Reinforcement Learning for Efficient Identification of Soliton System Parameters Across Expansive Domains

Cheng Hu[1,2], Zhiyang Zhang[1], Muwei Liu[1], Liuyu Xiang[2], Huijia Wu[2], Wenjun Liu[1,3*], and Zhaofeng He[2*]

[1]*State Key Laboratory of Information Photonics and Optical Communications, School of Science, Beijing University of Posts and Telecommunications, Beijing 100876, China*
[2]*School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing 100876, China*
[3]*Beijing Laser Creation Optoelectronics Technology Company Limited, Beijing 101400, China*

(Received 16 October 2024; accepted manuscript online 8 November 2024)

Optical solitons in mode-locked fiber lasers and optical communication links have various applications. The study of transmission modes of optical solitons necessitates the investigation of the relationship between the equation parameters and soliton evolution employing deep learning techniques. However, the existing identification models exhibit a limited parameter domain search range and are significantly influenced by initialization. Consequently, they often result in divergence toward incorrect parameter values. This study harnessed reinforcement learning to revamp the iterative process of the parameter identification model. By developing a two-stage optimization strategy, the model could conduct an accurate parameter search across arbitrary domains. The investigation involved several experiments on various standard and higher-order equations, illustrating that the innovative model overcame the impact of initialization on the parameter search, and the identified parameters are guided toward their correct values. The enhanced model markedly improves the experimental efficiency and holds significant promise for advancing the research of soliton propagation dynamics and addressing intricate scenarios.

DOI: 10.1088/0256-307X/41/12/120303

1. Introduction. Optical solitons are a special type of nonlinear localized wave packets that are formed when the dispersion and nonlinear effects reach an equilibrium during the propagation of pulses in a nonlinear medium.[1–7] The invariant characteristics of shape, velocity, and amplitude are maintained during the propagation of pulses.[8–12] Ultrafast mode-locked lasers can generate various transient soliton phenomena across different dispersion regions, including soliton molecules, optical rogue waves, and soliton explosions.[13,14] In optical communication systems, the optical soliton has been extensively utilized to maintain the integrity of the long-distance information transmission.[15–17]

Optical solitons are not inherently unpredictable; however, their formation is dependent on the nonlinear variation of the refractive index in optical materials owing to the intensity of light.[18] Through technical derivation, optical solitons can be described as specific solutions to the nonlinear Schrödinger (NLS) equation under the balance of dispersion and nonlinear effects.[19] Consequently, they are significantly affected by the interplay of dispersion and nonlinear parameters.[20] The dispersion and nonlinear parameters of the NLS equation can be manipulated to control the interactions between solitons, thereby enabling the stable transmission of multiple solitons and expanding the transmission channels for optical solitons.[21,22] However, the evolution of the solitons reflects the characteristics of the equation. Dark and antidark solitons are observable in the normal dispersion region. Further, by inducing appropriate parameters, breathers with oscillating periods can be obtained.[22–24]

The methodology of investigating the relationship between the evolution of solitons and the parameters of the equations is referred to as parameter identification. Currently, parameter identification typically employs oscilloscope measurements coupled with functional fitting.[25] The commonly used modulation phase shift method uses a fiber of known length and measures the phase shifts under various modulation intensity pulses. This is followed by the use of specific functions to fit and determine the dispersion coefficient of fiber.[26] However, these methods are often significantly affected by equipment accuracy and suffer from long experimental cycles. Thus, they are unsuitable for modern measurement environments characterized by short measurement windows and large data volumes.[27] With the enhancement of computational capabilities, data-driven machine learning, particularly deep learning, has progressively garnered increased research interest.[28] These end-to-end models can extract more profound features from the data. In contrast to conventional approaches, deep learning models, particularly physics-informed neural network (PINN), can significantly enhance the speed of recognition while maintaining accuracy. Consequently, they have emerged as a

*Corresponding author. Email: jungliu@bupt.edu.cn; zhaofenghe@bupt.edu.cn
© 2024 Chinese Physical Society and IOP Publishing Ltd

formidable method for the addressal of parameter identification challenges.[29]

However, parameter identification using a PINN is plagued by the problem of excessively small parameter search space of the PINN and not the precision of the identification. Consider an example involving the NLS equation [Eqs. (1)], where the parameters β and γ that are to be identified represent the dispersion and nonlinear effects, respectively, based on the soliton solution under specific conditions ($\beta = 1, \gamma = 2$). Herein, the soliton solution is employed to identify the dispersion and nonlinear coefficients of the equation. The random selection of coefficients within the range of $[-10, 10]$ causes the identification process of the PINN to diverge, resulting in failure. PINN can successfully converge on the correct results only when the identification parameters are initialized very close to the true values, such as $\beta = 0$ and $\gamma = 1$. The process of identification is presented in Fig. 1.

$$\begin{aligned} &\mathrm{i}\frac{\partial u}{\partial t} - \beta u_{xx} + \gamma|u|^2 u = 0, \\ &u = 0.6\mathrm{sech}(0.6x)e^{\mathrm{i}0.36t+\mathrm{i}}(\beta = 1, \gamma = 2). \end{aligned} \tag{1}$$

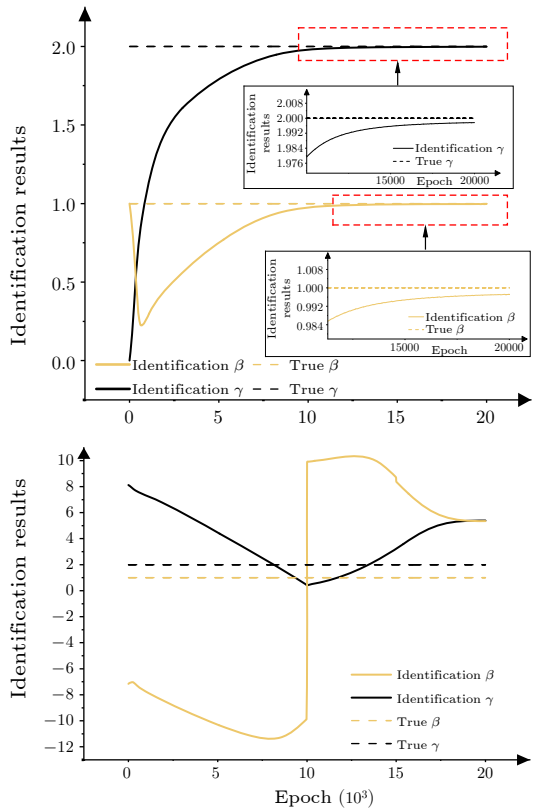

Fig. 1. Parameter identification of Eqs. (1) is conducted employing the soliton solution. The orange color represents the dispersion coefficient and the black denotes the nonlinear coefficient. The solid lines depict the parameter identification during the training process and the dashed lines indicate the actual values. Further, the horizontal axis represents the number of training iterations.

