

A Novel Method for Solving Nonlinear Schrödinger Equation with a Potential by Deep Learning

Chaojun Zhang^{1*}, Yuexing Bai²

¹College of Arts and Sciences, Shanghai Maritime University, Shanghai, China

²Ordos Science and Technology Development Center, Ordos, China

Email: *zhangcj@shmtu.edu.cn

How to cite this paper: Zhang, C.J. and Bai, Y.X. (2022) A Novel Method for Solving Nonlinear Schrödinger Equation with a Potential by Deep Learning. *Journal of Applied Mathematics and Physics*, 10, 3175-3190.

<https://doi.org/10.4236/jamp.2022.1010211>

Received: September 20, 2022

Accepted: October 24, 2022

Published: October 27, 2022

Copyright © 2022 by author(s) and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution-NonCommercial International License (CC BY-NC 4.0).

<http://creativecommons.org/licenses/by-nc/4.0/>



Open Access

Abstract

The improved physical information neural network algorithm has been proven to be used to study integrable systems. In this paper, the improved physical information neural network algorithm is used to study the defocusing nonlinear Schrödinger (NLS) equation with time-varying potential, and the rogue wave solution of the equation is obtained. At the same time, the influence of the number of network layers, neurons and the number of sampling points on the network performance is studied. Experiments show that the number of hidden layers and the number of neurons in each hidden layer affect the relative \mathbb{L}_2 -norm error. With fixed configuration points, the relative norm error does not decrease with the increase in the number of boundary data points, which indicates that in this case, the number of boundary data points has no obvious influence on the error. Through the experiment, the rogue wave solution of the defocusing NLS equation is successfully captured by IPINN method for the first time. The experimental results of this paper are also compared with the results obtained by the physical information neural network method and show that the improved algorithm has higher accuracy. The results of this paper will be contributed to the generalization of deep learning algorithms for solving defocusing NLS equations with time-varying potential.

Keywords

Physics-Informed Neural Networks, Improved Physics-Informed Neural Networks, Defocusing NLS Equation, Rogue Wave Solution

1. Introduction

It is well known that differential systems are nonlinear phenomena that describe

mathematics, physics, biology, chemistry, transportation, finance and other fields. Solving these differential systems can make people understand these phenomena as much as possible, especially in the study of differential systems in finance, such as financial soliton solutions and financial rogue wave solutions. The peaks and troughs of these waves correspond to the inflection points of stock prices in the financial market. Obviously, it provides a theoretical mechanism for people to understand the phenomenon of the financial crisis. In recent years, many classical analytical and numerical methods have been developed in the field of computing. Although these methods give numerical solutions or analytical solutions of partial differential systems to a certain extent, such as Lie symmetry in analytical methods, this method first calculates the characteristic sequence set, then calculates the infinitesimal generator, and finally transforms the partial differential system into an ordinary differential system solution. The amount of calculation involved in this process is very large. Although there is the help of a computer only reduces the computational burden to a certain extent, and some problems must be calculated manually. Especially when the partial differential problem is transformed into an ordinary differential problem, although the order of the differential equation is reduced, some ordinary differential problems cannot be solved, and the final problem is still unsolved. Another example is the finite difference method in the classical numerical method, which divides the solution domain into differential grids, but with the increase of the complexity of the boundary, the grid intersections cannot all be guaranteed to fall on the boundary conditions; this method lacks flexibility on complex boundaries. Therefore, the solution of differential systems has always been a research hotspot in the scientific community.

In recent years, with the development and application of artificial intelligence technology in various neighborhoods, experts have turned their attention to artificial intelligence, and they have considered using neural networks to study differential systems. For example, I.E. Lagaris and A.C. Likas studied the solutions of ordinary and partial differential equations using artificial neural networks in 1998 [1]. In 2000, S. He and K. Reif investigated the solution of partial differential equations by multilayer neural networks [2]. In the same year, I.E. Lagaris *et al.* studied differential equations with irregular boundaries by neural networks [3]. In 2006, A. Malek and R.S. Beidokhti obtained numerical solutions for high-order differential equations by a hybrid neural network method [4]. In 2008, Y. Shirvany *et al.* studied the numerical solution of the nonlinear Schrödinger equation by feedforward neural networks [5]. In 2011, H. Chen *et al.* got the numerical solution of PDEs by integrated radial basis function networks [6]. In 2015, N. Yadav *et al.* got the solution to nonlinear elliptic boundary value problems by the neural network [7]. Lin Z. studied the multiphase flow problem by Physics-Aware deep learning [8]. D.A. Maturi and H.M. Malaikah solving nonlinear partial differential equation by Adomian decomposition method [9]. In 2018, Y. Yang *et al.* studied the solution of ordinary differential equations by

Legendre neural network methods [10]. The breakthrough in using a neural network to solve differential equations is the physical information neural network (PINN) algorithm proposed by professor Raissi from Brown University in 2019 [11]. The physical information neural network algorithm does not need to assume the expression of the solution. It embeds the physical information and the initial-boundary value conditions in the neural network. Only a small number of random sample points on the initial-boundary value conditions can obtain the numerical solution of the differential equation. In view of these advantages, many scholars have studied them. Such as Z. Yan *et al.* studied forward and inverse problems of the Schrödinger equation with \mathcal{PT} -symmetric harmonic potential [12], the team of Y. Chen used the PINN method to study integrable systems [13] [14] [15] [16]. With the research on PINN, it is found that it does not converge on some complex problems, so many improvement methods are proposed. Such as Bayesian physics-informed neural networks (B-PINNs) [17], fractional Physics-Informed Neural Networks (fPINNs) [18], Parareal physics-informed neural network (PPINN) [19], Conservative physics-informed neural networks (CPINN) [20], nonlocal Physics-Informed Neural Networks (nPINNs) [21] and so on. Jagtap A. D. *et al.* added an adaptive activation function [22] to PINN and studied the inverse problem of differential equations. Subsequently, based on the work of Jagtap A. D., the team of Chen Yong used the PINN algorithm with an adaptive activation function to study the problem of solving differential equations and named it IPINN [23]. We also studied the rogue wave solution [24] and the soliton solution [25] using by IPINN method. Although the PINN algorithm and its improvement have made some achievements, the research of these new algorithms on differential systems is much more than that.

Recently, some scholars discovered an interesting thing in their research. It is found that the nonlinear wave phenomenon that occurs in the focused NLS equation also appears stably in the generalized defocusing NLS equation with an external potential [26], and they used PINN method to study the rogue wave solutions of the defocusing NLS equation with spatio-temporal potential. In this paper, we will use the IPINN algorithm to study defocusing NLS equation. It is known that the IPINN algorithm converges faster and has higher accuracy by comparing the results obtained by the IPINN algorithm and the PINN algorithm. The numerical experiments in this paper are performed on a computer with an 11th generation Intel(R) Core(TM) i7-11800H @ 2.30 GHz processor and 16.0 GB memory.

