation term. Modes $L3$ and $L4$ [9] are described for completeness and for alignment with the reference loss taxonomy, but they are not used in the experiments presented.

6.2 Numerical Results

The inclusion of the $L2$ conservation law enforced physical consistency regarding the total energy of the wave profile. Interestingly, the numerical accuracy for validation error on h was slightly inferior to the baseline, with $w_f = 1$ yielding an ℓ_2 error of 1.339×10^{-3} (Table 3), while the components u and v were more accurate in that comparison (1.559×10^{-3} and 2.173×10^{-3} , respectively). This observation corresponds to the nature of $L2$ conservation law, which provides a strong binding force that constrains the network to respect physical invariants like energy. The accuracies improved as the PDE loss weight increased, with $w_f = 4$ reaching 1.042×10^{-3} for h .



(a) Hybrid PINNsTF2-DeepXDE + $L2$ predictions



(b) Hybrid PINNsTF2-DeepXDE + L_2 training logs of errors for functions h, u, v , and $loss$

Figure 5: Hybrid PINNsTF2-DeepXDE with L_2 Conservation Law: (a) shows the predictions made after the training, while (b) illustrates the logging error plots from W&B system.

Figure 5b reveals a more complex optimization landscape in the training logs. The "spike" at the Adam-to-L-BFGS transition lasts longer, and the loss for function h exhibits several small oscillations before final convergence. This suggests that the additional L_2 term provides a strong regularization effect that can occasionally compete with the standard PDE residual during the early stages of optimization. Despite this, Figure 5a shows excellent agreement between the exact and predicted snapshots at various time intervals.

Table 3: Performance comparison of PDE loss weight values affecting the performance of Hybrid PINNsTF2-DeepXDE with L_2 .

PDE Loss Weight w_f	Epochs completed	Relative ℓ_2 Error	Validation Error on u	Validation Error on v
1	Adam 7k, L-BFGS 10k	1.339×10^{-3}	1.559×10^{-3}	2.173×10^{-3}
2	Adam 6.4k, L-BFGS 10k	1.255×10^{-3}	1.425×10^{-3}	2.071×10^{-3}
4	Adam 6.2k, L-BFGS 10k	1.042×10^{-3}	1.340×10^{-3}	1.689×10^{-3}

7 Combined Approach

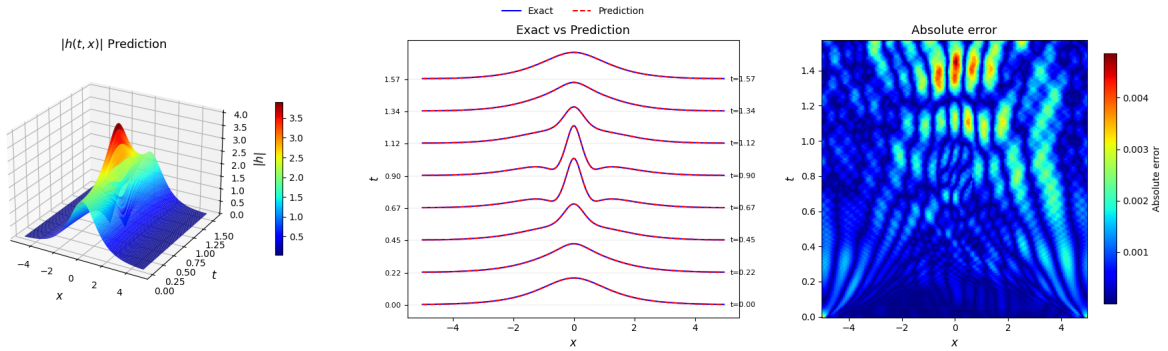
To evaluate whether RAR and conservation constraints provide complementary benefits, this study considers a combined configuration in which RAR is enabled while the collocation loss is

