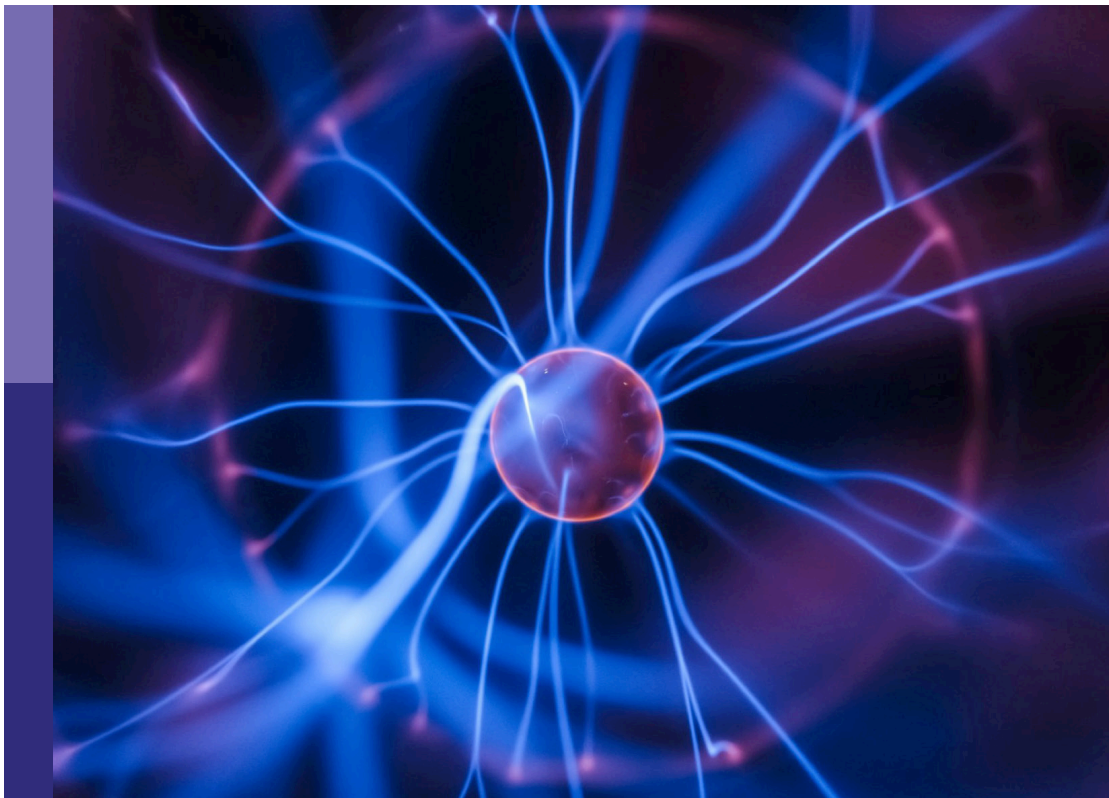
frontiers | Research Topics

Advances in nonlinear systems and networks

Edited by
Fei Yu, Hairong Lin and Viet-Thanh Pham

Published in
Frontiers in Physics



ISSN 1664-8714
ISBN 978-2-83252-046-8
DOI 10.3389/978-2-83252-046-8

About Frontiers

Frontiers is more than just an open access publisher of scholarly articles: it is a pioneering approach to the world of academia, radically improving the way scholarly research is managed. The grand vision of Frontiers is a world where all people have an equal opportunity to seek, share and generate knowledge. Frontiers provides immediate and permanent online open access to all its publications, but this alone is not enough to realize our grand goals.

Frontiers journal series

The Frontiers journal series is a multi-tier and interdisciplinary set of open-access, online journals, promising a paradigm shift from the current review, selection and dissemination processes in academic publishing. All Frontiers journals are driven by researchers for researchers; therefore, they constitute a service to the scholarly community. At the same time, the *Frontiers journal series* operates on a revolutionary invention, the tiered publishing system, initially addressing specific communities of scholars, and gradually climbing up to broader public understanding, thus serving the interests of the lay society, too.

Dedication to quality

Each Frontiers article is a landmark of the highest quality, thanks to genuinely collaborative interactions between authors and review editors, who include some of the world's best academicians. Research must be certified by peers before entering a stream of knowledge that may eventually reach the public - and shape society; therefore, Frontiers only applies the most rigorous and unbiased reviews. Frontiers revolutionizes research publishing by freely delivering the most outstanding research, evaluated with no bias from both the academic and social point of view. By applying the most advanced information technologies, Frontiers is catapulting scholarly publishing into a new generation.

What are Frontiers Research Topics?

Frontiers Research Topics are very popular trademarks of the *Frontiers journals series*: they are collections of at least ten articles, all centered on a particular subject. With their unique mix of varied contributions from Original Research to Review Articles, Frontiers Research Topics unify the most influential researchers, the latest key findings and historical advances in a hot research area.

Find out more on how to host your own Frontiers Research Topic or contribute to one as an author by contacting the Frontiers editorial office: frontiersin.org/about/contact

Advances in nonlinear systems and networks

Topic editors

Fei Yu — Changsha University of Science and Technology, China

Hairong Lin — Hunan University, China

Viet-Thanh Pham — Ton Duc Thang University, Vietnam

Citation

Yu, F., Lin, H., Pham, V.-T., eds. (2023). *Advances in nonlinear systems and networks*. Lausanne: Frontiers Media SA. doi: 10.3389/978-2-83252-046-8

Table of contents

frontiers | Frontiers in Physics

TYPE Editorial
PUBLISHED 14 March 2023
DOI 10.3389/fphy.2023.1180413

OPEN ACCESS

EDITED AND REVIEWED BY
Alex Hansen,
Norwegian University of Science and Technology, Norway

*CORRESPONDENCE
Fei Yu,
yufeiyfyf@csust.edu.cn
Hairong Lin,
haironglin@hnu.edu.cn
Viet-Thanh Pham,
phamvietthanh@tdtu.edu.vn

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 06 March 2023
ACCEPTED 06 March 2023
PUBLISHED 14 March 2023

CITATION
Yu F, Lin H and Pham V-T (2023), Editorial: Advances in non-linear systems and networks.
Front. Phys. 11:1180413.
doi: 10.3389/fphy.2023.1180413

Editorial: Advances in non-linear systems and networks

Fei Yu[1]*, Hairong Lin[2]* and Viet-Thanh Pham[3]*

[1]School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha, China, [2]College of Computer Science and Electronic Engineering, Hunan University, Changsha, China, [3]Faculty of Electrical and Electronics Engineering, Ton Duc Thang University, Ho Chi Minh City, Vietnam

KEYWORDS

editorial, non-linear systems, non-linear networks, non-linear device, application

Editorial on the Research Topic
Advances in non-linear systems and networks

1 Introduction

If there are non-linear elements in the system or network, and the input and output are not superimposed and uniform, such system or network is called non-linear system or non-linear network. Non-linearity makes the whole not equal to the sum of parts, and the superposition principle fails. At some joint points of non-linear system, small changes in parameters often lead to qualitative changes in the form of motion, and behaviors that are essentially different from external excitation appear. It is the non-linear effect that forms the infinite diversity, richness, tortuosity, singularity, complexity, variability and evolution of the material world.

Non-linear systems and networks have broad application prospects in the engineering fields of the Internet of Things, medical care, intelligent systems [1–5], *etc.* With the development of science and technology, according to the current research Frontier, it is not difficult to find that the research fields of non-linear systems and networks are also expanding, including chaotic systems and circuits [6–9], non-linear device models [10–12], memristors [13–16], neural networks [17–21], neural circuits [22–24], synchronous control [25–27] and application research in related fields [28–31].

Therefore, in this Research Topic, 12 articles about non-linear systems and networks and their applications are reported. For non-linear systems, a reverse single side band (RSSB) system with orthogonal frequency division multiplexing (OFDM) signal transmission is designed Chen et al., and a credible and adjustable load resource trading system based on blockchain networks is studied Jiang et al.. For non-linear networks, they have studied the zeroing neural network and its trajectory tracking Zhao et al. and Lan et al., the Hindmash-Rose neural network model Fan et al., the dynamic robustness of the directed network Sun et al., the advanced metering infrastructure (AMI) network Wang et al. and the critical brain wave dynamics of neural networks Galinsky et al.. For non-linear devices, second-order storage elements Liu et al. are studied. For the synchronization control of non-linear systems and networks, the finite-time tracking adaptive iterative learning control method of aircraft track angle system Zhang et al. and finite-time hybrid function projection synchronization Zhang et al. are studied. In addition, considering the application of non-linear systems and

networks, an image encryption algorithm based on three-dimensional (3D) chaotic Hopfield neural network is designed Yao et al.. Finally, the published articles are written by scientists working in major universities and research centers in China and the United States. However, in the second volume, more researchers from outside China should be attracted to participate.

2 Summary of papers presented in this Research Topic

Chen et al., in the paper "*Pilot Optimization for OFDM in the RSSB System*", proposed a reverse single sideband (RSSB) system with orthogonal frequency division multiplexing (OFDM) signal transmission based on the improved pilot interval scheme and pilot power scheme. The improved pilot power scheme proposed in this paper can compensate for frequency selective fading by increasing pilot power in areas with relatively poor channel conditions. The simulation results show that the improved pilot interval scheme and the improved pilot power scheme can improve the system reception sensitivity by 2 dB respectively. The authors believe that these schemes can improve the system performance without increasing the complexity of the algorithm and the cost of RSSB system.

Jiang et al., in the paper "*A Credible and Adjustable Load Resource Trading System Based on Blockchain Networks*", proposed a trusted and adjustable load resource transaction framework based on blockchain, which uses blockchain to realize trusted grid load resource transaction. This paper first proposes a two-layer blockchain architecture based on alliance chain. Then a distributed transaction processing mechanism based on hybrid consensus and fragmentation technology is designed. Finally, a two-level bidding model is proposed. Through a large number of experiments, the authors show that their proposed framework can achieve satisfactory results.

Zhao et al., in the paper "*A Novel Zeroing Neural Network for Dynamic Sylvester Equation Solving and Robot Trajectory Tracking*", proposed a new activation function to ensure fast convergence in predefined times and robustness in the presence of external noise perturbations. This paper theoretically analyzes the effectiveness and robustness of the zeroing neural network system, and verifies it by simulation results. Finally, the proposed theory is applied to robot trajectory tracking to further verify the effectiveness of the proposed method.

Lan et al., in the paper "*Towards Non-linearly Activated ZNN Model For Constrained Manipulator Trajectory Tracking*", proposed a non-linear activation function (NAF), and established a non-linear activation ZNN (NAZNN) model based on NAF. In this paper, the NAZNN model is applied to solve the exact solution of constrained TVLME (CTVLME), and the constrained robot manipulator trajectory tracking (CRMTT) problem is completed. In addition, the authors have also carried out theoretical analysis on the track tracking fault of wheeled robots with physical constraints, and applied the NAZNN model to the problem of manipulator track tracking fault. The experimental results of this paper prove that the NAZNN model can also effectively deal with the problem of manipulator track tracking fault.

Fan et al., in the paper "*Hidden firing patterns and memristor initial condition-offset boosting behavior in a memristive Hindmarsh-Rose neuron model*", proposed a 3D memristive Hindmash-Rose (mHR) neuron model based on an ideal flux-controlled memristor with sinusoidal memductance function and non-linearly modulated input. The numerical results show that the mHR neuron model can generate rich hidden dynamics. Then, memristor initial condition-offset boosting behavior is revealed. This can trigger the generation of an infinite number of coexisting excitation patterns along the variable coordinates of the memristor. Finally, this paper designs an analog circuit to implement the mHR neuron model, and carries out circuit simulation based on PSIM.

Sun et al., in the paper "*A New Effective Metric for Dynamical Robustness of Directed Networks*", studied the dynamic robustness of a directed complex network with additive noise. In the framework of mean square stochastic stability, a new robustness metric is proposed to characterize the synchronization of the network against additive noise. It is found that node dynamics plays a key role in the dynamic robustness of the directed network. They explained and verified it through numerical simulation.

Wang et al., in the paper "*Intrusion detection framework based on homomorphic encryption in AMI network*", proposed an advanced metering infrastructure (AMI) network intrusion detection method based on joint learning client security. First, calculate the direction similarity of the model trained by the data processing center and the model trained by each client. Then, normalize the size of each client model update to the same size as the data processing center model update. Finally, the normalized update and adaptive weights are weighted average. The research results show that this method can effectively resist inference attack and poisoning attack.

Galinsky et al., in the paper "*Critical brain wave dynamics of neuronal avalanches*", studied the potential collective process behind the phenomenological statistics of neuron avalanches, and analyzed that neuron avalanche is only the manifestation of different non-linear sides of the rich wave process in cortical tissue. In this paper, it is found that the wave mode system generates an anharmonic wave mode with time and space scale property through all possible combinations of the third-order non-linear terms described by the general wave Hamiltonian.

Liu et al., in the paper "*AC Power Analysis for Second-order Memory Elements*", deduced the real power, reactive power and apparent power of the proposed second-order memory element, and revealed the difference between the ideal memory element and the traditional passive memory element. The authors quote the corresponding curves, which prove the difference between storage elements, and verify that the harmonic value in the element means that it will continue to provide energy when AC power is used.

Zhang et al., in the paper "*Adaptive iterative learning control method for finite-time tracking of an aircraft track angle system based on a neural network*", proposed an adaptive iterative learning control method based on a neural network. This method can control the aircraft track inclination through the designed control input rudder deflection angle, so as to track the preset trajectory in a limited time interval. Through Lyapunov stability analysis, it can be seen that the designed controller and adaptive laws can stabilize the whole closed-loop system and realize the tracking of target trajectory in a limited time interval. Finally, the paper verifies the feasibility and effectiveness of the theory through a simulation example.

Zhang et al., in the paper "*A new adaptive iterative learning control of finite-time hybrid function projective synchronization for unknown*

time-varying chaotic systems", proposed a new adaptive iterative learning control scheme to solve the finite-time hybrid function projection synchronization problem of chaotic systems with unknown periodic time-varying parameters. Through Lyapunov stability analysis, two different chaotic systems achieve asymptotic synchronization in a finite time interval according to different proportion functions. Finally, the authors proved the feasibility and effectiveness of this method through simulation examples.

Yao et al., in the paper "*An image encryption algorithm based on a 3D chaotic Hopfield neural network and random row-column permutation*", proposed a color image encryption algorithm based on 3D chaotic Hopfield neural network and random row and column arrangement. Firstly, this paper proposes a 3D chaotic Hopfield neural network to generate random sequences for generating diffusion keys and permutation keys. Then, the rows and columns of the original image are randomly arranged according to the arrangement key. Finally, the separately encrypted sub-images are spliced together to obtain the final encrypted image. The simulation results and security analysis show that the encryption scheme has good performance.

3 Concluding remarks

It can be seen that in this Research Topic, we focus on multidisciplinary scientific research by considering non-linear systems and networks. They can be applied to different research fields, including non-linear physics, mathematics, medicine, economics, computer science and engineering. Through this Research Topic, we hope to encourage more scholars and researchers to promote innovative research in non-linear systems and networks and their applications.

Author contributions

All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

Funding

This work was supported by the Natural Science Foundation of Hunan Province under Grants 2022JJ30624, 2022JJ10052 and 2021JJ30741; the Scientific Research Fund of Hunan Provincial Education Department under grant 21B0345; the National Natural Science Foundation of China under Grant 62172058; and the Postgraduate Training Innovation Base Construction Project of Hunan Province under Grant 2020-172-48.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

References

1. Xu Q, Chen X, Chen B, Wu H, Li Z, Bao H. Dynamical analysis of an improved FitzHugh-Nagumo neuron model with multiplier-free implementation. *Nonlinear Dyn* (2023). doi:10.1007/s11071-023-08274-4

2. Yu F, Liu L, Xiao L, Li K, Cai S. A robust and fixed-time zeroing neural dynamics for computing time-variant nonlinear equation using a novel nonlinear activation function. *Neurocomputing* (2019) 350:108–116. doi:10.1016/j.neucom.2019.03.053

3. Yu F, Shen H, Yu Q, Kong X, Sharma PK, Cai S. Privacy protection of medical data based on multi-scroll memristive hopfield neural network. *IEEE Trans Netw Sci Eng* (2023) 10(2):845–58. doi:10.1109/tnse.2022.3223930

4. Ding S, Wang N, Bao H, Chen B, Wu H, Xu Q. Memristor synapse-coupled piecewise-linear simplified Hopfield neural network: Dynamics analysis and circuit implementation. *Chaos, Solitons and Fractals* (2023) 166:112899. doi:10.1016/j.chaos.2022.112899

5. Lin H, Wang C, Cui L, Sun Y, Zhang X, Yao W. Hyperchaotic memristive ring neural network and application in medical image encryption. *Nonlinear Dyn* (2022) 110:841–55. doi:10.1007/s11071-022-07630-0

6. Yu F, Xu S, Xiao X, Yao W, Huang Y, Cai S, et al. Dynamics analysis, FPGA realization and image encryption application of a 5D memristive exponential hyperchaotic system. *Integration* (2023) 90:58–70. doi:10.1016/j.vlsi.2023.01.006

7. Lin H, Wang C, Xu C, Zhang X, Iu HHC. A memristive synapse control method to generate diversified multi-structure chaotic attractors. *IEEE Trans Computer-Aided Des Integrated Circuits Syst* (2022) 42:942–55. doi:10.1109/TCAD.2022.3186516

8. Yu F, Shen H, Zhang Z, Huang Y, Cai S, Du S, et al. A new multi-scroll Chua's circuit with composite hyperbolic tangent-cubic nonlinearity: Complex dynamics, Hardware implementation and Image encryption application. *Integration* (2021) 81: 71–83. doi:10.1016/j.vlsi.2021.05.011

9. Yu F, Zhang W, Xiao X, Yao W, Cai S, Zhang J, et al. Dynamic analysis and FPGA implementation of a new, simple 5D memristive hyperchaotic sprott-C system. *Mathematics* (2023) 11(3):701. Article ID 701, 2023. doi:10.3390/math11030701

10. Hirshy H, Singh M, Casbon MA, Perks RM, Uren MJ, Martin T, et al. Evaluation of pulsed I–V analysis as validation tool of nonlinear RF models of GaN-based HFETs. *IEEE Trans Electron Devices* (2018) 65(12):5307–13. doi:10.1109/ted.2018.2872513

11. Bernardini A, Vergani AE, Sarti A. Wave digital modeling of nonlinear 3-terminal devices for virtual analog applications. *Circuits, Systems, Signal Process.* (2020) 39(7): 3289–319. doi:10.1007/s00034-019-01331-7

12. Cui H, Yao Z, Wang YE. Coupling electromagnetic waves to spin waves: A physics-based nonlinear circuit model for frequency-selective limiters. *IEEE Trans Microwave Theor Tech* (2019) 67(8):3221–9. doi:10.1109/tmtt.2019.2918517

13. Gao L, Ren Q, Sun J, Han ST, Zhou Y. Memristor modeling: Challenges in theories, simulations, and device variability. *J Mater Chem C* (2021) 9(47):16859–84. doi:10.1039/d1tc04201g

14. Ma M, Xiong K, Li Z, Sun Y. Dynamic behavior analysis and synchronization of memristor-coupled heterogeneous discrete neural networks. *Mathematics* (2023) 11: 375. Article ID 375, 2023. doi:10.3390/math11020375

15. Khalid M. Review on various memristor models, characteristics, potential applications, and future works. *Trans Electr Electron Mater* (2019) 20:289–98. doi:10.1007/s42341-019-00116-8

16. Ma M, Lu Y, Li Z, Sun Y, Wang C. Multistability and phase synchronization of rulkov neurons coupled with a locally active discrete memristor. *Fractal Fract* (2023) 7: 82. Article ID 82, 2023. doi:10.3390/fractalfract7010082

17. Yu F, Kong X, Mokbel AAM, Yao W, Cai S. Complex dynamics, hardware implementation and image encryption application of multiscroll memeristive hopfield neural network with a Novel local active memeristor. *IEEE Trans Circuits Systems–II: Express Briefs* (2023) 70(1):326–30. doi:10.1109/tcsii.2022.3218468

18. Lin H, Wang C, Sun Y, Wang T. Generating n-scroll chaotic attractors from a memristor-based magnetized hopfield neural network. *IEEE Trans Circuits Systems–II: Express Briefs* (2023) 70(1):311–5. doi:10.1109/tcsii.2022.3212394

19. Shen H, Yu F, Kong X. Dynamics study on the effect of memristive autapse distribution on Hopfield neural network. *Chaos* (2022) 32(8):083133. Article ID 083133, 2022. doi:10.1063/5.0099466

20. Wan Q, Li F, Chen S, Yang Q. Symmetric multi-scroll attractors in magnetized Hopfield neural network under pulse controlled memristor and pulse current stimulation. *Chaos, Solitons & Fractals* (2023) 169:113259. Article ID 113259. doi:10.1016/j.chaos.2023.113259

21. Shen H, Yu F, Wang C, Sun J, Cai S. Firing mechanism based on single memristive neuron and double memristive coupled neurons. *Nonlinear Dyn* (2022) 110:3807–22. doi:10.1007/s11071-022-07812-w

22. Deng Z, Wang C, Lin H, Sun Y. A memristive spiking neural network circuit with selective supervised Attention algorithm. *IEEE Trans Computer-Aided Des Integrated Circuits Syst* (2022) 1. doi:10.1109/TCAD.2022.3228896

23. Hong Q, Chen H, Sun J, Wang C. Memristive circuit implementation of a self-repairing network based on biological astrocytes in robot application. *IEEE Trans Neural networks Learn Syst* (2020) 33(5):2106–20. doi:10.1109/tnnls.2020.3041624

24. Yan R, Hong Q, Wang C, Sun J, Li Y. Multilayer memristive neural network circuit based on online learning for license plate detection. *IEEE Trans Computer-Aided Des Integrated Circuits Syst* (2021) 41(9):3000–11. doi:10.1109/tcad.2021.3121347

25. Zhou C, Wang C, Yao W, Lin H. Observer-based synchronization of memristive neural networks under DoSattacks and actuator saturation and its application to image encryption. *Appl Maths Comput* (2022) 425:127080. Article ID 127080. doi:10.1016/j.amc.2022.127080

26. Tan F, Zhou L. Analysis of random synchronization under bilayer derivative and nonlinear delay networks of neuron nodes via fixed time policies. *ISA Trans* (2022) 129:114–27. doi:10.1016/j.isatra.2022.01.023

27. Yao W, Wang CH, Sun YC, Zhou C, Lin HR. Exponential multistability of memristive Cohen-Grossberg neural networks with stochastic parameter perturbations. *Appl Maths Comput* (2020) 386:125483. Article ID 125483. doi:10.1016/j.amc.2020.125483

28. Yu F, Kong X, Chen H, Yu Q, Cai S, Huang Y, et al. A 6D fractional-order memristive hopfield neural network and its application in image encryption. *Front Phys* (2022) 10. Article ID 847385, 2022. doi:10.3389/fphy.2022.847385

29. Yu F, Yu Q, Chen H, Kong X, Mokbel AAM, Cai S, et al. Dynamic analysis and audio encryption application in IoT of a multi-scroll fractional-order memristive Hopfield neural network. *Fractal and Fractional* (2022) 6 (7):370. doi:10.3390/fractalfract6070370

30. Zhu Y, Wang C, Sun J, Yu F. A chaotic image encryption method based on the artificial fish swarms algorithm and the DNA coding. *Mathematics* (2023) 11:767. Article ID 767. doi:10.3390/math11030767

31. Yu F, Zhang Z, Shen H, Huang Y, Cai S, Jin J, et al. Design and FPGA implementation of a pseudo-random number generator based on a Hopfield neural network under electromagnetic radiation. *Front Phys* (2021) 9. Article ID 690651, 2021. doi:10.3389/fphy.2021.690651

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 20 December 2022
DOI 10.3389/fphy.2022.1048942

OPEN ACCESS

EDITED BY
Viet-Thanh Pham,
Ton Duc Thang University, Vietnam

REVIEWED BY
Govind Vashishtha,
Sant Longowal Institute of Engineering and Technology, India
NaNa Yang,
Lanzhou University of Technology, China

*CORRESPONDENCE
Chunli Zhang,
gaozhangchunli@163.com

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 10 November 2022
ACCEPTED 28 November 2022
PUBLISHED 20 December 2022

CITATION
Zhang C, Tian X and Yan L (2022), Adaptive iterative learning control method for finite-time tracking of an aircraft track angle system based on a neural network.
Front. Phys. 10:1048942.
doi: 10.3389/fphy.2022.1048942

Adaptive iterative learning control method for finite-time tracking of an aircraft track angle system based on a neural network

Chunli Zhang*, Xu Tian and Lei Yan

Xi'an University of Technology, Shaanxi Key Laboratory of Complex System Control and Intelligent Information Processing, Xi'an, China

Based on a neural network, this paper presents a new adaptive iterative learning control method for the finite-time tracking control problem of an uncertain aircraft track angle system, which can control the aircraft track inclination through the designed control input rudder deflection angle, so that it can track the preset trajectory in a finite time interval. First, the flight path angle system of the aircraft is abstractly modeled by variable substitution to obtain a triangular model in the form of strict feedback. Second, radial basis function neural network approximation is used to model the uncertain part of the system, aiming at the abstract strict feedback model, and two virtual quantities are designed through the three-layer inversion design method, and then, Lyapunov functions are designed for each subsystem to derive virtual control laws, the actual control law, and the neural network weight adaptive laws. Through Lyapunov stability analysis, it can be seen that the designed controller and adaptive laws can make the whole closed-loop system tend to be stable and realize the tracking of a target trajectory in a finite time interval. Finally, the feasibility and effectiveness of the theory are verified by a simulation example.

KEYWORDS

aircraft track angle system, adaptive iterative learning control, neural network, Lyapunov stability, finite-time interval tracking

1 Introduction

Today, aircraft has become an important tool for the human society. People are constantly considering the flight safety of an aircraft, which is followed by the rapid development of aircraft technology. In order to ensure the flight safety of the aircraft, it is necessary to find the optimal flight trajectory that satisfies the trajectory constraints. Therefore, a careful study of aircraft trajectories is required. With the continuous development of the technological era, the control process of the aircraft has become more and more complex [1, 2]. This has led to a new upsurge in the research on aircrafts. However, due to the strong coupling and highly non-linear characteristics of the aircraft

dynamics model, the design of the aircraft control law has certain challenges. This paper mainly studies the flight path angle of the aircraft, designs the control law of the aircraft in the finite time interval, and ensures the safety and stability of the flight process of the aircraft.

Under the current boom in aircraft research, many scholars who study aircraft trajectory planning have emerged. Up to now, it can be roughly divided into four categories: the online real-time trajectory search algorithm based on a large environment [3], target route planning for motion [4], aircraft planning method for multiple tracks, and path planning method for coordinated multiple aircraft working at the same time. For the research on the aircraft track angle system, it is generally adopted to abstract the aircraft track angle model into the aircraft longitudinal model for research [5–7].

Adaptive iterative learning control combines adaptive control and iterative learning control. In the iterative learning control, the characteristics of an adaptive control that can deal with systems with uncertain terms are introduced. Thus, the problem that the adaptive control [8, 9] cannot achieve the desired control effect in a given time is improved. Therefore, many scholars have joined in the theoretical research on the adaptive iterative learning control. For example, in [10], the method of adaptive iterative learning control combined with fuzzy control is introduced into the high-speed train model, which solves the problem that the system has a random varying iteration length and speed and input force constraints and realizes the tracking control of the non-linear and uncertain high-speed train motion system. In [11], a barrier adaptive iterative learning control scheme is proposed, which uses adaptive iterative learning control technology and robust control technology to compensate for parametric and non-parametric uncertainties and asymmetric dead zone non-linearity. The trajectory tracking problem of the tank gun control system under the condition of a non-zero initial error is solved.

A neural network [12–15] is an algorithm mathematical model for distributed parallel information processing by simulating the network behavior characteristics of a biological neural network using bionics ideas. The radial basis function (RBF) neural network is a neural network with RBF as the activation function. The existence of the RBF makes the neural network structure have the characteristics of a local response. Later, people found that a better system accuracy, system robustness, and adaptability can be obtained by using the RBF neural network to approximate. Therefore, they have been paid more attention in the field of non-linear control, which has triggered a large number of scholars' research. As in [16], self-organizing recursive radial basis function neural networks are studied, and a non-linear model predictive control scheme is designed to predict the future dynamic behavior of non-novel systems. In [17], an adaptive gradient multi-objective particle swarm optimization algorithm was designed, the AGMOPSO algorithm was proposed, and it was used in the RBF neural network so as to solve the problem that the RBF neural network converges to the local minimum value.

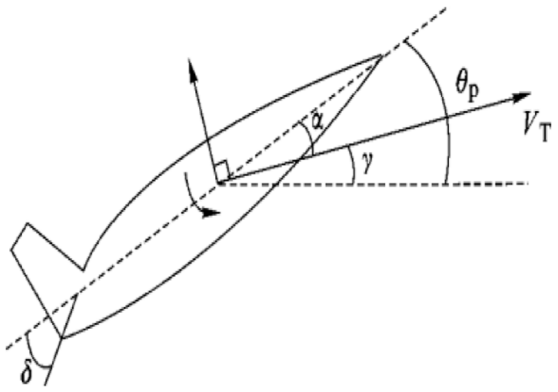


FIGURE 1
Longitudinal model of the aircraft.

The neural network was combined with adaptive iterative learning control to design the controller. The combined application of the neural network and the adaptive iterative learning control system [18, 19] greatly improves the information processing ability and adaptability of the system and has a great impact on the intelligence level of the system. In [20], an adaptive iterative learning control strategy is proposed by using the RBF neural network, which solves the non-uniform trajectory tracking problem of a class of non-linear pure feedback systems with initial state errors. In [21], an iterative learning control algorithm based on the RBF neural network is proposed, which solves the trajectory tracking control up to the rehabilitation robot.

According to the aforementioned discussion, this paper uses the RBF neural network algorithm and the adaptive iterative learning control method to control the longitudinal uncertainty model of the aircraft. Using the characteristics of the RBF neural network approximation model, the uncertainty function in the aircraft is approximated. Using the adaptive iterative learning control to design the control law, on the basis that the closed-loop system tends to converge and stabilize, the system output can better track the desired trajectory within a limited time. Finally, the reliability and stability of the modified method are verified by an example simulation.

2 Model building and a controller design

In this paper, the research on the flight path angle system of the aircraft takes the longitudinal model of the uncertain aircraft as the object, converts it into a strict feedback system with model uncertainty, and then, applies the designed neural network

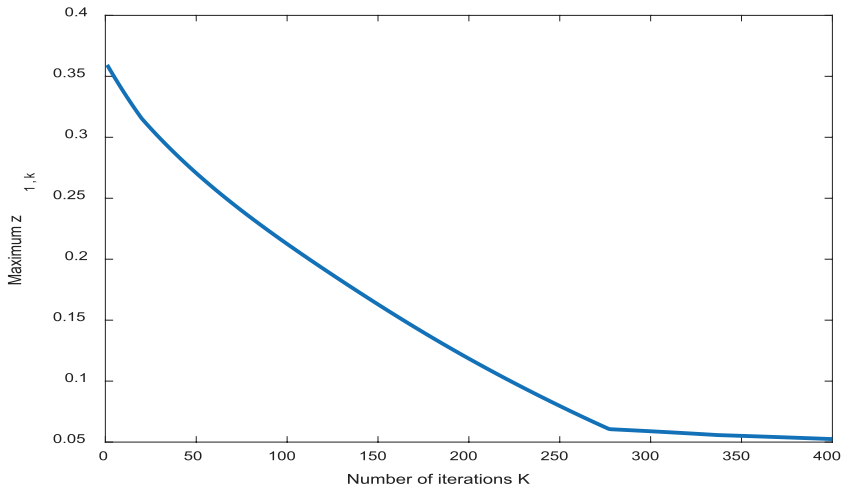


FIGURE 2
Curve of the maximum error $z_{1,k}$ with iteration times.

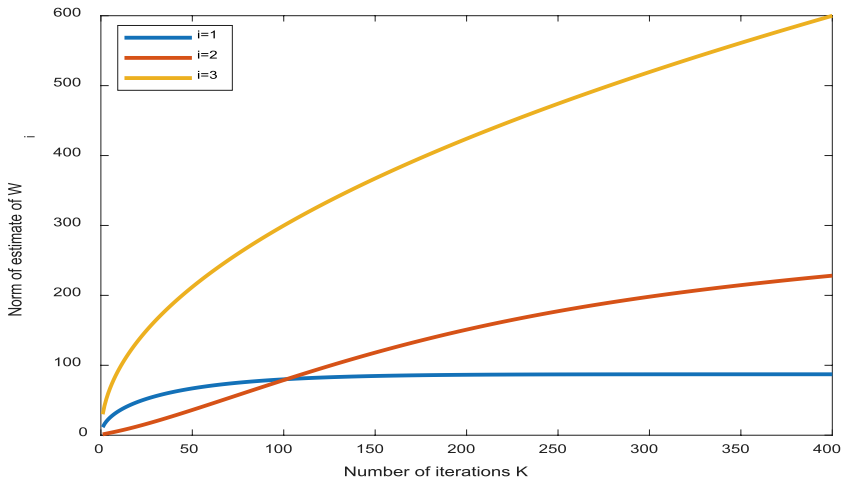


FIGURE 3
Curve of $\|\hat{\omega}_{i,k}\|$ with the number of iterations.

adaptive iterative learning controller to the system to complete the tracking of the ideal trajectory of the aircraft.

2.1 System specification

Due to the strong coupling and highly non-linear characteristics of the dynamic model of the aircraft, this paper considers controlling the inclination of the aircraft track by inputting the ideal inclination of the rudder surface of the aircraft.

The simplified longitudinal model of the aircraft is shown in Figure 1:

The simplified model is

$$\begin{cases} \dot{\gamma} = \bar{L}_\alpha \alpha - \dfrac{g}{V_T}\cos\gamma + \bar{L}_o, \\ \dot{\alpha} = q + \dfrac{g}{V_T}\cos\gamma - \bar{L}_\alpha \alpha - \bar{L}_o, \\ \dot{\theta}_p = q, \\ \dot{q} = M_o + M_\delta \delta, \end{cases} \tag{1}$$

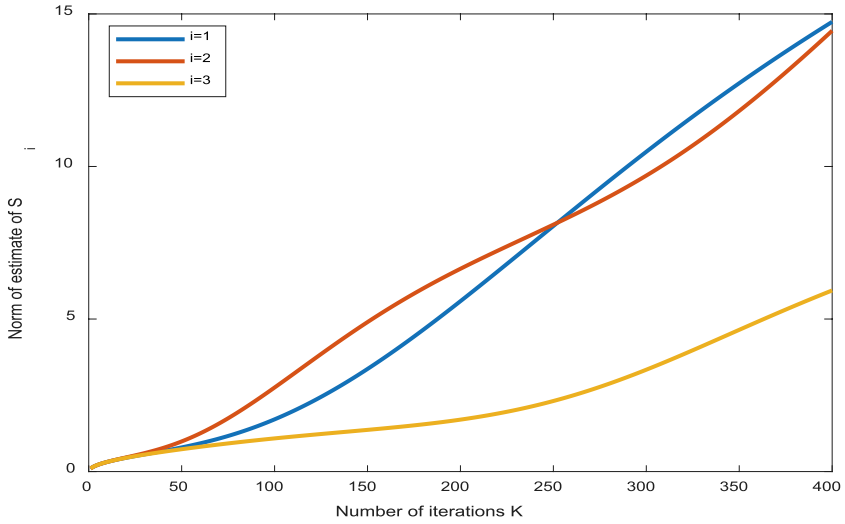


FIGURE 4
Curve of $\|\hat{S}_{i,k}\|$ with the number of iterations.

where we see $\bar{L}_o = \frac{L_o}{mV_T}$ and $\bar{L}_\alpha = \frac{L_\alpha}{mV_T}$, where γ is the inclination of the aircraft track. α is the angle of attack of the aircraft. θ_p is the pitch angle of the aircraft. q is the change speed of the pitch angle. V_T is the flight speed. m is the mass of the aircraft. g is the acceleration of gravity. $\bar{L}_\alpha$ is the slope of the lift curve. $\bar{L}_o$ is the other influencing factor of the lift. M_δ is the control pitch moment. M_o is the moment from other sources, which is approximately replaced by $M_o = M_\alpha + M_\delta\delta$. δ is the deflection angle of the rudder surface. At any time, the slope of the lift curve, other influencing factors of the lift, control pitch moment, other source moment, and other values are all unknown constants.

By defining the states $x_{1,k} = \gamma$, $x_{2,k} = \alpha$, and $x_{3,k} = q$, the control input is the declination angle of the rudder surface $u_k = \delta$; at this time, considering the uncertainty, the following triangular model under a strict feedback form is obtained:

$$\begin{cases} \dot{x}_{1,k} = a_1 x_{2,k} + W_{1,k}(x_{1,k}, t), \\ \dot{x}_{2,k} = x_{3,k} + W_{2,k}(x_{1,k}, x_{2,k}, t), \\ \dot{x}_{3,k} = a_3 u_k + W_{3,k}(x_{2,k}, x_{3,k}, t), \end{cases} \tag{2}$$

where $W_{1,k} = f_{1,k}(x_{1,k}) + \Delta_{1,k}(x_k, t)$, $W_{2,k} = f_{2,k}(x_{1,k}, x_{2,k}) + \Delta_{2,k}(x_k, t)$, $W_{3,k} = f_{3,k}(x_{2,k}, x_{3,k}) + \Delta_{3,k}(x_k, t)$, and $\Delta_{i,k}(x_k, t)$, $i = 1, 2, 3$ are the uncertain parts, $|\Delta_{i,k}(x_k, t)| \le \rho_i$, and ρ_i is a positive real number and

$$\begin{cases} f_{1,k}(x_{1,k}) = -\dfrac{g}{V_T}\cos x_{1,k} + \bar{L}_0, \\ f_{2,k}(x_{1,k}, x_{2,k}) = \dfrac{g}{V_T}\cos x_{1,k} - \bar{L}_0 - \bar{L}_\alpha x_{2,k}, \\ f_{3,k}(x_{2,k}, x_{3,k}) = M_\alpha x_{2,k} + M_q x_{2,k}. \end{cases} \tag{3}$$

At the same time, $a_1 = \bar{L}_\alpha > 0, a_3 = M_\delta > 0$.

The following assumptions about the model will be used in the controller design process.

Assumption 1. The speed V_T will stabilize within a small region of the ideal value through a linear controller, which is treated as a constant.

Assumption 2. All state variables can be solved and used for feedback.

Assumption 3. The bounds of the unknown parameters are known, that is to say, for $= 1, 3$, there are known positive numbers a_{im} and a_{iM} such that $a_{im} \le a_i \le a_{iM}$.

Assumption 4. The ideal trajectory is bounded, whose first and second derivatives exist, and $x_{1d}^2 + \dot{x}_{1d}^2 + \ddot{x}_{1d}^2 \le \chi$ is satisfied for a positive real number χ.

The control objective is to design a neural network adaptive iterative learning controller u_k, which makes the output of the system $y_k(t)$ track the ideal trajectory $y_r(t)$ in a limited time $[0, T]$.

2.2 Design of a neural network adaptive iterative learning controller for the aircraft track angle system

During the design of the controller, the following definition and lemma of the convergent series sequence will be used.

Definition 1. The convergent series sequence Δ_k is defined as

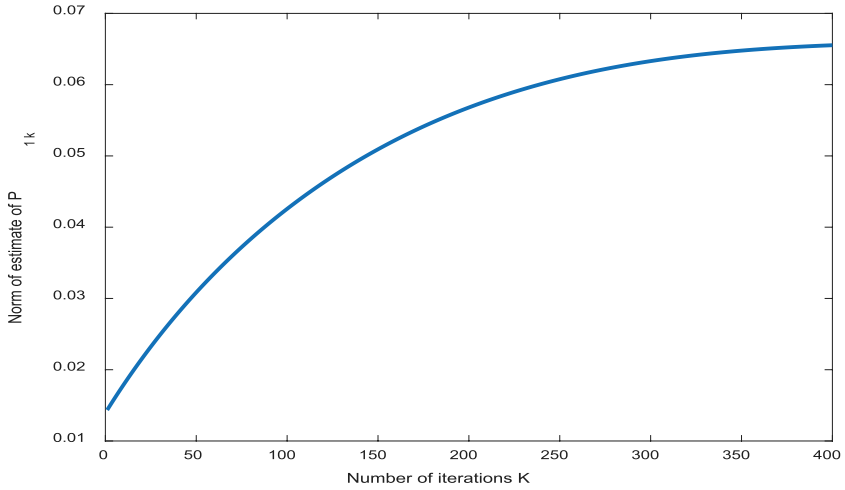


FIGURE 5
Curve of $\|\hat{P}_{1,k}\|$ with the number of iterations.

$$\Delta_k = \frac{a}{k^l}, \tag{4}$$

where $k = 1, 2, \cdots$; a and l are constant parameters that need to be designed, satisfying $a > 0 \in R$ and $l \geq 2 \in N$.

Lemma 1. For a given sequence $\left\{\frac{1}{k^l}\right\}$, where $k = 1, 2, \cdots$ and the positive integer $l \geq 2$, the following inequality holds:

$$\lim_{k\to\infty} \sum_{i=1}^{k} \frac{1}{i^l} \leq 2. \tag{5}$$

Next, the whole process of the controller design is given.

Step 1. Define three errors.

Define the error between the first actual trajectory and the ideal trajectory as

$$z_{1,k} = y_k - y_r = x_{1,k} - y_r. \tag{6}$$

Define the error between the first virtual control variable $x_{2,k}$ and the first virtual controller $\alpha_{1,k}$ as

$$z_{2,k} = x_{2,k} - \alpha_{1,k}. \tag{7}$$

Define the error between the second virtual control variable $x_{3,k}$ and the second virtual controller $\alpha_{2,k}$ as

$$z_{3,k} = x_{3,k} - \alpha_{2,k}. \tag{8}$$

Derive and combine it with model Eq. 2 to get

$$\dot{z}_{1,k} = \dot{x}_{1,k} - \dot{y}_r = a_1 x_{2,k} + W_{1,k} - \dot{y}_r = a_1\left(x_{2,k} + \frac{W_{1,k}}{a_1} - \frac{1}{a_1}\dot{y}_r\right), \tag{9}$$

$$\dot{z}_{2,k} = \dot{x}_{2,k} - \dot{\alpha}_{1,k} = x_{3,k} + W_{2,k} - \dot{\alpha}_{1,k}, \tag{10}$$

$$\dot{z}_{3,k} = \dot{x}_{3,k} - \dot{\alpha}_{2,k} = a_3 u_k + W_{3,k} - \dot{\alpha}_{2,k} = a_3\left(u_k + \frac{W_{3,k}}{a_3} - \frac{1}{a_3}\dot{\alpha}_{2,k}\right). \tag{11}$$

Step 2. Approximate the unknown parts in step 1 with RBF neural networks.

Let

$$\begin{cases} \dfrac{W_{1,k}}{a_1} = \omega_1^{*T}\xi_1\left(x_{1,k}, t\right) + \sigma_{1,k}\left(t\right), \\ W_{2,k} = \omega_2^{*T}\xi_2\left(x_{1,k}, x_{2,k}, t\right) + \sigma_{2,k}\left(t\right), \\ \dfrac{W_{3,k}}{a_3} = \omega_3^{*T}\xi_3\left(x_{2,k}, x_{3,k}, t\right) + \sigma_{3,k}\left(t\right), \end{cases} \tag{12}$$

where ω_1^*, ω_2^*, and ω_3^* are the ideal weight, $\|\omega_1^*\| \leq \omega_M$, $\|\omega_2^*\| \leq \omega_M$, $\|\omega_3^*\| \leq \omega_M$, and $\omega_1^*, \omega_2^*, \omega_3^*$ are unknown parameters, The corresponding adaptive control law needs to be designed for estimation, and the specific design will be explained later. $\sigma_{1,k}, \sigma_{2,k}, and\ \sigma_{3,k}$ are approximation errors, and $|\sigma_{1,k}| \leq \sigma_M, |\sigma_{2,k}| \leq \sigma_M, |\sigma_{3,k}| \leq \sigma_M$.

Simultaneously,

$$\dot{z}_{1,k} = a_1\left(x_{2,k} + \omega_1^{*T}\xi_1 + \sigma_{1,k} - \frac{1}{a_1}\dot{y}_r\right), \tag{13}$$

$$\dot{z}_{2,k} = x_{3,k} + \omega_2^{*T}\xi_2 + \sigma_{2,k} - \dot{\alpha}_{1,k}, \tag{14}$$

$$\dot{z}_{3,k} = a_3\left(u_k + \omega_3^{*T}\xi_3 + \sigma_{3,k} - \frac{1}{a_3}\dot{\alpha}_{2,k}\right). \tag{15}$$

Define

$$\omega_1^T = \omega_1^{*T}, \omega_2^T = \omega_2^{*T}, \omega_3^T = \omega_3^{*T}, \tag{16}$$

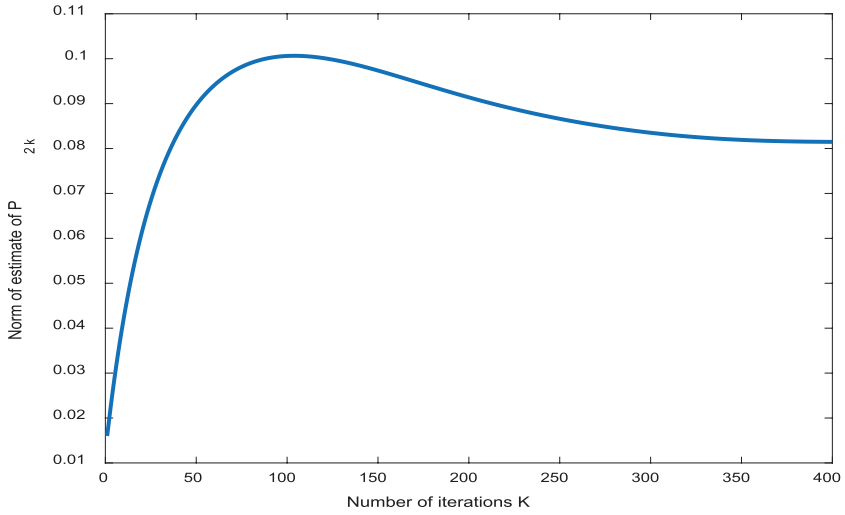


FIGURE 6
Curve of $\|\hat{P}_{2,k}\|$ with the number of iterations.

$$\frac{1}{a_1} = P_1,\ \frac{1}{a_3} = P_2. \tag{17}$$

Step 3. Select the Lyapunov function to design the controller.

According to the closed-loop system composed of the triangular model and the actual controller in the form of strict feedback, in the case of satisfying the four assumptions, it can be obtained from the second principle of Lyapunov stability in the preliminary knowledge, the set $V(x)$ must satisfy the positive definite condition, and the reciprocal satisfies the negative semidefinite condition to achieve asymptotic stability.

The Lyapunov function is designed as follows:

$$V_{1,k} = \frac{1}{2}z_{1,k}^2 + \frac{a_1}{2}\tilde{\omega}_{1,k}^T\Gamma_1^{-1}\tilde{\omega}_{1,k} + \frac{a_1}{2}\Gamma_2^{-1}\tilde{S}_{1,k}^2 + \frac{a_1}{2}\Gamma_3^{-1}\tilde{P}_{1,k}^2, \tag{18}$$

$$V_{2,k} = \frac{1}{2}z_{2,k}^2 + \frac{1}{2}\tilde{\omega}_{2,k}^T\Gamma_4^{-1}\tilde{\omega}_{2,k} + \frac{1}{2}\Gamma_5^{-1}\tilde{S}_{2,k}^2 + \frac{1}{a_1}V_{1,k}, \tag{19}$$

$$V_k = \frac{1}{2}z_{3,k}^2 + \frac{a_3}{2}\tilde{\omega}_{3,k}^T\Gamma_6^{-1}\tilde{\omega}_{3,k} + \frac{a_3}{2}\Gamma_7^{-1}\tilde{S}_{3,k}^2 + \frac{a_3}{2}\Gamma_8^{-1}\tilde{P}_{2,k}^2 + a_3V_{2,k}, \tag{20}$$

where $\tilde{\omega}_{i,k} = \hat{\omega}_{i,k} - \omega_i$, $\tilde{S}_{i,k} = \hat{S}_{i,k} - S$, $i = 1, 2, 3$, $\tilde{P}_{1,k} = \hat{P}_{1,k} - P_1$, $\tilde{P}_{2,k} = \hat{P}_{2,k} - P_2$, and $S = \sigma_M^2$. $\hat{\omega}_{i,k}$, $\hat{S}_{i,k}$, $\hat{P}_{1,k}$, and $\hat{P}_{2,k}$ are estimates of ω_i, S_i, P_1, and P_2, respectively.

The virtual control laws $\alpha_{1,k}$ and $\alpha_{2,k}$ and the actual control law u_k are designed as

$$\alpha_{1,k} = -\hat{\omega}_{1,k}^T\xi_1 - \frac{1}{\Delta_k}z_{1,k}\hat{S}_{1,k} - c_1z_{1,k} + \hat{P}_{1,k}\dot{y}_r, \tag{21}$$

$$\alpha_{2,k} = -\hat{\omega}_{2,k}^T\xi_2 - \frac{1}{\Delta_k}z_{2,k}\hat{S}_{2,k} - c_2z_{2,k} + \dot{\alpha}_{1,k} - z_{1,k}, \tag{22}$$

$$u_k = -\hat{\omega}_{3,k}^T\xi_3 - \frac{1}{\Delta_k}z_{3,k}\hat{S}_{3,k} - c_3z_{3,k} + \hat{P}_{2,k}\dot{\alpha}_{2,k} - z_{2,k}, \tag{23}$$

where c_1, c_2, and c_3 are normal parameters that can be designed.

Step 4. Design the parameter update law.

$$\dot{\hat{\omega}}_{1,k} = \Gamma_1\xi_1z_{1,k},\ \dot{\hat{S}}_{1,k} = \Gamma_2\frac{1}{\Delta_k}z_{1,k}^2,\ \dot{\hat{P}}_{1,k} = \Gamma_3\dot{y}_rz_{1,k}, \tag{24}$$

$$\dot{\hat{\omega}}_{2,k} = \Gamma_4\xi_2z_{2,k},\ \dot{\hat{S}}_{2,k} = \Gamma_5\frac{1}{\Delta_k}z_{2,k}^2, \tag{25}$$

$$\dot{\hat{\omega}}_{3,k} = \Gamma_6\xi_3z_{3,k},\ \dot{\hat{S}}_{3,k} = \Gamma_7\frac{1}{\Delta_k}z_{3,k}^2,\ \dot{\hat{P}}_{2,k} = \Gamma_8\dot{\alpha}_{2,k}z_{3,k}, \tag{26}$$

where $\Gamma_i, i = 1, \cdots, 8$ are a positive definite diagonal gain matrix of suitable dimension, $\Gamma_i = \Gamma_i^T > 0$.

Assumption 5. As far as the initial state is concerned, for any k, when $t = 0$, $x_{1,k}(0) = y_r(0)$, $\hat{\omega}_{i,k}(0) = \hat{\omega}_{i,k-1}(T)$, $\hat{S}_{i,k}(0) = \hat{S}_{i,k-1}(T)$, $\hat{P}_{1,k}(0) = \hat{P}_{1,k-1}(T)$, and $\hat{P}_{2,k}(0) = \hat{P}_{2,k-1}(T)$.

3 Stability analysis

According to the obtained strict feedback model Eq. 2 and the specific controller designed in Section 2.2, the stability analysis of the designed controller will be carried out in the following sections.

Theorem 1. Under the condition that assumptions 1–5 are satisfied and the stability function at the initial equilibrium

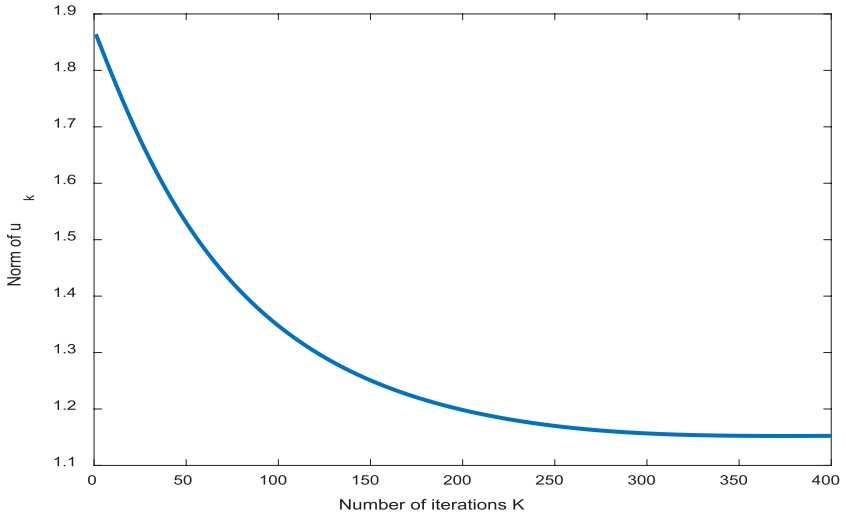


FIGURE 7
Curve of $\|u_k\|$ with the number of iterations.

state is less than any normal number, design virtual control laws (21) and (22), the actual control law (23), and the parameter update law (24–26) can observe that all signals of the closed-loop system are bounded on $[0, T]$, and the tracking error $z_{i,k}(t), i = 1, 2, 3$ converges asymptotically.

