

Lax pairs informed neural networks solving integrable systems

Juncai Pu^a, Yong Chen^{b,c,*}

^a Institute of Applied Physics and Computational Mathematics, Beijing, 100088, China

^b School of Mathematical Sciences, Key Laboratory of Mathematics and Engineering Applications (Ministry of Education) & Shanghai Key Laboratory of PMMP, East China Normal University, Shanghai, 200241, China

^c College of Mathematics and Systems Science, Shandong University of Science and Technology, Qingdao, 266590, China

ARTICLE INFO

Keywords:

Lax pairs informed neural networks
Integrable systems
Localized wave solutions
Lax pairs

ABSTRACT

Lax pairs are one of the most important features of integrable system. In this work, we propose the Lax pairs informed neural networks (LPNNs) tailored for the integrable systems with Lax pairs by designing novel network architectures and loss functions, comprising LPNN-v1 and LPNN-v2. The most noteworthy advantage of LPNN-v1 is that it can transform the solving of complex integrable systems into the solving of relatively simple Lax pairs, and it not only efficiently solves data-driven localized wave solutions, but also obtains spectral parameter and corresponding spectral function in spectral problems of the integrable systems. On the basis of LPNN-v1, we additionally incorporate the compatibility condition/zero curvature equation of Lax pairs in LPNN-v2, its major advantage is the ability to solve and explore high-accuracy data-driven localized wave solutions and associated spectral problems for integrable systems with Lax pairs. The numerical experiments focus on studying abundant localized wave solutions for very important and representative integrable systems with Lax pairs, including the soliton solution of the Korteweg-de Vries (KdV) equation and modified KdV equation, rogue wave solution of the nonlinear Schrödinger equation, kink solution of the sine-Gordon equation, non-smooth peakon solution of the Camassa-Holm equation and pulse solution of the short pulse equation, as well as the line-soliton solution of Kadomtsev-Petviashvili equation and lump solution of high-dimensional KdV equation. The innovation of this work lies in the pioneering integration of Lax pairs informed of integrable systems into deep neural networks, thereby presenting a fresh methodology and pathway for investigating data-driven localized wave solutions and spectral problems.

1. Introduction

As a unique category of nonlinear systems, integrable systems exhibit rich mathematical structures and distinctive properties, including Lax pairs, multiple soliton solutions, and infinite conservation laws, setting them apart from general nonlinear systems [1]. The exploration of integrable systems holds significant importance with wide-ranging applications in fields such as mathematics, physics, engineering, and various interdisciplinary fields, as a crucial research branch in nonlinear science, its advancements are also intricately tied to the progress in computer science. In recent years, amid the various revolutions in computer hardware and software technology, deep learning methods have emerged as a formidable approach for solving partial differential equations (PDEs). The

* Corresponding author.

E-mail address: ychen@sei.ecnu.edu.cn (Y. Chen).

<https://doi.org/10.1016/j.jcp.2024.113090>

Received 12 January 2024; Received in revised form 11 April 2024; Accepted 4 May 2024

Available online 10 May 2024

0021-9991/© 2024 Elsevier Inc. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

idea is to optimize by constructing appropriate loss functions and use deep neural network (NN) to approximate unknown functions. Various neural network methods have been proposed, such as the physics-informed neural network (PINN) [2] and the deep Galerkin method [3] with the loss established on the L^2 -residual of the PDEs, and the deep Ritz method [4] with the loss based on the Ritz formulation, and the asymptotic-preserving neural network [5] with the loss satisfied the asymptotic preserving scheme, and so on. For other methods, we refer the reader to [6–11].

For decades, benefiting from the rapid development of integrable system theory, a wealth of integrable models endowed with excellent properties has been accumulated, and numerous localized wave solutions as well as graceful program algorithms have been obtained, their provide a continuous stream of data samples and prior information for studying integrable systems using deep learning technology. Recent years, a series of significant research achievement have been achieved on the data-driven forward and inverse problems of integrable systems [or nearly integrable systems] by applying PINN and its improved algorithms. Chen's research group introduced deep learning methods to study integrable systems, and they have been dedicated to establishing an efficient framework for integrable deep learning methods, resulting in rich research outcomes [12]. On the one hand, various data-driven localized wave solutions have been predicted from small amount of initial/boundary data, among which the most significant work is the first successful learning and extraction of rogue wave solutions [13]. Additionally, they also obtained higher-order rational solutions, periodic wave solutions and rogue waves on the periodic background wave, vector rogue waves and interaction solutions between rogue wave and soliton, as well as new kink-bell type solution [14–16]. Furthermore, various complex integrable models have been studied by employing PINN and its improved algorithms, including nonlocal integrable systems, coupled integrable systems, high-dimensional integrable systems, and variable coefficients integrable systems [15,17]. On the other hand, various strategies have been utilized to make multiple improvements to deep NN algorithms, including adaptive activation functions, parameter regularization, time domain piecewise, and transfer learning [11,14,18]. Especially, the properties of integrable systems are used to design new deep NN models suitable for integrable systems with such properties, such as the two-stage PINN based on conserved quantities [19] and the PINN based on Miura transformations [16]. Moreover, there are many important works using deep learning algorithms to study integrable systems, we refer the reader to [20–24].

The inverse scattering method (IST) is one of the most important discoveries in modern mathematical physics [25], which was originally proposed by Gardner, Greene, Kruskal and Miura (GGKM) in 1967 to study the initial value problem of rapid decay of the Korteweg-de Vries (KdV) equation [26]. In 1968, inspired by GGKM's work, Lax proposed the concept of Lax pairs and creatively proposed a general framework for using Lax pairs to represent integrable systems [27]. In 1972, Zakharov and Shabat presented Lax pairs in matrix form for the nonlinear Schrödinger (NLS) equation, and utilized IST to study the inverse scattering transformation and exact solutions of the NLS equation [28]. Later, Ablowitz, Kaup, Newell and Segur proposed a class of integrable systems with a unified Lax pairs form, known as AKNS systems, and established a universal framework for inverse scattering theory for the initial-value problem of AKNS systems in 1974 [29,30]. Subsequently, many classic integrable systems with Lax pairs were proven to be solvable via IST [31–33], which was also considered a milestone in the integrable system theory and established the important position of integrable systems in many fields of mathematics and physics. Therefore, Lax pairs provide the theoretical basis for constructing a general framework of inverse scattering theory for integrable systems, promoting the development of integrable system theory.

Lax pairs are the key characteristic and important tool in the integrable system theory, which are closely related to the integrability, conservation laws, classification, and exact solution solving of integrable systems. Lax pairs were first introduced by Lax in 1968, who stated that soliton equations can be represented by Lax pairs and Lax equations [27]. Given a linear operator L involved space $\mathbf{x}$ and potential, let it satisfies spectral equation

$$L\phi = \lambda\phi, \quad (1.1)$$

here ϕ is eigenfunction, λ is spectral parameter. Consider the isospectral problem of λ that is independent of time t , i.e. $\lambda_t = 0$. Given an operator A , ϕ satisfies the linear equation

$$\phi_t = A\phi. \quad (1.2)$$

If ϕ is required to satisfy both Eqs. (1.1) and (1.2), then L and A satisfy the following operator equation

$$L_t - [A, L] = 0, \quad (1.3)$$

where $[A, L] = AL - LA$. Then Eq. (1.3) is called the Lax equation, and the spectral problems (1.1) and (1.2) are called the Lax pairs. Once given specific operators L and A , one can obtain the specific compatibility condition equation

$$f_{\text{cce}} : \underbrace{\phi_{\mathbf{x} \dots \mathbf{x}}}_s \phi_t - \underbrace{\phi_t}_{\mathbf{x} \dots \mathbf{x}} \phi = 0, \quad (1.4)$$

in which s is a finite positive integer, and the size of s is determined by the dimensions of space $\mathbf{x}$ and the form of Lax pairs spatial part, $\phi_{\mathbf{x} \dots \mathbf{x}}$ indicates ϕ taking s -th partial derivative of space $\mathbf{x}$, here $s \geq 1$. In fact, $\phi_{\mathbf{x} \dots \mathbf{x}} \phi_t$ is obtained by taking partial derivatives of t on both sides of Eq. (1.1), and $\phi_t \mathbf{x} \dots \mathbf{x}$ is obtained by taking the s -order partial derivative of $\mathbf{x}$ on both sides of Eq. (1.2). Certainly, different integrable systems correspond to different compatibility condition equations, and here we only provide a unified representation Eq. (1.4). Generally, compatibility condition equation (1.4) is equivalent to Lax equation (1.3), both of which can derive the desired soliton equation. In addition to the above operator representations, Lax pairs can also be expressed in matrix form,

usually Lax pairs with operator form can be rewritten as Lax pairs with matrix form. Specially, as we consider the case where space $x = x$ is one-dimensional, then let a pair of spectral problems (1.1) and (1.2) be rewritten as

$$\begin{aligned}\Phi_x &= M\Phi, \\ \Phi_t &= N\Phi.\end{aligned}\tag{1.5}$$

Where Φ is an n -dimensional column vector $\Phi = (\Phi_1, \Phi_2, \dots, \Phi_n)^T$, M and N are n -order matrices that depend on the potential $q = q(x, t)$ and spectral parameter λ . If requirements Eqs. (1.5) are compatible, that is Φ satisfies compatibility condition equation $\Phi_{xt} - \Phi_{tx} = 0$, then we can derive that M and N must be satisfied zero curvature equation

$$f_{zcc} : M_t - N_x + [M, N] = 0.\tag{1.6}$$

Usually, we refer to Eqs. (1.5) as Lax pairs, thus the compatibility condition equation (1.4) is equivalent to the zero curvature equation (1.6). The corresponding integrable systems can be derived from both the compatibility condition equation (1.4) and zero curvature equation (1.6). If a PDE can be generated via the compatibility condition (1.4) for operator Lax pairs (1.1)-(1.2) or zero curvature equation (1.6) for matrix Lax pairs (1.5), then this PDE is said to be Lax integrable. Lax pairs are a prominent feature in the integrable systems theory, which can simplify the dynamic equations of integrable systems, thereby helping researchers study the integrability, construct conserved quantities, and classify integrable families. Furthermore, some methods for solving exact solutions of integrable systems also rely on the Lax pairs of integrable systems, such as the Darboux transformation method [34]. In summary, the proposal of Lax pairs greatly promoted the development of exactly solving methods of integrable systems and established its important position in integrable system theory, mathematical physics, and other related fields. Hence, the successful application of Lax pairs in aforementioned fields reminds us whether Lax pairs can be successfully applied to the deep NN for studying data-driven problems of integrable systems?

We introduce Lax pairs and their compatibility conditions/zero curvature equation into deep NNs, and propose Lax pairs informed neural networks (LPNNs), the novel algorithm exhibits remarkable efficiency or high-accuracy in solving integrable systems. In order to fully demonstrate the high efficiency and high precision of LPNNs in solving integrable systems with Lax pairs, we utilized LPNNs to solve several important integrable systems and their spectral problems. The KdV equation is the first PDE to describe solitary wave and the first integrable model studied by applying the IST [26,35], and the KdV equation is closely related to the origin of soliton and the flourishing development of integrable system theory, which has had a significant impact on the fields of nonlinear science and mathematical physics [36], it is the most iconic model in the soliton and integrable system theory. The Camassa-Holm (CH) equation is the earliest integrable model discovered to both possess non-smooth solutions and Lax pairs, which can describe the unidirectional water wave motion in shallow water waves and has important physical significance [37]. The Kadomtsev-Petviashvili (KP) equation is a high-dimensional extension of the KdV equation, which describes the motion of two-dimensional water waves and has widely applied in the fields of fluid mechanics and theoretical physics [38]. The term “breather” arose from studies of the sine-Gordon (SG) equation, which is one of the most fundamental equations of integrable systems and has important applications in fields such as nonlinear optics, crystal dislocation, and superconductivity [39]. The modified KdV (mKdV) equation is a KdV equation with cubic nonlinear term, and its transformation relationship with the solution of the KdV equation has opened up a research boom of Miura transformations [40]. The NLS equation has important applications in various fields such as quantum mechanics, optics, plasmas, and Bose-Einstein condensates, and it is also the most fundamental equation for describing rogue wave phenomena [28]. Unlike the KdV, mKdV, SG and NLS equations in the AKNS hierarchy, the short pulse (SP) equation is the most classical integrable model in Wadati-Konno-Ichikawa (WKI) equations [41], and possess WKI-type Lax pairs, which has important applications in many physical fields such as nonlinear optics [42]. These integrable models are the most significant, foundational and classic models in the realm of integrable systems, and many integrable models can be derived from the deformation and generalization of these basic integrable models. Hence, the successful application of LPNNs in solving these significant, classic and representative integrable systems serves as a compelling demonstration of the efficacy of the LPNNs algorithm.

The main highlights of this article are as follows:

1. We cleverly transform complex integrable systems into relatively simple equations set [Lax pairs] to improve the training efficiency of the network. LPNN-v1 relies exclusively on the Lax pairs of integrable systems to extract information, significantly simplifying the solving of complex integrable systems. LPNN-v1 can not only effectively solve data-driven localized wave solutions in such systems, but also conveniently learn the spectral parameter and their corresponding spectral function in spectral problems. The numerical results indicate that LPNN-v1 is applicable to both low-dimensional and high-dimensional integrable systems, and exhibits efficient training performance in solving smooth and non-smooth solutions. Specifically, for localized wave solution of certain integrable systems, LPNN-v1 can reduce training time by more than 5 times than standard PINN. Moreover, owing to the fact that the Lax pairs representation of high-dimensional integrable systems is frequently considerably simpler than the equations themselves, LPNN-v1 enjoys substantial advantages in solving high-dimensional integrable systems.
2. In the case of certain simple integrable systems or intricate localized wave solutions, LPNN-v1 tends to lose its efficiency. Therefore, building upon LPNN-v1, we further introduce the compatibility condition/zero curvature equation of Lax pairs to propose the LPNN-v2. The numerical results indicate that LPNN-v2 can solve high-precision data-driven localized wave solutions and spectral problems for all integrable systems with Lax pairs, and can even improve training accuracy by more an order of magnitude when solving localized wave for certain integrable systems.

The structure of the paper unfolds as follows: In section 2, we present the innovative models of LPNNs. Section 3 provides a comprehensive display of numerical experiments conducted to validate the effectiveness of our proposed methods. We encapsulate our work and draw meaningful conclusions in section 4.

2. Methodology

Generally, we consider a multi-dimensional spatiotemporal real nonlinear integrable system with operator Lax pairs (1.1)-(1.2) or matrix Lax pairs (1.5) in the general form given by

$$F[q, q^2, \dots, \nabla_t q, \nabla_t^2 q, \dots, \nabla_x q, \nabla_x^2 q, \dots, q \cdot \nabla_t q, \dots, q \cdot \nabla_x q, \dots] = 0, \quad (2.1a)$$

$$f_{\text{LP}} : \begin{cases} L\phi = \lambda\phi \\ \phi_t = A\phi \end{cases}, \text{ usually for } \mathbf{x} \in \Omega, \text{ or } \begin{cases} \Phi_x = M\Phi \\ \Phi_t = N\Phi \end{cases}, \text{ usually for } \mathbf{x} = x, \quad (2.1b)$$

in which potential $q = q(\mathbf{x}, t) \in \mathbb{R}^{n \times 1}$ is the n -dimensional latent solution, $\mathbf{x} \in \Omega$ specifies the n -dimensional space and $t \in [T_i, T_f]$ denotes time [T_i and T_f respectively indicate the initial time and final time], ∇ is the gradient operator with respect to $\mathbf{x}$ and t , $F[\cdot]$ is a complex nonlinear operator of q and its spatiotemporal derivatives. Here linear operator L involves space $\mathbf{x}$ and potential q , λ indicates spectral parameter and ϕ represents spectral function corresponding to spectral parameter. $\Phi = \Phi(\mathbf{x}, t)$ stands for vector spectral function corresponding to spectral parameter λ in matrices M and N . Usually, from Lax pairs (2.1b), we can derive all integrable systems by means of compatibility condition equation (1.4) [for Lax pairs of operator form] and zero curvature equation (1.6) [for Lax pairs of matrix form].