augmented with the energy conservation law. Concretely, RAR is toggled on with *period*=1000 (and fixed 10,000 candidates and 10 add counts), and set the conservation mode to *L2*. Relative to the baseline *L1* pipeline, this modifies the model in two ways. First, the *physics loss* at interior points expands from the classical PDE residual pair to an extended set that additionally penalizes overseeing the energy conservation residual. As a result, the collocation objective becomes the sum of mean-squared errors over both the PDE residual terms and the energy residual term, while the initial-condition and periodic boundary-condition losses remain the same. Second, the *training point set* becomes dynamic, as every 1000 Adam epochs, the current network is used to score a large pool of randomly drawn candidate points by the magnitude of the active collocation residuals, which now includes the energy residual under *L2*. This combined approach therefore integrates expansion of physical constraints that encourages both equation satisfaction and conservation consistency, together with an adaptive data mechanism that further concentrates collocation capacity where the extended residual is large.

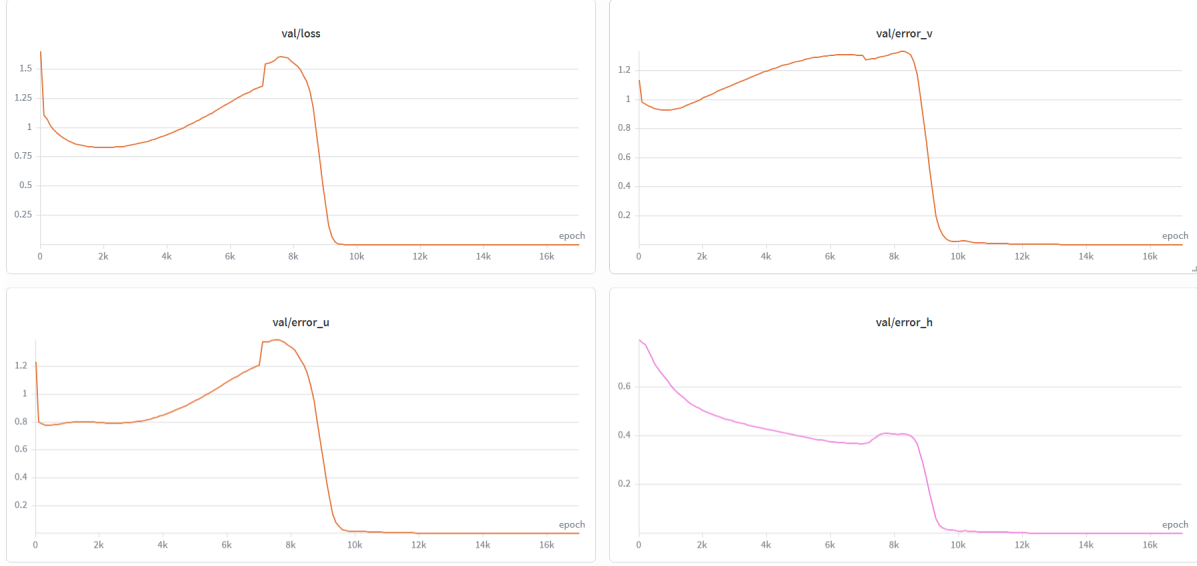
7.1 Numerical Results

The combined method achieved the following results for relative ℓ_2 errors 1.129×10^{-3} for h , 1.437×10^{-3} for u , 1.800×10^{-3} for v , demonstrating better accuracy and stability than baseline.

The visualizations in Figure 6a represent the highest degree of physical consistency. The absolute error map is notably sparse compared to the baseline, indicating that the RAR successfully focused on the soliton trajectory while the *L2* constraint maintained the wave's global energy profile. The logs in Figure 6b demonstrate that this combination entire span of L-BFGS epochs to reach stability (10,000 epochs).



(a) Hybrid PINNsTF2-DeepXDE + RAR + *L2* predictions



(b) Hybrid PINNsTF2-DeepXDE + RAR + L_2 training logs of errors for functions h, u, v , and $loss$

Figure 6: Hybrid PINNsTF2-DeepXDE with RAR and L_2 Conservation Law: (a) shows the predictions made after the training, while (b) illustrates the logging error plots from W&B system.