Such instances of identification failure owing to the initialization of the parameters to be identified are quite common.[30] However, in most cases, there exists only a vague understanding of the parameter range at the initial stage of training.[31] Consequently, numerous repetitive experiments are required to probe the parameters, thereby rendering successful identification challenging.[32] Consequently, the development of deep learning for the identification of soliton parameters is hindered and the application of this technology in real-world scientific research remains limited.[33,34]

This study addressed the aforementioned issue. Through the integration of reinforcement learning into the iterative process of PINNs, the parameter search space of PINNs was exponentially expanded. The proposed approach divided parameter identification into two steps. The first step involved the use of Q-learning,[35] a reinforcement learning technique, to segment the parameter search space and consequently arrange it in an optimal manner. The second step involved the application of the PINN algorithm within the optimally segmented space to obtain the identified parameters. Because this method is not dependent on the initial parameter selection, it can successfully identify parameters within any arbitrarily chosen search space. The proposed method is referred to as the deep q-learning physics-informed neural network (DQ-PINN).

The DQ-PINN approach did not prolong the computational time. In fact, when considering the time required for the original PINN to undergo multiple adjustments of the initial values, DQ-PINN significantly reduced the overall computation time. However, owing to DQ-PINN being a completely two-stage model, the computation in the second stage was solely reliant on the outcomes of the first stage. Consequently, the DQ-PINN constituted a plug-and-play model architecture that was compatible with any enhanced PINN structure.

This study first elucidated the model architecture and technical components of the DQ-PINN, followed by parameter identification and soliton evolution prediction for several standard NLS equations. Finally, the application scenarios of the DQ-PINN were extended to several complex equations. The contributions advanced by this study are expected to augment the efficiency of optical soliton system identification and further enhance the existing comprehension of soliton dynamical behaviors.

2. Method. Here, we elucidate the technical details of the DQ-PINN. First, the training process of the origin PINN is described.[36] The NLS equation is expressed as:

$$\mathscr{H}(u(x, t), \boldsymbol{\phi}) = 0, \tag{2}$$

where $\mathscr{H}$ is a continuous linear operator, $u(x, t) \in C^{\infty}$ is the solution of the NLS equation, and $\boldsymbol{\phi} \in \mathbb{R}$ is the identified parameter. We set the neural network of PINN as $\boldsymbol{W}$, and the result obtained from PINN is $u(x, t, \boldsymbol{W})$. As the calculation mode of a neural network requires discrete data, the continuous coordinates (x, t) are discretized into the $\{x_i, t_i\}_{i=1}^{N_c}$. The $\{x_i, t_i\}_1^{N_c}$ is referred to as the collocation dataset, and N_c is the number of samplings. To ensure that the iterations of the PINN proceeded along the correct direction, a forced dataset denoted as $\{x_i, t_i, u_i\}_{i=1}^{N_f}$

is introduced. The samples (x_i, t_i) from the forced dataset are obtained by sampling from continuous coordinates, and the sampling method may differ from that of the collocation dataset. Further, u_i is the true data corresponding to the sampled coordinates, and N_f indicates the number of samples.[37,38] With these considerations, we obtain the loss function for the PINN as follows:

$$\begin{aligned}
&\sum_{i=1}^{N_\mathrm{c}} |\mathrm{Re}(\mathscr{H}(x_i,\, t_i,\, u(x_i,\, t_i,\, \boldsymbol{W}), \boldsymbol{\phi}))| \\
&= 0, (x_i,\, t_i) \in \{x_i, t_i\}_{i=1}^{N_\mathrm{c}}, \\
&\sum_{i=1}^{N_\mathrm{c}} |\mathrm{Im}(\mathscr{H}(x_i,\, t_i,\, u(x_i,\, t_i,\, \boldsymbol{W}), \boldsymbol{\phi}))| \\
&= 0, (x_i,\, t_i) \in \{x_i, t_i\}_{i=1}^{N_\mathrm{c}}, \\
&\sum_{i=1}^{N_\mathrm{f}} (u(x_i,\, t_i,\, \boldsymbol{W}) - u_i)^2 \\
&= 0, (x_i,\, t_i) \in \{x_i, t_i, u_i\}_{1}^{N_\mathrm{f}}.
\end{aligned} \tag{3}$$

2.1. Solitons Prediction. $\boldsymbol{\phi}$ was encoded as the trainable weight of the neural network and was optimized in conjunction with other weights following the gradient descent method. The loss function was divided into two parts through different datasets. One part facilitated the network in learning the physical structure of the operator through the collocation dataset, which constituted the physical information component in the PINN. The other part restricted the iterative direction of the network on the manifold through the forced dataset. This ensured that the correct parameters were learned. However, upon completion of the training, the identified parameters were obtained and the evolution of the solution at different coordinates was also revealed. Therefore, we contended that the PINN effectively reconstructed the entire equation, and this reconstruction was mesh-free.

2.2. Deep Q-Network (DQN)-based Reinforcement Learning. PINN optimized the iterative process via the introduction of reinforcement learning to obtain optimization strategies. This innovative synthesis facilitated a groundbreaking method for control across diverse disciplines by leveraging both data-driven discoveries and fundamental physical laws. The utility of physical informed reinforcement learning spans various applications, including intricate flow-field reconstruction, detailed musculoskeletal simulations,[39] and the enhancement of power system stability,[40] in addition to other applications.