Then we consider the general initial and boundary conditions of spatiotemporal nonlinear integrable system denoted by

$$\begin{aligned} I[q, \phi/\Phi; \mathbf{x} \in \Omega, t = T_i] &= 0, \\ B[q, \phi/\Phi, \nabla_x q; \mathbf{x} \in \partial\Omega, t \in [T_i, T_f]] &= 0. \end{aligned} \quad (2.2)$$

If we consider a complex valued potential $\hat{q} \in \mathbb{C}^{n \times 1}$ for nonlinear integrable systems, we can utilize decomposition $\hat{q} = \hat{u} + i\hat{v}$ to derive two real-value functions $\hat{u} \in \mathbb{R}^{n \times 1}$ and $\hat{v} \in \mathbb{R}^{n \times 1}$, then back to the problem of Eq. (2.1a). The initial and boundary points set D_{ib} for training are sampled randomly through corresponding initial and boundary conditions (2.2), and the collocation points set D_c for training are generated by the Latin Hypercube Sampling method [43].

The next part, we utilize the Lax pairs, a prominent feature of integrable systems, to design novel deep learning method, namely LPNNs. The NN part of LPNNs still adopts fully connected networks, while the Lax pairs informed part is constructed through Lax pairs, compatibility condition equation/zero curvature equation. Given an input $\mathbf{x}'$ [for convenience, here $\mathbf{x}'$ include $\mathbf{x}$ and t], the output of the L -layer deep feedforward NN F is a combination of the affine transformations $\{\mathcal{A}_d\}_{d=1}^L$ and nonlinear activation functions $\{\sigma_d\}_{d=1}^{L-1}$, stated as follows

$$\begin{aligned} F(\mathbf{x}'; \theta) &= \mathcal{A}_L \circ \sigma_{L-1} \circ \mathcal{A}_{L-1} \cdots \circ \sigma_2 \circ \mathcal{A}_2 \circ \sigma_1 \circ \mathcal{A}_1(\mathbf{x}'), \\ \mathcal{A}_d(\mathbf{x}'_{d-1}) &\triangleq \mathbf{W}^d \mathbf{x}'_{d-1} + \mathbf{b}^d, \quad \mathbf{x}'_0 := \mathbf{x}', \\ \mathbf{x}'_d &= \sigma_d \circ \mathcal{A}_d \triangleq \sigma_d(\mathcal{A}_d(\mathbf{x}'_{d-1})), \end{aligned}$$

in which the weight and bias term are $\mathbf{W}^d \in \mathbb{R}^{N_d \times N_{d-1}}$ and $\mathbf{b}^d \in \mathbb{R}^{N_d}$ in the NN associated with the d -th layer, and the N_d denote the number of neurons contained in d -th hidden layer. Here popular choices of σ_d include the ReLU function, the sigmoid function, the hyperbolic tangent function, and so on. In this work, we all adopt the hyperbolic tangent as activation function, the set of trainable parameters $\theta \in \mathcal{N}$ consists of $\{\mathbf{W}^d, \mathbf{b}^d\}_{d=1}^D$, in which $\mathcal{N}$ is the parameter space.

For the construction of Lax pairs informed part of LPNN, we mainly divide it into two methods: LPNN-v1 only uses Lax pairs as Lax pairs informed; while LPNN-v2 utilizes Lax pairs and their zero curvature equation/compatibility condition equation as Lax pairs informed. Each of the two schemes has its unique advantages, among which the LPNN-v1 relies entirely on the Lax pairs of integrable systems to extract Lax pairs informed, greatly simplifying the solving of complex integrable systems, it is well-suited for studying localized wave solutions and spectral problems of complex integrable systems with uncomplicated Lax pairs, and it stands out for its high training efficiency. While the LPNN-v2 method is primarily tailored for delving into various localized wave solutions and spectral problems within all integrable systems with Lax pairs, and it usually attains high training accuracy. Fig. 1 depicts the schematic architecture of the LPNNs model, encompassing both the LPNN-v1 and LPNN-v2.

Specifically, Fig. 1 displays two versions of LPNNs through the upper and lower panels, namely LPNN-v1 and LPNN-v2. The upper panel Fig. 1 a showcase the NN part and Lax pairs informed part, where the left part of panel Fig. 1 a is the fully connected deep feedforward network, the input $\mathbf{x}'$ include $\mathbf{x}$ and t , the output F contains potential q and spectral function ϕ/Φ . While the right part of panel Fig. 1 a is Lax pairs informed network dominated by the Lax pairs and their related conditions. Different Lax pairs constraints correspond to different versions of LPNNs, that is LPNN-v1 only corresponds to “Lax pairs of integrable systems”, then LPNN-v2 corresponds to “Lax pairs of integrable systems” and corresponding “compatibility condition equation”/“zero curvature equation”. The spectral parameter λ corresponds to the spectral function Φ , which can be set as hyper-parameter or trained through network, and is reflected in the Lax pairs of integrable systems. The Lax pairs of certain high-dimensional integrable systems may not contain spectral parameter, we do not need to consider the spectral parameter term in this case. The left and right parts of panel Fig. 1

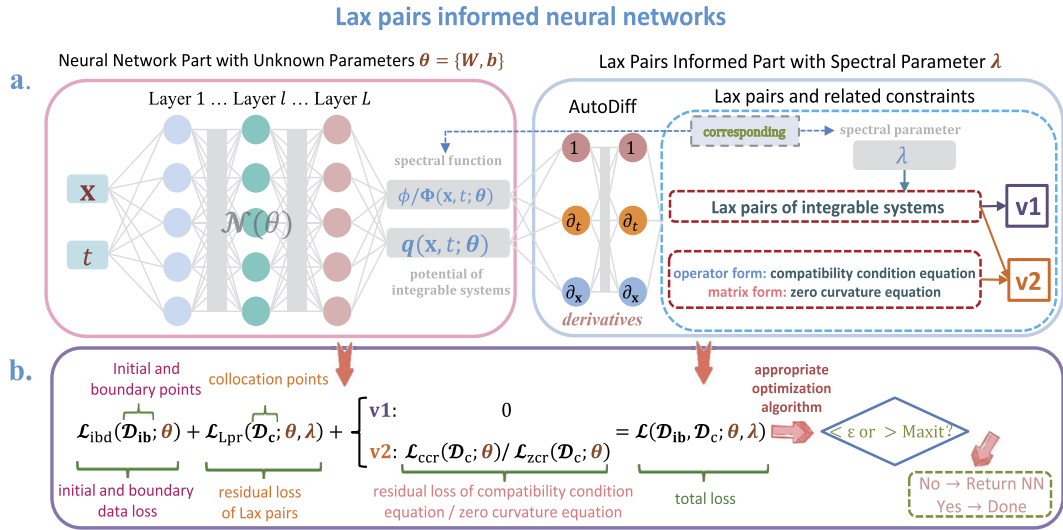


Fig. 1. (Color online) Schematic architecture of the LPNNs model for integrable systems with Lax pairs, including LPNN-v1 and LPNN-v2. **a** the NN part (left part) and Lax pairs informed part (right part). **b** the components of the loss function and optimize process in LPNNs.

a are connected using automatic differentiation [AutoDiff]. The loss function and optimize process of LPNNs are depicted in the lower panel Fig. 1 **b**, where the training parameters of total loss $\mathcal{L}$ are provided from two-part of panel Fig. 1 **a**, the optimization algorithms appropriately selected appropriate optimizer for training of LPNNs. Ultimately, employing appropriate optimization algorithms to seek optimal parameters θ^* , this endeavor aims to minimize the total loss function $\mathcal{L}$ effectively. The total loss function is defined as

$$\text{LPNN-v1} : \mathcal{L}(\mathcal{D}_{ib}, \mathcal{D}_c; \theta, \lambda) = \mathcal{L}_{ibd}(\mathcal{D}_{ib}; \theta) + \mathcal{L}_{Lpr}(\mathcal{D}_c; \theta, \lambda), \quad (2.3a)$$

$$\text{LPNN-v2} : \mathcal{L}(\mathcal{D}_{ib}, \mathcal{D}_c; \theta, \lambda) = \mathcal{L}_{ibd}(\mathcal{D}_{ib}; \theta) + \mathcal{L}_{Lpr}(\mathcal{D}_c; \theta, \lambda) + \begin{cases} \text{operator form} : \mathcal{L}_{ccr}(\mathcal{D}_c; \theta) \\ \text{matrix form} : \mathcal{L}_{zcr}(\mathcal{D}_c; \theta) \end{cases}. \quad (2.3b)$$

Here, $\mathcal{L}_{ibd}$ represents the initial and boundary data loss, $\mathcal{L}_{Lpr}$ indicates residual loss of Lax pairs, then $\mathcal{L}_{ccr}$ and $\mathcal{L}_{zcr}$ respectively represent the residual loss of compatibility condition equation [for Lax pairs of operator form] and residual loss of zero curvature equation [for Lax pairs of matrix form]. They can be defined as follows

$$\mathcal{L}_{ibd}(\mathcal{D}_{ib}; \theta) = \frac{1}{N_{ib}} \|q^{\theta, ib} - q^{m, ib}\|_2^2, \quad (2.4)$$

$$\mathcal{L}_{Lpr}(\mathcal{D}_c; \theta, \lambda) = \frac{1}{N_c} \|f_{Lp}^c\|_2^2, \quad (2.5)$$

and

$$\mathcal{L}_{ccr}(\mathcal{D}_c; \theta) = \frac{1}{N_c} \|f_{cce}^c\|_2^2, \quad (2.6)$$

$$\mathcal{L}_{zcr}(\mathcal{D}_c; \theta) = \frac{1}{N_c} \|f_{zce}^c\|_2^2, \quad (2.7)$$

where N_{ib} and N_c represent respectively the number of elements in sets $\mathcal{D}_{ib}$ and $\mathcal{D}_c$, $\|\cdot\|_2$ denotes the L^2 norm. Then $q^{\theta, ib}$ represents the learning result of q^θ acting on initial and boundary points set $\mathcal{D}_{ib}$. Besides, $q^{m, ib}$ represents the measurement data of q on initial and boundary points set $\mathcal{D}_{ib}$. The f_{Lp}^c is value of Lax pairs f_{Lp} on collocation points set $\mathcal{D}_c$. The f_{cce}^c is value of compatibility condition equation f_{cce} on collocation points set $\mathcal{D}_c$, and the f_{zce}^c is value of zero curvature equation f_{zce} on collocation points set $\mathcal{D}_c$. Finally, we summarize the main steps of LPNNs in Algorithm 1.

3. Numerical experiment

In this section, we utilize three different types of deep learning models to study data-driven localized wave solutions, and solve spectral problems for integrable systems with several different types of Lax pairs, then provide detailed numerical results and related dynamic behavior figures. In this work, unless otherwise specified, all deep learning methods [involve the PINN and LPNNs] both possess 5 hidden-layer NNs with 100 neurons per hidden layer, namely $L = 6$ and $N_d = 100$. Additionally, most of the numerical results [this section and Appendices A-B] in this article were obtained based on the TensorFlow framework, while a small number of numerical results [Appendix C] were obtained based on the JAX framework.

Algorithm 1 The Lax pairs informed neural networks for integrable systems.

Step 1: Specification of training set in computational domain:

initial and boundary training points: D_{ib} , collocation training points: D_c .

Step 2: Construct neural network F [including potential q and spectral function ϕ/Φ] with random initialization of parameters θ [contain/preset spectral parameter λ].

Step 3: Construct the Lax pairs informed part by substituting surrogate $F(x'; \theta)$ into the Lax pairs of integrable systems in LPNN-v1 [or the Lax pairs of integrable systems and compatibility condition equation/ zero curvature equation in LPNN-v2].

Step 4: Specification of the total loss function $\mathcal{L}(D_{ib}, D_c; \theta, \lambda)$.

Step 5: Seek the optimal parameters θ^* using suitable optimization algorithms for minimizing the total loss function $\mathcal{L}$ as

$$\theta^* = \arg \min_{\theta \in \mathcal{N}} \mathcal{L}(\theta).$$

3.1. Efficient training performance of LPNN-v1

In this subsection, we apply LPNN-v1 to study integrable systems with simple Lax pairs, and efficiently solve data-driven solutions and corresponding spectral problems.

- **Case 1: Korteweg-de Vries equation**

The KdV equation is a PDE that describes the motion of water waves, and is one of the most classic equations in the soliton theory and integrable systems. In 1834, British scientist Russell occasionally observed a type of water wave [also known as a solitary wave] whose shape and velocity did not change during the process of traveling, but he did not find an appropriate model or specific mathematical expression to describe this interesting natural phenomenon at the time. Until 1895, Korteweg and de Vries proposed the KdV equation and pointed out that the traveling wave solution of the equation could perfectly explain the solitary wave phenomenon discovered by Russell [35]. In 1965, Zabusky and Kruskal first discovered the connection between the FPU problem proposed by Fermi, Pasta, Ulam, and Tsingou in 1955 and the KdV equation by utilizing numerical calculation methods [44], and explained the FPU regression phenomenon using the dynamic behavior of solitary waves in the KdV equation [36]. Since then, Zabusky and Kruskal proposed the term “soliton” in 1965 to reflect the properties of solitary waves and particle interactions [36]. Solitons are nonlinear localized waves that can maintain their shape and velocity during propagation, even when interacting with other waves. Therefore, the KdV equation is closely related to the origin of solitons and the flourishing development of integrable system theory, which has had a significant impact on the fields of nonlinear science and mathematical physics. Until now, the KdV equation is widely recognized as a paradigm for the description of weakly nonlinear long waves in many branches of physics and engineering.

We consider the KdV equation with Lax pairs [operator form] as follows

$$u_t + 6uu_x + u_{xxx} = 0, \quad (3.1)$$

$$f_{LP} : \begin{cases} \phi_{xx} = (\lambda - u)\phi \\ \phi_t = u_x \phi - (4\lambda + 2u)\phi_x \end{cases}, \quad (3.2)$$

thus KdV equation (3.1) can be derived by means of Lax pairs (3.2) and specific formula $\phi_{xxt} - \phi_{txx} = 0$ of compatibility condition equation (1.4), where ϕ_{xxt} is computed from the first equation of Eq. (3.2) and ϕ_{txx} is given by the second equation of Eq. (3.2). Then we consider following initial condition $\mathcal{I}$ and boundary conditions $\mathcal{B}$ for KdV equation (3.1) in spatiotemporal region $[-5, 5] \times [-5, 5]$

$$\begin{aligned} u(x, t = -5) &= 2\text{sech}(20 + x)^2, \quad x \in [-5, 5], \\ u(-5, t) &= 2\text{sech}(-4t - 5)^2, \quad u(5, t) = 2\text{sech}(-4t + 5)^2, \quad t \in [-5, 5]. \end{aligned} \quad (3.3)$$

For the Lax pairs of KdV equation (3.1), spectral function $\phi(x, t) \in \mathbb{R}^{1 \times 1}$ satisfied free initial-boundary condition and initialized to $\phi(x, t) = 0$ in spatiotemporal region $[-5, 5] \times [-5, 5]$. Then we apply conventional PINN and novel LPNN-v1 to solve the KdV equation (3.1) with spectral problem (3.2), and set spectral parameter $\lambda = 1$ as well as training points $N_{ib} = 400$, $N_c = 10000$. The relative L^2 norm error of the LPNN-v1 model achieves 5.756885e-03 for data-driven single-soliton solution $u(x, t)$ in 64.6284 seconds, and the number of iterations is 327.

Fig. 2 manifests the deep learning results of the data-driven single-soliton solution $u(x, t)$ and spectral function $\phi(x, t)$ stemming from the LPNN-v1 for the KdV equation (3.1). In Fig. 2(a), we display the density plots of the true dynamics, prediction dynamics and error dynamics, then showcase its corresponding amplitude scale size on the right side of density plots, and exhibit the sectional drawings which contain the learned and true solution at three different moments. The evolution curve figures of the loss function arising from the LPNN-v1 are displayed in Fig. 2(b). Figs. 2(c) and (d) indicate respectively the three-dimensional plot with contour map on three planes for the predicted single-soliton solution $u(x, t)$ and learned spectral function $\phi(x, t)$.