2. Rogue Wave Solution for Defocusing NLS Equation

The defocusing NLS equation with spatio-temporal potential [27] is written as

$$\begin{cases} iq_t + 0.5q_{xx} - V(t, x)q - |q|^2 q = 0, x \in [-L, L], t \in [-T, T], \\ q(-T, x) = q(T, x), x \in [-L, L], \\ q(t, -L) = q(t, L), t \in [-T, T]. \end{cases} \quad (1)$$

where $V(t, x)$ denotes the spatio-temporal potential and it can be written as $V(t, x) = \frac{4(x^2 - t^2) - 1}{(x^2 + t^2 + 0.25)^2} - 2$, q is the complex solution with the independent variables x, t of Equation (1). Li Wang and Zhenya Yan [27] proved for the first time that there is also an analytical rogue wave solution for the defocusing NLS equation. We assume $q = u(t, x) + v(t, x)i$, the real part and imaginary part of $q(t, x)$ are $u(t, x)$ and $v(t, x)$. Equation (1) can be inverted into

$$\begin{cases} v_t - 0.5u_{xx} + \left[\frac{4(x^2 - t^2) - 1}{(x^2 + t^2 + 0.25)^2} - 2 \right] u + u(u^2 + v^2) = 0, x \in [-L, L], t \in [t_0, t_1], \\ u_t + 0.5v_{xx} - \left[\frac{4(x^2 - t^2) - 1}{(x^2 + t^2 + 0.25)^2} - 2 \right] v - v(u^2 + v^2) = 0. \end{cases} \tag{2}$$

In this paper, we focus on solving the rogue wave solution of Equation (1) by applying an improved PINN algorithm. Suppose the network has 8 hidden layers and each hidden layer has 40 neurons. The real parts $f_R(t, x)$ and imaginary parts $f_I(t, x)$ of the network residual are defined as

$$\begin{cases} f_R(t, x) := \tilde{v}_t - 0.5\tilde{u}_{xx} + \left[\frac{4(x^2 - t^2) - 1}{(x^2 + t^2 + 0.25)^2} - 2 \right] \tilde{u} + \tilde{u}(\tilde{u}^2 + \tilde{v}^2), \\ f_I(t, x) := \tilde{u}_t + 0.5\tilde{v}_{xx} - \left[\frac{4(x^2 - t^2) - 1}{(x^2 + t^2 + 0.25)^2} - 2 \right] \tilde{v} - \tilde{v}(\tilde{u}^2 + \tilde{v}^2), \end{cases} \tag{3}$$

Loss function is defined as $Loss = Loss_f + Loss_0 + Loss_b + Loss_a$, where

$$\begin{aligned} Loss_f &= \frac{1}{N_f} \sum_{i=1}^{N_f} \left[\|f_R(t^i, x^i)\|^2 + \|f_I(t^i, x^i)\|^2 \right], \\ Loss_0 &= \frac{1}{N_0} \sum_{i=1}^{N_0} \left[\|\tilde{u}(t_0^i, x_0^i) - u(t_0^i, x_0^i)\|^2 + \|\tilde{v}(t_0^i, x_0^i) - v(t_0^i, x_0^i)\|^2 \right], \\ Loss_b &= \frac{1}{N_b} \sum_{i=1}^{N_b} \left[\|\tilde{u}(t_b^i, x_b^i) - u(t_b^i, x_b^i)\|^2 + \|\tilde{v}(t_b^i, x_b^i) - v(t_b^i, x_b^i)\|^2 \right], \\ Loss_a &= \frac{1}{\frac{1}{L-1} \sum_{l=1}^{L-1} \exp\left(\frac{\sum_{i=1}^{n_l} a_i^l}{n_l}\right)}. \end{aligned} \tag{4}$$

where $\{t_0^i, x_0^i\}_{i=1}^{N_0}$ and $\{t_b^i, x_b^i\}_{i=1}^{N_b}$ denote randomly sampling N_0 and N_b points on the initial-boundary value conditions. The exact solutions $u(t_0^i, x_0^i), v(t_0^i, x_0^i), u(t_b^i, x_b^i), v(t_b^i, x_b^i)$ are given by initial-boundary value conditions. $\tilde{u}(t_0^i, x_0^i), \tilde{v}(t_0^i, x_0^i), \tilde{u}(t_b^i, x_b^i), \tilde{v}(t_b^i, x_b^i)$ are the predicted values by improved PINN of $u(t_0^i, x_0^i), v(t_0^i, x_0^i), u(t_b^i, x_b^i), v(t_b^i, x_b^i)$ respectively. We obtain configuration points $\{t^i, x^i\}_{i=1}^{N_f}$ of real part $f_R(t, x)$ and imaginary part $f_I(t, x)$ through the Latin hypercube sampling strategy [28]. The optimal weights \mathbf{W} , bias \mathbf{b} and activation function slopes \mathbf{a} are obtained by automatic differentiation

technology, Adam [29] and L-BFGS [30] algorithm to minimize the loss function $Loss$, and the numerical solution of Equation (1) is finally determined. We also give the deep neural network in Figure 1 and schematic of IPINN for the defocusing NLS equation in Figure 2.

The rogue wave solution [27] of Equation (1) is

$$q(t, x) = \left[1 - \frac{4(1+2it)}{4(x^2+t^2)+1} \right] \exp(it),$$

we use the IPINN algorithm to study the Cauchy problem of Equation (1), and assuming that the initial conditions are

$$q(t, x) = \left[1 - \frac{4(1-3i)}{4(x^2+2.25)+1} \right] \exp(-1.5i), \quad x \in [-\pi, \pi].$$

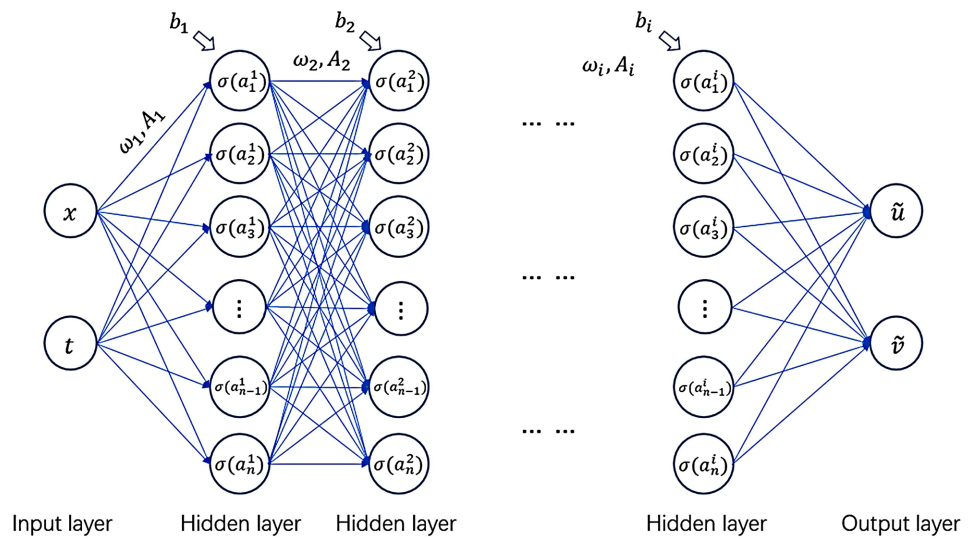


Figure 1. Schematic of deep neural network.