The final experiment evaluated the synergy between dynamic point refinement and physical constraints. As summarized in Table 4, the combination proved highly effective, particularly at $w_f = 2$, which achieved a relative ℓ_2 error of 8.689×10^{-4} . Moreover, the absolute minimum error was slightly higher than the RAR-only model at $w_f = 4$.

Table 4: Performance comparison of PDE loss weight values affecting the performance of Hybrid PINNsTF2-DeepXDE with L_2 and RAR .

PDE Loss Weight w_f	Epochs completed	Relative ℓ_2 Error	Validation Error on u	Validation Error on v
1	Adam 7k, L-BFGS 10k	1.139×10^{-3}	1.475×10^{-3}	1.771×10^{-3}
2	Adam 6.4k, L-BFGS 10k	8.689×10^{-4}	1.340×10^{-3}	1.449×10^{-3}
4	Adam 6.2k, L-BFGS 10k	1.129×10^{-3}	1.437×10^{-3}	1.800×10^{-3}

8 Discussion

This study compares a PINNsTF2-DeepXDE baseline model to enhanced variants incorporating Residual-based Adaptive Refinement (RAR), conservation-law (L_2) constraint, and different PDE loss weights for solving the nonlinear Schrödinger equation (NLSE).

The results indicate that the accuracy of the PINN is governed by the interaction between RAR, physics-based regularization through the conservation-law term in $L2$, and the weighting of the PDE residual in the total loss. Overall, the strongest improvement comes from RAR (see 5), while the effect of the conservation-law constraint is more nuanced and appears to depend on the chosen loss weight.

Table 5: Performance comparison of various PINN configurations.

RAR	Cons. Law ($L2$)	PDE Loss Weight w_f	Epochs	Relative ℓ_2 Error	Validation Error on u	Validation Error on v
—	—	1	Adam 10k, L-BFGS 8.78k	1.041×10^{-3}	2.089×10^{-3}	3.115×10^{-3}
—	—	2	Adam 9.4k, L-BFGS 8.3k	7.827×10^{-4}	1.168×10^{-3}	1.641×10^{-3}
—	—	4	Adam 8.2k, L-BFGS 7.38k	9.259×10^{-4}	1.167×10^{-3}	1.521×10^{-3}
✓	—	1	Adam 10k, L-BFGS 8.37k	8.700×10^{-4}	1.332×10^{-3}	1.699×10^{-3}
✓	—	2	Adam 9.3k, L-BFGS 7.47k	8.755×10^{-4}	1.414×10^{-3}	2.124×10^{-3}
✓	—	4	Adam 8.1k, L-BFGS 8.97k	7.566×10^{-4}	9.538×10^{-4}	1.215×10^{-3}
—	✓	1	Adam 7k, L-BFGS 10k	1.339×10^{-3}	1.559×10^{-3}	2.173×10^{-3}
—	✓	2	Adam 6.4k, L-BFGS 10k	1.255×10^{-3}	1.425×10^{-3}	2.071×10^{-3}
—	✓	4	Adam 6.2k, L-BFGS 10k	1.042×10^{-3}	1.340×10^{-3}	1.689×10^{-3}
✓	✓	1	Adam 7k, L-BFGS 10k	1.139×10^{-3}	1.475×10^{-3}	1.771×10^{-3}
✓	✓	2	Adam 6.4k, L-BFGS 10k	8.689×10^{-4}	1.138×10^{-3}	1.449×10^{-3}
✓	✓	4	Adam 6.2k, L-BFGS 10k	1.129×10^{-3}	1.437×10^{-3}	1.800×10^{-3}