To further showcase the efficiency of LPNN-v1 in studying localized wave solutions of integrable systems with concise Lax pairs, we provide a comparison of the training errors and training time between LPNN-v1 and conventional PINNs in Table 1, which spans same spatio-temporal domain as well as possesses identical hyper-parameters and network setting. From Table 1, we can obtain that although the training error using LPNN-v1 is slightly greater than that using standard PINN, the training time of LPNN-v1 is much shorter than that required using PINN, benefiting from the simpler Lax pairs form of the KdV equation compared to the KdV equation itself. In realistic problems, faster and efficient network training is often more favored when the training errors of the network are not significantly different. From Table 1, we can intuitively observe that the feature of LPNN-v1 is that they can not only solve integrable system, but also solve spectral problem of integrable system and numerically obtain spectral parameter and corresponding spectral function, which PINN cannot obtain. The numerical spectral parameter and numerical spectral function obtained by LPNNs

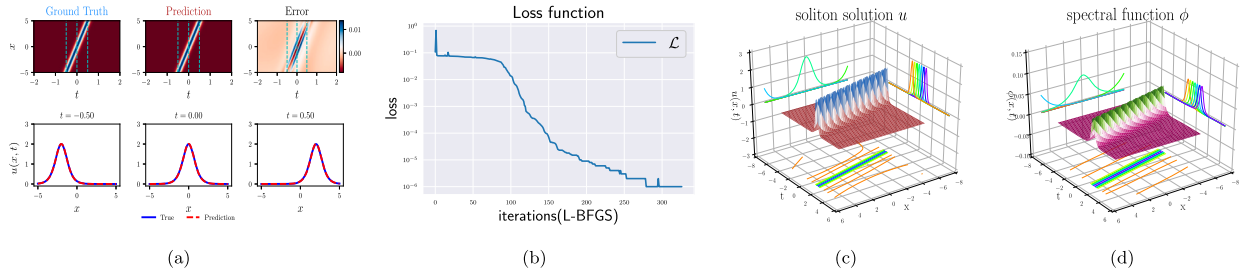


Fig. 2. (Color online) The training results of single-soliton solution $u(x,t)$ and spectral function $\phi(x,t)$ for KdV equation arising from the LPNN-v1. (a) The ground truth, prediction and error dynamics density plots, as well as sectional drawings which contain the true and prediction single-soliton solution at three distinct moments $t = -0.5, 0, 0.5$; (b) Evolution graph of the loss function in LPNN-v1; (c) The three-dimensional plot with contour map for the data-driven single-soliton solution; (d) The three-dimensional plot with contour map for the data-driven spectral function corresponding to spectral parameter $\lambda = 1$.

Table 1

Performance comparison between LPNN-v1 and conventional PINN for solving KdV equation.

Types	$\mathbf{x} \times t$	D_{ib}, D_c	optimizer	λ	ϕ	L^2 norm error	training time
PINN	$[-5,5] \times [-5,5]$	400,10000	L-BFGS	$\times$	$\times$	2.742974e-03	200.5237 s
LPNN-v1	$[-5,5] \times [-5,5]$	400,10000	L-BFGS	1.0	$\checkmark$	5.756885e-03	64.6287 s

play an important role in many classical theories, such as inverse scattering theory and Darboux transformation theory. It may bring some novel results to these classical theories, and providing important research foundations for the discovery of new localized wave solutions and new Lax pairs, which is of great significance. Therefore, the numerical results in this subsection indicate that LPNN-v1 is particularly suitable for studying cases where the integrable system model itself is complicated but possesses simple Lax pairs when training certain data-driven localized wave solutions.

• Case 2: Camassa-Holm equation

For verifying the effectiveness of LPNN-v1 in solving non-smooth solutions of integrable systems, we considered the classical CH equation

$$u_t - u_{xx} + 2\omega u_x + 3uu_x - 2u_x u_{xx} - uu_{xxx} = 0, \quad (3.4)$$

arises as a model describing the unidirectional propagation of shallow water waves over a flat bottom, where $u(x,t)$ represent the height of the free surface about a flat bottom and the real constant ω being related to the critical shallow water speed. The CH equation was first appeared in Ref. [45] as an abstract bi-Hamiltonian equation with infinitely many conservation laws. Camassa and Holm subsequently discovered Eq. (3.4) can serve as a model for describing unidirectional propagation of waves on shallow water in 1993 [37], and pointed out the CH equation is a completely integrable equation and manifested its solitary waves are peakons if $\omega = 0$ [37,46]. Especially, the CH equation is the earliest integrable models discovered to possess peakon solution different from the soliton solution, and the first-order derivative of this non-smooth peakon solution at the wave peak does not exist. The peakon solution is also considered as a unique type of non-smooth soliton solution to the integrable CH equation [37]. From Ref. [37], the CH equation (3.4) with $\omega = 0$ possesses the following Lax pairs

$$f_{Lp} : \begin{cases} \phi_{xx} = \left[\frac{1}{4} + \lambda(u - u_{xx}) \right] \phi \\ \phi_t = \left(\frac{1}{2\lambda} - u \right) \phi_x + \frac{1}{2} u_x \phi \end{cases}, \quad (3.5)$$

one can deduce the CH equation (3.4) with $\omega = 0$ by employing the Lax pairs (3.5) and compatibility condition equation $\phi_{xxt} - \phi_{txx} = 0$. We set spectral function $\phi(x,t) \in \mathbb{R}^{1 \times 1}$ and spectral parameter $\lambda \in \mathbb{R}$, while spectral parameter λ is initialized to $\lambda = 1.5$ as well as we initialize $\phi = 0$ and make it satisfy the free initial-boundary conditions, then we consider the initial and boundary condition of CH equation (3.4) with $\omega = 0$ in spatiotemporal region $[-5,5] \times [0,3]$, shown as below

$$\begin{aligned} u(x, t=0) &= 0.9e^{-|x|}, \quad x \in [-5,5], \\ u(-5, t) &= 0.9e^{-| -5 - 0.9t |}, \quad u(5, t) = 0.9e^{-| 5 - 0.9t |}, \quad t \in [0,3]. \end{aligned} \quad (3.6)$$

We randomly select $N_{ib} = 400$ points based on the aforementioned initial and boundary conditions (3.6), and extract $N_c = 10000$ collocation points in residual spatiotemporal region, then we produce the data-set for the LPNN-v1 pertaining to the CH equation. After that, by utilizing LPNN-v1 with training data-set, we successfully and efficiently obtained data-driven peakon solution, as well as corresponding spectral parameter and spectral function. The relative L^2 norm error of the LPNN-v1 model achieves 6.878529e-02 for data-driven peakon solution $u(x,t)$ in 148.1191 seconds, and the number of iterations is 530. The spectral parameter learned from LPNN-v1 is $\lambda = 1.4812$.

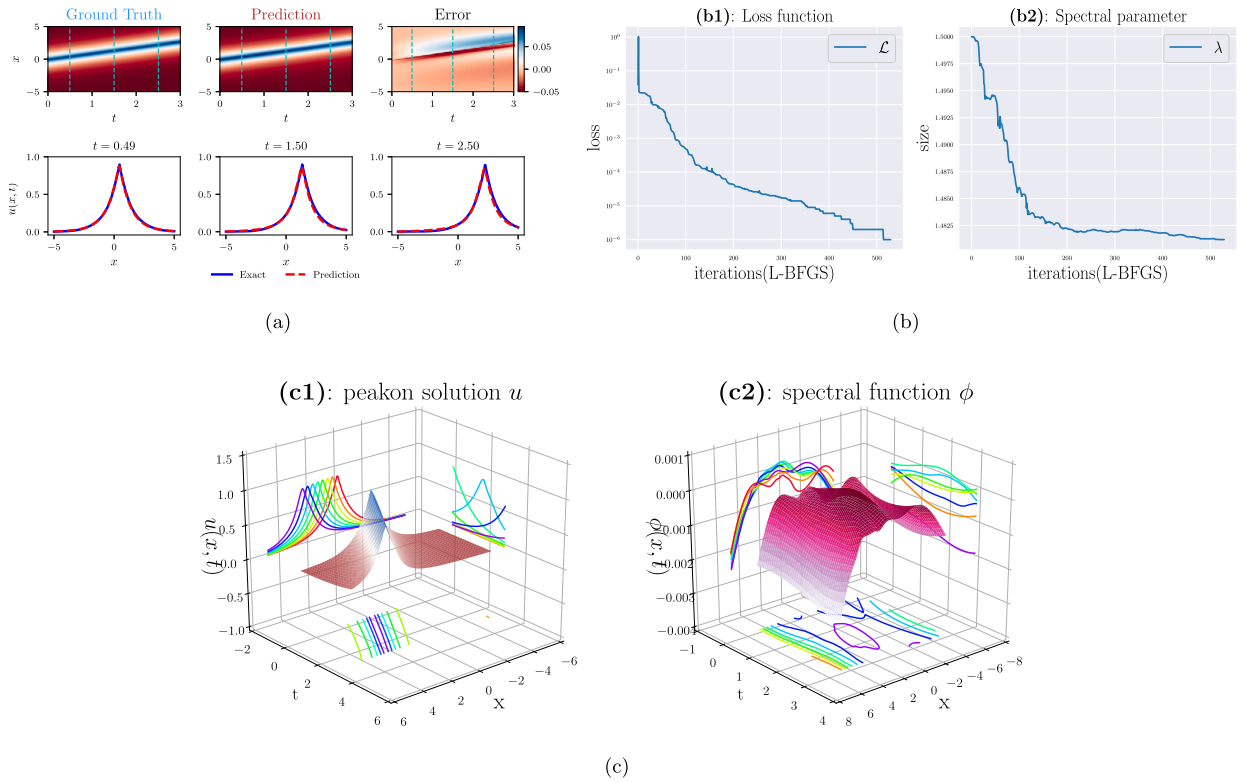


Fig. 3. (Color online) The training results of peakon solution $u(x,t)$ and spectral function $\phi(x,t)$ for CH equation arising from the LPNN-v1. (a) The ground truth, prediction and error dynamics density plots, as well as sectional drawings which contain the true and prediction peakon solution at three distinct moments $t = 0.49, 1.5, 2.5$; (b) Evolution graphs of the loss function [panel b1] and spectral parameter λ [panel b2] in LPNN-v1; (c) The three-dimensional plot with contour map for the data-driven peakon solution [panel c1] and the data-driven spectral function [panel c2] corresponding to spectral parameter $\lambda = 1.4812$.

Table 2

Performance comparison between LPNN-v1 and conventional PINN for solving CH equation.

Types	$\mathbf{x} \times t$	$D_{\text{tr}}, D_{\text{c}}$	optimizer	λ	ϕ	L^2 norm error	training time
PINN	$[-5,5] \times [0,3]$	400,10000	L-BFGS	$\times$	$\times$	$6.882064\text{e-}02$	819.0868 s
LPNN-v1	$[-5,5] \times [0,3]$	400,10000	L-BFGS	1.4812	$\checkmark$	$6.878529\text{e-}02$	148.1191 s

Fig. 3 exhibits the deep learning results of the data-driven peakon solution $u(x,t)$ and spectral function $\phi(x,t)$ stemming from the LPNN-v1 for the CH equation (3.4) with $\omega = 0$. Fig. 3(a) showcases the density plots of the ground truth dynamics, prediction dynamics and error dynamics, then give its corresponding amplitude scale size on the right side of density plots, and exhibit the sectional drawings which contain the learned and true solution at three different moments. The evolution curve figures of the loss function [panel b1] and spectral parameter λ [panel b2] resulting from the LPNN-v1 are displayed in Fig. 3(b). Figs. 3(c1) and (c2) indicate respectively the three-dimensional plot with contour map on three planes for the predicted peakon solution and learned spectral function corresponding to spectral parameter $\lambda = 1.4812$.

Similarly, in order to further demonstrate the efficiency of LPNN-v1, we compared the training performance with the standard PINN and provided detailed comparison results in Table 2. By the way, the abundant data-driven peakon and periodic peakon solutions of the CH equation have been investigated by utilizing the PINN algorithm, interested readers refer to [47]. Unlike case 1, in this example we have no fixed spectral parameter λ for training, but instead train through LPNN-v1 after giving the initial spectral parameter value. From Table 2, we can clearly see that the training error of LPNN-v1 is not significantly improved compared to the standard PINN algorithm, but the training time is more than 5 times faster than the standard PINN algorithm. In contrast with the numerical result of KdV equation in case 1, the training of CH equation in this case is significantly more efficient, it benefits from the complexity for the Lax pairs of the CH equation and KdV equation is not significantly different, but the form of the CH equation itself is more complex than that of the KdV equation. Additionally, the peakon solution in this case is a non-smooth solution, indicating that LPNN-v1 is also capable of efficiently recovering non-smooth solution.

• Case 3: Kadomtsev-Petviashvili equation

In the first two examples, we efficiently solved two classical (1+1)-dimensional integrable models using LPNN-v1. The (2+1)-dimensional integrable systems usually have three independent variables $\{x, y, t\}$, where x and y usually refer to space variables and t refers to time variable. To further verify the performance of LPNN-v1 in high-dimensional integrable systems, we consider a

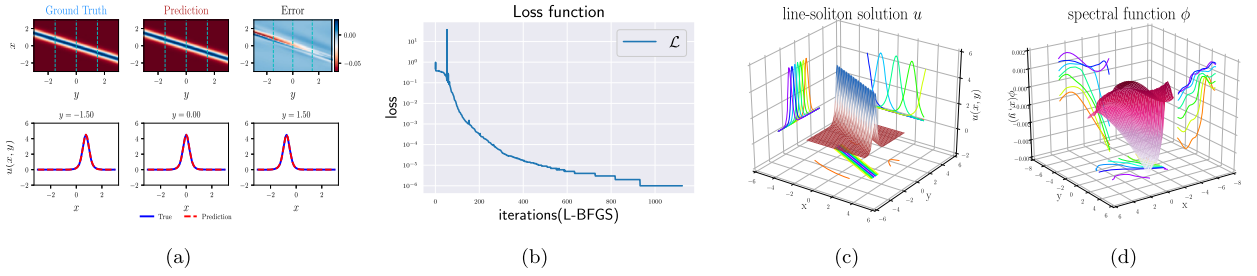


Fig. 4. (Color online) The training results of line-soliton solution $u(x, y, t)$ and spectral function $\phi(x, y, t)$ at time $t = 0$ for KP equation arising from the LPNN-v1. (a) The ground truth, prediction and error dynamics density plots, as well as sectional drawings which contain the true and prediction line-soliton solution at three distinct y -dots $y = -1.5, 0, 1.5$; (b) Evolution graph of the loss function in LPNN-v1; (c) The three-dimensional plot with contour map for the data-driven line-soliton solution; (d) The three-dimensional plot with contour map for the data-driven spectral function.

Table 3

Performance comparison between LPNN-v1 and conventional PINN for solving KP equation.

Types	$\mathbf{x} \times t$	D_{ib}, D_c	optimizer	ϕ	L^2 norm error	training time
PINN	$[-3, 3] \times [-3, 3] \times [-0.5, 0.5]$	500, 10000	L-BFGS	$\times$	1.037711×10^{-2}	4757.5630 s
LPNN-v1	$[-3, 3] \times [-3, 3] \times [-0.5, 0.5]$	500, 10000	L-BFGS	$\checkmark$	5.867296×10^{-3}	2767.3306 s

typical (2+1)-dimensional integrable PDE in this example, namely the KP equation [38]. The KP equation was derived by Kadomtsev and Petviashvili to examine the stability of the single-soliton of the KdV equation under transverse perturbations in 1970, and it is relevant for most applications in which the KdV equation arises. The KP equation can be written as

$$(u_t + 6uu_x + u_{xxx})_x + 3\sigma^2 u_{yy} = 0, \quad (3.7)$$

where $\sigma^2 = -1$ or $\sigma^2 = 1$. The Eq. (3.7) is called the KPI equation if $\sigma^2 = -1$, and the KPII equation if $\sigma^2 = 1$. Since the KP equation (3.7) degenerates into the one-dimensional KdV equation (3.1) as u is independent of y , the KP equation is the natural generalization of the KdV equation. The KP equation describes the motion of two dimensional water wave and possesses important applications in fluid mechanics and theoretical physics.