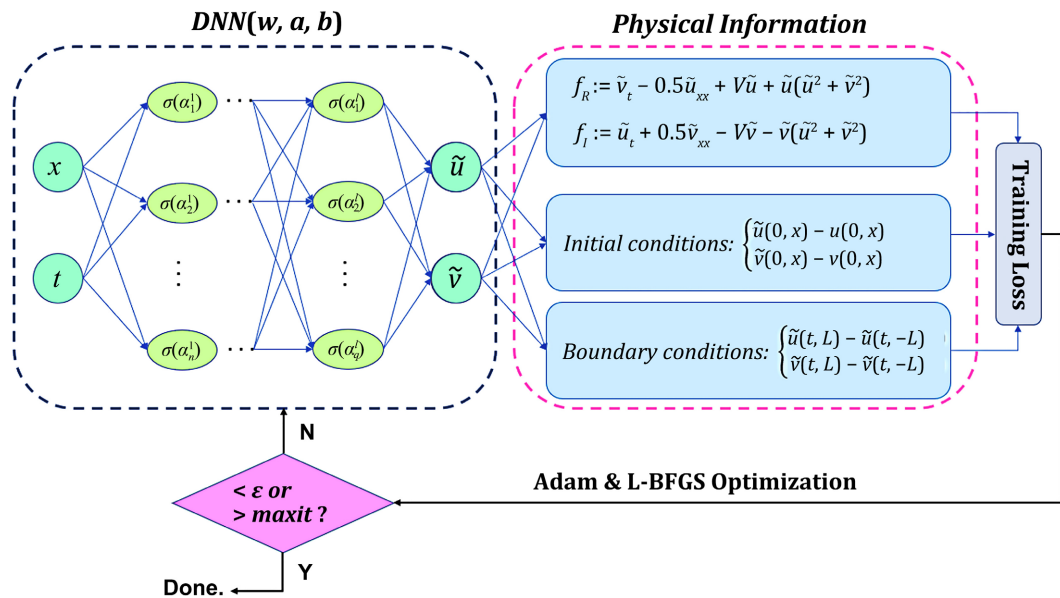
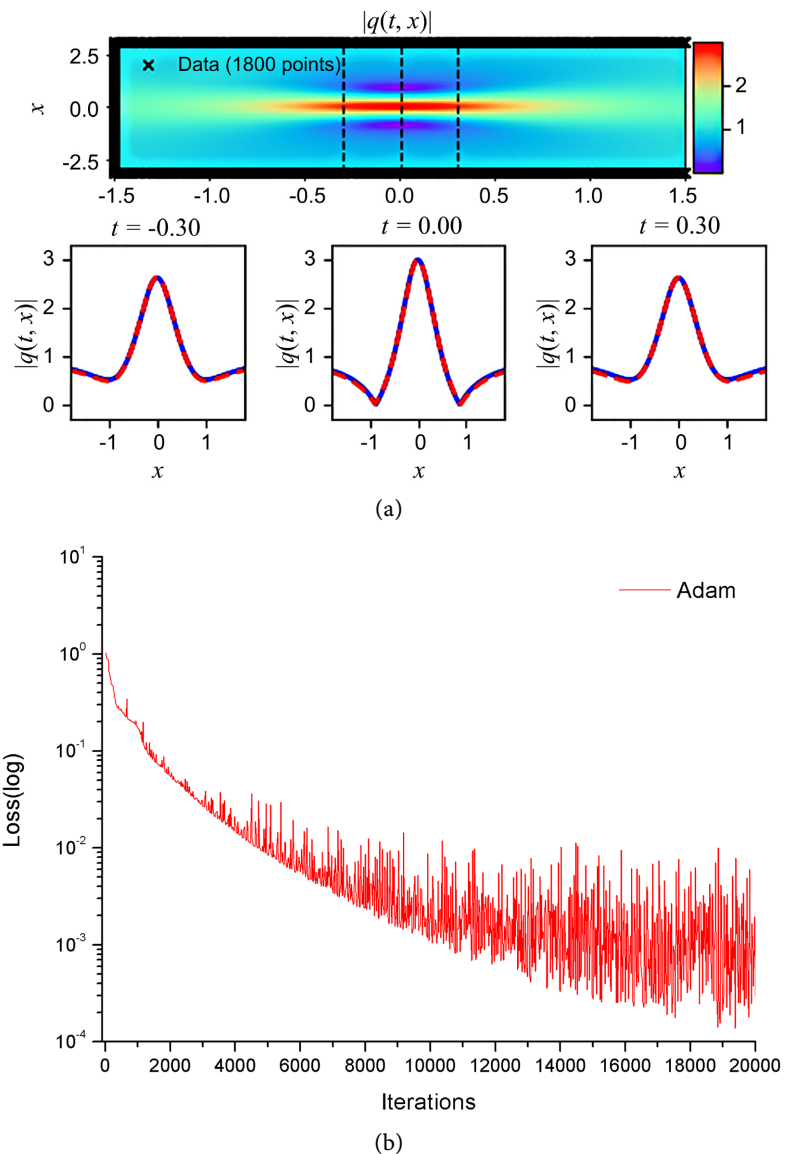


Figure 2. Schematic of Improved PINN for the defocusing NLS equation.

into 3000 points and $t \in [-1.5, 1.5]$ into 2000 points by MATLAB. The improved physical information neural network is composed of 8 hidden layers, and each hidden layer has 40 neurons. We sample $N_0 = N_b = 600$ points from the initial-boundary value conditions by random sub-sampling and $N_f = 10000$ points in the feasible region by Latin hypercube sampling strategy as our experimental training point. Without loss of generality, we initialize the scalable parameter as $n = 10, a_i^l = 0.1$. Hyperbolic tangent is chosen as an activation function. Finally, the training time is 8174.6736 seconds, the network residual is $1.6892622e-05$, relative \mathbb{L}_2 norm errors of $u(t, x)$, $v(t, x)$, and $h(t, x)$ are $3.55117e-02$, $5.829360e-02$, $2.273524e-02$ respectively. We show the fitting diagram of accuracy (blue solid line) and prediction (red dotted line) at different times $t = -0.30, 0.00, 0.30$, the curve diagram of network residual changing with the number of iterations under the condition of the Adam optimization algorithm, and the density diagram of accurate rogue wave and predicted rogue wave in **Figure 3**.