According to the assumptions, definition, and lemma, it is easy to prove that the conclusion of the theorem holds. The proof process is as follows:

$V_{1,k}$ and the derivation process of the error system (13) are as follows:

$$\begin{aligned}
\dot{V}_{1,k} &= z_{1,k}\dot{z}_{1,k} + a_1\tilde{\omega}_{1,k}^T\Gamma_1^{-1}\dot{\hat{\omega}}_{1,k} + a_1\Gamma_2^{-1}\tilde{S}_{1,k}\dot{\hat{S}}_{1,k} + a_1\Gamma_3^{-1}\tilde{P}_{1,k}\dot{\hat{P}}_{1,k}\\
&= a_1 z_{1,k}\left(z_{2,k} + \alpha_{1,k} + \omega_1^T\xi_1 + \sigma_{1,k} - P_1\dot{y}_r\right)\\
&\quad + a_1\tilde{\omega}_{1,k}^T\Gamma_1^{-1}\dot{\hat{\omega}}_{1,k} + a_1\Gamma_2^{-1}\tilde{S}_{1,k}\dot{\hat{S}}_{1,k} + a_1\Gamma_3^{-1}\tilde{P}_{1,k}\dot{\hat{P}}_{1,k}\\
&= a_1 z_{1,k}\left(z_{2,k} + \alpha_{1,k} + \omega_1^T\xi_1 - P_1\dot{y}_r\right) + a_1 z_{1,k}\sigma_{1,k}\\
&\quad + a_1\tilde{\omega}_{1,k}^T\Gamma_1^{-1}\dot{\hat{\omega}}_{1,k} + a_1\Gamma_2^{-1}\tilde{S}_{1,k}\dot{\hat{S}}_{1,k} + a_1\Gamma_3^{-1}\tilde{P}_{1,k}\dot{\hat{P}}_{1,k} \le a_1 z_{1,k}\\
&\quad \left(z_{2,k} + \alpha_{1,k} + \omega_1^T\xi_1 - P_1\dot{y}_r\right) + a_1\left|z_{1,k}\right|\left|\sigma_{1,k}\right|\\
&\quad + a_1\tilde{\omega}_{1,k}^T\Gamma_1^{-1}\dot{\hat{\omega}}_{1,k} + a_1\Gamma_2^{-1}\tilde{S}_{1,k}\dot{\hat{S}}_{1,k} + a_1\Gamma_3^{-1}\tilde{P}_{1,k}\dot{\hat{P}}_{1,k} \le a_1 z_{1,k}\\
&\quad \left(z_{2,k} + \alpha_{1,k} + \omega_1^T\xi_1 - P_1\dot{y}_r\right) + a_1\left|z_{1,k}\right|\sigma_M\\
&\quad + a_1\tilde{\omega}_{1,k}^T\Gamma_1^{-1}\dot{\hat{\omega}}_{1,k} + a_1\Gamma_2^{-1}\tilde{S}_{1,k}\dot{\hat{S}}_{1,k} + a_1\Gamma_3^{-1}\tilde{P}_{1,k}\dot{\hat{P}}_{1,k} \le a_1 z_{1,k}\\
&\quad \left(z_{2,k} + \alpha_{1,k} + \omega_1^T\xi_1 - P_1\dot{y}_r\right) + \frac{a_1}{\Delta_k}z_{1,k}^2\sigma_M^2 + \frac{a_1}{4}\Delta_k\\
&\quad + a_1\tilde{\omega}_{1,k}^T\Gamma_1^{-1}\dot{\hat{\omega}}_{1,k} + a_1\Gamma_2^{-1}\tilde{S}_{1,k}\dot{\hat{S}}_{1,k} + a_1\Gamma_3^{-1}\tilde{P}_{1,k}\dot{\hat{P}}_{1,k}\\
&= a_1 z_{1,k}\left(z_{2,k} + \alpha_{1,k} + \omega_1^T\xi_1 - P_1\dot{y}_r\right) + \frac{a_1}{\Delta_k}z_{1,k}^2 S + \frac{a_1}{4}\Delta_k\\
&\quad + a_1\tilde{\omega}_{1,k}^T\Gamma_1^{-1}\dot{\hat{\omega}}_{1,k} + a_1\Gamma_2^{-1}\tilde{S}_{1,k}\dot{\hat{S}}_{1,k} + a_1\Gamma_3^{-1}\tilde{P}_{1,k}\dot{\hat{P}}_{1,k}. \qquad (27)
\end{aligned}$$

$V_{2,k}$ and the derivation process of the error system (14) are as follows:

$$\begin{aligned}
\dot{V}_{2,k} &= z_{2,k}\dot{z}_{2,k} + \tilde{\omega}_{2,k}^T\Gamma_4^{-1}\dot{\hat{\omega}}_{2,k} + \Gamma_5^{-1}\tilde{S}_{2,k}\dot{\hat{S}}_{2,k} + \frac{1}{a_1}\dot{V}_{1,k} = z_{2,k}\\
&\quad \left(z_{3,k} + \alpha_{2,k} + \omega_2^T\xi_2 - \dot{\alpha}_{1,k}\right) + z_{2,k}\sigma_{2,k}\\
&\quad + \tilde{\omega}_{2,k}^T\Gamma_4^{-1}\dot{\hat{\omega}}_{2,k} + \Gamma_5^{-1}\tilde{S}_{2,k}\dot{\hat{S}}_{2,k} + \frac{1}{a_1}\dot{V}_{1,k} \le z_{2,k}\left(z_{3,k} + \alpha_{2,k} + \omega_2^T\xi_2 - \dot{\alpha}_{1,k}\right)\\
&\quad + \left|z_{2,k}\right|\left|\sigma_{2,k}\right|\\
&\quad + \tilde{\omega}_{2,k}^T\Gamma_4^{-1}\dot{\hat{\omega}}_{2,k} + \Gamma_5^{-1}\tilde{S}_{2,k}\dot{\hat{S}}_{2,k} + \frac{1}{a_1}\dot{V}_{1,k} \le z_{2,k}\left(z_{3,k} + \alpha_{2,k} + \omega_2^T\xi_2 - \dot{\alpha}_{1,k}\right)\\
&\quad + \left|z_{2,k}\right|\sigma_M\\
&\quad + \tilde{\omega}_{2,k}^T\Gamma_4^{-1}\dot{\hat{\omega}}_{2,k} + \Gamma_5^{-1}\tilde{S}_{2,k}\dot{\hat{S}}_{2,k} + \frac{1}{a_1}\dot{V}_{1,k} \le z_{2,k}\left(z_{3,k} + \alpha_{2,k} + \omega_2^T\xi_2 - \dot{\alpha}_{1,k}\right)\\
&\quad + \frac{1}{\Delta_k}z_{2,k}^2\sigma_M^2 + \frac{1}{4}\Delta_k\\
&\quad + \tilde{\omega}_{2,k}^T\Gamma_4^{-1}\dot{\hat{\omega}}_{2,k} + \Gamma_5^{-1}\tilde{S}_{2,k}\dot{\hat{S}}_{2,k} + \frac{1}{a_1}\dot{V}_{1,k} = z_{2,k}\left(z_{3,k} + \alpha_{2,k} + \omega_2^T\xi_2 - \dot{\alpha}_{1,k}\right)\\
&\quad + \frac{1}{\Delta_k}z_{2,k}^2 S + \frac{1}{4}\Delta_k + \tilde{\omega}_{2,k}^T\Gamma_4^{-1}\dot{\hat{\omega}}_{2,k} + \Gamma_5^{-1}\tilde{S}_{2,k}\dot{\hat{S}}_{2,k} + \frac{1}{a_1}\dot{V}_{1,k}. \qquad (28)
\end{aligned}$$

V_k and the derivation process of the error system (15) are as follows:

$$\begin{aligned}
\dot{V}_k &= z_{3,k}\dot{z}_{3,k} + a_3\tilde{\omega}_{3,k}^T\Gamma_6^{-1}\dot{\hat{\omega}}_{3,k} + a_3\Gamma_7^{-1}\tilde{S}_{3,k}\dot{\hat{S}}_{3,k} + a_3\Gamma_8^{-1}\tilde{P}_{2,k}\dot{\hat{P}}_{2,k} + a_3\dot{V}_{2,k}\\
&= a_3 z_{3,k}\left(u_k + \omega_3^T\xi_3 + \sigma_{3,k} - P_{2,k}\dot{\alpha}_{2,k}\right) + a_3\tilde{\omega}_{3,k}^T\Gamma_6^{-1}\dot{\hat{\omega}}_{3,k} + a_3\Gamma_7^{-1}\tilde{S}_{3,k}\dot{\hat{S}}_{3,k}\\
&\quad + a_3\Gamma_8^{-1}\tilde{P}_{2,k}\dot{\hat{P}}_{2,k} + a_3\dot{V}_{2,k} = a_3 z_{3,k}\left(u_k + \omega_3^T\xi_3 - P_2\dot{\alpha}_{2,k}\right) + a_3 z_{3,k}\sigma_{3,k}\\
&\quad + a_3\tilde{\omega}_{3,k}^T\Gamma_6^{-1}\dot{\hat{\omega}}_{3,k} + a_3\Gamma_7^{-1}\tilde{S}_{3,k}\dot{\hat{S}}_{3,k} + a_3\Gamma_8^{-1}\tilde{P}_{2,k}\dot{\hat{P}}_{2,k} + a_3\dot{V}_{2,k} \le a_3 z_{3,k}\\
&\quad \left(u_k + \omega_3^T\xi_3 - P_2\dot{\alpha}_{2,k}\right) + a_3\left|z_{3,k}\right|\left|\sigma_{3,k}\right|\\
&\quad + a_3\tilde{\omega}_{3,k}^T\Gamma_6^{-1}\dot{\hat{\omega}}_{3,k} + a_3\Gamma_7^{-1}\tilde{S}_{3,k}\dot{\hat{S}}_{3,k} + a_3\Gamma_8^{-1}\tilde{P}_{2,k}\dot{\hat{P}}_{2,k} + a_3\dot{V}_{2,k} \le a_3 z_{3,k}\\
&\quad \left(u_k + \omega_3^T\xi_3 - P_2\dot{\alpha}_{2,k}\right) + a_3\left|z_{3,k}\right|\sigma_M\\
&\quad + a_3\tilde{\omega}_{3,k}^T\Gamma_6^{-1}\dot{\hat{\omega}}_{3,k} + a_3\Gamma_7^{-1}\tilde{S}_{3,k}\dot{\hat{S}}_{3,k} + a_3\Gamma_8^{-1}\tilde{P}_{2,k}\dot{\hat{P}}_{2,k} + a_3\dot{V}_{2,k} \le a_3 z_{3,k}\\
&\quad \left(u_k + \omega_3^T\xi_3 - P_2\dot{\alpha}_{2,k}\right) + \frac{a_3}{\Delta_k}z_{3,k}^2\sigma_M^2 + \frac{a_3}{4}\Delta_k\\
&\quad + a_3\tilde{\omega}_{3,k}^T\Gamma_6^{-1}\dot{\hat{\omega}}_{3,k} + a_3\Gamma_7^{-1}\tilde{S}_{3,k}\dot{\hat{S}}_{3,k} + a_3\Gamma_8^{-1}\tilde{P}_{2,k}\dot{\hat{P}}_{2,k} + a_3\dot{V}_{2,k} = a_3 z_{3,k}\\
&\quad \left(u_k + \omega_3^T\xi_3 - P_2\dot{\alpha}_{2,k}\right) + \frac{a_3}{\Delta_k}z_{3,k}^2\sigma_M^2\\
&\quad + \frac{a_3}{4}\Delta_k + a_3\tilde{\omega}_{3,k}^T\Gamma_6^{-1}\dot{\hat{\omega}}_{3,k} + a_3\Gamma_7^{-1}\tilde{S}_{3,k}\dot{\hat{S}}_{3,k} + a_3\Gamma_8^{-1}\tilde{P}_{2,k}\dot{\hat{P}}_{2,k} + a_3\dot{V}_{2,k}. \qquad (29)
\end{aligned}$$

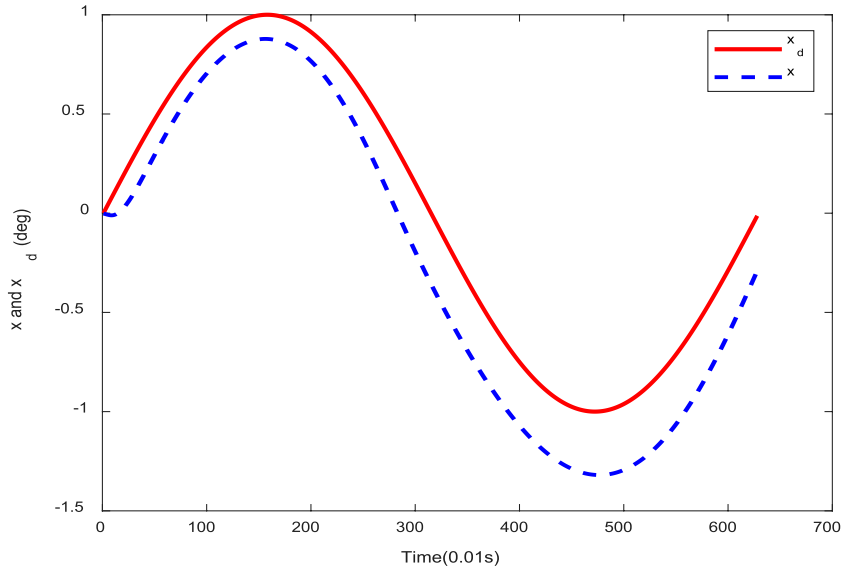


FIGURE 8
Trajectory trace graph without iterations.

Substitute Eqs 21 and 24 into Eq. 27 to get

$$\dot{V}_{1,k} \le a_1 z_{1,k} z_{2,k} - a_1 c_1 z_{1,k}^2 + \frac{a_1}{4}\Delta_k. \tag{30}$$

Substitute Eqs 22 and 25 into Eq. 28 to get

$$\dot{V}_{2,k} \le z_{2,k} z_{3,k} - c_1 z_{1,k}^2 - c_2 z_{2,k}^2 + \frac{2}{4}\Delta_k. \tag{31}$$

Substitute Eqs 23 and 26 into Eq. 29 to get

$$\dot{V}_k \le -a_3 c_1 z_{1,k}^2 - a_3 c_2 z_{2,k}^2 - a_3 c_3 z_{3,k}^2 + \frac{3a_3}{4}\Delta_k, \tag{32}$$

where for any $r > 0$, we have $mn \le \frac{1}{r}m^2 + \frac{1}{4}n^2 r \quad (r = \Delta_k)$.

According to Assumption 1, there are $z_{i,k}(0)^2 = 0 \le z_{i,k}(T)^2$ and $i = 1, 2, 3$, and by Eq. 20, we get

$$\begin{aligned} &V_k\left(z_{i,k}(0), \hat{\omega}_{i,k}(T), \hat{S}_{i,k}(T), \hat{P}_{1,k}(T), \hat{P}_{2,k}(T)\right) \\ &\le V_k\left(z_{i,k}(0), \hat{\omega}_{i,k}(0), \hat{S}_{i,k}(0), \hat{P}_{1,k}(0), \hat{P}_{2,k}(0)\right) + \int_0^T V_k dt. \end{aligned} \tag{33}$$

Substituting Eq. 32 into Eq. 33, we get

$$\begin{aligned} &V_k\left(z_{i,k}(0), \hat{\omega}_{i,k}(T), \hat{S}_{i,k}(T), \hat{P}_{1,k}(T), \hat{P}_{2,k}(T)\right) \\ &\le V_1\left(z_{i,k}(0), \hat{\omega}_{i,k}(0), \hat{S}_{i,k}(0), \hat{P}_{1,k}(0), \hat{P}_{2,k}(0)\right) \\ &- \sum_{i=1}^{3}\sum_{j=1}^{k}\int_0^T a_3 c_i z_{i,j}^2 dt + \frac{3a_3}{4}\sum_{j=1}^{k}\Delta_j T. \end{aligned} \tag{34}$$

Let $V_0 = V_1\left(z_{i,k}(0), \hat{\omega}_{i,k}(0), \hat{S}_{i,k}(0), \hat{P}_{1,k}(0), \hat{P}_{2,k}(0)\right) + \frac{3a_3}{4}\sum_{j=1}^{k}\Delta_j T$ be substituted into Eq. 34, rewritten as

$$\begin{aligned} &\sum_{i=1}^{3}\sum_{j=1}^{k}\int_0^T a_3 c_i z_{i,j}^2 dt \le V_0(k) \\ &- V_k\left(z_{i,k}(0), \hat{\omega}_{i,k}(T), \hat{S}_{i,k}(T), \hat{P}_{1,k}(T), \hat{P}_{2,k}(T)\right). \end{aligned} \tag{35}$$

According to Eq. 5, $\lim_{k\to\infty} V_0(k) \le V_1 + \frac{2a}{4}(3a_3)T$, $V_0(k)$ is bounded, and $V_k\left(z_{i,k}(0), \hat{\omega}_{i,k}(T), \hat{S}_{i,k}(T), \hat{P}_{1,k}(T), \hat{P}_{2,k}(T)\right) \ge 0$, so

$$\lim_{k\to\infty}\sum_{i=1}^{3}\int_0^T a_3 c_i z_{i,k}^2 dt = 0. \tag{36}$$

According to Eq. 20, for any k, $V_k(t) = V_k(0) + \int_0^t \dot{V}_k(\tau)d\tau$, Eq. 29 is substituted, then

$$V_k(t) = V_k(0) - \sum_{i=1}^{3}\int_0^T a_3 c_i z_{i,k}^2 d\tau + t\frac{3a_3}{4}\Delta_k. \tag{37}$$

According to Eq. 36, $\sum_{i=1}^{3}\int_0^T a_3 c_i z_{i,k}^2 dt$ is bounded. According to Definition 1, Δ_k is bounded, and $t \in [0, T]$; therefore, $t\frac{3a_3}{4}\Delta_k$ is bounded.

According to $\hat{\omega}_{i,k}(0) = \hat{\omega}_{i,k-1}(T)$, $\hat{S}_{i,k}(0) = \hat{S}_{i,k-1}(T)$, $\hat{P}_{1,k}(0) = \hat{P}_{1,k-1}(T)$, $\hat{P}_{2,k}(0) = \hat{P}_{2,k-1}(T)\,(i = 1, 2, 3)$, and Eq. 34, for any k, $V_k(0, \hat{\omega}_{i,k}(T), \hat{S}_{i,k}(T), \hat{P}_{1,k}(T), \hat{P}_{2,k}(T))$ is bounded and $V_k(0, \hat{\omega}_{i,k}(0), \hat{S}_{i,k}(0), \hat{P}_{1,k}(0), \hat{P}_{2,k}(0)) = V_{k-1}(0, \hat{\omega}_{i,k-1}(T), \hat{S}_{i,k-1}(T), \hat{P}_{1,k-1}(T), \hat{P}_{2,k-1}(T))$ is bounded. It can be seen that for any k, $V_k(t)$, $\hat{\omega}_{i,k}(T)$, $\hat{S}_{i,k}(T)$, $\hat{P}_{1,k}(T)$, and $\hat{P}_{2,k}(T)$ are bounded. Therefore, u_k and $\dot{z}_{i,k}\,(i = 1, 2, 3)$ are bounded, $z_{i,k}$ is consistent and continuous, so $\lim_{k\to\infty} z_{i,k}(t) = 0$, $(i = 1, 2, 3)$.

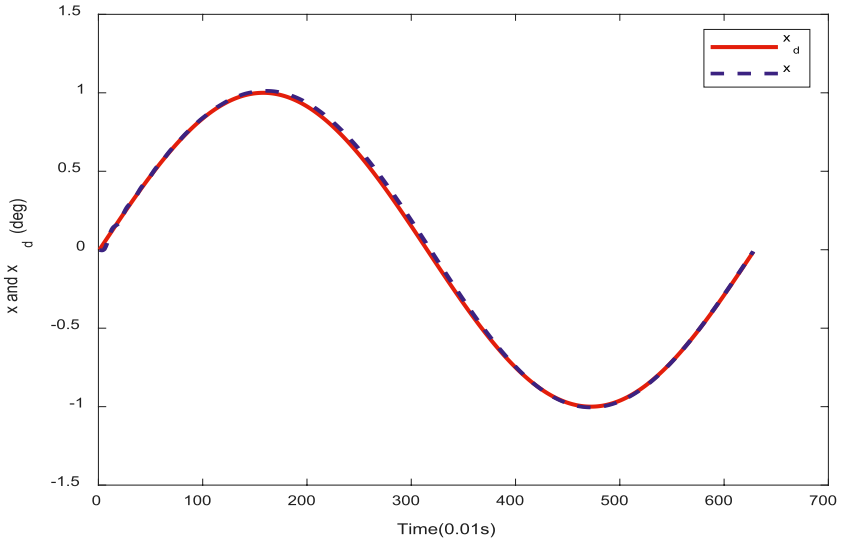


FIGURE 9
Trajectory trace graph for iteration 400.

4 Simulation analysis

According to the control model established in the design part of Section 1,

$$\begin{cases} \dot{x}_{1,k} = a_1 x_{2,k} + W_{1,k}(x_{1,k}, t), \\ \dot{x}_{2,k} = x_{3,k} + W_{2,k}(x_{1,k}, x_{2,k}, t), \\ \dot{x}_{3,k} = a_3 u_k + W_{3,k}(x_{2,k}, x_{3,k}, t), \end{cases} \tag{38}$$

where $W_{1,k} = f_{1,k}(x_{1,k}) + \Delta_{1,k}(x_k, t)$, $W_{2,k} = f_{2,k}(x_{1,k}, x_{2,k}) + \Delta_{2,k}(x_k, t)$, $W_{3,k} = f_{3,k}(x_{2,k}, x_{3,k}) + \Delta_{3,k}(x_k, t)$, $\Delta_{i,k}(x_k, t)$, $i = 1, 2, 3$ is the uncertain part, $|\Delta_{i,k}(x_k, t)| \le \rho_i$, and ρ_i is a positive real number and satisfies

$$\begin{cases} f_{1,k}(x_{1,k}) = -\dfrac{g}{V_T}\cos x_{1,k} + \bar{L}_0, \\ f_{2,k}(x_{1,k}, x_{2,k}) = \dfrac{g}{V_T}\cos x_{1,k} - \bar{L}_0 - \bar{L}_\alpha x_{2,k}, \\ f_{3,k}(x_{2,k}, x_{3,k}) = M_\alpha x_{2,k} + M_q x_{3,k}, \end{cases} \tag{39}$$

where $a_1 = \bar{L}_\alpha > 0, a_3 = M_\delta > 0$.

The ideal trajectory $x_{1d} = \sin t$ is given corresponding to the example hypothesis. Take $\Delta_{1,k} = 0.01 \sin 2t$, $\Delta_{2,k} = 0.1 \cos 2t$, and $\Delta_{3,k} = 0.05 \sin t \cos 2t$.

The unknown physical parameters are selected as follows: $\bar{L}_0 = -0.1, \bar{L}_\alpha = 0.74, M_\alpha = 0.1, M_q = -0.02, and\ M_\delta = 1.36$.

Assume the stable speed $V_T = 200m/s$ and $g = 9.8m/s^2$, where the initial state of the model takes $x(0) = [0\ 0\ 0]^T$.

From the controller design part in Section 2, it can be seen that the functions that need to be approximated by the RBF neural network are $\frac{W_1}{\alpha_1}$, W_2, and $\frac{W_3}{\alpha_3}$. Nine hidden nodes are selected, the center of the Gaussian basis function is evenly distributed in the range of $[-1, 1]$, and the width is 5.3, then the initial values of the network weights are set as

$$W_{10} = [11.6\ 0.1\ 0.1\ 0.1\ 0.1\ 0.01\ 0.01\ 0.01\ 0.01]^T,$$
$$W_{20} = [1\ 0.001\ 0.01\ 0.01\ 0.001\ 0.01\ 0.01\ 0.01\ 0.001]^T,$$
$$W_{30} = [30\ 0.01\ 0.1\ 0.01\ 0.01\ 0.01\ 0.00\ 0.01\ 0.009]^T.$$

Let all the initial values $x_{1,0}(0)$, $x_{2,0}(0)$, and $x_{3,0}(0)$ be zero. All the initial values $S_{1,0}(0), S_{2,0}(0), S_{3,0}(0)$ are 0.1, and the initial values $P_{1,0}(0)$ and $P_{2,0}(0)$ are both 0.01. All the initial errors are zero, $\rho_1 = 0.01$, $\rho_2 = 0.1$, $\rho_3 = 0.05$, $a_{1M} = 0.74$, and $a_{3M} = 0.36$.

The control parameters are selected under the condition that Lyapunov stability is satisfied. $c_1 = 23, c_2 = 3, c_3 = 40, \Gamma_1 = diag\{42\ 0.1\ 0.1\ 2\ 2\ 0.1\ 0.1\ 0.1\ 0.1\}, \Gamma_2 = 70, \Gamma_3 = 0.1,$

$$\Gamma_4 = diag\{37\ 0.1\ 0.1\ 0.1\ 0.1\ 0.1\ 0.1\ 0.1\ 0.1\}, \Gamma_5 = 1,$$

$$\Gamma_6 = diag\{40\ 0.1\ 0.1\ 0.1\ 0.1\ 0.1\ 0.1\ 0.1\ 0.1\}, \Gamma_7 = 1000, \Gamma_8 = 0.1.$$

The simulation is carried out under the actual control law $u_k = -\hat{\omega}_{3,k}^T \xi_3 - \frac{1}{\Delta_k} z_{3,k} \hat{S}_{3,k} - c_3 z_{3,k} + \hat{P}_2 \dot{\alpha}_{2,k} - z_{2,k}$ and the adaptive law $\dot{\hat{\omega}}_{1,k} = \Gamma_1 \xi_1 z_{1,k}$, $\dot{\hat{S}}_{1,k} = \Gamma_2 \frac{1}{\Delta_k} z_{1,k}^2$, $\dot{\hat{P}}_{1,k} = \Gamma_3 \dot{y}_r z_{1,k}$, $\dot{\hat{\omega}}_{2,k} = \Gamma_4 \xi_2 z_{2,k}$, $\dot{\hat{S}}_{2,k} = \Gamma_5 \frac{1}{\Delta_k} z_{2,k}^2$, $\dot{\hat{\omega}}_{3,k} = \Gamma_6 \xi_3 z_{3,k}$, $\dot{\hat{S}}_{3,k} = \Gamma_7 \frac{1}{\Delta_k} z_{3,k}^2$, $\dot{\hat{P}}_{2,k} = \Gamma_8 \dot{\alpha}_{2,k} z_{3,k}$, and the given initial state. The simulation results are as follows.

From Figure 2, the inclination error of the aircraft trajectory can basically tend to zero with the increase in the number of iterations. The simulation results from Figures 3–6 show the boundness of the designed

parameters. Figure 7 shows that as the number of iterations increases, the control input can remain unchanged, and the stable state of the controller and the control effect are good. By comparing the tracking effect in Figures 8, 9, the tracking effect is more significant with the increase in iteration times.

In summary, the designed neural network adaptive iterative learning controller is suitable for a tracking control in a limited time period and the RBF neural network has a very good effect on approximating any unknown parameters.

5 Conclusion

This paper proposed a new adaptive iterative learning control method for the flight path of the aircraft to complete the tracking control problem in the finite time interval based on the RBF neural network. According to the feedback system in the form of strict feedback abstracted by the longitudinal model of the aircraft, the method of an inversion design is adopted, and the virtual control law is designed to control each subsystem, and finally, the actual control law is obtained by inversion. For each subsystem, a neural network is used to approximate the unknown function in the control, which can greatly improve the control performance of the uncertain system. According to the Lyapunov stability function set by each subsystem, the adaptive law of the neural network that meets the constraints is derived. According to the error between the system output and the ideal trajectory, the adaptive weights and the adjustment parameters are updated to make the entire closed-loop system tend to convergence and stability, and the control objectives of system stability and all signals in a bounded area are achieved. Finally, the effectiveness and feasibility of applying the controller designed by the neural network adaptive iterative learning control method to the aircraft track system are verified by the example simulation.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material; further inquiries can be directed to the corresponding author.

Author contributions

CZ and XT contributed to the conception and design of the study. XT performed the statistical analysis and wrote the first draft of the manuscript. LY wrote sections of the manuscript. All authors contributed to manuscript revision, read, and approved the submitted version.

Funding

This work was supported by the National Natural Science Foundation (NNSF) of China under grants 61603296 and 62073259. This work was also supported by the Key Laboratory of Complex System Control and Intelligent Information Processing in Shaanxi Province.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Zheng C, Yan P, Ding M. Research status and trend of aircraft track planning in Chinese[J]. *J Astronautics* (2007) 2007(06) 1441–6. doi:10.3321/j.issn:1000-1328.2007.06.001

2. Zhuang H, Sun Q, Chen Z, Zeng X. Robust adaptive sliding mode attitude control for aircraft systems based on back-stepping method. *Aerospace Sci Tech* (2021) 118:107069. doi:10.1016/j.ast.2021.107069

3. Yan K, Wu Q. Adaptive tracking flight control for unmanned autonomous helicopter with full state constraints and actuator faults. *ISA Trans* (2021) 128: 32–46. doi:10.1016/j.isatra.2021.11.012

4. Wang Y, Hu J. Robust control for a quadrotor aircraft with small overshoot and high-precision position tracking performance. *J Franklin Inst* (2020) 357(18): 13386–409. doi:10.1016/j.jfranklin.2020.09.033

5. Yao L. Disturbance observer-based backstepping control for hypersonic flight vehicles without use of measured flight path angle. *Chin J Aeronautics* (2021) 34(2): 396–406. doi:10.1016/j.cja.2020.09.053

6. Yang L, Dong C, Zhang W, Wang Q. Phase plane design based fast altitude tracking control for hypersonic flight vehicle with angle of attack constraint. *Chin J Aeronautics* (2021) 34(2):490–503. doi:10.1016/j.cja.2020.04.026

7. Yue F, Wang Y, Sun Z, Xi B, Wu L. Robust modification of nonlinear L1 adaptive flight control system via noise attenuation. *Aerospace Sci Tech* (2021) 117:106938. doi:10.1016/j.ast.2021.106938

8. Yue H, Gong C. Adaptive tracking control for a class of stochastic nonlinearly parameterized systems with time-varying input delay using fuzzy logic systems. *J Low Frequency Noise, Vibration Active Control* (2022) 41(3):1192–213. doi:10.1177/14613484211045761

9. Yue H, Yang W, Li S. Fuzzy adaptive tracking control for a class of nonlinearly parameterized systems with unknown control directions. *Iranian J Fuzzy Syst* (2019) 16(5):97–112. doi:10.22111/IJFS.2019.4909

10. Yu Q, Hou Z. Adaptive fuzzy iterative learning control for high-speed trains with both randomly varying operation lengths and system constraints.

IEEE Trans Fuzzy Syst (2021) 29(8):2408–18. doi:10.1109/tfuzz.2020.2999958

11. Zhou Z, Wang W, Zhang Y, Yan Q, Cai J. Barrier adaptive iterative learning control for tank gun control systems under nonzero initial error condition. *IEEE Access* (2022) 10:8664–72. doi:10.1109/access.2022.3144326

12. Fei Y, Kong X, Mokbel AAM. Complex dynamics, hardware implementation and image encryption applica tion of multiscroll memeristive hopfield neural network with a novel local active memeristor[J]. *IEEE Trans Circuits Systems–II* (2022). Express Briefs 1–1. doi:10.1109/TCSII.2022.3218468

13. Shen H, Fei Y, Wang C, Sun J. Firing mechanism based on single memristive neuron and double m emristive coupled neurons[J]. *Nonlinear Dyn* (2022) 110:3807–3822. doi:10.1007/s11071-022-07812-w

14. Lin H, Wang C, Sun Y, Ting W. Generating n-scroll chaotic attractors from a memristor-based magne tized hopfield neural network[J]. *IEEE Trans Circuits Systems–II* (2022). Express Briefs 1–1. doi:10.1109/TCSII.2022.3212394

15. Yu F, Shen H, Yu Q, Kong X, Sharma PK, Cai S. Privacy protection of medical data based on multi-scroll memristive hopfield neural network. *IEEE Trans Netw Sci Eng* (2022) 2022:1–14. doi:10.1109/TNSE.2022.3223930

16. Han H, Zhang L, Hou Y, Qiao JF. Nonlinear model predictive control based on a self-organizing recurrent neural network. *IEEE Trans Neural Netw Learn Syst* (2016) 27(2):402–15. doi:10.1109/tnnls.2015.2465174

17. Han H, Wu X, Zhang L, Tian Y, Qiao J. Self-organizing RBF neural network using an adaptive gradient multiobjective particle swarm optimization. *IEEE Trans Cybern* (2019) 49(1):69–82. doi:10.1109/tcyb.2017.2764744

18. Jin X, He T, Wu X, Wang H, Chi J. Robust adaptive neural network-based compensation control of a class of quadrotor aircrafts. *J Franklin Inst* (2020) 357(17):12241–63. doi:10.1016/j.jfranklin.2020.09.009

19. Song J, Yan M, Yang P. Neural adaptive dynamic surface asymptotic tracking control for a class of uncertain nonlinear system. *Circuits Syst Signal Process* (2020) 40:1673–98. doi:10.1007/s00034-020-01558-9

20. Zhang C, Tian X. Non-uniform trajectory tracking adaptive iterative learning control for nonlinear pure-feedback systems with initial state error based on RBF-neural network[C]. In: 2021 40th Chinese Control Conference (CCC); July 26-28, 2021; Shanghai, China (2021). p. 532–9.

21. Pang Z, Wang T, Liu S. An iterative learning algorithm based on RBF neural network in upper limb rehabilitation robot[J]. In: IEEE 10th Data Driven Control and Learning Systems Conference (DDCLS); May 14-16, 2021; China (2021). p. 293–8.

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 22 December 2022
DOI 10.3389/fphy.2022.1102892

OPEN ACCESS

EDITED BY
Chunbiao Li,
Nanjing University of Information Science and Technology, China

REVIEWED BY
Qiang Lai,
East China Jiaotong University, China
Yuexi Peng,
Xiangtan University, China

*CORRESPONDENCE
Yaling Chen,
chenyaling0307@163.com

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 19 November 2022
ACCEPTED 05 December 2022
PUBLISHED 22 December 2022

CITATION
Wang J, Xia Z, Chen Y, Hu C and Yu F (2022), Intrusion detection framework based on homomorphic encryption in AMI network.
Front. Phys. 10:1102892.
doi: 10.3389/fphy.2022.1102892

Intrusion detection framework based on homomorphic encryption in AMI network

Jing Wang[1], Zhuoqun Xia[1], Yaling Chen[2,1]*, Chang Hu[1] and Fei Yu[1]

[1]School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha, China, [2]School of Electrical and Information Engineering, Hunan Institute of Traffic Engineering, Hengyang, China

In order to alleviate the privacy issue of traditional smart grids, some researchers have proposed a power metering system based on a federated learning framework, which jointly trains the model by exchanging gradients between multiple data owners instead of raw data. However, recent research shows that the federated learning framework still has privacy and security issues. Secondly, since the server does not have direct access to all parties data sets and training process, malicious attackers may conduct poisoning attacks. This is a new security threat in federated learning - poisoning attack. However, solving the two problems at the same time seems to be contradictory because privacy protection requires the inseparability of the training gradients of all parties, and security requires the server to be able to identify the poisoned client. To solve the above issues, this paper proposes an intrusion detection method based on federated learning client-side security in AMI networks, which uses CKKS to protect model parameters. In addition, to resist the poisoning attack in federated learning, the model trained by the data processing center and the model trained by each client are firstly calculated for the direction similarity, and the similarity value is scaled as the adaptive weight of the aggregation model. Then, the size of each client model update is normalized to be the same size as the data processing center model update. Finally, the normalized updates and adaptive weights are weighted averaged to form a global model update. The research results show that the method in this paper can effectively resist inference attacks and poisoning attacks. In the AMI network, the intrusion detection method based on federated learning can maintain a good detection performance.

KEYWORDS

advanced metering infrastructure, intrusion detection, federated learning, homomorphic encryption, privacy protection, poisoning attack

1 Introduction

Because smart grid Advanced Metering Infrastructure (AMI) is an important part of the power system, and the power system is also related to information security and computer network, it is vulnerable to various network threats, and advanced technologies need to be adopted to protect the security of smart grid AMI. With the gradual development of machine learning technology, machine learning is widely used in the field of smart grid, such as intrusion detection Li et al. [1], electricity stealing detection Hasan et al. [2], private data sharing Su et al. [3], and so on. However, despite the rapid development of machine learning-based power systems, their privacy and security problems are still exposed. Federated learning is proposed as a promising privacy-preserving method, which can jointly learn a common machine learning model in a decentralized training manner under the coordination of a central server. During this training process, each participant uses the local data set to learn the model parameters, and the local data does not need to be uploaded to the data center, so it will not be exposed. All in all, federated learning can protect the privacy of training data and detection data to a certain extent. However, recent work shows that federated learning still confront many privacy and security issues.

From a privacy perspective, direct communication of gradients may reveal sensitive information. Whether it is a semi-honest server or a third party, once the gradient information is obtained, a large amount of sensitive information about the client may be obtained through inference, thereby revealing the privacy of the client. From a security point of view, if there are malicious actors in the federated training process, they can perform poisoning attacks on data or models, send poisonous local model updates to the server, or insert hidden backdoors into the global model. So in order to build a privacy and security federated learning framework, these two problems must be settled.

In order to deal with the privacy attacks faced by federated learning, researchers have proposed many solutions, which are mainly based on the following technologies: methods such as differential privacy, secure multi-party computation, and homomorphic encryption. For federated learning poisoning attack defense, the main idea is to eliminate malicious models before local model parameters are aggregated. Then the corresponding defense methods are as follows: similarity-based defense, statistics-based defense, performance-based defense, and feature extraction-based defense. Both the privacy and poisoning attacks of federated learning are solved by constructing linear or non-linear functions. The machine learning model is composed of linear and non-linear models. For example, the neural network (Neural Network,NN) used in the machine learning model is a mathematical model that simulates the structure and function of biological networks and is composed of neurons and synapses. Among them, neurons have complex non-linear characteristics and are the basic unit of biological information processing nervous system [4–6].

In order to solve the privacy and poisoning attack security problems of smart grid AMI federated intrusion detection concurrently, this paper proposes a privacy protection method using homomorphic CKKS encryption. When the client upload gradient is not obtained by the untrustworthy, we calculate the directional similarity between the client training model and the data center training model, and then extract the similarity by using a non-linear function (logarithmic function). Specifically, this paper makes three contributions:

1) This paper adopts a privacy-enhanced federated learning framework based on CKKS in the smart grid AMI network, which can prevent malicious clients from inferring client information through their uploaded gradients (malicious clients may be sham clients injected by attackers or real clients compromised by attackers), thereby revealing client privacy. It also prevents semi-honest data centers and control servers from violating client privacy.
2) This paper improves a poisoning attack defense method. First, use the Pearson correlation coefficient to calculate the directional similarity between the model trained by each client and the model trained by the data center, and then scale the similarity value. The data center model is obtained by training on a clean dataset. Second, the gradients are normalized to avoid malicious attackers from affecting the global model by enlarging the gradient size. Finally, the scaled similarity is used as the adaptive weight to calculate the aggregation model.
3) This paper provides algorithm, security analysis and tests on the dataset to prove that the proposed scheme can resist the Trim attack of federated learning.

The rest of this article is organized as follows. In Section 2, we analyze related work. In Section 3, the relevant knowledge is needed to supplement the research content. In Section 4, we make a problem statement. Section 5 then details a privacy-enhanced robust federated learning framework. Next, analysis and performance evaluation are performed in Section 6. Finally, Section 7 concludes the paper.

2 Related work

2.1 Research status of federated learning privacy issues

For user privacy in federated learning, the existing work mainly focuses on guaranteeing the secrecy of gradients. These solutions are mainly based on the following techniques: Differential Privacy [7–9], Secure Multi-Party Computation [10–12], Homomorphic Encryption [13–15] and other methods.

Wu et al. [7] proposed a new Local Differential Private (LDP) algorithm that redesigned the training process. This enables data owners to add layers of randomization before the data leaves and reaches potentially untrusted machine learning services. Kumar et al. [8] proposed Privacy-aware and asynchronous Deep-Learning-assisted IoT

applications (PADL) to enable several data collection sites to cooperatively train deep neural networks (Deep Neural Networks). Neural Networks, DNN) while maintaining the secrecy of private data. Liu et al. [9] proposed an adaptive privacy-preserving federated learning framework APFL. These differential privacy-based methods may result in lower federated learning accuracy.

Bonawitz et al. [10] proposed a novel, communication-efficient, fault-robust protocol for secure aggregation of high-dimensional data. Mohassel and Zhang [11] proposed SecureML, which conducts privacy-preserving learning through Security Multi-Party Computation (SMC). During the initial setup phase, data owners operate, encrypt, and secretly share their data between two non-colluding servers. Riazi et al. [12] pointed out an SMC-based protocol that enables multiple participants to collaboratively compute an agreed-upon function without revealing any participant's input information except that which can be inferred from the computation results. As a result, secure multi-party computation will not only lead to large communication overhead, but also cannot fully guarantee the prevention of information leakage.

Aono et al. [13] conducted model training through asynchronous SGD, and then homomorphically encrypted the gradient, which could obtain high accuracy while protecting the security of federated learning. Xu et al. [14] designed a new solution to decrease the negative influence of irregular users on training accuracy, thereby guaranteeing that the training results are mainly computed from the contribution of high-quality data, while using chaotic circuits and additional homomorphic ciphers system to ensure the confidentiality of all user-related information. Fang and Qian [15] encrypted the uploaded gradients using an improved paillier algorithm, which is almost the same as the training accuracy of the multi-party privacy-preserving machine learning framework.

According to the analysis, this paper proposes a homomorphic CKKS encryption method to protect the privacy of the gradient uploaded by the client.

2.2 Research status of federated learning poisoning attack

In order to solve the federated learning poisoning attack, researchers have proposed some defense schemes, which are divided into four types: similarity-based defense, statistics-based defense, performance-based defense and feature extraction-based defense.

Similarity-based defense is to design aggregation algorithms based on the similarity between gradients, such as Krum Blanchard et al. [16] for Euclidean distance, which selects one gradient with the lowest predefined score as the aggregated gradient from all uploaded gradients. Xia et al. [17] iteratively filter the models that are farthest from the mean of the remaining models, taking the mean of the last remaining models as the aggregated model. These methods are more effective when there are few malicious attackers. However, when there are many attackers colluding with each other, they will lead to choosing a malicious value as the comparison standard, then the similarity calculation value lacks reliability.

Statistics-based defense is to use statistical features such as median and mean to circumvent malicious model parameters. Yin et al. [18] first eliminated some extreme values and selected the median result of the corresponding dimension of the client model as the aggregated model. Chen et al. [19] used the gradient dimension median as a global update. These methods can bypass malicious models with high probability by selecting the median or mean as the aggregated result. However, the global model lacks a lot of normal gradient information, which weakens the accuracy of the model.

Fang et al. [20] argue that neither of the above two aggregation rules is efficacious enough for an opponent with certain knowledge, and they can carefully design a similar set of gradients to confuse the aggregation rules.

Performance-based defense is to use auxiliary datasets to test the accuracy, loss value, *etc.* Of each client model to eliminate malicious models. Xie et al. [21] used the validation dataset to directly test the loss value and modulo length of the model, and calculate the score of each model to remove outliers. Zhao et al. [22] assign each client's submodel to other clients for cross-validation to find toxic updates. These defenses are to use a new dataset on the server to test the client's model for anomalies.

Feature extraction-based defenses can extract features from high-dimensional models and then discriminate feature spaces. Zhao et al. [22] directly decomposed the uploaded model by Principal Components Analysis (PCA), and then considered the model corresponding to the principal component to be the normal model. Nguyen et al. [23] introduced the FLAME defense framework for estimating a sufficient amount of noise to inject to ensure backdoor elimination. To minimize the amount of noise required, model clustering and weight clipping methods are also used to maintain good performance of the aggregated model. A problem with these methods is that which type represents normal and which type represents abnormal, and a benchmark needs to be determined in advance, which is a technical problem.

3 Preliminaries

3.1 Homomorphic encryption

Homomorphic encryption is an encryption scheme that can still perform homomorphic computations in an encrypted situation. Under the same operation rules, the result of its ciphertext calculation is the same as the ciphertext of the plaintext calculation result. According to different operation rules, homomorphic encryption can be divided into two types: one is additive homomorphism and the other is multiplicative homomorphism. According to the supported operation types and operation times, homomorphic encryption schemes can be divided into three types: Partially Homomorphic Encryption (PHE), somewhat Homomorphic Encryption (SHE), and Fully

Homomorphic Encryption (FHE). Some of the homomorphic encryption methods can only perform one type of homomorphic calculation, addition or multiplication. Somewhat homomorphism supports addition and multiplication operations on ciphertext at the same time, but can only perform ciphertext operations for a limited number of times. While the homomorphic encryption method can simultaneously perform addition and multiplication homomorphic calculations, and can support infinite ciphertext operations. CKKS Cheon et al. [24] is a fully homomorphic encryption method, which supports the addition and multiplication of floating-point vectors in the ciphertext space. Homomorphic encryption CKKS consists of key generation (KeyGen), key distribution (DisKey), encryption algorithm (Enc), and decryption algorithm (Dec).

- **KeyGen**: Responsible for the generation of the public key (*PK*) and private key (*SK*) of the entire system.
- **DisKey**: Responsible for distributing keys.
- **Encryption**: Indicates the encryption function, which uses the public key to encrypt the plaintext to attain the ciphertext.
- **Decrypt**: Indicates the decryption function, which uses the private key to decrypt the ciphertext to obtain the corresponding plaintext.

The homomorphic encryption CKKS used satisfies the additive homomorphic property and the multiplicative homomorphic property as shown in Equations 1, 2.

$$E(x_1) + E(x_2) = E(x_1 + x_2) \tag{1}$$

$$E(x_1)*r = E(r*x_1) \tag{2}$$

3.2 Pearson correlation coefficient

The Pearson correlation coefficient is a method of measuring the similarity between vectors, which measures whether the vectors are linearly related. In addition, it can be regarded as the cosine similarity after centering the vectors. The disadvantage of cosine similarity is that the two vectors calculated cannot be empty in each dimension, and the Pearson correlation coefficient overcomes this defect.

When there are two vectors $X = [x_1, x_2, \ldots, x_n]$, $Y = [y_1, y_2, \ldots, y_n]$, the similarity calculation between vector X and vector Y is shown in formula (3).

$$\rho(X,Y) = \frac{Cov(X,Y)}{\sigma(X)*\sigma(Y)} = \frac{\sum_{i=1}^{n}(x_i - \bar{X})*\sum_{i=1}^{n}(y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{X})^2}*\sqrt{\sum_{i=1}^{n}(y_i - \bar{Y})^2}} \tag{3}$$

The Pearson correlation coefficient value $\rho(X, Y)$ is between -1 and 1. Different values of the Pearson correlation coefficient indicate that the two vectors have different relationships, and different value ranges represent different correlation strengths of the vectors.

When the Pearson correlation coefficient value is 0, it means that there is no linear correlation between the two vectors.

When the value of the Pearson correlation coefficient is between (-1, 0), it means that when the value of one vector increases (decreases), the value of the other vector decreases (increases), indicating an inverse relationship between them.

When the value of the Pearson correlation coefficient is between (0,1), it means that when the value of one vector increases (decreases), the value of the other vector increases (decreases), indicating the same change between them.

It is assumed that the larger the absolute value of the correlation coefficient of two vectors, the stronger their correlation. And the closer their correlation coefficient values are to 0, the weaker the correlation between the two vectors, as shown in Table1.

This paper uses the Pearson correlation coefficient to compute the correlation between the federated learning data center training model and the client-side training model. When the correlation coefficient value is less than 0.25, we consider that the model trained by the client has nothing to do with the model trained by the data center, and set the aggregation weight of the model trained by the client to 0, thereby suppressing the poisoning model. Here we have tested and set the Pearson correlation coefficient to 0.25, 0.5 equivalent. It is found that when the Pearson correlation coefficient is 0.25, the effect is better. The focus of this paper is to use CCKS and similarity based robust aggregation algorithm to build our intrusion detection framework and implement intrusion detection. Therefore, the setting process of Pearson correlation coefficient correlation value is not shown in the paper.

4 Problem statement

4.1 Threat model

When the intrusion detection method based on federated learning is deployed in the smart grid AMI network, there may still be security risks. For example, an attacker may want to obtain client information, and can intercept the model updates uploaded by each client to conduct inference attacks. Other attackers want to influence certain decisions of smart grid AMI by reducing global model performance. Among them, these clients may be sham clients injected by attackers, or real clients invaded by attackers. But the attacker will not compromise the data center and control server (too expensive), which means that the data center and control server can honestly do everything correctly and according to the regulations. Client privacy may also be compromised as data centers and control servers have access to model updates for individual clients. Therefore, we consider the data center and control server to be semi-honest, and furthermore, assume that the four entities do not collude with each other.

When the attacker has the following information about the FL system: training data, local updates, loss function and learning

TABLE 1 Correlation table.

Absolute value	0–0.2	0.2–0.4	0.4–0.6	0.6–0.8	0.8-1
Correlation	extremely weak	weak	medium	Strong	extremely Strong

rate on the client. When the attacker knows the training data of the client, he may carry out a data poisoning attack; When an attacker knows the local update, loss function and learning rate of the client, he has the ability to modify the local model, that is, to poison the model. We believe that the proposed method can defend against attacks such as Trim attacks initiated by attackers possessing the above information.

4.2 Client security objective

The goal of this paper is to design a scheme that achieves robustness against poisoning attacks launched by malicious clients while preserving privacy and without loss of accuracy. Specifically, we have three design goals:

- **Privacy**: An attacker can reveal privacy by inferring gradients or parameters to recover training samples. In order to protect the privacy of users, it is proposed to protect the gradient uploaded by the client.
- **Robustness**: refers to the presence of attacks, the proposed method can still maintain the classification accuracy of the global model. That is, regardless of whether there is an attack, the proposed method should have the same performance as FedAvg without an attack.
- **Accuracy**: Poisoning attacks cause model performance degradation by poisoning datas or models, such as inaccurate predictions, misclassifications, *etc.* Therefore, the federated learning intrusion detection method must guarantee that the accuracy is within a rational range.

4.3 System model of smart grid AMI

As shown in Figure 1, the system model of this AMI has four types of entities:

- **Key Generation Center (KGC)**: An independent and trusted third-party agency responsible for distributing and managing the public and private keys required by the AMI federation system.
- **Clients**: The data owner trains a unified model under the coordination of the data center. Based on security considerations, the client uses its own data for local training, and then uploads the cryptographic gradients or parameters to the data center, and we presume that the data of each client is independent and identically distributed.
- **Data Center (DC)**: Receives model updates from individual clients and aggregates them. DC has a small dataset of no attack samples, which provides a basis of trust for the model to resist poisoning attacks.
- **Control Server (CS)**: completes the aggregation work together with the DC, and has the public-private key pair (pk_c, sk_c) generated by the KGC for encryption and decryption. aaaaaaaaaaaaaaaa

For ease of reading, symbols and descriptions appearing in this paper are listed in Table 2.

5 Federated learning intrusion detection method based on homomorphic encryption

In this paper, the full connected neural network is used to train the detection model, and the trained model parameters or gradients are floating point, while homomorphic encryption CKKS can support the addition and multiplication of floating point vectors in the ciphertext space. Therefore, In order to settle the issue that the client model update of each client suffers from inference attacks and leaks user privacy, this paper uses the homomorphic encryption method (CKKS) to encrypt the data generated by each entity, so as to protect the client upload gradient from being obtained by untrusted people. A trusted key generation center generates keys for individual entities. All authorized entities can access the public key generated by KGC for other entities, and the private key generated by KGC for an entity is kept by that entity. For example, the key (pk_c, sk_c) generated by KGC for the control server, in which all authorized entities (each client, data center) can use the public key of the control server, and the private key is stored by CS.

This paper believes that there is a deviation between the malicious gradient vector and the benign gradient vector, and judges whether the gradient is malicious according to the similarity with the benign gradient, so that the influence of the malicious gradient can be identified and reduced. This article is based on such a foundation. In order to train an intrusion detection model that can resist poisoning attacks, this paper also considers the model updates of the DC and each client to jointly build a global model. Specifically, in each iteration, the data center trains a benchmark model Wang et al. [25] based on its data set, and performs

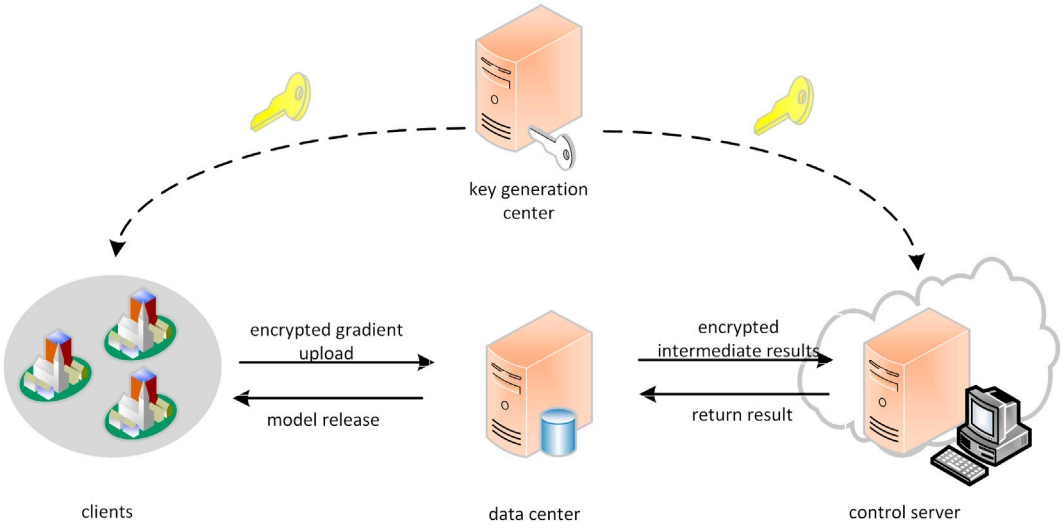


FIGURE 1
Smart grid AMI system model.

TABLE 2 Symbol definition.

Symbol	Describe
Pk	public key
Sk	private key
ω	model weights
G	Gradient
α	learning rate
V	magnified person correlation coefficient
T	Federated Learning Iterations
t	Federated Learning Iteration Index
K	number of clients
A	Number of malicious clients

directional similarity calculation with the model update of each client, and then scales the calculated value as a model Aggregated adaptive weights. The closer the directions are, the greater the adaptive weight. This protects against attacks where an attacker manipulates the direction of the client's model. At the same time, the model update amplitude of each client is normalized according to the data center model update amplitude, so that it has the same size as the data center model update, so that it can resist the scaling attack of the client model by the attacker. The entire federated learning model building process repeats three steps.

5.1 Client local training

In round t, the client U_x, $x \in [1, 2, \ldots, K]$ participating in the training obtains the global model parameters $[\omega_x^{t-1}]_{pk_x}$ issued by the DC, wherein the global model parameters are encrypted with pk_x. After decryption by the private key sk_x, use the client's data for training to update the model, the expression is $\omega_x^t = \omega_x^{t-1} - \alpha \nabla G_X^{t-1}$. In order to prevent attackers or semi-honest data centers from obtaining updates of each client and compromising user privacy, the gradient $[\omega_x^t]_{pk_c}$ is obtained by encrypting the local update with the public key pk_c of the control server.

5.2 Secure aggregation

The DC interacts with the CS to identify and defend against user-initiated poisoning attacks. Specifically, since the data saved by DC is safe, the gradient $[G_d^t]_{pk_c}$ obtained by its training model is used as a comparison benchmark.

Since an attacker may manipulate the direction of model updates on malicious clients, so that the global model is updated in the opposite direction to the update direction, we use the Person correlation coefficient to calculate the encryption gradient $[G_x^t]_{pk_c}{}_{x=1}^{x=m}$ of the selected m clients in a similar direction to the baseline encryption gradient, as shown in Algorithm 1. To compute the similarity without exposing client privacy, DC blurs the gradients of client and DC by formula (4) and formula (5):

$$E_x = [G_x^t]_{pk_c} * r_x, x \in [1, 2, \ldots, m] \quad (4)$$

$$E_d = [G_d^t]_{pk_c} * r_d \tag{5}$$

where r_x and r_d are two randomly selected non-zero numbers.

Input: DC owns $[G_x^t]_{pk_c}$ and $[G_d^t]_{pk_c}, x \in [1, 2, \ldots, m]$, CS owns private key sk_c
Output: similarity between $[G_x^t]_{pk_c}, x \in [1, 2, \ldots, m]$ and $[G_d^t]_{pk_c}$
DC:
1: Randomly select two non-zero numbers $r_i, i \in \{x, d\}$
2: Using Equation 4 and 5 to get: $E_i = [G_i^t]_{pk_c} * r_i, i \in \{x, d\}$
3: Send E_i to CS
CS:
1: Decrypt E_i using its private key sk_c: $D_i = Dec(sk_c, E_i), i \in \{x, d\}$
2: Calculate the similarity $\rho_{x,d}$between client parameters and data center parameters using formula (6)

Algorithm 1. Similarity Calculation

The DC then sends the obfuscated data to the CS, which decrypts using its private key to obtain D_x and D_d. CS uses the Pearson correlation coefficient to calculate the update direction similarity of the decrypted client model gradient and data center model gradient, as shown in formula (6):

$$\rho_{x,d} = \frac{Cov(D_x, D_d)}{\sigma(D_x) * \sigma(D_d)}, x \in [1, 2, \ldots, m] \tag{6}$$

Among them, the decrypted data and are still obscure.

After the DC performs the similarity calculation, the Person correlation coefficient is scaled by Eq. 7.

$$v_x = max\left\{0, In\left(\frac{1 + \rho_{x,d}}{1 - \rho_{x,d}}\right) - 0.5\right\} \tag{7}$$

For those with low similarity, we consider them to be abnormal. Eq. 7 indicates that the higher the correlation is, the more information is extracted by the scaling function, and the lower the correlation is, the less information is extracted by the scaling function. For below a certain value, the aggregation weight v_x is 0.

Since it is possible for an attacker to launch an attack by expanding the scale of client model updates Wang et al. [25], thus affecting the global model update scale, this paper normalizes the magnitude of model updates for each client. The specific operation is to normalize the gradient obtained by each client training model to the same magnitude as the gradient of the DC training model, as shown in Eq. 8.

$$\bar{G}_i = \frac{\|G_d\|}{\|G_i\|} * G_i \tag{8}$$

where G_i represents the gradient of the ith client in the current training round, $\overline{G_i}$ represents the normalized gradient of the ith client, G_d represents the gradient obtained by data center training, and $\|.\|$ represents the L2 norm of a vector. Normalization guaranteess that updates to individual client models do not have too much influence on the aggregated global model. At the same time, the normalization in this paper also enlarges the smaller value of the update of the client to make it the same as the update value of the data center, because we think that the smaller update is more likely to be a normal model. Such normalization can mitigate the influence of malicious models on the global model.

Considering that the higher the similarity, the closer the model parameters are to the model trained by the data center, and the greater the impact on the final model. In this paper, the scaled similarity value is used as the weight of model aggregation, and the parameters of the global model are shown in formula (9).

$$\omega^t = \omega^{t-1} - \alpha \sum_{i \in [1,2,\ldots,m]} \frac{v_i}{\sum_{i \in [1,2,\ldots,m]} v_i} \overline{G_i} \tag{9}$$

where α is the global learning rate.

Algorithm 2 represents safe aggregation based on similarity and normalized gradient magnitude, which enables safe aggregation in the case that neither DC nor CS knows the true gradient of the client and CS does not know the true gradient of the data center.

Input: DC owns $[\omega^{t-1}]_{pk_c}$, $[G_x]_{pk_c}, x \in [1, 2, \ldots, m]$ and $[G_d]_{pk_c}$, CS owns private key sk_c and v_x, $x \in [1, 2, \ldots, m]$
Output: $[G^t]_{pk_c}$
DC:
1: randomly select m non-zero numbers r_x, $x \in [1, 2, \ldots, m]$ and one non-zero positive number r_d
2: Calculate $E_x' = [G_x^t]_{pk_c} * r_x, x \in [1, 2, \ldots, m]$ and $E_d' = [G_d^t]_{pk_c} * r_d$
3: Send E_i' and E_d' to CS
CS:
1: Decrypt E_i' using its private key sk_c: $D_i = Dec(sk_c, E_i'), i \in \{x, d\}$
2: Use Eq. 7 to obtain v_x
3: Compute the client-side model normalization using Eq. 8 to obtain $r_d * \overline{G_x} = \frac{\|D_d\|}{\|D_x\|} * D_x$
4: Calculate $e_x = \alpha \frac{v_x}{\sum_{i \in [1,2,\ldots,m]} v_x} * r_d * \overline{G_x}, x \in [1, 2, \ldots, m]$
5: Use CS's public key pk_c for encryption: $E_x = [e_x]_{pk_c}, x \in [1, 2, \ldots, m]$
6: send $\{E_{e_x}\}_{x=1}^{x=m}$ to DC
DC:
1: Remove noise: $F_x = E_{e_x} * \frac{1}{r_d}, x \in [1, 2, \ldots, m]$
2: Update the global model to: $[\omega^t]_{pk_c} = [\omega^{t-1}]_{pk_c} - (\sum_{i \in [1,2,\ldots,m]} F_x)$

Algorithm 2. Safe Aggregation Based on Similarity and Normalized Gradients

5.3 Model distribution

Since the global model parameters after security aggregation are encrypted with the public key of CS, if they are sent directly to the client, the client cannot decrypt it without the private key [22]. Therefore, it is necessary to communicate between the DC and CS, obtain the global model parameter $[\omega^t]_{pk_x}$ re-encrypted by the public key held by all users, and then broadcast the parameter to all clients. The global model security broadcast Algorithm 3 is as follows:

Input: DC owns $[\omega^t]_{pk_c}$, CS owns private key sk_c
Output: Client public key pk_x encrypted $[\omega^t]_{pk_x}$
DC:
1: Randomly choose n non-zero numbers r
2: Calculate $R = [\omega^t]_{pk_c} + [r]_{pk_c}$
3: Send R to CS
CS:
1: Decrypt R using its private key sk_c: $d = Dec(sk_c, R)$
2: Re-encrypt with public key pk_x: $R' = [d]_{pk_x}$
3: Send R' to DC
DC:
1: Remove noise: $[\omega^t]_{pk_x} = R' + [-r]_{pk_x}$
2: broadcast $[\omega^t]_{pk_x}$ to all clients

Algorithm 3. Global Model Safe Broadcast Algorithm

6 Analysis and performance evaluation

This section analyzes the algorithm, security and experimental results of the proposed method.