Correspondingly, we can directly write the Lax pairs of the KP equation (3.7) [25], as shown in the following equation

$$f_{Lp} : \begin{cases} \phi_y = -\sigma^{-1} \phi_{xx} - \sigma^{-1} u \phi \\ \phi_t = -4\phi_{xxx} - 6u\phi_x - (3u_x - 3\sigma \partial_x^{-1} u_y) \phi \end{cases}. \quad (3.8)$$

We can directly derive the KP equation (3.7) by utilizing the compatibility condition equation $\phi_{yt} - \phi_{ty} = 0$. Due to the complexity and peculiarity of high-dimensional integrable systems, Lax pairs of some high-dimensional integrable systems [including the KP equation (3.7) in this part and high-dimensional KdV equation (3.10) in next example] do not involve corresponding spectral parameter λ and only contain spectral function $\phi(x, y, t)$ and potential function $u(x, y, t)$. Specifically, we consider the KPII equation with taking $\sigma = 1$, then study the following initial and boundary conditions

$$\begin{aligned} u(x, y, t = -0.5) &= \frac{18e^{3x+6y+31.5}}{(1 + e^{3x+6y+31.5})^2}, \quad \mathbf{x} \in \Omega, \\ u(x, y, t) &= \frac{18e^{3x+6y-63t}}{(1 + e^{3x+6y-63t})^2}, \quad \mathbf{x} \in \partial\Omega, \quad t \in [-0.5, 0.5]. \end{aligned} \quad (3.9)$$

Here $\mathbf{x} = \{x, y\}$, $\Omega = [-3, 3] \times [-3, 3]$. The spectral function $\phi \in \mathbb{R}^{1 \times 1}$ satisfied free initial-boundary condition and initialized to $\phi(x, y, t) = 0$. We select initial and boundary points $N_{ib} = 500$ from the initial-boundary conditions (3.9), and collocation points $N_c = 10000$. After 1125 L-BFGS optimization in LPNN-v1, the relative L^2 norm error for line-soliton solution $u(x, y, t)$ reaches 5.867296×10^{-3} in 2767.3306 seconds. Generally, if the region where u is far from zero forms a band on the (x, y) plane, this kind of solution is called “line-soliton”. This does not happen in (1+1)-dimensional integrable systems.

Fig. 4 displays the numerical results obtained by solving the KP equation by applying LPNN-v1. Fig. 4(a) showcases three different density plots and sectional drawings at three distinct y -dots. In the process of LPNN-v1 optimization training using L-BFGS, the evolution figure of loss function is given in Fig. 4(b). Figs. 4(c) and Fig. 4(d) exhibit the three-dimensional plots of the line-soliton solution $u(x, y, t)$ and spectral function $\phi(x, y, t)$, respectively. Table 3 provides a detailed results comparison of using PINN and LPNN-v1 to solve the line-soliton solution of the KP equation. Surprisingly, comparing with the PINN, one can find that the LPNN-v1 not only reduces training time but also improves training accuracy by about twice. Since the Lax pairs of the KP equation does not contain spectral parameter, Table 3 only displays that LPNN-v1 can learn and obtain spectral function, which is an important numerical result that PINN cannot obtain. The numerical results indicate that LPNN-v1 is also suitable for efficiently solving high-dimensional integrable systems and their spectral problems.

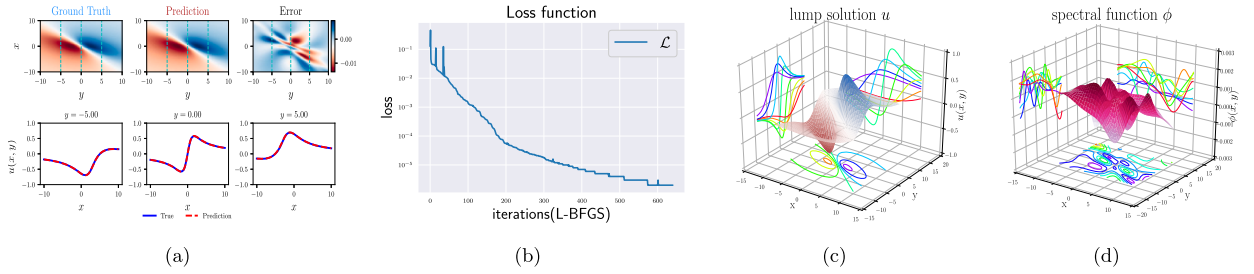


Fig. 5. (Color online) The training results of lump solution $u(x, y, t)$ and spectral function $\phi(x, y, t)$ at time $t = 0$ for high-dimensional KdV equation arising from the LPNN-v1. (a) The ground truth, prediction and error dynamics density plots, as well as sectional drawings which contain the true and prediction lump solution $u(x, y)$ at three distinct y -dots $y = -5, 0, 5$; (b) Evolution graph of the loss function in LPNN-v1; (c) The three-dimensional plot with contour map for the data-driven lump solution $u(x, y)$; (d) The three-dimensional plot with contour map for the data-driven spectral function $\phi(x, y)$.

• Case 4: High-dimensional Korteweg-de Vries equation

In this part, we further consider the following high-dimensional KdV equation

$$u_{ty} + u_{xxx}y + 3(u_y u_x)_x - u_{xx} + 2u_{yy} = 0, \quad (3.10)$$

here $\mathbf{x} = \{x, y\} \in \Omega$ indicates the 2-dimensional space in Eq. (2.1a). Wazwaz firstly put forward the new (2+1)-dimensional KdV equation, and pointed out the integrability of the new (2+1)-dimensional KdV equation is investigated via using the Painlevé test [48]. Afterwards, with the aid of Bell polynomials theory, we derived the bilinear formalism, bilinear Bäcklund transformations and Lax pairs of the (2+1)-dimensional KdV equation (3.10), and obtained the N -soliton solution base on the bilinear formalism, the lump solution and quasiperiodic wave solutions along with their asymptotic properties in Ref. [49]. Therefore, we can directly write the Lax pairs of operator form (2.1b) for the high-dimensional KdV equation (3.10), as shown below

$$f_{\text{LP}} : \begin{cases} \phi_{xy} = \frac{1}{3}\phi - \phi u_y \\ \phi_t = -3u_x \phi_x - \phi_{xxx} - 2\phi_y \end{cases}. \quad (3.11)$$

We can derive the high-dimensional KdV equation (3.10) via the compatibility condition equation $\phi_{xyt} - \phi_{txy} = 0$. Now we let spectral function $\phi(x, y, t) \in \mathbb{R}^{1 \times 1}$ satisfied free initial-boundary condition and initialized to $\phi(x, y, t) = 0$ in spatiotemporal region $[-10, 10] \times [-10, 10] \times [-5, 5]$, namely $\Omega = [-10, 10] \times [-10, 10]$. Then we consider the following initial and boundary conditions of high-dimensional KdV equation (3.10)

$$\begin{aligned} u(x, y, t = -5) &= \frac{15 + 2x + 2y}{(0.5x + y + 10)^2 + (-2.5 + 0.5x)^2 + 3}, \quad \mathbf{x} \in \Omega, \\ u(x, y, t) &= \frac{-3t + 2x + 2y}{(0.5x + y - 2t)^2 + (0.5t + 0.5x)^2 + 3}, \quad \mathbf{x} \in \partial\Omega, \quad t \in [-5, 5]. \end{aligned} \quad (3.12)$$

Owing to the 2-dimensional space of high-dimensional KdV equation (3.10), we can obtain the $\partial\Omega$ has four boundary surfaces. Then we use 400 initial and boundary points, as well as 20000 collocation points in LPNN-v1, and obtain the $9.171556\text{e-}03$ relative L^2 norm error for data-driven lump solution $u(x, y, t)$ and spectral function $\phi(x, y, t)$ of Lax pairs (3.11) by using 641 L-BFGS optimization in 963.9654 seconds. The lump solution is a special type of rational function solution in high-dimensional integrable systems, which is localized in all directions of space, it appears in various fields such as Bose-Einstein condensates, optics and marine science [50].

Fig. 5 depicts the deep learning results of the data-driven lump solution $u(x, y, t)$ and spectral function $\phi(x, y, t)$ at time $t = 0$ stemming from the LPNN-v1 for the high-dimensional KdV equation (3.10). Specifically, as shown in Fig. 5(a), we display the density plots of the true dynamics, prediction dynamics and error dynamics, then showcase its corresponding amplitude scale size on the right side of density plots, and exhibit the sectional drawings which contain the learned and true solution at three different y -dots. The evolution curve figure of the loss function arising from the LPNN-v1 is displayed in Fig. 5(b). Figs. 5(c) and (d) indicate respectively the three-dimensional plot with contour map on three planes for the predicted lump solution $u(x, y)$ and learned spectral function $\phi(x, y)$. Generally, we can also fix the x -axis [or y -axis] to visualize the relevant deep learning results for data-driven lump solution and spectral function in the (y, t) -dimension [or (x, t) -dimension].

Under the same training conditions, we also used traditional PINN to solve the high-dimensional KdV equation, and presented a comparison of the efficiency and accuracy of the two algorithms in Table 4. As shown in Table 4, compared with PINN, we can know that the LPNN-v1 decreased training time by more than twice at the cost of sacrificing training accuracy. The numerical results further demonstrate that LPNN-v1 is equally effective for solving high-dimensional integrable systems.

As is well known, the L-BFGS optimizer is a second-order optimizer and is more expensive to use. This work mainly utilized the L-BFGS optimizer for training, owing to the L-BFGS optimizer performs better overall in numerical experiments. In fact, we can also use first-order Adam optimizer or stochastic gradient descent optimizer for optimization. We also performed numerical experiments using the Adam optimizer with default settings for all the examples in this subsection, and detailed numerical re-

Table 4

Performance comparison between LPNN-v1 and conventional PINN for solving high-dimensional KdV equation.

Types	$\mathbf{x} \times \mathbf{t}$	D_b, D_c	optimizer	ϕ	L^2 norm error	training time
PINN	$[-10,10] \times [-10,10] \times [-5,5]$	400,20000	L-BFGS	$\times$	5.416409e-03	2002.7561 s
LPNN-v1	$[-10,10] \times [-10,10] \times [-5,5]$	400,20000	L-BFGS	$\checkmark$	9.171556e-03	963.9654 s

sults are attached in Tables 9–12 of Appendix A. From Table 9 in Appendix A, we can also observe that the training errors of LPNN-v1 and PINN are similar when using the Adam optimizer, but the training time of LPNN-v1 is shorter. When solving the non-smooth localized wave solution of the CH equation, it can be seen from Table 10 in Appendix A that although the training efficiency of LPNN-v1 is more than 3 times faster than that of PINN, the training error of LPNN-v1 is very large. By combining Table 2 and meticulous analysis, we argue that the optimization performance of Adam optimizer used in PINN and LPNN-v1 is worse than that of L-BFGS optimizer when solving non-smooth solutions. In addition, LPNN-v1 solved not only the non-smooth solution of the CH equation, but also the corresponding spectral problem, which may lead to a large training error using the Adam optimizer. By combining Table 11 in Appendix A and Table 3, the LPNN-v1 using the Adam optimizer achieved nearly 3x accelerations in training efficiency and nearly 40x improvements in training accuracy by comparing to the PINN using the Adam optimizer, which is similar to the training effect using the L-BFGS optimizer. From Table 12 in Appendix A and Table 4, it can be seen that when using the Adam optimizer to solve the high-dimensional KdV equation, the training time of LPNN-v1 is shorter but the training accuracy is slightly lower than that of PINN, which is consistent with the training effect of using the L-BFGS optimizer.

In this subsection, we efficiently solved the KdV equation, CH equation, KP equation and high-dimensional KdV equation using LPNN-v1 with a novel network framework. In addition to obtaining data-driven soliton solution, non-smooth peakon solution, line-soliton solution and lump solution, we also learned and obtained the corresponding spectral parameter and spectral function of spectral problems for the first time. Due to the fact that these integrable systems have simpler Lax pairs than these equations itself, the Lax pairs informed part of LPNN-v1 is simpler and more direct than the PDE informed of standard PINN, thus efficiently solving integrable systems and spectral problems. Surprisingly, our proposed LPNN-v1 significantly improves the training efficiency in solving non-smooth solution and studying high-dimensional integrable system, with training accuracy is similar to or even higher than the standard PINN. Although the training error of LPNN-v1 in Table 1 is larger than that of vanilla PINN, and the training errors of the two in Table 2 are almost the same, the major advantage of LPNN-v1 is that it requires less training time, which is suitable for some practical scenarios where accuracy is not as high but time requirements are faster. Specifically, the training time of vanilla PINN in Table 1 is more than 3 times longer than LPNN-v1, while the training time of vanilla PINN in Table 2 is more than 5 times longer than LPNN-v1. For some integrable systems with more complex equations and simpler Lax pairs, the training time of LPNN-v1 can even be more than an order of magnitude less than that of vanilla PINN. When studying certain integrable systems, LPNN-v1 can achieve higher training accuracy and shorter training time than vanilla PINN, as shown in the numerical results of training high-dimensional KP equation using LPNN-v1 in Table 3. In summary, LPNN-v1 may exhibit different training results for different research objects in terms of training accuracy. Although LPNN-v1 may also achieve higher training accuracy than vanilla PINN, its main advantage is that LPNN-v1 requires shorter training time than vanilla PINN. This is due to the fact that the training performance of LPNN-v1 generally depends more on the mathematical structure of the integrable model and its Lax pairs that we are studying. Usually, high-dimensional integrable systems themselves have a relatively complex form, while their Lax pairs have a simpler form compared to PDE itself, hence the LPNN-v1 is particularly suitable for studying high-dimensional integrable systems.

3.2. High-accuracy solving performance of LPNN-v2

In the following subsections, we delve into the data-driven localized wave solutions and spectral problems for other significant and classical integrable systems with Lax pairs by applying LPNN-v2, and provide comprehensive numerical results along with kinetic analysis.

• Case 1: Sine-Gordon equation

The SG equation is classical integrable model, which is introduced in 1939 by Frenkel and Kontorova as a model for the dynamics of crystal dislocations [39]. The SG equation has found a variety of applications, including Bloch wall dynamics in ferromagnetics and ferroelectrics, fluxon propagation in long Josephson (superconducting) junctions, self-induced transparency in nonlinear optics, spin waves in the A-phase of liquid ^3He at temperatures near to 2.6 mK, and a simple, one-dimensional model for elementary particles [51]. The SG equation and its Lax pairs [matrix form] can be represented as

$$u_{xt} = \sin(u), \quad (3.13)$$

$$f_{\text{LP}} : \begin{cases} \Phi_x = M\Phi \\ \Phi_t = N\Phi \end{cases}, M = \begin{bmatrix} -i\lambda & -\frac{1}{2}u_x \\ \frac{1}{2}u_x & i\lambda \end{bmatrix}, N = \begin{bmatrix} \frac{i}{4\lambda} \cos(u) & \frac{i}{4\lambda} \sin(u) \\ \frac{i}{4\lambda} \sin(u) & -\frac{i}{4\lambda} \cos(u) \end{bmatrix}, \quad (3.14)$$

here “i” indicates imaginary unit, and we have $u(x, t) \in \mathbb{R}^{1 \times 1}$, spectral function $\Phi(x, t) \in \mathbb{C}^{2 \times 1}$, spectral parameter $\lambda = \lambda_1 + \lambda_2 i \in \mathbb{C}$. Thus we set $\Phi(x, t) = (\phi_1(x, t), \phi_2(x, t))^T$ and $\phi_1(x, t) = \phi_{11}(x, t) + i\phi_{12}(x, t)$, $\phi_2(x, t) = \phi_{21}(x, t) + i\phi_{22}(x, t)$, in which $\phi_{ij}(x, t) \in$