Based on observations, RAR is the clearest driver of error reduction in this study. The best overall result in Table 5 is obtained by the RAR-only model with $w_f = 4$, which achieves a relative ℓ_2 error of 7.566×10^{-4} , together with the lowest validation errors for both component functions u and v . This suggests that adaptive refinement is especially effective for the NLSE, where localized structures such as soliton peaks and rapidly varying regions are difficult to resolve with uniformly distributed collocation points [2, 17]. This is further confirmed by the fact that RAR extended the

number of collocation points (interior), but not initial or boundary condition points, throughout the experiments. By sampling new points from regions of high residual, RAR allows the network to concentrate representational capacity where the solution is hardest to learn. At the same time, these gains are achieved without a substantial increase in the training budget, since its mean epoch time was 0.1523s for baseline PINNsTF2-DeepXDE model compared to 0.1555s with RAR enabled ($period=1000$, $n_add=10$). The findings support the claim that RAR improves efficiency as well as accuracy [2, 6].

However, the benefit of RAR is not completely uniform across all settings. For example, at $w_f = 2$, the baseline model slightly outperforms the RAR-only model in relative ℓ_2 error. This leads to conclude that RAR is most beneficial when it is paired with a suitably balanced loss functions and weights, and that adaptive refinement alone does not guarantee improvement if the optimization is already well-conditioned [5]. In other words, RAR is highly effective, but the scale of impact still depends on how strongly the model is being forced to satisfy the PDE.

The effect of the $L2$ conservation law loss function is more mixed than the adaptive refinement results. In the Conservation-law-only block (Table 5), the relative ℓ_2 error for h (1.339×10^{-3} , 1.255×10^{-3} , and 1.042×10^{-3}) is higher than the baseline for all three values of w_f (1, 2, and 4, respectively), which indicates that the additional constraint does not directly improve numerical accuracy in this implementation. A plausible explanation is that the conservation-law penalty makes the optimization landscape harder to minimize, especially because all conservation-law runs reach the full 10k L-BFGS budget [3, 5, 9]. This suggests that the model has greater difficulty reconciling pointwise fitting, PDE residual minimization, and global invariant preservation simultaneously.

Even so, the conservation-law term should not be viewed as ineffective. Its role appears to be closer to that of a physical regularizer than a pure error-reduction mechanism. Rather than minimizing the relative error directly, it encourages the learned solution to remain compatible with the underlying structure of the NLSE [9]. Lower numerical error and stronger physical consistency are related, but they are not always identical objectives. The conservation-law term may therefore be valuable for preventing non-physical behavior, even when it does not produce the lowest validation error. In fact, $L2$ -only model continued to improve predictions accuracy as the PDE residual loss weights w_f increased.

The role of the PDE loss weight w_f has shown to influence quality of predictions, but Table 5 shows that there is no single universally optimal value. In the baseline and hybrid settings, the best performance occurs at $w_f = 2$, whereas in the RAR-only and Conservation-law-only settings the best result is achieved at $w_f = 4$. This means that the effect of loss weighting depends on the rest of the training strategy. Increasing w_f can help correct the common PINN problem of under-enforcing the governing equation, but an excessively strong PDE term may also make the optimization less flexible when additional constraints are present [2, 5]. Thus, w_f should be interpreted as a tuning parameter that controls the balance between data fitting, boundary/initial

condition satisfaction, and physical law enforcement.

The hybrid model, which combines RAR and the conservation law, provides the most balanced picture. Its best configuration, at $w_f = 2$, achieves a relative ℓ_2 error of 8.689×10^{-4} for h , which is much better than the conservation-law-only runs, although still slightly worse than the best RAR-only result and the baseline at $w_f = 2$. This suggests that the combination of local refinement and global physical constraint can be beneficial, but the synergy is only partial. Within the framework of this study, the conservation law does not amplify the gains of RAR enough to surpass the best adaptive-refinement result. Instead, the hybrid model offers a compromise that retains most of the accuracy benefit of RAR while also embedding an additional level of physical consistency.