6.1 Analysis of algorithms

1) **Optimization problem**: optimize the algorithm by reducing the number of communications. For Algorithm 1, when calculating the correlation coefficient between the gradient of the data center and the gradient of the client, there are m clients training the model, then call Algorithm 1 m times, and the model parameters trained by the data center will be sent each time, so it can be sent by sending 1-time data center model parameters to optimize. Furthermore, we found that E_i and E_i' in Algorithm 1 and Algorithm 2 are independent of each other and blinded in the same way, where E_d' needs to be blinded with a positive number, so one blinding can be performed in DC (using Algorithm 2 blinded). And reduce the number of communication rounds by sending E_I' to CS in one round.
2) **Correctness of the security aggregation algorithm**: In order to ensure that the proposed scheme can effectively discern malicious gradients, we need to ensure that the data with blinding factors added in Algorithm 1 can be calculated correctly. It can be seen from the literature Cheon et al. [24] that the Pearson correlation coefficient can calculate the blinded variables in the right direction.

6.2 Security analysis

1) This paper discusses the security of three subjects: raw data, model parameters, and Pearson's correlation coefficient. For the original data, in the framework based on federated learning in this paper, the security of the raw data has been greatly guaranteed. The data analysis platform in the AMI network is transferred from the data center to the concentrator, and the small distance transmission of the original data from the smart meter to the concentrator reduces the risk of long-distance movement from the smart meter to the data center, so the privacy of data has been greatly protected. For model parameters, the model parameters trained using the CKKS encryption concentrator are then uploaded to the data center for aggregation. In order to prevent the semi-honest data center from leaking client privacy, let the control server and the data center aggregate model parameters together, so that neither the data center nor the control server can know the original data of the model parameters. This paper believes that it is unnecessary to protect the privacy of the Pearson correlation coefficient. Although the control server can directly obtain the Person correlation coefficient, since the control server is semi-honest, it will perform operations correctly as required, so it will not change the correlation coefficient value. Although there is literature showing that the similarity between gradients contains more information about the training data than the gradient values, samples can be reconstructed using gradient similarity. However, this method relies on the sign of the gradient, and in the context of homomorphic encryption, the method of obtaining data information through gradient similarity is ineffective Cheon et al. [24].
2) Discuss security when it comes to secure aggregation and model distribution. During secure aggregation, the client first uses CKKS to encrypt its training parameter $[G_x^t]_{pk_c}$, and then sends it to the DC. The DC performs a homomorphic operation on the encrypted parameters (to achieve the purpose of fuzzy parameters) to obtain $E_i = [G_i^t]_{pk_c} * r_i, i \in \{x, d\}$, and sends it to the CS. After the CS is decrypted, the similarity calculation is performed to obtain the similarity value $\rho_{x,d}$. CS calculates the aggregation weight v_x according to the similarity value, then CS normalizes the client model to obtain $r_d * \overline{G_x}$, then calculates the aggregated gradient $e_x = \alpha \frac{v_x}{\sum_{x \in [1,2,\ldots,m]} v_x} r_d * \overline{G_x}$

with noise added, and then encrypts it and sends it to DC. After DC removes noise, the model is updated. In this aggregation process, except for Person correlation coefficient and aggregation weight, DC and CS have no direct contact with parameters, and all parameters are invisible during the aggregation process, so the data in the aggregation process is safe. Among them, Person correlation coefficient privacy does not need to be protected (explained in the previous paragraph), and v_x is obtained by Person correlation coefficient, so it does not need to be protected, so the secure aggregation process is safe. When distributing the model, the model $[\omega^t]_{pk_c}$ owned by the DC is encrypted with the CS public key. The homomorphic encryption operation of $[\omega^t]_{pk_c}$ is performed to obtain $R = [\omega^t]_{pk_c} + [r]_{pk_c}$, and R is sent to CS. CS re-encrypts it to obtain $R' = [d]_{pk_x}$, and then sends it to DC, DC After removing the noise, the $[\omega^t]_{pk_x}$ is obtained and sent to each client. Neither DC nor CS obtains the original model ω^t in this distribution process, so the model distribution process is safe.

6.3 Performance evaluation

6.3.1 Experimental setup

1) **Dataset**: In order to evaluate the performance of the intrusion detection method based on federated learning in the smart grid AMI network, this paper adopts the NSL-KDD dataset for testing, and the data includes training set and test set. The data set is randomly allocated on the client side, and the data center has non-attack NSL-KDD data. It is assumed that the sample distribution of the data center is biased towards a certain type, that is, the proportion of the samples of the data center to a certain type of samples is q, and the other types the sample probability is $(1 - q)/4$, and the remaining types of samples have the same proportion, this q is called the data center sample bias.
2) **Evaluated Poisoning Attack**: Trim attack is a kind of off-target local model poisoning attack optimized for Trim mean and median aggregation rules. Because attackers poison the client's data, it is also reflected in the impact training model. Therefore, this paper uses Trim attack to test the method in this paper, which can protect the client from data poisoning attacks and model poisoning attacks.
3) **Evaluation Metrics**: Since the Trim attack aims to improve the test error rate, this paper uses the test error rate of the global model to evaluate the robustness, where the test error rate of the global model refers to the proportion of labels that the global model mispredicts. When the proposed method obtains a lower test error rate under this attack, the method is robust against this attack. This paper also uses the accuracy rate as one of the evaluation metrics of our method.
4) **System settings**: In this paper, the number of clients is set to 50, the proportion of malicious clients is 40%, the data of each client is randomly allocated, the data center is allocated 100 training data, and the distribution of data in the data center is the same as the distribution of the overall training data. The model in this paper is an intrusion detection model with 2000 federated learning iterations under the fully connected neural network. Unless otherwise stated, experiments were performed with this system setup. In order to test whether the proposed method can resist the poisoning attack and reasoning attack of federated learning in smart grid AMI intrusion detection, this paper uses a simple fully connected neural network, which can be replaced by other neural networks later.

TABLE 3 Test error rate.

Method	FedAvg	FLTrust	PEFL	Proposed method
No attack	0.1465	0.1602	0.1523	0.1494
Trim attack	0.2881	0.1611	0.1553	0.1523

6.3.2 Experimental results

1) **Robustness evaluation**: This paper compares with FLTrust Wang et al. [25], FedAvg Cheng et al. [26], PEFL Liu et al. [27] methods. Table 3 shows the test error rate when the distribution of the data center samples is the same as the distribution of the overall training samples. At this time, the number of clients is 50, the data of each client is randomly allocated, the data center allocates 100 training data, and the distribution of data in the data center is the same as that of the overall training data. When there is no attack, the proportion of malicious clients is 0%, and when there is an attack, the proportion of malicious clients is 40%.

Without the attack, the test error rate of the proposed method is lower, while FedAvg, FLTrust and PEFL have higher test error rate. It shows that this method has no negative impact on the federated aggregation process when there is no attack. For example, the test error rate of the proposed method is 0.1494, while the test error rates of FedAvg, FLTrust and PEFL are 0.1465, 0.1602 and 0.1523. The results show that the proposed method is more accurate than other methods against poisoning attacks in the absence of attacks. In the presence of Trim attack, the test error rate of the proposed method, FLTrust and PEFL is higher than that of FedAvg without attack, but the test error rate of the proposed method is the closest to FedAvg without attack, which indicates that the proposed method is robust. When there is an attack, the proposed method has a lower test error rate than other methods (FedAvg, FLTrust, PEFL) because our method integrates all client data information for detection. FedAvg has no defenses, and when there is a malicious attack, it may be

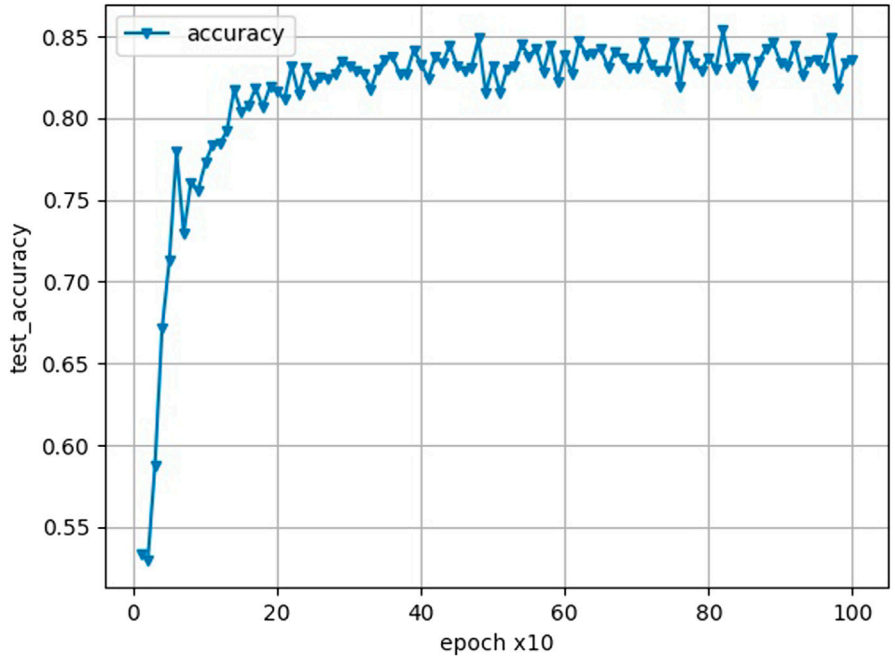


FIGURE 2
The effect of iteration on accuracy.

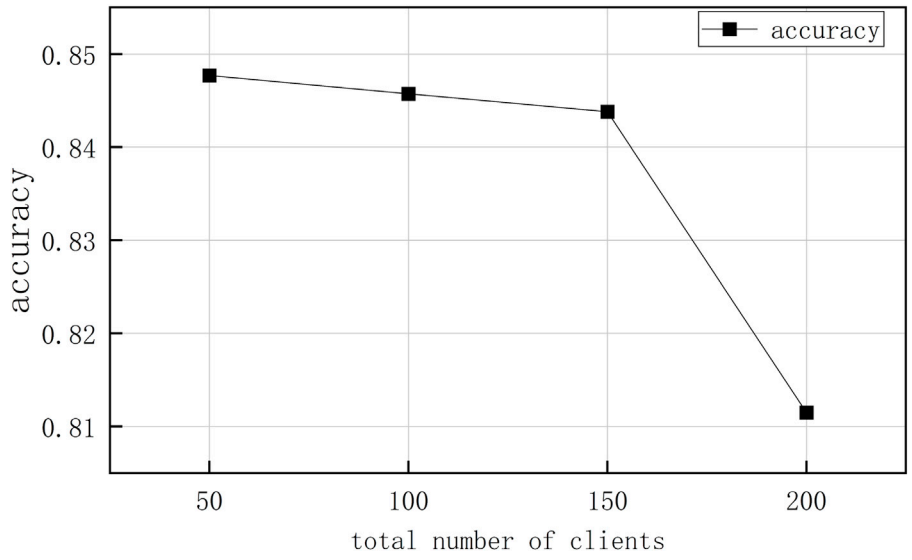


FIGURE 3
The effect of the total number of clients on the accuracy.

guided by the malicious attack model. FLTrust updates the model through the joint training of the data center model and the client model, and uses the pruning cosine method to eliminate the client models with a similarity less than zero, and retain all the client models greater than zero. So when the client model with low similarity and not less than zero participates in model aggregation, the global model will also be affected. The accuracy of PEFL is similar to the

TABLE 4 The effect of malicious client ratio on accuracy.

Malicious ratio (%)	0	10	20	40	60	80	90	100
FedAvg	0.8535	0.8271	0.7559	0.7119	0.6807	0.6299	0.4639	0.2490
FLTrust	0.8477	0.8467	0.8428	0.8447	0.5479	0.4648	0.4648	0.4596
PEFL	0.8398	0.8389	0.8359	0.8389	0.8389	0.8379	0.8135	0.2549
Proposed method	0.8506	0.8447	0.8438	0.8477	0.8467	0.8496	0.8350	0.2510

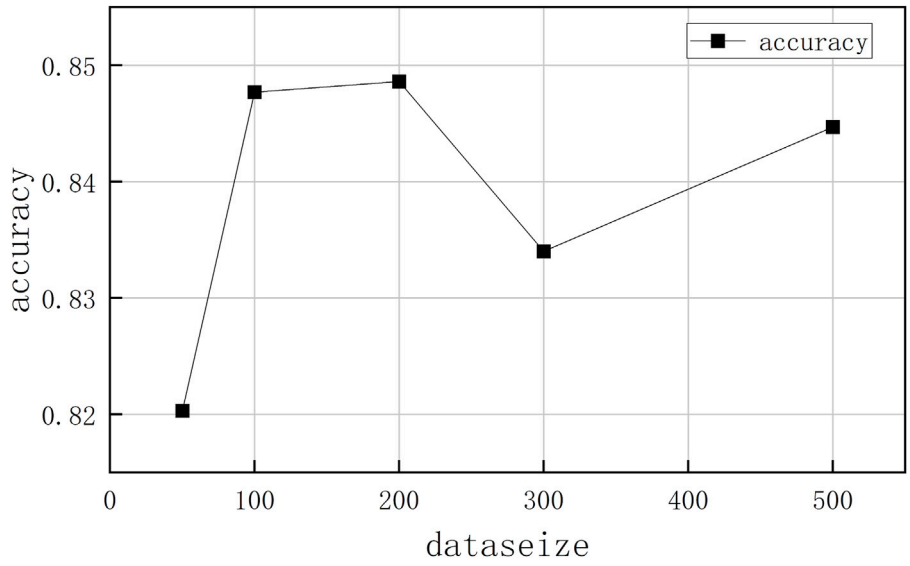


FIGURE 4
The impact of data center sample size on accuracy.

proposed method, because when the number of malicious attackers is less than 50% of the total number, the baseline model selected by PEFL according to the median dimension is highly likely to bypass the malicious model, and the Pearson correlation coefficient and A logarithmic function is used to generate aggregate weights, assigning larger weights to clients with high similarity and suppressing their weight values for smaller clients.

2) A**ccuracy evaluation**: Many factors affect the size of the accuracy, s. ch as model type and structure, sample quality, number of federated learning iterations, total number of clients and proportion of malicious clients, and data center data volume and distribution (bias). This paper tests the effect of the number of federated learning iterations, the number of clients and the proportion of malicious clients, as well as the number and distribution of samples in the data center on the accuracy.

Figure 2 shows the change in the accuracy of the proposed method for 1,000 federated learning iterations when the number of clients is 50, the data of each client is randomly allocated, the data center allocates 100 training data, and the distribution of data in the data center is the same as that of the overall training data, and the proportion of malicious clients is 20%. When the number of federated learning iterations raises, the precision of the model improves. Before 250 federated learning iterations, the model's accuracy raises significantly, and after 400 federated learning iterations, the model's precision stabilizes.

Figure 3 shows the effect of the total number of clients on the model accuracy. At this time, the proportion of malicious clients is 40%, the data of each client is randomly allocated, and the data center allocates 100 training data, and the distribution of data in the data center is the same as that of the overall training data. As the number of clients raises, the precision of the model decreases. The total number of clients ranges from 50 to 200. When the total number of clients is 50, 100, 150, and 200, the intrusion detection rates of the proposed method are 0.8477, 0.8457, 0.8438, and 0.8115, respectively. Since the number of samples of the client is more, the training effect of the model is more friendly. Therefore, when the total number of clients increases, the

TABLE 5 The impact of data center sample distribution on accuracy.

deviation	$q = 0.2$	$q = 0.4$	$q = 0.6$	$q = 0.8$	$q = 1$
No attack	0.8411	0.8496	0.8389	0.4951	0.3574
Trim attack	0.8320	0.8154	0.8096	0.3818	0.3496

number of samples on each client decreases, which may have a certain impact on the performance of the model.

Table 4 shows the effect of different malicious client proportions on the model accuracy. At this time, the number of clients is 50, the data of each client is randomly allocated, the data center allocates 100 training data, and the distribution of data in the data center is the same as that of the overall training data. As the number of poisoned clients raises, the precision of all models decreases. This is because the amount of malicious data raises, the amount of normal data decreases, and there is too little correct information for global model training, resulting in a drop in accuracy. However, the method proposed in this paper can maintain a similar accuracy rate as FedAvg without attack when the proportion of malicious clients is 90%. Other methods, such as FedAvg and FLTrust, can only tolerate less than 60% of malicious clients. When the proportion of malicious clients is 20%, the FedAvg accuracy decreases from 0.8535 to 0.7559. When the proportion of malicious clients is 60%, the accuracy of FLTrust decreases from 0.8477 to 0.5479. This shows that the method proposed in this paper has better performance against Trim.

Figure 4 shows the effect of the number of data center samples on the accuracy of the intrusion detection model. At this time, the number of clients is 50, the proportion of malicious clients is 40%, the data of each client is randomly distributed, and the distribution of data in the data center is the same as that of the overall training data. Here, the number of data center samples is set to 50–500, corresponding to different accuracy rates. When the number of samples in the data center is 100, the accuracy rate reaches 84.77%, and then when the number of samples in the data center increases, the accuracy rate decreases slightly, indicating that the data center can verify the proposed method with 100 data samples, and obtain Good detection effect.

Table 5 describes the impact of the data distribution of the data center on the model accuracy. At this time, the number of clients is 50, and the proportion of malicious clients is 40%. The data of each client is randomly allocated, and the data center allocates 100 training data. The accuracy rate is the highest when the sample distribution deviation of the data center is 0.2, and the larger the deviation, the lower the accuracy rate. When $q = 0.2$, it means that the number of various samples in the data center is the same, and the more uniform the distribution of samples in the data center, the better the trained model can reflect the characteristics of various samples. The model trained in the data center is used as the benchmark model for eliminating malicious models. The more reliable the benchmark model is, the more accurate the malicious model can be eliminated.

7 Conclusion

This paper studies the possible inference attacks and poisoning attacks in the joint training of smart grid AMI using federated learning technology. Firstly, the reasons for inference attack and poisoning attack in smart grid AMI using federated learning technology are analyzed. Secondly, the threat model of smart grid AMI scenario when applying federated learning technology is proposed, and our defense target is proposed according to the threat model, and then the system model is proposed. Next, this paper proposes a federated learning intrusion detection method against inference attacks and poisoning attacks, which is achieved through three processes:

1) Client local training: This process is that each local client trains the model and encrypts the trained model using the encryption technology CKKS and then uploads it.
2) Secure aggregation: In the CKKS environment, the Pearson correlation coefficient is used to calculate the directional similarity between the model trained by the client and the model trained by the data center, and the scaled similarity value is used as the adaptive weight value of the server model aggregation. At the same time, the model gradient amplitudes uploaded by each client are normalized according to the data center model gradient amplitudes.
3) Model distribution: The data center distributes the aggregated global model to each client. The initial aggregated global model is encrypted with the public key of the control server. If it is directly distributed to each client, the client cannot access the specific model. Therefore, the global model is re-encrypted using the client's public key with the cooperation of the data center and the control server.

Finally, the optimization analysis of the algorithm, the safety calculation analysis and the performance evaluation of the proposed

method are carried out. The research results show that the proposed method can effectively resist inference attacks and poisoning attacks, and the intrusion detection based on federated learning can maintain a good detection performance in the AMI network.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: https://www.unb.ca/cic/datasets/nsl.html.

Author contributions

Writing—original draft, JW; methodology and resources, JW and YC; funding acquisition and supervision, YC and ZX; software and visualization, ZX, CH, and FY. All authors have read and agreed to the published version of the manuscript.

Funding

This work was funded by National Natural Science Foundation of China (NSFC) 52177067, the Research Project on Teaching Reform of General Colleges and Universities in Hunan Province (Grant No. HNJG-2020-0261), China.

Acknowledgments

This is a short text to acknowledge the contributions of specific colleagues, institutions, or agencies that aided the efforts of the authors.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Li Y, Qiu R, Jing S. Intrusion detection system using online sequence extreme learning machine (os-elm) in advanced metering infrastructure of smart grid. *PloS one* (2018) 13:e0192216. doi:10.1371/journal.pone.0192216

2. Hasan MN, Toma RN, Nahid A-A, Islam MM, Kim J-M. Electricity theft detection in smart grid systems: A cnn-lstm based approach. *Energies* (2019) 12: 3310. doi:10.3390/en12173310

3. Su Z, Wang Y, Luan TH, Zhang N, Li F, Chen T, et al. Secure and efficient federated learning for smart grid with edge-cloud collaboration. *IEEE Trans Ind Inf* (2021) 18:1333–44. doi:10.1109/tii.2021.3095506

4. Yu F, Kong X, Mokbel AAM, Yao W, Cai S. Complex dynamics, hardware implementation and image encryption application of multiscroll memeristive hopfield neural network with a novel local active memeristor. *IEEE Trans Circuits Syst* (2022) 1. doi:10.1109/tcsii.2022.3218468

5. Shen H, Yu F, Wang C, Sun J, Cai S. Firing mechanism based on single memristive neuron and double memristive coupled neurons. *Nonlinear Dyn* (2022) 110:3807–22. doi:10.1007/s11071-022-07812-w

6. Yu F, Shen H, Yu Q, Kong X, Sharma PK, Cai S. Privacy protection of medical data based on multi-scroll memristive hopfield neural network. *IEEE Trans Netw Sci Eng* (2022) 1–14. doi:10.1109/tnse.2022.3223930

7. Wu C, Luo C, Xiong N, Zhang W, Kim T-H. A greedy deep learning method for medical disease analysis. *IEEE Access* (2018) 6:20021–30. doi:10.1109/access.2018.2823979

8. Kumar P, Kumar R, Srivastava G, Gupta GP, Tripathi R, Gadekallu TR, et al. Ppsf: A privacy-preserving and secure framework using blockchain-based machine-learning for iot-driven smart cities. *IEEE Trans Netw Sci Eng* (2021) 8:2326–41. doi:10.1109/tnse.2021.3089435

9. Liu X, Li H, Xu G, Lu R, He M. Adaptive privacy-preserving federated learning. *Peer-to-peer Netw Appl* (2020) 13:2356–66. doi:10.1007/s12083-019-00869-2

10. Bonawitz K, Ivanov V, Kreuter B, Marcedone A, McMahan HB, Patel S, et al. Practical secure aggregation for privacy-preserving machine learning. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, Dallas, Texas, October 30–November 03, 2017 (2017). p. 1175–91.

11. Mohassel P, Zhang Y. Secureml: A system for scalable privacy-preserving machine learning. In: 2017 IEEE symposium on security and privacy (SP); 22-26 May 2017; San Jose, CA, USA. IEEE (2017). p. 19–38.

12. Riazi MS, Weinert C, Tkachenko O, Songhori EM, Schneider T, Koushanfar F. Chameleon: A hybrid secure computation framework for machine learning applications. In: Proceedings of the 2018 on Asia conference on computer and communications security, Incheon, South Korea, June 04–June 04, 2018 (2018). p. 707–21.

13. Phong LT, Aono Y, Hayashi T, Wang L, Moriai S Privacy-preserving deep learning via additively homomorphic encryption. *IEEE Trans Inf Forensics Security* (2017) 13:1333–45.

14. Xu G, Li H, Zhang Y, Xu S, Ning J, Deng R. Privacy-preserving federated deep learning with irregular users. *IEEE Trans Dependable Secure Comput* (2020) 1. doi:10.1109/tdsc.2020.3005909

15. Fang H, Qian Q. Privacy preserving machine learning with homomorphic encryption and federated learning. *Future Internet* (2021) 13:94. doi:10.3390/fi13040094

16. Blanchard P, El Mhamdi EM, Guerraoui R, Stainer J. Machine learning with adversaries: Byzantine tolerant gradient descent. *Adv Neural Inf Process Syst* (2017) 30.

17. Xia Q, Tao Z, Hao Z, Li Q. Faba: An algorithm for fast aggregation against byzantine attacks in distributed neural networks. In: Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence Main track, Macao, China, August 10–August 16, 2019 (2019). p. 4824–30. doi:10.24963/ijcai.2019/670

18. Yin D, Chen Y, Kannan R, Bartlett P. Byzantine-robust distributed learning: Towards optimal statistical rates. In: International Conference on Machine Learning (PMLR), Stockholm Sweden, July 10–July 15, 2018 (2018). p. 5650–9.

19. Chen Y, Su L, Xu J. Distributed statistical machine learning in adversarial settings: Byzantine gradient descent. *Proc ACM Meas Anal Comput Syst* (2017) 1: 1–25. doi:10.1145/3154503

20. Fang M, Cao X, Jia J, Gong N. Local model poisoning attacks to {Byzantine – Robust} federated learning. In: 29th USENIX Security Symposium, Boston, United States, August 12–August 14, 2020. USENIX Security (2020). p. 1605–22.

21. Xie C, Koyejo S, Gupta I. Zeno: Distributed stochastic gradient descent with suspicion-based fault-tolerance. In: International Conference on Machine Learning (PMLR), Long Beach, California, United States, June 9–June 15, 2019 (2019). p. 6893–901.

22. Zhao L, Hu S, Wang Q, Jiang J, Shen C, Luo X, et al. Shielding collaborative learning: Mitigating poisoning attacks through client-side detection. *IEEE Trans Dependable Secure Comput* (2020) 18:1–2041. doi:10.1109/tdsc.2020.2986205

23. Nguyen TD, Rieger P, Chen H, Yalame H, Möllering H, Fereidooni H, et al. {FLAME}: Taming backdoors in federated learning. In: 31st USENIX Security Symposium, Boston, MA, United States, August 10–August 12, 2022. USENIX Security (2022). p. 1415–32.

24. Cheon JH, Kim A, Kim M, Song Y. Homomorphic encryption for arithmetic of approximate numbers. In: International conference on the theory and application of cryptology and information security. Springer (2017). p. 409–37.

25. Wang J, Ding W, He M, Hu J, Xiong N. Reducing tail latency with coding-based packet spraying in edge datacenters. *J Syst Architecture* (2022) 134:102783. doi:10.1016/j.sysarc.2022.102783

26. Cheng H, Xie Z, Shi Y, Xiong N. Multi-step data prediction in wireless sensor networks based on one-dimensional cnn and bidirectional lstm. *IEEE Access* (2019) 7:117883–96. doi:10.1109/access.2019.2937098

27. Liu X, Li H, Xu G, Chen Z, Huang X, Lu R. Privacy-enhanced federated learning against poisoning adversaries. *IEEE Trans Inform Forensic Secur* (2021) 16: 4574–88. doi:10.1109/tifs.2021.3108434

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 23 January 2023
DOI 10.3389/fphy.2023.1129844

OPEN ACCESS

EDITED BY
Fei Yu,
Changsha University of Science and Technology, China

REVIEWED BY
Zhengquan Yang,
Civil Aviation University of China, China
Junjie Fu,
Southeast University, China

*CORRESPONDENCE
Linying Xiang,
xianglinying@neuq.edu.cn

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 22 December 2022
ACCEPTED 11 January 2023
PUBLISHED 23 January 2023

CITATION
Sun J, Xiang L and Chen G (2023), A new effective metric for dynamical robustness of directed networks.
Front. Phys. 11:1129844.
doi: 10.3389/fphy.2023.1129844

A new effective metric for dynamical robustness of directed networks

Jiashuo Sun[1], Linying Xiang[1]* and Guanrong Chen[2]

[1]School of Control Engineering, Northeastern University at Qinhuangdao, Qinhuangdao, Hebei, China, [2]Department of Electrical Engineering, City University of Hong Kong, Kowloon, China

In this article, dynamical robustness of a directed complex network with additive noise is inverstigated. The failure of a node in the network is modeled by injecting noise into the node. Under the framework of mean-square stochastic stability, a new robustness metric is formulated to characterize the robustness of the network in terms of synchronization to the additive noise. It is found that the node dynamics plays a pivotal role in dynamical robustness of the directed network. Numerical simulations are shown for illustration and verification.

KEYWORDS

complex network, directed network, dynamical robustness, noise, synchronization

1 Introduction

In practical applications, power grids [1, 2], communication networks, secure communication [3] and public transportation systems [4, 5] often encounter failures and attacks [6–9]. A failure of a very small fraction of nodes in a network may lead to complete fragmentation of the whole network. Therefore, the robustness of complex networks subjected to failures or attacks is an important issue to study in network science and engineering [10]. Exploring the network robustness can help better understand various networked systems and enable us to design more robust infrastructural or social systems.

In the past 2 decades, the issue of network robustness has attracted a lot of attention [10–15]. Most of the previous works focus on the *structural robustness* of complex networks, which is defined as the ability to maintaining their functionalities when they are disturbed or attacked [16, 17]. Therein, the failure of a node or an edge in the network is modeled by the removal of the node or the edge. To quantify structural robustness of complex networks, many measures with respect to the network structure have been formulated, such as connectivity, maximum strongly connected subgraph, natural connectivity, and average shortest path [18, 19]. Moreover, a variety of attack strategies, such as random attack and deliberate attack, have been proposed to test the robustness of different kinds of networks [20–22]. The structural robustness of complex networks can also be measured using some metrics derived from statistical physics and percolation theory [23].

Recently, the network robustness with respect to the system dynamics has stimulated even more interest [24–26]. A new concept of *dynamical robustness* [27] can be used to quantify the ability of a network to maintain its dynamical activities against local perturbations. Different from structural robustness where topological perturbations are considered, dynamical robustness is concerned with the robustness of network dynamics. In [27], node failure is modeled as the inactivation of diffusively coupled oscillators. In [28], dynamical robustness is quantified through the synchronization error as a function of the noise variance, where node failure is modeled by injecting noise into a node. In [29], a mathematical framework is established to quantify the effect of noise injected at one of the nodes on the synchronization performance of coupled dynamical systems. Indeed, noise is inevitable in real-world networks

[30, 31]. It is natural to ask whether a networked system subjected to noise can recover to its synchronous state? On the one hand, noise may destroy the network's stability and prevent network synchronization [32, 33]. On the other hand, noise-induced synchronization can be beneficial for coupled chaotic systems [34]. In order to clarify the influence of noise on the network, in [35] networks of different structure and complexity are analyzed, showing that many networks are better in coping with both intrinsic and extrinsic noise.

Motivated by the above discussions, this article further investigates the dynamical robustness of stochastic complex networks. The network topology is directed and the nodes are higher-dimensional non-linear dynamical systems. Similar to [29], the failure of a node in the network is modeled by injecting noise into the node. However, differing from [29], in this article it is not assumed that the network is symmetric. From a technical perspective, this introduces more challenges than its undirected counterpart [36]. A novel metric measuring the dynamical robustness of a directed networked system with additive noise is formulated. Notice that the proposed robustness metric uncovers the complex interplay between node dynamics and network topology on the overall network robustness.

The main contributions of this article are as follows. First, a mathematical framework is established to examine the dynamical robustness of a directed network of coupled dynamical systems. In this article, the notion of dynamical robustness refers to the ability of a network of coupled dynamical systems to return to its synchronous state when it encounters the disturbance of noise. The new metric is used to characterize the degree to which the networked system withstand failures and perturbations. Assume that the networked system synchronizes before the noise is introduced. The system's robustness is defined related to the synchronization error of the network. Moreover, different from the methods used for undirected networks, the Laplacian matrix of a directed network is decomposed to two simpler matrices. In the context of mean-square stochastic stability, the new robustness metric is precisely formulated. This metric highlights the importance of the node dynamics in network robustness. Finally, numerical simulations are presented for illustration and verification using three chaotic systems (namely, Rössler system, Chen system and Wang system). The study of dynamical robustness can help better understand the roles of node dynamics and network topology, thereby better designing noise-tolerant networks.

The remainder of this article is organized as follows. Section 2 introduces the notation and some basic graph theory. In Section 3, problem formulation is presented and a new robustness metric is formulated. In Section 4, numerical simulations are shown for illustration and verification. Section 5 concludes the investigation.

2 Preliminaries

2.1 Notation

Let $\mathbb{R}$ denote the set of real numbers, $\mathbb{R}^n$ the set of the n-dimensional real vectors, and $\mathbb{R}^{n\times m}$ the set of $n\times m$ real matrices. Let I_n be the $n\times n$ identity matrix, $\mathbf{1}_n$ the column vector of all ones, $\mathbf{0}$ the zero matrix with appropriate dimension, and diag $(a_1,\ldots,a_n)$ the $n\times n$ diagonal matrix with the diagonal elements being $a_1,\ldots,a_n$. Let the trace of matrix A be denoted by $\mathbf{Tr}(A)$. Moreover, let $\|\cdot\|$ denote the 2-norm of a matrix or a vector, $\otimes$ the Kronecker product, and $\oplus$ the Kronecker sum. Let the superscript T denote the transpose. Let $\mathbf{j}$ denote the imaginary unit satisfying $\mathbf{j}^2=-1$. For a matrix $A\in\mathbb{R}^{m\times n}$, $Vec(A)=[col_1^T(A),\ldots,col_n^T(A)]^T\in\mathbb{R}^{mn}$ is the colum vector of size $mn\times 1$ obtained by stacking all columns of A, where $col_i(A)\in\mathbb{R}^m$ denotes the ith column of A.

2.2 Graph theory

A directed graph $\mathcal{G}=(\mathcal{V},\mathcal{E})$ consists of a node set $\mathcal{V}=\{1,\ldots,N\}$ and an edge set $\mathcal{E}=\{(j,i)\}$. Let $\mathcal{A}=(a_{ij})\in\mathbb{R}^{N\times N}$ denote the adjacency matrix of a digraph, where $a_{ij}=1$ if there is a directed edge from node j to node i, and $a_{ij}=0$ otherwise. Moreover, $a_{ii}=0$ for all $i=1,\ldots,N$. Let $\mathcal{D}=\mathrm{diag}(d_1^{in},\ldots,d_N^{in})$ be the in-degree matrix, where d_i^{in} represents the in-degree of node i. The Laplacian matrix is then defined by $\mathcal{L}=\mathcal{D}-\mathcal{A}$. The Laplacian matrix can be decomposed as $\mathcal{L}=U+\Delta$, where $U=\frac{1}{2}(\mathcal{L}+\mathcal{L}^T)$ is a symmetric matrix and $\Delta=\frac{1}{2}(\mathcal{L}-\mathcal{L}^T)$ is an anti-symmetric matrix satisfying $\Delta^T=-\Delta$.

For the anti-symmetric matrix Δ, the following lemma is obtained.

Lemma 1. Let $\Delta\in\mathbb{R}^{N\times N}$ be an anti-symmetric matrix satisfying $\Delta^T=-\Delta$. Then, there exists an orthogonal matrix C such that

$$C^T\Delta C=\mathrm{diag}\left(\mathbf{0},\ldots,\mathbf{0},\begin{pmatrix}0&b_1\\-b_1&0\end{pmatrix},\ldots,\begin{pmatrix}0&b_l\\-b_l&0\end{pmatrix}\right),\tag{1}$$

where $0,\ldots,0,\pm b_1\mathbf{j},\ldots,\pm b_l\mathbf{j}$ $(b_i\neq 0)$ *are the eigenvalues of the matrix* Δ. *Here,* $l=\frac{N-r}{2}$ *with* r *being the multiplicity of the zero eigenvalue of* Δ.

Proof: Let $\mu_1=b_1\mathbf{j}$ be an eigenvalue of Δ and χ_1 the corresponding eigenvector. Let $\bar{\mu}_1=-b_1\mathbf{j}$. It follows that $\Delta\chi_1=\mu_1\chi_1$, $\Delta\bar{\chi}_1=\bar{\mu}_1\bar{\chi}_1$, and $\chi_1\neq\bar{\chi}_1$, where $\bar{\chi}_1$ is an eigenvector of Δ associated with $\bar{\mu}_1$.

Recall that Δ is anti-symmetric. Consequently, it is a normal matrix and is unitary similar to a diagonal matrix. Particularly, let $P=(0,\ldots,0,\varsigma_1,\bar{\varsigma}_1,\ldots,\varsigma_l,\bar{\varsigma}_l)$, where $\varsigma_i\in\mathbb{R}^N$, $\bar{\varsigma}_i\in\mathbb{R}^N$, $i=1,\ldots,l$, and $\varsigma_i\neq\bar{\varsigma}_i$. It follows that

$$P^{-1}\Delta P=\mathrm{diag}(0,\ldots,0,\mu_1,\bar{\mu}_1,\ldots,\mu_l,\bar{\mu}_l).\tag{2}$$

Let $\varphi_1=\frac{\varsigma_1+\bar{\varsigma}_1}{\sqrt{2}}$ and $\varphi_2=\frac{\varsigma_1-\bar{\varsigma}_1}{\sqrt{2}\mathbf{j}}$. One has

$$\Delta\varphi_1=\frac{1}{\sqrt{2}}(\Delta\varsigma_1+\Delta\bar{\varsigma}_1)=\frac{1}{\sqrt{2}}(\mu_1\varsigma_1+\bar{\mu}_1\bar{\varsigma}_1)=-b_1\varphi_2,$$
$$\Delta\varphi_2=\frac{1}{\sqrt{2}\mathbf{j}}(\Delta\varsigma_1-\Delta\bar{\varsigma}_1)=\frac{1}{\sqrt{2}\mathbf{j}}(\mu_1\varsigma_1-\bar{\mu}_1\bar{\varsigma}_1)=b_1\varphi_1.$$

Because p is a unitary matrix, $\varphi_1^T\varphi_1=1$, $\varphi_2^T\varphi_2=1$, and $\bar{\varsigma}_j^T\varphi_1=0$ $(j=2,\ldots,l)$. Similarly, $\varphi_3,\ \ldots\ ,\ \varphi_{2l}$ have the same property. It follows that $C=(0,\ \ldots\ ,\ 0,\ \varphi_1,\ \ldots\ ,\ \varphi_{2l})$ is an orthogonal matrix, in which the number of zero eigenvalues is r and $r+2l=N$. Therefore,

$$C^T\Delta C=\mathrm{diag}\left(\mathbf{0},\ldots,\mathbf{0},\begin{pmatrix}0&b_1\\-b_1&0\end{pmatrix},\ldots,\begin{pmatrix}0&b_l\\-b_l&0\end{pmatrix}\right).$$

3 The new robustness metric

Consider a directed network consisting of N identical nodes with linearly diffusive couplings, in which each node is an n-dimensional

dynamical system. The network can be described by the following coupled stochastic differential equation:

$$\dot{x}_i(t) = f(x_i(t)) - \alpha \sum_{j=1}^{N} l_{ij} h\big(x_j(t)\big) + v_i H_\eta \eta(t), \quad i = 1, \ldots, N, \tag{3}$$

where $x_i(t) \in \mathbb{R}^n$ is the state vector of node i, $f(\cdot)$ a smooth function describing the self-dynamics of each node, $\alpha > 0$ the coupling strength, l_{ij} the (i, j)th entry of the Laplacian matrix, and $h(\cdot)$ the inner-coupling function of the nodes. The variable v_i indicates whether node i is subjected to noise. That is, $v_i = 1$ when node i is contaminated with additive noise, and $v_i = 0$ otherwise. The vector $H_\eta \in \mathbb{R}^n$ describes how the noise $\eta(t)$ enters the dynamics of a node, where $\eta(t)$ is a zero-mean Gaussian white noise with variance $\frac{\theta}{2}$ $(\theta > 0)$.

For network (3), $\mathcal{L} = (l_{ij}) \in \mathbb{R}^{N\times N}$ is the Laplacian matrix defined by $l_{ij} = -1$ if there is a directed edge from node j to node i, and $l_{ij} = 0$ otherwise, with $l_{ii} = -\sum_{j=1, j\neq i}^{N} l_{ij}$, for all $i, j = 1, \ldots, N$. Therefore, $\mathcal{L}$ is a zero row-sum matrix. Recall from Section 2.2 that $\mathcal{L} = U + \Delta$, in which $U = \frac{1}{2}(\mathcal{L} + \mathcal{L}^T)$ is a symmetric matrix and $\Delta = \frac{1}{2}(\mathcal{L} - \mathcal{L}^T)$ is an anti-symmetric matrix with $\Delta^T = -\Delta$. Therefore, network (3) can be rewritten as

$$\begin{aligned}\dot{x}_i(t) = f(x_i(t)) - \alpha \sum_{j=1}^{N} u_{ij} h\big(x_j(t)\big) - \alpha \sum_{j=1}^{N} \delta_{ij} h\big(x_j(t)\big) \\ + v_i H_\eta \eta(t), \quad i = 1, \ldots, N,\end{aligned} \tag{4}$$

where u_{ij} is the (i, j)th element of U and δ_{ij} is the (i, j)th element of Δ.

The network is said to achieve synchronization if $\lim_{t\to\infty} \|x_i(t) - s(t)\| = 0$ for all $i = 1, \ldots, N$, where $s(t)$ is the solution of $\dot{s}(t) = f(s(t))$ (see [37]).

Define the node synchronization error $\xi_i(t) = x_i(t) - s(t)$. The linearized error system is given by

$$\begin{aligned}\dot{\xi}_i(t) = J_f(s)\xi_i(t) - \alpha \sum_{j=1}^{N} u_{ij} J_h(s)\xi_j(t) \\ -\alpha \sum_{j=1}^{N} \delta_{ij} J_h(s)\xi_j(t) + v_i H_\eta \eta(t), \quad i = 1, \ldots, N,\end{aligned} \tag{5}$$

where $J_f(s) \in \mathbb{R}^{n\times n}$ and $J_h(s) \in \mathbb{R}^{n\times n}$ are, respectively, the Jacobian matrices of f and h, i.e., $J_f(s) = \frac{\partial f(x)}{\partial x}|_{x=s(t)}$ and $J_h(s) = \frac{\partial h(x)}{\partial x}|_{x=s(t)}$.

Let $\xi(t) = [\xi_1^T(t), \ldots, \xi_N^T(t)]^T \in \mathbb{R}^{Nn}$. The linearized error system 5) can be rewritten in a compact form as

$$\begin{aligned}\dot{\xi}(t) = \big[I_N \otimes J_f(s) - \alpha U \otimes J_h(s) - \alpha\Delta \otimes J_h(s)\big]\xi(t) \\ +\big(v \otimes H_\eta\big)\eta(t),\end{aligned} \tag{6}$$

where $v = [v_1, \ldots, v_N]^T \in \mathbb{R}^N$ denotes the location of the node where noise is injected.

The objective is to provide a thorough analysis of (6) to illustrate the overall effect of noise on network synchronization. Define the network synchronization error as

$$\phi(t) = \|\xi(t)\|^2. \tag{7}$$

The expected value of $\phi(t)$ is given by

$$\mathbb{E}[\phi(t)] = \mathbb{E}\big[\xi^T(t)\xi(t)\big] = \mathbf{Tr}\big(\mathbb{E}\big[\xi(t)\xi^T(t)\big]\big). \tag{8}$$

Let $\Sigma(t) = \mathbb{E}[\xi(t)\xi^T(t)]$. Note that the analysis of the effect of noise injected at the network node can be reduced to the study of the time evolution of the trace of the correlation matrix $\Sigma(t)$. Since $U^T = U$ and $\Delta^T = -\Delta$, one has

$$\begin{aligned}\dot{\Sigma}(t) &= \mathbb{E}\Big[\dot{\xi}(t)\xi^T(t) + \xi(t)\dot{\xi}^T(t)\Big] \\ &= \big[I_N \otimes J_f(s) - \alpha U \otimes J_h(s) - \alpha\Delta \otimes J_h(s)\big]\Sigma(t) \\ &\quad +\Sigma(t)\big[I_N \otimes J_f^T(s) - \alpha U \otimes J_h^T(s) + \alpha\Delta \otimes J_h^T(s)\big] \\ &\quad +\big(v \otimes H_\eta\big)\mathbb{E}\big[\eta(t)\xi^T(t)\big] + \mathbb{E}\big[\xi(t)\eta(t)\big]\big(v^T \otimes H_\eta^T\big).\end{aligned} \tag{9}$$

Notice that the solution of (6) is

$$\xi(t) = \Phi_\xi(t, 0)\xi(0) + \int_0^t \Phi_\xi(t, \tau)\big(v \otimes H_\eta\big)\eta(\tau)d\tau, \tag{10}$$

where $\Phi_\xi(t, \tau)$ is the state transition matrix associated with the state matrix $I_N \otimes J_f(s) - \alpha U \otimes J_h(s) - \alpha\Delta \otimes J_h(s)$, and $\xi(0)$ is the initial value. Recall that $\eta(t)$ is a zero-mean Gaussian white noise with variance $\frac{\theta}{2}$. According to the analysis in [38], $\mathbb{E}[\eta(t)\eta(\tau)] = \frac{\theta}{2}\delta(t - \tau)$ and $\mathbb{E}[\xi(0)\eta(\tau)] = \frac{\theta}{2}\mathbf{1}_{Nn}$, where $\delta(t)$ is the Dirac delta function. Eq. 9 can be rewritten as the following time-varying Lyapunov equation for the time evolution of the correlation matrix:

$$\begin{aligned}\dot{\Sigma}(t) = \big[I_N \otimes J_f(s) - \alpha U \otimes J_h(s) - \alpha\Delta \otimes J_h(s)\big]\Sigma(t) \\ +\Sigma(t)\big[I_N \otimes J_f^T(s) - \alpha U \otimes J_h^T(s) + \alpha\Delta \otimes J_h^T(s)\big] \\ +\theta\big(vv^T \otimes H_\eta H_\eta^T\big).\end{aligned} \tag{11}$$

Recall the definition of the matrix C in Lemma 1. Particularly, let $C = [c_1, \ldots, c_N]$, $c_i \in \mathbb{R}^N$. Let $D = C^T\Delta C$, $M = C^T U C$, $\tilde{C} = C \otimes I_n$, and $\tilde{\Sigma}(t) = \tilde{C}^T\Sigma(t)\tilde{C}$. Multiplying (11) from the left by $\tilde{C}^T$ and from the right by $\tilde{C}$ leads to

$$\begin{aligned}\dot{\tilde{\Sigma}}(t) = \big[I_N \otimes J_f(s) - \alpha M \otimes J_h(s) - \alpha D \otimes J_h(s)\big]\tilde{\Sigma}(t) \\ +\tilde{\Sigma}(t)\big[I_N \otimes J_f^T(s) - \alpha M \otimes J_h^T(s) + \alpha D \otimes J_h^T(s)\big] \\ +\theta\big(C^T vv^T C \otimes H_\eta H_\eta^T\big).\end{aligned} \tag{12}$$

Since the trace of a matrix does not change under a similarity transformation, one has $\mathrm{Tr}(\Sigma(t)) = \mathrm{Tr}(\tilde{\Sigma}(t))$ and $\mathbb{E}[\phi(t)] = \mathrm{Tr}(\tilde{\Sigma}(t))$. Let

$$\tilde{\Sigma}(t) = \sum_{i,j=1}^{N} e_i e_j^T \otimes \tilde{\sigma}_{ij}(t), \tag{13}$$

where $e_i \in \mathbb{R}^N$ denotes the ith canonical vector and $\tilde{\sigma}_{ij}(t) \in \mathbb{R}^{n\times n}$ is the (i, j)th block of the matrix $\tilde{\Sigma}(t)$. It then follows that

$$\mathbb{E}[\phi(t)] = \sum_{i=1}^{N} \mathbf{Tr}(\tilde{\sigma}_{ii}(t)). \tag{14}$$

The dynamics of $\tilde{\sigma}_{ii}(t)$ are given by

$$\begin{aligned}\dot{\tilde{\sigma}}_{ii}(t) = \big[J_f(s) - \alpha m_{ii} J_h(s)\big]\tilde{\sigma}_{ii}(t) \\ +\tilde{\sigma}_{ii}(t)\big[J_f^T(s) - \alpha m_{ii} J_h^T(s)\big] + \theta\big(c_i^T v\big)^2 H_\eta H_\eta^T, \\ i = 1, \ldots, N,\end{aligned} \tag{15}$$

where m_{ii} is the ith diagonal element of M.

Let $\zeta_i(t) = \tilde{\sigma}_{ii}(t)$, $\kappa_i = \alpha m_{ii}$, and $\beta_i = c_i^T v$. Eq. 15 can be rewritten as

$$\begin{aligned}\dot{\zeta}_i(t) = \big[J_f(s) - \kappa_i J_h(s)\big]\zeta_i(t) \\ +\zeta_i(t)\big[J_f^T(s) - \kappa_i J_h^T(s)\big] + \theta\beta_i^2 H_\eta H_\eta^T, \\ i = 1, \ldots, N.\end{aligned} \tag{16}$$

Rewrite (16) as follows:

$$\begin{aligned}Vec\big(\dot{\zeta}_i(t)\big) = \Big(\big[J_f(s) - \kappa_i J_h(s)\big] \oplus \big[J_f(s) - \kappa_i J_h(s)\big]\Big) \\ \times Vec(\zeta_i(t)) + \theta\beta_i^2 Vec\big(H_\eta H_\eta^T\big), \\ i = 1, \ldots, N,\end{aligned} \tag{17}$$

where *Vec* indicates matrix vectorization defined in Section 2.1. The solution of (17) is given by

$$\begin{aligned} Vec(\zeta_i(t)) = \ & \Phi_{Vec(\zeta_i)}(t,0)Vec(\zeta_i(0)) \\ & +\theta\beta_i^2\int_0^t \Phi_{Vec(\zeta_i)}(t,\tau)d\tau Vec\left(H_\eta H_\eta^T\right), \\ i = 1,\ldots,N, & \end{aligned} \tag{18}$$

where $\Phi_{Vec(\zeta_i)}(t,\tau)$ is the state transition matrix, which is associated with the state matrix $[J_f(s)-\kappa_i J_h(s)]\oplus[J_f(s)-\kappa_i J_h(s)]$, and $Vec(\zeta_i(0))$ is the initial value.

Based on (14)–(18), the expectation of the network synchronization error can be rewritten as

$$\begin{aligned} \mathbb{E}[\phi(t)] = \ & \sum_{i=1}^{N} Vec^T(I_n)\Phi_{Vec(\zeta_i)}(t,0)Vec(\zeta_i(0)) \\ & +\sum_{i=1}^{N} Vec^T(I_n)\theta\beta_i^2\int_0^t \Phi_{Vec(\zeta_i)}(t,\tau)d\tau Vec\left(H_\eta H_\eta^T\right). \end{aligned} \tag{19}$$

In the following, consider the stochastic linear system (6), where $J_f(s)$ and $J_h(s)$ are Jacobian matrics of f and h evaluated at $s(t)$, respectively. Constant $\alpha > 0$ is the coupling strength and $v \in \mathbb{R}^N$ denotes the location of the node where noise is injected. The vector $H_\eta \in \mathbb{R}^n$ describes how the noise $\eta(t)$ enters the dynamics of a node, where $\eta(t)$ represents zero-mean Gaussian white noise with variance $\frac{\theta}{2}$. Assume that one of the network nodes denoted by i_{noise} is contaminated with additive noise. Let $\rho = Vec^T(I_n)Vec(\zeta_{i_{noise}}(t))$ be the measurement metric of the system error, which is referred to as the *robustness metric*.

Remark 1. *It follows from the above analysis that the robustness of the directed network subjected to noise is quantified by the robustness metric ρ. Here, the networked system synchronizes before noise is introduced. The notion of dynamical robustness refers to the ability of a network of coupled dynamical systems to return to its synchronous state after it encountered the disturbance of noise. The robustness metric ρ is used to characterize the degree to which the networked system withstand failures and perturbations. It is thus defined related to the synchronization error of the network. The smaller the value of ρ, the more robust the network. Furthermore, given the location of the node where noise is injected, the dynamical robustness of the network depends not only on the node dynamics but also on the network topology. In particular, it is determined by the inherent dynamics of the isolated node $J_f(s)$, the inner-coupling function $J_h(s)$, the coupling strength α, the variance θ of the noise, and the network topology. Note that $J_f(s)$ and $J_h(s)$ are the Jacobian matrices of f and h evaluated at s(t), respectively. This implies that the robustness matric ρ is also determined by the synchronization trajectory s(t).*

4 Interplay between dynamics and topology

In this section, the effects of node dynamics and network topology on the system robustness are investigated in detail.

4.1 Node dynamics

In the following, three representative non-linear systems, namely, Rössler system, Chen system, and Wang system, are introduced. In simulations, these three systems with chaotic behaviors are adopted as the self-dynamics of the nodes, respectively.

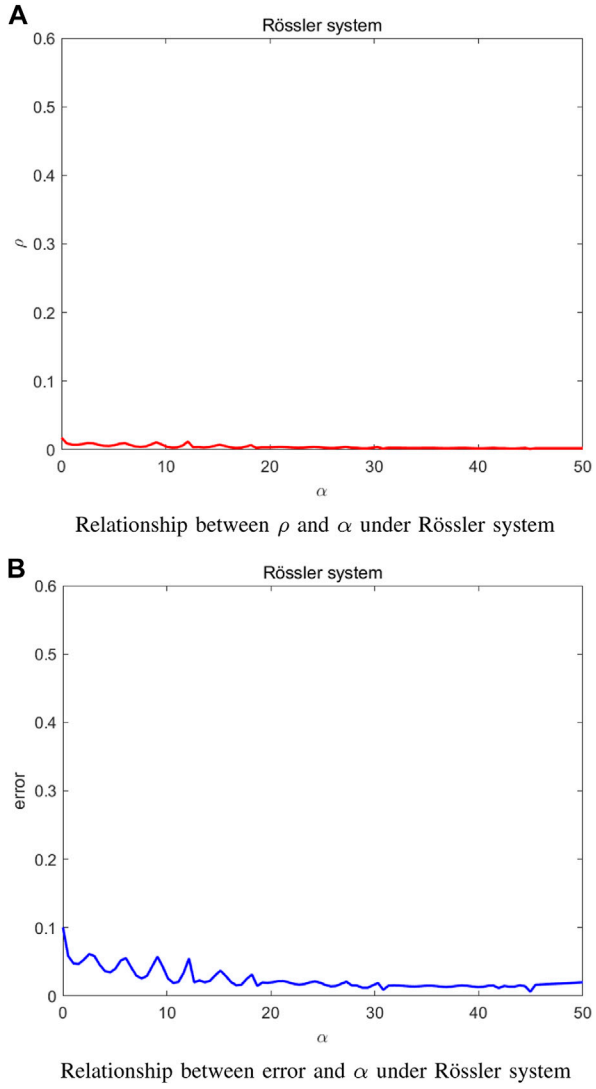


FIGURE 1
The results are obtained under Rössler systems when a directed path graph is considered.

4.1.1 Rössler system

A single Rössler system [39] is described by

$$\begin{cases} \dot{x}_1 = -x_2 - x_3, \\ \dot{x}_2 = x_1 + ax_2, \\ \dot{x}_3 = b + x_1x_3 - cx_3, \end{cases} \tag{20}$$

which has a chaotic attractor when $a = b = \frac{1}{5}$ and $c = 9$. The Jacobian matrix evaluated at $s(t) = [s_1(t), s_2(t), s_3(t)]^T$ is given by

$$J_f(s) = \begin{bmatrix} 0 & -1 & -1 \\ 1 & \frac{1}{5} & 0 \\ s_3 & 0 & s_1 - 9 \end{bmatrix}. \tag{21}$$

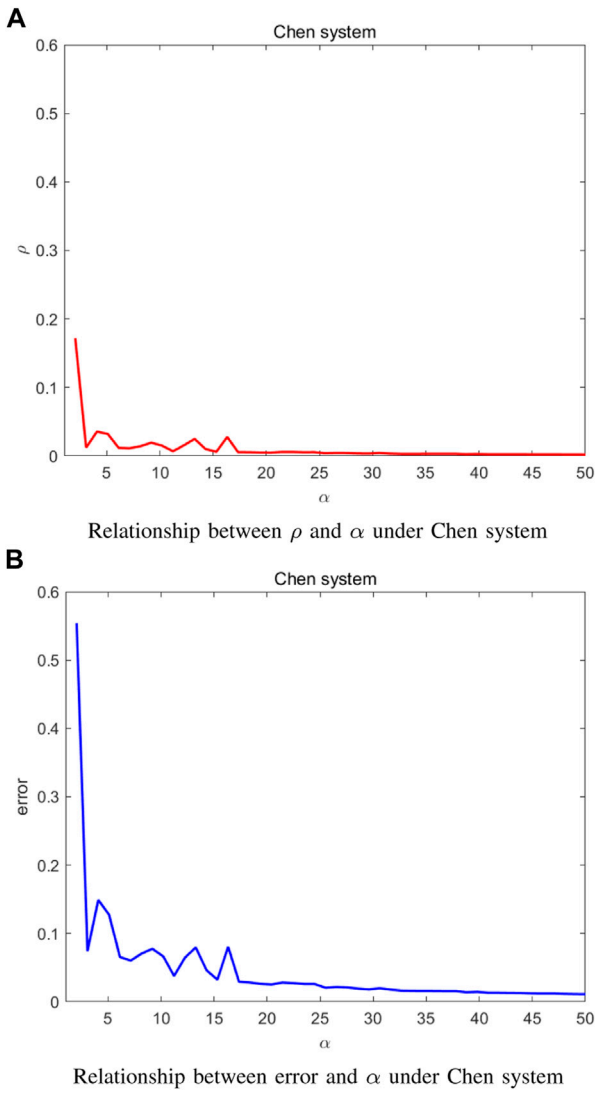


FIGURE 2
The results are obtained under Chen systems when a directed path graph is considered.