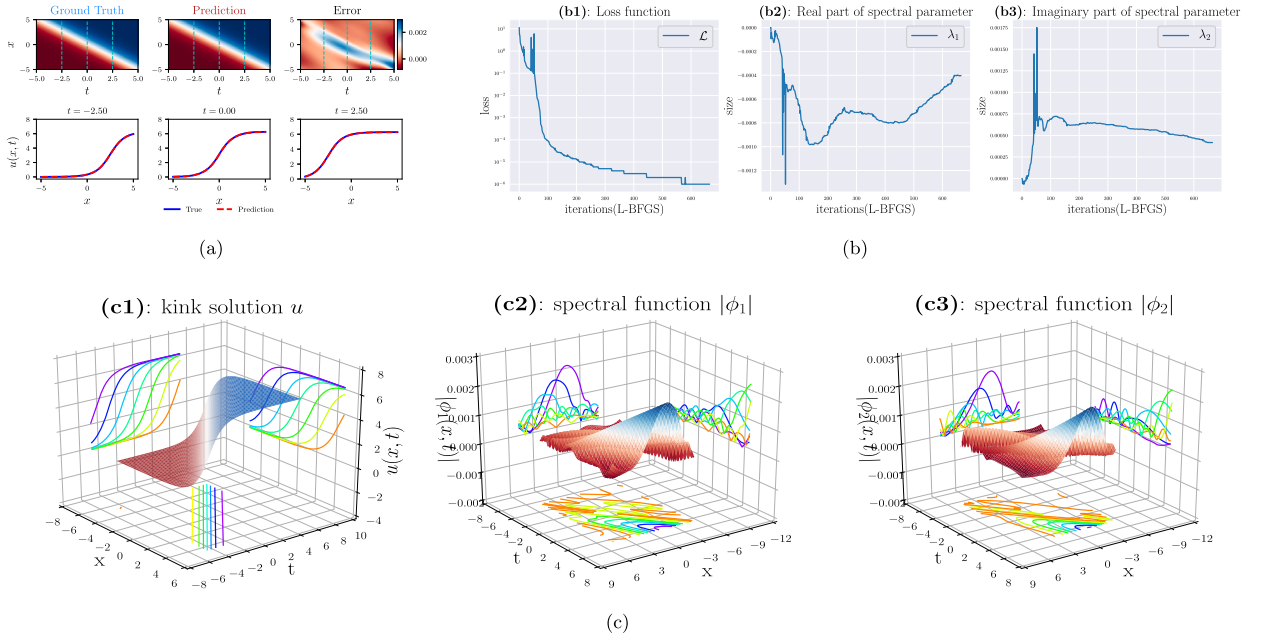


Fig. 6. (Color online) The training results of kink solution $u(x, t)$ and spectral functions $\phi_1(x, t), \phi_2(x, t)$ for SG equation arising from the LPNN-v2. (a) The ground truth, prediction and error dynamics density plots, as well as sectional drawings which contain the true and prediction kink solution at three distinct moments $t = -2.5, 0, 2.5$; (b) Evolution graphs of the loss function [panel b1] and spectral parameter $\lambda = \lambda_1 + i\lambda_2$ [panels b2 and b3] in LPNN-v2; (c) The three-dimensional plots with contour map for the data-driven kink solution [panel c1] and spectral functions [panels c2, c3] corresponding to spectral parameter $\lambda = -0.000402 + 0.000416i$.

Table 5

Performance comparison between LPNN-v2, LPNN-v1 and conventional PINN for solving SG equation.

Types	$x \times t$	D_{ib}, D_c	optimizer	λ	Φ	L^2 norm error	training time
PINN	$[-5, 5] \times [-5, 5]$	500, 10000	L-BFGS	$\times$	$\times$	$7.025371e-04$	19.8829 s
LPNN-v1	$[-5, 5] \times [-5, 5]$	500, 10000	L-BFGS	$0.004767-0.01552i$	$\checkmark$	$2.711373e-03$	38.0464 s
LPNN-v2	$[-5, 5] \times [-5, 5]$	500, 10000	L-BFGS	$-0.000402+0.000416i$	$\checkmark$	$1.962527e-04$	215.7550 s

$\mathbb{R}^{1 \times 1}[i, j = 1, 2]$. Accordingly, we can derive the SG equation (3.13) through the zero curvature equation (1.6), we let ϕ_{ij} satisfy free initial-boundary conditions and they are initialized to $\phi_{ij} = 0$, and we initialize spectral parameter λ to $0 + 0i$. Then we consider the initial and boundary conditions of SG equation in spatiotemporal region $[-5, 5] \times [-5, 5]$ as bellow

$$\begin{aligned} u(x, t = -5) &= 4 \arctan(e^{x-5}), \quad x \in [-5, 5], \\ u(-5, t) &= 4 \arctan(e^{-5+t}), \quad u(5, t) = 4 \arctan(e^{5+t}), \quad t \in [-5, 5]. \end{aligned} \quad (3.15)$$

We use 500 initial and boundary points, as well as 10000 collocation points in LPNN-v2, then obtain the $1.962527e-04$ relative L^2 norm error for data-driven kink solution $u(x, t)$ by using 666 L-BFGS optimization in 215.7550 seconds. Moreover, we also numerically learned the spectral parameter $\lambda = -0.000402 + 0.000416i$ and their corresponding spectral function $\Phi(x, t) = (\phi_1(x, t), \phi_2(x, t))^T$ in spectral problem (3.14).

Fig. 6 displays the data-driven training results of the kink solution $u(x, t)$ and spectral functions $\phi_1(x, t), \phi_2(x, t)$ by utilizing the LPNN-v2 with the initial-boundary value conditions of the SG equation. The upper panel of Fig. 6(a) depicts various dynamic density plots, including true, learned dynamics as well as error dynamics with corresponding amplitude scale size on the right side, and the bottom panel of Fig. 6(a) presents sectional drawing at different moments. The evolution curve figures for the loss function [panel (b1)] and spectral parameter [panels (b2), (b3)] arising from the LPNN-v2 with L-BFGS are displayed in Fig. 6(b). The three-dimensional plots with contour map on three planes for the data-driven kink solution and spectral functions have been displayed in Fig. 6(c), in which left panel is three-dimensional figures of kink solution $u(x, t)$, while the middle and right panel are three-dimensional figures of spectral functions $\phi_1(x, t), \phi_2(x, t)$.

Similarly, we exhibit that comparison of relative L^2 norm error and training time between LPNN-v2, LPNN-v1 and conventional PINN for solving SG equation in Table 5. From this table, we found that in the context of consistent training parameter information, traditional PINN has the shortest training time, LPNN-v2 has the highest training accuracy, while LPNN-v1 has both time and accuracy at the intermediate level. Since the Lax pairs of the SG equation is expressed in matrix form, the Φ learned and obtained by LPNNs in Table 5 is a vector spectral function, which contains ϕ_1 and ϕ_2 . Owing to the SG equation form is more concise compared

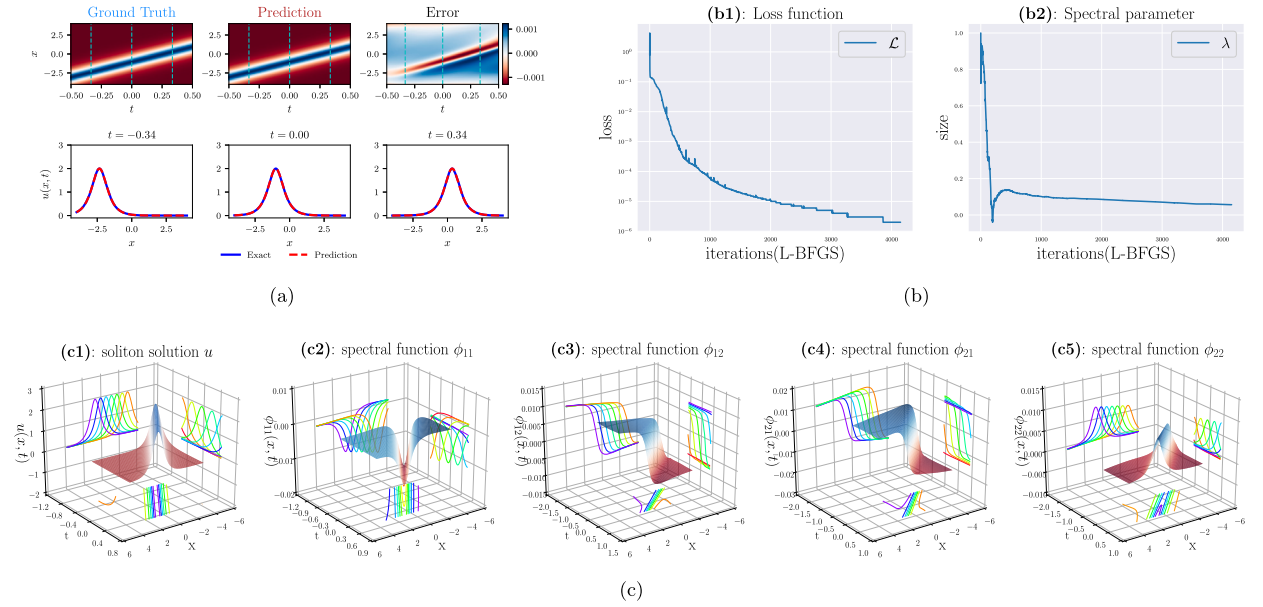


Fig. 7. (Color online) The training results of single-soliton solution $u(x, t)$ and spectral functions $\phi_{ij}(x, t)$ for mKdV equation arising from the LPNN-v2. (a) The ground truth, prediction and error dynamics density plots, as well as sectional drawings which contain the true and prediction soliton solution at three distinct moments $t = -0.34, 0, 0.34$; (b) Evolution graphs of the loss function [panel b1] and spectral parameter λ [panels b2] in LPNN-v2; (c) The three-dimensional plots with contour map for the data-driven soliton solution [panel c1] and spectral functions [panels c2-c5] corresponding to spectral parameter $\lambda = 0.056701$.

to its Lax pairs, so LPNN-v1 cannot complete efficient training, but LPNN-v2 can achieve high training accuracy. Therefore, we can see that LPNN-v2 can achieve high training accuracy, but at the same time it may consume more training time.

• Case 2: Modified KdV equation

The mKdV equation, which was derived in the study of an-harmonic lattices, is also one of the most important models in integrable systems and can be regarded as the KdV equation (3.1) with a cubic nonlinearity [40]. Moreover, the exact solutions of the KdV equation and the mKdV equation can be transformed into each other via the explicit Miura transformation, which provides an important foundation for the development of Miura transformation theory [52]. We directly present the mKdV equation and its Lax pairs, as follows:

$$u_t + 6u^2u_x + u_{xxx} = 0, \quad (3.16)$$

$$f_{Lp} : \begin{cases} \Phi_x = M\Phi \\ \Phi_t = N\Phi \end{cases}, M = \begin{bmatrix} \lambda & u \\ -u & -\lambda \end{bmatrix}, \quad (3.17)$$

$$N = \begin{bmatrix} -4\lambda^3 - 2u^2\lambda & -4u\lambda^2 - 2u_x\lambda - 2u^3 - u_{xx} \\ 4u\lambda^2 - 2u_x\lambda + 2u^3 + u_{xx} & 4\lambda^3 + 2u^2\lambda \end{bmatrix}.$$

Different from the KdV equation, the Lax pairs (3.17) of the mKdV equation are more complex compared to the mKdV equation itself, so we choose LPNN-v2 to solve and study its localized wave solution and spectral problem. Unlike the previous handling of the SG equation, we take spectral parameter $\lambda \in \mathbb{R}$ and initialize it to $\lambda = 1$. Particularly, we select spectral function $\Phi(x, t) \in \mathbb{R}^{2 \times 2}$, namely we have second-order matrix $\Phi(x, t) = [\phi_{ij}]_{2 \times 2}$, $i, j = 1, 2$. We initialize the spectral functions to $\phi_{ij} = 0$ while satisfying the free initial-boundary condition. When the spatiotemporal variables $\{x, t\} \in [-4, 4] \times [-0.5, 0.5]$, we consider the solution of the mKdV equation with Lax pairs to have the following initial and boundary conditions

$$u(x, t = -0.5) = 2\text{sech}(2x + 6), \quad x \in [-4, 4], \quad (3.18)$$

$$u(-3, t) = 2\text{sech}(-8t - 6), \quad u(3, t) = 2\text{sech}(-8t + 10), \quad t \in [-0.5, 0.5].$$

Utilizing the initial and boundary conditions (3.18), we select $N_{ib} = 400$ initial-boundary points and $N_c = 10000$ collocation points to generate the training data set for LPNN-v2. We successfully trained and obtained the data-driven single-soliton solution of the mKdV equation with high accuracy using LPNN-v2, here the relative L^2 norm error of the solution $u(x, t)$ is $7.578669e-04$, the learned spectral parameter is $\lambda = 0.056701$, and the training time and loss function iteration times of the network are 2396.8047 s and 4150 times, respectively.

We present the vivid numerical results of LPNN-v2 for solving the mKdV equation in Fig. 7. In Fig. 7(a), we display detailed density plots for the ground truth, prediction and error dynamics, as well as showcase the sectional drawings at three distinct moments corresponding to the blue dashed line in density plots. The evolution graphs of the loss function [panel b1] and spectral

Table 6

Performance comparison between LPNN-v2, LPNN-v1 and conventional PINN for solving mKdV equation.

Types	$x \times t$	D_{ib}, D_c	optimizer	λ	Φ	L^2 norm error	training time
PINN	$[-4,4] \times [-0.5,0.5]$	400,10000	L-BFGS	$\times$	$\times$	8.767018e-03	447.3520 s
LPNN-v1	$[-4,4] \times [-0.5,0.5]$	400,10000	L-BFGS	0.455595	$\checkmark$	7.656110e-01	667.0884 s
LPNN-v2	$[-4,4] \times [-0.5,0.5]$	400,10000	L-BFGS	0.056701	$\checkmark$	7.578669e-04	2396.8047 s

parameter λ [panels b2] are revealed in Fig. 7(b). Fig. 7(c1) displays the three-dimensional plot with contour map of the data-driven soliton solution $u(x, t)$, while Figs. 7(c2-c5) display the spectral functions corresponding to spectral parameter $\lambda = 0.056701$. Moreover, we provide a detailed performance comparison of using three different algorithms to solve the mKdV equation in Table 6. Due to the Lax pairs of the mKdV equation is expressed in matrix form, the Φ learned and obtained by LPNNs in Table 6 is a matrix spectral function, which contains ϕ_1, ϕ_2, ϕ_3 and ϕ_4 . From Table 6, owing to the complexity of the Lax pairs of mKdV equation, LPNN-v1 performs poorly in terms of training efficiency and accuracy. If readers prefer to use LPNN-v1, the accuracy and efficiency of LPNN-v1 can be improved by adjusting the appropriate training spatiotemporal region, resetting the hyper-parameters of NN and suitable initialization of spectral parameter. Although LPNN-v2 consumes more training time than PINN, it improves training accuracy by an order of magnitude.