The problem of searching for superior parameters within a defined parameter search space can be formulated as a Markov decision process that comprises a state space, action space, state transition mechanism, and reward system. The efficient resolution of this process can be achieved via the integration of the Q-learning algorithms with the deep neural network technology.[41] Specifically, we estimated the Q-value function $Q_\theta(S,\, a)$ by training a neural network governed by the parameters θ, which accepted the state S as the input and output the corresponding Q-values for all potential actions a. These Q-values are the expected cumulative rewards for the execution of different actions. This approach was based on the findings from previous research in the field of reinforcement learning, with the aim of facilitating the trained network Q_θ in effectively predicting the "most suitable" subsequent actions.[42] In the proposed approach, the essential elements of the DQN-based learning procedure were outlined as follows:

State Space: The state space was a four-dimensional vector; specifically, it included the values of the parameters from the previous time step, u_pred at the current time step, and v_pred at the current time step. As the values of u_pred and v_pred are multidimensional, a simple neural network was employed to convert each of them into a scalar, which was then incorporated as a part of the state space.

Action Space: The action space comprised integers within a predefined parameter search range, that is, integers between $[-10,\, 10]$, resulting in a state space size of 21. Each parameter had the same action space.

Transition function: The transition function defined the process through which the current state transitioned to the subsequent state. This process involved the calculation of the next state by substituting the parameters obtained from the search at the current time step into the equations.

Reward function: The reward function evaluated the quality of the executed actions, which was the accuracy of the searched parameters. The closer the parameter values were to the true values, the greater the reward. The reward function used to guide the learning process is expressed as:

$$R(s_t,\, a_t) = \begin{cases} -10, & \text{if all the searched parameters are zero,} \\ -\mathscr{H}(u(x,\, t), \boldsymbol{\phi}) & \text{otherwise.} \end{cases} \tag{4}$$

Using a proficiently trained network Q_θ, the action chosen to optimize the functional value concurrently maximized the overall cumulative reward.

Now, we introduce the learning process of the DQN algorithm in obtaining the optimal policy Q_θ. First, we initialized two networks, Q_θ and $\hat{Q}_\theta$, where $\hat{Q}_\theta$ was set to be equal to Q_θ. At the beginning, the target network $\hat{Q}_\theta$ was identical to the original Q_θ network. Beginning with an initial state s_t, during the interaction process, an action a_t was undertaken based on this state S_t. Subsequently, a reward r was received, followed by a transition to the subsequent state S_{t+1}. This formed a piece of data $(s_t,\, a_t,\, r,\, s_{t+1})$, which was stored in the replay buffer. If the replay buffer became full, certain of the older data were

discarded. Subsequently, data were sampled from the replay buffer to train the neural network Q_θ for estimating Q-values that satisfied the Bellman equation as

$$y = r_t + \gamma \max_{a^*} \hat{Q}(s_{t+1},\, a^*). \tag{5}$$

This equation calculates the target Q-value y with $\gamma \in (0,\, 1)$ as the discount factor. The network Q_θ was optimized by performing gradient descent steps to minimize the loss as

$$L = (y - Q(s_t,\, a_t))^2, \tag{6}$$

$$\theta \leftarrow \theta - \alpha \nabla_\theta L(\theta). \tag{7}$$

The parameter α was the learning rate. The weights of the network θ were updated to minimize this loss and improve the Q-value estimates. The target network $\hat{Q}_\theta$ in the Bellman equation t was only updated by Q_θ at regular intervals (e.g., every 10 steps) to address the challenge of instability and divergence in training. The pseudocode for the training phase of the algorithm is formulated as the first stage of Algorithm 1. The model flowchart is illustrated in Fig. 2.

Algorithm 1 DQ-PINN

Stage 1:
Initialize replay buffer $\mathcal{D}$
Initialize original network $Q(s,\, a; \theta)$ and target network $\hat{Q}(s',\, a'; \theta')$ with random weights
for each episode **do**
    Initialize state s_1
    for each step t in episode **do**
        With probability ϵ, select random action a_t
        Otherwise, select $a_t = \arg\max_a Q(s_t,\, a; \theta)$
        Execute action a_t
        Observe reward r_t and next state s_{t+1}
        Generate noisy human input $\hat{h}_t$:
        if With probability $1 - p$ **then**
            set $\hat{h}_t = h_t^{\text{ideal}}$
        else
            set $\hat{h}_t = \text{Flip}(h_t^{\text{ideal}})$
        end if
        Store transition $(s_t,\, a_t,\, r_t,\, s_{t+1})$
        Sample mini-batch of transitions $(s_j,\, a_j,\, r_j,\, s_{j+1})$
        Compute target $y_j = r_j + \gamma \max_{a'} \hat{Q}(s_{j+1},\, a'; \theta')$
        Perform gradient descent step on $(y_j - Q(s_j,\, a_j; \theta))^2$
        if step mod target update frequency == 0 **then**
            Update target network: $\theta' \leftarrow \theta$
        end if
        Update state $s_t \leftarrow s_{t+1}$
    end for
end for

Stage 2:
Initialize dataset
Initialize CNN model with random weights
Define loss_function and optimizer
for each epoch **do**
    Initialize loss = 0
    for each data (input, label) **do**
        Forward pass: Compute prediction = CNN(input)
        Compute loss = loss_function(prediction, label)
        Backward pass: optimizer.zero_grad()
        Compute gradients: loss.backward()
        Update model weights: optimizer.step()
    end for
end for

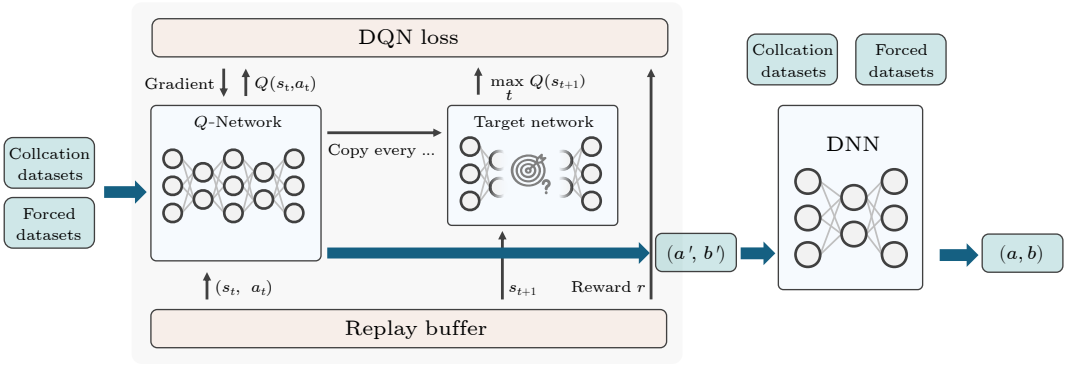

Fig. 2. DQ-PINN model flowchart. This process includes the parameter identification process under two-stage learning.