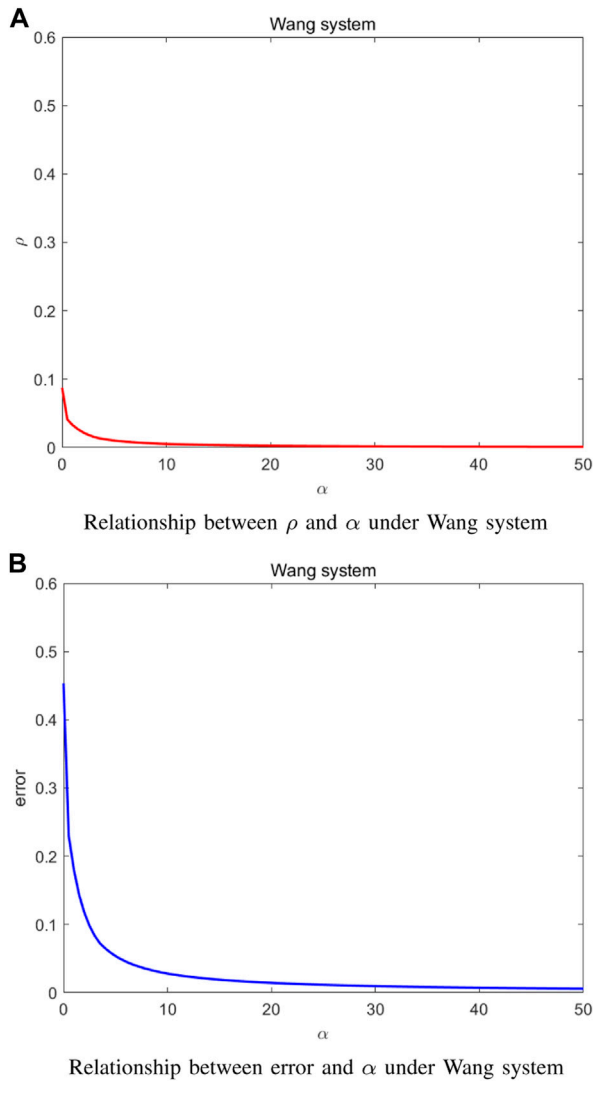


FIGURE 3
The results are obtained under Wang systems when a directed path graph is considered.

4.1.2 Chen system

A single Chen system [40] is described by

$$\begin{cases} \dot{x}_1 = a(x_2 - x_1), \\ \dot{x}_2 = (c - a - x_3)x_1 + cx_2, \\ \dot{x}_3 = x_1x_2 - bx_3, \end{cases} \tag{22}$$

which has a chaotic attractor when $a = 35$, $b = 3$, and $c = 28$. The Jacobian matrix evaluated at $s(t) = [s_1(t), s_2(t), s_3(t)]^T$ is given by

$$J_f(s) = \begin{bmatrix} -35 & 35 & 0 \\ -7 - s_3 & 28 & -s_1 \\ s_2 & s_1 & -3 \end{bmatrix}. \tag{23}$$

4.1.3 Wang system

A single Wang system [41] is described by

$$\begin{cases} \dot{x}_1 = x_2x_3 + a, \\ \dot{x}_2 = x_1^2 - x_2, \\ \dot{x}_3 = 1 - 4x_1, \end{cases} \tag{24}$$

which has a chaotic attractor when $a = 0.006$. The Jacobian matrix evaluated at $s(t) = [s_1(t), s_2(t), s_3(t)]^T$ is given by

$$J_f(s) = \begin{bmatrix} 0 & s_3 & s_2 \\ 2s_1 & -1 & 0 \\ -4 & 0 & 0 \end{bmatrix}. \tag{25}$$

4.2 Dynamical robustness beyond directed path graphs

In simulations, consider a directed path graph with five nodes. Assume that $\theta = 1$. The noise is injected into the fourth node and the third state variable of this node, that is the fourth element of v is one and $H_\eta = [0, 0, 1]^T$.

Figure 1 shows the simulation results for the network of Rössler systems. Assume that the nodes are coupled on the second and third variables thereby $J_h(s) = [0,0,0;\ 0.1,0;\ 0,0,1]^T$. As can be seen from

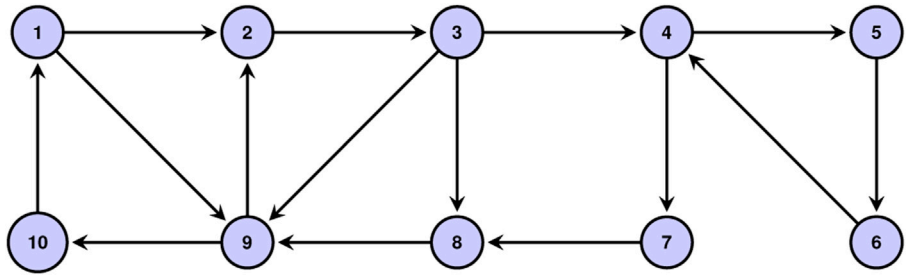


FIGURE 4
Network topology.

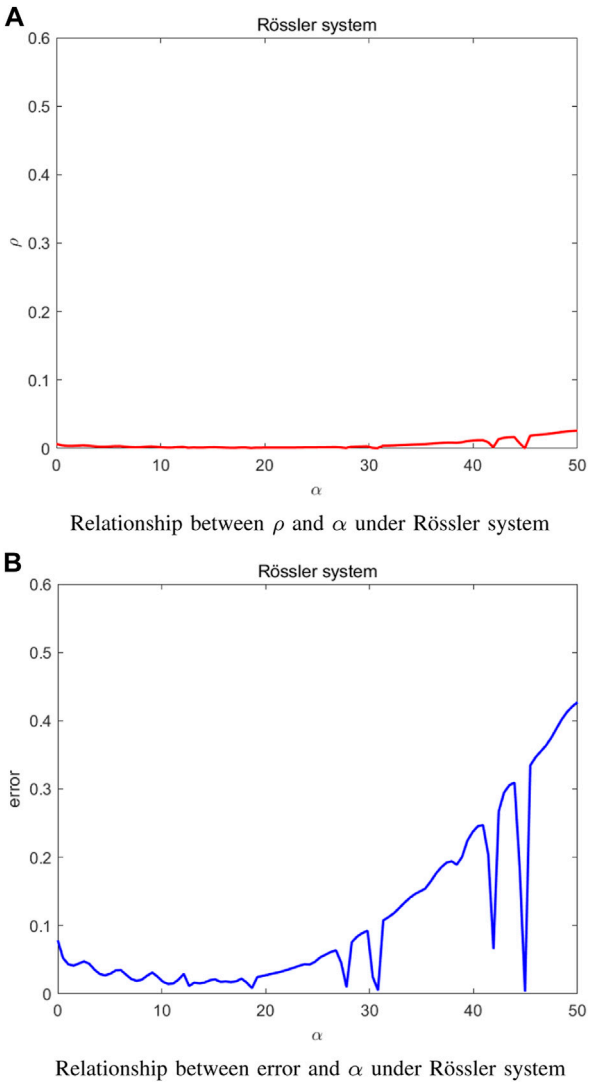


FIGURE 5
The results are obtained under Rössler systems when a general directed graph is considered.

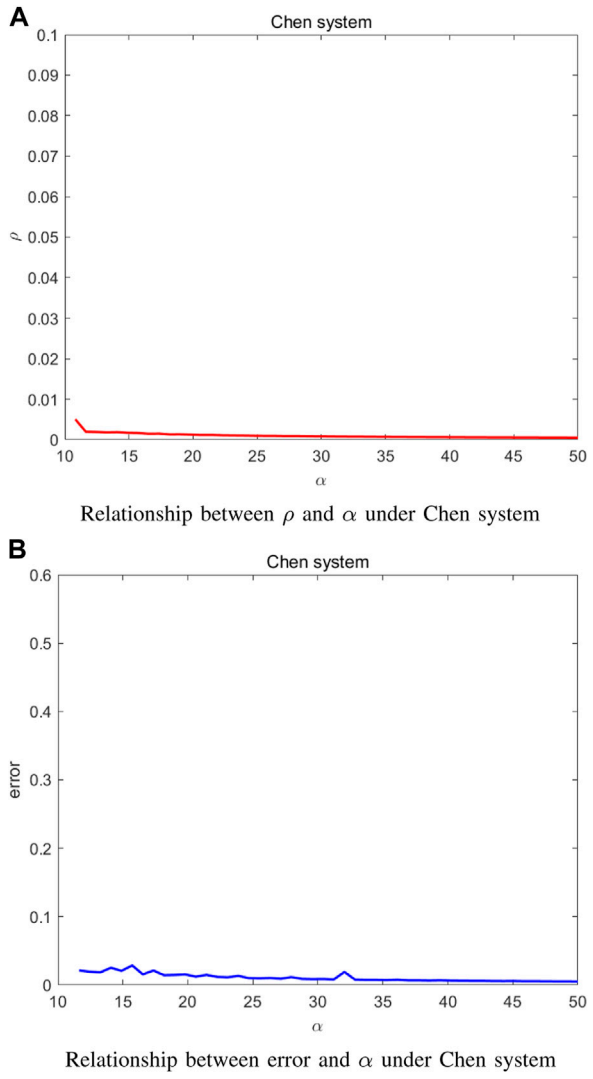


FIGURE 6
The results are obtained under Chen systems when a general directed graph is considered.

Figure 1A, the robustness metric ρ first continuously oscillates and then converges to around zero with the increase of the coupling strength α. As shown in Figure 1B, the synchronization error exhibits a similar behavior.

Figure 2 shows the simulation results for the network of Chen systems. Assume that the nodes are coupled through all state variables thereby $J_h(s) = [1,0,0;0,1,0;0,0,1]^T$. It is interesting to see from

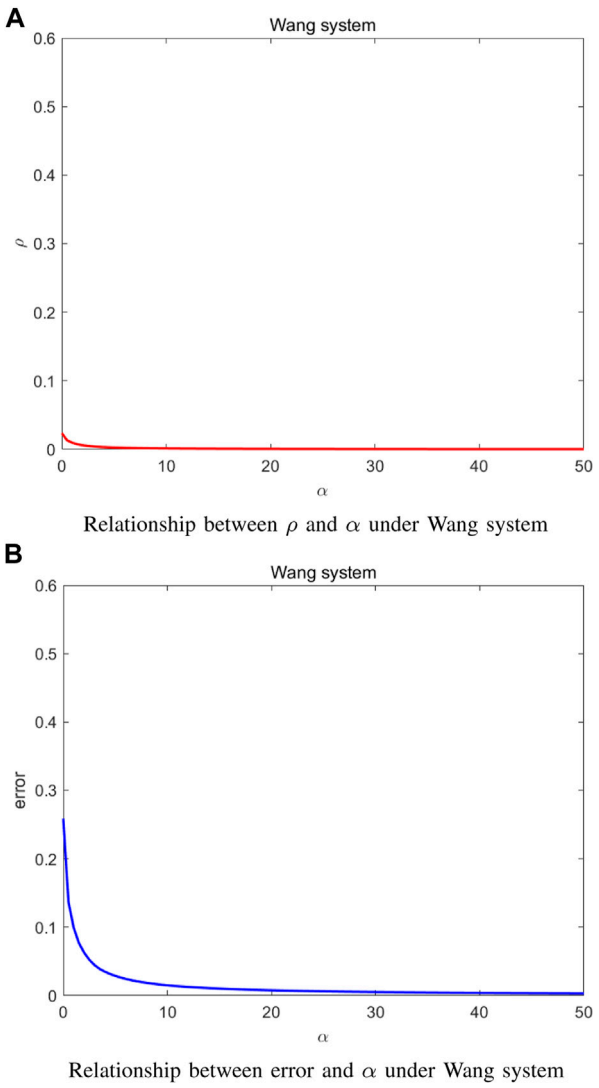


FIGURE 7
The results are obtained under Wang systems when a general directed graph is considered.

Figure 2A that, after a sharp fall, the robustness metric ρ begins to continuously oscillate and then converges to zero with the increase of the coupling strength α. Figure 2B shows that the synchronization error exhibits a similar behavior of the robustness metric.

Figure 3 illustrates the results for the network of Wang systems. Assume that the nodes are coupled on the first and third variables thereby $J_h(s) = [1,0,0; 0,0,0; 0,0,1]^T$. It follows from Figure 3A that the robustness metric first sharply decreases and gradually converges to zero with the increase of the coupling strength. Also, the synchronization error approaches to zero when the coupling strength increases.

Recall that the networked system synchronizes before the noise is introduced. This means that the choice of $J_h(s)$ in the simulation section can ensure the existence of the synchronized region.

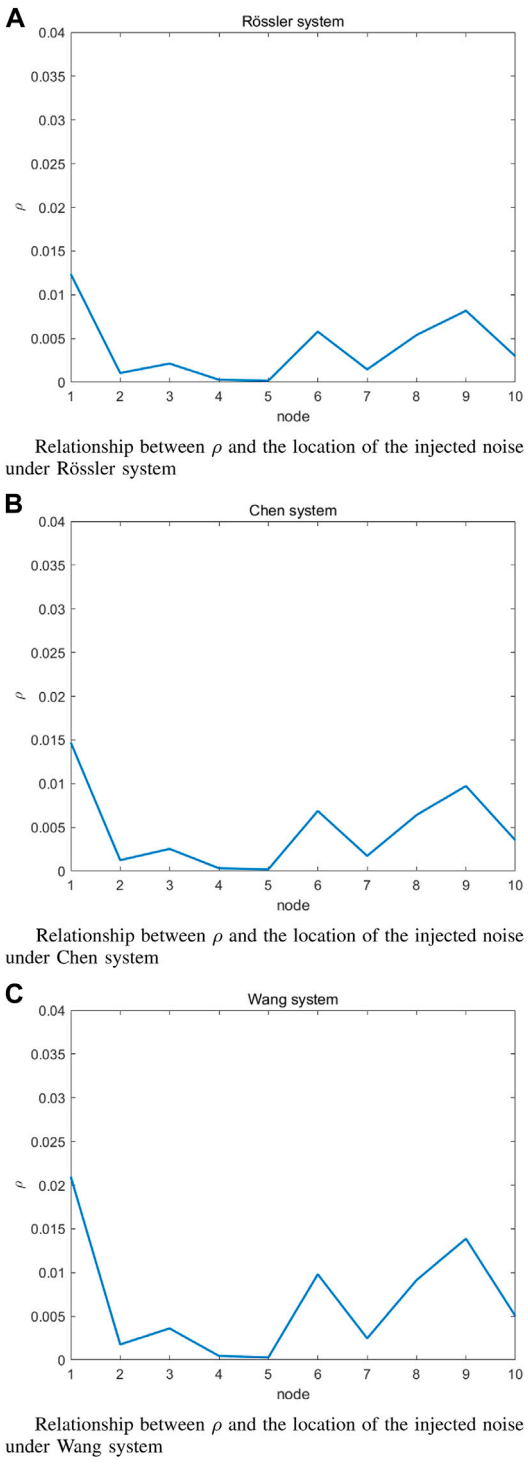


FIGURE 8
The main results are obtained under a general directed graph: $\alpha = 20$.

4.3 Dynamical robustness beyond general digraphs

In this subsection, consider a general directed graph [42] as shown in Figure 4. Assume that noise is injected into the fourth node, so the

element of v is one into the fourth row and others are 0. In addition, $\theta = 1$ and $H_\eta = [0, 0, 1]^T$.

Figures 5A, B are obtained under Rössler systems with $J_h(s) = [0,0,0;\ 0.1,0;\ 0,0,1]^T$. Notice that a much more complex scenario emerges. The robustness metric increases rapidly with intermittent descent when the coupling strength is larger than 19. From Figure 5B, the synchronization error decreases after continuously oscillating. When the coupling strength is larger than 19, the synchronization error begins to increase rapidly with intermittent descent.

Figures 6A, B are obtained under Chen systems with $J_h(s) = [1,0,0;\ 0.1,0;\ 0,0,1]^T$. After a sharp fall, the robustness metric is monotonically decreasing. The synchronization error exhibits a similar behavior to the robustness metric.

Figures 7A, B are obtained under Wang systems with $J_h(s) = [1,0,0;\ 0.0,0;\ 0,0,1]^T$. When the coupling strength gradually increases, the curves in Figure 7A and Figure 7B both converge to zero, which illustrates that the network of Wang systems is robust to noise and can reach the synchronization when the coupling strength is enough large.

In summary, the 10-node network of Rössler systems shows greatly different robustness from other systems.

In order to illustrate the effect of the location of the injected noise, the robustness metrics are examined in Figure 8 for Rössler system, Chen system, and Wang system, respectively, all with $\alpha = 20$. Recall that a smaller value of ρ implies a more dynamically robust network. It is clear that, for Rössler system, Chen system, and Wang system, the network shows a similar robustness. The results shown in Figure 8 provide guidance for minimizing the effect of noise on the network robustness. For example, the network is more robust to noise if node five is subjected to noise. Therefore, node five is the best choice to minimize the effect of noise from the dynamical robustness perspective.

5 Conclusion

This article investigates the dynamical robustness of a directed network with noise. A novel robustness metric is formulated and analyzed under the framework of mean-square stochastic stability. It is found that the dynamical robustness of the directed network is determined by both the node dynamics and the network topology. Particularly, for networks of Rössler systems, with the increase of the coupling strength, different network topologies show different effects on the robustness metric. While for Wang systems, the robustness to noise is stronger than other systems. These findings demonstrate that node dynamics plays an important role in the network robustness. The results of this study can provide theoretical and technical guidances for designing a dynamically robust networked system.

In the future, it will be interesting to investigate dynamical robustness of higher-order networks [37, 43]. Moreover, it will be interesting to investigate the effects of different types of noise on the network robustness. The dynamical robustness of stochastic complex networks with time-delays [44] or heterogeneous node dynamics [45] are challenging but also worthy of further investigation.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

All authors designed and did the research. JS and LX did the analytical and numerical calculations. JS, LX, and GC were the lead writer of the manuscript. All authors read and approved the final manuscript.

Funding

This work was supported by the National Natural Science Foundation of China (No. 61973064), Natural Science Foundation of Hebei Province of China (No. F2022501024), Hebei Provincial Postgraduate Student Innovation Ability Training Funding Project (No. CXZZSS2023202), and Hong Kong Research Grants Council under the GRF Grant CityU11206320.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Wu J, Yu X, Li X. Global frequency synchronization of complex power networks via coordinating switching control. *IEEE Trans Circuits Syst Regular Pap* (2019) 66:3123–33. doi:10.1109/tcsi.2019.2908085

2. Mingotti A, Peretto L, Tinarelli R. Accuracy evaluation of an equivalent synchronization method for assessing the time reference in power networks. *IEEE Trans Instrumentation Meas* (2018) 67:600–6. doi:10.1109/tim.2017.2779328

3. Yu F, Shen H, Yu Q, Kong X, Kumar Sharma P, Cai S. Privacy protection of medical data based on multi-scroll memristive hopfield neural network. *IEEE Trans Netw Sci Eng* (2022) 1–14:1–14. doi:10.1109/tnse.2022.3223930

4. Wu J, Liu M, Sun H, Li T, Gao Z, Wang DZ. Equity-based timetable synchronization optimization in urban subway network. *Transportation Res C: Emerging Tech* (2015) 51: 1–18. doi:10.1016/j.trc.2014.11.001

5. Liu T, Ceder A. Synchronization of public transport timetabling with multiple vehicle types. *Transportation Res Rec* (2016) 2539:84–93. doi:10.3141/2539-10

6. Buldyrev SV, Parshani R, Paul G, Stanley HE, Havlin S. Catastrophic cascade of failures in interdependent networks. *Nature* (2010) 464:1025–8. doi:10.1038/nature08932

7. Zhang L, Xiang L, Zhu J. Relationship between fragility and resilience in complex networks. *Physica A* (2022) 605:128039. doi:10.1016/j.physa.2022.128039

8. Mao B, Wu X, Lü J, Chen G. Predefined-time bounded consensus of multiagent systems with unknown nonlinearity via distributed adaptive fuzzy control. *IEEE Trans Cybernetics* (2022) 1–14. doi:10.1109/tcyb.2022.3163755

9. Zhu Y, Xia C-Y, Wang Z, Chen Z. Networked decision-making dynamics based on fair, extortionate and generous strategies in iterated public goods games. *IEEE Trans Netw Sci Eng* (2022) 9(4):2450–62. doi:10.1109/tnse.2022.3164094

10. Lou Y, Wu R, Li J, Wang L, Tang C-B, Chen G. Classification-based prediction of network connectivity robustness. *Neural Networks* (2023) 157:136–46. doi:10.1016/j.neunet.2022.10.013

11. Callaway DS, Newman ME, Strogatz SH, Watts DJ. Network robustness and fragility: Percolation on random graphs. *Phys Rev Lett* (2000) 85(25):5468–71. doi:10.1103/physrevlett.85.5468

12. Wang X, Pournaras E, Kooij RE, Mieghem PV. Improving robustness of complex networks via the effective graph resistance. *The Eur Phys J B* (2014) 87:221–12. doi:10.1140/epjb/e2014-50276-0

13. Liu X, Sun S, Wang J, Xia C. Onion structure optimizes attack robustness of interdependent networks. *Physica A* (2019) 535:122374. doi:10.1016/j.physa.2019.122374

14. Liao L, Shen Y, Liao J. Robustness of dispersal network structure to patch loss. *Ecol Model* (2020) 424:109036. doi:10.1016/j.ecolmodel.2020.109036

15. Huang D, Bian J, Jiang H, Yu Z. Consensus indices of two-layered multi-star networks: An application of Laplacian spectrum. *Front Phys* (2021) 9:803941. doi:10.3389/fphy.2021.803941

16. Liu X, Li D, Ma M, Szymanski BK, Stanley HE, Gao J. Network resilience. *Phys Rep* (2022) 971:1–108. doi:10.1016/j.physrep.2022.04.002

17. Nie S, Wang X-W, Zhang H, Li Q, Wang B. Robustness of controllability for networks based on edge-attack. *PLoS ONE* (2014) 9:e89066. doi:10.1371/journal.pone.0089066

18. Shargel B, Sayama H, Epstein IR, Bar-Yam Y. Optimization of robustness and connectivity in complex networks. *Phys Rev Lett* (2003) 90(6):068701. doi:10.1103/physrevlett.90.068701

19. Schwartz N, Cohen R, Ben-Avraham D, Barabási A-L, Havlin S. Percolation in directed scale-free networks. *Phys Rev E* (2002) 66(1):015104. doi:10.1103/physreve.66.015104

20. Albert R, Jeong H, Barabási A-L. Error and attack tolerance of complex networks. *Nature* (2000) 406(6794):378–82. doi:10.1038/35019019

21. Holme P, Kim BJ, Yoon CN, Han SK. Attack vulnerability of complex networks. *Phys Rev E* (2002) 65(5):056109. doi:10.1103/physreve.65.056109

22. Cohen R, Erez K, Ben-Avraham D, Havlin S. Breakdown of the internet under intentional attack. *Phys Rev Lett* (2001) 86(16):3682–5. doi:10.1103/physrevlett.86.3682

23. Cohen R, Erez K, Ben-Avraham D, Havlin S. Resilience of the internet to random breakdowns. *Phys Rev Lett* (2000) 85:4626–8. doi:10.1103/physrevlett.85.4626

24. Nagata S, Kikuchi M. Emergence of cooperative bistability and robustness of gene regulatory networks. *PLoS Comput Biol* (2020) 16(6):e1007969. doi:10.1371/journal.pcbi.1007969

25. Faci-Lázaro S, Lor T, Ródenas G, Mazo JJ, Soriano J, Gómez-Gardeñes J. Dynamical robustness of collective neuronal activity upon targeted damage in interdependent networks. *Eur Phys J Spec Top* (2022) 231(3):195–201. doi:10.1140/epjs/s11734-021-00411-7

26. Franci A, O'Leary T, Golowasch J. Positive dynamical networks in neuronal regulation: How tunable variability coexists with robustness. *IEEE Control Syst Lett* (2020) 4(4):946–51. doi:10.1109/lcsys.2020.2997214

27. Tanaka G, Morino K, Aihara K. Dynamical robustness in complex networks: The crucial role of low-degree nodes. *Scientific Rep* (2012) 2(1):232–6. doi:10.1038/srep00232

28. Buscarino A, Gambuzza LV, Porfiri M, Fortuna L, Frasca M. Robustness to noise in synchronization of complex networks. *Scientific Rep* (2013) 3(1):2026–6. doi:10.1038/srep02026

29. Porfiri M, Frasca M. Robustness of synchronization to additive noise: How vulnerability depends on dynamics. *IEEE Trans Control Netw Syst* (2019) 6(1):375–87. doi:10.1109/tcns.2018.2825024

30. Chen F, Xiang L, Lan W, Chen G. Coordinated tracking in mean square for a multi-agent system with noisy channels and switching directed network topologies. *IEEE Trans Circuits Systems-II: Express Briefs* (2012) 59(11):835–9. doi:10.1109/tcsii.2012.2218395

31. Hu J, Wu Y, Li T, Ghosh BK. Consensus control of general linear multiagent systems with antagonistic interactions and communication noises. *IEEE Trans Automatic Control* (2019) 64(5):2122–7. doi:10.1109/tac.2018.2872197

32. Meng JH, Riecke H. Synchronization by uncorrelated noise: Interacting rhythms in interconnected oscillator networks. *Scientific Rep* (2018) 8:6949. doi:10.1038/s41598-018-24670-y

33. Zhang C, Yang Y. Synchronization of stochastic multi-weighted complex networks with lévy noise based on graph theory. *Physica A* (2020) 545:123496. doi:10.1016/j.physa.2019.123496

34. Sun Y, Zhao D. Effects of noise on the outer synchronization of two unidirectionally coupled complex dynamical networks. *Chaos* (2012) 22(2):023131. doi:10.1063/1.4721997

35. Cardelli L, Csikász-Nagy A, Dalchau N, Tribastone M, Tschaikowski M. Noise reduction in complex biological switches. *Scientific Rep* (2016) 6(1):20214–2. doi:10.1038/srep20214

36. Sun S, Chen F, Ren W. Distributed average tracking in weight-unbalanced directed networks. *IEEE Trans Automatic Control* (2020) 66(9):4436–43. doi:10.1109/tac.2020.3046029

37. Chen G. Searching for best network topologies with optimal synchronizability: A brief review. *IEEE/CAA J Automatica Sinica* (2022) 9(4):573–7. doi:10.1109/jas.2022.105443

38. Young GF, Scardovi L, Leonard NE. Robustness of noisy consensus dynamics with directed communication. In: *Proceedings of the American Control conference* (2010). p. 6312–7.

39. Rössler OE. An equation for continuous chaos. *Phys Lett A* (1976) 57:397–8. doi:10.1016/0375-9601(76)90101-8

40. Chen G, Ueta T. Yet another chaotic attractor. *J Bifurcation Chaos* (1999) 9(7):1465–6. doi:10.1142/s0218127499001024

41. Wang X, Chen G. A chaotic system with only one stable equilibrium. *Commun Nonlinear Sci Numer Simulation* (2011) 17:1264–72. doi:10.1016/j.cnsns.2011.07.017

42. Song Q, Cao J. On pinning synchronization of directed and undirected complex dynamical networks. *IEEE Trans Circuits Syst Regular Pap* (2010) 57:672–80. doi:10.1109/tcsi.2009.2024971

43. Shi D, Chen G. Simplicial networks: A powerful tool for characterizing higher-order interactions. *Natl Sci Rev* (2022) 9(5):nwac038–2. doi:10.1093/nsr/nwac038

44. Chen F, Chen X, Xiang L, Ren W. Distributed economic dispatch via a predictive scheme: Heterogeneous delays and privacy preservation. *Automatica* (2021) 123:109356. doi:10.1016/j.automatica.2020.109356

45. Chen F, Feng G, Liu L, Ren W. Distributed average tracking of networked Euler-Lagrange systems. *IEEE Trans Automatic Control* (2015) 60(2):547–52. doi:10.1109/tac.2014.2343111

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 30 January 2023
DOI 10.3389/fphy.2023.1127884

OPEN ACCESS

EDITED BY
Fei Yu,
Changsha University of Science and Technology, China

REVIEWED BY
Fei Tan,
Xiangtan University, China
Ciyan Zheng,
Guangdong Polytechnic Normal University, China

*CORRESPONDENCE
Chunli Zhang,
gaozhangchunli@163.com

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 20 December 2022
ACCEPTED 06 January 2023
PUBLISHED 30 January 2023

CITATION
Zhang C, Yan L, Gao Y, Wang W, Li K, Wang D and Zhang L (2023), A new adaptive iterative learning control of finite-time hybrid function projective synchronization for unknown time-varying chaotic systems.
Front. Phys. 11:1127884.
doi: 10.3389/fphy.2023.1127884

A new adaptive iterative learning control of finite-time hybrid function projective synchronization for unknown time-varying chaotic systems

Chunli Zhang[1]*, Lei Yan[1], Yangjie Gao[1], Wenqing Wang[1], Keming Li[2], Duo Wang[2] and Long Zhang[2]

[1]Xi'an University of Technology, Shaanxi Key Laboratory of Complex System Control and Intelligent Information Processing, Xi'an, China, [2]State Grid Xi'an Electric Power Supply Company, Xi'an, China

A new adaptive iterative learning control (AILC) scheme is proposed to solve the finite-time hybrid function projective synchronization (HFPS) problem of chaotic systems with unknown periodic time-varying parameters. Fourier series expansion (FSE) is introduced to deal with the problem of uncertain time-varying parameters. The bound of the expanded remaining items is unknown. A typical convergent series is used to deal with the unknown bound in the design process of the controller. The adaptive iterative learning synchronization controller and parameter update laws are designed. Two different chaotic systems are synchronized asymptotically according to different proportional functions on a finite time interval by Lyapunov stability analysis. The simulation example proves the feasibility and effectiveness of the proposed method.

KEYWORDS

hybrid function projective synchronization, chaotic systems, adaptive iterative learning control, Fourier series expansion, finite time

Introduction

Chaos synchronization has a wide range of applications in physical systems, biological networks, and secure communications [1]. However, due to the unpredictability, pseudo-randomness, and extreme sensitivity to initial values of chaotic systems [2], hybrid function projection synchronization of chaotic systems is widely concerned. In 1990, the pioneer workers Pecora and Carroll first proposed the concept of chaos synchronization and solved the synchronization problem of two identical chaotic systems under different initial conditions [3]. Since then, chaos synchronization control has become a very active research topic [4–16], for example, projective synchronization control and hybrid function projective synchronization control. Many control methods have been used to synchronize different chaotic systems, such as feedback linearization method [17], optimal control [18], and neural network control [19–24]. However, few people study the problem of chaos synchronization on a finite time interval.

When studying chaos synchronization, the problem of system parameter uncertainty will be encountered. The adaptive control method is often used to solve this problem and improve control performance [25–27]. Chaotic systems are vulnerable to encounter the problem of uncertain parameters due to external interference. These uncertain parameters may disrupt synchronization. Therefore, this is an important problem to study the synchronization of chaotic systems with unknown time-varying parameters.

The adaptive iterative learning control method can solve the finite-time tracking control problem of uncertain non-linear systems [28] by modifying the input information according to the previous output data. This ensures that the output of the system tracks the given trajectory on a finite time interval accurately. This is one of the most effective control methods of the non-linear systems' repetitive tracking control. Because synchronization is similar to tracking, the synchronization problem of chaotic systems on a finite time interval can be solved by adaptive iterative learning control [29, 30]. Achieving finite-time hybrid function projection synchronization control of chaotic systems with time-varying parameters is an interesting topic.

This paper proposes a new adaptive iterative learning control method to solve the HFPS problem of chaotic systems with uncertain periodic time-varying parameters. FSE is introduced to deal with the problem of uncertain parameters. Convergent series is used to deal with the unknown upper bounds of uncertain parameters. The parameter updating laws and AILC law are designed to synchronize the states of two chaotic systems according to different proportion functions asymptotically. The simulation example illustrates the correctness and effectiveness of the research results in this paper.

The main contributions of this paper are listed as follows:

(1) The problem of chaos synchronization on a limited time interval is very important. We proposed an adaptive iterative learning control scheme to solve the finite-time hybrid function projective synchronization of chaotic systems with unknown periodic time-varying parameters.
(2) The problem of system parameter uncertainty of chaos synchronization must be solved. Fourier series expansion (FSE) is introduced to solve the problem of time-varying parameters.
(3) The residual term after expansion is bounded, but the bound is unknown. Based on the author's previous work, a typical convergent series is used to deal with the remaining items after expansion.

System specification and synchronization controller design

System specification

Consider the following continuous time chaotic systems with unknown time-varying parameters:

$$\dot{x}_k = Ax_k + f(x_k) + D(x_k)M(t), \tag{1}$$

where $x_k \in \mathrm{R}^n$ is the state. $A \in \mathrm{R}^{n\times n}$ is the coefficient matrix which is linear. $f(x_k)$: $\mathrm{R}^n \to \mathrm{R}^n$is the non-linear part. $D(x_k)$: $\mathrm{R}^n \to \mathrm{R}^{n\times d}$, $M(t) = M + \Delta M(t) = [m_1 + \Delta m_1(t), \cdots, m_d + \Delta m_d(t)]^T \in \mathrm{R}^d$ is the uncertain parameter vector. k is the number of iterations. Here, M is the nominal value of $M(t)$ and$\Delta M(t)$ is a time-varying parameter. System (1) is a driving system, and the response system containing controller $u_k(t) \in \mathrm{R}^n$ is shown in system (2):

$$\dot{y}_k = By_k + g(y_k) + u_k(t), \tag{2}$$

where $y_k \in \mathrm{R}^n$ is the state. $B \in \mathrm{R}^{n\times n}$ is the matrix. $g(y_k)$: $\mathrm{R}^n \to \mathrm{R}^n$ is the non-linear function of (2) which is continuous. $u_k(t) \in \mathrm{R}^n$ is the control vector. Here, suppose $M(t) \in \mathrm{R}^d$ is an uncertain function with a known period T, then $M(t) = M(t-T)$. This assumption is reasonable because many things in nature occur periodically, so the parameters of the system often exist periodically.

According to the Fourier series expansion, the continuous period vector $\Delta \boldsymbol{m_i}(t)$, $i = 1, \cdots, d$ can be expressed as shown in (3):

$$\Delta m_i(t) = \Xi_i^T(t)\Theta_i + \sigma_i(t), \ |\sigma_i(t)| \le \bar{\sigma}_i, \tag{3}$$

where $\Theta_i = [\varphi_{i1}, \varphi_{i2}, \cdots, \varphi_{iq}]^T \in \mathrm{R}^{q\times d}$, $\varphi_{ij} \in \mathrm{R}^q$, $j = 1, \cdots, q$ is composed of the first q parameters in the FSE of $m_i(t)$. $\sigma_i(t)$ is the truncation error and $\bar{\sigma}_i > 0$. $\Xi_i(t) = [\varphi_{i1}(t), \cdots, \varphi_{iq}(t)]^T$, where $\varphi_{i1}(t) = 1$, $\varphi_{i(2j)}(t) = \sin(\frac{2\pi jt}{T})$, $\varphi_{i(2j+1)}(t) = \cos(\frac{2\pi jt}{T})$, $j = 1, \cdots, \frac{q-1}{2}$, and their nth derivative are smooth and bounded. Take $\boldsymbol{\Theta} = [\Theta_1, \cdots, \Theta_d]^T$,$\Xi(t) = \mathrm{diag}\{\Xi_1(t), \cdots, \Xi_d(t)\}$, $\sigma(t) = [\sigma_1(t), \cdots, \sigma_d(t)]^T$, and $\bar{\sigma} = [\bar{\sigma}_1, \cdots, \bar{\sigma}_d]^T$. Then, by substituting (3) into system (1), system (1) can be rewritten as follows:

$$\dot{x}_k = Ax_k + f(x_k) + D(x_k)\left(M + \Xi^T(t)\Theta + \sigma(t)\right). \tag{4}$$

According to (3), $\|\sigma(t)\| \le \|\bar{\sigma}\| = s$, suppose the upper bound s is an unknown parameter.

Define the synchronization error as

$$e_k(t) = x_k - H(t)y_k. \tag{5}$$

Here, $H(t) = diag\{h_1(t), h_2(t), \cdots h_n(t)\}$ is the scale matrix and $h_i(t), i = 1, 2, \cdots, n$ is a bounded non-zero continuous differentiable function.

The control objective of this paper is to design an appropriate controller $u_k(t)$ and related parameter adaptive laws for response system 2) on the finite time $[0, T]$, so that response system 2) can synchronize with drive system 1) using different scale functions as $k \to \infty$ asymptotically. That is to say, $\lim_{k\to\infty} e_k(t) = 0$.

Design of the adaptive iterative learning synchronization controller

In the process of controller design, the following definition and lemma of the convergence series sequence will be used.

Definition. 1[31]: The sequence $\{Z_k\}$ is defined as

$$Z_k = \frac{b}{k^l}. \tag{6}$$

This sequence is a convergence series. Here, $k = 1, 2, \cdots$; b and l are parameters to be designed, $b > 0 \in \mathbf{R}$, $l \ge 2 \in N$.

Lemma. 1 [31]: For the sequence $\left\{\frac{1}{k^l}\right\}$, $k = 1, 2, \cdots, l \ge 2$, the following conclusion holds:

$$\lim_{k\to\infty} \sum_{j=1}^{k} \frac{1}{j^l} \le 2. \tag{7}$$

The specific design process of the controller is as follows:
From system (4) and system (2), it is easy to get (8)

$$\begin{aligned}\dot{e}_k = {} & Ax_k + f(x_k) + D(x_k)\Xi^T(t)\Theta + D(x_k)(M + \sigma(t)) \\ & -H(t)By_k - H(t)g(y_k) - H(t)u_k(t).\end{aligned} \tag{8}$$

From (8), the following controller is designed:

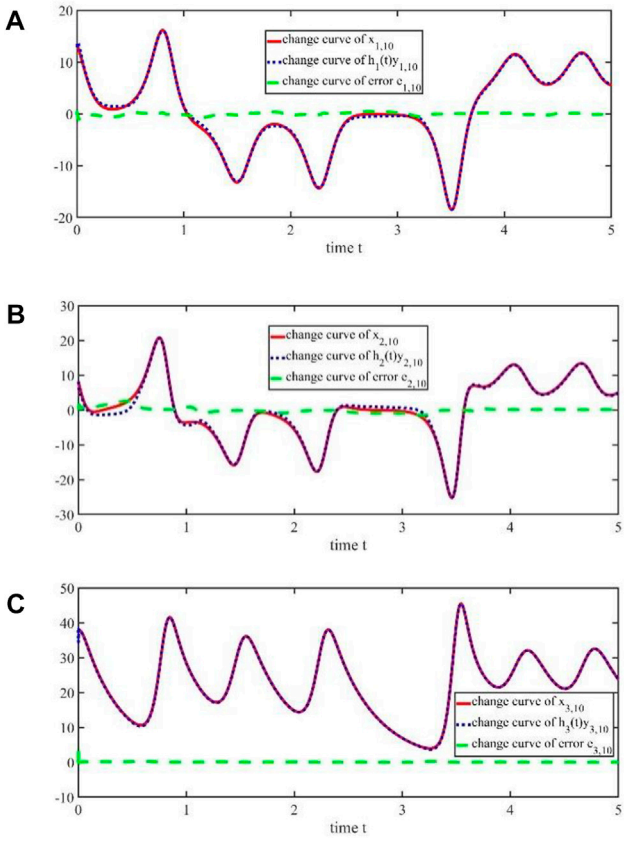


FIGURE 1
(A) Change of the 10th iteration trajectories $x_{1,10}(t), h_1(t)y_{1,10}(t)$ and error $e_{1,10}(t)$ on time t; **(B)** Change of the 10th iteration trajectories $x_{2,10}(t), h_2(t)y_{2,10}(t)$ and error $e_{2,10}(t)$ on time t; **(C)** Change of the 10th iteration trajectories $x_{3,10}(t), h_3(t)y_{3,10}(t)$ and error $e_{3,10}(t)$ on time t.

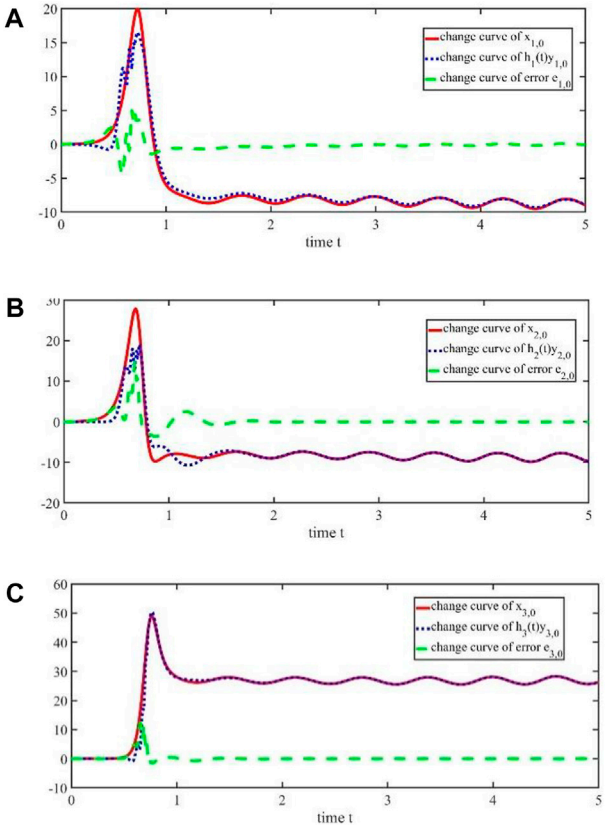


FIGURE 2
(A) Change of trajectories $x_{1,0}(t), h_1(t)y_{1,0}(t)$ and error $e_{1,0}(t)$ without iteration on time t; **(B)** Change of trajectories $x_{2,0}(t), h_2(t)y_{2,0}(t)$ and error $e_{2,0}(t)$ without iteration on time t; **(C)** Change of trajectories $x_{3,0}(t), h_3(t)y_{3,0}(t)$ and error $e_{3,0}(t)$ without iteration on time t.

$$u_k(t) = H^{-1}(t)\left[Ke_k + Ax_k + f(x_k) + D(x_k)\Xi^T(t)\hat{\Theta}_k + D(x_k)\hat{M}_k\right] + \frac{1}{Z_k}D(x_k)D^T(x_k)e_k\hat{S}_k - H(t)By_k - H(t)g(y_k). \tag{9}$$

Here, $K \in R^{n\times n}$ is a positive feedback gain matrix, $S = s^2$. $\hat{\Theta}_k$, $\hat{M}_k$, and $\hat{S}_k$ are estimates of Θ, M, and S, respectively.

Select the following parameter update laws:

$$\begin{aligned}\dot{\hat{\Theta}}_k &= \Gamma_1\Xi(t)D^T(x_k)e_k,\\ \dot{\hat{M}}_k &= \Gamma_2 D^T(x_k)e_k,\\ \dot{\hat{S}}_k &= \Gamma_3\frac{1}{Z_k}e_k^T D(x_k)D^T(x_k)e_k,\end{aligned} \tag{10}$$

where $\Gamma_i, i = 1, 2, 3$ is a positive definite diagonal gain matrix of the appropriate dimension.

The initial conditions satisfy the following relations: for any k, when $t = 0$, $x_k(0) = H(t)y_k(0), \hat{\Theta}_k(0) = \hat{\Theta}_{k-1}(T), \hat{M}_k(0) = \hat{M}_{k-1}(T), \hat{S}_k(0) = \hat{S}_{k-1}(T)$.

Stability analysis

According to the aforementioned controller design process, the following theorem is given in this paper:

Theorem 1. For the given scaling function $H(t)$, designing the control law 9) and parameter update laws (10) for response system (2) can obtain that all closed-loop signals are bounded on $[0, T]$, and $\lim_{k\to\infty} e_k(t) = 0$.

Proof. For convergence analysis, the following Lyapunov functions are selected:

$$V_k(t) = \frac{1}{2}e_k^T e_k + \frac{1}{2}\tilde{\Theta}_k^T\Gamma_1^{-1}\tilde{\Theta}_k + \frac{1}{2}\tilde{M}_k^T\Gamma_2^{-1}\tilde{M}_k + \frac{1}{2}\Gamma_3^{-1}\tilde{S}_k^2, \tag{11}$$

where $\tilde{\Theta}_k = \Theta - \hat{\Theta}_k, \tilde{M}_k = M - \hat{M}_k, \tilde{S}_k = S - \hat{S}_k$.

According to the assumptions, definition, and lemma, it is easy to prove that the conclusion of Theorem 1 is true.

The derivation process of $\dot{V}_k(t)$ according to system 8) is as follows:

$$\begin{aligned}\dot{V}_k(t) &= e_k^T e_k - \tilde{\Theta}_k^T\Gamma_1^{-1}\dot{\hat{\Theta}}_k - \tilde{M}_k^T\Gamma_2^{-1}\dot{\hat{M}}_k - \Gamma_3^{-1}\tilde{S}\dot{\hat{S}}_k\\ &= e_k^T\left(Ax_k + f(x_k) + D(x_k)\Xi^T(t)\Theta + D(x_k)(M + \sigma(t))\right.\\ &\quad \left. - H(t)By_k - H(t)g(y_k) - H(t)u_k(t)\right) - \tilde{\Theta}_k^T\Gamma_1^{-1}\dot{\hat{\Theta}}_k\\ &\quad - \tilde{M}_k^T\Gamma_2^{-1}\dot{\hat{M}}_k - \Gamma_3^{-1}\tilde{S}\dot{\hat{S}}_k \le e_k^T\left(Ax_k + f(x_k) + D(x_k)\Xi^T(t)\Theta\right.\\ &\quad \left. + D(x_k)M - H(t)By_k - H(t)g(y_k) - H(t)u_k(t)\right)\\ &\quad + \left\|e_k^T D(x_k)\right\|\|\sigma(t)\| - \tilde{\Theta}_k^T\Gamma_1^{-1}\dot{\hat{\Theta}}_k - \tilde{M}_k^T\Gamma_2^{-1}\dot{\hat{M}}_k - \Gamma_3^{-1}\tilde{S}\dot{\hat{S}}_k\end{aligned}$$

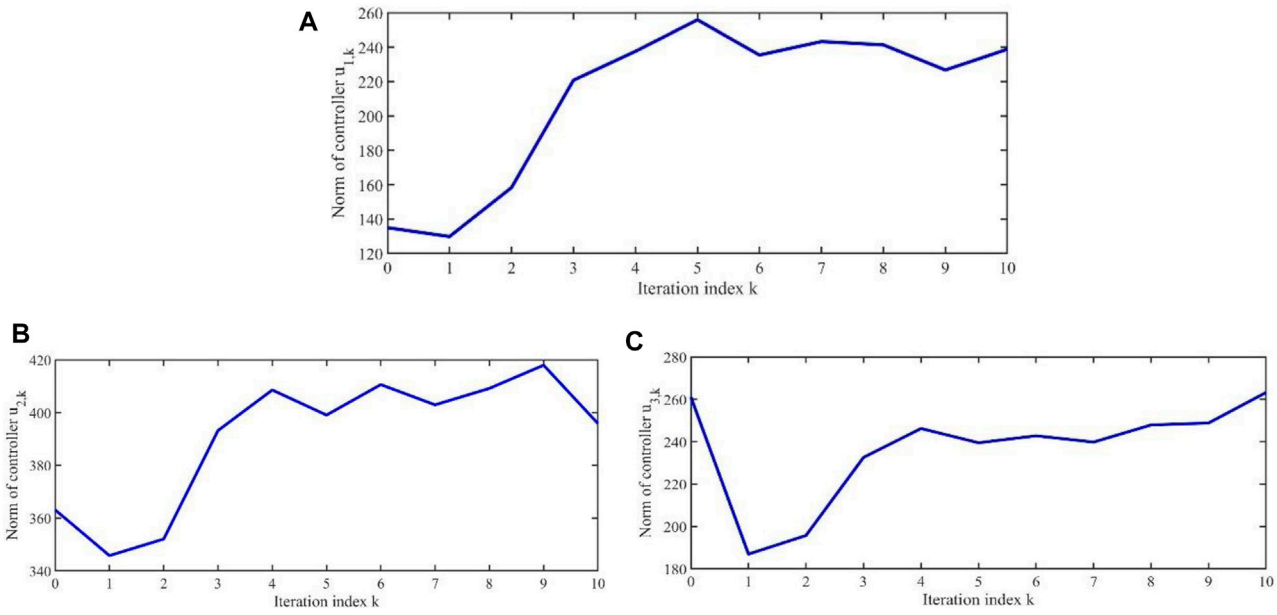


FIGURE 3
Norm variation curve of controller $\|u_{i,k}\|$ ($A.\ i = 1; B.\ i = 2; C.\ i = 3$) with iteration k.

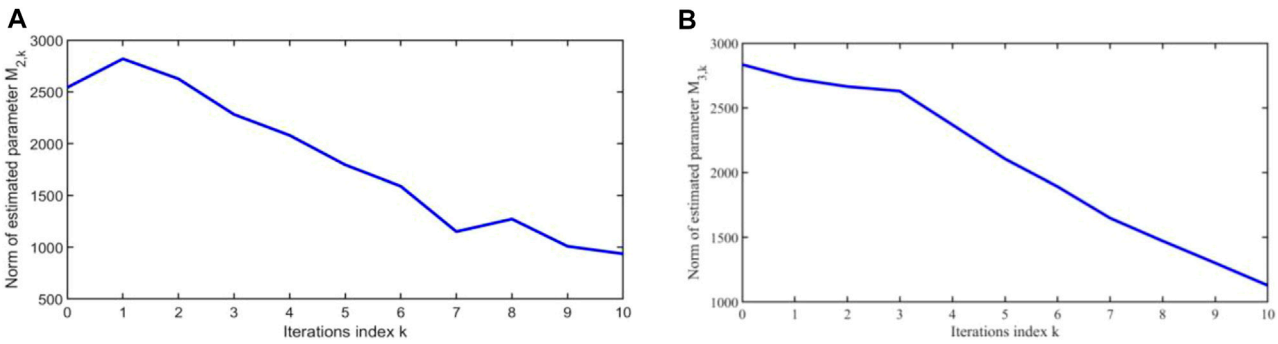


FIGURE 4
Norm variation curve of parameter $\|M_{i,k}\|$ ($A.\ i = 2; B.\ i = 3$) with iteration k.

$$
\begin{aligned}
&\le e_k^T\left(Ax_k + f(x_k) + D(x_k)\Xi^T(t)\Theta + D(x_k)M - H(t)By_k\right.\\
&\quad - H(t)g(y_k) - H(t)u_k(t)) + \left\|e_k^T D(x_k)\right\|s - \tilde{\Theta}_k^T\Gamma_1^{-1}\dot{\hat{\Theta}}_k - \tilde{M}_k^T\Gamma_2^{-1}\dot{\hat{M}}_k\\
&\quad - \Gamma_3^{-1}\tilde{S}\dot{\hat{S}}_k\\
&\le e_k^T\left(Ax_k + f(x_k) + D(x_k)\Xi^T(t)\Theta + D(x_k)M - H(t)By_k\right.\\
&\quad - H(t)g(y_k) - H(t)u_k(t)) + \frac{1}{Z_k}e_k^T D(x_k)D^T(x_k)e_k s^2 + \frac{1}{4}Z_k\\
&\quad - \tilde{\Theta}_k^T\Gamma_1^{-1}\dot{\hat{\Theta}}_k - \tilde{M}_k^T\Gamma_2^{-1}\dot{\hat{M}}_k - \Gamma_3^{-1}\tilde{S}\dot{\hat{S}}_k\\
&= e_k^T\left(Ax_k + f(x_k) + D(x_k)\Xi^T(t)\Theta + D(x_k)M - H(t)By_k\right.\\
&\quad - H(t)g(y_k) - H(t)u_k(t)) + \frac{1}{Z_k}e_k^T D(x_k)D^T(x_k)e_k S + \frac{1}{4}Z_k\\
&\quad - \tilde{\Theta}_k^T\Gamma_1^{-1}\dot{\hat{\Theta}}_k - \tilde{M}_k^T\Gamma_2^{-1}\dot{\hat{M}}_k - \Gamma_3^{-1}\tilde{S}\dot{\hat{S}}_k\\
&= e_k^T\left(Ax_k + f(x_k) + D(x_k)\Xi^T(t)\Theta + D(x_k)M + \frac{1}{Z_k}D(x_k)D^T(x_k)e_k S\right.\\
&\quad - H(t)By_k - H(t)g(y_k) - H(t)u_k(t)) + \frac{1}{4}Z_k - \tilde{\Theta}_k^T\Gamma_1^{-1}\dot{\hat{\Theta}}_k\\
&\quad - \tilde{M}_k^T\Gamma_2^{-1}\dot{\hat{M}}_k - \Gamma_3^{-1}\tilde{S}\dot{\hat{S}}_k.
\end{aligned}
\tag{12}
$$

Substituting (9) and (10) into (12), we get

$$
\dot{V}_k(t) \le -e_k^T K e_k + \frac{1}{4}Z_k, \tag{13}
$$

where $mn \le \frac{1}{r}m^2 + \frac{1}{4}n^2 r \quad (r = Z_k)$ for any $r > 0$.

According to the initial conditions, we have $e_k(0)^2 = 0 \le e_k(T)^2$. By (11), we get

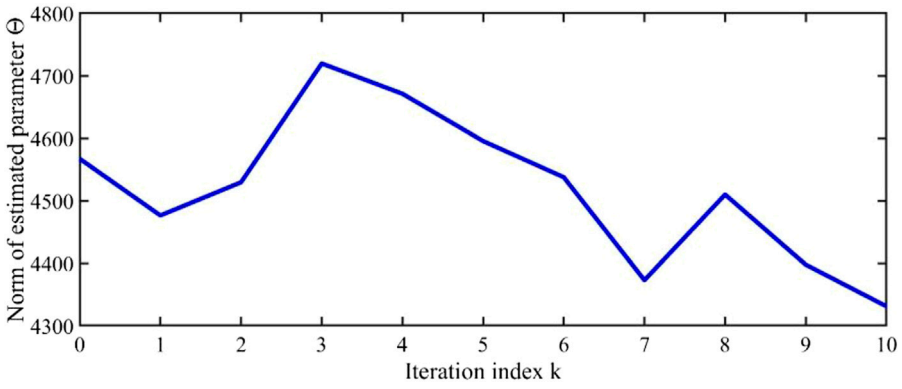


FIGURE 5
Variation curve of parameter $\|\hat{\Theta}_k\|$ with iteration k.

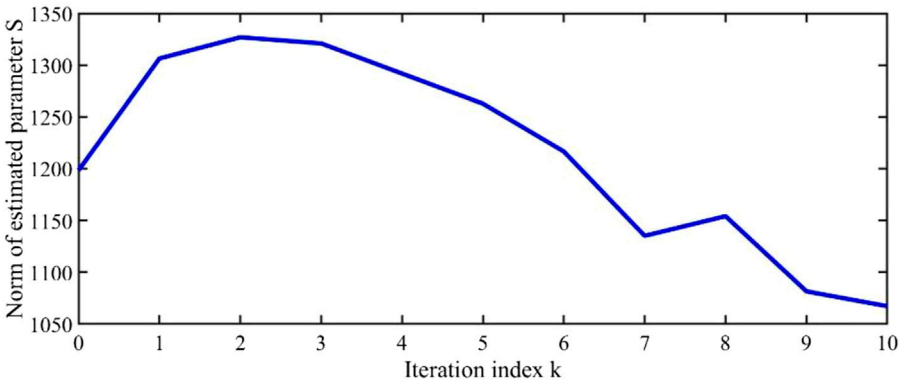


FIGURE 6
Variation curve of parameter $\|\hat{S}_k\|$ with iteration k.

$$V_k\left(e_k(0),\hat{\Theta}_k(T),\hat{M}_k(T),\hat{S}_k(T)\right)\le V_k\left(e_k(0),\hat{\Theta}_k(0),\hat{M}_k(0),\hat{S}_k(0)\right)+\int_0^T \dot{V}_k dt. \tag{14}$$

Substituting (12) into (14), we get

$$V_k\left(e_k(0),\hat{\Theta}_k(T),\hat{M}_k(T),\hat{S}_k(T)\right)\le V_1\left(e_1(0),\hat{\Theta}_1(0),\hat{M}_1(0),\hat{S}_1(0)\right)-\sum_{i=1}^{k}\int_0^T e_i^T K e_i dt+\frac{1}{4}T\sum_{i=1}^{k}Z_i \tag{15}$$

Taking $V_0(k)=V_1(e_1(0),\hat{\Theta}_1(0),\hat{M}_1(0),\hat{S}_1(0))+\frac{1}{4}T\sum_{i=1}^{k}Z_i$, formula (15) is rewritten as

$$\sum_{i=1}^{k}\int_0^T e_i^T K e_i dt\le V_0(k)-V_k\left(e_k(0),\hat{\Theta}_k(T),\hat{M}_k(T),\hat{S}_k(T)\right). \tag{16}$$

From (7), we get $\lim_{k\longrightarrow\infty} V_0(k)\le V_1+2b\frac{1}{4}T, V_0(k)$, which is bounded, and $V_k(e_k(0),\hat{\Theta}_k(T),\hat{M}_k(T),\hat{S}_k(T))\ge 0$. Therefore,

$$\lim_{k\longrightarrow\infty}\int_0^T e_k^T K e_k dt=0. \tag{17}$$

From (11), for any k, we have $V_k(t)=V_k(0)+\int_0^t \dot{V}_k(\tau)d\tau$. Substituting (12), we get

$$V_k(t)\le V_k(0)-\int_0^t e_k^T K e_k d\tau+t\frac{1}{4}Z_k. \tag{18}$$

By (17), $\int_0^t e_k^T K e_k d\tau$ is bounded. According to Definition 1, when $t\in[0,T]$, Z_k is bounded. So $t\frac{1}{4}Z_k$ is bounded and $\hat{\Theta}_k(0)=\hat{\Theta}_{k-1}(T)$, $\hat{M}_k(0)=\hat{M}_{k-1}(T)$, $\hat{S}_k(0)=\hat{S}_{k-1}(T)$. From (11), for any k, $V_k(0,\hat{\Theta}_k(T),\hat{M}_k(T),\hat{S}_k(T))$ is bounded; $V_k(0,\hat{\Theta}_k(0),\hat{M}_k(0),\hat{S}_k(0))=V_{k-1}(0,\hat{\Theta}_{k-1}(T),\hat{M}_{k-1}(T),\hat{S}_{k-1}(T))$ is bounded. When $V_k(t)$ is bounded, $\hat{\Theta}_k(T),\hat{M}_k(T),\hat{S}_k(T)$ is also bounded. According to 9), $u_k(t)$ is bounded. By (8), $\dot{e}_k$ is bounded, so e_k is uniformly continuous. According to the aforementioned parameters, we can draw the conclusion $\lim_{k\longrightarrow\infty} e_k(t)=0$.