• Case 3: Nonlinear Schrödinger equation

The integrable NLS equation is closely related to many nonlinear problems in fields such as nonlinear optics, plasma and Bose-Einstein condensates. Unlike the integrable models studied previously, the potential function of the NLS equation is a complex valued function, and it is also one of the most classic integrable models in integrable systems, playing a crucial role in the integrable systems theory. The NLS equations have attracted much attention after Zakharov and Shabat in 1972 constructed the matrix Lax pairs and studied the inverse scattering transform and exact solutions for the NLS equation [28]. The NLS equation with two auxiliary equations [Lax pairs] can be written as

$$iq_t + \frac{1}{2}q_{xx} + |q|^2q = 0, \quad (3.19)$$

$$f_{Lp} : \begin{cases} \Phi_x = M\Phi \\ \Phi_t = N\Phi \end{cases}, M = \begin{bmatrix} i\lambda & iq^* \\ iq & -i\lambda \end{bmatrix}, N = \begin{bmatrix} i\lambda^2 - \frac{1}{2}iqq^* & i\lambda q^* + \frac{1}{2}q_x^* \\ i\lambda q - \frac{1}{2}q_x & -i\lambda^2 + \frac{1}{2}iqq^* \end{bmatrix}, \quad (3.20)$$

where the superscript “*” represents complex conjugation, $|q|$ indicate modulus of complex-value solution $q(x, t) \in \mathbb{C}^{1 \times 1}$, then we set $q(x, t) = u(x, t) + iv(x, t)$, $u(x, t)/v(x, t) \in \mathbb{R}^{1 \times 1}$ for network training. Similarly, spectral function $\Phi(x, t) \in \mathbb{C}^{2 \times 1}$, thus we let $\Phi(x, t) = (\phi_1(x, t), \phi_2(x, t))^T$ and $\phi_1(x, t) = \phi_{11}(x, t) + i\phi_{12}(x, t)$, $\phi_2(x, t) = \phi_{21}(x, t) + i\phi_{22}(x, t)$, in which $\phi_{ij}(x, t) \in \mathbb{R}^{1 \times 1}$ [$i, j = 1, 2$]. Furthermore, spectral parameter $\lambda \in \mathbb{C}$, then we take $\lambda = \lambda_1 + \lambda_2 i$, here we have $\lambda_1 \in \mathbb{R}$ and $\lambda_2 \in \mathbb{R}$. The NLS equation (3.19) can be derived via the Lax pairs (3.20) and zero curvature equation (1.6). Next we initialize $\phi_{ij} = 0$ and make it satisfy the free initial-boundary conditions. Then according to Ref. [13,53], we provide the following initial-boundary value conditions of NLS equation in spatiotemporal region $[-3, 3] \times [-1.5, 1.5]$

$$\begin{aligned} q(x, t = -1.5) &= \left[1 - \frac{4(1 - 3i)}{4x^2 + 10} \right] e^{-1.5i}, x \in [-3, 3], \\ q(-3, t) &= \left[1 - \frac{4(1 + 2it)}{4t^2 + 37} \right] e^{it}, q(3, t) = \left[1 - \frac{4(1 + 2it)}{4t^2 + 37} \right] e^{it}, t \in [-1.5, 1.5]. \end{aligned} \quad (3.21)$$

Likewise, according to the initial and boundary conditions (3.21), we choose $N_{ib} = 400$ initial and boundary points, as well as $N_c = 10000$ collocation points in LPNN-v2. The complex-value spectral parameter λ is initialized to $-0.5 + 0.5i$, then the relative L^2 norm error reaches $1.264692e-03$ for data-driven rogue wave solution $q(x, t)$ by applying 13733 L-BFGS optimization in 6446.000261 seconds. Meanwhile, the spectral parameter of numerical discovery is $\lambda = 0.38823 + 0.018209i$, and the corresponding vector spectral function Φ in spectral problem (3.20) has learned with high accuracy from LPNN-v2. Rogue wave is a very rare and short-lived isolated large amplitude wave, characterized by “coming without a shadow, going without a trace”, and its wave height is generally more than twice that of the surrounding highest wave [54]. The NLS equation (3.19) is also the most basic mathematical model for describing rogue wave phenomena. Hitherto, rogue waves have emerged in numerous research fields, such as optics and fluid mechanics [55,56].

Fig. 8 displays the training results of the data-driven rogue wave solution and corresponding spectral problem for the NLS equation stemming from the LPNN-v2. Similarly, the abundant density plots and sectional drawings are revealed in Fig. 8(a), then Fig. 8(b) reveals the loss function curve [panel (b1)] and the numerical evolution of the real and imaginary parts of the spectral parameters λ [panels (b2) and (b3)]. The three-dimensional plots and its contour map on three planes for the data-driven rogue wave solution and spectral functions are exhibited in Fig. 8(c). We provide the performance comparison of utilizing three different methods to solve the NLS equation in Table 7. From Table 7, we can observe that LPNN-v2 has the highest training accuracy after consuming more training time, while LPNN-v1 has the shortest training time but lower training accuracy. Compared with traditional PINN, LPNN-v2 has improved training accuracy by more than twice. Certainly, we can further improve the training accuracy of LPNN-v1 and LPNN-v2 by readjusting the training spatiotemporal region, resetting hyper-parameters and spectral parameter.

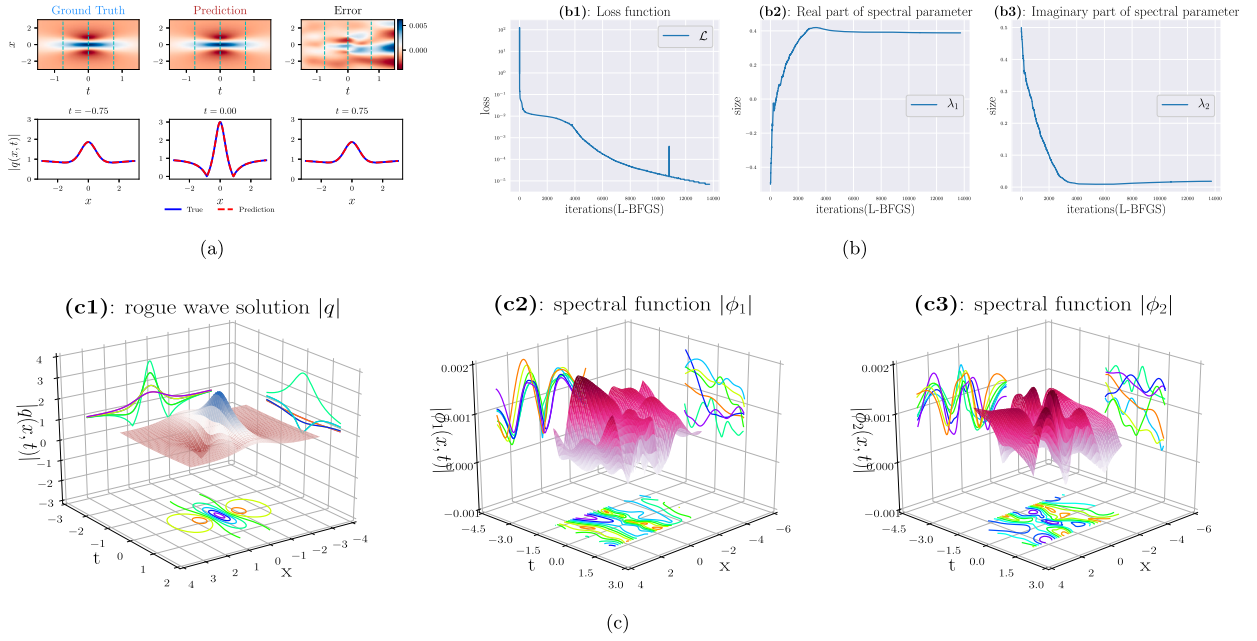


Fig. 8. (Color online) The training results of rogue wave solution $|q(x,t)|$ and spectral functions $|\phi_1(x,t)|$, $|\phi_2(x,t)|$ for NLS equation arising from the LPNN-v2. (a) The ground truth, prediction and error dynamics density plots, as well as sectional drawings which contain the true and prediction rogue wave solution at three distinct moments $t = -0.75, 0, 0.75$; (b) Evolution graphs of the loss function [panel b1] and spectral parameter $\lambda = \lambda_1 + i\lambda_2$ [panels b2, b3] in LPNN-v2; (c) The three-dimensional plots with contour map for the data-driven rogue wave solution [panel c1] and spectral functions [panels c2, c3] corresponding to spectral parameter $\lambda = 0.38823 + 0.018209i$.

Table 7

Performance comparison between LPNN-v2, LPNN-v1 and conventional PINN for solving NLS equation.

Types	$\mathbf{x} \times t$	$D_{\text{tr}}, D_{\text{c}}$	optimizer	λ	Φ	L^2 norm error	training time
PINN	$[-3,3] \times [-1.5,1.5]$	100,10000	L-BFGS	$\times$	$\times$	3.375881e-03	2911.307015 s
LPNN-v1	$[-3,3] \times [-1.5,1.5]$	100,10000	L-BFGS	$-0.374518 + 0.393163i$	$\checkmark$	4.112298e-01	979.621199 s
LPNN-v2	$[-3,3] \times [-1.5,1.5]$	100,10000	L-BFGS	$0.38823 + 0.018209i$	$\checkmark$	1.264692e-03	6446.000261 s

• Case 4: Short pulse equation

The short pulse (SP) equation

$$u_{xt} = u + \frac{1}{6}(u^3)_{xx} \quad (3.22)$$

is proposed by Schäfer and Wayne [42] as an alternative (to the NLS equation) model for approximating the evolution of ultrashort intense infrared pulses in silica optical, in which u represents the dimensionless electric field, t and x standing for time-like and space-like independent variables, respectively. Subsequent numerical analyses have demonstrated that the SP equation as an approximation of the Maxwell's equation has superior applicability, particularly when characterizing the propagation of extremely short-duration light pulses [57]. This equation was initially introduced as one of Rabelo's equations in the realm of differential geometry [58]. Furthermore, aforementioned KdV equation, mKdV equation, SG equation and NLS equation all belong to the AKNS hierarchy [30], while the SP equation is the most classical integrable model in WKI equations and possesses a Lax pairs of the WKI type [41], and it can be transformed into the integrable SG equation [59]. One can obtain the Lax pairs of SP equation, as shown in following

$$f_{\text{Lp}} : \begin{cases} \Phi_x = M\Phi \\ \Phi_t = N\Phi \end{cases}, M = \begin{bmatrix} \lambda & \lambda u_x \\ \lambda u_x & -\lambda \end{bmatrix}, N = \begin{bmatrix} \frac{1}{2}\lambda u^2 + \frac{1}{4\lambda} & \frac{1}{2}\lambda u^2 u_x - \frac{1}{2}u \\ \frac{1}{2}\lambda u^2 u_x + \frac{1}{2}u & -\frac{1}{2}\lambda u^2 - \frac{1}{4\lambda} \end{bmatrix}. \quad (3.23)$$

The SP equation (3.22) can be derived by utilizing the zero curvature equation (1.6) along with Lax pairs (3.23). Similar to case 3, we set spectral function $\Phi(x,t) \in \mathbb{C}^{2 \times 1}$ and spectral parameter $\lambda \in \mathbb{C}$, thus we take $\lambda = \lambda_1 + \lambda_2 i$, $\Phi(x,t) = (\phi_1(x,t), \phi_2(x,t))^T$ and $\phi_1(x,t) = \phi_{11}(x,t) + i\phi_{12}(x,t)$, $\phi_2(x,t) = \phi_{21}(x,t) + i\phi_{22}(x,t)$, where $\lambda_1/\lambda_2 \in \mathbb{R}$ and $\phi_{ij}(x,t) \in \mathbb{R}^{1 \times 1}$ [$i, j = 1, 2$]. Next we initialize $\phi_{ij} = 0$ and make it satisfy the free initial-boundary conditions, then we refer to Ref. [60] and consider the initial and boundary value conditions of SP equation in spatiotemporal region $[-6, 6] \times [-3, 3]$, as shown in following

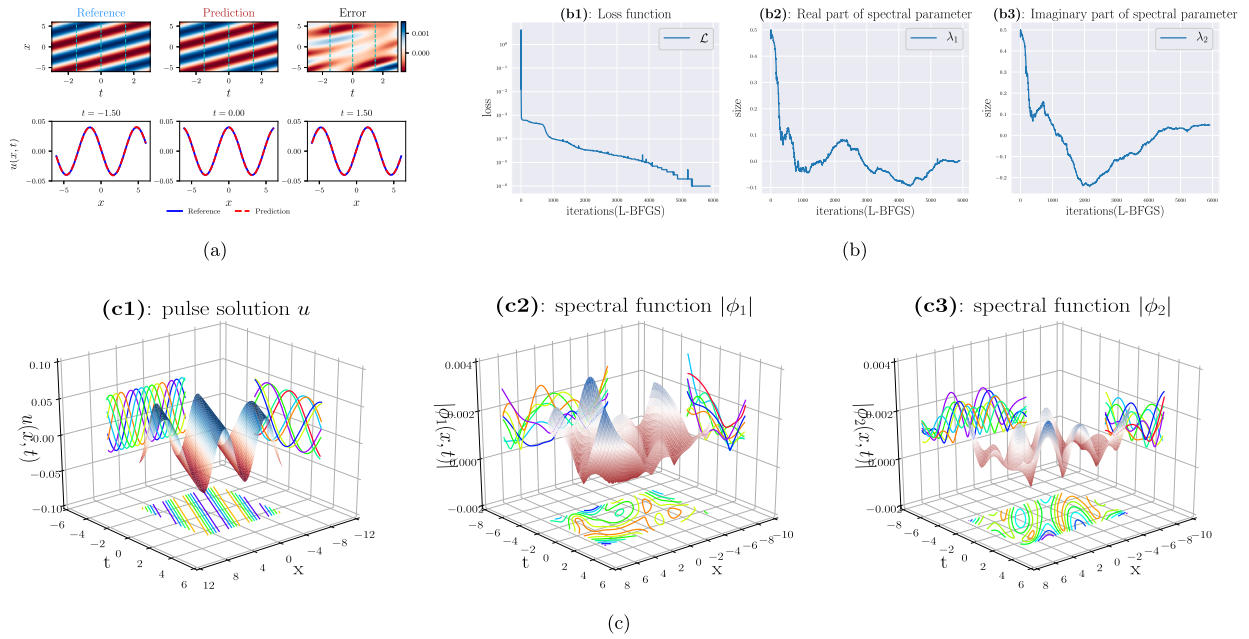


Fig. 9. (Color online) The training results of pulse solution $u(x, t)$ and spectral functions $|\phi_1(x, t)|, |\phi_2(x, t)|$ for SP equation arising from the LPNN-v2. (a) The ground truth, prediction and error dynamics density plots, as well as sectional drawings which contain the reference and prediction pulse solution at three distinct moments $t = -1.5, 0, 1.5$; (b) Evolution graphs of the loss function [panel b1] and spectral parameter [panels b2, b3] in LPNN-v2; (c) The three-dimensional plots with contour map for the data-driven pulse solution [panel c1] and spectral functions [panels c2, c3] corresponding to spectral parameter $\lambda = 0.002629 + 0.047998i$.

Table 8

Performance comparison between LPNN-v2, LPNN-v1 and conventional PINN for solving SP equation.

Types	$\mathbf{x} \times \mathbf{t}$	D_{ib}, D_c	optimizer	λ	Φ	L^2 norm error	training time
PINN	$[-6, 6] \times [-3, 3]$	400, 10000	L-BFGS	$\times$	$\times$	$3.432290e-02$	577.9960 s
LPNN-v1	$[-6, 6] \times [-3, 3]$	400, 10000	L-BFGS	$0.037306 + 0.040709i$	$\checkmark$	$1.664159e+00$	532.3504 s
LPNN-v2	$[-6, 6] \times [-3, 3]$	400, 10000	L-BFGS	$0.002629 + 0.047998i$	$\checkmark$	$1.528543e-02$	2400.7532 s

$$\begin{aligned}
 u(x, t = -3) &= \frac{0.04 \cos(x + 3)}{\cosh(0.01x - 0.03)}, \quad x \in [-6, 6], \\
 u(-6, t) &= \frac{0.04 \cos(-6 - t)}{\cosh(-0.06 + 0.01t)}, \quad u(6, t) = \frac{0.04 \cos(6 - t)}{\cosh(0.06 + 0.01t)}, \quad t \in [-3, 3].
 \end{aligned} \tag{3.24}$$

From aforementioned initial and boundary value conditions (3.24), we select $N_{ib} = 400$ and $N_c = 10000$ for training of LPNN-v2, while the spectral parameter was initialized as $\lambda = 0.5 + 0.5i$. After 5913 L-BFGS optimization in LPNN-v2, the relative L^2 norm error reaches $1.528543e-02$ in 2400.7532 seconds, and the optimal spectral parameter learned is $\lambda = 0.002629 + 0.047998i$. As shown in Ref. [60], the data-driven solution learned is been called the pulse solution, it represents a single-valued nonsingular pulse or a wave packet, and the shape of the pulse is similar to that of the NLS soliton: the sech-shaped envelope modulates the cos-shaped wave. Moreover, owing to the initial and boundary training points in our training data come from approximate pulse solution of the SP equation (3.22), we can find that the L^2 error trained by LPNN is relatively large. Based on the high-precision characteristics of LPNN-v2, we have reason to believe that the pulse solution learned in this part is more realistic.

The training results of the data-driven pulse solution stemming from the LPNN-v2 are indicated in Fig. 9, in which Fig. 9(a) displays the abundant density plots and sectional drawings. Fig. 9(b) shows the loss function curve in panel (b1), and the numerical evolutions for the real and imaginary parts of the spectral parameter λ with network training are shown in (b2) and (b3), respectively. While Fig. 9(c) showcases the three-dimensional plot and its contour map on three planes for pulse solution $u(x, t)$ and spectral functions ϕ_1, ϕ_2 . Table 8 exhibits a detailed comparison of training results by applying three algorithms to solve the SP equation. Similar to the numerical results of solving the NLS equation in case 3, LPNN-v2 has improved training accuracy by more than twice compared to the traditional PINN algorithm. Due to the fact that the Lax pairs of the SP equation are more complicated compared to the SP equation itself, so LPNN-v1 has poor training accuracy in solving the SP equation. Therefore, as solving this type of integrable systems with intricate Lax pairs, we usually recommend using LPNN-v2 to obtain higher training accuracy.