3. Basic Experiments. Comparative experiments between PINN and DQ-PINN were conducted to demonstrate the capability of the proposed method to accurately identify parameters over a broad search range. Therefore, two scenarios were set up for the original PINN. In the first scenario, the identified parameters were randomly initialized within the fixed range of $[-10, 10]$. In the second scenario, the identified parameters were initialized in the vicinity of the true values. These two scenarios were set in a manner that illustrated that the search domain of the original PINN was considerably narrow and unable to locate the correct parameters within a large initialization area. The model in the first scenario is referred to as "PINN-far", whereas the second is denoted as "PINN-close". Moreover, we compared the accuracy of parameter identification and the ability to reconstruct solitons, which was the prediction accuracy of the evolution of the soliton system.

The fundamental experiments were divided into two parts. The first part involved the use of the DQ-PINN and the original PINN for parameter identification in various normal NLS equations to validate their performance. The second part was focused on the assessment of their capability to predict the evolution of solitons. The errors in the parameter identification and the prediction errors of soliton evolution were both expressed using the following equation:

$$\text{Error} = \frac{1}{\text{Number}} \sum \frac{|\text{data}_{\text{predict}} - \text{data}_{\text{true}}|}{\text{data}_{\text{true}}}. \quad (8)$$

3.1. Parameter Identification. The equations utilized in the experiment are presented in Table 1. β denotes the dispersion coefficient and γ is the nonlinear coefficient; both these parameters were to be identified. To elucidate the objectivity of the experiment, the characteristics inherent to the experimental setup are delineated as follows.

Regardless of it being PINN-far, PINN-close, or DQ-PINN, the datasets, learning rates, optimizers, and initialization methods for the neural network weights were identical for all three approaches. This ensured a fair comparison of the outcomes within an equitable environment. The collocation dataset comprised coordinates (x, t) uniformly sampled at 40000 points from the interval $[0, 5] \times [-2.5, 2.5]$. Further, the forced dataset comprised 20000 uniformly sampled coordinates and function values from the same range. Multiple hyperparameters were systematically evaluated to ensure optimal performance across different experimental scenarios. Specifically, we tested various learning rates (e.g., 0.001, 0.0005, 0.0001, and 0.00005) and different training epochs (e.g., 10000, 20000, and 25000). As evident, higher learning rates, while accelerating convergence, tended to weaken the generalization performance on the test set, thereby resulting in overfitting. Conversely, lower learning rates decreased the convergence speed, thereby reducing the training efficiency. In terms of the number of training epochs, models with fewer than 20000 epochs exhibited insufficient convergence, and particularly struggled with accuracy in complex parameter search tasks. Although extending the training to 25000 epochs introduced additional computational costs, no significant improvements in the results were observed. Based on these findings, the learning rate was set at 0.0001, the train epoch was 20000, the Adam optimizer was employed, and the weights of the neural network were initialized using the Kaiming initialization method.

Table 1. Different fundamental soliton systems and the different dispersion and nonlinear parameters.

Name	Basic NLS equations	β	γ
NLS equation-1	$\mathrm{i}\frac{\partial u}{\partial t} - u_{\boldsymbol{xx}} + 2\lvert u\rvert^2 u = 0$	1	2
NLS equation-2	$\mathrm{i}\frac{\partial u}{\partial t} - u_{\boldsymbol{xx}} + \lvert u\rvert^2 u = 0$	1	1
NLS equation-3	$\mathrm{i}\frac{\partial u}{\partial t} - 0.5u_{\boldsymbol{xx}} + \lvert u\rvert^2 u = 0$	0.5	1

To facilitate a more accurate comparison of the predictive accuracy of different models for soliton evolution, the test coordinate point dataset for the evolution of the soliton solutions of the equation was obtained based on a sampling step size that differed from the training dataset. The sampling strategy involved the uniform collection of 5000 coordinate points from the interval $[0, 5] \times [-2.5, 5.5]$. Finally, we introduced Gaussian noise at levels of 1%, 5%, and 10% into the dataset to assess the robustness of the DQ-PINN model against anomalous environmental conditions. Table 2 presents the parameter identification results for various equations using different models. The PINN-far demonstrated divergence in parameter identification for all equations, whereas the PINN-close achieved successful identification in almost all cases. Thus, the success of pa-

rameter identification using PINN was highly dependent on the selection of the initial parameters. In case of the initial values chosen being close to the true values, the PINN could successfully identify the parameters; whereas, if the initial parameters were very far from the true values, the identification process failed.

Table 2. NLS equation parameter identification absolute error. "With 1%Noise" indicates 1% of Gaussian noise is led into the forced point dataset. "With 5%Noise" indicates 5% of Gaussian noise is led into the forced points dataset. "With 10%Noise" indicates 10% of Gaussian noise is led into the forced points dataset.

Case	**NLS equation-3**		**NLS equation-2**		**NLS equation-3**	
Parameters	β	γ	β	γ	β	γ
Without noise						
PINN-far	**434.2%**	**652.2%**	**339.7%**	**821.5%**	**557.9%**	**813.6%**
PINN-close	0.915%	0.423%	0.321%	0.0762%	0.0988%	0.0338%
DQ-PINN	0.533%	0.671%	0.171%	0.0559%	0.0971%	0.0208%
***Noise* 1%**						
PINN-far	**762.4%**	**311.5%**	**631.3%**	**305.4%**	**7.905%**	**3.643%**
PINN-close	1.37%	0.409%	1.48%	0.455%	0.198%	0.177%
DQ-PINN	0.452%	0.556%	0.931%	0.0385%	0.245%	0.0324%
***Noise* 5%**						
PINN-far	**461.8%**	**643.4%**	**75.1%**	**16.5%**	**497.5%**	**97.8%**
PINN-close	0.971%	0.739%	1.93%	0.653%	0.237%	0.898%
DQ-PINN	0.164%	0.01%	0.189%	0.0356%	1.367%	0.0453%
***Noise* 10%**						
PINN-far	**524.1%**	**432.5%**	**210.4%**	**701.5%**	**907.1%**	**616.2%**
PINN-close	0.904%	0.982%	2.31%	0.726%	0.308%	1.091%
DQ-PINN	1.723%	0.511%	0.545%	0.0259%	1.937%	0.0601%

The DQ-PINN successfully identified the parameters in every scenario, and the precision of the identification was comparable to that of the PINN-close. Thus, the new method transcended the limitations of the original PINN by expanding the search scope to the entire parameter space $[-10, 10]$.