Simulation analysis

Consider the hybrid function projection synchronization of the following two systems. The specific system is as follows:

The drive system is the Lorenz system:

$$\begin{aligned}\dot{x}_{1,k} &= m_1(t)(x_{2,k} - x_{1,k}),\\ \dot{x}_{2,k} &= -x_{1,k}x_{3,k} - x_{2,k} + m_2(t)x_{1,k},\\ \dot{x}_{3,k} &= x_{1,k}x_{2,k} - m_3(t)x_{3,k}.\end{aligned} \tag{19}$$

The response system is the Chen system:

$$\begin{aligned}\dot{y}_{1,k} &= 35(y_{2,k} - y_{1,k}) + u_{1,k},\\ \dot{y}_{2,k} &= -y_{1,k}y_{3,k} + 28y_{2,k} - 7y_{1,k} + u_{2,k},\\ \dot{y}_{3,k} &= y_{1,k}y_{2,k} - 3y_{3,k} + u_{3,k}.\end{aligned} \tag{20}$$

By comparing system (19) and system (20) with system (1) and system (2), we can get

$$\boldsymbol{A} = \begin{pmatrix} 0 & 0 & 0\\ 0 & -1 & 0\\ 0 & 0 & 0\end{pmatrix}, \boldsymbol{f}(\boldsymbol{x_k}) = \begin{pmatrix} 0\\ -x_{1,k}x_{3,k}\\ x_{1,k}x_{2,k}\end{pmatrix}, \boldsymbol{D}(\boldsymbol{x_k})$$
$$= \begin{pmatrix} x_{2,k} - x_{1,k} & 0 & 0\\ 0 & x_{1,k} & 0\\ 0 & 0 & -x_{3,k}\end{pmatrix},$$
$$\boldsymbol{B} = \begin{pmatrix} 0 & 0 & 0\\ 0 & -1 & 0\\ 0 & 0 & 0\end{pmatrix}, \boldsymbol{g}(\boldsymbol{x_k}) = \begin{pmatrix} 0\\ -y_{1,k}y_{3,k}\\ y_{1,k}y_{2,k}\end{pmatrix},$$

where the uncertain parameters are $m_1(t) = 10 + a_1 \sin t$, $m_2(t) = 28 + a_2 \cos t$, and $m_3(t) = \frac{8}{3} - a_3 \sin t$. During the simulation, the parameters are chosen as $T = 5$, $q = 5$, $l = 3$, $b = 100$, $\Gamma_1 = 0.0001I_5$, $\Gamma_1 = 0.1I_3$, and $\Gamma_3 = 1$. The proportional functions are chosen as $h_1(t) = 100 + 10\sin(\frac{2\pi t}{99})$, $h_2(t) = 100 - 10\cos(\frac{2\pi t}{99})$, and $h_3(t) = 100 + 10\sin(\frac{2\pi t}{99})$. The initial values of the drive system and response system are chosen as $\boldsymbol{x_0}(\boldsymbol{0}) = [0.1, 0.1, 0.1]^T$ and $\boldsymbol{y_0}(\boldsymbol{0}) = [0.0001, 0.0001, 0.0001]^T$, respectively.

According to controller 9) and adaptive iterative learning laws (10) given in this paper, the results of Figure 1, Figure 2, Figure 3, Figure 4, Figure 5, and Figure 6 can be obtained. It can be seen from Figure 1 and Figure 2 that there will be large errors between the drive signals $x_{i,0}(t)$ and the response signals $h_i(t)y_{i,0}(t)$, $i = 1, 2, 3$ (Figures 2A–C) when there is no iteration, but in the 10th iteration, their errors tend to zero (Figures 1A–C) basically. It can be seen that the synchronization errors decrease as the number of iterations increases. Therefore, this method can realize synchronization using different scale functions between two different chaotic systems, which proves the correctness and effectiveness of the proposed method.

At the same time, it can be seen from Figure 3 to Figure 6 that as the number of iterations increases, the controllers $u_{i,k}, i = 1, 2, 3$ and the estimated system parameters will remain within a certain range, which ensures the boundedness of system parameters $\hat{M}_k$, $\hat{\Theta}_k$, and $\hat{S}_k$. This also satisfies the content of the theorem.

Conclusion

The AILC method is proposed to solve the different proportional synchronization problems on finite time interval with uncertain parameters. The adaptive iterative learning controller and parameter update laws make the drive system and the response system synchronize along different scale functions asymptotically. The proposed controller solves the problem of hybrid function projection synchronization between the Lorenz system and Chen system successfully. The correctness of the method is verified by theory and numerical simulation. Further work is needed on how to realize the finite-time synchronization between hyper-chaotic systems with time-varying parameters or chaotic systems with different dimensions. As far as the author knows, these studies are very difficult and interesting topics in the field of chaos synchronization control.

Data availability statement

The raw data supporting the conclusion of this article will be made available by the authors, without undue reservation.

Author contributions

Conceptualization, CZ; methodology, CZ; software, LY; validation, LY, YG, and KL.; writing—original draft preparation, LY; writing—review and editing, CZ and WW; visualization, DW; supervision, LZ. All authors have read and agreed to the published version of the manuscript.

Funding

This work is supported by National Natural Science Foundation (NNSF) of China (Grant 61603296 and Grant 62073259) and supported by the Natural Science Basis Research Plan in Shaanxi Province of China (Program No. 2023-JC-QN-0752, 2023-JC-YB-533 and 2022JQ-609). This work is also supported by Key Laboratory of complex system control and intelligent information processing in Shaanxi Province.

Conflict of interest

KL, DW, and LZ were employed by State Grid Xi'an Electric Power Supply Company.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

References

1. Samimi M, Hassan Majidi M, Khorashadizadeh S. Secure communication based on chaos synchronization using brain emotional learning. *J Int J Elect Commun* (2020) 127: 153424. doi:10.1016/j.aeue.2020.153424

2. Niu Y, Zhang X. A novel plaintext-related image encryption scheme based on chaotic system and pixel permutation. *J IEEE Access* (2020) 8:22082–93. doi:10.1109/ACCESS.2020.2970103

3. Pecora LM, Carroll TL. Synchronization in chaotic systems. *J Phys Rev Lett* (1990) 64: 821–4. doi:10.1103/physrevlett.64.821

4. Li Z, Xu D. A secure communication scheme using projective chaos synchronization. *Chaos J Solitons Fractals* (2004) 22:477–81. doi:10.1016/j.chaos.2004.02.019

5. Tan F, Zhou L, Lu J, Quan H, Liu K. Adaptive quantitative control for finite time synchronization among multiplex switched nonlinear coupling complex networks. *Eur J Control* (2022) 100764. doi:10.1016/j.ejcon.2022.100764

6. Li X, Zhou L, Tan F. An image encryption scheme based on finite-time cluster synchronization of two-layer complex dynamic networks. *Soft Comp* (2022) 26:511–25. doi:10.1007/s00500-021-06500-y

7. Chee CY, Xu D. Secure digital communication using controlled projective synchronisation of chaos. *J Chaos Solitons Fractals* (2005) 23:1063–70. doi:10.1016/s0960-0779(04)00366-2

8. Zhou L, Tan F, Yu F. A robust synchronization-based chaotic secure communication scheme with double-layered and multiple hybrid networks. *IEEE Syst J* (2020) 14(2): 2508–19. doi:10.1109/jsyst.2019.2927495

9. Tan F, Zhou L. Analysis of random synchronization under bilayer derivative and nonlinear delay networks of neuron nodes via fixed time policies. *ISA Trans* (2022) 129: 114–27. doi:10.1016/j.isatra.2022.01.023

10. Li GH. Modified projective synchronization of chaotic system. *J Chaos, Solitons Fractals* (2007) 32:1786–90. doi:10.1016/j.chaos.2005.12.009

11. Tan F, Zhou L, Lu J, Chu Y, Li Y. Fixed-time outer synchronization under double-layered multiplex networks with hybrid links and time-varying delays via delayed feedback control. *Asian J Control* (2022) 24:137–48. doi:10.1002/asjc.2420

12. Zhou L, Tan F, Li X, Zhou L. A fixed-time synchronization-based secure communication scheme for two-layer hybrid coupled networks. *Neurocomputing* (2021) 433:131–41. doi:10.1016/j.neucom.2020.12.033

13. Lin H, Wang C, Cui L, Sun Y, Xu C, Fei Y. Brain-like initial-boosted hyperchaos and application in biomedical image encryption. *IEEE Trans Ind Inform* (2022) 18(12): 8839–50. doi:10.1109/tii.2022.3155599

14. Zhou L, Tan F, Fei Y, Liu W. Cluster synchronization of two-layer nonlinearly coupled multiplex networks with multi-links and time-delays. *Neurocomputing* (2019) 359:264–75. doi:10.1016/j.neucom.2019.05.077

15. Lin H, Wang C, Xu C, Zhang X, Herbert HC. A memristive synapse control method to generate diversified multi-structure chaotic attractors. *IEEE Trans Computer-Aided Des Integrated Circuits Syst* (2022) 1. doi:10.1109/TCAD.2022.3186516

16. Zhou L, Tan F. A chaotic secure communication scheme based on synchronization of double-layered and multiple complex networks. *Nonlinear Dynamic* (2019) 96:869–83. doi:10.1007/s11071-019-04828-7

17. Liu L, Wang J, Liu C. Fast synergetic control for chaotic oscillation in the power system based on input-output feedback linearization. *J Math Probl Eng Theor Methods Appl* (2021) 2021:1–14. doi:10.1155/2021/8404192

18. Letellier C, Barbot J-P. Optimal flatness placement of sensors and actuators for controlling chaotic systems. *J Nonlinear Sci* (2021) 31(10):103114. doi:10.1063/5.0055895

19. Yu F, Shen H, Yu Q, Kong X, Sharma PK, Cai S. Privacy protection of medical data based on multi-scroll memristive hopfield neural network. *IEEE Trans Netw Sci Eng* (2022) 1–14. doi:10.1109/TNSE.2022.3223930

20. Rabah K, Ladaci S. A fractional adaptive sliding mode control configuration for synchronizing disturbed fractional-order chaotic systems. *J Circuits, Syst Signal Process* (2020) 39(3):1244–64. doi:10.1007/s00034-019-01205-y

21. Yu F, Kong X, Mokbel AAM, Yao W, Cai S. Complex dynamics, hardware implementation and image encryption application of multiscroll memeristive hopfield neural network with a novel local active memeristor. *IEEE Trans Circuits Systems–II: Express Briefs* (2022) 70:326–30. doi:10.1109/TCSII.2022.3218468

22. Shen H, Yu F, Wang C, Sun J, Cai S. Firing mechanism based on single memristive neuron and double memristive coupled neurons. *Nonlinear Dyn* (2022) 110:3807–22. doi:10.1007/s11071-022-07812-w

23. Tan F, Zhou L, Xia J. Adaptive quantitative exponential synchronization in multiplex Cohen-Grossberg neural networks under deception attacks. *J Franklin Inst* (2022) 359(18): 10558–77. doi:10.1016/j.jfranklin.2022.09.020

24. Lin H, Wang C, Sun Y, Wang T. Generating -scroll chaotic attractors from a memristor-based magnetized hopfield neural Network<i/>. *IEEE Trans Circuits Systems–II: Express Briefs* (2023) 70:311–5. doi:10.1109/TCSII.2022.3212394

25. Sepestanaki MA, Barhaghtalab MH, Mobayen S, Jalilvand A, Fekih A, Skruch P. Chattering-free terminal sliding mode control based on adaptive barrier function for chaotic systems with unknown uncertainties. *J IEEE Access* (2022) 10:103469–84. doi:10.1109/access.2022.3209993

26. Yan M, Song J, Zuo L, Yang P. Neural adaptive sliding-mode control of a vehicle platoon using output feedback. *J Energies* (2017) 10(11):1906. doi:10.3390/en10111906

27. Zhang C, Li J. Adaptive iterative learning control of non-uniform trajectory tracking for strict feedback nonlinear time-varying systems with unknown control direction. *Appl Math Model* (2015) 39:2942–50. doi:10.1016/j.apm.2014.10.070

28. Qiang H, Lin ZZ, Zou XG, Sun C, Lu W. Synchronizing non-identical time-varying delayed neural network systems via iterative learning control. *C Neurocomputing* (2020) 411:406–15. doi:10.1016/j.neucom.2020.05.053

29. Cheng CK, Chao PCP. Chaos synchronization between Josephson junction and classical chaotic system *via* iterative learning control. In: 2018 IEEE International Conference on Applied System Invention (ICASI); April 2018; Chiba, Japan (2018). p. 1232–1235. doi:10.1109/ICASI.2018.8394512

30. Zhu S, Sun M-X, He XX. Iterative learning control of strict-feedback nonlinear time-varying systems. *J ACTA AUTOMATICA SINICA* (2010) 36(3):454–8. doi:10.3724/sp.j.1004.2010.00454

31. Zhang C, Li J. Hybrid function projective synchronization of chaotic systems with uncertain time-varying parameters via fourier series expansion. *Int J Automation Comput* (2012) 9(4):388–94. doi:10.1007/s11633-012-0659-8

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 16 February 2023
DOI 10.3389/fphy.2023.1145361

OPEN ACCESS

EDITED BY
Fei Yu,
Changsha University of Science and Technology, China

REVIEWED BY
Xin Yao,
Central South University, China
Shiwen Zhang,
Hunan University of Science and Technology, China

*CORRESPONDENCE
Xiaoming Lin,
411833214@qq.com

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 16 January 2023
ACCEPTED 07 February 2023
PUBLISHED 16 February 2023

CITATION
Jiang W, Lin X, Yang Z, Xiao Y, Zhang K, Zhou M and Qian B (2023), A credible and adjustable load resource trading system based on blockchain networks. *Front. Phys.* 11:1145361. doi: 10.3389/fphy.2023.1145361

A credible and adjustable load resource trading system based on blockchain networks

Wenqian Jiang[1], Xiaoming Lin[2,3]*, Zhou Yang[1], Yong Xiao[2,3], Kun Zhang[1], Mi Zhou[2,3] and Bin Qian[2,3]

[1]Metrology Center of Guangxi Power Grid Co., Ltd, Nanning, China, [2]Electric Power Research Institute, CSG, Guangzhou, China, [3]Key Laboratory of Intelligent Measurement and Advanced Measurement Enterprises of Guangdong Power Grid, Guangzhou, China

In recent years, the residents' demands for power supply is increasing. The load resource trading system responds to the demands through intelligent scheduling, which can effectively relieve the severe power load pressure. The load resource trading system is a type of nonlinear system because the trading price is adjustable with user's credit, instead of being linear to the power trading volume. The adjustable load resource trading is faced with the challenges of large demands, strong user autonomy, and secure and tamper-proof transaction data. The blockchain technology has been widely concerned by industrial and academic domains due to its decentralization, strong encryption of account information, and traceability of transaction behaviour. In this paper, we propose a credible and adjustable load resource trading framework based on blockchain, which uses blockchain to achieve credible grid load resource trading. Firstly, we propose a two-layer blockchain architecture based on the alliance chain. The main chain maintains all the data of the system, and the station-area nodes constitute the alliance chain. We design a distributed trading processing mechanism based on hybrid consensus and sharding technologies, which promotes the speed of cross-station transaction consensus. Next, we propose a two-level bidding model, which determines the trading price of load resources based on the maximum benefit of users and the lowest cost of grid companies. The results of extensive experiments shows that our proposed framework can achieve the promising result.

KEYWORDS

blockchain networks, nonlinear systems, data security, load resource trading, bidding

1 Introduction

The current social power resource distribution is unbalanced, and the urban power burden is increasing day by day. Especially, residents' demands for stable and flexible power supply are increasing with the improvement of quality of life. For example, during the summer power peak, the peak load of the grid continues to climb, but the duration is very short. High peak-valley difference brings great challenges to the stability of the power network and economic operation. Credible and adjustable load resource trading can start, stop, and adjust the operation status of the power supply equipment and the energy storage equipment, according to electricity price and transaction information. It alleviates the contradiction between supply and demand, and ensures the stable and efficient operation of the large smart grid.

Targets for credible and adjustable load resource trading include decentralized autonomy and reliable transactions. Decentralized autonomy refers to that grid companies only need to publish relevant power consumption control indicators, and do not need to participate in residents' power load management in depth. Users are allowed to trade electricity, manage their peak power consumption range, and monitor the specific electricity consumption of each household appliance. To achieve decentralized autonomy, grid participants from all regions need to spontaneously coordinate, respond, and jointly maintain the operation of the low-voltage interactive response management platform. This can not only reduces the business burden of grid companies but also makes power load resource scheduling more flexible and improves the utilization of sustainable power resources. Reliable transactions refer to the integrity, consistency and accuracy of electricity and transaction data. All data cannot be tampered with. Specifically, data records shall truly reflect the actual operation or process. Original data shall be directly and synchronously recorded into the formal record according to the corresponding procedures or regulations at the time when it is generated or observed. Transaction can be traced to the creator of the data through the signature and other information in the record.

Traditional load resource trading relies on third-party central institutions or trusted intermediaries, which manage and endorse user data in the form of central nodes or service platforms. However, the third-party central institutions will not only increase the data processing and trading costs, but also may cause a reduction in service quality and even a crisis of trust. Blockchain technology provides decentralized services, with the characteristics of strong encryption of account information and traceability of transaction behaviour [1,2]. Blockchain stores, verifies, transmits transaction data through distributed nodes of the network, and ensures the tamper-proof and consistency of transaction data through consensus technology and Merkle hash tree, thereby having attracted extensive attention from industry and academia. Although some researchers have applied blockchain technology to the smart grid, such as data trading [3–5], electric vehicle charging and discharging management [6,7], and micro-grid energy auction [8], it is still in its infancy and needs to be improved.

Therefore, we propose a framework on credible and adjustable load resource trading based on blockchain, which employs blockchain technology to achieve reliable grid load resource trading, and encourages users to actively respond and participate in reasonable power resource scheduling. The blockchain adopts the bookkeeping method of distributed ledger, and blockchain data are transparent and open and cannot be tampered with, which can prevent the power demand response business from recording false and wrong accounts. As shown in Figure 1, the low-voltage interactive response terminal collects the power consumption data of each smart home appliance in the user's home, and sends part of the data to the IoT platform through the WiFi module, and controls the smart home appliances to realize information interaction with users. At the same time, all the collected information is transmitted to the low-voltage interactive response management platform through the blockchain encryption link, and the acquired data are synchronized to the station-area node. The station-area node collects the data of all low-voltage interactive response terminals in the station area and makes consensus with other station-area nodes in the station area. The transaction data between different station areas are agreed and stored by the selected special station-area nodes. As the full node of the blockchain, the master station is responsible for synchronizing the node data of station areas and storing all the data in the whole system. There are four roles in the system: low-voltage users, grid companies, load aggregators, and regulators. Low-voltage users are the main participants in the trading. The grid company is mainly responsible for making decisions on electricity prices and smart contract writing. The load aggregator is responsible for aggregating user demand responses and other information. The regulatory authority supervises and manages the whole process.

Based on the blockchain architecture, we propose a reliable synchronous diffusion mechanism of transaction data based on sharding. Full nodes of the grid master station maintain a main chain, and the station-area node forms an alliance chain. The data of the alliance chain consensus is synchronously spread to full nodes of the master station, and full nodes maintain all data of the system. We design a hybrid consensus mechanism of "response strength + PoW + PBFT". Then the distributed trading processing is implemented based on the sharding technology to improve the scalability of the system. At last, we build a transaction system of users' low-voltage load demand response, and design a two-layer bidding model based on the maximization of users' interests and the lowest cost of grid companies. The upper level model is designed for the transaction matching stage, and the goal is to maximize the interests of all groups of users participating in demand response transactions. The lower level model corresponds to the settlement stage. It runs smart contracts, and achieves information release, transaction matching, transaction settlement, price incentive and other trading processes.The goal is to minimize the cost of the grid company.

To sum up, our main contributions are as follows.

- We propose a framework for credible and adjustable load resource trading, which uses blockchain technology to achieve reliable grid load resource scheduling and power resource trading for low-voltage users.
- We propose a two-layer blockchain architecture based on the alliance chain. We design the mixed consensus mechanism and distributed trading processing based on sharding, in order to achieve reliable synchronization of blockchain node data.
- For grid load trading, we design the trading rules and a two-layer bidding model, with the goal of making balance between different groups.
- We conduct extensive experiments on blockchain prototype system. The experimental results show the effectiveness of our proposed framework.

The remainder of this paper is organized as follows. We summarize the related work in Section 2. The system model is presented in Section 3. We discuss about the synchronous diffusion mechanism of transactions in Section 4. We present the low-voltage load demand response trading scheme in Section 5. The experimental design and results are presented in Section 6. Finally, Section 7 presents the conclusion.

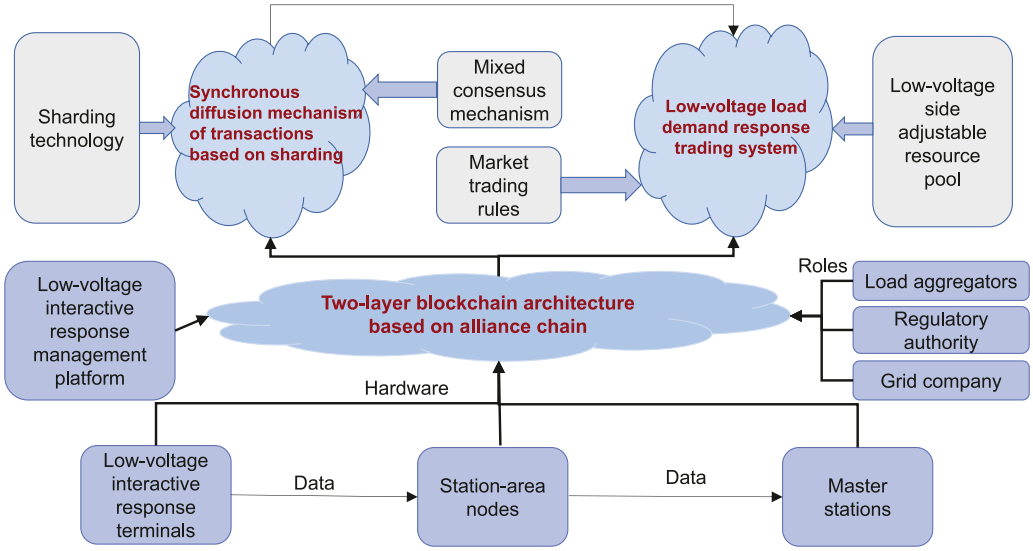


FIGURE 1
Framework of the credible and adjustable load resource trading system.

2 Related work

Recently, researchers begin to use the alliance blockchain framework for credible and adjustable load resource trading. In [9,10], an alliance blockchain based system is proposed for reviewing and verifying transaction data. The schemes in [3–5] enable secure data sharing and storage in the internet of vehicles as well as the reliability and integrity of data transmission. In [6,7], an alliance chain for building a secure energy trading and an electric power trading system are proposed. [11] proposed a P2P energy blockchain system in the energy transaction scenario of the industrial Internet of things. [8] proposed a micro-grid energy auction method based on alliance blockchain to protect the privacy information of each participant. [12] proposed a security privacy protection scheme based on medical health network.

A large number of researchers have begun to increase the throughput of transactions on blockchain *via* sharding and improving the consensus mechanism. [13] proposed to use a three-tier architecture of shards to ensure the availability of consumers' data, limit competitors' access, and provide scalability for processing transaction loads. [14] proposed a polynomial coding sharding scheme, which realized the information's theoretical upper limit of storage efficiency, system throughput and trust. [15] proposed a novel two-layer scalable blockchain architecture. In the upper layer of the chain, sharing-oriented shards are used to achieve safe and efficient processing of macro-transactions on the chain. Off-chain layer is designed for handling real-time shared transactions. [16] proposed the membership-based hierarchical sharding system. [17] designed a layered and fragmented consensus based on the collaboration between multiple partitions.

For the demand response system, many researchers begin to pay a lot of attention to the competition model that maximizes the interests of all parties. For instance, an interactive system is designed in the context of power trading market using distributed controllers and centralized auction trading markets in [18–21]. [22] studied the operation models of multiple virtual power plants. [23] designed a distributed iterative algorithm that converges to the variational decomposition of the generalized nash equilibrium problem and a supplementary demand side management program. [24] proposed the optimal bidding model of electric vehicle aggregator based on the relationship between market price and bidding price. [25] proposed an optimal bidding strategy model for implementing the demand response program. [26] formed a daily demand curve or minimize peak demand. [27] modelled demand response bidding in the real-time balanced market. [28] proposed to manage the power distribution (charging and discharging) of plug-in electric vehicles that support vehicles to the power grid. [29] proposed an optimal control strategy and an optimal bidding strategy. [30] proposed a one-way adjustment algorithm for aggregators. [31] proposed a new scheme to optimize virtual power plant operation and bidding strategy. [32] devised a short-term planning framework that predicts the load under dynamic electricity prices *via* building a bidding curve. [33] proposed a robust optimal bidding strategy based on risk measurement. [34] proposed the concepts of just-in-time transmission and process-gate bidding. [35] proposed a compensation mechanism of load aggregators to perform the direct thermostat control program. [36] proposed a two-stage process based on robust optimization. [37] established a new method in to analyze the economic impact of vehicle-to-grid regulatory reserves.

3 System model

The blockchain system in smart grid consists of a main chain and alliance chains. The main chain is composed of nodes in the cloud data center of the grid company. Each node is a full node in the blockchain and stores all data of the whole system. An alliance chain

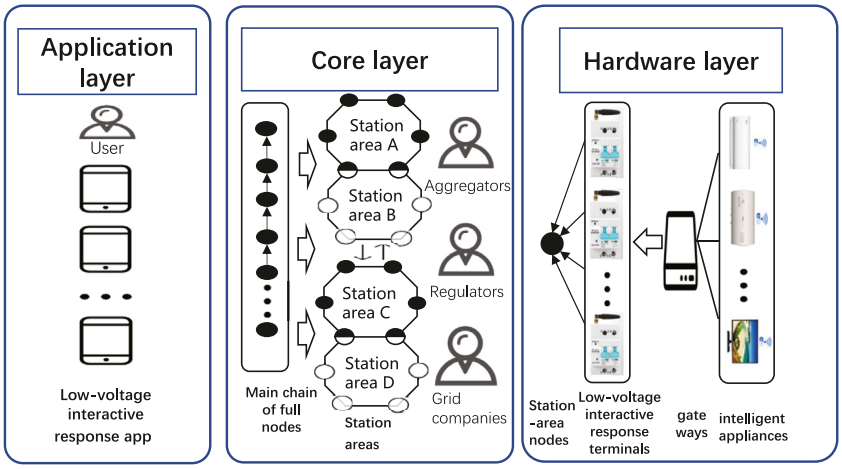


FIGURE 2
System model for load resource trading.

is composed of station-area nodes, where each station-area node manages the low-voltage users of the corresponding zone and stores all the transaction data of those users. Each household uses intelligent terminal equipment to manage household power consumption and join the alliance chain as a lightweight node. It is noted that the number of low-voltage users is huge, and various power load transactions and incentive response businesses incur a considerable amount of data. If the traditional blockchain is used, all nodes need to broadcast data to the whole system to reach a consensus before they can successfully add data to the main chain. This will not only consume a lot of computing resources, but also affect the data transmission efficiency. In order to improve the throughput and scalability of the entire blockchain system, we take the main chain of the grid company as the beacon chain, and build an alliance chain for each station area. A station area contains multiple station-area nodes. Each station-area node manages multiple lightweight low-voltage user nodes. Each lightweight low-voltage user node conducts various business transactions with the station-area node by controlling the low-voltage interactive response terminal equipped by the family. The alliance chain diffuses the data to the main chain, and the main chain performs the data synchronization of the whole network. The advantage of designing lightweight interactive nodes for users lies in their strong real-time performance, which can accurately record the electric energy transactions of low-voltage users in a timely manner. The data transmission mechanism optimized layer by layer by vertical structure can improve the operation efficiency of the blockchain system.

The blockchain system in smart grid takes smart contract as the main body. The main chain has strict requirements on data integrity, privacy, security, and supervision, because it involves all data of power consumption and demand response. The business responses of the grid companies, load aggregators, lightweight low-voltage users and other implementation entities must be uploaded to the regional alliance chain before being synchronized to the main chain network. The function of the alliance chain is to connect a large number of demand response terminals, at the same time, collect the demand response services of various low-voltage users, thereby improving the data throughput and scalability of the entire system.

As shown in Figure 2, the blockchain-based smart grid is mainly divided into three layers: hardware layer, core layer and application layer.

3.1 The hardware layer

The hardware layer is composed of intelligent appliances, gateways, low-voltage interactive response terminals, station-area nodes and master stations. The main control chip inside the intelligent gateway uploads the collected data such as voltage, current, power, household appliance switching time and household appliance status to the terminal. Considering the hardware storage and performance problems, the low-voltage interactive response terminal only stores the power consumption information related to the home, and joins the alliance chain where the station area is located as a lightweight node. In general, the user's transaction, electricity information and other data are synchronously diffused from the low-voltage interactive response terminal to the station-area nodes in this station area. One station-area node manages multiple intelligent terminal devices. Each terminal device is bound to each user. Finally, the data is synchronously spread from the station-area node to full nodes of the master station.

3.2 The core layer

The station-area nodes are organized into an alliance chain according to geographical regions. Each alliance chain stores and maintains all transactions and data of users associated with low-voltage interactive response terminals in the chain, and runs its own consensus mechanism independently. These alliance chains will synchronously spread all data to full nodes of the main chain, make consensus on the main chain and store it on the chain.

The main chain, as a beacon chain, sends corresponding demand response instructions to the network and records the response results of all station-area nodes. Full nodes of the main chain will audit and evaluate the response effect according to the implementation requirements of the deployed smart contract, and maintain the data of the entire system. All data information stored on the chain cannot be changed. Users can view their own related transactions, electricity and other information on the chain. To improve the efficiency of the whole blockchain-based grid system, we also adopt blockchain sharding technology to design a reliable synchronization mechanism for cross-station-area transactions, which enables users from different station areas to conduct trusted transactions.

At the core level, there are three entity roles: load aggregator, grid company and regulatory authority. The load aggregator is responsible for aggregating the demand response and other information of low-voltage users. The grid company is mainly responsible for making decisions on electricity price and smart contract. The regulatory authority supervises and manages the whole trading process and the operation of the grid system. When the grid company needs to release new power data, it will synchronously spread the data to each low-voltage user's intelligent interactive device with the help of blockchain network broadcast service information, and feed back to the user through the low-voltage interactive response APP on the mobile terminal. Similarly, when users conduct transactions, the information needs to be uploaded to the main chain, which integrates the transactions and power consumption data within a period of time and broadcasts them to the whole network. In this process, the regulatory authority can meet user needs at any time to view and trace relevant transaction data to ensure the security of data on the chain.

3.3 The application layer

The low-voltage interactive response APP receives the interactive response demand from the grid company, collects and reports the bidding response information from users in a unified way, and realizes the low-voltage user interactive response business processing, interactive response strategy generation, user management, and data query and display. Low-voltage users will generate their own independent user codes when accessing the blockchain network, and control the low-voltage interactive response terminals loaded in the home through the low-voltage interactive response APP, so as to meet the needs of low-voltage users in the same or another station area. For example, a low-voltage user works overtime tonight, so the household electricity load is small during this period. After this situation is reported to the system, the system will release the invitation information and distribute the power to other low-voltage users who have successfully bid; The low-voltage interactive response APP will reward or punish users according to their performance, thus reducing the peak pressure of power consumption in the whole grid. Users can view their own uplink historical data, which cannot be changed or deleted. However, to protect the privacy of each low-voltage user, users cannot view others' uplink data. When the system generates a new block, it also generates a large number of rewards, that is, power incentive points. The points will be managed by the grid company in a unified way. Low-voltage users can reasonably allocate their own peak power consumption interval by participating in the inductive load reduction compensation notice invitation sent by the power supplier, so as to reduce the power load and respond to the power supply, thus ensuring the stability of the grid. On the other hand, in combination with the incentive compatibility demand in the new energy electric vehicle Internet, the battery of the electric vehicle is used as a distributed energy storage system to select discharge at the peak load and charge at the low load, balance the peak load of energy, reduce the fluctuation level of resource supply, and realize the scheduling and commercial use of intelligent energy. In addition, photovoltaic new energy technology is used to realize the self-use of photovoltaic power generation. Excess electricity can be sold to grid companies at the discretion of low-voltage users. Each low-voltage user will have the credit value of his/her own account. By actively responding to the energy-saving incentive plan, accurately reporting electricity information, and conducting power resource transactions among users in good faith, high credit points can be maintained. Reliable users with integrity will get more preferential rights and interests, while users with low integrity will not only be included in the credit file, but also have some impact on personal daily electricity use.

4 Synchronous diffusion mechanism of transaction data based on sharding

In this section, we study the synchronous diffusion mechanism of transaction data. We make full use of the tamper-resistant feature of the blockchain to ensure the accuracy of transactions, and achieve reliable synchronization of blockchain data. All nodes of the power grid master station maintain a main chain, and the low-voltage interactive response terminal synchronously diffuses the generated comprehensive energy service data to the nodes of the local station area. Nodes of the same station area reach a consensus on the data. Each alliance chain is associated with the main chain, and the data are synchronously diffused from the station-area node to all nodes of the master station. All nodes agree and store the data in the main chain, which can ensure the synchronous diffusion of demand response transaction data in the station area. When the block is coming out, the grid company will calculate the demand response strength of each station according to the detailed information of the demand response report during the block out period and the user feedback data, and determine the bookkeeping node in each station area.

4.1 Design of consensus mechanism

The grid system is based on the alliance chain. The nodes themselves have independent identification codes and have certain credibility with each other. We design a hybrid consensus mechanism of "response strength + PoW + PBFT", which groups the nodes according to the transaction granularity, reduces the number of nodes participating in the consensus, and improves the consensus efficiency. The response mechanism of the station-area nodes is based on the response value of the station-area node within the station area to the business needs of grid companies. If a user actively

participates in the energy conservation incentive plan of the grid, he/she will obtain high response value. The higher the response value, the more incentive points, and the higher the credit certificate. The full name of PoW is Proof of Work. All nodes in the station region use their computing power to solve a mathematical problem. The nodes that get the correct results will be eligible to generate new blocks, thus obtaining the accounting right of the block. PBFT is a distributed system consensus algorithm that can tolerate Byzantine errors. Its purpose is to improve the node fault tolerance of the blockchain network and improve the throughput of cross-region transactions.

In the hybrid consensus mechanism, because of the credibility provided by the alliance chain, the station-area node does not need to do more PoW consensus proof, which can avoid the formation of computing power competition and resource waste in the whole network and reduce the computing pressure of nodes. The rewards obtained from the blocks will be uniformly recovered and managed by the grid, and then a certain amount of power incentive points will be issued according to the credit value and the response business strength. Considering the strong interconnection of the station-area nodes on the physical equipment, in order to get a better experience when users interact with the blockchain grid using lightweight interactive terminals, nodes in the blockchain can be divided into groups *via* the sharding technology. Each partition processes transactions in parallel, which can significantly improve the throughput and scalability of the blockchain.

4.2 Trading process within the station area

The transactions within the station are all in the same alliance chain. Each user node only stores the transaction data related to itself. The station-area node stores the data of the entire station. The synchronization consensus is completed within the station area. In order to facilitate users to manage and operate their own nodes in the mobile device and lightweight network environment, the low-voltage interactive response terminal will only store key information related to users, such as user name, password, power equipment, and power quota statistics, and redundant data will be uploaded to the station. For example, in a certain period of time, low-voltage users *A* and *B* conduct demand response transactions: *A* has a small demand for electricity, which is reported to the system to respond to, and *B* purchases the electricity. If two users are judged to belong to the same station, user *B* confirms whether *A*'s credit value is lower than the penalty threshold and whether his balance is sufficient. After confirmation, both parties can quickly conduct transactions. *A* needs to save the submitted response power within this time period. After the transaction is completed, the smart contract will compare the response value submitted by user *A* with the actual load of user *A* obtained from the low-voltage interactive response terminal, in order to determine whether user *A* performs. According to the performance of users, corresponding rewards and punishments will be given.

4.3 Cross-area Trading process

We propose the reliable synchronous blockchain model for cross-area transactions based on blockchain sharding. Existing blockchain sharding methods can be divided into two types: full sharding and partial sharding. In the full sharding model, each node only stores the transaction data in its own shard. Hence cross-shard transactions will incurs large time and communication overhead. In the partial sharding model, cross-shard nodes store data of multiple shards. Even if the sender and receiver of a transaction are located in two different shards, the cross-shard nodes can verify the legitimacy and atomicity of the transaction based on their own kept data, without the cross-shard communication. In this way, the cross-shard transactions are transformed into intra-shard transactions, thereby reducing time cost. Therefore, we choose partial shards. The nodes in each state area comprise a shard. We divide the station-area nodes into two categories. The station-area nodes used for cross-area interaction are called cross-nodes, and other nodes are called normal-nodes. Cross-nodes store data of different shards, and they are not resident. They are selected by the credit mechanism and then re-selected periodically in the area. Power incentive points will be awarded to corresponding lightweight users.

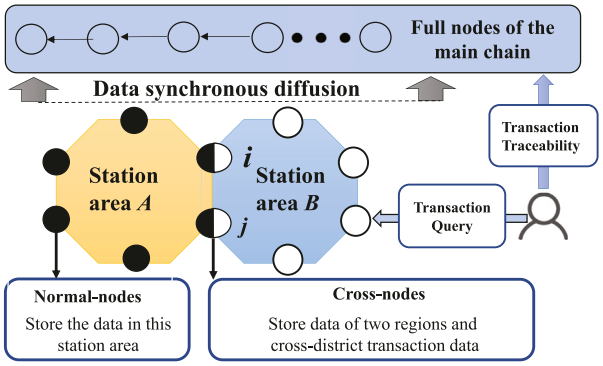


FIGURE 3
Reliable synchronous diffusion mechanism of transaction data.

According to the total credit value and the variance of credit value of users in the station area, we combine the two factors to select the station-area node with a stable upward trend in credit value. These key nodes use the client driven mechanism to submit cross-area transactions, store all cross-area transaction data, and complete cross-area consensus, storage, and communication. As shown in Figure 3, when a user sends a request for the cross-area transaction, the node of the area where the user belongs to will initiate a cross-area consensus and call a universal smart contract that supports cross-area transactions.

A cross-area transaction means that both low-voltage users of the transaction are not in the same station area. If both parties to the transaction are judged to be located in different areas, they will initiate a transaction request at the cross-node in their respective station. It is worth mentioning that since transactions in different areas are asynchronous, data synchronization across areas is required. In addition, the atomicity of cross-area transactions must be guaranteed. The process of cross-area transactions is as follows.

(1) After a low-voltage user node in area *A* sends a cross-area transaction request, he/she broadcasts the request to the alliance

chain of *A*. For example, if user *B* works overtime tonight and does not use electricity at home, he/she will report the situation to the system and distribute electricity to the user *A* in different areas.

(2) Cross-nodes *i* and *j*, both located in areas *A* and *B*, will initially verify the legitimacy and atomicity of the transaction after receiving the request. Atomicity means that the sender has enough balance, so that the sender has enough expenses to pay. The receiver actually saved the response power within the specified time according to the content agreed in the smart contract. Therefore, the alliance chain of the sender must confirm that there is enough balance in user *A*'s account to pay, and the alliance chain of the receiver must confirm that user *B* has saved the response power in this period of time. If the user fails to perform, the user can be warned and powered off.

(3) After the transaction, all nodes of the grid company will judge whether user *B* performs according to the provisions of the smart contract, and give corresponding rewards or punishments to users according to the performance. Finally, the two cross-nodes, *i* and *j*, will package all the transaction data into blocks after completing the consensus of the *A*'s and *B*'s areas, and broadcast them to the alliance chains to complete the synchronization of cross region data. At the same time, the cross-node will synchronize the block to all nodes of the main chain, and the block will recorded on the main chain after all nodes have reached a consensus.

4.4 Tracing and querying blockchain Transactions

To ensure the authenticity and reliability of transaction data, we have provided low-voltage users with query services for tracing blockchain transactions, blocks, transaction time, by making use of the openness and transparency of blockchain data. For business transaction data in question, low-voltage users can first submit a query service request to nodes of their own areas. In order to protect the privacy of other low-voltage users, the data queried are limited to those existing with them. If it is found that the sender or receiver of the transaction to be queried is not in this area, the relevant request shall be submitted to the regulatory authority, which will view the main chain information of the grid company and feed back the detailed transaction data records on the user chain.

To achieve transaction traceability, it is necessary to increase the transparency of data governance. The trusted and queriable transaction system can open the door to new digital reliability sharing services. All online data are processed with encryption technology to fully protect the privacy of low-voltage users. A transaction chain governance model is established to provide network supervision, so as to optimize the reliable synchronous diffusion mechanism of the entire transaction data.

5 Low-voltage load demand response trading

Low-voltage load demand response of users refers to the power consumption adjustment made by users according to the demand response of the grid when the power consumption peak or low peak occurs due to the unbalanced load of the grid. The user completes the demand response transaction based on the principle of signing a contract before demand response and clearing after demand response according to the point to point transaction power consumption time and load of his own demand. Low-voltage load demand response of users is realized by running the smart contract in the lightweight blockchain system. The grid companies, low-voltage users, aggregators and regulators in the system run the smart contract to achieve information release, transaction matching, transaction settlement, price incentive and other functions to complete the trading process. The trading system consists of three modules: low-voltage side adjustable capacity resource pool, trading rules and price incentive mechanism. Among them, the response volume and expected price provided by users during bidding form a low-voltage side adjustable capacity resource pool, which together with the terminal equipment provides the data source for the smart contract. When users run the smart contract, they will match and conduct point-to-point transactions according to the data and transaction rules of the resource pool, and settle after the end of the demand response cycle. The transaction matching and transaction settlement constitute a two-level bidding model with the goal of maximizing the interests of users and systems. After the demand response cycle is completed, the price incentive mechanism will be implemented when verifying whether the user performs the contract. A corresponding proportion of price incentives will be given to the users who perform the contract, and economic penalties will be given to the users who break the contract.

In the low-voltage load demand response trading system, the grid company issues demand response signals according to the load dispatching demand; low-voltage users independently choose to participate in the demand response trading process according to the optimal strategy provided by the signal and bidding model. The aggregator conducts transaction settlement with the goal of maximizing the return of users and grid demand response. The regulatory authority is responsible for formulating trading rules and supervising the process of trading. The specific transaction system is shown in Figure 4.

We present a two-level bidding model in subsection 5.1. As shown in Figure 4, this model is used for transaction matching and transaction settlement. Next, we present the details of the trading process based on smart contracts. The two-level bidding model is the theoretical basis of transaction processing.

5.1 A Two-level bidding model to maximize the interests of users and minimize the cost of grid companies

The bidding model consists of two levels: the upper level corresponds to the transaction matching stage, with the goal of maximizing the interests of all groups of users participating in demand response transactions; The lower level model corresponds to the settlement stage, and the optimization goal is to minimize the cost of the grid company.

5.1.1 The upper level model

The upper level model aims to maximize the interests of group users under the constraints of communication cost, signing and

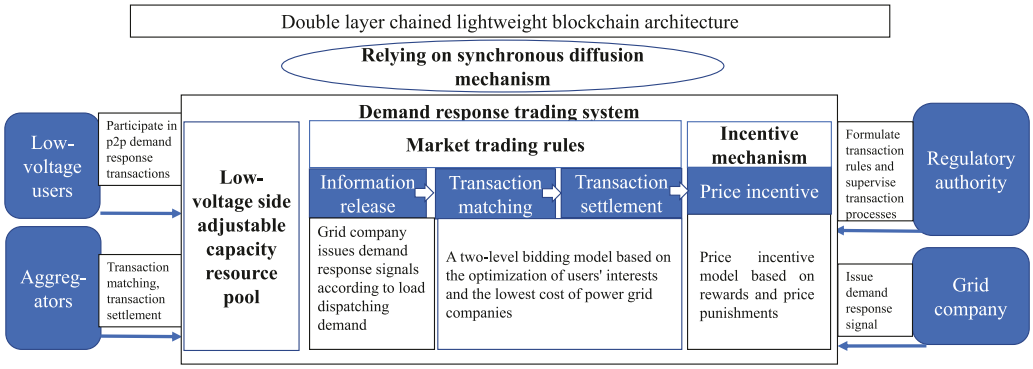


FIGURE 4
Demand response trading.

actual power load, signing and actual power price. We use I_{users} to denote the income of low-voltage users. We use a and b to denote the buyer and seller of point-to-point transactions, respectively. i and j denote the number of the station where low-voltage users are located. t denotes the time. C denotes the comfort cost. W_{tia} represents the load signed by the seller in the i station at time t. p_{tia} denotes the unit price of the seller's contracted electricity in the i station at time t, and $\sum_{tia} W_{tia} p_{tia}$ denotes total contract amount in the system, namely W_{tia} and p_{tia} product. Correspondingly, we use W_{tjb} to denote the actual power load of the buyer in the j station at time t. However, it should be noted that the power load may exceed the contracted load at this time due to the possibility of the seller's breach of contract. We use p_{tjb} to denote the unit price of the buyer's actual power load in the $j-th$ station at time t, so the sum of the demand response amount in the system can also be expressed by the sum of the product of the two (W_{tjb} and p_{tjb}). Therefore, we use $\sum_{tia} W_{tia} p_{tia} - \sum_{tjb} W_{tjb} p_{tjb}$ to denote the difference between the sum of the contract amount in the system and the sum of the demand response amount. It also means the amount and profit generated by the point-to-point transactions of the total low-voltage user groups in the entire demand response system. After the subtraction of the cost of comfort, it can be used to express the goal of maximizing the interests of group users. As shown in Eqs 2, 3, we use D to denote the communicate cost, $Temp$ to denote temperature, $\alpha_{tTemp}$to denote comfort coefficient, X to denote the time cost of a single communication between a cross-node and a node in a station, and Y to denote the communication time cost between two cross-nodes in different station. In the system, the comfort cost of low-voltage users equivalent to the power supply side in demand response is mainly considered. The cost affected by natural factors such as temperature and time can be obtained by summing the product of comfort coefficient and actual power load, that is, $\sum \alpha_{tTemp} \cdot W_{tjb}$. The communication cost needs to be discussed according to the situation of different stations. When users of both sides of the transaction are in the same station, after submission and confirmation, the communication time cost is twice the time cost of a single communication, that is, $2X$. If both parties are in different station, the communication time cost should be added to the communication between two *CrossNodes* in different stations, that is, $2X + Y$. Then the comfort cost can be calculated by the cost affected by natural factors and communication cost, namely $C = \sum_{tTemp} \alpha_{tTemp} \cdot \sum_{tjb} W_{tjb} + D$. In Eq 4–7, we set the contracted load and its unit price. Eq. 4 makes the constraint that the total load signed by sellers must be non-negative. Eq. 5 makes the constraint that the total actual power load of buyers must be non-negative. Eq. 6 makes the constraint that the total price of the seller's contracted electricity must be non-negative. Eq. 7 makes the constraint that the total price of buyer's actual power load must be non-negative. In Eq. 8, we use B to denote the comfort budget, representing the upper limit of the comfort cost in the system.

$$maximize \quad I_{users} = \sum_{tia} W_{tia} p_{tia} - \sum_{tjb} W_{tjb} p_{tjb} - C \tag{1}$$

$$C = \sum_{tTemp} \alpha_{tTemp} \cdot \sum_{tjb} W_{tjb} + D \tag{2}$$

$$D = \begin{cases} 2X + Y & i \neq j \\ 2X & i = j \end{cases} \tag{3}$$

$$\sum_{tia} W_{tia} \geq 0 \tag{4}$$

$$\sum_{tjb} W_{tjb} \geq 0 \tag{5}$$

$$\sum_{tia} p_{tia} \geq 0 \tag{6}$$

$$\sum_{tjb} p_{tjb} \geq 0 \tag{7}$$

$$C \leq B \tag{8}$$

5.1.2 The lower level model

The lower level model aims to minimize the cost of grid companies under the constraint of marginal generation cost price. We use I_{CSG} to denote the cost of the grid company, p_{tjm} to denote the generation price of the grid beyond the demand response plan, and β to denote the real-time price adjustment coefficient. If all users fulfill the agreement, the cost of the grid

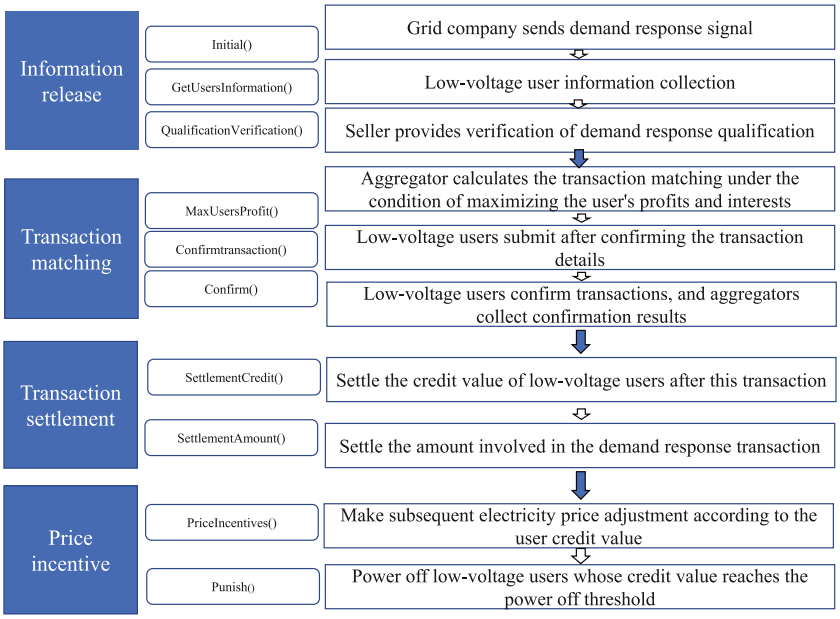


FIGURE 5
Smart contract based trading flow.

will just offset the settlement amount of the user's participation in the demand response. However, in the settlement stage, when a user defaults, there may be two situations: excessive power consumption and failure to reach the agreed power load. Therefore, we use $|\sum_{tjb} W_{tjb} - \sum_{tia} W_{tia}|$ to denote the unplanned power load. As shown in Eq. 10, when $\sum_{tia} W_{tia} > \sum_{tjb} W_{tjb}$, the grid actually provided more power to users than the planned load, and the cost that could not be offset came from the cost of these multiple generating loads. So the real-time price adjustment coefficient was 0. When $\sum_{tia} W_{tia} \le \sum_{tjb} W_{tjb}$, the situation of excessive power consumption by low-voltage users can be expressed. We use $\sum_{tjm} p_{tjm} - \sum_{tjb} \beta . p_{tjb}$ to denote the marginal cost price of grid generation. The product of the unplanned power load and the marginal cost price of grid generation can represent the cost of the grid, as shown in Eq. 9. In addition, it should also ensure that the grid generation price is non negative, that is, Eq. 11.

$$Minimize \quad I_{CSG} = \left(\sum_{tjm} p_{tjm} - \sum_{tjb} \beta . p_{tjb} \right) \left| \sum_{tjb} W_{tjb} - \sum_{tia} W_{tia} \right| \tag{9}$$

$$\beta = \begin{cases} 0 & \sum_{tiab} W_{tia} > W_{tjb} \\ \beta & \sum_{tiab} W_{tia} \le W_{tjb} \end{cases} \tag{10}$$

$$\sum_{tjm} p_{tjm} \ge 0 \tag{11}$$

5.2 Trading process

The trading process can be generally divided into five stages: user registration, information release, transaction matching, transaction settlement and price incentive. The specific trading process is shown in Figure 5.

5.2.1 User registration stage

Firstly, all roles involved in the user low-voltage load demand response trading system are registered in the blockchain system, including grid companies, aggregators, regulatory companies, and low-voltage users. According to the different roles of registration, they perform different functions in the low-voltage load response trading system. After registering in the blockchain system, users can participate in demand response through smart contracts.

5.2.2 Release demand response stage

The initialization function *Initial*(): according to the load scheduling demand, the grid company sends a demand signal to all users in the blockchain system through the *Initial*() function. A global variable of boolean type will be set in the function. If the variable is true, it means that it is allowed to participate in this demand response.

Information collection function *GetUsersInformation*(): the aggregator collects the information of low-voltage users participating in this demand response point-to-point transaction using the information collection function *GetUsersInformation*(). The information collected is divided into two parts: buyer information and seller information. The information collected from these two parties is used to calculate the maximum profit of the user in the transaction matching of the bidding model. The

buyer information includes the demand load, the price acceptable to the user, the desired response time, and the station area where the user is located. The seller information includes the load, price, response time, user's station area and user's credit value that the seller intends to provide.

Qualification verification function *QualificationVerification*(): at the same time, it is necessary to ensure that the seller is qualified to provide demand response quantity, so it should be verified by the function *QualificationVerification*().

5.2.3 Transaction matching stage

Profit maximization function for low-voltage users *MaxUsersProfit*(): At this stage, the aggregator calculates the maximum profit of the user through the function *MaxUsersProfit*() and lists the matching pairs of transactions under the condition of maximizing the benefits. This trading pair is fed back to the user so that the user can decide whether to choose the trading matching result, which is implemented by the bidding model. The variables calculated by the user are from the information collection function.

Submission function *Confirmtransaction*(): After the aggregator calculates the user's maximum profit, the *Confirmtransaction*() function is required for the user to submit.

Confirmation function *Confirm*(): To ensure the user's privacy, the confirmation function *Confirm*() needs to be confirmed by the buyer and the seller respectively. After the buyer and the seller confirm respectively, the aggregator collects the confirmation results and submits them to the blockchain system for the next stage of transaction settlement.

5.2.4 Transaction settlement stage

Credit value settlement function *SettlementCredit*(): according to the comparison between the data obtained from the user intelligent terminals of the buyer and the seller after the completion of the demand response phase and the electricity quantity in the contract, if the contract is performed, the credit value of the user in this transaction will be increased by two points. If the contract is breached, the credit value will be decreased by four points for the settlement of credit value.

Amount settlement function *SettlementAmount*(): the load and price at the time of signing the demand response contract and after the completion of the demand response phase are compared. The point-to-point trading parties settle the transaction amount. The real-time electricity price at the time of demand response is priced by the grid company with the lowest cost as the optimization goal.

5.2.5 Price incentive stage

Price incentive function *PriceIncentives*(): in this stage, according to the credit value obtained by the user whether to perform the contract in the transaction settlement and the real-time electricity price in response, the reward and penalty prices for normal electricity use in the later period are calculated by the RP price incentive model based on the user's credit value.

Penalty function *Punish*(): according to the user's credit value when completing the transaction, it can judge whether the low-voltage user has reached the power outage threshold. That is, if the credit value is lower than 50, the intelligent terminal will be controlled to power off.

TABLE 1 Experimental settings.

Parameters	Values
The number s of shards	[2,3,4,5,6,7,8,9,10]
The number n of nodes	[2,3,4,5]
The number m of transactions	[250,500,750,1000]

6 Evaluation

In the evaluation, we test the reliable synchronous blockchain model. Notice that this synchronous blockchain model is built on partial sharding where cross-nodes store transactions of different station areas. Hence, we compare the partial sharding with the full sharding technique.

6.1 Experiment settings

Experimental data sets. The experimental settings are shown in Table 1 s represents the number of shards, n represents the number of nodes, and m represents the number of transactions. The number of shards varies in [2,3,4,5,6,7,8,9,10], and the number of nodes varies in [2,3,4,5]. In the storage overhead experiment, assuming that the number of transactions stored in different shards is the same. The number of transactions in each shard varies in [250,500,750,1000]. The number of slices varies in [2,3,4]. The blockchain accounts participating in the power points transaction are distributed in different shards according to the station area they are located in. The cross-node stores transactions in their own regions and cross-district transactions, while a normal-node only stores cross-sharding transactions in their own regions and related transactions. Transactions are generated by a blockchain prototype systems[1].

Metrics. Time cost and storage cost are used to evaluate the performance of the methods. In terms of time cost, when the number of shards is fixed, the more nodes, the more consensus time may be required, and the time cost will increase accordingly. When the number of nodes is fixed, the more shards there are, the more consensus time may be required. In terms of storage overhead, since a cross-node needs to store two or more pieces of data, the storage overhead will be much higher than that of a cross-node in the sharding. As the number of shards increases, the storage overhead of the cross-node will increase accordingly.

Experimental environment. A blockchain prototype system has been built for the evaluation. The prototype system runs in a server with a 12th Gen Intel® Core™ i7-12700F processor and 128 *GB* memory. We use the *Ubuntu*22.04 operating system. All algorithms were coded with *Java*. Multiple virtual nodes were generated with *Docker*.