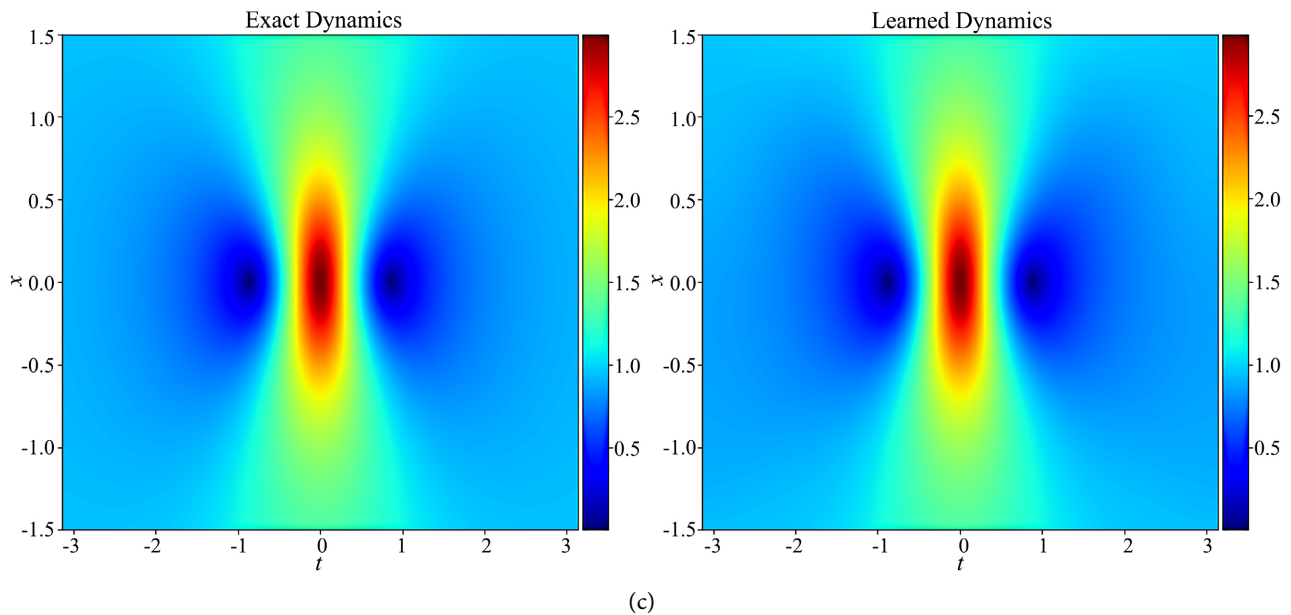


Figure 3. We choose a neural network with 8 hidden layers, each with 40 neurons, and choose 1800 points on the initial boundary value condition. (a) shows the density diagrams and predicted numerical solution at $t = -0.3, 0.0, 0.3$; (b) shows the variation of the loss function of the network with the number of iterations when Adam optimization algorithm is adopted; (c) represents the exact density diagram and predicted density diagram of rogue waves of Equation (1).

In order to study the influence of the number of hidden layers and the neurons in each layer on the relative \mathbb{L}_2 norm errors of $q(t, x)$, we select the training point $N_0 = N_b = 100$, $N_f = 10000$, the number of hidden layers are 2, 4, 6 and 8 respectively, and the neurons in each hidden layer are 10, 15, 20, 25 and 30 respectively, as is shown in **Table 1**. According to the data from **Table 1**, relative \mathbb{L}_2 norm errors of $q(t, x)$ of neurons with fixed hidden layers do not decrease monotonically with the increase in the number of hidden layers. When the hidden layer is fixed, the norm error does not decrease monotonically with the increase of neurons. Therefore, we know that the norm error is affected by the number of hidden layers and neurons. At the same time, we study the influence of sampling points $N_q = N_0 + N_b$ and N_f on the relative \mathbb{L}_2 norm errors of $q(t, x)$. We fix the hidden layer of the improved physical information neural network as 8 and each layer has 10 neurons, then the corresponding relative \mathbb{L}_2 norm errors of $q(t, x)$ value is given when we assume that sampling points $N_q = 100, 200, 300$ and $N_f = 2000, 4000, 6000$, as is shown in **Table 2**. When N_q is fixed, the relative \mathbb{L}_2 norm error of $q(t, x)$ decreases with the increase of N_f . However, when $N_f = 2000$, the relative \mathbb{L}_2 norm error of $q(t, x)$ does not decrease with the increase of N_q . When $N_f = 4000, 6000$, the relative \mathbb{L}_2 norm error of $q(t, x)$ decreases with the increase of N_q , which also shows that the boundary sampling points and regional configuration points jointly affect the relative \mathbb{L}_2 norm error of $q(t, x)$.

In order to understand the influence of regional configuration points N_f on the predicted rogue wave solution, we fixed the improved physical information neural network, which has 8 hidden layers and each layer contains 10 neurons.

Table 1. The rogue wave solution of Equation (1): The relative \mathbb{L}_2 -norm error of $q(t, x)$ when taking the $N_0 = 100, N_b = 100, N_f = 10000$ with different network layers and different numbers of neurons per hidden layers.

\mathbb{L}_2 -norm error	Neurons					
		10	15	20	25	30
Layers						
	2	5.5093e-01	1.9859e-02	2.0163e-02	2.1916e-02	2.2094e-02
	4	2.0949e-02	2.1834e-02	2.1865e-02	2.2016e-02	2.2101e-02
	6	2.1460e-02	2.1518e-02	2.2099e-02	2.2101e-02	2.2102e-02
	8	2.1219e-02	2.1730e-02	2.2034e-02	2.2067e-02	2.2312e-02

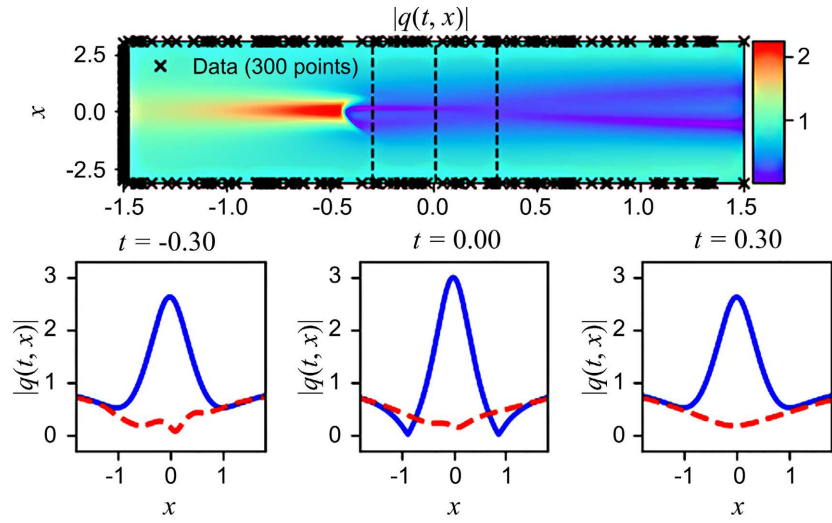
Table 2. The rogue wave solution of the Equation (1): We chose the neural network with 8 hidden layers, one input layer and one output layer, and each hidden layer has 10 neurons. The relative \mathbb{L}_2 -norm error of q was studied under different $N_q = N_0 + N_b$, $N_0 = N_b$ and N_f conditions.