3.2. Solitons Prediction. Figure 3 shows the predictions and errors of various models on soliton evolution in a noise-free environment. As evident, the PINN-far model exhibited prediction errors that were approximately two orders of magnitude higher than those of the other models, which was attributable to significant errors in parameter identification. In addition, the DQ-PINN and PINN-close models exhibited nearly identical prediction accuracy. This demonstrated that the novel method did not suffer from a loss of precision in the prediction across the entire search domain, even without the initialization of the identified parameters.

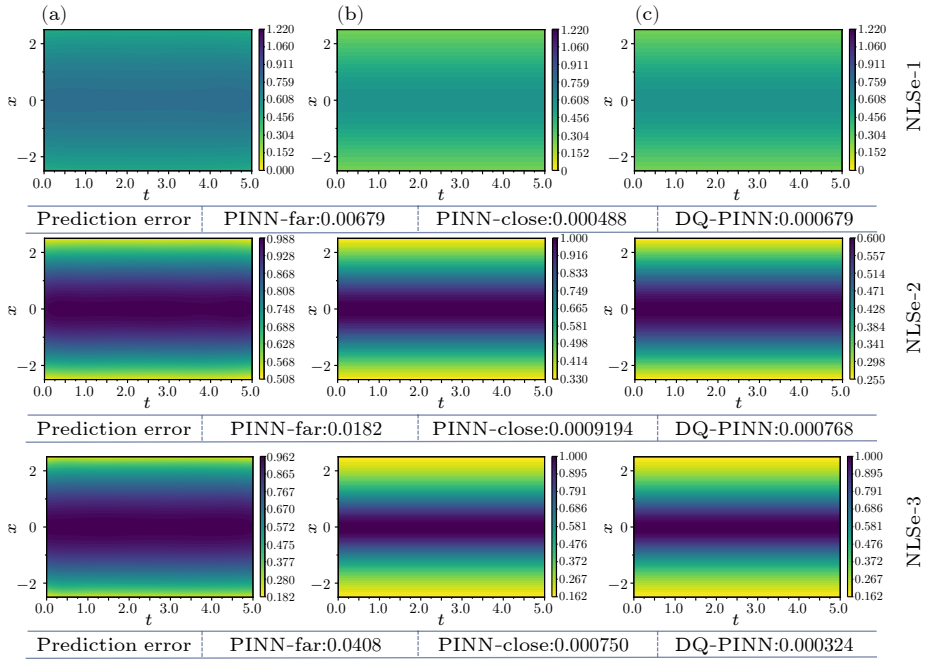

Fig. 3. Evolution images and errors of the solitons for various NLS equations in a no-noise environment depicted using different models. Column A presents the prediction outcomes of PINN-far, column B presents the prediction results of PINN-close, and column C presents the prediction outcomes of DQ-PINN.

These experiments served several purposes. We confirmed that the proposed method overcame the limitations of the original PINN in the parameter search space, thereby requiring only the establishment of an arbitrary search region for the DQ-PINN to autonomously accomplish parameter identification. Furthermore, the DQ-PINN maintained precision in both parameter identification and subsequent accuracy prediction and could even enhance performance. Finally, the DQ-PINN demonstrated robust noise resistance and was capable of effectively completing identification tasks even in varying noise conditions. Subsequently, we extended the application of the DQ-PINN to two types of higher order equations that more accurately described the transmission modes of optical solitons in the real world.

4. Identify Complex Equation.

4.1. Gerdjikov–Ivanov Equation. Considering the influence of higher order perturbations, the normal NLS equation can be generalized to the Gerdjikov–Ivanov (GI) equation:[43]

$$\mathrm{i}u_x + u_{tt} - \mathrm{i}\beta u^2 u_t^* + \gamma|u|^4 u = 0, \tag{9}$$

where u is a complex potential function, u^* is the complex conjugate of u, and x and t are the spatial and temporal variables, respectively. Further, β and γ are the real-valued constants representing nonlinear dispersion and quintic nonlinearity, respectively. The GI equation accepts rare solitary wave solutions with a zero background, which can be transformed into each other through gauge transformations using the Kaup–Newell (KN) and Chen–Lee–Liu (CLL) equations, and manifest in various physical contexts.[44,45]

Table 3. Parameter identification results using GI and KE equations by DQ-PINN.

Equation	Parameter α	Parameter β	Parameter γ
GI equation	–	0.545%	0.417%
KE equation	0.00119%	0.1894%	0.516%

The GI equation has been utilized to model the transmission of solitary waves in nonlinear fibers under higher order perturbations, thereby resulting in extensive applications within fiber-optic communication systems.[46] In recent years, exact solutions to the GI equation have been obtained employing the Darboux transformation,[47] and the bifurcation theory for the GI equation has been presented in Ref. [48]. The coupled form of the GI equation along with its n-soliton solutions has been reported in Ref. [49]. The solitary wave solutions of the GI equation employed in the experiments are as follows:

$$u = \frac{4\mathrm{i}\tau e^{\mathrm{i}\upsilon} e^{2\mathrm{i}\tau^2(t-2\tau^2 x)}}{\mathrm{i} + 4\tau^2(t-4\tau^2 x)} (\beta = 1,\ \gamma = 0.5), \tag{10}$$