Competitors. We tested the time cost and the storage cost. We tested the influence of the number of nodes and the number of shards. We compare two sharding methods: full sharding and partial sharding. The sharding methods used in this paper is partial shard. We also tested the storage overhead of the two types of nodes, that is,

1 https://github.com/reveup/myproject.git

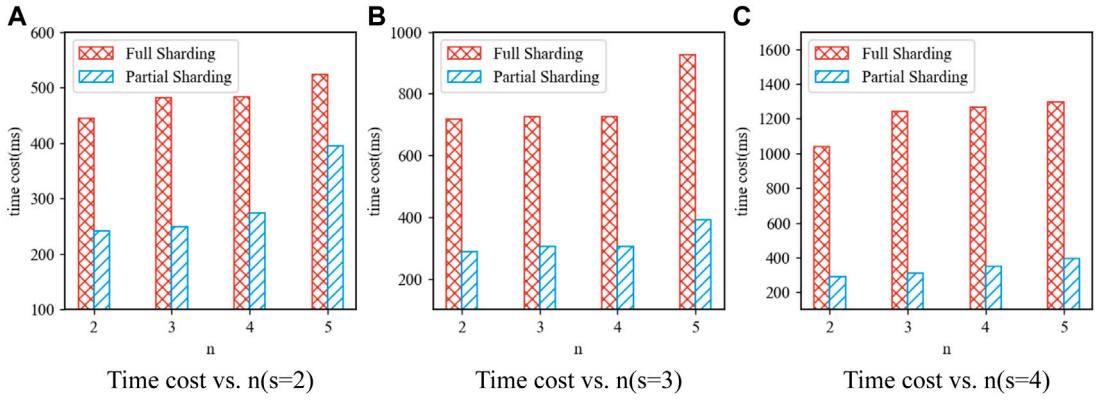


FIGURE 6
Time cost vs n ($s \in [2, 3, 4]$).

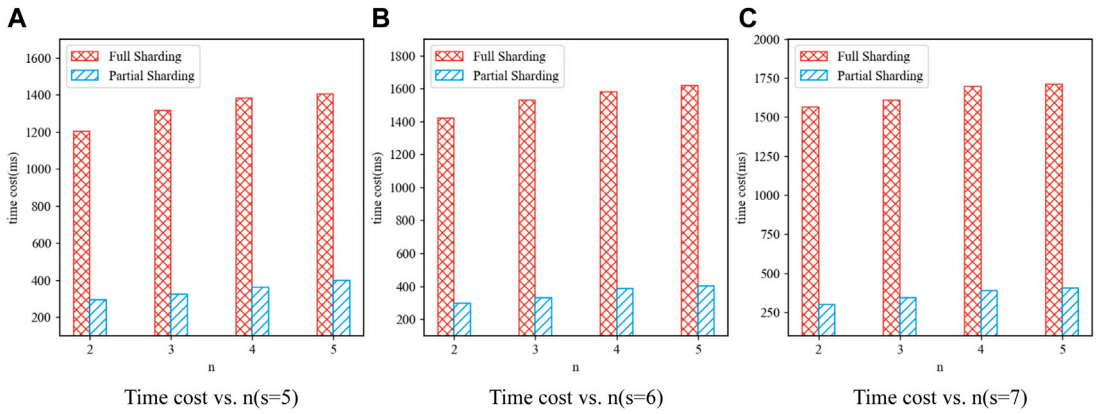


FIGURE 7
Time cost vs n ($s \in [5, 6, 7]$).

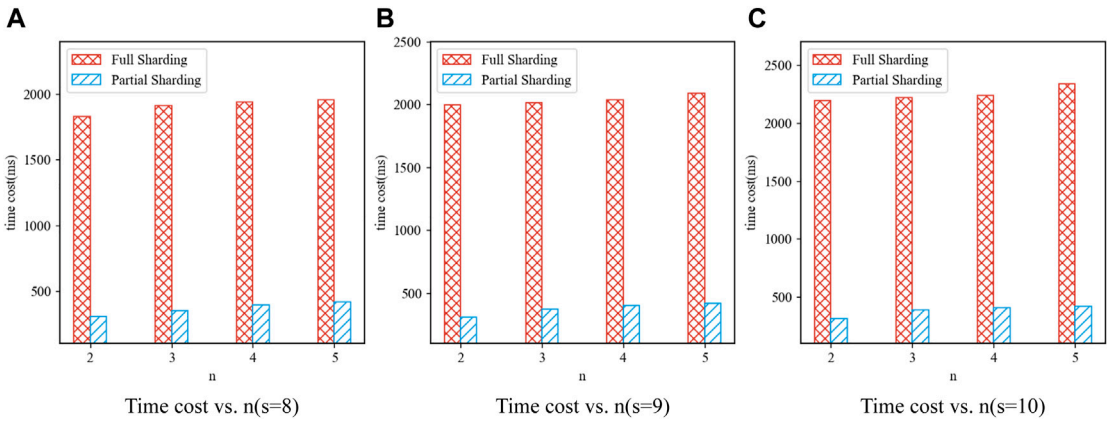


FIGURE 8
Time cost vs n ($s \in [8, 9, 10]$).

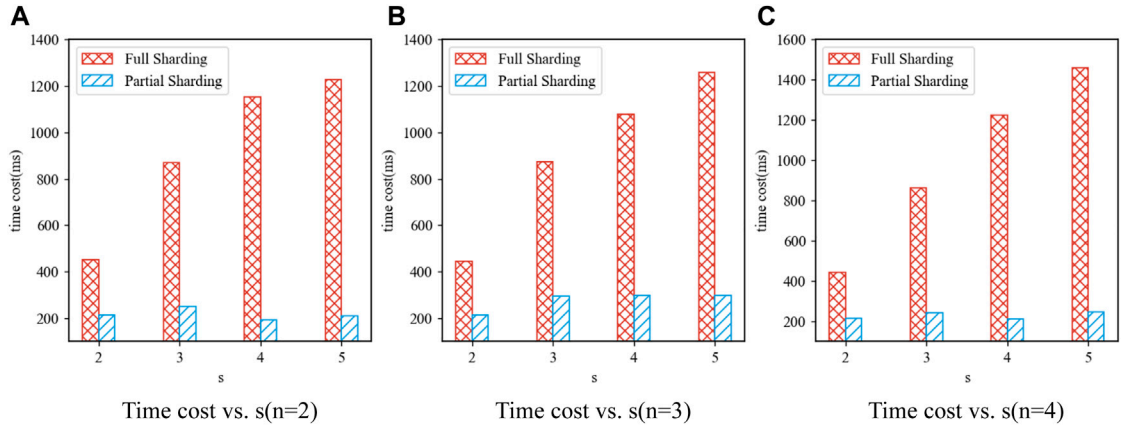


FIGURE 9
Time cost vs s ($n \in [2, 3, 4]$).

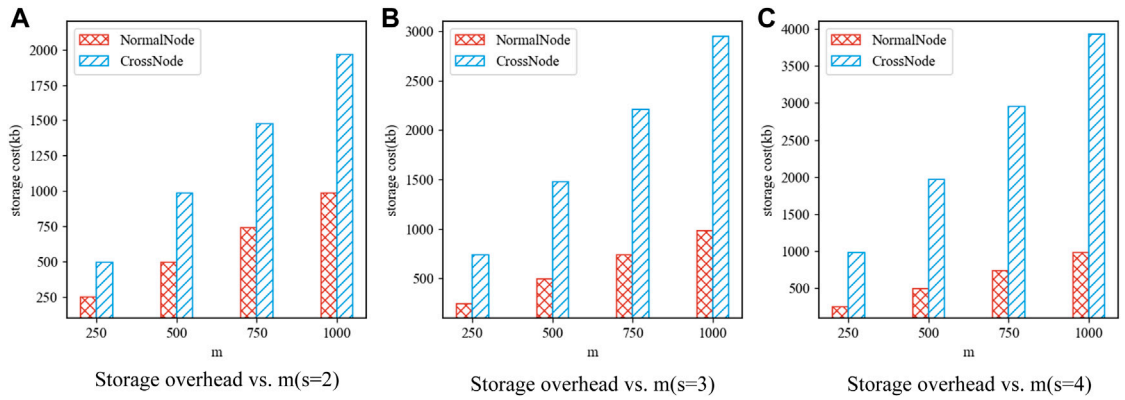


FIGURE 10
Storage overhead vs number of transactions (number of shards: 2, 3, 4).

normal-node and cross-node, in the partial sharding method. The full sharding storage model only contains normal-nodes, and the entire blockchain ledger is divided into several disjoint ledgers and stored in normal-nodes. Each shard maintains its own ledger, and all shards of the entire blockchain network have a complete ledger together. In the partial sharding storage model, a cross-node stores data of multiple shards and processes cross-sharding transactions, so as to shorten the consensus time, but this may bring a huge storage.

6.2 Results of the experiments

6.2.1 Influence of the number of nodes on time cost

In this experiment, we fixed the number of shards as 2,3,4,5,6,7,8,9, and 10, respectively. We tested the time cost required for full shards and partial shards, respectively. The experimental results are shown in Figures 6–8.

From the experimental results, we can see that the time cost of full shards is much higher than that of partial shards, and with the increase of the number of nodes, the time cost shows an overall upward trend. The main reason is that the increase in the number of nodes will affect the consensus time. We use $s = 2$ as an example. When $n = 2$, the time cost of partial shards is 54.3% of that of full shards. When $n = 3$, the time cost of partial shards is 51.9% of that of full shards. When $n = 4$, the time cost of partial shards is 54.3% of that of full shards. When $n = 5$, the time cost of partial shards is 75.5% of that of full shards.

6.2.2 Influence of the number of shards on time cost

In this experiment, we fixed the number of nodes as 2, 3, and 4, respectively. When the number of shards is in [2,3,4,5]. We compare the time cost of the cross-chain transaction between full shards and partial shards. Figure 9 shows the impact of the number s of shards on the time cost of cross-chain transactions of full shards and partial shards. The time cost of cross-chain transactions of full shards is much higher than that of partial shards. With the increase in the number of shards, the time cost of full shards is on the rise as a whole, while the partial shards are relatively stable. The main reason is that the increase in the number of shards involved in cross-chain transactions will lead to an increase in

consensus and communication time for full shards. However, because partial shards use cross-nodes to store multiple pieces of data, the consensus needs to be conducted only once. We use $n = 2$ as an example. When $s = 2$, the time cost of partial shards is 47.5% of that of full shards. When $s = 3$, the time cost of partial shards is 28.7% of that of full shards. When $s = 4$, the time cost of partial shards is 16.6% of that of full shards. When $s = 5$, the time cost of partial shards is 17.1% of that of full shards.

6.2.3 Storage overhead

We compare the storage overhead required by a normal-node and a cross-node in partial sharding model. The number of transactions is in [250,500,750,1000]. We assume that each shard stores the same number of transactions. The results of the experiment are shown in Figure 10. We can see that the storage overhead of a cross-node is much higher than that of a normal-node. With the increase in the number of partitions, the storage overhead of a cross-node will continue to increase. This is because a cross-node will store all the data of the shards, so the storage overhead must be higher than that of a normal-node. When $s = 2$, the storage overhead of a normal-node is 50% of that of a cross-node. When $s = 3$, the storage overhead of the a normal-node is 33.5% of that of a cross-node. When $s = 4$, the storage overhead of a normal-node is 25% of that of a cross-node.

6.2.4 Conclusion of the experiments

With respect to the time cost, for full shards, with the increase of the number of nodes or the number of shards, the time cost becomes larger. For partial sharding, the time cost increases slightly with the number of nodes. When the number of shards involved in cross-chain transactions is more, the time cost of partial shards will be far less than that of full shards. Therefore, partial sharding performs better than full sharding. With the increase in the number of shards, the storage of a cross-node will become larger.

7 Conclusion

We have proposed a credible and adjustable load resource trading framework based on blockchain. The blockchain system uses the master station as the full node and the intelligent interactive terminal/module as the lightweight node. To realize the reliable synchronization of blockchain data, we have proposed a synchronous diffusion mechanism. Combined with the low-voltage side adjustable capacity resource pool and market trading rules, We have constructed the user low-voltage load demand response trading system. A two-level bidding model based on maximizing the interests of users and minimizing the cost of grid companies is proposed. The upper-level model corresponds to the transaction matching stage, and the lower-level model corresponds to the settlement stage. We have tested the proposed method on the blockchain platform, and the experimental results demonstrates the effectiveness of our method.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material further inquiries can be directed to the corresponding author.

Author contributions

WJ, XL, and ZY contributed to conception and design of the study. YX, KZ, BQ, and MZ organized the database and performed the statistical analysis. All authors contributed to manuscript writing, revision, and approved the submitted version.

Funding

This work was supported by the Science and Technology Project of Guangxi Power Grid Co., Ltd (044400KK52200003).

Acknowledgments

The authors would like to thank all of the people who participated in the studies.

Conflict of interest

Authors WJ, ZY, and KZ were employed by the Metrology Center of Guangxi Power Grid Co., Ltd.

The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

References

1. Zheng Z, Xie S, Dai H-N, Chen X, Wang H. Blockchain challenges and opportunities: A survey. *Int J web grid Serv* (2018) 14:352–375. doi:10.1504/IJWGS.2018.10016848

2. Nofer M, Gomber P, Hinz O, Schiereck D. Blockchain. *Business Inf Syst Eng* (2017) 59:183–7. doi:10.1007/s12599-017-0467-3

3. Zhou Z, Wang B, Guo Y, Zhang Y. Blockchain and computational intelligence inspired incentive-compatible demand response in internet of electric vehicles. *IEEE Trans Emerging Top Comput Intelligence* (2019) 3:205–16. doi:10.1109/TETCI.2018.2880693

4. Kang J, Yu R, Huang X, Wu M, Maharjan S, Xie S, et al. Blockchain for secure and efficient data sharing in vehicular edge computing and networks. *IEEE Internet Things J* (2018) 6:4660–70. doi:10.1109/JIOT.2018.2875542

5. Zhang X, Chen X. Data security sharing and storage based on a consortium blockchain in a vehicular ad-hoc network. *IEEE Access* (2019) 7:58241–54. doi:10.1109/ACCESS.2018.2890736

6. Li M, Hu D, Lal C, Conti M, Zhang Z. Blockchain-enabled secure energy trading with verifiable fairness in industrial internet of things. *IEEE Trans Ind Inform* (2020) 16: 6564–74. doi:10.1109/TII.2020.2974537

7. Guo J, Ding X, Wu W. A blockchain-enabled ecosystem for distributed electricity trading in smart city. *IEEE Internet Things J* (2020) 8:2040–50. doi:10.1109/JIOT.2020.3015980

8. Hassan MU, Rehmani MH, Chen J. Deal: Differentially private auction for blockchain-based microgrids energy trading. *IEEE Trans Serv Comput* (2019) 13: 1–275. doi:10.1109/TSC.2019.2947471

9. Chen C, Wu J, Lin H, Chen W, Zheng Z. A secure and efficient blockchain-based data trading approach for internet of vehicles. *IEEE Trans Vehicular Tech* (2019) 68: 9110–21. doi:10.1109/TVT.2019.2927533

10. Gai K, Wu Y, Zhu L, Qiu M, Shen M. Privacy-preserving energy trading using consortium blockchain in smart grid. *IEEE Trans Ind Inform* (2019) 15:3548–58. doi:10.1109/TII.2019.2893433

11. Li Z, Kang J, Yu R, Ye D, Deng Q, Zhang Y. Consortium blockchain for secure energy trading in industrial internet of things. *IEEE Trans Ind Inform* (2017) 14:1–3700. doi:10.1109/TII.2017.2786307

12. Zhang A, Lin X. Towards secure and privacy-preserving data sharing in e-health systems via consortium blockchain. *J Med Syst* (2018) 42:140–18. doi:10.1007/s10916-018-0995-5

13. Malik S, Kanhere SS, Jurdak R. Productchain: Scalable blockchain framework to support provenance in supply chains. In: 2018 IEEE 17th International Symposium on Network Computing and Applications (NCA) (IEEE); 01-03 November 2018; Cambridge, MA, USA (2018). 1–10. doi:10.1109/NCA.2018.8548322

14. Li S, Yu M, Yang C-S, Avestimehr AS, Kannan S, Viswanath P. Polyshard: Coded sharding achieves linearly scaling efficiency and security simultaneously. *IEEE Trans Inf Forensics Security* (2020) 16:249–61. doi:10.1109/TIFS.2020.3009610

15. Cai T, Chen W, Psannis KE, Goudos SK, Yu Y, Zheng Z, et al. Scalable on-chain and off-chain blockchain for sharing economy in large-scale wireless networks. *IEEE Wireless Commun* (2022) 29:32–8. doi:10.1109/MWC.004.2100616

16. Wang E, Cai J, Yang Y, Liu W, Wang H, Yang B, et al. Trustworthy and efficient crowdsensed data trading on sharding blockchain. *IEEE J Selected Areas Commun* (2022) 40:3547–61. doi:10.1109/JSAC.2022.3213331

17. Hong Z, Guo S, Li P, Chen W. Pyramid: A layered sharding blockchain system. In: IEEE INFOCOM 2021-IEEE Conference on Computer Communications (IEEE); 10-13 May 2021; Vancouver, BC, Canada (2021). p. 1–10. doi:10.1109/INFOCOM42981.2021.9488747

18. Fuller JC, Schneider KP, Chassin D. Analysis of residential demand response and double-auction markets. In: 2011 IEEE power and energy society general meeting (IEEE); 24-28 July 2011; Detroit, MI, USA (2011). p. 1–7. doi:10.1109/PES.2011.6039827

19. Ansari M, Al-Awami AT, Sortomme E, Abido M. Coordinated bidding of ancillary services for vehicle-to-grid using fuzzy optimization. *IEEE Trans Smart Grid* (2014) 6:261–70. doi:10.1109/TSG.2014.2341625

20. Li N, Chen L, Dahleh MA. Demand response using linear supply function bidding. *IEEE Trans Smart Grid* (2015) 6:1827–38. doi:10.1109/TSG.2015.2410131

21. Nekouei E, Alpcan T, Chattopadhyay D. Game-theoretic frameworks for demand response in electricity markets. *IEEE Trans Smart Grid* (2014) 6:748–58. doi:10.1109/TSG.2014.2367494

22. Wang Y, Ai X, Tan Z, Yan L, Liu S. Interactive dispatch modes and bidding strategy of multiple virtual power plants based on demand response and game theory. *IEEE Trans Smart Grid* (2015) 7:510–9. doi:10.1109/TSG.2015.2409121

23. Atzeni I, Ordóñez LG, Scutari G, Palomar DP, Fonollosa JR. Noncooperative day-ahead bidding strategies for demand-side expected cost minimization with real-time adjustments: A gnep approach. *IEEE Transactions Signal Process.* (2014) 62:2397–412. doi:10.1109/TSP.2014.2307835

24. Yang H, Zhang S, Qiu J, Qiu D, Lai M, Dong Z. Cvar-constrained optimal bidding of electric vehicle aggregators in day-ahead and real-time markets. *IEEE Trans Ind Inform* (2017) 13:2555–65. doi:10.1109/TII.2017.2662069

25. Wang F, Ge X, Li K, Mi Z. Day-ahead market optimal bidding strategy and quantitative compensation mechanism design for load aggregator engaging demand response. *IEEE Trans Industry Appl* (2019) 55:5564–73. doi:10.1109/TIA.2019.2936183

26. Rassaei F, Soh W-S, Chua K-C. Demand response for residential electric vehicles with random usage patterns in smart grids. *IEEE Trans Sustain Energ* (2015) 6:1367–76. doi:10.1109/TSTE.2015.2438037

27. Vlachos AG, Biskas PN. Demand response in a real-time balancing market clearing with pay-as-bid pricing. *IEEE Trans Smart Grid* (2013) 4:1966–75. doi:10.1109/TSG.2013.2256805

28. Rassaei F, Soh W-S, Chua K-C. Distributed scalable autonomous market-based demand response via residential plug-in electric vehicles in smart grids. *IEEE Trans Smart Grid* (2016) 9:3281–90. doi:10.1109/TSG.2016.2629515

29. Xu B, Shi Y, Kirschen DS, Zhang B. Optimal battery participation in frequency regulation markets. *IEEE Trans Power Syst* (2018) 33:6715–25. doi:10.1109/TPWRS.2018.2846774

30. Sortomme E, El-Sharkawi MA. Optimal charging strategies for unidirectional vehicle-to-grid. *IEEE Trans Smart Grid* (2010) 2:131–8. doi:10.1109/TSG.2010.2090910

31. Tang W, Yang H-T. Optimal operation and bidding strategy of a virtual power plant integrated with energy storage systems and elasticity demand response. *IEEE Access* (2019) 7:79798–809. doi:10.1109/ACCESS.2019.2922700

32. Song M, Amelin M. Purchase bidding strategy for a retailer with flexible demands in day-ahead electricity market. *IEEE Trans Power Syst* (2016) 32:1839–50. doi:10.1109/TPWRS.2016.2608762

33. Thatte AA, Xie L, Viassolo DE, Singh S. Risk measure based robust bidding strategy for arbitrage using a wind farm and energy storage. *IEEE Trans Smart Grid* (2013) 4:2191–9. doi:10.1109/TSG.2013.2271283

34. Hedman KW, O'Neill RP, Fisher EB, Oren SS. Smart flexible just-in-time transmission and flowgate bidding. *IEEE Trans Power Syst* (2010) 26:93–102. doi:10.1109/TPWRS.2010.2047660

35. Chen S, Chen Q, Xu Y. Strategic bidding and compensation mechanism for a load aggregator with direct thermostat control capabilities. *IEEE Trans Smart Grid* (2016) 9: 1–2336. doi:10.1109/TSG.2016.2611611

36. Rahimiyan M, Baringo L. Strategic bidding for a virtual power plant in the day-ahead and real-time markets: A price-taker robust optimization approach. *IEEE Trans Power Syst* (2015) 31:2676–87. doi:10.1109/TPWRS.2015.2483781

37. Dallinger D, Krampe D, Wietschel M. Vehicle-to-grid regulation reserves based on a dynamic simulation of mobility behavior. *IEEE Trans smart grid* (2011) 2:302–13. doi:10.1109/TSG.2011.2131692

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 16 February 2023
DOI 10.3389/fphy.2023.1133745

OPEN ACCESS

EDITED BY
Shiping Wen,
University of Technology Sydney, Australia

REVIEWED BY
Xianming Wu,
Guizhou University, China
Long Jin,
Lanzhou University, China

*CORRESPONDENCE
Lv Zhao,
iadozhao@foxmail.com
Xiaolei Yang,
120135472@qq.com

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 29 December 2022
ACCEPTED 31 January 2023
PUBLISHED 16 February 2023

CITATION
Zhao L, Shao H, Yang X, Liu X, Tang Z and Lin H (2023), A novel zeroing neural network for dynamic sylvester equation solving and robot trajectory tracking. *Front. Phys.* 11:1133745. doi: 10.3389/fphy.2023.1133745

A novel zeroing neural network for dynamic sylvester equation solving and robot trajectory tracking

Lv Zhao[1,2]*, Huaiyuan Shao[1], Xiaolei Yang[3]*, Xin Liu[1], Zhijun Tang[1] and Hairong Lin[4]

[1]School of Information and Electrical Engineering, Hunan University of Science and Technology, Xiangtan, China, [2]School of Information Engineering, Changsha Medical University, Changsha, China, [3]Guangxi University of Finance and Economics, Nanning, China , [4]Hunan University, Changsha, China

To solve the theoretical solution of dynamic Sylvester equation (DSE), we use a fast convergence zeroing neural network (ZNN) system to solve the time-varying problem. In this paper, a new activation function (AF) is proposed to ensure fast convergence in predefined times, as well as its robustness in the presence of external noise perturbations. The effectiveness and robustness of this zeroing neural network system is analyzed theoretically and verified by simulation results. It was further verified by the application of robotic trajectory tracking.

KEYWORDS

dynamic sylvester equation, zeroing neural network, predefined-time convergence, robustness, activation function, robot trajectory tracking

1 Introduction

Recurrent neural networks (RNN) have been extensively studied and applied in many scientific and engineering fields in the past two decades, such as automatic control theory [1], image processing [2, 3], data processing [4], and matrix equations solving [5–7]. The Sylvester equation plays a very important role in mathematics and control fields, and it will be involved in many practical applications [8–12].

The theoretical solution of the general Sylvester equation can be obtained through the conventional gradient descent method [13]. Using the norm of the error matrix as a performance indicator, a neural network is evolved along the gradient descent direction so that the error norm in the fixed-constant case vanishes to zero over time [14]. However, in the time-varying case, due to the lack of velocity compensation for the time-varying parameters, the error norm may not converge to zero, even after an infinitely long time. To solve the problem of solving the dynamic Sylvester equations, a design method based on zeroing neural network system is adopted. Zeroing neural network is a special recurrent neural network proposed by Zhang et al. [6, 15], and plays a crucial role in solving various dynamic problem fields. Generally speaking, zeroing neural networks have better accuracy and higher efficiency than recurrent neural networks based on gradient descent in solving time-varying problems [16, 17]. In recent years, ZNN has been studied more and more, and many new models have been developed. For example, the VP-CZNN in Ref. [5] and the FT-VP-CZNN in Ref. [18] realize super exponential convergence, and the ZNN model activated by SBPAF in Ref. [19] realizes finite-time convergence.

There are often many kinds of external noise interference in real life, and most neural network systems do not consider the tolerance problem of noise, which reduces the effect of dynamic systems. Through the above discussion analysis, a new activation function can be introduced to improve the performance of the whole system. At the same time, it can also ensure the predefined time convergence and noise resistance.

The rest of this paper is organized as follows. In Section 2, we introduce zeroing neural network model and dynamic Sylvester equation. In Section 3, we review some activation functions and propose a new activation function with predefined time convergence under noise interference. In Section 4, we focus on the predefined-time convergence and anti-noise capability of the model that we established. In Section 5, we verify the theory through simulation experiments of solving the dynamic Sylvester equations and the trajectory tracking problem of the robotic arm. Finally, we give a brief conclusion to the paper in Section 6. In addition, all the abbreviations used in this work are listed in the following Table 1.

2 ZNN model and dynamic sylvester equation

In this work, we are concerned with the Sylvester equation, and its general expression is directly given as:

$$A(t)X(t) - X(t)B(t) = -C(t) \in R^{n\times n} \tag{1}$$

where $A(t) \in R^{n\times n}, B(t) \in R^{n\times n}, C(t) \in R^{n\times n}$ stand for known coefficient matrices, and $X(t) \in R^{n\times n}$ stands for an unknown matrix we need to solve.

To solve the Sylvester equation, we define an error function:

$$E(t) = A(t)X(t) - X(t)B(t) + C(t) \in R^{n\times n} \tag{2}$$

If each elements of the error function E(t) converges to 0, we can obtain the theoretical solution X(t). To make the error function converge to 0, the ZNN design formula is designed by the following derivative equation:

$$\frac{d(E(t))}{d(t)} = -\gamma\varphi(E(t)) \tag{3}$$

where $\varphi(\cdot) \in R^{n\times n}$ stands for an array of the activation function, and $\gamma > 0$ are a known adjustable parameter which adjust the convergence rate.

The time derivative of the error function is defined as:

TABLE 1 All the abbreviations used in this work.

Full names	Abbreviations
Dynamic Sylvester equation	DSE
Second-order dynamic Sylvester equation	SDSE
Third-order dynamic Sylvester equation	TDSE
Zeroing neural network	ZNN
Recurrent neural network	RNN
Activation functions	AF
Linear activation function	LAF
Power-sigmoid activation function	PSAF
New activation function	NAF
State solution	SS
Theoretical solution	TS

$$\dot{E}(t) = A(t)\dot{X}(t) + \dot{A}(t)X(t) - \dot{X}(t)B(t) - X(t)\dot{B}(t) + \dot{C}(t) \tag{4}$$

Then, by substituting error function into the ZNN design formula and considering that the time derivative of the error function, the following ZNN model for Sylvester equation is derived:

$$\begin{aligned} A(t)\dot{X}(t) - \dot{X}(t)B(t) = &-\dot{A}(t)X(t) + X(t)\dot{B}(t) - \dot{C}(t) \\ &-\gamma\varphi(A(t)X(t) - X(t)B(t) + C(t)) \end{aligned} \tag{5}$$

3 Activation function

The activation function is a monotonically increasing odd function and is a key component of the ZNN model. It has important implications for the convergence and robustness of the ZNN models. Especially in the presence of noise interference, the selection of an appropriate activation function can play a great positive role in the ZNN model. In the past decade, many activation functions have been proposed to improve the performance of neural networks, and different activation functions can be found to have different performance through comparison [20–22].

Several common activation functions.

1) Linear activation function (LAF)

$$\phi(x) = x \tag{6}$$

2) Power activation function

$$\phi(x) = x^p \tag{7}$$

with $p \geq 3$ indicating an odd integer.

3) Power-sigmoid activation function (PSAF)

$$\phi(x) = \begin{cases} x^p & ,|x| \geq 1 \\ \dfrac{1+\exp(-\xi)}{1-\exp(-\xi)}\dfrac{1-\exp(-\xi x)}{1+\exp(-\xi x)} & ,|x| < 1 \end{cases} \tag{8}$$

4) Sign-bi-power activation function

$$\phi(x) = \frac{1}{2}\mathrm{sgn}^{\xi_1}(x) + \frac{1}{2}\mathrm{sgn}^{\xi_2}(x) \tag{9}$$

where $\xi_1 \in (0,1)$ and $\xi_2 > 1$, $\mathrm{sgn}(\cdot)$ is defined as:

$$\mathrm{sgn}^n(x) = \begin{cases} |x|^n, x > 0 \\ 0, x = 0 \\ -|x|^n, x < 0 \end{cases} \tag{10}$$

Different activation functions have different convergence performance, and non-linear activation function generally possesses better performance than the linear activation function in the rate of the accelerated ZNN model convergence. All of the above activation functions can effectively improve the convergence rate, but they do not consider the noise factor. The convergence performance is greatly reduced in the presence of noise interference. Considering this factor, we propose a new activation function (NAF) with predefined time convergence under noise interference:

$$\phi(x) = \left(k_1|x|^p + k_2|x|^{\frac{1}{p}} + k_3\right)\mathrm{sgn}(x) + k_4x^{2\lambda-1} \tag{11}$$

where $0 < p < 1, k_1 > 0, k_2 > 0, k_3 > 0, k_4 > 0, \lambda \in N+$, and $\mathrm{sgn}(\cdot)$ is defined as described above.

4 Network model and theoretical analysis

Before the main theoretical results of the ZNN are given, the following lemma is first presented as a basis for further discussion [5].

Lemma. There is a non-linear dynamical system such as $\dot{X}(t) = g(x(t), t)$, $t \in [0, +\infty)$, where g(t) is a non-linear function. If a continuous radial unbounded function V: $R^n \to R_+ \cup \{0\}$ exists, making $U(\zeta) = 0$ and any solution satisfied:

$$\dot{U}(t) \le -\tau U^{\varsigma}(\zeta(t)) - \rho U^{\mu}(\zeta(t)) \tag{12}$$

where $\tau > 0, \rho > 0, 0 < \varsigma < 1, \mu > 0$ all are constants.

The predefined convergence time is:

$$T_{\max} = \frac{1}{\tau(1-\varsigma)} + \frac{1}{\rho(\mu-1)} \tag{13}$$

Theorem 1. Starting with the random initial matrix, the exact solution of the ZNN model predefined convergence is as:

$$t_c \le \frac{1}{\gamma k_1 (1-p)} + \frac{1}{\gamma k_2 (1/p - 1)} \tag{14}$$

where γ, k_1, k_2, p stand for the preset parameters.

Proof of Theorem 1. The ZNN model can be represented as $\dot{E}(t) = -\gamma\phi(E(t))$, where $E(t)$ represents error function, and subsystem (i, j) can be expressed as:

$$\dot{e}_{i,j}(t) = -\gamma\phi\left(e_{i,j}(t)\right) \tag{15}$$

with $i, j \in \{1, 2, ..., n\}$. If the subsystem possesses predefined time stability, the ZNN model also possesses it. To prove the stability of the predefined time, we define a Lyapunov function as:

$$u(t) = \left|e_{i,j}(t)\right| \tag{16}$$

The time derivative is:

$$\begin{aligned}\dot{u}(t) &= -\dot{e}_{i,j}(t)\mathrm{sgn}\left(e_{i,j}(t)\right) = -\gamma\phi\left(e_{i,j}(t)\right)\mathrm{sgn}\left(e_{i,j}(t)\right)\\ &= -\gamma\left(k_1\left|e_{i,j}(t)\right|^{p} + k_2\left|e_{i,j}(t)\right|^{1/p} + k_4\left|e_{i,j}(t)\right|^{2\lambda-1} + k_3\right)\\ &\le -\gamma\left(k_1\left|e_{i,j}(t)\right|^{p} + k_2\left|e_{i,j}(t)\right|^{1/p}\right)\\ &= -\gamma\left(k_1 u^{p}(t) + k_2 u^{1/p}(t)\right)\end{aligned} \tag{17}$$

Comparing with the lemma, predefined convergence time is available:

$$t_c \le \frac{1}{\gamma k_1 (1-p)} + \frac{1}{\gamma k_2 (1/p - 1)} \tag{18}$$

Because there may be various disturbances in reality, the following ZNN model perturbed by noise is studied:

$$\begin{aligned}A(t)\dot{X}(t) - \dot{X}(t)B(t) = &-\dot{A}(t)X(t) + X(t)\dot{B}(t) - \dot{C}(t)\\ &-\gamma\varphi(A(t)X(t) - X(t)B(t) + C(t)) + Y(t)\end{aligned} \tag{19}$$

where Y(t) represents an additional noise.

Case 1. Dynamic bounded vanishing noise.

Theorem 2. There is a dynamically bounded vanishing noise, where its elements (i, j) satisfy $|y_{i,j}(t)| \le \delta|e_{i,j}(t)|$ and $\delta \in (0, +\infty)$, $|e_{i,j}(t)|$ is the absolute value of the element (i, j) in the error function E (t). Starting with a random initial matrix $X(0) \in R^{n\times n}$, If it satisfy $\gamma k_4 \ge \delta$, it will output exact solutions of the time-varying Sylvester equation at the predefined time t_c.

$$t_c \le \frac{1}{\gamma k_1 (1-p)} + \frac{1}{\gamma k_2 (1/p - 1)} \tag{20}$$

Proof of Theoreim 2. The ZNN model which is perturbed by noise can be reduced to $\dot{E}(t) = -\gamma\phi(E(t)) + Y(t)$, The subsystem (i, j) is expressed as:

$$\dot{e}_{i,j}(t) = -\gamma\phi\left(e_{i,j}(t)\right) + y_{i,j}(t) \tag{21}$$

where $y_{i,j}(t)$ stands for the element (i, j) of matrix Y(t).

To prove the predefined time stability of the subsystem subject to noise perturbation, we define a Lyapunov function as:

$$u(t) = \left[e_{i,j}(t)\right]^2 \tag{22}$$

The time derivative is:

$$\begin{aligned}\dot{u}(t) &= 2e_{i,j}(t)\dot{e}_{i,j}(t) = 2e_{i,j}(t)\left(-\gamma\phi(e_{i,j}(t)) + y_{i,j}(t)\right)\\ &= -2\gamma\left(k_1\left|e_{i,j}(t)\right|^{p+1} + k_2\left|e_{i,j}(t)\right|^{1/p+1}\right) - 2\gamma k_3\left|e_{i,j}(t)\right|\\ &\quad +2\left(e_{i,j}(t)y_{i,j}(t) - \gamma k_4\left|e_{i,j}(t)\right|^{2(2\lambda-1)}\right)\\ &\le -2\gamma\left(k_1\left|e_{i,j}(t)\right|^{p+1} + k_2\left|e_{i,j}(t)\right|^{1/p+1}\right)\\ &\quad +2\left(\delta\left|e_{i,j}(t)\right|^{2} - \gamma k_4\left|e_{i,j}(t)\right|^{2(2\lambda-1)}\right)\\ &\le -2\gamma\left(k_1\left|e_{i,j}(t)\right|^{p+1} + k_2\left|e_{i,j}(t)\right|^{1/p+1}\right)\\ &= -2\gamma\left(k_1 u^{(p+1)/2}(t) + k_2 u^{(1/p+1)/2}(t)\right)\end{aligned} \tag{23}$$

with λ is positive integer. Therefore, $2(2\lambda - 1)$ is certainly greater than or equal to 2. If it satisfy $\gamma k_4 \ge \delta$, the upper formula must hold.

Comparing with the lemma, predefined convergence time is available:

$$t_c \le \frac{1}{\gamma k_1 (1-p)} + \frac{1}{\gamma k_2 (1/p - 1)} \tag{24}$$

Case 2. Dynamic bounded non-vanishing noise.

Theorem 3. There exists a dynamically bounded non-vanishing noise, where its elements (i, j) satisfy $|y_{i,j}(t)| \le \delta$, and $\delta \in (0, +\infty)$. Starting with a random initial matrix, if it satisfy $\gamma k_3 \ge \delta$, it will output exact solutions of the time-varying Sylvester equation at the predefined time t_c.

$$t_c \le \frac{1}{\gamma k_1 (1-p)} + \frac{1}{\gamma k_2 (1/p - 1)} \tag{25}$$

Proof of Theorem 3. The additive noise is just different, comparing with Theorem 2. Therefore, we still choose the following Lyapunov function:

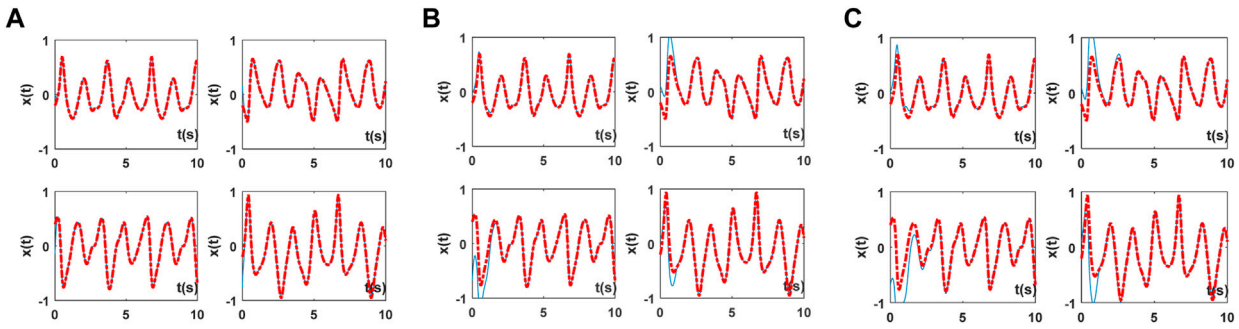


FIGURE 1
The comparison of the state solution and the theoretical solution: **(A)** SS of NAF for solving SDSE; **(B)** SS of PSAF for solving SDSE; **(C)** SS of LAF for solving SDSE.

$$u(t) = \left[e_{i,j}(t)\right]^2 \tag{26}$$

The time derivative is:

$$\begin{aligned}
\dot{u}(t) &= 2e_{i,j}(t)\dot{e}_{i,j}(t) = 2e_{i,j}(t)\left(-\gamma\phi\left(e_{i,j}(t)\right) + y_{i,j}(t)\right) \\
&= -2\gamma\left(k_1\left|e_{i,j}(t)\right|^{p+1} + k_2\left|e_{i,j}(t)\right|^{1/p+1}\right) - 2\gamma k_4\left|e_{i,j}(t)\right|^{2(2\lambda-1)} \\
&\quad +2\left(e_{i,j}(t)y_{i,j}(t) - \gamma k_3\left|e_{i,j}(t)\right|\right) \\
&\leq -2\gamma\left(k_1\left|e_{i,j}(t)\right|^{p+1} + k_2\left|e_{i,j}(t)\right|^{1/p+1}\right) \\
&\quad +2\left(\delta\left|e_{i,j}(t)\right| - \gamma k_3\left|e_{i,j}(t)\right|\right) \\
&\leq -2\gamma\left(k_1\left|e_{i,j}(t)\right|^{p+1} + k_2\left|e_{i,j}(t)\right|^{1/p+1}\right) \\
&= -2\gamma\left(k_1 u^{(p+1)/2}(t) + k_2 u^{(1/p+1)/2}(t)\right)
\end{aligned} \tag{27}$$

If it satisfy $\gamma k_3 \geq \delta$, the above formula is established.

Comparing with the lemma, predefined convergence time is available:

$$t_c \leq \frac{1}{\gamma k_1(1-p)} + \frac{1}{\gamma k_2\left(1/p - 1\right)} \tag{28}$$

According to the above discussion and analysis, it can be concluded that the proposed network model can effectively achieve the predefined time convergence, and can attain better performance in environments where noise disturbance exists.

5 Simulation effect verification

A network model based on a novel activation function is proposed above. Unlike the common activation function, this activation function can achieve predefined time convergence in a noisy environment. And we perform a convergence analysis for different types of noise. Then we will verify the convergence effect and compare it with the simulation effect of other common activation functions. Firstly, we apply the contrast by simulations of time-varying Sylvester equations.

5.1 Simulation 1: second order coefficient matrix

Generally, coefficient matrix A(t), B(t) and C(t) can be chosen randomly, as long as A(t) and B(t) are both invertible matrix. For no loss of generality, the following second-order matrices are selected:

$$A(t) = \begin{pmatrix} \sin(2t) & \cos(4t) \\ -\cos(4t) & \sin(3t) \end{pmatrix}, B(t) = \begin{pmatrix} 3 & 0 \\ 1 & 2 \end{pmatrix},$$
$$C(t) = \begin{pmatrix} \sin(4t) & \cos(4t) \\ -\cos(4t) & \sin(4t) \end{pmatrix}$$

For random initial state $X(0) \in [2,2]^{2\times 2}$, the network model with the classical linear activation function, the power-sigmoid activation function, and the proposed activation function is simulated in a noise-free environment.

The results are presented in Figure 1. The following figures are the comparison of the state solution (SS) and the theoretical solution (TS) of the three activation functions in a noiseless environment, where the red dashed line represents the theoretical solution, and the blue solid line indicates the state solution. Each group of status figure is composed of four components, representing the four elements $(x_{11}\ x_{12}; x_{21}\ x_{22})$ in the matrix in turn.

The first set of graphs shows a comparison of the state solution produced by the newly proposed activation function with the theoretical solution. And the second and third sets of graphs represent the state contrast figures which are produced by the power-sigmoid activation function and linear activation function. It is obvious from the three groups that the state solutions produced by the newly proposed activation function are closer to the theoretical values.

The following sets of graphs in Figure 2 show the residual graphs generated by the three activation functions in different noise environments, where the blue line represents the newly proposed activation function, the red line represents the power-sigmoid activation function, and the yellow line represents the linear activation function. The first figure shows the residual figure in a noiseless environment; the remaining two figures are the residual figures in the presence of an external noise of y(t) = 0.3t and y (t) = cos (t). Obviously, the newly proposed activation function can achieve faster convergence and possesses a better anti-noise interference ability.

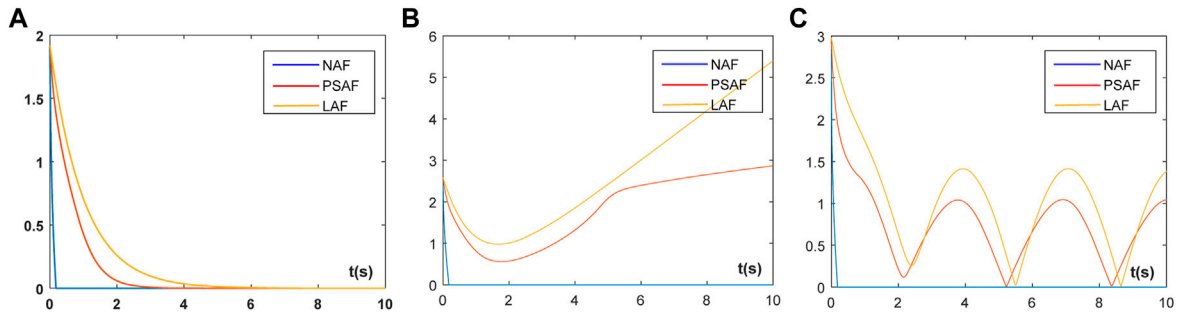


FIGURE 2
Residual graphs generated by the three activation function: **(A)** Simulated residual errors of ZNN for solving SDSE without noise; **(B)** Simulated residual errors of ZNN for solving SDSE with y(t) = 0.3t; **(C)** Simulated residual errors of ZNN for solving SDSE with y(t) = cos(t).

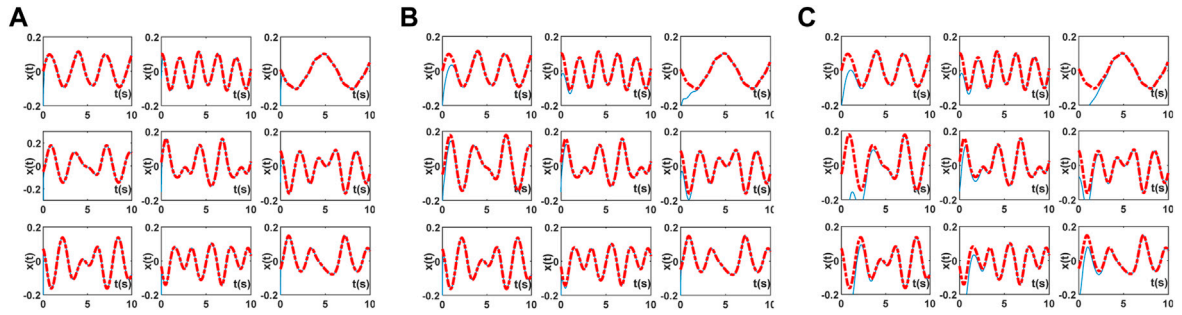


FIGURE 3
The comparison of the state solution and the theoretical solution: **(A)** SS of NAF for solving TDSE; **(B)** SS of PSAF for solving TDSE; **(C)** SS of LAF for solving TDSE.

5.2 Simulation 2: Third order coefficient matrix

The principle as above, coefficient matrix A(t), B(t) and C(t) are chosen randomly, as long as A(t) and B(t) are both invertible matrix. For no loss of generality, select following random third-order matrices:

$$A(t) = \begin{pmatrix} \sin(2t) & \sin(2t)\cos(t) \\ \cos(3t) & \sin(2t) - \sin(2t) \\ \sin(t) & \cos(2t) \quad \sin(2t) \end{pmatrix}, B(t) = \begin{pmatrix} 10 & 0 & 5 \\ 0 & 20 & 10 \\ 0 & 15 & 20 \end{pmatrix}$$

$$C(t) = \begin{pmatrix} \sin(2t) & \sin(2t)\cos(3t) \\ \cos(3t) & \sin(2t) - \sin(3t) \\ -\sin(t)\cos(3t) & \sin(2t) \end{pmatrix}$$

For random initial state $X(0) \in [1, 1]^{3\times3}$, the network model with the classical linear activation function, the power-sigmoid activation function, and the proposed activation function is simulated in the presence of noise (Y(t) = 0.5t) environment.

The results are presented in Figure 3. The figure shows the comparison of the state solutions and the theoretical solutions of the three activation functions in this environment, where the red dashed line represents the theoretical solution, and the blue solid line indicates the state solution. Each group of status figure is composed of nine components, representing the nine elements $(x_{11}\ x_{12}\ x_{13}; x_{21}\ x_{22}\ x_{23}; x_{31}\ x_{32}\ x_{33})$ in the matrix in turn.

The first set of graphs shows a comparison of the state solution produced by the newly proposed activation function with the theoretical solution. The second and third sets of graphs represent the state contrast figures which are produced by the power-sigmoid activation function and linear activation function. It is obvious from the three groups that the state solutions produced by the newly proposed activation function are closer to the theoretical values.

The supra sets of graphs in Figure 4 show the residual graphs generated by the three activation functions in different noise environments, where the blue line represents the newly proposed activation function, the red line represents the power-sigmoid activation function, and the yellow line represents the linear activation function. The first figure shows the residual figure in a noiseless environment; the remaining two figures are the residual figures in the presence of an external noise of y(t) = 0.3t and y (t) = cos(t). Obviously, the newly proposed activation function can achieve faster convergence and possesses better anti-noise interference ability.

5.3 Simulation 3: Mechanical arm trajectory tracking

The joint angle vector θ of the robotic arm and the actual trajectory L of the end actuator are related to:

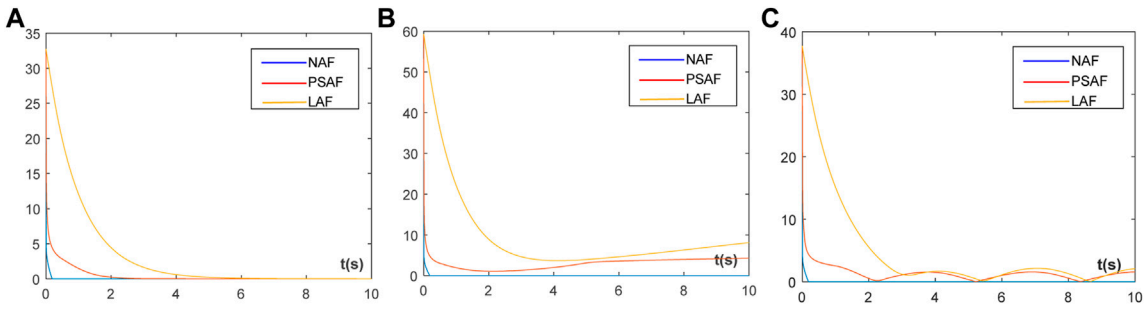


FIGURE 4
Residual graphs generated by the three activation function: **(A)** Simulated residual errors of ZNN for solving TDSE without noise; **(B)** Simulated residual errors of ZNN for solving TDSE with y(t) = 0.3t; **(C)** Simulated residual errors of ZNN for solving TDSE with y(t) = cos(t).

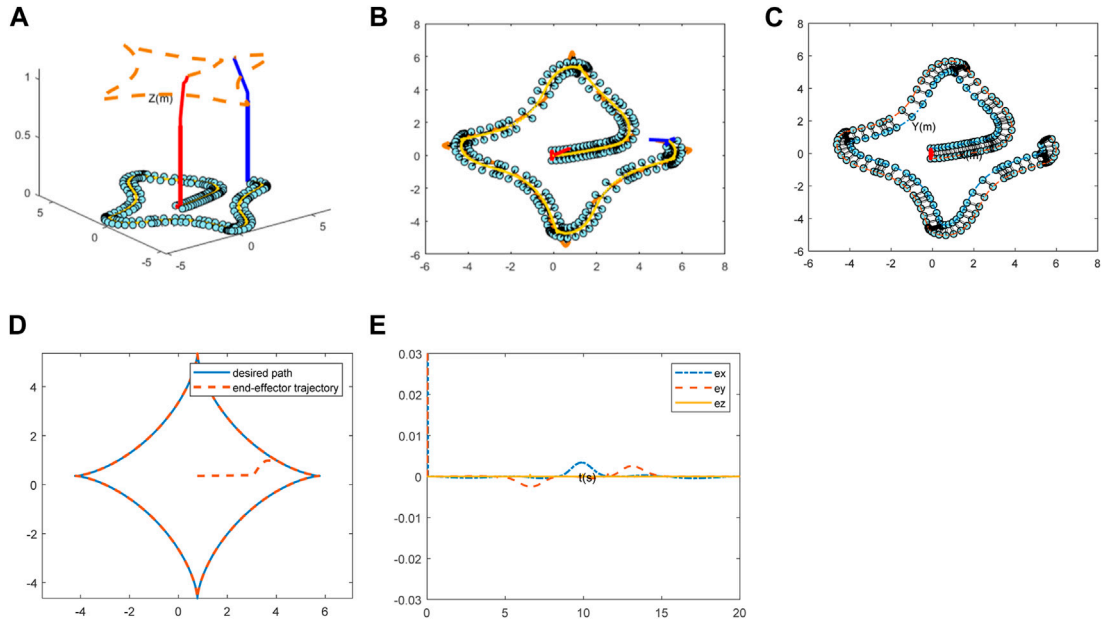


FIGURE 5
Simulations of mechanical arm tracking trajectory: **(A)** The whole track schematic; **(B)** Top plot of the traced trajectory; **(C)** Movement trajectory of the mobile platform; **(D)** Expected path and actual trajectory contrast; **(E)** Tracking error.

$$L = f(\theta) \tag{29}$$

where $f(\cdot)$ is a known non-linear one-to-many function between the joint angle and the actual trajectory of end-effector. Given the movement path L of the end-effector, the motion change of the corresponding joint angle vector θ can be determined.

From the time-varying view, the equations of motion of the robotic arm can be expressed as:

$$L(t) = f(\theta(t)) \tag{30}$$

The track-tracking problem eventually turns into a speed problem. Therefore, the two sides of the above equation are guided to obtain the motion equation of the mechanical arm at the velocity level:

$$\dot{L}(t) = J(\theta)\dot{\theta}(t) \tag{31}$$

where $J(\theta)$ represents a Jacobian matrix $J(\theta) = \partial f(\theta)/\partial\theta$. After introducing the Jacobian matrix, the time-varying joint angular velocity $\dot{\theta}(t)$ and the mechanical arm end time-varying velocity $\dot{L}(t)$ can be regarded as a linear relationship. Generally, $\dot{L}(t)$ is known quantity and $\dot{\theta}(t)$ is unknown quantity. Thus, the upper equation belongs to an inversion equation of motion and we can consider the trajectory tracking problem as a problem of solving a time-varying linear matrix equation.

The proposed network model is used to solve the time-varying problem and to construct the corresponding model:

$$J(\theta)\dot{\theta}(t) = \dot{L}(t) - \gamma\phi(L(t) - J(\theta)\theta(t)) + y(t) \tag{32}$$

where y(t) is additive noise and set to y(t) = 0.1t.

The expected tracking trajectory of the mechanical arm is a four-pointed star type, and the simulations in noisy environments are shown in Figure 5. The first graph in Figure 5 depicts the whole track schematic of the mechanical arm, and the second figure illustrates traced trajectory of it from the top perspective. The movement trajectory of the mobile platform is described in (c). Furthermore, a contrast drawing between expected

path and actual trajectory is exhibited subsequently. Finally, the tracking error of this test in three dimensions is captured in (e). Therefore, the novel neural network model can accurately complete the trajectory tracking with small error under the noise polluted environment.

6 Conclusion

In this paper, we propose a zeroing neural network model by introducing a novel activation function. Through the theoretical analysis and simulation verification, the network model possesses the characteristics of predefined time convergence and strong noise resistance. In dealing with the problem of solving the dynamical Sylvester equations, it has faster convergence rate, higher accuracy and better robustness, comparing with several classical network models constructed with the activation functions. Moreover, the effectiveness and reliability of the network model have been validated by theoretical analysis and simulation in the robotic arm trajectory tracking problem.

In addition, there are some difficulties in future research. On the one hand, in the interest of improving the effectiveness of ZNN model, the structure require further optimized, such as designing other new outstanding activation functions and convergence factor. On the other hand, for the sake of improving the practicality of the ZNN model, it is necessary to expand the practical application scope of the model to other scientific and engineering fields, such as applying it to time-varying electronic circuits, chaotic systems, multi-agent research, chaotic systems.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding authors.

Author contributions

Conceptualization, LZ and HS; methodology, HS; validation, LZ, XL, and XY; formal analysis, LZ; writing—original draft preparation, HS; writing—review and editing, LZ and XL; visualization, XY; supervision, ZT and HL. All authors have read and agreed to the published version of the manuscript.

Funding

This research was funded by National Natural Science Foundation of China, grant number 61875054; Natural Science Foundation of Hunan Province, grant number 2020JJ4315 and 2020JJ5199; Scientific Research Fund of Hunan Provincial Education Department, grant number 20B216 and 20C0786.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

References

1. Li S, Zhang Y, Jin L. Kinematic control of redundant manipulators using neural networks. *IEEE Trans Neural Netw Learn Syst* (2017) 28:2243–54. doi:10.1109/tnnls.2016.2574363

2. Nazemi A. A capable neural network framework for solving degenerate quadratic optimization problems with an application in image fusion. *Neural Process Lett* (2017) 47:167–92. doi:10.1007/s11063-017-9640-4

3. Yu. F, Kong X, Mokbel AAM, Yao. W, Cai S. Complex dynamics, hardware implementation and image encryption application of multiscroll memeristive Hopfield neural network with a novel local active memeristor. *IEEE Trans Circuits Systems–II: Express Briefs.* (2023) 70:326–30. doi:10.1109/tcsii.2022.3218468

4. Yu. F, She H, Yu. Q, Kong X, Sharma PK, Cai S. Privacy protection of medical data based on multi-scroll memristive Hopfield neural network. In: IEEE Transactions on Network Science and Engineering; 23 November 2022 (2022). doi:10.1109/TNSE.2022.3223930

5. Li W, Liao B, Xiao L, A recurrent neural network with predefined-time convergence and improved noise tolerance for dynamic matrix square root finding. *Neurocomputing* (2019) 337:262–73. doi:10.1016/j.neucom.2019.01.072

6. Zhao L, Jin J, Gong J. Robust zeroing neural network for fixed-time kinematic control of wheeled mobile robot in noise-polluted environment. *Math Comput Simul* (2021) 185:289–307. doi:10.1016/j.matcom.2020.12.030

7. Jin J, Chen W, Chen C, Chen Z, Tang L, Chen L, et al. A predefined fixed-time convergence ZNN and its applications to time-varying quadratic programming solving and dual-arm manipulator cooperative trajectory tracking. *IEEE Trans. Ind. Inform.* (2022). doi:10.1109/TII.2022.3220873

8. Jian Z, Xiao L, Li K, Zuo Q, Zhang Y. Adaptive coefficient designs for nonlinear activation function and its application to zeroing neural network for solving time-varying Sylvester equation. *J Frankl Inst.-Eng Appl Math* (2020) 357:9909–29. doi:10.1016/j.jfranklin.2020.06.029

9. Zhang Z, Zheng L, Weng. J, Mao. Y, Lu W, Xiao L. A new varying-parameter recurrent neural-network for online solution of timevarying Sylvester equation. *IEEE Trans Cybern* (2018) 48:3135–48. doi:10.1109/tcyb.2017.2760883

10. Song C, Feng J, Wang. X, Zhao J. Finite iterative method for solving coupled Sylvester-transpose matrix equations. *J Appl Math Comput* (2014) 46:351–72. doi:10.1007/s12190-014-0753-x

11. Jin J, Xiao L, Lu M, Li J. Design and analysis of two FTRNN models with application to time-varying Sylvester equation. *IEEE Access* (2019) 7:58945–50. doi:10.1109/access.2019.2911130

12. Jin J, Zhao L, Chen L, Chen W. A robust zeroing neural network and its applications to dynamic complex matrix equation solving and robotic manipulator trajectory tracking. *Front. in Neurorobotics* (2022) 16:1065256.