\mathbb{L}_2 -norm error	N_q			
		100	200	300
N_f				
	2000	5.487201e-01	4.770664e-01	5.487201e-01
	4000	2.156434e-02	2.176194e-02	2.156434e-02
	6000	2.119324e-02	2.013049e-02	1.019324e-02

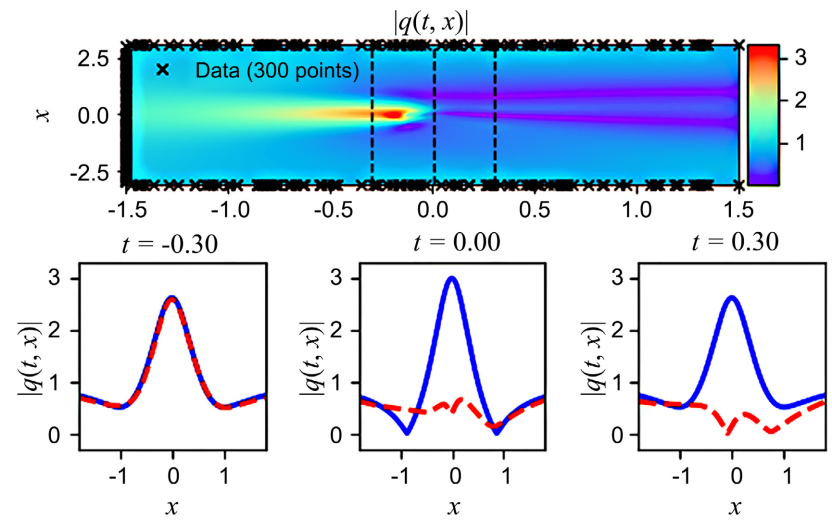
We set the initial-boundary value sampling point 300. From **Figure 4**, we can see that with the increase in regional configuration points

$N_f = 1000, 2000, 3000, 4000$, the fitting effect of the exact rogue wave solution and predicted rogue wave solution is getting better and better. In order to study the influence of sampling point N_f on the relative norm error of $u(t, x), v(t, x), q(t, x)$ of the improved physical information neural network, when the fixed network has 8 hidden layers and their neurons (10, 15, 20, 25, 30), the corresponding norm error is shown in **Figures 5(a)-(e)**. From the figure, we can see that the relative norm error ($u(t, x), v(t, x), q(t, x)$) roughly shows a decreasing trend with the increase of N_f .

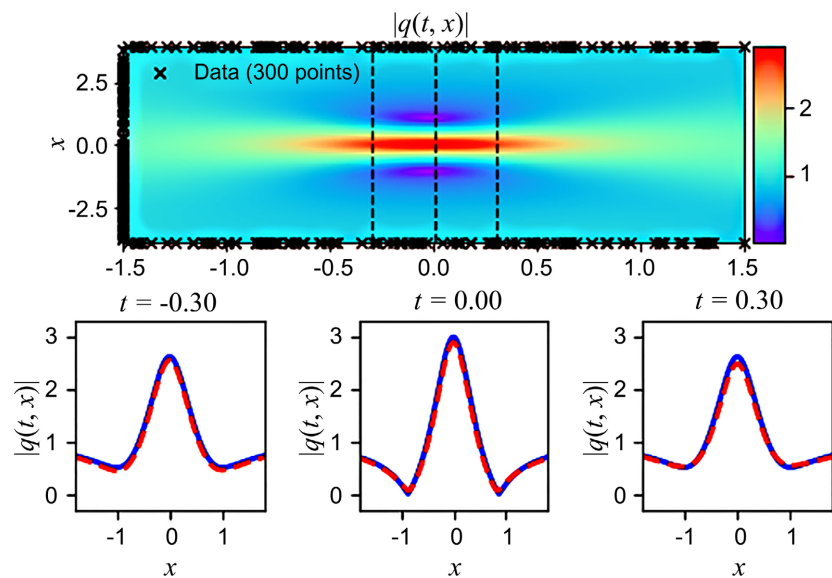
Finally, we compare the performance of PINN and IPINN in solving rogue wave solutions. The two algorithms have the same hidden layer and its neurons, and both have the same random sampling points 300 from initial-boundary value condition. The network residuals is 0.009715796 and relative norm error of $u(t, x), v(t, x), q(t, x)$ are 9.229974e-01, 1.630527e+00, 3.833559e-01 respectively of PINN. The network residuals is 8.936103e-05 and the relative norm error of $u(t, x), v(t, x), q(t, x)$ are 3.415088e-02, 6.299041e-02, 2.121869e-02 respectively of IPINN. It is known from the experimental data that the IPINN has a better fitting effect on the exact solution and prediction solution, as shown in **Figure 6(a) & Figure 6(b)**. At the same time, the curve of the network residual corresponding to the Adam algorithm and the L-BFGS algorithm of the two



(a)



(b)



(c)

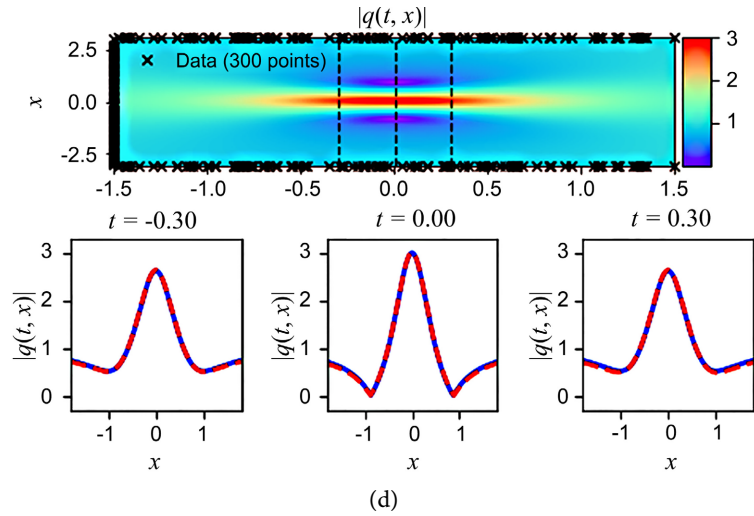
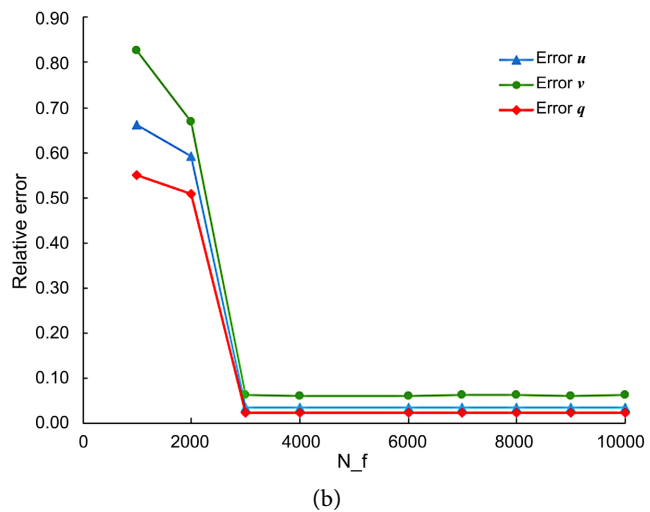
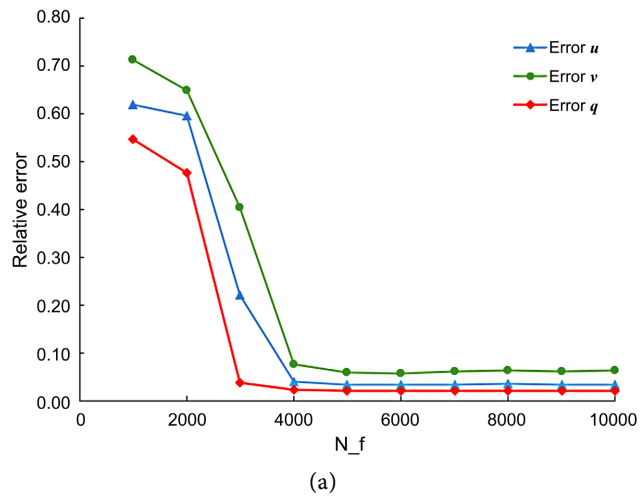
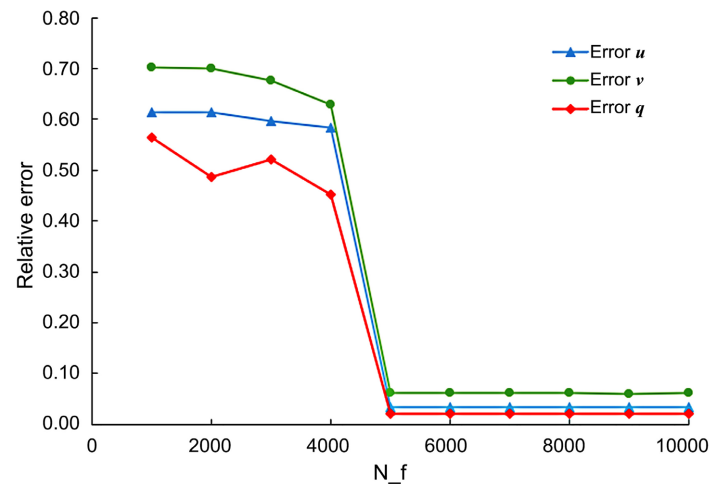
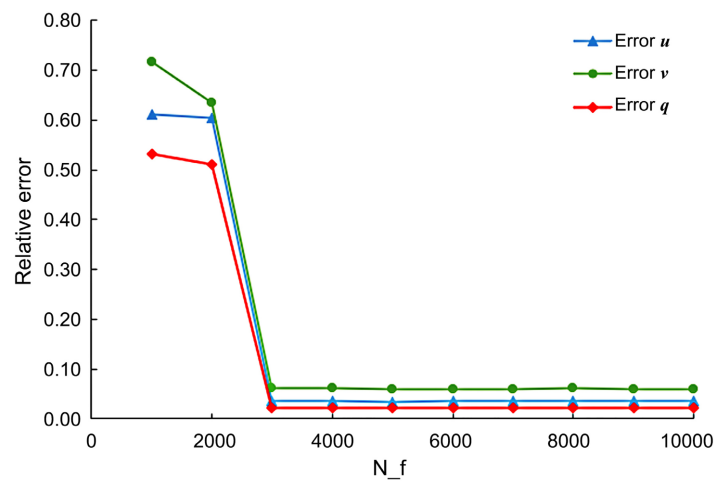