Taken together, these findings suggest that, for the NLSE considered here, dynamic collocation is the most effective mechanism for reducing solution error, while the conservation-law term plays a secondary role as a regularizer to help the solution satisfy physical constraints.

Several factors limit the scope and definitive nature of the current results. First, the hyperparameter exploration was restricted to three PDE residuals loss weights ($w_f \in \{1, 2, 4\}$), even though the literature suggests that the primary driver for improved calculation accuracy often resides in the appropriate manual or automatic tuning of these specific coefficients [2]. Second, this study lacks a variance and statistical robustness analysis. Given that PINN training is susceptible to getting stuck in local minima and frequently exhibits non-monotonic trends or instability based on initial parameters, reporting single outcomes per configuration makes it difficult to separate true method effects from stochastic training noise [3]. Third, physical fidelity was inferred indirectly through error behavior, rather than through a direct quantitative report of invariant drift (e.g., energy or mass conservation error over time), which is a standard metric used to validate the integrity of nonlinear wave structures like solitons as stated by Lawal et al. [3]. Finally, because PINN models encounter significant optimization challenges and poor scalability as the nonlinearity of equations increases or the solution domains expand [3], [17], the performance of the proposed framework for higher dimensions or longer simulation windows is currently unknown.

9 Conclusion

In this study, the baseline PINNsTF2-DeepXDE model was systematically compared with variants that incorporate Residual-based Adaptive Refinement (RAR), L_2 conservation law, and different PDE loss weights to solve the Nonlinear Schrödinger Equation. The results demonstrate that RAR is the most significant contributor to accuracy of predictions, with the RAR-only setting, using PDE residual loss weight $w_f = 4$, producing the lowest relative ℓ_2 errors for solution h and its components u and v (7.566×10^{-4} , 9.538×10^{-4} , and 1.215×10^{-3} respectively) among all tested configurations (see Table 5). Although the L_2 conservation law loss function does not always minimize total error as the original, it provided meaningful physics-based regularization that helps keep the model within the constraints [9]. Evaluation of the performance of PINNsTF2-DeepXDE

models that incorporate RAR, $L2$, or both, indicated that adaptive refinement was the best method to improve predictive quality. Meanwhile, the most practically balanced strategy is achieved by combining refinement with physical constraints from $L2$ under appropriate loss weighting. Overall, the study confirms that PINN performance on nonlinear wave dynamics depends on the joint design of sampling strategy and loss composition. Future work should validate these conclusions across broader PDE families, incorporate multiple independent trials using various random seeds, reporting the results as mean values with their corresponding positive and negative standard deviations, and investigate adaptive or automated loss-balancing schemes for more robust training. Furthermore, one may include a targeted experiment to investigate how variance scales with specific hyperparameters, as existing research suggests [5] that while certain weights can improve accuracy, excessively large weights can lead to an unstable training process and higher deviations in predictive error.