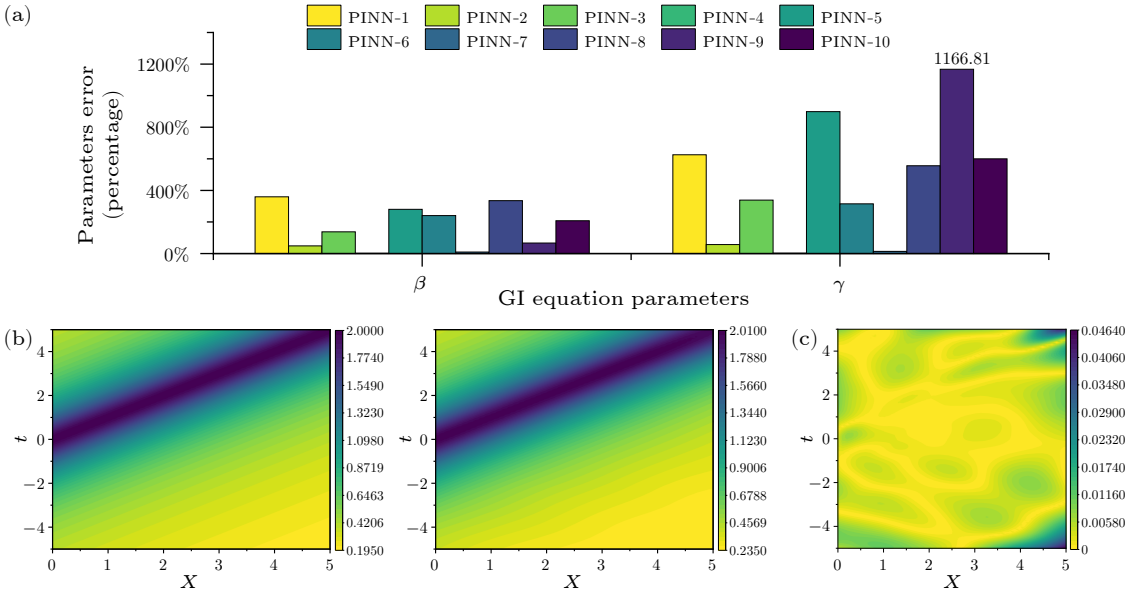

Fig. 4. Identification and predictive outcomes of various models for the GI equation. Subfigure (a) shows the parameter identification errors of 10 distinct PINNs for the GI equation. Subfigure (b) illustrates the true solutions of the solitons along with the evolutionary outcomes of the DQ-PINN for soliton dynamics. Subfigure (c) shows the evolutionary errors of the DQ-PINN.

We established 10 PINNs with their parameters randomly initialized within the range of $[-20, 20]$, which are designated as "PINN-1" through "PINN-10." The parameter search range for the DQ-PINN was also set to $[-20, 20]$. Each model was subjected to 20000 training epochs. Table 3 presents the identification error results of the DQ-PINN for the GI equation, and Fig. 4(a) shows the corresponding identification errors for the 10 PINNs. As evident, the majority of the original PINN models exhibited identification errors that exceeded 100%, indicating a lack of convergence. Whereas, the DQ-PINN yielded identification errors of 0.545% and 0.417%, thereby demon-

strating its robust capability in handling complex scenarios. Figure 4(b) illustrates the true solitary wave solution of the GI equation along with the solitary wave solution evolved by the DQ-PINN. Figure 4(c) shows the absolute error associated with these solutions. Notably, we did not provide the solitary wave evolution results for the 10 PINN models; this is because these models barely converged in parameter identification. Thus, their predictive outcomes merely represented interpolative fits to the forced dataset, thereby lacking a referential value.

4.2. Kundu–Eckhaus Equation. For an accurate description of the propagation process of the fiber optic soliton pulses, we incorporated high-order nonlinear terms and constructed the Kundu–Eckhaus (KE) equation, which is formulated as follows:

$$\mathrm{i}u_x + \alpha u_{tt} + \gamma|u|^2u + 4\beta^2|u|^4u - 4\mathrm{i}\beta|u|_t^2u = 0. \tag{11}$$

This equation can be applied to investigate the dynamics of optical solitons in nanomaterial and polarization-maintaining fibers. Parasuraman E at 2021 [50] reported bright and dark soliton solutions obtained using the KE model under birefringent fibers in the absence of four-wave mixing effects. The dynamics of singular solitons under the effect of both four-wave mixing and intermodal dispersion have also been studied. [51] Notably, the algebraic structure of nonlinear local chirped solitons in the KE equation has been reported based on the widespread application of chirped local structures in fiber amplifiers and all-optical communication links. [52] In the equation, α, β, and γ represent the parameters that are to be identified. Further, γ represents the non-Kerr nonlinear effect, and the β represents the Raman effect, which facilitates the self-frequency shift of the waves. The specific soliton solutions are provided in the following equations:

$$\begin{aligned} u =& -2\eta_1 e^{(\mathrm{i}f(x,\,t))}\mathrm{sech}(2\eta_1 t + 8\xi_1\eta_1 x - \varrho_1),\\ f(x,\,t) =& -2\xi_1 t - 4(\xi_1^2 - \eta_1^2)x - \eta_2 + 2\eta_3\\ &+ 4\eta_1 e^{(2\eta_1 t + 8\xi_1\eta_1 x - \varrho_1)}\mathrm{sech}(2\eta_1 t + 8\xi_1\eta_1 x - \varrho_1). \end{aligned} \tag{12}$$

Similar to the experimental process employed in case of the GI equation, we constructed 10 PINNs whose identification parameters were randomly initialized within the range of $[-20,\,20]$. The collocation datasets for both the DQ-PINN and the original PINN models comprised 20000 points sampled on the interval $[2,\,8]\times[-5,\,5]$. Whereas, the forced dataset comprised 15000 points sampled on the same interval. All models were trained under identical conditions. The coefficients in Eq. (12) were set as $\alpha = 1$, $\beta = 1$, $\gamma = 2$, $\eta_1 = 0.2$, $\xi_1 = 0.5$, $\varrho_1 = 0.3$, $\eta_2 = 0.6$, and $\eta_3 = 0.4$.