Figure 4. A hidden layer with 8 hidden layers, each of which has 10 neurons, was selected and 300 sample points were randomly selected under initial boundary value condition to study the influence of N_f on the numerical rogue wave solution of Equation (1). The experimental results under the conditions of $N_f = 1000, 2000, 3000, 4000$ are shown in Figure 4(a)-(d) respectively.

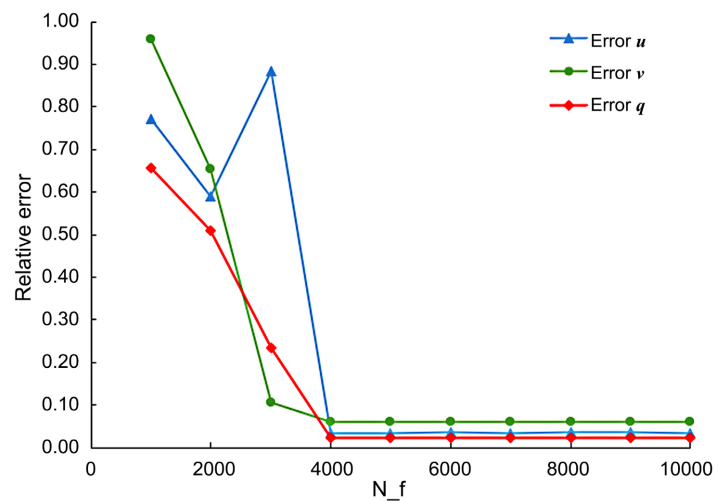




(c)

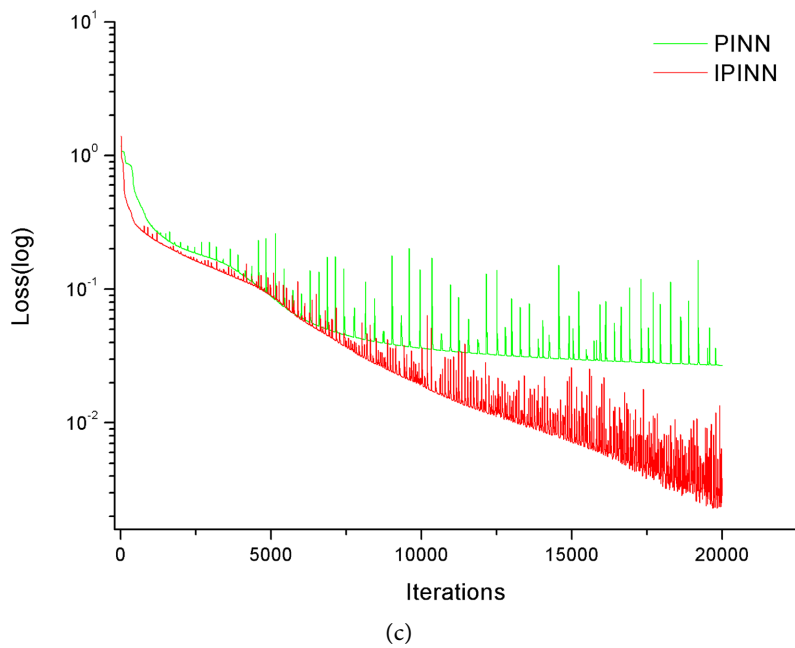
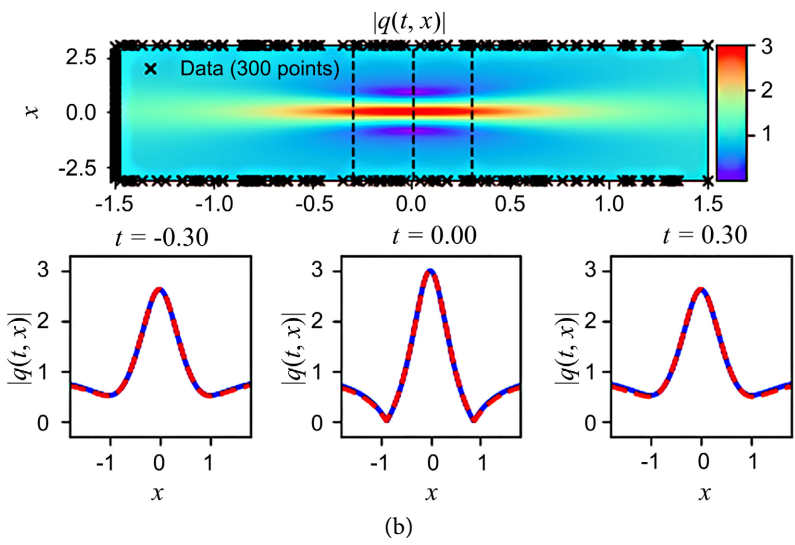
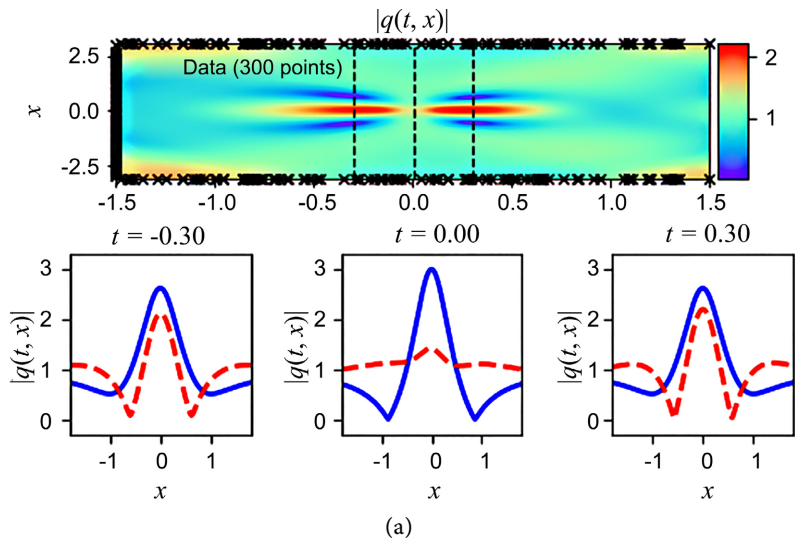