Similarly, we performed numerical experiments using the Adam optimizer for all the examples in this subsection, and detailed numerical results are attached in Tables 13-16 of Appendix B. From Table 13 in Appendix B, we can observe that the training error of LPNN-v2 is the smallest among the numerical results of solving the SG equation using the Adam optimizer in the three neural networks, which is similar to the numerical results of the three neural networks using the L-BFGS optimizer in Table 5. In Table 14 of

Appendix B, according to numerical results of solving mKdV equation with Adam optimizer for three kinds of neural networks, we can obviously observe that the training error of LPNN-v2 is larger than that of PINN, which seems to violate the high-accuracy training feature of LPNN-v2. Since the Adam optimizer terminates the training iteration by setting the maximum number of iterations, and the loss function of LPNN-v2 is much more complex than that of PINN, Table 14 also showcases the numerical results of LPNN-v2 after 200,000 Adam iterations. The numerical results indicate that the training accuracy of LPNN-v2 is further improved, but the improvement is not significant due to the influence of highly nonlinear terms in the mKdV equation. However, by combining Table 6, we find that PINN and LPNN-v2 have much higher training accuracy when using the L-BFGS optimizer than when using the Adam optimizer. Therefore, we believe that LPNN-v2 using Adam optimizer is difficult to present high-accuracy training characteristic when solving integrable systems with highly nonlinear terms. According to Table 15 in Appendix B and Table 7, once training intricate rogue wave of NLS equation, the training error values of LPNN-v2 and PINN are very close by applying 60000 Adam iterations, so the advantage of higher training accuracy of LPNN-v2 is not obvious. Similar to the previous case, Table 15 also shows the numerical results of LPNN-v2 after 100,000 Adam iterations, and the numerical results show that the training accuracy of LPNN-v2 is further improved, although the improvement is not very dramatic. Therefore, we argue that the training effect of the L-BFGS optimizer is better than that of the Adam optimizer when solving complex localized wave solutions of integrable systems. Finally, from Table 16 in Appendix B and Table 8, we can see that the training accuracy of LPNN-v2 using the Adam optimizer is higher than that of PINN, which is similar to the numerical results of the two neural networks using the L-BFGS optimizer. In summary, we can conclude that using the lower cost Adam optimizer, then LPNNs can still outperform standard PINNs in most training scenarios. However, in some special training scenarios, LPNNs can outperform standard PINNs only when using second-order L-BFGS optimizers, such as simulating non-smooth solution and intricate rogue wave, and solving integrable systems with highly nonlinear terms.

In this subsection, we further introduce compatibility condition/zero curvature equation constraints into Lax pairs informed based on LPNN-v1, then propose LPNN-v2 that can achieve higher accuracy. For practical problems requiring higher training accuracy and studying more complex localized wave solutions, we can generally utilize LPNN-v2 for training. We investigated the data-driven localized wave solutions and spectral problems of several important integrable systems by means of LPNN-v2, and found that LPNN-v2 achieved higher training accuracy than standard PINN and LPNN-v1 by sacrificing training time. Specifically, we employed the LPNN-v2 to solve data-driven localized wave solutions and their spectral problems for the SG equation, mKdV equation, NLS equation and SP equation, where localized wave solutions include kink solution, soliton solution, rogue wave solution and pulse solution. Strikingly, the training accuracy of LPNN-v2 in Table 6 can be improved by an order of magnitude. Moreover, the training accuracy of LPNN-v2 also depends on the mathematical structure of the integrable system and its Lax pairs, and LPNN-v2 may have limited improvement in training accuracy when studying extremely complex scenarios of integrable systems and their Lax pairs, owing to the loss function of the neural network is too intricate, which may generate negative impact on the training performance. The numerical results indicate that the LPNN-v2 has a wider range of applications than LPNN-v1, and can usually be utilized to study all integrable systems with Lax pairs with high accuracy.

The main goal of this work is to propose a pure LPNNs, similar to the PINN proposed by Rassi et al. [2], upon which various strategies can be applied for improvement according to specific research needs. Specifically, the LPNNs proposed in this paper are designed based on the PINN architecture, thus besides the TensorFlow framework, we can also implement LPNNs in the JAX framework [61], achieving efficient and high-precision training performance as well as learning and obtaining spectral parameter and spectral function. Notably, the Taylor-mode automatic differentiation (Taylor-mode AD) based on the `jax.jet` module in JAX is implemented using truncated Taylor polynomial approximation [62], allowing effective computation of high-order derivatives of function compositions through the *faà di Bruno* formula [63]. Numerical results indicated that PINNs using Taylor-mode AD exhibit higher training efficiency compared to those using traditional forward-mode or reverse-mode automatic differentiation (AD), with efficiency increasing significantly with the order of differentiation [64]. Lax pairs serve as advantageous mathematical tools to reduce the order of high-order derivatives in integrable systems, although high-order derivative terms of the potential function may still exist in Lax pairs, such as those Lax pairs (3.5) and (3.17) of this paper which contain second-order derivative terms. Additionally, since the benchmarks in this paper are classical low-order integrable models, the Lax pairs of more complex higher-order integrable models may involve higher-order derivative terms, such as the Lax pairs of seventh-order KdV equation in Ref. [65]. Taylor-mode AD efficiently calculates high-order derivatives in PDEs using clever automatic differentiation techniques to accelerate training, while LPNNs utilize the alternative mathematical structures (i.e., Lax pairs) of integrable systems to simplify or enhance the training objectives of networks, thereby achieving efficient or high-precision training performance. Furthermore, the proposed LPNNs, besides their characteristics of efficiency and high precision, can also learn and obtain spectral parameter and spectral function, which play a crucial role in integrable system theory and serve as the research basis for future work.

For demonstrating that LPNN-v1 based on the JAX framework can further improve training efficiency after using Taylor-mode AD, we use the CH equation (3.4) and its Lax pairs (3.5) as example, we consider the initial and boundary conditions (3.6) in the spatiotemporal domain $[0, 0.1] \times [-3, 3]$, then obtain the training results of PINN and LPNN-v1 using Taylor-mode AD and traditional AD in JAX, and display detailed numerical results in Table 17 and Fig. 10 of Appendix C. From Table 17, it is evident that under fixed spectral parameter and associated hyper-parameter settings, LPNN-v1 using Adam optimizer demonstrates faster training efficiency compared to PINN while sacrificing training accuracy. Moreover, LPNN-v1 employing Taylor-mode AD exhibits faster training than LPNN-v1 utilizing conventional AD, which is attributed to the second-order derivative terms of the potential function in Lax pairs (3.5). Furthermore, Fig. 10 illustrates a comparative chart of training time between PINN and LPNN-v1 with varying numbers of neurons and hidden layers. It is observed that with a fixed number of hidden layers, increasing the number of neurons per hidden layer results in longer training times, with LPNN-v1 employing Taylor-mode AD exhibiting the shortest training time. Similarly, with a fixed number of neurons per hidden layer, an increase in the number of hidden layers leads to longer training times, with LPNN-v1

employing Taylor-mode AD again showing the shortest training time. In summary, although Taylor-mode AD can enhance the training efficiency of PINN, the extent of its improvement greatly depends on the order and quantity of high-order derivative terms in the PDE. However, Taylor-mode AD can enhance the training efficiency of LPNN-v1, with its improvement not only contingent upon the order and quantity of high-order derivative terms in Lax pairs, but also on their own mathematical structures of integrable systems and their Lax pairs. Consequently, LPNN-v1 utilizing Taylor-mode AD is more likely to significantly enhance training efficiency, especially when the greater the number of neurons and hidden layers, the more pronounced the improvement in training efficiency, as displayed in Fig. 10. Moreover, one can also introduce Taylor-mode AD and various other improvement strategies based on the JAX framework into the LPNNs (including LPNN-v1 and LPNN-v2) to achieve various training performance expected by users.

4. Conclusions

In this article, we introduce Lax pairs, which is the most important feature of integrable systems, into deep NN algorithms and propose the LPNNs suitable for integrable systems. LPNN-v1 relies entirely on the Lax pairs of integrable systems and introduces Lax pairs informed into the loss function, which can efficiently study the localized wave solutions and spectral problems of certain integrable systems. Based on the LPNN-v1, the Lax pairs constraint and loss function of LPNN-v2 also depend on the compatibility condition equation or zero curvature equation of the integrable systems, which can accurately solve all integrable systems with high-accuracy and obtain the corresponding spectral parameter and spectral function in the spectral problem.

This article designs several deep learning numerical experiments for important integrable systems with Lax pairs. Specifically, the KdV equation, CH equation, KP equation and high-dimensional KdV equation were efficiently solved by utilizing LPNN-v1, and the data-driven soliton solution, non-smooth peakon solution, line-soliton solution, lump solution, spectral parameter and corresponding spectral function were learned and obtained, here the training time of LPNN-v1 was more than two to five times faster than that of the standard PINN. The data-driven localized wave solutions and their corresponding spectral problems of the SG equation, mKdV equation, NLS equation and SP equation were studied with high precision by means of LPNN-v2, in which the data-driven localized wave solutions include kink solution, soliton solution, rogue wave solution and pulse solution. LPNN-v2 achieved higher accuracy than standard PINN and LPNN-v1, and it is suitable for all integrable systems with Lax pairs. Generally, setting different spectral parameter initialization in LPNNs often leads to different spectral function, training time and training accuracy, which often need to be adjusted based on results and experience in practical numerical experiments.

By means of replacing or assisting the PDE in Lax pairs informed network, we introduce the most important feature of integrable systems, namely Lax pairs, into deep NNs in the first time. Then we propose novel LPNNs suitable for the integrable systems with Lax pairs, in which LPNN-v1 is generally superior to standard PINN in terms of training efficiency, while LPNN-v2 usually outperforms vanilla PINN in terms of training accuracy, namely the training performance of LPNNs can be superior to vanilla PINN in studying practical problems of integrable systems with Lax pairs. Since LPNNs rely heavily on the mathematical structure of integrable systems and Lax pairs, it must be pointed out that we cannot guarantee that the training efficiency and accuracy of LPNNs simultaneously outperform vanilla PINN in any scenario. Nevertheless, we provide a novel deep learning method that not only cleverly utilizes Lax pairs to achieve more efficient and accurate training results from the mathematical structure of integrable system, but also obtains spectral parameter and spectral function in spectral problem, they play a crucial role in the classical integrable system theory, such as inverse scattering theory and Darboux transformation theory. The study of numerical spectral problems in integrable systems using deep learning methods has not yet been reported, so the research results of this work will be highly distinctive and unique. The research work of this article provides new idea, approach and pathway for solving data-driven localized wave solutions and spectral problems, and discovering novel localized wave solutions and even new Lax pairs for integrable systems.

CRedit authorship contribution statement

Juncai Pu: Writing – review & editing, Writing – original draft, Software, Methodology, Data curation, Conceptualization. **Yong Chen:** Writing – review & editing, Supervision, Project administration, Methodology.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

We thank Academician Song Jiang of Chinese Academy of Science for many useful conversations and valuable suggestions. The authors gratefully acknowledge the support of the National Natural Science Foundation of China (No. 12175069 and No. 12235007), Science and Technology Commission of Shanghai Municipality (No. 21JC1402500 and No. 22DZ2229014), and Natural Science Foundation of Shanghai (No. 23ZR1418100).

Appendix A. The training results of solving integrable systems by utilizing Adam optimizer in the LPNN-v1 and PINN

After utilizing Adam optimizer in the LPNN-v1 and standard PINN, the training results of solving integrable systems are shown as follows Tables 9-12.

Table 9

Performance comparison of solving KdV equation by utilizing Adam optimizer in the LPNN-v1 and standard PINN.

Types	$\mathbf{x} \times t$	D_{ib}, D_c	Adam	λ	ϕ	L^2 norm error	training time
PINN	$[-5,5] \times [-5,5]$	400,10000	12000	$\times$	$\times$	5.287849e-03	352.5421 s
LPNN-v1	$[-5,5] \times [-5,5]$	400,10000	12000	1.0	$\sqrt{\quad}$	6.036824e-03	165.5438 s

Table 10

Performance comparison of solving CH equation by utilizing Adam optimizer in the LPNN-v1 and standard PINN.

Types	$\mathbf{x} \times t$	D_{ib}, D_c	Adam	λ	ϕ	L^2 norm error	training time
PINN	$[-5,5] \times [0,2]$	800,10000	40000	$\times$	$\times$	8.451458e-02	1919.9885 s
LPNN-v1	$[-5,5] \times [0,2]$	800,10000	40000	0.399974	$\sqrt{\quad}$	7.185118e-01	863.6540 s

Table 11

Performance comparison of solving KP equation by utilizing Adam optimizer in the LPNN-v1 and standard PINN.

Types	$\mathbf{x} \times t$	D_{ib}, D_c	Adam	ϕ	L^2 norm error	training time
PINN	$[-3,3] \times [-3,3] \times [-0.5,0.5]$	500,10000	30000	$\times$	2.328239e-01	6007.5784 s
LPNN-v1	$[-3,3] \times [-3,3] \times [-0.5,0.5]$	500,10000	30000	$\sqrt{\quad}$	5.900874e-03	2128.5921 s

Table 12

Performance comparison of solving high-dimensional KdV equation by utilizing Adam optimizer in the LPNN-v1 and standard PINN.

Types	$\mathbf{x} \times t$	D_{ib}, D_c	Adam	ϕ	L^2 norm error	training time
PINN	$[-10,10] \times [-10,10] \times [-5,5]$	400,20000	20000	$\times$	3.890484e-03	6187.1247 s
LPNN-v1	$[-10,10] \times [-10,10] \times [-5,5]$	400,20000	20000	$\sqrt{\quad}$	6.323496e-03	2507.5781 s

Appendix B. The training results of solving integrable systems by utilizing Adam optimizer in the LPNN-v2, LPNN-v1 and PINN

After utilizing Adam optimizer in the LPNN-v2, LPNN-v1 and standard PINN, the training results of solving integrable systems are shown as follows Tables 13-16.

Table 13

Performance comparison of solving SG equation by utilizing Adam optimizer in the LPNN-v2, LPNN-v1 and standard PINN.

Types	$\mathbf{x} \times t$	D_{ib}, D_c	Adam	λ	Φ	L^2 norm error	training time
PINN	$[-5,5] \times [-5,5]$	500,10000	50000	$\times$	$\times$	1.241263e-03	465.0311 s
LPNN-v1	$[-5,5] \times [-5,5]$	500,10000	50000	-0.005755-0.000759i	$\sqrt{\quad}$	2.807519e-03	1182.7721 s
LPNN-v2	$[-5,5] \times [-5,5]$	500,10000	50000	0.000975-0.003289i	$\sqrt{\quad}$	2.987313e-04	1545.2205 s

Table 14

Performance comparison of solving mKdV equation by utilizing Adam optimizer in the LPNN-v2, LPNN-v1 and standard PINN.

Types	$\mathbf{x} \times \mathbf{t}$	D_{ib}, D_c	Adam	λ	Φ	L^2 norm error	training time
PINN	$[-4,4] \times [-0.1,0.1]$	400,10000	150000	$\times$	$\times$	4.915344e-02	4454.1543 s
LPNN-v1	$[-4,4] \times [-0.1,0.1]$	400,10000	150000	0.004356	$\checkmark$	4.495894e-01	6313.4711 s
LPNN-v2	$[-4,4] \times [-0.1,0.1]$	400,10000	150000	-0.008034	$\checkmark$	5.903415e-02	7322.0335 s
LPNN-v2	$[-4,4] \times [-0.1,0.1]$	400,10000	200000	0.00347	$\checkmark$	4.264126e-02	9351.7990 s

Table 15

Performance comparison of solving NLS equation by utilizing Adam optimizer in the LPNN-v2, LPNN-v1 and standard PINN.