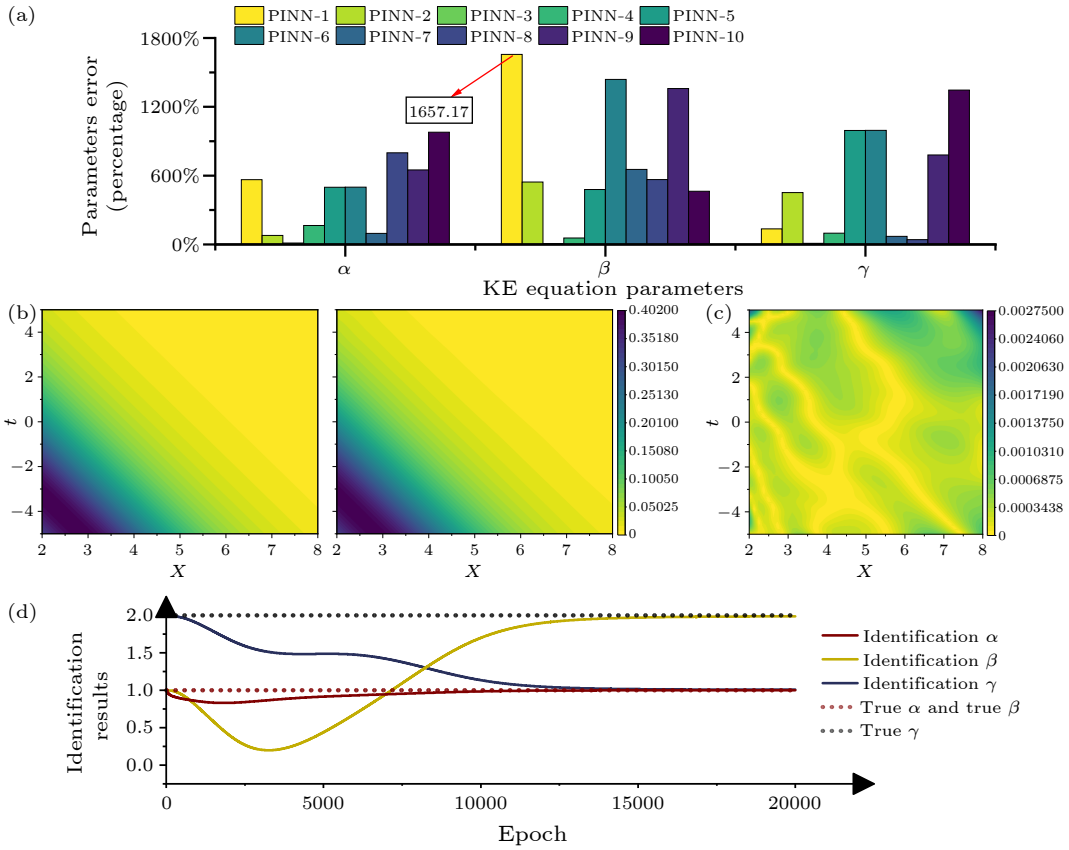

Fig. 5. Identification and predictive outcomes of various models using the KE equation. Subfigure (a) depicts the parameter identification errors of 10 distinct PINNs for the GI equation. Subfigure (b) illustrates the true solutions of the solitons along with the evolutionary outcomes of the DQ-PINN for soliton dynamics. Subfigure (c) shows the evolutionary errors of the DQ-PINN. Subfigure (d) shows the iterative process of the parameters in the KE equation by DQ-PINN.

The different PINN models are denoted as "PINN-1" through "PINN-10." As evident, even for relatively complex equations, the DQ-PINN achieved accurate identification results, as presented in Table 3. Compared to the identification results of the 10 different PINN models, the DQ-PINN demonstrated convergence over a large parameter domain. In contrast, the parameter identification error of the original PINN reached a maximum of 1657.17, as shown in Fig. 5(a). The evolution of the soliton solutions using the DQ-PINN is illustrated in Fig. 5. Figure 5(b) shows the true and predicted solutions by the DQ-PINN, and Fig. 5(c) presents the error between the two.

5. Conclusion. Theoretical research on optical soliton systems is crucial for the development of mode-locked fiber lasers, precision optical component fabrication, and advancement of all-optical communication links. However, the existing methods for identifying soliton systems are unsuitable for practical research environments. This is attributed to the narrow parameter search domain of existing PINN methods coupled with a vague understanding of the parameters. This often prevents PINNs from converging to the correct parameters.

This study addressed this issue by introducing Q-learning, which is a reinforcement learning technique, and adopting a two-stage identification methodology to reconstruct the parameter identification process. The first stage involved the segmentation of the parameter space and conduction of a search optimization. This prevented the PINN from diverging within the parameter search domain. The proposed method, DQ-PINN, obviated the need for parameter initialization and instead facilitated the specification of an arbitrary search range. This approach required only a general understanding of the system to be identified, thereby facilitating the use of DQ-PINN for system identification.

Table 4. Errors of solitons prediction results for GI and KE equations using DQ-PINN.

Equation	GI equation	KE equation
Error	0.00469%	0.00311%

We conducted several experiments to validate the performance of the proposed DQ-PINN. Fundamental experiments demonstrated the precision of the new method in identification and its resilience to abnormal environments. Moreover, they highlighted the limitations of the original PINN. Furthermore, we extended the equations by incorporating additional nonlinear terms to construct the GI and KE equations. In addition, identification was performed on their soliton solutions. The identification results indicated that the DQ-PINN could function effectively in complex scenarios, and the errors of the identification results are presented in Table 4.

In the current experiment, the DQ-PINN model was primarily applied to relatively simple physical systems; however, it may be plagued by issues such as training instability or slow convergence when applied to more complex systems. This is because the nonlinearity and high dimensionality of complex systems necessitates that the model exhibit enhanced adaptability and stability. Further, the fully connected layers used in the current DQ-PINN model may lack sufficient generalization capability for handling high-dimensional data or complex spatial features. This would render the effective capture of multi-scale characteristics challenging, thereby affecting performance. In future work, we plan to improve the convergence efficiency of the model by incorporating more sophisticated physical constraints or appropriate regularization techniques. In addition, we aim to integrate more advanced network structures, such as convolutional neural networks (CNNs) or autoencoders, to enhance feature extraction capabilities and consequently, improve the performance of the model in complex physical systems.

Acknowledgements. This research was supported by the National Key Research and Development Program of China (Grant No. 2022YFA1604200), Beijing Municipal Science and Technology Commission, Administrative Commission of Zhongguancun Science Park (Grant No. Z231100006623006).