References

- [1] M. Raissi, P. Perdikaris, and G. E. Karniadakis, “Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations,” *Journal of Computational Physics*, vol. 378, pp. 686–707, 2019. DOI: 10.1016/j.jcp.2018.10.045.
- [2] J. Guo, H. Wang, and C. Hou, *A novel adaptive causal sampling method for physics-informed neural networks*, 2022. arXiv: 2210.12914 [cs.LG]. [Online]. Available: <https://arxiv.org/abs/2210.12914>.
- [3] Z. K. Lawal, H. Yassin, D. T. C. Lai, and A. Che Idris, “Physics-informed neural network (pinn) evolution and beyond: A systematic literature review and bibliometric analysis,” *Big Data and Cognitive Computing*, vol. 6, no. 4, p. 140, 2022. DOI: 10.3390/bdcc6040140.
- [4] E. Haghighat, D. Amini, and R. Juanes, “Physics-informed neural network simulation of multiphase poroelasticity using stress-split sequential training,” *Computer Methods in Applied Mechanics and Engineering*, vol. 397, p. 115 141, 2022. DOI: 10.1016/j.cma.2022.115141.
- [5] S. Wang, X. Yu, and P. Perdikaris, *When and why pinns fail to train: A neural tangent kernel perspective*, 2020. arXiv: 2007.14527 [cs.LG]. [Online]. Available: <https://arxiv.org/abs/2007.14527>.
- [6] I. A. Moloshnikov et al., “Analysis of neural network methods for obtaining soliton solutions of the nonlinear schrödinger equation,” *Chaos, Solitons & Fractals*, vol. 192, p. 115 943, 2025. DOI: 10.1016/j.chaos.2024.115943.
- [7] S.-F. Tian, Y.-X. Yu, and B. Li, “A residual-based adaptive refinement physics-informed neural networks (rar-pinns) method for fifth-order kdv equation,” *Chinese Physics B*, 2025. DOI: 10.1088/1674-1056/ae1458.
- [8] L. Serkin and T. L. Belyaeva, “Physics-informed neural networks for higher-order nonlinear schrödinger equations: Soliton dynamics in external potentials,” *Mathematics*, vol. 13, no. 11, p. 1882, 2025. DOI: 10.3390/math13111882.

- [9] G.-Z. Wu, Y. Fang, Y.-Y. Wang, and C.-Q. Dai, *Modified physics-informed neural network method based on the conservation law constraint and its prediction of optical solitons*, 2021. arXiv: 2108.13192 [nlin.PS]. [Online]. Available: <https://arxiv.org/abs/2108.13192>.
- [10] A. D. Jagtap, E. Kharazmi, and G. E. Karniadakis, “Conservative physics-informed neural networks on discrete domains for conservation laws: Applications to forward and inverse problems,” *Computer Methods in Applied Mechanics and Engineering*, vol. 365, p. 113028, 2020. DOI: 10.1016/j.cma.2020.113028.
- [11] A. D. Jagtap and G. E. Karniadakis, “Extended physics-informed neural networks (xpinns): A generalized space-time domain decomposition based deep learning framework for nonlinear partial differential equations,” *Communications in Computational Physics*, vol. 28, no. 5, pp. 2002–2041, 2020. DOI: 10.4208/cicp.0A-2020-0164.
- [12] R. Bischof and M. Kraus, “Mixture-of-experts-ensemble meta-learning for physics-informed neural networks,” in *Forum Bauinformatik*, Sep. 2022. [Online]. Available: <https://mediatum.ub.tum.de/doc/1688403/uic8b0xn1c845e7rac1or092o.Bischof%20et%20A1.%202022.pdf>.
- [13] L. Lu, X. Meng, Z. Mao, and G. E. Karniadakis, “Deepxde: A deep learning library for solving differential equations,” *SIAM Review*, vol. 63, no. 1, pp. 208–228, 2021. DOI: 10.1137/19M1274067.
- [14] L. Biewald, *Experiment tracking with weights and biases*, 2020. [Online]. Available: <https://www.wandb.com/>.
- [15] I. Sobol’, “On the distribution of points in a cube and the approximate evaluation of integrals,” *USSR Computational Mathematics and Mathematical Physics*, vol. 7, no. 4, pp. 86–112, 1967. DOI: [https://doi.org/10.1016/0041-5553\(67\)90144-9](https://doi.org/10.1016/0041-5553(67)90144-9).
- [16] R. Akbarian Bafghi and M. Raissi, *Pinns-tf2: Fast and user-friendly physics-informed neural networks in tensorflow v2*, 2023. arXiv: 2311.03626 [cs.CE]. [Online]. Available: <https://arxiv.org/abs/2311.03626>.
- [17] L. Zhang, M. Du, X. Bai, Y. Chen, and D. Zhang, “Complex-valued physics-informed machine learning for efficient solving of quintic nonlinear schrödinger equations,” *Physical Review Research*, vol. 7, p. 013164, 1 2025. DOI: 10.1103/PhysRevResearch.7.013164.