Types	$\mathbf{x} \times \mathbf{t}$	D_{ib}, D_c	Adam	λ	Φ	L^2 norm error	training time
PINN	$[-3,3] \times [-1.5,1.5]$	100,10000	60000	$\times$	$\times$	2.014013e-02	1255.2755 s
LPNN-v1	$[-3,3] \times [-1.5,1.5]$	100,10000	60000	0.018866-0.023404i	$\checkmark$	5.003518e-01	1607.8049 s
LPNN-v2	$[-3,3] \times [-1.5,1.5]$	100,10000	60000	-0.014245-0.023910i	$\checkmark$	1.958669e-02	2462.2456 s
LPNN-v2	$[-3,3] \times [-1.5,1.5]$	100,10000	100000	0.0091-0.02619i	$\checkmark$	1.548688e-02	6659.3198 s

Table 16

Performance comparison of solving SP equation by utilizing Adam optimizer in the LPNN-v2, LPNN-v1 and standard PINN.

Types	$\mathbf{x} \times \mathbf{t}$	D_{ib}, D_c	Adam	λ	Φ	L^2 norm error	training time
PINN	$[-6,6] \times [-3,3]$	400,10000	30000	$\times$	$\times$	3.565266e-02	553.2875 s
LPNN-v1	$[-6,6] \times [-3,3]$	400,10000	30000	-0.033059-0.010942i	$\checkmark$	1.584708e+00	842.4056 s
LPNN-v2	$[-6,6] \times [-3,3]$	400,10000	30000	-0.003506+0.008045i	$\checkmark$	2.783789e-02	1173.6363 s

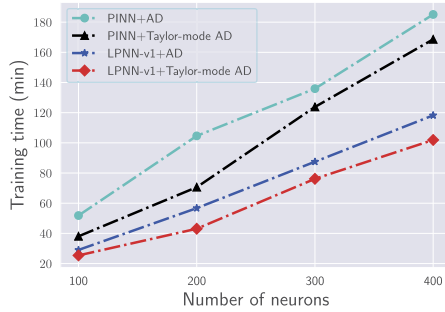
Appendix C. The training results of solving CH equation by utilizing Taylor-mode AD and AD in the LPNN-v1 and PINN

After utilizing Taylor-mode AD and AD in the LPNN-v1 and PINN, the training results of solving CH equation are shown as follows Table 17 and Fig. 10.

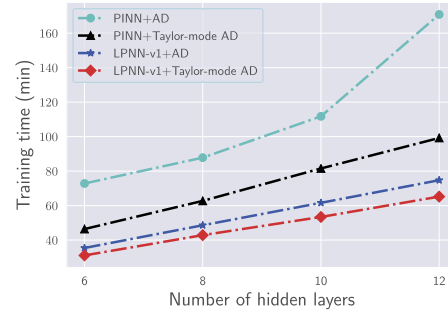
Table 17

Performance comparison of solving CH equation by utilizing Taylor-mode AD and AD in the LPNN-v1 and PINN.

Types	L, N_d	D_{ib}, D_c	Adam	λ	ϕ	L^2 norm error	training time
PINN+AD	6,100	800,20000	40000	$\times$	$\times$	2.381e-02	3110 s
PINN+Taylor-mode AD	6,100	800,20000	40000	$\times$	$\times$	2.420e-02	2283 s
LPNN-v1+AD	6,100	800,20000	40000	1.5	$\checkmark$	5.602e-02	1742 s
LPNN-v1+Taylor-mode AD	6,100	800,20000	40000	1.5	$\checkmark$	6.514e-02	1524 s



(a)



(b)

Fig. 10. (Color online) The comparison chart of training time for solving CH equation by utilizing Taylor-mode AD and AD in the LPNN-v1 and PINN. (a) The comparison chart of training time for PINN and LPNN-v1 applying 5 hidden layers with different numbers of neurons per hidden layer; (b) The comparison chart of training time for PINN and LPNN-v1 applying different numbers of hidden layers with 100 neurons per hidden layer.

References

- [1] A. Scott, Encyclopedia of Nonlinear Science, Routledge Press, London, 2004.
- [2] M. Raissi, P. Perdikaris, G.E. Karniadakis, Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, *J. Comput. Phys.* 378 (2019) 686–707.
- [3] J. Sirignano, K. Spiliopoulos, DGM: a deep learning algorithm for solving partial differential equations, *J. Comput. Phys.* 375 (2018) 1339–1364.
- [4] W. E, B. Yu, The deep Ritz method: a deep learning-based numerical algorithm for solving variational problems, *Commun. Math. Stat.* 6 (2018) 1–12.
- [5] S. Jin, Z. Ma, K. Wu, Asymptotic-preserving neural networks for multiscale time-dependent linear transport equations, *J. Sci. Comput.* 94 (2023), vol. 57.
- [6] J.R. Chen, X.R. Chi, W. E, Z.W. Yang, Bridging traditional and machine learning-based algorithms for solving PDEs: the random feature method, *J. Mach. Learn.* 1 (2022) 268–298.
- [7] S.C. Dong, Z.W. Li, Local extreme learning machines and domain decomposition for solving linear and nonlinear partial differential equations, *Comput. Methods Appl. Mech. Eng.* 387 (2021) 114129.
- [8] L. Lu, P.Z. Jin, G.F. Pang, Z.Q. Zhang, G.E. Karniadakis, Learning nonlinear operators via deepnet based on the universal approximation theorem of operators, *Nat. Mach. Intell.* 3 (2021) 218–229.
- [9] S.F. Wang, H.W. Wang, P. Perdikaris, On the eigenvector bias of Fourier feature networks: from regression to solving multi-scale PDEs with physics-informed neural networks, *Comput. Methods Appl. Mech. Eng.* 384 (2021) 113938.
- [10] H.Y. Li, S. Jiang, W.J. Sun, L.W. Xu, G.Y. Zhou, A model-data asymptotic-preserving neural network method based on micro-macro decomposition for gray radiative transfer equations, *Commun. Comput. Phys.* (2023), in press.
- [11] J.C. Pu, Y. Chen, Complex dynamics on the one-dimensional quantum droplets via time piecewise PINNs, *Physica D* 454 (2023) 133851.
- [12] J. Li, Y. Chen, Solving second-order nonlinear evolution partial differential equations using deep learning, *Commun. Theor. Phys.* 72 (2020) 105005.
- [13] J.C. Pu, J. Li, Y. Chen, Soliton, breather and rogue wave solutions for solving the nonlinear Schrödinger equation using a deep learning method with physical constraints, *Chin. Phys. B* 30 (2021) 060202.
- [14] J.C. Pu, J. Li, Y. Chen, Solving localized wave solutions of the derivative nonlinear Schrödinger equation using an improved PINN method, *Nonlinear Dyn.* 105 (2021) 1723–1739.
- [15] J.C. Pu, Y. Chen, Data-driven vector localized waves and parameters discovery for Manakov system using deep learning approach, *Chaos Solitons Fractals* 160 (2022) 112182.
- [16] S.N. Lin, Y. Chen, Physics-informed neural network methods based on Miura transformations and discovery of new localized wave solutions, *Physica D* 445 (2023) 133629.
- [17] H.J. Zhou, J.C. Pu, Y. Chen, Data-driven forward-inverse problems for the variable coefficients Hirota equation using deep learning method, *Nonlinear Dyn.* 111 (2023) 14667–14693.
- [18] J.C. Pu, Y. Chen, Data-driven forward-inverse problems for Yajima-Oikawa system using deep learning with parameter regularization, *Commun. Nonlinear Sci. Numer. Simul.* 118 (2023) 107051.
- [19] S.N. Lin, Y. Chen, A two-stage physics-informed neural network method based on conserved quantities and applications in localized wave solutions, *J. Comput. Phys.* 457 (2022) 111053.
- [20] Z.J. Zhou, L. Wang, Z.Y. Yan, Deep neural networks learning forward and inverse problems of two-dimensional nonlinear wave equations with rational solitons, *Comput. Math. Appl.* 151 (2023) 164–171.
- [21] X.F. Zhou, C.Z. Qu, W.J. Yan, B. Li, Mastering the Cahn-Hilliard equation and Camassa-Holm equation with cell-average-based neural network method, *Nonlinear Dyn.* 111 (2023) 4823–4846.
- [22] Y. Fang, W.B. Bo, R.R. Wang, Y.Y. Wang, C.Q. Dai, Predicting nonlinear dynamics of optical solitons in optical fiber via the SCPINN, *Chaos Solitons Fractals* 165 (2022) 112908.
- [23] X.Y. Yang, Z. Wang, Solving Benjamin-Ono equation via gradient balanced PINNs approach, *Eur. Phys. J. Plus* 137 (2022) 864.
- [24] Y.F. Mo, L.M. Ling, D.L. Zeng, Data-driven vector soliton solutions of coupled nonlinear Schrödinger equation using a deep learning algorithm, *Phys. Lett. A* 421 (2022) 127739.
- [25] S. Novikov, S.V. Manakov, L.P. Pitaevskii, V.E. Zakharov, The Theory of Solitons: The Inverse Scattering Method, Consultants Bureau Press, New York, 1984.
- [26] C.S. Gardner, J.M. Greene, M.D. Kruskal, R.M. Miura, Method for solving the Korteweg-de Vries equation, *Phys. Rev. Lett.* 19 (1967) 1095–1097.
- [27] P.D. Lax, Integrals of nonlinear equations of evolution and solitary waves, *Commun. Pure Appl. Math.* 21 (1968) 467–490.
- [28] V.E. Zakharov, A.B. Shabat, Exact theory of two-dimensional self-focusing and one dimensional self-modulation of waves in nonlinear media, *Sov. Phys. JETP* 34 (1972) 62–69.
- [29] M.J. Ablowitz, D.J. Kaup, A.C. Newell, H. Segur, The inverse scattering transform-Fourier analysis for nonlinear problems, *Stud. Appl. Math.* 53 (1974) 249–315.
- [30] M.J. Ablowitz, D.J. Kaup, A.C. Newell, H. Segur, Nonlinear-evolution equations of physical significance, *Phys. Rev. Lett.* 31 (1973) 125–127.
- [31] M. Wadati, The modified Korteweg-de Vries equation, *J. Phys. Soc. Jpn.* 34 (1973) 1289–1296.
- [32] M.J. Ablowitz, D.J. Kaup, A.C. Newell, H. Segur, Method for solving the sine-Gordon equation, *Phys. Rev. Lett.* 30 (1973) 1262–1264.
- [33] M.J. Ablowitz, D.B. Yaacov, A.S. Fokas, On the inverse scattering transform for the Kadomtsev-Petviashvili equation, *Stud. Appl. Math.* 69 (1983) 135–143.
- [34] V.B. Matveev, M.A. Salle, Darboux Transformations and Solitons, Springer, Berlin, 1991.
- [35] D.J. Korteweg, H. de Vries, On the change of form of long waves advancing in a rectangular canal, and on a new type of long stationary waves, *Philos. Mag.* 39 (1895) 422–443.
- [36] N.J. Zabusky, M.D. Kruskal, Interactions of solitons in a collisionless plasma and the recurrence of initial states, *Phys. Rev. Lett.* 15 (1965) 240–243.
- [37] R. Camassa, D. Holm, An integrable shallow water equation with peaked solitons, *Phys. Rev. Lett.* 71 (1993) 1661–1664.
- [38] B.B. Kadomtsev, V.I. Petviashvili, On the stability of solitary waves in weakly dispersing media, *Dokl. Akad. Nauk. Russ. Acad. Sci.* 192 (1970) 753–756.
- [39] F. Frenkel, T. Kontorova, On the theory of plastic deformation and twinning, *J. Phys. (USSR)* 1 (1939) 137–149.
- [40] N.J. Zabusky, A synergetic approach to problems of nonlinear dispersive wave propagation and interaction, *Nonlin. Part. Differ. Equ.* (1967) 223–258.
- [41] M. Wadati, K. Konno, Y.H. Ichikawa, New integrable nonlinear evolution equations, *J. Phys. Soc. Jpn.* 47 (1979) 1698.
- [42] T. Schäfer, C.E. Wayne, Propagation of ultra-short optical pulse in nonlinear media, *Physica D* 196 (2004) 90–105.
- [43] M. Stein, Large sample properties of simulations using Latin hypercube sampling, *Technometrics* 29 (1987) 143–151.
- [44] E. Fermi, P. Pasta, S. Ulam, M. Tsingou, Studies of the Nonlinear Problems, Los Alamos National Lab. (LANL), Los Alamos, NM (United States), 1955, No. LA-1940.
- [45] A. Fokas, B. Fuchssteiner, Symplectic structures, their Bäcklund transformation and hereditary symmetries, *Physica D* 4 (1981) 47–66.
- [46] R. Camassa, D. Holm, J. Hyman, A new integrable shallow water equation, *Adv. Appl. Mech.* 31 (1994) 1–33.
- [47] L. Wang, Z.Y. Yan, Data-driven peakon and periodic peakon solutions and parameter discovery of some nonlinear dispersive equations via deep learning, *Physica D* 428 (2021) 133037.
- [48] A.M. Wazwaz, Two new Painlevé-integrable (2+1) and (3+1)-dimensional KdV equations with constant and time-dependent coefficients, *Nucl. Phys. B* 954 (2020) 115009.
- [49] J.C. Pu, Y. Chen, Integrability and exact solutions of the (2+1)-dimensional KdV equation with Bell polynomials approach, *Acta Math. Appl. Sin.-E* 38 (2022) 861–881.

- [50] J. Satsuma, M.J. Ablowitz, Two-dimensional lumps in nonlinear dispersive systems, *J. Math. Phys.* 20 (1979) 1496–1503.
- [51] A.C. Scott, *Nonlinear Science: Emergence and Dynamics of Coherent Structures*, 2nd edition, Oxford and, New York Oxford University Press, 2003.
- [52] A.P. Fordy, J. Gibbons, Factorization of operators I. Miura transformations, *J. Math. Phys.* 21 (1980) 2508–2510.
- [53] N. Akhmediev, A. Ankiewicz, J.M. Soto-Crespo, Rogue waves and rational solutions of the nonlinear Schrödinger equation, *Phys. Rev. E* 80 (2009) 026601.
- [54] C. Kharif, E. Pelinovsky, Physical mechanisms of the rogue wave phenomenon, *Eur. J. Mech. B, Fluids* 22 (2003) 603–634.
- [55] D.R. Solli, C. Ropers, P. Koonath, B. Jalali, Optical rogue waves, *Nature* 450 (2007) 1054–1057.
- [56] A. Chabchoub, N.P. Hoffmann, N. Akhmediev, Rogue wave observation in a water wave tank, *Phys. Rev. Lett.* 106 (2011) 204502.
- [57] Y. Chung, C.K.R.T. Jones, T. Schäfer, C.E. Wayne, Ultra-short pulses in linear and nonlinear media, *Nonlinearity* 18 (2005) 1351–1374.
- [58] M.L. Rabelo, On equations which describe pseudospherical surfaces, *Stud. Appl. Math.* 81 (1989) 221–248.
- [59] A. Sakovich, S. Sakovich, The short pulse equation is integrable, *J. Phys. Soc. Jpn.* 74 (2005) 239–241.
- [60] A. Sakovich, S. Sakovich, Solitary wave solutions of the short pulse equation, *J. Phys. A, Math. Gen.* 39 (2006) L361–L367.
- [61] J. Bradbury, R. Frostig, P. Hawkins, M.J. Johnson, C. Leary, D. Maclaurin, G. Necula, a. Paszke, J. VanderPlas, S. Wanderman-Milne, Q. Zhang, *JAX: composable transformations of Python+NumPy programs*, 2018.
- [62] J. Bettencourt, M.J. Johnson, D. Duvenaud, Taylor-mode automatic differentiation for higherorder derivatives in jax, 2019.
- [63] A. Griewank, A. Walther, *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation*, SIAM, 2008.
- [64] S.F. Wang, S. Sankaran, P. Perdikaris, Respecting causality is all you need for training physics-informed neural networks, *arXiv:2203.07404*, 2022.
- [65] G.Q. Xu, The integrability for a generalized seventh-order KdV equation: Painlevé property, soliton solutions, Lax pairs and conservation laws, *Phys. Scr.* 89 (2014) 125201.