References

[1] Zabusky N J and Kruskal M D 1965 *Phys. Rev. Lett.* **15** 240
[2] Lederer F, Stegeman G I, Christodoulides D N, Assanto G, Segev M, and Silberberg Y 2008 *Phys. Rep.* **463** 1
[3] Khaykovich L, Schreck F, Ferrari G, Bourdel T, Cubizolles J, Carr L D, Castin Y, and Salomon C 2002 *Science* **296** 1290
[4] Yi H H, Li X F, Zhang J L, Zhang X, and Ma G L 2024 *Chin. Phys. Lett.* **41** 074204
[5] Yi H H, Zhang X, Shi L X, Yao Y L, Wang S B, and Ma G L 2024 *Chin. Phys. Lett.* **41** 044204
[6] Yao Y L, Yi H H, Zhang X, and Ma G L 2023 *Chin. Phys. Lett.* **40** 100503
[7] Zhang X Y and Wu Y 2023 *Chin. Phys. Lett.* **40** 080502
[8] Denschlag J, Simsarian J E, Feder D L, Clark C W, Collins L A, Cubizolles J, Deng L, Hagley E W, Helmerson K, Reinhardt W P, Rolston S L, Schneider B I, and Phillips W D 2000 *Science* **287** 97
[9] Burger S, Bongs K, Dettmer S, Ertmer W, Sengstock K, Sanpera A, Shlyapnikov G V, and Lewenstein M 1999 *Phys. Rev. Lett.* **83** 5198
[10] Liu X M, Zhang Z Y, and Liu W J 2023 *Chin. Phys. Lett.* **40** 070501
[11] Cui W W, Xing X W, Chen Y Q, Xiao Y J, Ye H, and Liu W J 2023 *Chin. Phys. Lett.* **40** 024201
[12] Chen Y Q, Wu Z T, Xiao P Y, Xiao W D, and Liu W J 2024 *Opt. Lett.* **49** 2437
[13] Heidemann R, Zhdanov S, Sütterlin R, Thomas H M, and Morfill G E 2009 *Phys. Rev. Lett.* **102** 135002
[14] Qi J W, Li Z D, Yang Z Y, and Yang W L 2017 *Phys. Lett. A* **381** 1874
[15] Xing X, Liu Y, Han J, Liu W, and Wei Z 2023 *ACS Photonics* **10** 2264
[16] Liu M, Wu H, Liu X, Wang Y, Lei M, Liu W, Guo W, and Wei Z 2021 *Opto-Electron. Adv.* **4** 200029
[17] Liu M, Wang H, and Yang H 2024 *Nonlinear Dyn.* **112** 1327
[18] Chen W and Mills D L 1987 *Phys. Rev. Lett.* **58** 160
[19] Kartashov Y V, Vysloukh V A, and Torner L 2006 *Phys. Rev. Lett.* **96** 073901
[20] Du Y J, Yu J Y, and Bai J T 2017 *Commun. Theor. Phys.* **67** 425
[21] Wang T Y, Zhou Q, and Liu W J 2022 *Chin. Phys. B* **31** 020501
[22] Ege S M and Misirli E 2014 *Adv. Differ. Equ.* **2014** 135

[23] Kumar D, Seadawy A R, and Joardar A K 2018 *Chin. J. Phys.* **56** 75
[24] Yu W T, Zhou Q, Mirzazadeh M, Liu W J, and Biswas A 2019 *J. Adv. Res.* **15** 69
[25] Tanyu D N, Ning J, Freudenberg T, Heilenkötter N, Rademacher A, Iben U, and Maass P 2023 *Inverse Probl.* **39** 103001
[26] Frins E, Failache H, Ferrari J, Costa G D, and Lezama A 1994 *Appl. Opt.* **33** 7472
[27] Rudy S H, Brunton S L, Proctor J L, and Kutz J N 2017 *Sci. Adv.* **3** e1602614
[28] Ji W, Qiu W, Shi Z, Pan S, and Deng S 2021 *J. Phys. Chem. A* **125** 8098
[29] Chiu P H, Wong J C, Ooi C, Dao M H, and Ong Y S 2022 *Comput. Meth. Appl. Mech. Eng.* **395** 114909
[30] Narkhede M V, Bartakke P P, and Sutaone M S 2022 *Artif. Intell. Rev.* **55** 291
[31] Zhang Z, Liu M, and Liu W 2024 *J. Syst. Sci. Complex.* 1
[32] Yin Y H and Lü X 2024 *Chaos, Solitons & Fractals* **181** 114595
[33] Wang S, Yu X, and Perdikaris P 2022 *J. Comput. Phys.* **449** 110768
[34] Penwarden M, Zhe S, Narayan A, and Kirby R M 2023 *J. Comput. Phys.* **477** 111912
[35] Moodie E E M, Dean N, and Sun Y R 2014 *Stat. Biosci.* **6** 223
[36] Psaros A F, Kawaguchi K, and Karniadakis G E 2022 *J. Comput. Phys.* **458** 111121
[37] Kharazmi E, Zhang Z Q, and Karniadakis G E M 2021 *Comput. Meth. Appl. Mech. Eng.* **374** 113547
[38] Goswami S, Yin M, Yu Y, and Karniadakis G E 2022 *Comput. Meth. Appl. Mech. Eng.* **391** 114587
[39] Korivand S, Jalili N, and Gong J 2023 *Sensors* **23** 2698
[40] Zhao T, Wang J, and Yue M 2023 *IEEE Trans. Power Syst.* **38** 5356
[41] Watkins C J C H and Dayan P 1992 *Mach. Learn.* **8** 279
[42] Wang Y H, Li T H S, and Lin C J 2013 *Eng. Appl. Artif. Intell.* **26** 2184
[43] Gerdjikov V S and Ivanov M I 1983 *Bulg. J. Phys.* **10** 130
[44] Xu S, He J, and Wang L 2011 *J. Phys. A* **44** 305203
[45] Kaup D J and Newell A C 1978 *J. Math. Phys.* **19** 798
[46] Moses J, Malomed B A, and Wise F W 2007 *Phys. Rev. A* **76** 021802
[47] Guo L J, Zhang Y S, Xu S W, Wu Z W, and He J S 2014 *Phys. Scr.* **89** 035501
[48] He B and Meng Q 2010 *Commun. Nonlinear Sci. Numer. Simul.* **15** 1783
[49] Luo J H and Fan E G 2021 *Appl. Math. Lett.* **120** 107297
[50] Parasuraman E 2021 *Optik* **243** 167429
[51] Vega-Guzman J, Biswas A, Mahmood M F, Zhou Q, Khan S, and Moshokoa S P 2019 *Optik* **181** 499
[52] Triki H, Sun Y Z, Biswas A, Zhou Q, Yıldırım Y, Zhong Y, and Alshehri H M 2022 *Results Phys.* **34** 105272