(d)



(e)

Figure 5. (a)-(e) respectively represent 8 fixed hidden layers, and study the influence of neurons contained in hidden layers on the relative errors of q , u and v of Equation (1) with the change of N_f . (a)-(e) respectively show the line diagram of the relative error of q , u and v as N_f changes when neurons are 10, 15, 20, 25 and 30.



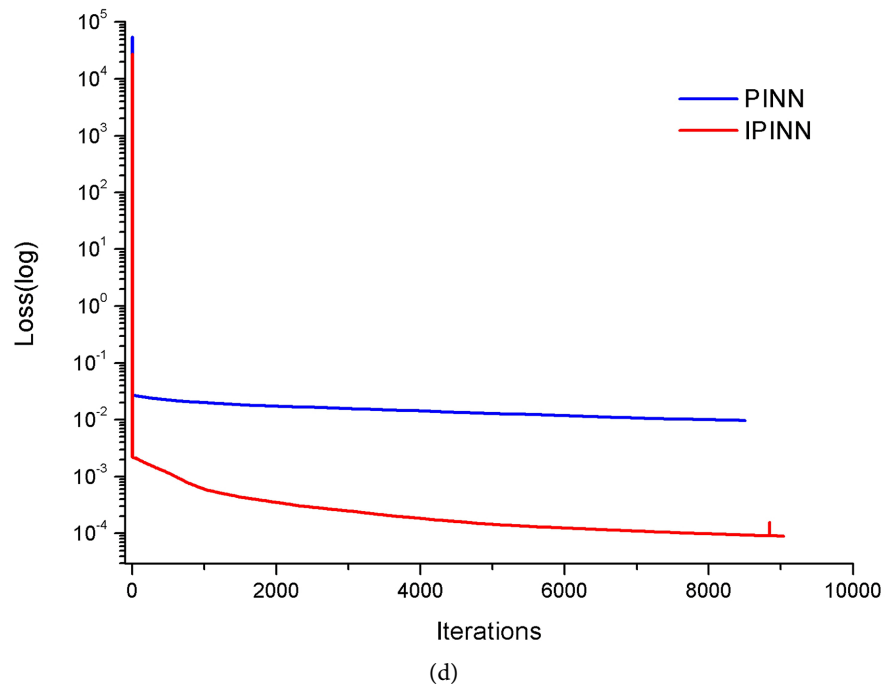


Figure 6. Equation (1): We study the influence of PINN algorithm and IPINN algorithm on the accuracy of the rogue wave solution of the equation respectively. A neural network with 8 hidden layers and 10 neurons in each hidden layer is selected, and the same tanh activation function is sampled, to sample the same activation function $\tanh = \frac{e^x - e^{-x}}{e^x + e^{-x}}$. (a) and (b) are the numerical solutions of the rogue waves obtained by the PINN and IPINN algorithms, respectively; (c) shows the variation of network residual with the number of iterations when Adam optimization algorithm is adopted; (d) shows the variation of network residual with the number of iterations when the Adam optimization algorithm is adopted and then L-BFGS optimization algorithm is adopted.

methods with the number of iterations is given, as shown in **Figure 6(c)** & **Figure 6(d)**. It is known from the curve that whether it is the Adam optimization algorithm or the L-BFGS optimization algorithm, the IPINN algorithm has faster convergence and higher accuracy.

3. Conclusion

In this paper, we mainly use the IPINN algorithm to solve the rogue wave solution of the defocusing NLS equation and study the influence of the network's hidden layer and its neurons on the relative norm error value of $u(t, x), v(t, x), q(t, x)$. At the same time, in order to illustrate that IPINN has better network performance than PINN, such as faster convergence and higher precision. The experimental results show that IPINN has better potential in solving high-order and nonlinear differential equations. Whether to apply it to more complex high-order differential equations or adding conservation laws or more integrable system properties to the IPINN algorithm to expand IPINN algorithm are our research content in the future.

Acknowledgements

This work is supported by University Experimental Technology Team Construction in Shanghai.

Availability of Data and Code

The datasets generated or analysed during the current study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare that they have no conflict of interest.

References

- [1] Lagaris, I.E. and Likas, A.C. (1998) Artificial Neural Networks for Solving Ordinary and Partial Differential Equations. *IEEE Transactions on Neural Networks*, **9**, 987-1000. <https://doi.org/10.1109/72.712178>
- [2] He, S., Reif, K. and Unbehauen, R. (2000) Multilayer Networks for Solving a Class of Partial Differential Equations. *Neural Networks*, **13**, 385-396. [https://doi.org/10.1016/S0893-6080\(00\)00013-7](https://doi.org/10.1016/S0893-6080(00)00013-7)
- [3] Lagaris, I.E., Likas, A.C. and Papageorgiou, D.G. (2000) Neural-Network Methods for Boundary Value Problems with Irregular Boundaries. *IEEE Transactions on Neural Networks*, **11**, 1041-1049. <https://doi.org/10.1109/72.870037>
- [4] Malek, A. and Beidokhti, R.S. (2006) Numerical Solution for High Order Differential Equations Using a Hybrid Neural Network-Optimization Method. *Applied Mathematics and Computation*, **183**, 260-271. <https://doi.org/10.1016/j.amc.2006.05.068>
- [5] Shirvany, Y., Hayati, M. and Moradian, R. (2008) Numerical Solution of the Nonlinear Schrödinger Equation by Feed forward Neural Networks. *Communications in Nonlinear Science and Numerical Simulation*, **13**, 2132-2145. <https://doi.org/10.1016/j.cnsns.2007.04.024>
- [6] Chen, H., Kong, L. and Leng, W. (2011) Numerical Solution of PDEs via Integrated Radial Basis Function Networks with Adaptive Training Algorithm. *Applied Soft Computing*, **11**, 855-860. <https://doi.org/10.1016/j.asoc.2010.01.005>
- [7] Yadav, N., Yadav, A. and Deep, K. (2015) Artificial Neural Network Technique for Solution of Nonlinear Elliptic Boundary Value Problems. *Advances in Intelligent Systems and Computing*, **335**, 113-121. https://doi.org/10.1007/978-81-322-2217-0_10
- [8] Lin, Z. (2021) Physics-Aware Deep Learning on Multiphase Flow Problems. *Communications and Network*, **13**, 1-11. <https://doi.org/10.4236/cn.2021.131001>
- [9] Maturi, D.A. and Malaikah, H.M. (2021) The Adomian Decomposition Method for Solving Nonlinear Partial Differential Equation Using Maple. *Advances in Pure Mathematics*, **11**, 595-603. <https://doi.org/10.4236/apm.2021.116038>
- [10] Yang, Y., Hou, M. and Luo, J. (2018) A Novel Improved Extreme Learning Machine Algorithm in Solving Ordinary Differential Equations by Legendre Neural Network Methods. *Advances in Difference Equations*, **2018**, 469-493. <https://doi.org/10.1186/s13662-018-1927-x>
- [11] Raissi, M., Perdikaris, P. and Karniadakis, G.E. (2019) Physics-Informed Neural Networks: A Deep Learning Framework for Solving forward and Inverse Problems