13. Jin J, Zhu J, Zhao L, Chen L, Chen L, Gong J. A robust predefined-time convergence zeroing neural network for dynamic matrix inversion. *IEEE T Cybern* (2022) 2022:1–14. doi:10.1109/TCYB.2022.3179312

14. Zhang H. Gradient-based iteration for a class of matrix equations. In: The 26th Chinese Control and Decision Conference (2014 CCDC); 31 May 2014 - 02 June 2014; Changsha (2014). p. 1201–5.

15. Zhao L, Jin J, Gong J. A novel robust fixed-time convergent zeroing neural network for solving time-varying noise-polluted nonlinear equations. *Int J Comput Math* (2021) 98:2514–32. doi:10.1080/00207160.2021.1902512

16. Jin J, Zhao L, Li M, Yu. F Improved zeroing neural networks for finite time solving nonlinear equations. *Neural Comput Appl* (2020) 32:4151–60. doi:10.1007/s00521-019-04622-x

17. Yu. F, Liu L, Xiao L, Li K, Cai S. A robust and fixed-time zeroing neural dynamics for computing time-variant nonlinear equation using a novel nonlinear activation function. *Neurocomputing* (2019) 350:108–16. doi:10.1016/j.neucom.2019.03.053

18. Zhang Z, Lu Y, Zheng L, Li S, Yu. Z, Li Y. A new varying-parameter convergent-differential neural-network for solving time-varying convex QP problem constrained by linear-equality. *IEEE Trans Automatic Control* (2018) 63:4110–25. doi:10.1109/tac.2018.2810039

19. Jin J, Chen W, Qiu L, Zhu H, Liu H. A noise tolerant parameter-variable zeroing neural network and its applications. *Mathematics and Computers in Simulation* (2023) 207:482–498.

20. Shen Y, Miao P, Huang Y, Shen Y. Finite-time stability and its application for solving time-varying Sylvester equation by recurrent neural network. *Neural Process Lett* (2015) 42:763–84. doi:10.1007/s11063-014-9397-y

21. Qiu B, Zhang Y, Yang Z. New discrete-time ZNN models for least-squares solution of dynamic linear equation system with time-varying rank-deficient coefficient. *IEEE Trans Neural Netw Learn Syst.* (2018) 29:5767–76. doi:10.1109/tnnls.2018.2805810

22. Xiao L. Design and analysis of robust nonlinear neural dynamics for solving dynamic nonlinear equation within finite time. *Nonlin Dyn* (2019) 96:2437–47. doi:10.1007/s11071-019-04932-8

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 17 February 2023
DOI 10.3389/fphy.2023.1135739

OPEN ACCESS

EDITED BY
Hairong Lin,
Hunan University, China

REVIEWED BY
Ho Ching Iu,
University of Western Australia, Australia
Xiaoyuan Wang,
Hangzhou Dianzi University, China

*CORRESPONDENCE
Hui Li,
lihui@ccut.edu.cn

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 01 January 2023
ACCEPTED 16 January 2023
PUBLISHED 17 February 2023

CITATION
Liu Y, Liu F, Luo W, Wu A and Li H (2023), AC power analysis for second-order memory elements.
Front. Phys. 11:1135739.
doi: 10.3389/fphy.2023.1135739

AC power analysis for second-order memory elements

Yue Liu[1], Fang Liu[1], Wanbo Luo[2], Aoyun Wu[1] and Hui Li[1]*

[1]College of Electrical and Electronic Engineering, Changchun University of Technology, Changchun, China, [2]Changchun Veterinary Research Institute, Chinese Academy of Agricultural Sciences, Changchun, China

As the product of a circuit's voltage and current, apparent power (*S*) is of paramount necessity and importance in electrical utilities, electronics, communication, and neural network systems. Based on the existing AC power analysis on the two-terminal passive elements (i.e., *R*, *L*, and *C*), some in-depth research on AC apparent power calculations for second-order memory elements and memristive systems is introduced to help with revealing their complex and unique non-linear phenomena. This paper derives the forms of real power, reactive power, and apparent power for the proposed second-order memory elements (i.e., *MR*, *MC*, and *ML*) and reveals the difference between ideal memory elements and traditional passive ones (i.e., *R*, *C*, and *L*). For all involved memory elements, harmonic values and an extra term occur in the expression of powers to represent their memory characteristics. Especially, the real power is a function of a circuit's dissipative elements (usually resistances *R*), but not exactly the memristor (*MR*). Then, the corresponding curves could be depicted, which demonstrate the differences between *R*/*C*/*L* and *MR*/*MC*/*ML* and verified that harmonic values existed in $S_{MR}/S_{MC}/S_{ML}$, meaning that it would perpetually supply energy when operated with an alternating current.

KEYWORDS

memristor, apparent power, reactive power, memcapacitor, meminductor

1 Introduction

Since the memristor was founded by L. O Chua in 1971 [1] and fabricated by HP Laboratories in 2008 [2,3], the notion of the *MR* was expanded to encompass 'memristive systems' and has demonstrated that the existence of a fingerprint (known as the 'pinched hysteresis loop') is the sufficient condition of a memory system [1,4]. Subsequently, memcapacitor (*MC*, abbreviation of 'memory capacitor') and meminductor (*ML*, abbreviation of 'memory inductor') were postulated in 1978 [2,3]. Up to now, plenty of memristive systems with different memory elements have been implemented. In the field of neural systems and networks, in 2015, both single-associative memory and multi-associative memories based on a memristive Hopfield network have been realized with memristors and memristive systems [5]. In 2019, a novel synaptic unit with double identical memristors and its neural network circuit architecture was built to update the weight matrices [6]. In 2020, C Y Lin and his co-workers demonstrated one resistive random-access memory with a novel memristor to mimic biological synapses, which offered a multi-bit functionality and synaptic plasticity for simulating various strengths in neuronal connections [7]. In 2021, spiking and burst phenomena were successfully simulated based on memristor circuits [8]. In 2022, Juan Pablo Carbajal and his co-workers introduced a training algorithm for a memristor network, which has been implemented in the hardware [9]. Also, Yi and his team reported an activity-difference-based training on co-designed tantalum oxide analog memristor crossbars, which has been termed memristor activity-difference energy minimization and trained one-layer and multilayer neural networks that can classify Braille words with high accuracy [10]. Then, Sun and his team proposed a multimode generalization and differentiation

circuit for the Pavlov associative memory based on memristors [11]. Also, Liao M et al. realized the associative memory neural network and the gradual learning, gradual forgetting, and gradual transferring processes of emotions and designed a memristor-based circuit of the affective associative memory neural network [12]. Based on the memristive Hopfield neural network, neural bursting and synchronization have been imitated by modeling two neural network models [13]. Moreover, the famous Hodgkin–Huxley neuron model with a memristor [14] and firing mechanism for both single memristive neuron and double memristive coupled neurons [15] have been built. From the aforementioned works, it has been widely recognized that memristors have been successfully employed to configure neurons and synapses in a series of neuromorphic circuits.

In the field of emulator and oscillation circuits, the following non-linear behaviors have been founded, such as spiking and bursting oscillation [8,16], coexistent and hidden attractors [17,18], two-parameter bifurcations [8,19], chaotic dynamics [20,21], memristive diode bridge-coupled oscillator [22], neural oscillation [23,24], and the unified floating and grounded mem-element emulator [3,3]. Furthermore, there are some other applications. For example, in memory computing, both charge-based and resistance-based memory devices are used to analyze their physical attributes [25]. In the machine learning and neuromorphic hardware, the memristor has been applied for proving the effectiveness for edge detection [16]. In the privacy protection of medical data [26], image encryption [27], and audio encryption application [28], multi-scroll memristive Hopfield neural networks have played an important role. In 2011, D Biolek et al. presented is a proof that the 'non-crossing-type pinched hysteretic loops' phenomenon cannot occur in ideal memory elements, which are defined axiomatically *via* corresponding constitutive relations or *via* other equivalent characteristics and pointed that the 'crossing-type hysteretic loop' is one of their typical fingerprints [29,30]. In 2020, Guo Z et al introduced a phasor analysis method for memory elements to help with the understanding of complex non-linear phenomena in circuits with a memristor, memcapacitor, meminductor, and second-order memristor [31]. In 2021, the expression of equivalent admittance and impedance connected in parallel and series memristive circuits were derived [32], which are still in their infancy. Also, these existing researches have opened new realms for non-linear circuit investigations.

The second-order memristor, such as the ideal HP memory elements, could be considered as one of the most closely related ways to reflect the constitutive relationship of a physical memristor and are also the keys to developing a new generation of intelligent and neuromorphic devices. There are few pieces of literature that involve power analysis for these memory elements. Although some effort has been applied and published in AC circuit analyses, they are not sufficient in obtaining entry characteristics for an electric circuit in practical engineering. For the sake of the completeness of the non-linear electric circuit theory, power analyses and calculations should be given more and more attention. In this paper, based on constitutive relationships, some in-depth research on AC power calculations for memory elements are introduced to help in revealing their complex and unique non-linear phenomena and memory features. The difference between ideal second-order memory elements (i.e., *MR*, *MC*, and *ML*) and traditional passive ones (i.e., *R*, *C*, and *L*) is presented according to the forms of apparent powers for them. For all involved memory elements, harmonic values and an extra term occur in the expression of apparent power to represent their memory characteristics. Especially, the real power

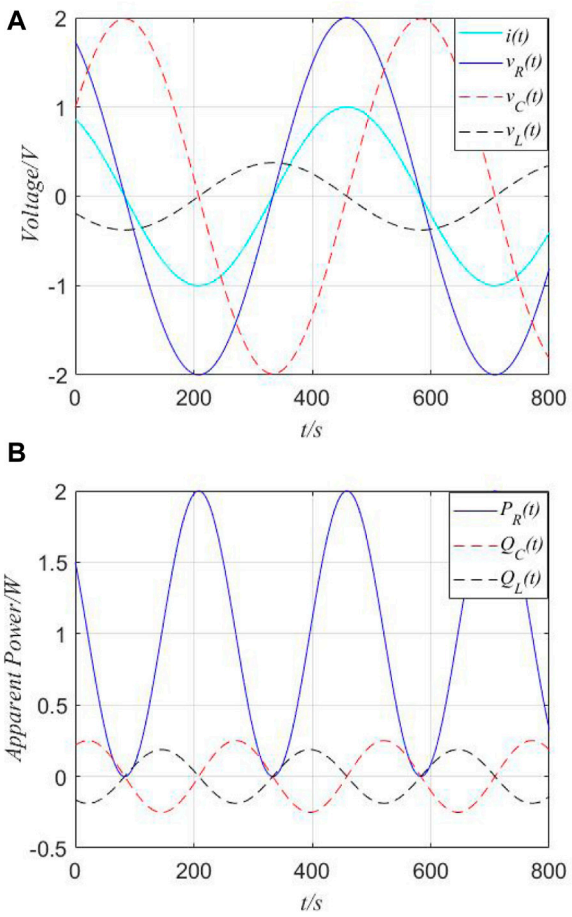


FIGURE 1
Instantaneous voltage and instantaneous power in the time domain. **(A)** Instantaneous voltage in the time domain. **(B)** Instantaneous power in the time domain.

equals the apparent power for a resistor (*R*), which is the positive value, but this result is not available for the memristor (*MR*) in the unit of Ohm (Ω). Moreover, observed from the curves of $P_R/Q_C/Q_L$ and $S_{MR}/S_{MC}/S_{ML}$, harmonic values exist in all expressions of apparent power. These harmonic (and negative) values represent that it would perpetually supply energy when operated with an alternating current.

The remainder of this paper is organized as follows: in Section II, the background on the apparent power for basic 2-terminal passive elements (i.e., *R*, *L*, and *C*) and a brief introduction on ideal memory elements are presented. Then, three apparent power models for an ideal memristor (*MR*), memcapacitor (*MC*), and meminductor (*ML*) are derived in Section III. In Section V, the apparent power for the combination of memory elements is fully studied and analyzed. Finally, the conclusions are summarized in Section VI.

2 Background

2.1 AC power analysis of R, L, and C

Based on the circuit theory, instantaneous power ($P(t)$) could be defined as the product of the instantaneous voltage $v(t)$ across the

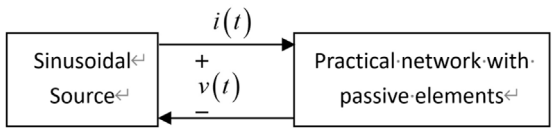


FIGURE 2
Practical network with passive elements excited by the sinusoidal source.

element and instantaneous current $i(t)$ through the load element. The combination of real power and reactive power is called apparent power, without a reference to the phase angle. In a simple circuit with the passive element, the applied current $i(t) = I\cos(\omega t + \theta_i)$ through R, L, or C, instantaneous apparent power ($S_R(t)$, $S_C(t)$, or $S_L(t)$), and relations between the voltage and current are characterized by the following representation:

$$\begin{cases} v_R(t) = RI\cos(\omega t + \theta_i) \\ v_C(t) = \frac{I}{\omega C}\cos\left(\omega t + \theta_i - \frac{\pi}{2}\right) \\ v_L(t) = \omega L\cos\left(\omega t + \theta_i + \frac{\pi}{2}\right) \end{cases} \quad and \quad \begin{cases} S_R(t) = P_R(t) = \frac{RI^2}{2}[1 + \cos(2\omega t + 2\theta_i)] \\ S_C(t) = Q_C(t) = \frac{I^2}{2\omega C}\cos\left(2\omega t + 2\theta_i - \frac{\pi}{2}\right) \\ S_L(t) = Q_L(t) = \frac{\omega LI^2}{2}\cos\left(2\omega t + 2\theta_i + \frac{\pi}{2}\right) \end{cases}, \tag{1}$$

where variables v_R, v_C, and v_L present the voltages; S_R, S_C, and S_L are the apparent powers; P_R, Q_C, and Q_L stand for the real power for a resistor (R) and reactive powers for both the capacitor (C) and inductor (L), respectively.

Then, considering a current source (i) is applied as the input excitation, setting the parameters ($I = 1A$, $\omega = 0.002\ rad/s$, $\theta = \pi/6$, $R = 2\Omega$, $C = 40F$, and $L = 30H$), the following curves of instantaneous voltage ($v_R(t)$, $v_C(t)$, or $v_L(t)$) and instantaneous power ($P_R(t)$, $Q_C(t)$, or $Q_L(t)$) are drawn in Figure 1.

From Figure 1, the instantaneous real power ($P_R(t)$) is always positive, and reactive power ($Q_C(t)$ and $Q_L(t)$) may be positive or negative values.

Next, AC power analysis should be present, which is of paramount importance that involves the transmission of power from one point to another. It could be considered as a basic and useful technique for analyzing circuits with AC signals.

Recalling from physics, the phasor-domain representation of impedances for passive elements (i.e., R, L, and C) can be given as follows:

$$\begin{cases} \dot{Z}_R = R \\ \dot{Z}_C = \frac{1}{j\omega C} = \frac{1}{\omega C}\angle\left(-\frac{\pi}{2}\right) \\ \dot{Z}_L = j\omega L = \omega L\angle\frac{\pi}{2} \end{cases}. \tag{2}$$

We consider a practical circuit network, which is the arbitrary combination of passive elements under sinusoidal excitations, as shown in Figure 2.

Both the voltage and current at the terminals of the network can be described as follows:

$$i(t) = I\cos(\omega t + \theta_i) \quad and \quad v(t) = U\cos(\omega t + \theta_v), \tag{3}$$

where both variables I and U present amplitudes (or peak values); θ_i and θ_v stand for phase angles of the voltage and current, respectively.

Thus, instantaneous power for the network is computed as follows:

$$P(t) = \frac{1}{2}UI[\cos(2\omega t + \theta_v + \theta_i) + \cos(\theta_v - \theta_i)]. \tag{4}$$

From Eq. 4, there are two terms in the form of instantaneous power. The first part is a sinusoidal function whose frequency is 2ω, which is twice the angular frequency of the voltage or current, plus the sum of the phase of the voltage and current. The second one is time independent, which depends upon the phase difference between the voltage and current.

Practically, instantaneous power is difficult to measure. Also, the value measured by the wattmeter is the average power, which shows the average of instantaneous power over a period of time and is given by

$$P_{avg} = \frac{UI}{2T}\int_0^T \cos(2\omega t + \theta_v + \theta_i)dt + \frac{1}{2}UI\cos(\theta_v - \theta_i), \tag{5}$$

where P_{avg} means the average of power. It has two integrals. The first integral is a sinusoid. The average of this sinusoid over a period of time is zero. The second integral term is constant. Thus, average power can be denoted as $P_{avg} = \frac{1}{2}UI\cos(\theta_v - \theta_i)$.

Subsequently, based on the concept of the effect value or the root of the mean of the square of the AC signal, the effect value of power (P_{rms}) can be written as follows:

$$P_{rms} = \frac{U}{\sqrt{2}}\frac{I}{\sqrt{2}}\cos(\theta_v - \theta_i) = U_{rms}I_{rms}\cos(\theta_v - \theta_i). \tag{6}$$

Moreover, in order to clearly show the related concepts on load impedance ($Z = R + j(\omega L - \frac{1}{\omega C})$) in an AC circuit, apparent power (S) and reactive power (Q) can be presented as follows:

$$\begin{cases} S = I_{rms}^2 \cdot Z = P + jQ \\ P = \mathrm{Re}(S) = U_{rms}I_{rms}\cos(\theta_v - \theta_i) \\ Q = \mathrm{Im}(S) = U_{rms}I_{rms}\sin(\theta_v - \theta_i) \end{cases}, \tag{7}$$

where real power ($P = P_{rms} = P_{avg}$) is delivered to a load in watts, which is the only useful and actual power dissipated by the load. Reactive power (Q) is related to the energy exchange between the source and the reactive part of the load.

2.2 Mathematical models of the ideal second-order memristor, memcapacitor, and meminductor

According to the concepts of memristors in [1], there are three mathematical representations of the time-invariant ones, which have been named as extended memristor, generic memristor, and ideal memristor; each one has two forms depending on whether the input signal is a current source (current-controlled memristor) or a voltage source (voltage-controlled memristor). In this section, we focus on one of the specific cases based on the constitutive relationship, i.e., the ideal second-order memristor (MR), second-order memcapacitor (MC), and second-order meminductor (ML).

Considering a charge-controlled memristor for an ideal second-order one, its constitutive relationship can be described analytically by a proposed cubic polynomial:

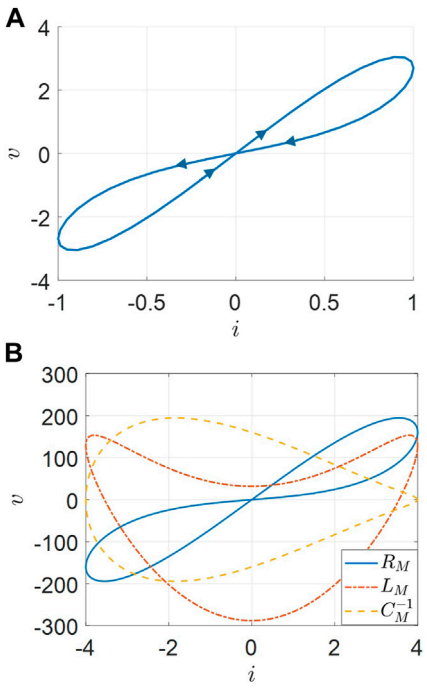


FIGURE 3
Pinched hysteresis loop and Lissajous figure of *v*–*i*. **(A)** Pinched hysteresis loop of an ideal memristor. **(B)** *v*–*i* for the inverse memcapacitance and meminductance.

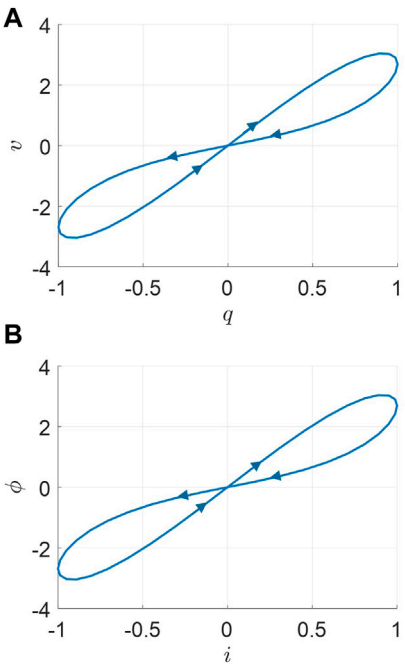


FIGURE 4
Pinched hysteresis loops. **(A)** Double-valued Lissajous figure of $q(t)$ and $v(t)$ for an ideal second-order memcapacitor. **(B)** Double-valued Lissajous figure of $i(t)$ and $\phi(t)$ for an ideal second-order meminductor.

$$\varphi = \frac{1}{3}a_{11}q^3 + \frac{1}{2}b_{11}q^2 + Rq \tag{8}$$

where φ and q are the accumulated flux and charge, respectively; a_{11} and b_{11} are the two parameters; R represents the initial memristance in Ohm (Ω).

The pinched hysteresis loop occurs at the origin $(v, i) = (0, 0)$ and is depicted in Figure 3A. The Lissajous figures of v–i for C_m^{-1} and L_m are the approximative conical ellipse loops and semi-ellipse loops in Figure 3B, respectively.

Also, the proposed ideal second-order memristance $R(q)$ can be calculated as follows:

$$R_m(q) = a_{11}q^2 + b_{11}q + R. \tag{9}$$

Then, constitutive relations and notions of the ideal second-order memcapacitor and meminductor can be expanded as follows:

$$\begin{cases} C_m(\varphi) = a_{12}\varphi^2 + b_{12}\varphi + C \\ L_m(q) = a_{13}q^2 + b_{13}q + L \end{cases}, \tag{10}$$

where C_m and L_m stand for the memcapacitance and meminductance, respectively; φ and q are the time-domain integrals of u and i, respectively; a_{12}, b_{12}, a_{13}, and b_{13} are the parameters; both C and L represent the initial memcapacitance in farad (F) and meminductance in henry (H).

Both pinched hysteresis loops occur at the origin $(v, i) = (0, 0)$ and are depicted in Figure 4.

From Figure 3 and Figure 4, 'crossing-type hysteretic loops' are exhibited as one of their typical fingerprints for ideal memory elements [4].

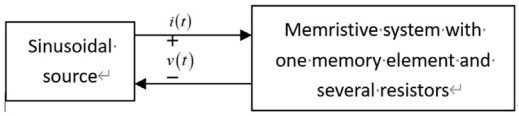


FIGURE 5
Memristive system with a memory element and several resistors excited by a sinusoidal source.

3 Power analysis for memory elements

We consider a non-linear circuit which is a combination of a memory element and several resistors under sinusoidal excitation. It is tested with bipolar periodic input sinusoidal signals, which result in a periodic sinusoidal response with a different frequency. Both the resistor and memristor have the unit of Ohm, and the "in-phase" relationship could be found for the purely resistive circuit, but different conclusions could occur for the non-linear circuit with a memory element. As mentioned in the previous section, this section begins by defining and deriving the apparent power for this special non-linear circuit with only one memory element, as shown in Figure 5.

3.1 Apparent power of an ideal *MR*

For a charge-controlled ideal second-order memristor (see Eq. 9), we assume that the applied current source is $i(t) = I\cos(\omega t + \theta_i)$,

and the relation between the voltage and current can be given as follows:

$$\begin{cases} i(t) = I\cos(\omega t + \theta_i), \quad t \neq 0 \\ v = \left(RI + \frac{a_{11}I^3}{4\omega^2}\right)\cos(\omega t + \theta_i) - \frac{a_{11}I^3}{4\omega^2}\cos(3\omega t + 3\theta_i) + \frac{b_{11}I^2}{2\omega}\sin(2\omega t + 2\theta_i) \end{cases}, \tag{11}$$

where the variables i and v_{MR} with the period t could be presented in the form of the Fourier series; I, $(RI + \frac{a_{11}I^3}{4\omega^2})$, $(-\frac{a_{11}I^3}{4\omega^2})$, and $(\frac{b_{11}I^2}{2\omega})$ are real coefficients. The voltage (v_{MR}) has three parts, and all of them are sinusoidal functions; the frequency in the first part is also ω, which is the same angular frequency and phase between the voltage and current. The frequency in the second one is 3ω, which is triple the angular frequency and phase of the current. The frequency in the third one is 2ω, which is double the angular frequency and phase of the current.

Then, instantaneous power $P_{MR}(t)$ for a memristor in Figure 5 could be also defined as the product of instantaneous voltage $v_{MR}(t)$ across this element and the instantaneous current $i(t)$ through it, given as follows:

$$\begin{aligned} P_{MR}(t) = v_{MR}(t)i(t) &= \left(\frac{RI^2}{2} + \frac{a_{11}I^4}{8\omega^2}\right) \\ &+ \frac{RI^2}{2}\cos(2\omega t + 2\theta_i) - \frac{a_{11}I^4}{8\omega^2}\cos(4\omega t + 4\theta_i) \\ &+ \frac{b_{11}I^3}{4\omega}[\sin(3\omega t + 3\theta_i) + \sin(\omega t + \theta_i)], \end{aligned} \tag{12}$$

where the variable P_{MR} is also presented in the form of the Fourier series; $(\frac{RI^2}{2} + \frac{a_{11}I^4}{8\omega^2})$, $(\frac{RI^2}{2})$, $(-\frac{a_{11}I^4}{8\omega^2})$, and $(\frac{b_{11}I^3}{4\omega})$ are real coefficients. In Eq. (12), there are five terms in the form of memristor instantaneous power. The first part is constant or time independent, which depends on the angular frequency of the current. The second part is a sinusoidal function whose frequency is 2ω, which is twice the angular frequency and phase of the current. The third part is a sinusoidal function whose quadruple frequency and phase are 4ω and $4\theta_i$, respectively. The last part is also a sinusoidal function.

Moreover, the average value of memristor instantaneous power over on period can be given as follows:

$$\begin{aligned} P_{avgMR}(t) = &\left(\frac{RI^2}{2} + \frac{a_{11}I^4}{8\omega^2}\right) + \frac{RI^2}{2T}\int_0^T \cos(2\omega t + 2\theta_i)dt - \frac{a_{11}I^4}{8t\omega^2}\int_0^T \cos(4\omega t + 4\theta_i)dt \\ &+ \frac{b_{11}I^3}{4T\omega}\int_0^T [\sin(3\omega t + 3\theta_i) + \sin(\omega t + \theta_i)]dt \end{aligned}, \tag{13}$$

where P_{avgMR} has two terms. The first one is a non-linear function with frequency instead of the constant for purely resistive circuits. The second integer term is a sinusoid, which equals to zero over a period of time. Therefore, memristor average power could be denoted as $P_{avgMR} = (\frac{RI^2}{2} + \frac{a_{11}I^4}{8\omega^2})$, which is quite different from the value of one linear resistor.

Furthermore, as mentioned in the concept of the effect value of power (P_{rmsMR}), it can be derived as follows:

$$P_{rmsMR} = R\left(\frac{I}{\sqrt{2}}\right)^2 + \frac{a_{11}}{2\omega^2}\left(\frac{I}{\sqrt{2}}\right)^4 = RI_{rms}^2 + \frac{a_{11}}{2\omega^2}I_{rms}^4, \tag{14}$$

where P_{rmsMR} has two parts. The first part is a constant which is similar to the effect power of the resistor. The second one is a non-linear function, which could change with the frequency affecting the memristive circuit effect power.

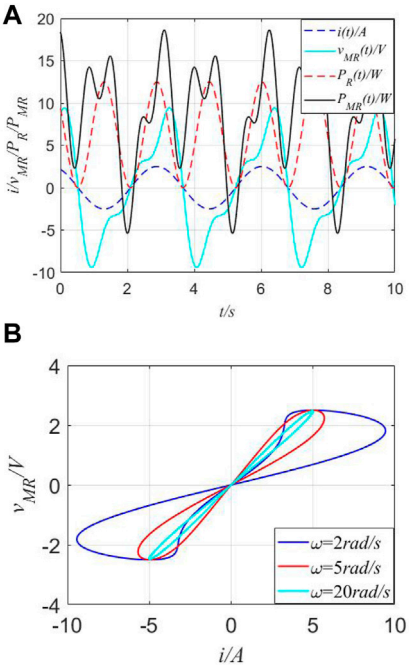


FIGURE 6
Different curves of instantaneous power in the time domain and fingerprint of *MR*. **(A)** Curves of the current ($i(t)$), voltage ($v_{MR}(t)$), and instantaneous powers ($P_R(t)$ and $P_{MR}(t)$). **(B)** Pinched hysteresis loops of the ideal *MR*.

According to the concepts in the circuit theory, for an AC purely resistive circuit, the current and voltage are in-phase and power at any instant can be found by multiplying the voltage by the current at that instant, and because of this "in-phase" relationship, $P(t)$ and $P_{rms}(t)$ values can be used to find the equivalent DC power or heating effect.

$$\begin{cases} P_R(t) = RI^2\cos^2(\omega t + \theta_i) = \frac{RI^2}{2}[1 + \cos(2\omega t + 2\theta_i)] \\ P_{MR}(t) = \frac{RI^2}{2}[1 + \cos(2\omega t + 2\theta_i)] + \frac{a_{11}I^4}{8\omega^2}[1 - \cos(4\omega t + 4\theta_i)] \end{cases} \tag{15}$$

and

$$\begin{cases} P_{rmsR} = \frac{UI}{2}\cos(\theta_v - \theta_i) = RI_{rms}^2 \\ P_{rmsMR} = R\left(\frac{I}{\sqrt{2}}\right)^2 + \frac{a_{11}}{2\omega^2}\left(\frac{I}{\sqrt{2}}\right)^4 = RI_{rms}^2 + \frac{a_{11}}{2\omega^2}I_{rms}^4 \end{cases}. \tag{16}$$

While comparing the instantaneous power between (P_R) and (P_{MR}) (see Eq. 15) and effect power between (P_{rmsR}) and (P_{rmsMR}) (see Eq. 16), there is an extra term in each equation; they are unpublished and important special variables that change with the frequency. Both of them can be considered as key points in exhibiting memory characteristics for a memristor.

Hereby, when a current source (i) is applied through this memory element, the parameters ($I = 3A$, $\omega = 1.5\ rad/s$, $\theta = \pi/6$, $R = 2$, $b_{11} = 2$, and $a_{11} = 2$), curves of the current $(i(t))$, voltage $(v_{MR}(t))$, and instantaneous powers ($P_R(t)$ and $P_{MR}(t)$) are shown in Figure 6.

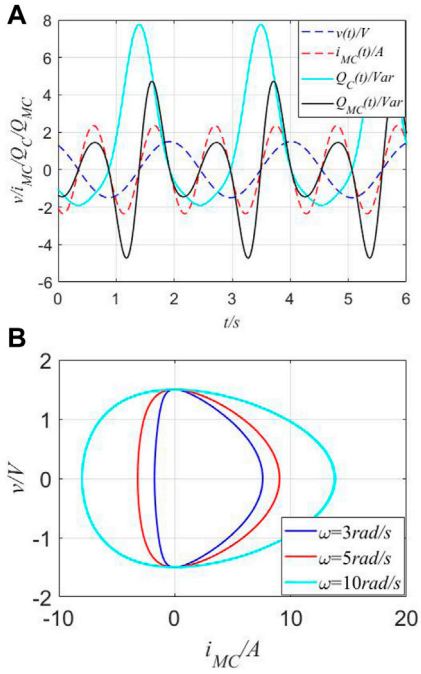


FIGURE 7
Different curves of instantaneous power in the time domain and fingerprint of *MC*. **(A)** Curves of $v(t)$, $i_{MC}(t)$, and $Q_C(t)$ and $Q_{MC}(t)$ in the time domain. **(B)** Pinched hysteresis loops of the ideal *MC*.

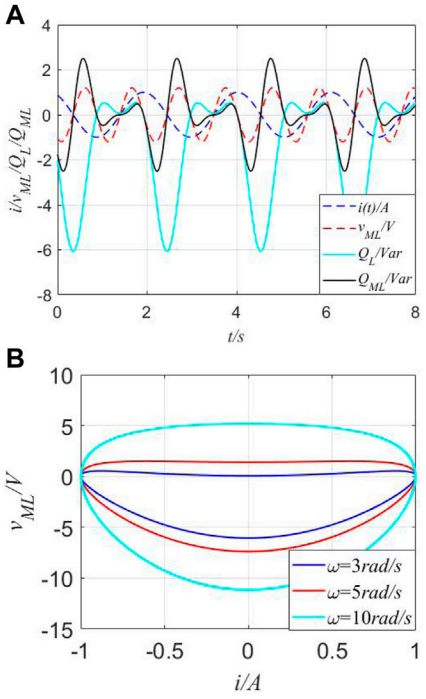


FIGURE 8
Instantaneous values in the time domain and fingerprint of *ML*. **(A)** Instantaneous values of the current ($i(t)$), voltage ($v_{ML}(t)$), and instantaneous reactive powers ($Q_L(t)$ and $Q_{ML}(t)$) in the time domain. **(B)** Pinched hysteresis loops of the ideal *ML*.

From Figure 6, multiple frequencies between the input signal ($i(t)$)/response signal ($v_{MR}(t)$) and instantaneous power ($P_R(t)$ and $P_{MR}(t)$) are observed. Also, the existence of fingerprints for the proposed *MR* could be verified. For a traditional resistor (R), when the current source $i(t)$ is applied through $R = 2\Omega$, it is absorbed power in watts (W) and can be illustrated through red-dotted lines, which present the twice frequency relationship between $i(t)$ and $P_R(t)$. Complex curves of $v_{MR}(t)$ and $P_{MR}(t)$ demonstrate unique memory characteristics by negative values, i.e., the negative value would mean that it would perpetually supply energy when operated with an alternating current. However, according to concepts of real power for a resistor, it should always be a positive value. Thus, real power is not suitable for defining the features of an MR or memristive system, with apparent power being applied ($S_{MR}(t)$).

Furthermore, in general electrical engineering, the power factor (abbreviated as *pf*) of an AC power system is defined as the ratio of the real power absorbed by the load to the apparent power flowing in the circuit. Real power is the average of the instantaneous product of voltage and current and represents the capacity of electricity for performing the work. Therefore, there are the following relations: 1) $Q = 0$ for resistive loads (unify *pf*); 2) $Q < 0$ for capacitive loads (leading *pf*); 3) $Q > 0$ for inductive loads (lagging *pf*). However, from Figure 6A, both positive and negative values exist in instantaneous power ($P_{MR}(t)$), instead of only the positive values in real instantaneous power ($P_R(t)$ in Figure 1B).Here, there are two meanings: One means that the perpetual supply energy could occur to keep its unique memory characteristics and the other implies that the effect of the power factor disappeared in such types of circuits and only the definition of apparent power is still working.

3.2 Apparent power for an ideal *MC*

For a φ-controlled (φ denotes the time-domain integral of the voltage) ideal second-order memcapacitor (see Eq. 10) and applied voltage source $v(t) = U\cos(\omega t + \theta_v)$, the following relation between voltage and current can be computed:

$$\begin{cases} v(t) = U\cos(\omega t + \theta_v), \quad t \neq 0 \\ i_{MC}(t) = -\left(UC\omega + \dfrac{3\alpha_{12}U^3}{4\omega}\right)\sin(\omega t + \theta_v) + \dfrac{\alpha_{12}U^3}{4\omega}\sin(3\omega t + 3\theta_v) \\ \qquad\qquad + \dfrac{b_{12}U^2}{2}\cos(2\omega t + 2\theta_v) - \dfrac{b_{12}U^2}{2}, \end{cases} \tag{17}$$

where the variables v and i_{MC} with the period t can be presented in the form of the Fourier series; $(\pm\frac{b_{12}U^2}{2})$, $(-UC\omega - \frac{3\alpha_{12}U^3}{4\omega})$, and $(\frac{\alpha_{12}U^3}{4\omega})$ are real coefficients. The current (i_{MC}) has four parts, and three of them are sinusoidal functions, which are the multiple angular frequencies and phases between the voltage and current. The last part is a constant.

In an AC circuit, the product of voltage and current is expressed as volt-ampere (VA) and is known as apparent power, symbol "S". Also, the "in-phase" relationship between the current and voltage exists for an AC purely resistive circuit. However, if the circuit contains reactive components, the voltage and current waveforms will be "out-of-phase" by some amount determined by the circuit's phase angle. If the phase angle between the voltage and current is at its maximum of $\pi/2$, the volt-ampere product will have equal positive and negative values. In other words, there is also another power component that is present

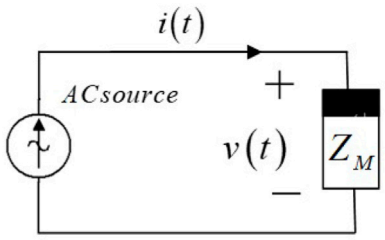


FIGURE 9
Special circuit with a current source $i(t)$ through impedance (Z_M).

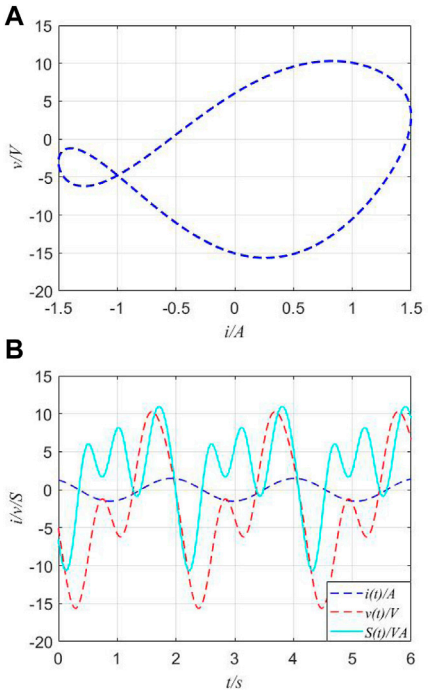


FIGURE 10
v–i curves and apparent power of series-connected circuits. **(A)** *v–i* curves of the series-connected combination of *MR*, *MC*, and *ML*. **(B)** Voltage and apparent power in the time domain.

whenever there is a phase angle. This component is called reactive power (sometimes referred to as imaginary power) and is expressed in a unit called "volt-amperes reactive", (Var), and symbol "*Q*." When the reactive circuit returns as much power to the supply as it consumes, it results in the average power consumed by the circuit becoming zero. Then, the expression of active power $P(t)$ and $P_{rms} = U_{rms}I_{rms}$ is no longer suitable. These reactive components include the capacitor, inductor, and memory elements (*i.e.*, memcapacitor and meminductor). Also, both *MC* and *ML* are not considered as new fundamental circuit elements (Liu, 2020b).

Then, instantaneous reactive power $Q_{MC}(t)$ could be defined as the product of the instantaneous voltage $v(t)$ across this element and the instantaneous current $i_{MC}(t)$ through it, given as follows:

$$\begin{aligned} Q_{MC}(t) = &\left(-\frac{1}{2}U^2C\omega - \frac{a_{12}U^4}{4\omega}\right)\sin(2\omega t + 2\theta_v) \\ &+\frac{\alpha_{12}U^4}{8\omega}\sin(4\omega t + 4\theta_v) + \frac{b_{12}U^3}{4}\cos(3\omega t + 3\theta_v) \\ &-\frac{b_{12}U^3}{4}\cos(\omega t + \theta_v), \end{aligned} \tag{18}$$

where the variable Q_{MC} is also presented in the form of the Fourier series; $(-\frac{1}{2}U^2C\omega - \frac{a_{12}U^4}{4\omega})$, $(\frac{a_{12}U^4}{8\omega})$, $(\frac{b_{12}U^3}{4})$, and $(-\frac{b_{12}U^3}{4})$ are real coefficients. There are four parts in Q_{MC}, whose values all depend on changes with the angular frequency.

While comparing instantaneous reactive powers between $(Q_C(t))$ and $(Q_{MC}(t))$, there is an extra term in Eq. (19), which is unpublished and important because special variables change with the frequency. They could be considered as key points for exhibiting memory characteristics for the memcapacitor.

$$\begin{cases} Q_C(t) = -\dfrac{U^2C\omega}{2}\sin(2\omega t + 2\theta_v) \\ Q_{MC}(t) = \left(-\dfrac{U^2C\omega}{2} - \dfrac{a_{12}U^4}{4\omega}\right)\sin(2\omega t + 2\theta_v) \\ \quad +\left[\dfrac{b_{12}U^3}{4}\cos(3\omega t + 3\theta_v) - \dfrac{b_{12}U^3}{4}\cos(\omega t + \theta_v) + \dfrac{a_{12}U^4}{8\omega}\sin(4\omega t + 4\theta_v)\right]. \end{cases} \tag{19}$$

When the phase angle between the voltage and current is at its maximum of $\pi/2$, the relationship can be given as follows:

$$\begin{cases} Q_C = -\dfrac{CU^2}{2}\omega \\ Q_{MC} = -\dfrac{U^2C\omega}{2} + \dfrac{a_{12}U^4}{4\omega} \end{cases}. \tag{20}$$

Next, for an φ-controlled memcapacitor, when a voltage source (v) is applied through *MC*, the parameters ($U = 1.5A$, $\omega = 3\ rad/s$, $\theta = \pi/6$, $C = 0.7$, $a_{12} = 1.3$, and $b_{12} = -1.3$), the curves of voltage $(v(t))$, current $(i_{MC}(t))$, and instantaneous reactive powers between $(Q_C(t))$ and $(Q_{MC}(t))$ are depicted in Figure 7.

As observed from Figure 7, multiple frequencies between the input signal $(v(t))$/response signal $(i_{MC}(t))$ and instantaneous powers $(Q_C(t)$ and $Q_{MC})$ are observed. Then, the existence of fingerprints for the constructed *MC* could be verified. For one traditional capacitor (C), when the voltage source $v(t)$ is applied cross $C = 0.7F$, its reactive power can be represented through a cyan-solid line, which presents the twice frequency relationship between $v(t)$ and $Q_C(t)$. Complex curves of $i_{MC}(t)$ and $Q_{MC}(t)$ demonstrate unique memory characteristics through negative values with the unit of volt-ampere reactive (Var).

3.3 Apparent power for an ideal *ML*

Similar to the description on *MC* and motivated by the aforementioned analysis on the ideal *ML*, for a q-controlled ideal meminductor (see Eq. 10) and applied current source $i(t) = I\cos(\omega t + \theta_i)$, the relation between the voltage and current can be given as follows:

$$\begin{cases} i(t) = I\cos(\omega t + \theta_i), \quad t \neq 0 \\ v_{ML}(t) = -\left(IL\omega + \dfrac{3a_{13}I^3\omega}{4\omega^2}\right)\sin(\omega t + \theta_i) + \dfrac{a_{13}I^3}{4\omega}\sin(3\omega t + 3\theta_i) + \dfrac{b_{13}I^2}{2}\cos(2\omega t + 2\theta_i) - \dfrac{b_{13}I^2}{2}, \end{cases} \tag{21}$$

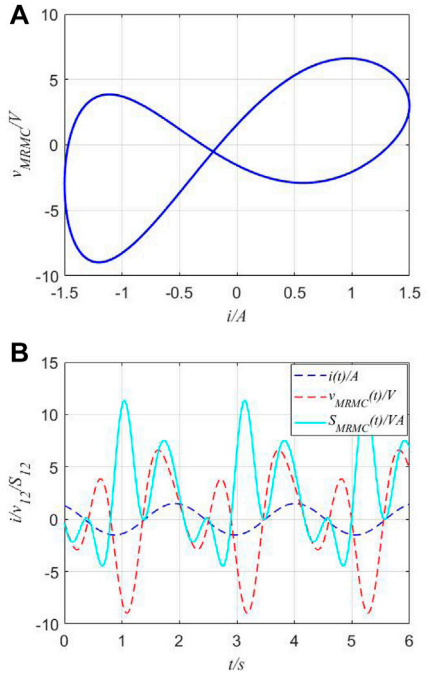


FIGURE 11
v–i curves and apparent power of series-connected circuits. **(A)** *v–i* curves of the series-connected combination of *MR* and *MC*. **(B)** Voltage and apparent power in the time domain.

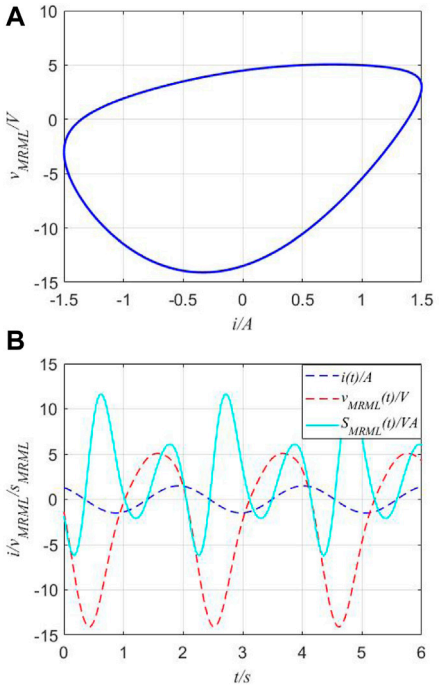


FIGURE 12
v–i curves and apparent power of series-connected circuits. **(A)** *v–i* curves of the series-connected combination of *MR* and *ML*. **(B)** Voltage and apparent power.

where the I and V_{ML} with the period t can be presented in the form of the Fourier series. $(-IL\omega - \frac{3a_{13}I^3\omega}{4\omega^2})$, $(\frac{a_{13}I^3}{4\omega})$, and $(\pm\frac{b_{13}I^2}{2})$ are real coefficients. Similar to MC, the voltage (v_{ML}) has four parts; they are one constant and three sinusoidal functions, which are multiple angular frequencies and phases between the voltage and current.

Then, instantaneous reactive power $Q_{ML}(t)$ can be derived as follows:

$$\begin{aligned} Q_{ML}(t) = &\left(-\frac{1}{2}I^2L\omega - \frac{a_{13}I^4\omega}{4\omega^2}\right)\sin(2\omega t + 2\theta_i) \\ &+\frac{a_{13}I^4}{8\omega}\sin(4\omega t + 4\theta_i) + \frac{b_{13}I^3}{4}\cos(3\omega t + 3\theta_i), \\ &-\frac{b_{13}I^3}{4}\cos(\omega t + \theta_i) \end{aligned} \tag{22}$$

where $(-\frac{LI^2}{2}\omega - \frac{a_{12}U^4}{4\omega})$, $(\frac{a_{13}I^4}{8\omega})$, $(\frac{b_{13}I^3}{4})$, and $(-\frac{b_{13}I^3}{4})$ are real coefficients. In Eq. 22, there are also four parts, whose values could be determined by the change in angular frequency.

The instantaneous reactive power between $(Q_L(t))$ and $(Q_{ML}(t))$ (see Eq. 23) is computed as follows:

$$\begin{cases} Q_L(t) = -\dfrac{I^2L\omega}{2}\sin(2\omega t + 2\theta_i) \\ Q_{ML}(t) = \left(-\dfrac{I^2L\omega}{2} - \dfrac{a_{13}I^4\omega}{4\omega^2}\right)\sin(2\omega t + 2\theta_i) \\ \quad +\dfrac{a_{13}I^4}{8\omega}\sin(4\omega t + 4\theta_i) + \dfrac{b_{13}I^3}{4}\cos(3\omega t + 3\theta_i) - \dfrac{b_{13}I^3}{4}\cos(\omega t + \theta_i) \end{cases}, \tag{23}$$

where two extra negative terms exist in $Q_L(t)$ and $Q_{ML}(t)$, which are important special variables and change with the frequency. They can be considered as key points in exhibiting memory characteristics for the meminductor.

When the phase angle between the voltage and current is at its maximum of $\pi/2$, the relationship can be given as follows:

$$\begin{cases} Q_L = -\dfrac{I^2L\omega}{2} \\ Q_{ML} = -\dfrac{I^2L\omega}{2} - \dfrac{a_{13}I^4\omega}{4\omega^2} \end{cases}. \tag{24}$$

Setting the parameters ($I = 1A$, $\omega = 3\ rad/s$, $\theta = \pi/6$, $L = 0.8$, $b_{13} = 3$, and $a_{13} = 2$), curves of voltage $(v_m(t))$ and instantaneous reactive power $(Q_{ML}(t))$ are shown in Figure 8.

Observed from Figure 8, similar to *MR* and *MC*, multiple frequencies are verified. Then, the fingerprint of the MC does exist. For a traditional inductor (L), when the current source $i(t)$ is applied through $L = 0.8H$, its reactive power can be represented through a cyan-solid line, which presents the twice frequency relationship between $i(t)$ and $v_{ML}(t)$. Complex curves of $v_{ML}(t)$ and $Q_{ML}(t)$ demonstrate unique memory characteristics with the negative volt-ampere product.

4 Apparent power for the combination of memory elements

When a specific circuit is built by an AC source and a combination of the proposed memory elements, its apparent power could be expressed by a function of the circuit's total memory impedance (Z_M). In the previous section, the true power or reactive power is

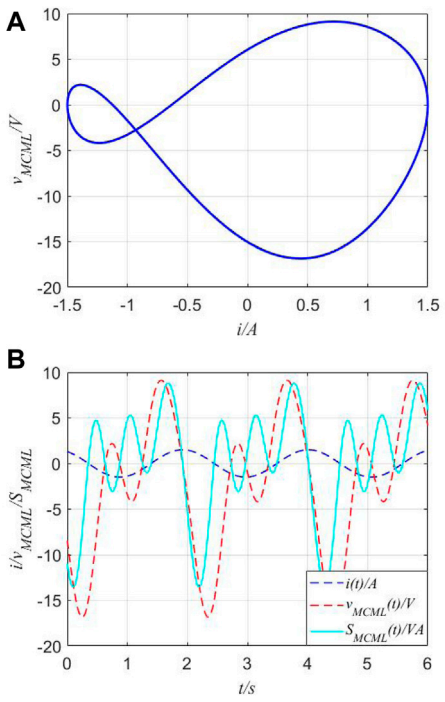


FIGURE 13
v–i curves and apparent power of the series-connected circuit. **(A)** *v–i* curves of the series-connected combination of MR and ML. **(B)** Voltage and apparent power.

discussed for *MR*, *MC*, and *ML*, respectively. According to the definition of impedance ($Z = v/i$) and Ohm's law, the impedance of the memory elements can be derived as follows:

$$\begin{cases} Z_{MR} = \left(\frac{a_{11}I^2}{2\omega^2} + R\right) - \frac{a_{11}I^2}{2\omega^2}\cos(2\omega t + 2\theta_i) + \frac{b_{11}I}{\omega}\sin(\omega t + \theta_i) \\ Z_{ML} = -\left[\left(L\omega + \frac{a_{13}I^2}{\omega^2}\right)\tan(\omega t + \theta_i) + \frac{b_{13}I}{\cos(\omega t + \theta_i)}\right] + \frac{a_{13}I^2}{2\omega}\sin(2\omega t + 2\theta_i) + b_{13}I\cos(\omega t + \theta_i) \\ Z_{MC}^{-1} = -\left[\left(C\omega + \frac{a_{12}U^2}{\omega}\right)\tan(\omega t + \theta_v) + \frac{b_{12}U}{\cos(\omega t + \theta_v)}\right] + \frac{a_{12}U^2}{2\omega}\sin(2\omega t + 2\theta_v) + b_{12}U\cos(\omega t + \theta_v) \end{cases}, \quad (25)$$

where Z_{MC}^{-1} is the inverse reactance of a memcapacitor. For the proposed memory elements, the relationship of the lead and lag between the current and voltage is not available and is difficult to be described by any existing rule.

Hereby, a series-connected combination of all memory elements denoted as (Z_M) is designed in Figure 9.

When a series-connected circuit with an unknown combination of memory elements is configured, the following four special cases occur, and they are analyzed and discussed in this section. A typical situation for the first case is the combination of *MR*, *MC*, and *ML*. Then, parameter values are fixed as $I = 1.5A$, $\omega = 3\ rad/s$, $\theta = \pi/6$, $R = 2$, $C = 0.07$, $L = 0.5$, $a_{11} = 2$, $b_{11} = 2$, $a_{12} = 1.3$, $b_{12} = -1.3$, $a_{13} = 2$, and $b_{13} = 2$; the *v–i* curves, $v(t)$ curves, and apparent powers in the time domain are shown in Figure 10.

In Figure 10, the *v–i* curve of $Z_M = Z_{MR} + Z_{MC}^{-1} + Z_{ML}$ does not cross its origin, and it is a loop structure bent clearly. Also, the loop is asymmetrical with the origin-/x-/*y*-axis. Its apparent power presents diversity but it could not be simply attributed to the individual expression by any certain memory element or traditional component.

A typical situation for the second case is the combination of *MR* and *MC*; the *v–i* curves, $v(t)$ curves, and apparent power are shown in Figure 11. For convenience, the parameter values of memory elements used in this case are exactly the same as the ones in the last case.

In Figure 11, the *v–i* curve of $Z_M = Z_{MR} + Z_{MC}^{-1}$, similar to Figure 10, also does not cross the origin and asymmetry.

The combination of *MR* and *ML* is a typical situation for the third case as shown in Figure 12. Its *v–i* of $Z_M = Z_{MR} + Z_{ML}$ curves is a triangle frisbee-like loop surrounding the origin.

The combination of *MC* and *ML* is a typical situation for the fourth case; the *v–i* curves and apparent power are shown in Figure 13.

In practice, some different or same types of multiple memory elements can be connected together, and the combination of the memory elements is very complex. Hereby, four typical situations are introduced and discussed to illustrate the physical characteristics of *v–i* and apparent powers, which could conveniently help in analyzing non-linear behaviors and finding phenomena of the combination of the proposed second-order memristor, memcapacitor, and meminductor. The theoretical analysis of the combination problem of memory elements can conveniently reveal whether some traditional definitions and rules are still available for these memory elements. This method can also be suitable for the physical connection problem when the connected memory elements operate nearly in their ideal ranges.

5 Conclusion

Since memory elements have been considered as the key for developing the new generation of intelligent devices postulated by some researchers, some neuromorphic systems and basic memristive circuits should become one of the hotspots, such as for physical expression and power analysis. According to the concepts on constitutive relationships, ideal second-order memory elements are proposed; their expressions of current/voltage are derived according to the input excitation. Then, the difference between ideal second-order memory elements (i.e., *MR*, *MC*, and *ML*) and traditional passive ones (i.e., *R*, *C*, and *L*) is presented according to forms of true power, reactive power, and apparent power for them. Moreover, the corresponding curves in the time domain are depicted. Observed from the curves of $P_R/Q_C/Q_L$ and $S_{MR}/S_{MC}/S_{ML}$, harmonic (and negative) values exist in all expressions of apparent power. These harmonic values represent that they would perpetually supply energy when operated with an alternating current. Finally, a series-connected circuit with an unknown combination of memory elements is configured; the *v–i* curves, voltages, and apparent power of four special cases are shown in detail. For memristive circuits, analyses show that the traditional relationship of the lead and lag between the current and voltage is not available and is difficult to be described by any existing rule. Their apparent power presents diversity, but it could not be simply attributed to an individual expression by any certain memory element or traditional component.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Materials; further inquiries can be directed to the corresponding author.

Author contributions

YL, FL, and AW contributed to the conception and design of the study. WL and HL organized the database. YL wrote the first draft of the manuscript. All authors contributed to manuscript revision and read and approved the submitted version.