- Involving Nonlinear Partial Differential Equations. *Journal of Computational Physics*, **378**, 686-707. <https://doi.org/10.1016/j.jcp.2018.10.045>
- [12] Zhou, Z. and Yan, Z. (2020) Solving Forward and Inverse Problems of the Logarithmic Nonlinear Schrodinger Equation with PT-Symmetric Harmonic Potential via Deep Learning. *Physics Letters A*, **387**, Article ID: 127010. <https://doi.org/10.1016/j.physleta.2020.127010>
- [13] Li, J. and Chen, Y. (2020) A Deep Learning Method for Solving Third-Order Nonlinear Evolution Equations. *Communications in Theoretical Physics*, **72**, Article ID: 115003. <https://doi.org/10.1088/1572-9494/abb7c8>
- [14] Miao, Z.W. and Chen, Y. (2022) Physics-Informed Neural Networks Method in High-Dimensional Integrable Systems. *Modern Physics Letters B*, **36**, Article ID: 2150531. <https://doi.org/10.1142/S021798492150531X>
- [15] Peng, W.Q., Pu, J.C. and Chen, Y. (2022) PINN Deep Learning Method for the Chen-Lee-Liu Equation: Rogue Wave on the Periodic Background. *Communications in Nonlinear Science and Numerical Simulations*, **105**, Article ID: 106067. <https://doi.org/10.1016/j.cnsns.2021.106067>
- [16] Pu, J., Li, J. and Chen, Y. (2021) Soliton, Breather, and Rogue Wave Solutions for Solving the Nonlinear Schrödinger Equation Using a Deep Learning Method with Physical Constraints. *Chinese Physics B*, **30**, Article ID: 060202. <https://doi.org/10.1088/1674-1056/abd7e3>
- [17] Yang, L., Meng, X. and Karniadakis, G.E. (2021) B-PINNs: Bayesian Physics-Informed Neural Networks for Forward and Inverse PDE Problems with Noisy Data. *Journal of Computational Physics*, **425**, Article ID: 109913. <https://doi.org/10.1016/j.jcp.2020.109913>
- [18] Pang, G., Lu, L. and Karniadakis, G.E. (2019) fPINNs: Fractional Physics-Informed Neural Networks. *SIAM Journal on Scientific Computing*, **41**, A2603-A2626. <https://doi.org/10.1137/18M1229845>
- [19] Meng, X., et al. (2020) PPINN: Parareal Physics-Informed Neural Network for Time-Dependent PDEs. *Computer Methods in Applied Mechanics and Engineering*, **370**, Article ID: 113250. <https://doi.org/10.1016/j.cma.2020.113250>
- [20] Jagtap, A.D., Kharazmi, E. and Karniadakis, G.E. (2020) Conservative Physics-Informed Neural Networks on Discrete Domains for Conservation Laws: Applications to Forward and Inverse Problems. *Computer Methods in Applied Mechanics and Engineering*, **365**, Article ID: 113028. <https://doi.org/10.1016/j.cma.2020.113028>
- [21] Pang, G., et al. (2020) nPINNs: Nonlocal Physics-Informed Neural Networks for a Parametrized Nonlocal Universal Laplacian Operator. Algorithms and Applications. *Journal of Computational Physics*, **422**, Article ID: 109760. <https://doi.org/10.2172/1614899>
- [22] Jagtap, A.D., Kawaguchi, K. and Karniadakis, G.E. (2019) Locally Adaptive Activation Functions with Slope Recovery Term for Deep and Physics-Informed Neural Networks. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, **476**, 1-20. <https://doi.org/10.1016/j.jcp.2019.109136>
- [23] Pu, J., Peng, W. and Chen, Y. (2021) The Data-Driven Localized Wave Solutions of the Derivative Nonlinear Schrodinger Equation by Using Improved PINN Approach. *Nonlinear Dynamics*, **105**, 1723-1739. <https://doi.org/10.1007/s11071-021-06554-5>
- [24] Bai, Y., Chaolu, T. and Bilige, S. (2022) The Application of Improved Physics-Informed Neural Network (IPINN) Method in Finance. *Nonlinear Dynamics*, **107**, 3655-3667. <https://doi.org/10.1007/s11071-021-07146-z>

- [25] Bai, Y., Chaolu, T. and Bilige, S. (2021) Solving Huxley Equation Using an Improved PINN Method. *Nonlinear Dynamics*, **105**, 3439-3450.
<https://doi.org/10.1007/s11071-021-06819-z>
- [26] Wang, L. and Yan, Z.Y. (2021) Data-Driven Rogue Waves and Parameter Discovery in the Defocusing Nonlinear Schrodinger Equation with a Potential Using the PINN Deep Learning. *Physics Letters A*, **404**, Article ID: 127408.
<https://doi.org/10.1016/j.physleta.2021.127408>
- [27] Wang, L. and Yan, Z.Y. (2021) Rogue Wave Formation and Interactions in the Defocusing Nonlinear Schrödinger Equation with External Potentials. *Applied Mathematics Letters*, **111**, Article ID: 106670.
<https://doi.org/10.1016/j.aml.2020.106670>
- [28] Stein, M. (1987) Large Sample Properties of Simulations Using Latin Hypercube Sampling. *Technometrics*, **29**, 143-151.
<https://doi.org/10.1080/00401706.1987.10488205>
- [29] Kingma, D. and Ba, J. (2014) Adam: A Method for Stochastic Optimization. *3rd International Conference on Learning Representations, ICLR 2015*, San Diego, 7-9 May 2015, 1-15.
- [30] Liu, D.C. and Nocedal, J. (1989) On the Limited Memory BFGS Method for Large Scale Optimization. *Mathematical Programming*, **45**, 503-528.
<https://doi.org/10.1007/BF01589116>