Funding

This work was supported in part by three Science and Technology Projects of Jilin Province under grant nos 20210201105GX, 202522YY010173621, and 20220204090YY.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

References

1. Chua LO. If it's pinched it's a memristor. *Semiconductor Sci Tech* (2014) 49(10): 104001. doi:10.1088/0268-1242/29/10/104001

2. Liu Y, Iu HH. Novel floating and grounded memory interface circuits for constructing mem-elements and their applications. *IEEE Access* (2020) 8(1):114761–72. doi:10.1109/ACCESS.2020.3004160

3. Liu Y, Iu HH, Guo Z, Si G. The simple charge-controlled grounded/floating mem-element emulator. *IEEE Trans Circ Syst.–II: Express Briefs* (2020) 68(6):2177–81. doi:10.1109/TCSII.2020.3041862

4. Chua LO. Resistance switching memories are memristors. *Appl Phys A* (2011) 102: 765–83. doi:10.1007/s00339-011-6264-9

5. Hu SG, Liu Y, Liu Z, Chen TP, Wang JJ, Yu Q, et al. Associative memory realized by a reconfigurable memristive Hopfield neural network. *Nat Commun* (2015) 6(1):7522–8. doi:10.1038/ncomms8522

6. Wang C, Xiong L, Sun J, Yao W. Memristor-based neural networks with weight simultaneous perturbation training. *Nonlinear Dyn* (2019) 95(4):2893–906. doi:10.1007/s11071-018-4730-z

7. Lin CY, Chen J, Chen PH, Chang TC, Wu Y, Eshraghian JK, et al. Adaptive synaptic memory via lithium-ion modulation in RRAM devices. *Small* (2020) 16(42):2003964. doi:10.1002/smll.202003964

8. Innocenti G, Di MM, Tesi A, Forti M. Memristor circuits for simulating neuron spiking and burst phenomena. *Front Neurosci* (2021) 2021:681035. doi:10.3389/fnins.2021.681035

9. Carbajal JP, Martin DA, Chialvo DR. Learning by mistakes in memristor networks. *Phys Rev E* (2022) 105(5):054306. doi:10.1103/PhysRevE.105.054306

10. Yi S, Kendall JD, Williams RS, Kumar S. Activity-difference training of deep neural networks using memristor crossbars. *Nat Elect* (2022) 2022:1–7. doi:10.1038/s41928-022-00869-w

11. Sun J, Wang Y, Liu P, Wen S, Wang Y. Memristor-based neural network circuit with multimode generalization and differentiation on pavlov associative memory. *IEEE Trans Cybernetics* (2022) 2022:1–12. doi:10.1109/TCYB.2022.3200751

12. Liao M, Wang C, Sun Y, Lin H, Xu C. Memristor-based affective associative memory neural network circuit with emotional gradual processes Neural Computing and Applications. *Neural Comput Appl* (2022) 34:1–16. doi:10.1007/s00521-022-07170-z

13. Lin H, Wang C, Chen C, Zhou C, Xu C, Hong Q, et al. Neural bursting and synchronization emulated by neural networks and circuits. *IEEE Trans Circ Syst.I: Regular Pap* (2021) 68(8):3397–410. doi:10.1109/TCSI.2021.3081150

14. Liu Y, Iu HHC, Qian Y. Implementation of Hodgkin-Huxley neuron model with the novel memristive oscillator. *IEEE Trans Circ Syst Express Briefs* (2021) 68(8):2982–6. doi:10.1109/TCSII.2021.3066471

15. Shen H, Yu F, Wang C, Sun J, Cai S. Firing mechanism based on single memristive neuron and double memristive coupled neurons. *Nonlinear Dyn* (2022) 110:3807–22. doi:10.1007/s11071-022-07812-w

16. Mannion DJ, Mehonic A, Ng WH, Linderman RW, Kenyon AJ. Memristor-based edge detection for spike encoded pixels. *Front Neurosci* (2020) 13:1386. doi:10.3389/fnins.2019.01386

17. Wang X, Yu J, Jin C, Iu HHC, Yu S. Chaotic oscillator based on memcapacitor and meminductor. *Nonlinear Dyn* (2019) 96:161–73. doi:10.1007/s11071-019-04781-5

18. Wang X, Gao M, Iu HHC, Wang C. Tri-valued memristor-based hyper-chaotic system with hidden and coexistent attractors. *Chaos Solitons and Fractals* (2022) 159: 112177. doi:10.1016/j.chaos.2022.112177

19. Sadecki J, Marszale W. Complex oscillations and two-parameter bifurcations of a memristive circuit with diode bridge rectifier. *Microelectronics J* (2019) 93(1):104636. doi:10.1016/j.mejo.2019.104636

20. Liu Y, Iu HHC, Guo S, Li H. Chaotic dynamics in memristive circuits with different ϕ - q characteristics. *Int J Circuit Theor Appl* (2021) 49(11):3540–58. doi:10.1002/cta.3112

21. Xu KD, Li D, Jiang Y, Chen Q. SPICE behaviors of double memristor circuits using cosine window function. *Front Phys* (2021) 9:648737. doi:10.3389/fphy.2021.648737

22. Bao B, Wu P, Bao H, Chen M, Xu Q. Chaotic bursting in memristive diode bridge-coupled Sallen-Key lowpass filter. *Elect Lett* (2017) 53(16):1104–5. doi:10.1049/el.2017.1647

23. Liu Y, Li H, Guo SX, Iu HHC. Generation of multi-lobe Chua corsage memristor and its neural oscillation. *Micromachines* (2022) 13(8):1330. doi:10.3390/mi13081330

24. Wang X, Zhou P, Jason K, Eshraghian H, Lin C, Iu HHC, et al. High-density memristor-CMOS ternary logic family. *IEEE Trans Circuits Systems–I: Regular Pap* (2021) 68(1):264–74. doi:10.1109/TCSI.2020.3027693

25. Marszalek W. Autonomous models of self-crossing pinched hystereses for mem-elements. *Nonlinear Dyn* (2018) 92(4):1975–83. doi:10.1007/s11071-018-4175-4

26. Yu F, Shen H, Yu Q, Kong X, Sharma PK, Cai S. Privacy protection of medical data based on multi-scroll memristive Hopfield neural network. *IEEE Trans Netw Sci Eng* (2022) 2022:1–14. doi:10.1109/TNSE.2022.3223930

27. Yu F, Kong X, Mokbel AAM, Yao W, Cai S. Complex dynamics, hardware implementation and image encryption application of multiscroll memeristive Hopfield neural network with a novel local active memeristor. *IEEE Trans Circ Syst.–II: Express Briefs* (2022) 70:326. doi:10.1109/TCSII.2022.3218468

28. Yu F, Yu Q, Chen H, Kong X, Mokbel AAM, Yao W, et al. Dynamic Analysis and audio encryption application in IoT of a Multi-Scroll fractional-order memristive Hopfield neural network. *Fractal and Fractional* (2022) 6(7):370. doi:10.3390/fractalfract6070370

29. Biolek D, Biolek Z, Biolkova V. Pinched hysteretic loops of ideal memristors, memcapacitors and meminductors must be 'self-crossing. *Elect Lett* (2011) 47(25):1385–7. doi:10.1049/el.2011.2913

30. Biolek Z, Biolek D, Biolkova V. Hysteresis versus PSM of ideal memristors, memcapacitors, and meminductors. *Elect Lett* (2016) 52(20):1669–71. doi:10.1049/el.2016.2138

31. Guo Z, Iu HHC, Si G, Xu X, Oresanya BO, Bie Y. A phasor analysis method for charge-controlled memory elements. *Int J Bifurcation Chaos* (2020) 30(14):2030041. doi:10.1142/S0218127420300414

32. Liu Y, Guo Z, Chau TK, Iu HHC, Si G. Nonlinear circuits with parallel-/series-connected HP-type memory elements and their characteristic analysis. *Int J Circuit Theor Appl* (2021) 49(2):513–32. doi:10.1002/cta.2915

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 22 February 2023
DOI 10.3389/fphy.2023.1138643

OPEN ACCESS

EDITED BY
Fei Yu,
Changsha University of Science and Technology, China

REVIEWED BY
Bengier Ulgen Kilic,
University at Buffalo, United States
Henrik Jeldtoft Jensen,
Imperial College London, United Kingdom
Paolo Grigolini,
University of North Texas, United States

*CORRESPONDENCE
Vitaly L. Galinsky,
vit@ucsd.edu

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 05 January 2023
ACCEPTED 02 February 2023
PUBLISHED 22 February 2023

CITATION
Galinsky VL and Frank LR (2023), Critical brain wave dynamics of neuronal avalanches.
Front. Phys. 11:1138643.
doi: 10.3389/fphy.2023.1138643

Critical brain wave dynamics of neuronal avalanches

Vitaly L. Galinsky[1]* and Lawrence R. Frank[1,2]

[1]Center for Scientific Computation in Imaging, University of California, San Diego, San Diego, CA, United States, [2]Center for Functional MRI, University of California, San Diego, San Diego, CA, United States

Analytical expressions for scaling of brain wave spectra derived from the general non-linear wave Hamiltonian form show excellent agreement with experimental "neuronal avalanche" data. The theory of the weakly evanescent non-linear brain wave dynamics reveals the underlying collective processes hidden behind the phenomenological statistical description of the neuronal avalanches and connects together the whole range of brain activity states, from oscillatory wave-like modes, to neuronal avalanches, to incoherent spiking, showing that the neuronal avalanches are just the manifestation of the different non-linear side of wave processes abundant in cortical tissue. In a more broad way these results show that a system of wave modes interacting through all possible combinations of the third order non-linear terms described by a general wave Hamiltonian necessarily produces anharmonic wave modes with temporal and spatial scaling properties that follow scale free power laws. To the best of our knowledge this has never been reported in the physical literature and may be applicable to many physical systems that involve wave processes and not just to neuronal avalanches.

KEYWORDS

non-linear waves, critical exponent, Hamiltonian system, neuronal avalanches, critical dynamics

1 Introduction

The complexity of oscillatory and wave patterns across a wide range of spatial and temporal scales of brain activity results in multiple independent models for these activity patterns. The standard view of brain electromagnetic activity classifies this activity into two significant but essentially independent classes. The first class includes a variety of the oscillatory and wave-like patterns that show relatively high level of coherence across a wide range of spatial and temporal scales [3]. The second class focusses on the asynchronous, seemingly incoherent spiking activity at scales of a single neuron and often uses various *ad hoc* neuron models [4–8] to describe this activity. Linking these two seemingly disparate classes to explain the emergence of oscillatory rhythms from incoherent activity is essential to understanding brain function and is typically posed in the form using the construct of networks of incoherently spiking neurons [9–11].

Coherent macroscopic behavior arising from seemingly incoherent microscopic processes naturally suggests the influence of critical phenomena, a potential model from brain activity that was bolstered by the experimental discovery of the "neuronal avalanches" [12, 13] where both spatial and temporal distributions of spontaneous propagating neuronal activity in 2D cortex slices were shown to follow scale-free power laws. This discovery has generated significant interest in the role and the importance of criticality in brain activity [14–20]. Crucial events, as a manifestation of criticality, have been discussed in [21] using Diffusion Entropy Analysis [22]. It was also hypothesized that an existence of crucial events facilitates information transmission in various states of brain functioning [23, 24].

Although the precise neuronal mechanisms leading to the observed scale-free avalanche behavior is still uncertain after almost 20 years since their discovery, the commonly agreed upon paradigm is that this collective neuronal avalanche activity represents a unique and specialized pattern of brain activity that exists somewhere between the oscillatory, wave-like coherent activity and the asynchronous and incoherent spiking. Central to this claim of neuronal avalanches as a unique brain phenomena is that they do not show either wave-like propagation or synchrony at short scales, and thus constitute a new mode of network activity [12, 13] that can be phenomenologically described using the ideas of the self-organized criticality [25, 26], and extended to the mean-field theory of the self-organized branching processes (SOBP) [27–29].

However, despite the success of the SOBP theory in describing neuronal avalanche statistical properties, i.e., replicating the power law exponents based on the criticality considerations, the SOBP theory provides no explanation about the physical mechanisms of the critical behavior and its relationship to the development of the observed collective neuronal "avalanche" behavior. Because similar statistics can result from several mechanisms other than critical dynamics [30–32], it is essential to have a physical model that explains the relationship between the statistical properties and the existence, if any, of critical neural phenomena arising from the actual collective behavior of neuronal populations. While it is generally accepted in that some form of critical phenomena is at work, this has led to the presupposition of *ad hoc* descriptive models [33–36] that exhibit critical behavior, but provide no insight into the actual physical mechanisms that might produce such critical dynamics. It has been suggested that the brain can be at the edge of a synchronization phase transition [36–38], but the usual agreement is that avalanches belong to the mean-field directed percolation universality class, which does not seem to be compatible with a synchronization transition, as synchronization transitions do not fulfill spatial correlations observed in experiments, and the exponents tend to differ from directed percolation ones [20].

In this paper we show that the above important observational phenomenon, the so-called "neuronal avalanches", which have been noted and studied for almost 2 decades, can be naturally explained by the wave activity model. Our recently described theory of weakly evanescent brain waves (WETCOW) originally developed in [1, 2] and then reformulated in a general Hamiltonian framework [39] provides a physical theory, based on the propagation of electromagnetic fields through the highly complex geometry of inhomogeneous and anisotropic domain of real brain tissues, that explains the broad range of observed seemingly disparate brain wave characteristics. This theory produces a set of non-linear equations for both the temporal and spatial evolution of brain wave modes that includes all possible non-linear interaction between propagating modes at multiple spatial and temporal scales and degrees of non-linearity. This theory bridges the gap between the two seemingly unrelated spiking and wave "camps" as the generated wave dynamics includes the complete spectra of brain activity ranging from incoherent asynchronous spatial or temporal spiking events, to coherent wave-like propagating modes in either temporal or spatial domains, to collectively synchronized spiking of multiple temporal or spatial modes.

We further demonstrate that the origin of these "avalanche" properties emerges directly from the same theory that produces this wide range of activity and does not require one to posit the existence of either new brain activity states, nor construct analogies between brain activity and *ad hoc* generic "sandpile" models. Both temporal and spatial scaling expressions analytically derived from non-linear amplitude/phase evolutionary equations show excellent agreement with the experimental neural avalanche probability spectra reproducing not only general average power law exponent values and falloffs in the vicinity of the critical point, but also finding some very subtle but nevertheless clearly experimentally evident fine details, like bumps in the transition region at the edge of the power leg of the spatial probability spectra. Overall, the quantitative theoretical analysis presented in the paper clearly shows the relevance of the wave Hamiltonian framework to the neuronal avalanche dynamics and suggests that instead of relying on clever but *ad hoc* analogies between live brain tissues and lifeless loose sand piles often used to construct a phenomenological statistical description of the scaling exponents, both the origin of the critical phenomena and the physics behind the neuronal avalanches can be understood from the same non-linear wave dynamics that is responsible for the wide range of activities in the brain tissue, ranging from the linear coherently propagating waves to the non-linear incoherent asynchronous spiking, including in between the peculiar power law-like coherence of the neuronal avalanches.

We emphasize that although the general WETCOW theory describes complex interactions between modes, the explanation for neuronal avalanches and their attendant scaling properties presented in this paper are based on the analysis of a *single wave mode* with completely arbitrary set of mode parameters. This includes arbitrary amplitude, phase, frequency, and criticality. No interaction between modes, except a general form of the third order non-linearity that characterizes anharmonicity of the non-linear wave modes due to non-resonant interaction of the linear modes, is needed to derive the scaling result. Thus a key result of this paper is the demonstration that neuronal avalanches and their attendant scaling properties are obtained within the simplest form of the WETCOW theory where mode coupling is ignored, but significantly without the *ad hoc* and physically implausible assumptions typically made that the parameters of all network nodes are either constant and the same for all nodes [36], sometimes even including inter-mode coupling [38], or are generated from some *ad hoc* artificial distributions [40], and require the addition of stochastic noise properties [41], *etc.* This emphasizes generality and importance of our derivation.

2 Weakly evanescent brain waves

Beginning from our non-linear Hamiltonian formulation of the WETCOW theory [39], we have for an anharmonic wave mode

$$H^s(a,a^\dagger) = \Gamma a a^\dagger + a a^\dagger\left[\beta_a a + \beta_{a^\dagger} a^\dagger - 2\alpha\left(a a^\dagger\right)^{1/2}\right] \qquad (1)$$

where a is a complex wave amplitude and $a^\dagger$ is its conjugate. The amplitude a denotes either temporal $a_k(t)$ or spatial $a_\omega(x)$ wave mode amplitudes that are related to the spatiotemporal wave field $\psi(x, t)$ through a Fourier integral expansions.

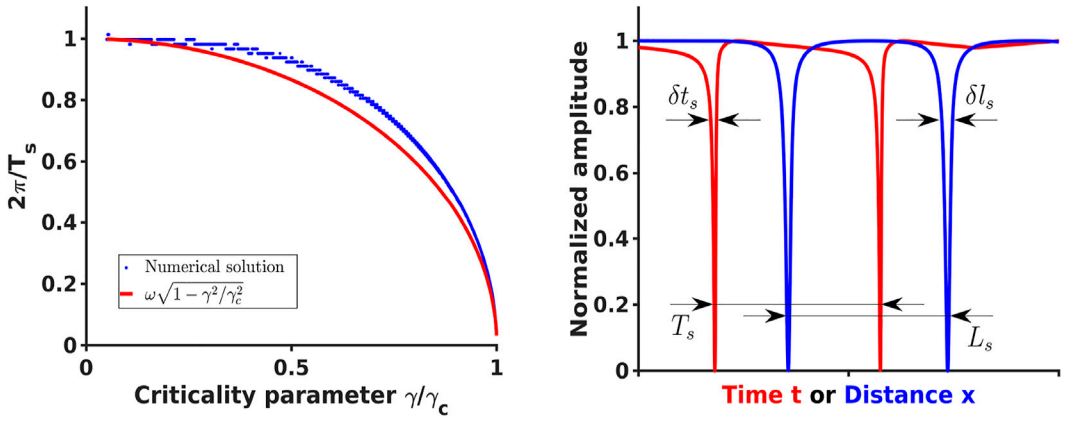


FIGURE 1
[Left] Comparison of the analytical expression (29) for the effective spiking frequency $\omega_s = 2\pi/T_s$ (red) and the frequency estimated from numerical solution of Eq. 21 and Eq. 22 (blue) as a function of the criticality parameter γ/γ_c. In the numerical solution only γ was varied and the remaining parameters were the same as parameters reported in [39] [Right] Spiking solutions for typical parameters producing temporal ((21) and (22), red) and spatial ((38) and (39), blue) spiking profiles where some signal of width δt_s or δl_s was detected and surrounded by quiet area with the total effective period T_s or L_s.

$$a_k(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \psi(x,t)e^{-i(kx+\omega_k t)}dx, \tag{2}$$

$$a_\omega(x) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \psi(x,t)e^{-i(k_\omega x+\omega t)}dt, \tag{3}$$

where for the sake of clarity we use one dimensional scalar expressions for spatial variables x and k, but it can be easily generalized for the multi dimensional wave propagation as well. The spatiotemporal wave field $\psi(x, t)$ is a superposition of multiple waves, that may include neuronal firings, membrane sub-threshold oscillations, LFPs, *etc.* The frequency ω and the wave number k of the wave modes satisfy dispersion relation $D(\omega, k) = 0$, and ω_k and k_ω denote the frequency and the wave number roots of the dispersion relation (the structure of the dispersion relation and its connection to the brain tissue properties has been discussed in [1]).

The first term $\Gamma aa^\dagger$ in Eq.1 denotes the harmonic (quadratic) part of the Hamiltonian with either the complex valued frequency $\Gamma = i\omega + \gamma$ or the wave number $\Gamma = ik + \lambda$ that both include a pure oscillatory parts (ω or k) and possible weakly excitation or damping rates, either temporal γ or spatial λ. The second anharmonic term is cubic in the lowest order of non-linearity and describes the interactions between various propagating and non-propagating wave modes, where α, β_a and $\beta_{a^\dagger}$ are the complex valued strengths of those different non-linear processes. As it was shown in [1, 2] the inverse proportionality of frequency and wave number in the dispersion relation 6) results in the third order expressions for the non-resonant coupling between multiple waves.

Distribution of various charges in brain tissue, including free ionic charges (sodium, potassium, etc), bonded macromolecular volume charges, membrane polarization and/or surface charges, *etc.*, determines brain electrodynamics. The voltages and currents measured in real brains are produced by those electrodynamic processes that in the most general form can be represented by the Maxwell equations together with state or motion equations for the brain matter, particularly by a generalized Ohm's law, that describes electro-diffusive flow of charged particles through inhomogeneous media (that may include both concentration and voltage gradients). The neuron action potential itself is nothing more than propagating non-linear electrostatic wave described by the same electrodynamics formalism. A set of derivations that lead to this description was presented in details in [1, 2, 39] and is based on considerations that follow from the most general form of brain electromagnetic activity expressed by Maxwell equations in inhomogeneous and anisotropic medium

$$\nabla\cdot\boldsymbol{D} = \rho, \quad \nabla\times\boldsymbol{H} = \boldsymbol{J} + \frac{\partial \boldsymbol{D}}{\partial t} \quad \Rightarrow \quad \frac{\partial\rho}{\partial t} + \boldsymbol{\nabla}\cdot\boldsymbol{J} = 0.$$

Using the electrostatic potential $\boldsymbol{E} = -\nabla\psi$, Ohm's law $\boldsymbol{J} = \boldsymbol{\sigma}\cdot\boldsymbol{E}$ (where $\boldsymbol{\sigma} \equiv \{\sigma_{ij}\}$ is an anisotropic conductivity tensor), a linear electrostatic property for brain tissue $\boldsymbol{D} = \varepsilon\boldsymbol{E}$, assuming that the scalar permittivity ε is a "good" function (i.e. it does not go to zero or infinity everywhere) and taking the change of variables $\partial x \to \varepsilon\partial x'$, the charge continuity equation for the spatial-temporal evolution of the potential ψ can be written in terms of a permittivity scaled conductivity tensor $\boldsymbol{\Sigma} = \{\sigma_{ij}/\varepsilon\}$ as

$$\frac{\partial}{\partial t}\left(\nabla^2\psi\right) = -\boldsymbol{\nabla}\cdot\boldsymbol{\Sigma}\cdot\nabla\psi + \mathcal{F}, \tag{4}$$

where we have included a possible external source (or forcing) term $\mathcal{F}$. For brain fiber tissues the conductivity tensor $\boldsymbol{\Sigma}$ might have significantly larger values along the fiber direction than across them. The charge continuity without forcing i.e., ($\mathcal{F} = 0$) can be written in tensor notation as

$$\partial_t\partial_i^2\psi + \Sigma_{ij}\partial_i\partial_j\psi + \left(\partial_i\Sigma_{ij}\right)\left(\partial_j\psi\right) = 0, \tag{5}$$

where repeating indices denote summation. Simple linear wave analysis, i.e. substitution of $\psi \sim \exp(-i\ (\boldsymbol{k}\cdot\boldsymbol{r} - \Omega t))$, where $\boldsymbol{k}$ is the wavenumber, $\boldsymbol{r}$ is the coordinate, $\Omega = \omega + i\gamma$ is the frequency and t is the time, gives the following complex dispersion relation:

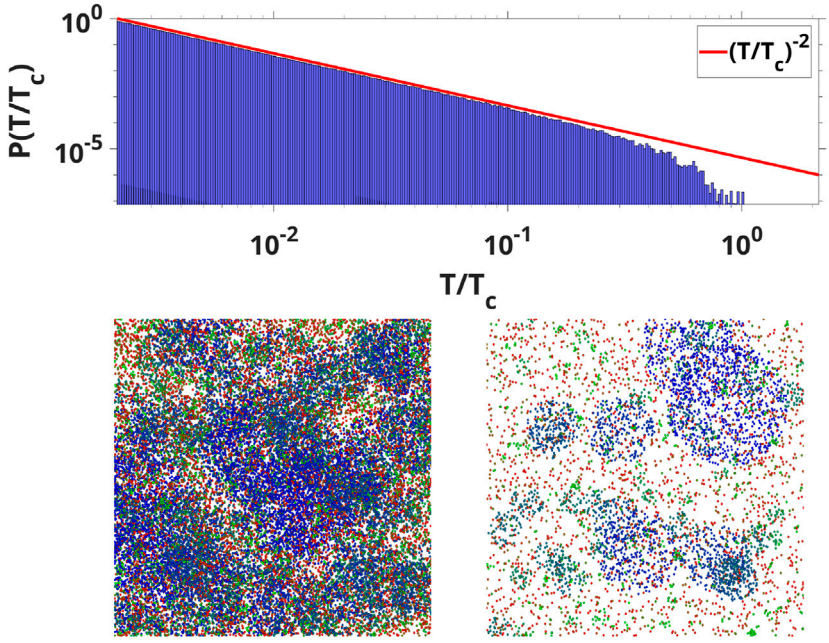


FIGURE 2
(Top) The avalanche durations distribution for all wave modes compared with the -2 exponent (Bottom) WETCOW modes randomly distributed and propagated on a 1,000 by 1000 grid. Two examples of temporal signal snapshots with different values of signal threshold are shown (color pallet encodes the change of frequencies from the smallest (blue) to the largest (red). Localized regions of wave activity in the spatial domain are clearly evident.

$$D(\Omega, \boldsymbol{k}) = -i\Omega k_i^2 - \Sigma_{ij} k_i k_j - i\partial_i \Sigma_{ij} k_j = 0, \tag{6}$$

which is composed of the real and imaginary components:

$$\gamma \equiv \mathfrak{I}\Omega = \Sigma_{ij}\frac{k_i k_j}{k^2} \qquad \omega \equiv \mathfrak{R}\Omega = -\frac{\partial_i \Sigma_{ij} k_j}{k^2} \tag{7}$$

Although in this general form the electrostatic potential ψ, as well as the dispersion relation $D(\Omega, \boldsymbol{k})$, describe three dimensional wave propagation, we have shown [1, 2] that in anisotropic and inhomogeneous media some directions of wave propagation are more equal than others with preferred directions determined by the complex interplay of the anisotropy tensor and the inhomogeneity gradient. While this is of significant practical importance, in particular because the anisotropy and inhomogeneity can be directly estimated from non-invasive methods, for the sake of clarity we focus here on the one dimensional scalar expressions for spatial variables x and k that can be easily generalized for the multi dimensional wave propagation as well.

The multiple temporal $a_k(t)$ or spatial $a_\omega(x)$ wave mode amplitudes can be used to define the time dependent wave number energy spectral density $\Pi_k(t)$ or the position dependent frequency energy spectral density $\Pi_\omega(x)$ for the spatiotemporal wave field $\psi(x, t)$ as

$$\Pi_k(t) = |a_k(t)|^2, \qquad \Pi_\omega(x) = |a_\omega(x)|^2, \tag{8}$$

or alternatively we can add additional length or time normalizations to convert those quantities to power spectral densities instead.

The network Hamiltonian form that describes discrete spectrum of those multiple wave modes was presented in [39] as

$$H(\boldsymbol{a}, \boldsymbol{a}^\dagger) = \sum_n \left[H^s(a_n, a_n^\dagger) + \sum_{m \neq n} \left(a_n r_{nm} a_m^\dagger + a_n^\dagger r_{nm}^* a_m \right) \right] \tag{9}$$

where the single mode amplitude a_n again denotes either a_k or a_ω, $\boldsymbol{a} \equiv \{a_n\}$ and $r_{nm} = R_{nm} e^{i\Delta_{nm}}$ is the complex network adjacency matrix with R_{nm} providing the coupling power and Δ_{nm} taking into account any possible differences in phase between network nodes. This description includes both amplitude $\mathfrak{R}(a)$ and phase $\mathfrak{I}(a)$ mode coupling and as shown in [39] allows for significantly unique synchronization behavior different from both phase coupled Kuramoto oscillator networks and from networks of amplitude coupled integrate-and-fire neuronal units.

The third order nature of the theory is of particular interest, and provide the theory with a broad range of applicability. It has distinctly different characteristics than the harmonic oscillator. Of particular importance is the fact that the third order terms become important when wave amplitudes are high enough but only if or until higher order terms are absent or suppressed by some physical mechanism. This suppression becomes significant in incorporating the anisotropic inhomogeneous and resistive nature of brain tissues. An important consequence derived in [1] is that the inverse frequency–wave number proportionality of the linear wave dispersion guarantees that the resonant terms higher than the third order are not important and can be neglected and, at the same time, the non-resonant third order terms (that are typically excluded when compared to the resonant terms) should now be retained resulting in the third order form of Hamiltonian 1). It is our contention, and the subject of future studies, that the anharmonic third order forms 1) and 9) are not brain specific

and can be used to describe oscillations and waves in active media abundant in various areas of physics.

Although the Fourier integrals 2 used for expansion of the spatiotemporal wave field ψ into a set of wave modes imply presence of a large (actually infinite) number of modes in the network Hamiltonian 9 the derivation of neuronal avalanches is evident even without this generality of this coupling between modes expressed by the coupling parameters r_{nm}, as it was done in [39]. Thus we will consider an ensemble of non-interacting modes, effectively setting $r_{nm} = 0$, for the analysis of this paper. But importantly we will *not* make any assumptions about parameters of the single mode Hamiltonian form 1, assuming that all parameters (Γ, β_a, $\beta_{a^\dagger}$, α) are arbitrary and do not carry any mode dependence. This is a non-trivial point worth emphasizing, as it is a departure from the extant literature wherein the *ad hoc*, and physically implausible, assumption of the equivalence of network nodes is made. Therefore, we will proceed with our analysis of a single mode amplitude a suppressing all subscripts and indices, and assuming that a denotes a_n where n may represent either an arbitrary wave number k from a range of wave numbers $k_0 \le k \le k_1$ or an arbitrary frequency ω from a range of wave frequencies $\omega_0 \le \omega \le \omega_1$.

3 Single anharmonic mode criticality

An equation for the non-linear oscillatory amplitude a then can be expressed as a derivative of the Hamiltonian form

$$\frac{da}{dt} = \frac{\partial H^s}{\partial a^\dagger} \equiv \Gamma a + \beta_{a^\dagger} a a^\dagger + \beta_a a^2 - \alpha a \left(a a^\dagger\right)^{1/2}, \tag{10}$$

after removing the constants with a substitution of $\beta_{a^\dagger} = 1/2\tilde{\beta}_{a^\dagger}$ and $\alpha = 1/3\tilde{\alpha}$ and dropping the tilde. As frequencies and wave numbers for linear waves satisfy the dispersion relation 6), they are related and the same Hamiltonian expression 1) can be used either for temporal $a_k(t)$ or spatial $a_\omega(x)$ wave amplitudes. Therefore, we note that although (10) is an equation for the temporal evolution, the spatial evolution of the mode amplitudes $a_\omega(x)$ can be described by a similar equation substituting temporal variables by their spatial counterparts, i.e., $(t, \omega, \gamma) \to (x, k, \lambda)$.

Splitting (10) into an amplitude/phase pair of equations using $a = Ae^{i\phi}$, assuming $\beta_a = \tilde{\beta}_a e^{-i\delta_a}$, $\beta_{a^\dagger} = \tilde{\beta}_{a^\dagger} e^{i\delta_{a^\dagger}}$, and scaling the variables as

$$A = \gamma\tilde{A}, \quad t = \frac{\tau}{\gamma}, \quad \omega = \tilde{\omega}\gamma, \tag{11}$$

gives the set of equations.

$$\frac{d\tilde{A}}{d\tau} = \tilde{A} + \tilde{A}^2\left(\beta_{a^\dagger}\cos\Psi_{a^\dagger} + \beta_a \cos\Psi_a - \alpha\right) \tag{12}$$

$$\frac{d\phi}{d\tau} = \tilde{\omega} + \tilde{A}\left(-\beta_{a^\dagger}\sin\Psi_{a^\dagger} + \beta_a \sin\Psi_a\right) \tag{13}$$

where $\Psi_a \equiv \phi - \delta_a$, $\Psi_{a^\dagger} \equiv \phi - \delta_{a^\dagger}$.

These equations can further be cast into a more compact form by defining

$$\boldsymbol{\beta} = \begin{pmatrix} \beta_a \\ \beta_{a^\dagger} \end{pmatrix}, \ \boldsymbol{u} = \begin{pmatrix} e^{i\delta_a} \\ e^{i\delta_{a^\dagger}} \end{pmatrix}, \ \boldsymbol{v} = \begin{pmatrix} ie^{i\delta_a} \\ -ie^{i\delta_{a^\dagger}} \end{pmatrix} \tag{14}$$

so that.

$$z_a = \boldsymbol{\beta}\cdot\boldsymbol{u} = X_a + iY_a \tag{15}$$

$$z_\phi = \boldsymbol{\beta}\cdot\boldsymbol{v} = X_\phi + iY_\phi \tag{16}$$

where.

$$R_a = |z_a| = \sqrt{X_a^2 + Y_a^2} \tag{17}$$

$$R_\phi = |z_\phi| = \sqrt{X_\phi^2 + Y_\phi^2} \tag{18}$$

$$\Phi_a = \arg(z_a) = \arctan\frac{Y_a}{X_a} \tag{19}$$

$$\Phi_\phi = \arg\left(z_\phi\right) = \arctan\frac{Y_\phi}{X_\phi} \tag{20}$$

whereupon (12) and (13) can be rewritten.

$$\frac{d\tilde{A}}{d\tau} = \tilde{A} + \tilde{A}^2\left[R_a\cos\left(\phi - \Phi\right) - \alpha\right], \tag{21}$$

$$\frac{d\phi}{d\tau} = \tilde{\omega} + \tilde{A}R_\phi\cos\phi, \tag{22}$$

where $\Phi = \Phi_a - \Phi_\phi$.

A stationary (i.e., $d\tilde{A}/d\tau = d\phi/d\tau = 0$) solution of (21) and (22) can be found from

$$-\frac{R_\phi}{\tilde{\omega}}\cos\phi + R_a\cos\left(\phi - \Phi\right) - \alpha = 0, \tag{23}$$

as $\phi_e = \phi_0 \equiv$ const and $\tilde{A}_e = \tilde{\omega}/R_\phi\cos\phi_0 \equiv$ const. This shows that for $\alpha > R_a$ there exist critical values of $\tilde{\omega}$ and A_e, where the stationary solution disappears and is replaced by non-linear oscillations, such that.

$$\tilde{\omega}_c = \frac{R_\phi\cos\phi_c}{\alpha + R_a\cos\left(\phi_c + \Phi\right)}, \quad \tilde{A}_c = \tilde{\omega}_c/R_\phi, \tag{24}$$

$$\phi_c = \arctan\left[\frac{R_a\sin\Phi}{\sqrt{\alpha^2 - \left(R_a\sin\Phi\right)^2}}\right], \tag{25}$$

which can also be expressed in terms of critical value of one of the unscaled variables, either ω or γ

$$\omega_c = \gamma\tilde{\omega}_c, \quad \text{or} \quad \gamma_c = \frac{\omega}{\tilde{\omega}_c}, \tag{26}$$

This stationary solution provides the locus of the saddle node on an invariant circle bifurcation point at where the non-linear spiking oscillations occur (as was shown both in [1, 2] and in [39]).

4 Effective spiking rate

The effective period $\mathcal{T}_s$ of spiking solutions of (21) and (22) (or its inverse–either the firing rate $1/\mathcal{T}_s$ or the effective firing frequency $2\pi/\mathcal{T}_s$) can be estimated from (22) by substituting $\tilde{A}_c$ for $\tilde{A}$ (assuming that the change of amplitude $\tilde{A}$ is slower than the change of the phase ϕ) as

$$\mathcal{T}_s = \int_0^{2\pi}\frac{d\phi}{\tilde{\omega} + \tilde{\omega}_c\cos\phi} = \frac{2\pi}{\sqrt{\tilde{\omega}^2 - \tilde{\omega}_c^2}}, \tag{27}$$

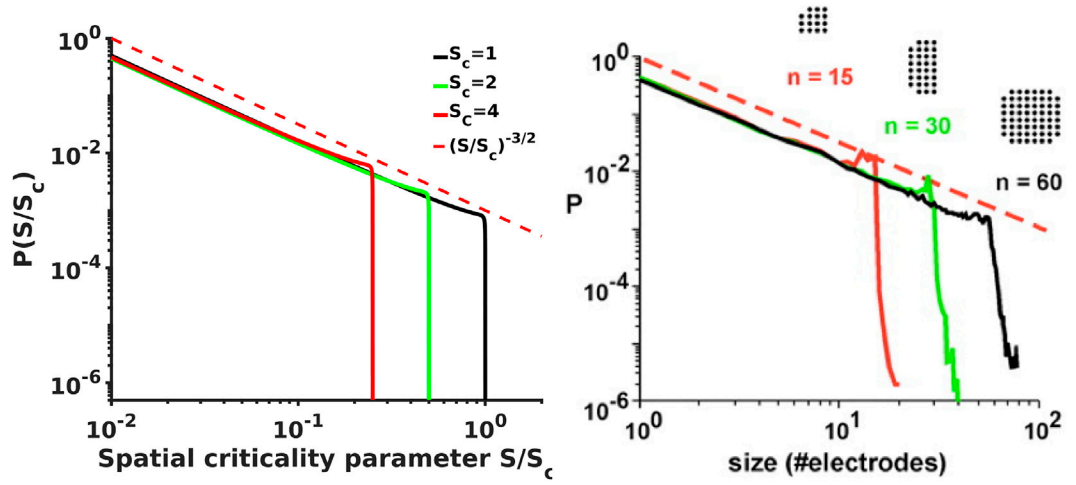


FIGURE 3
[Left] Analytical probability density spectra as a function of brain waves criticality parameter S/S_c show excellent agreement with the experimental avalanche data [Right, from [12, 13]] reproducing not only the overall shape of the spectra with the -3/2 power exponent at the initial scale free part of the spectra and the steep falling edge in the vicinity of the critical point, but also reproduce the fine details such as the small bump-like flattening of the spectra at the transition from -3/2 leg to the steep falling edge that is clearly evident in experimental spectra.

giving the unscaled effective spiking period T_s and the effective firing frequency ω_s

$$T_s = \frac{\mathcal{T}_s}{\gamma} = \frac{2\pi}{\omega\sqrt{1-\gamma^2/\gamma_c^2}} = \frac{2\pi}{\omega\sqrt{1-\omega_c^2/\omega^2}}, \tag{28}$$

$$\omega_s = \frac{2\pi}{T_s} = \omega\sqrt{1-\omega_c^2/\omega^2}, \tag{29}$$

with the periodic amplitude $\tilde{A}$ reaching the maximum $\tilde{A}_{max} = 1/(\alpha - R_a)$ and the minimum $\tilde{A}_{min} = 1/(\alpha + R_a)$ for $d\tilde{A}/d\tau = 0$ when $\phi = \Phi$ and $\phi = \Phi + \pi$ respectively.

The expressions (28) and (29) are more general than typically used expressions for the scaling exponent in the close vicinity $|\gamma - \gamma_c| \ll \gamma_c$ of the critical point [42–44]. They allow recovery of the correct T limits both at $\gamma \to \gamma_c$ with the familiar $T \sim 1/\sqrt{\gamma_c - \gamma}$ scaling and at $\gamma \sim 0$ with the period T approaching T_0 as $T \sim T_0 + O(\gamma^2) \equiv 2\pi/\omega + O(\gamma^2)$, where T_0 is the period of linear wave oscillations with the frequency ω. In the intermediate range $0 < \gamma < \gamma_c$ the expressions (28) and (29) show reasonable agreement (Figure 1) with peak–to–peak period/frequency estimates from direct simulations of system (12) and (13).

5 Temporal probability of single spike detection

As the periodic solution of Eq. 21 and Eq. 22 in the $0 \ll \gamma \ll \gamma_c$ range looks like linear waves at γ close to zero, but transforms to spike as γ increases, we can approximate the probability of detecting a single spike by a ratio of a spike peak duration (recorded above some threshold) to a total peak-to-peak time. Taking into account that the initial phase of spiking solutions of Eq. 21 and Eq. 22 is a random variable uniformly distributed on $[0, 2\pi]$ interval, the probability that a spike (either positive or the more frequently experimentally reported negative [12, 13]) with duration width δt_s and with the total period between the spikes (T_s) will be detected is simply $\delta t_s/T_s$–where the distance between spikes is determined as the time interval needed for 2π radian phase change, that is the effective spiking period T_s. Assuming initially that the spike width δt_s does not change when approaching the critical point ω_c, δt_s can be approximated by some fixed fraction of the linear wave period, i.e., $\delta t_s \sim \pi/\omega$, that gives for the probability density

$$P_k^{\{\omega\}}(\omega, \omega_c) \sim \omega^{-1}\sqrt{\omega^2/\omega_c^2 - 1}, \tag{30}$$

for every wave mode with the wavenumber k. It should be noted that the probability density $P_k^{\{\omega\}}$ has no relation to, and should not be confused, with the frequency energy spectral density $\Pi_\omega(x)$ (or with the power spectral density).

Transforming the frequency dependence of the wavenumber spectra $P_k^{\{\omega\}}$ to the temporal domain ($T = 2\pi/\omega$, $T_c = 2\pi/\omega_c$)

$$\begin{aligned}\int_{\omega_c}^{\infty} P_k^{\{\omega\}}(\omega, \omega_c)d\omega &= \int_0^{T_c} P_k^{\{\omega\}}\left(\frac{2\pi}{T}, \frac{2\pi}{T_c}\right)\frac{2\pi}{T^2}dT \\ &= \int_0^{T_c} P_k^{\{T\}}(T, T_c)dT,\end{aligned} \tag{31}$$

gives for the temporal probability density $P_k^{\{T\}}$

$$P_k^{\{T\}}(T, T_c) \sim T^{-2}\sqrt{1 - T^2/T_c^2}, \tag{32}$$

hence the scaling of the temporal probability density $P_k^{\{T\}}$ follows the power law with -2 exponent with additional $\sqrt{1 - T/T_c}$ falloff in close vicinity of the critical point in agreement with temporal scaling of neuronal avalanches reported in [12, 13].

6 Multi-mode avalanche probability

The above single wave mode analysis shows that the probability density $P_k^{\{T\}}$ for any arbitrary selected wave mode k with arbitrary chosen threshold follows a power law distribution with -2 exponent, therefore, a mixture of multiple wave modes that enters into the

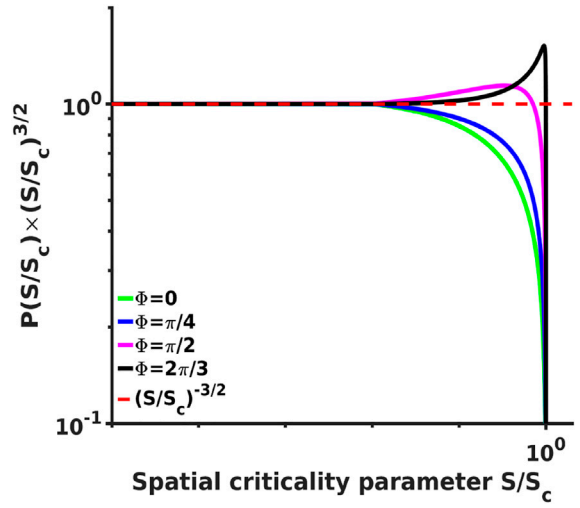


FIGURE 4
Analytical probability density spectra multiplied by a $(S/S_c)^{3/2}$ as a function of brain waves criticality parameter S/S_c plotted for several values of the phase shift Φ.

spatiotemporal wave field $\psi(x, t)$ with different amplitudes and different thresholds will again show nothing more than the same power law distribution.

To clearly demonstrate that the probability density function $P_k^{\{T\}}$ of finding a spike reflects the avalanche duration distribution we conducted a simple numerical experiment using a procedure that replicates the original experimental method of computing neuronal avalanches employed in the original papers by Beggs and Plenz [12, 13]. We used 10^6 wave modes with arbitrary parameters and computed avalanches by cutting the temporal series with a threshold (converting the event to a single dot or "spike"), then binning the signal using a time equal to the average inter-spike interval. After that, an avalanche duration is given by the time between two empty bins. Figure 2 compares the avalanche distribution for all wave modes with the -2 exponent.

Similar proof of equivalence of the single mode probability density function $P_k^{\{T\}}$ and a probability density of a multi-mode avalanche event obtained by the method replicating the experimental demarcation of the quiescence, that we will denote as $p^a(T)$, can be also derived using simple analytical considerations. The probability $P^a_{0\le T'\le T+\Delta T}$ that an avalanche happens at any time between 0 and $T + \Delta T$, where ΔT is some small binning interval used by the above experimental method, can be expressed as

$$P^a_{0\le T'\le T+\Delta T} = \int_0^{T+\Delta T} p^a(T')dT' = \int_0^{T} p^a(T')dT' + \int_T^{T+\Delta T} p^a(T')dT' \approx P^a_{0\le T'\le T} + p^a(T)\Delta T. \tag{33}$$

Since the probability

$$P_{k_j}^{\{T\}}\left(T, T_{c_j}\right)\Delta T = \nu_j T^{-2}\sqrt{1 - T^2/T_{c_j}^2}\,\Delta T, \tag{34}$$

(where ν_j is an arbitrary mode specific proportionality constant) describes the probability of finding a signal for a single mode j ($j = 1 \ldots N$) in a time interval between T and $T + \Delta T$, the probability that the condition for detection of a multi mode avalanche is recorded in the same interval can be expressed as

$$p^a(T)\Delta T = [1 - P_0(T - \Delta T)] \times P_0(T) \quad \text{where} \quad P_0(T) = \prod_{j=1}^{N}\left(1 - P_{k_j}^{\{T\}}\left(T, T_{c_j}\right)\Delta T\right) \tag{35}$$

where all wave modes are assumed to be independent. The second factor P_0T) represents the probability that there is no signal for any of the modes detected between T and $T + \Delta T$. The first factor ($1 - P_0(T - \Delta T)$) makes sure that no avalanche was recorded in the previous ΔT bin, that is a signal for at least one mode has been found in the interval between $T - \Delta T$ and T.

An expansion of Eq. 35 in the leading order of ΔT gives for the avalanche probability density $p^a(T)$

$$p^a(T) \approx T^{-2}\sum_{j=1}^{N}\nu_j\sqrt{1 - T^2/T_{c_j}^2}, \tag{36}$$

that is the avalanche probability density $p^a(T)$ shows the same T^{-2} scaling as the probability density of finding signal for a single mode.

If additionally the criticality parameters T_{c_j} for all wave modes k_j are assumed to be the same ($T_{c_j} \equiv T_c$) then the avalanche probability density scaling takes exactly the same form as the single mode probability density

$$p^a(T) \sim P_k^{\{T\}}(T) \sim T^{-2}\sqrt{1 - T^2/T_c^2}. \tag{37}$$

7 Spatial spike detection probability

Due to the reciprocity of the temporal and spatial representations of the Hamiltonian form Eq. 1 equations for the spatial wave amplitude have the same form as the temporal equations Eq. 21 and Eq. 22

$$\frac{d\tilde{A}}{d\xi} = \tilde{A} + \tilde{A}^2\left[R_a\cos(\phi - \Phi) - \alpha\right], \tag{38}$$

$$\frac{d\phi}{d\xi} = \tilde{k} + \tilde{A}R_\phi\cos\phi, \tag{39}$$

under similar scaling (the spatial equivalent of Eq. 11) of the wave amplitude, the coordinate, and the wave number

$$A = \lambda\tilde{A}, \quad x = \frac{\xi}{\lambda}, \quad k = \tilde{k}\lambda. \tag{40}$$

In the spatial domain, this leads to the critical parameters $\tilde{A}_c$ and. $\tilde{k}_c$

$$\tilde{k}_c = \frac{R_\phi\cos\phi_c}{\alpha + R_a\cos(\phi_c + \Phi)}, \quad \tilde{A}_c = \tilde{k}_c\Big/R_\phi, \tag{41}$$

$$\phi_c = \arctan\left[\frac{R_a\sin\Phi}{\sqrt{\alpha^2 - (R_a\sin\Phi)^2}}\right], \tag{42}$$

Although our simple one dimensional scaling estimates do not take into account the intrinsic spatial scales of the brain, e.g., cortex radius of curvature, cortical thickness, *etc.*, nevertheless,

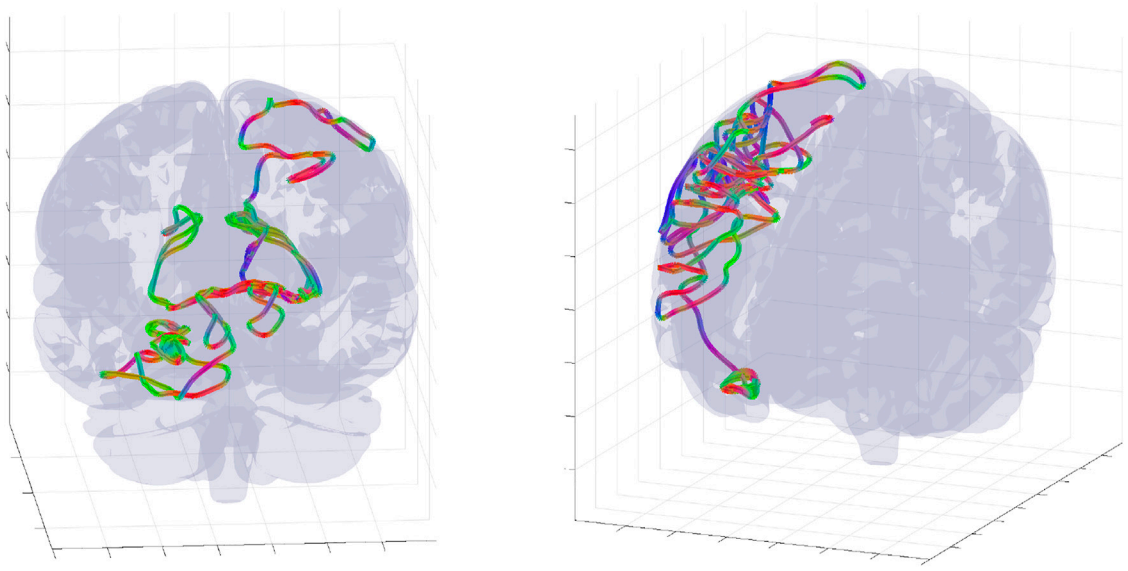


FIGURE 5
Examples of complete wave mode trajectory snapshots for two randomly selected parameters and initial conditions. The trajectories was randomly selected from an ensemble of 10^6 WETCOW modes used for generation of probability distributions of Figure 6.

even in this simplified form the similarity between spatial and temporal non-linear equations suggests that the non-linear spatial wave behavior will generally look like spiking in the spatial domain where some localized regions of activity are separated by areas that are relatively signal free and this separation will increase near the critical point. Exactly this behavior was reported in the original experimental studies of the neuronal avalanches [12, 13], where it was stated that the analysis of the contiguity index revealed that activity detected at one electrode is most often skipped over the nearest neighbors. Interestingly, this experimental observation of near critical non-linear waves was presented as an indicator that the activity propagation is not wave-like. But we see here that they are directly explained within the context of the WETCOW wave model. Of significant practical importance will be the effects of the intrinsic spatial scales of the brain that will certainly affect the details of the spatial critical wave dynamics and so their inclusion will be important for more completely characterizing the details of brain criticality and will be the focus of future investigations.

Using the spatial equations Eq. 38 and Eq. 39 similar scaling results can be obtained for the wave number k and the linear spatial dimension L probabilities for every wave mode with the frequency ω as.

$$P_{\omega}^{\{k\}}(k,k_c) \sim k^{-1}\sqrt{k^2/k_c^2-1}, \tag{43}$$

$$P_{\omega}^{\{L\}}(L,L_c) \sim L^{-2}\sqrt{1-L^2/L_c^2}, \tag{44}$$

where L is the linear spatial scale related to the wave number as $k = 2\pi/L$.

The linear spatial dimension of the avalanche L is related to its area S on a 2 dimensional surface as $L = \sqrt{S}$, hence

$$\int_0^{L_c} P_{\omega}^{\{L\}}(L,L_c)dL = \int_0^{S_c} \frac{P_{\omega}^{\{L\}}\left(\sqrt{S},\sqrt{S_c}\right)}{2\sqrt{S}}dS = \int_0^{S_c} P_{\omega}^{\{S\}}(S,S_c)dS, \tag{45}$$

$$P_{\omega}^{\{S\}}(S,S_c) \sim S^{-3/2}\sqrt{1-S/S_c}, \tag{46}$$

hence the spatial probability scaling for the size S follows the power law with -3/2 exponent again with additional $\sqrt{1-S/S_c}$ falloff in close vicinity of the critical point, that is also in agreement with experimentally reported spatial scaling of neuronal avalanches [12, 13]. We would like to mention that the non-linear anharmonic oscillations described by the (21) and (22) only exists for frequencies and wave numbers that are above the critical frequency ω_c or the critical wave number k_c values that define maximal possible temporal T_c or spatial L_c scales of the non-linear oscillations. If the finite system sizes are below those maximal values the cutoffs will be defined by the system scales.

We would like emphasize again the generality of our analysis that makes no assumptions about parameters used in Hamiltonian form Eq. 1, and hence in the equations Eq. 21 and Eq. 22 or (38) and (39), analytically deriving scaling valid for a wide (and arbitrary) range of those parameters. This is in striking difference from analyses and results based on oversimplified *ad hoc* numerical studies of synchronization in networks [36, 38]. Those typical numerical analysis studies consider networks of *completely identical* individual nodes sometimes even *globally* connected with *completely identical* weights. Therefore, all these studies require artificial (and significantly high levels of) noise added to each node just to be able to impose some range of scales into the system. This is an artificial and, as demonstrated here, unnecessary complication. The consequence of such models is that they are capable of obtaining something that resembles scale free behavior with exponent values that are in general rather vague and strongly noise dependent.

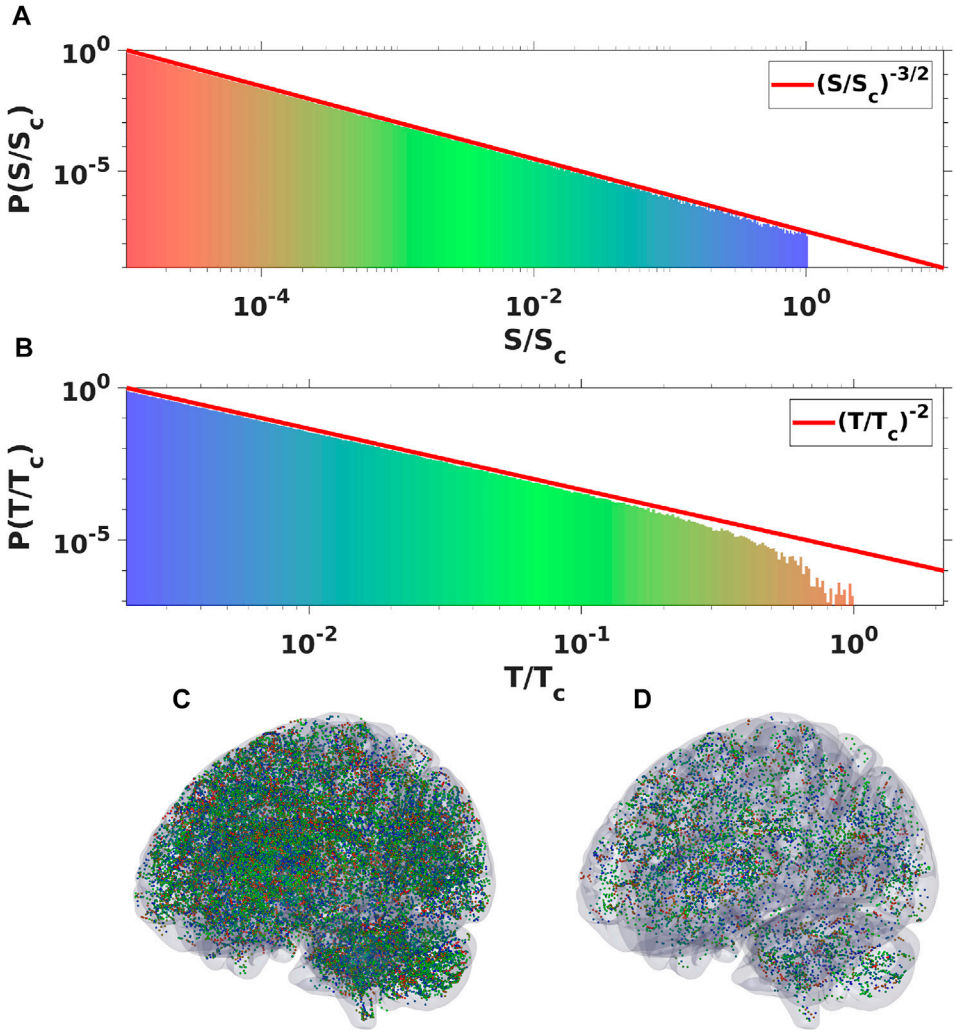


FIGURE 6
Plots of spatial **(A)** and temporal **(B)** probability density spectra obtained by binning oscillatory signal of ensemble of 10^6 WETCOW modes randomly distributed and propagated through cortical tissue. Two examples of temporal signal (dots or "spikes") snapshots with different values of signal threshold are shown in **(C)** and **(D)** (color pallet encodes the change of frequencies from the smallest (blue) to the largest (red).

Without this sufficiently strong noise those studies of course are not capable to show any scale free behavior. It is essential to realize that such models are thus highly dependent on the noise properties, and less so on the actual properties of the brain tissue itself as in the WETCOW theory, which is the critical link to practical applications of any brain activity theory. By contrast, no externally induced stochasticity in the form of additional noise term is required for our analysis.

Another important point is that for deriving scale free exponents in our approach we do not require to know the details of the coupling between nodes, essentially viewing all nodes as completely non-interacting. Presence of interactions in the form of (9) will not modify our analysis, and will not require any of the common *ad hoc* assumptions of identical global coupling between nodes [38]. When coupling between some of the nodes in (9) is sufficiently strong and these nodes are completely synchronized, we can always replace this subset of completely synchronized nodes by a single node and continue again with the same presented in this paper "coupling-independent" node analysis.

8 Effects of criticality on spike length

The assumption of the fixed spike duration δt_s used in Eq. 30 and 32 (or the spike length for spatial spiking in Eq. 43 and Eq. 44) can be improved by estimating the scaling of the spike width as a function of the criticality parameter from the amplitude equation (we will use the temporal form of the equation Eq. 21 but the spatial analysis with equation Eq. 38 is exactly the same).

Dividing equation Eq. 21 by $\tilde{A}$ and taking an integral around some area in the vicinity of the amplitude peak $\tilde{A}_{max}$ we can write

$$\int_{\tilde{A}_-}^{\tilde{A}_+} \frac{1}{\tilde{A}} d\tilde{A} = \int_{\tau_-}^{\tau_+} d\tau \int_{\Phi_-}^{\Phi_+} \frac{\tilde{\omega}_c}{R_\phi} \frac{R_a \cos(\phi - \Phi) - \alpha}{\tilde{\omega} + \tilde{\omega}_c \cos\phi} d\phi, \tag{47}$$

where $\tau_\pm = \tau_{max} \pm \delta\tau$, and τ_{max} is the location of spiking peak. Neglecting the spike shape asymmetries, i.e., assuming that $\tau_\pm$ correspond to symmetric changes in both the amplitudes

$\tilde{A}_{\pm} = \tilde{A}(\tau_{\pm}) = \tilde{A}_{max} - \delta\tilde{A}$, and the phases $\Phi_{\pm} = \Phi(\tau_{\pm}) = \Phi \pm \delta\Phi$, we can then estimate the spike duration $\delta t_s \equiv (\tau_+ - \tau_-)/\gamma$ as

$$\delta t_s = \frac{1}{\gamma}\int_{\Phi-\delta\Phi}^{\Phi+\delta\Phi} \frac{1 - R\left(\cos(\Phi) + \cos\left(\phi - \Phi\right)\right)}{\tilde{\omega} + \tilde{\omega}_c \cos\phi} d\phi, \tag{48}$$

where, similar to the spiking period estimation in Eq. 28, we again assume that the main contribution comes from the change of the oscillation phase, hence $\tilde{A}_c$ can be substituted for $\tilde{A}$. For $\delta\Phi$ some fixed value that is smaller or around a quarter of the period (i.e., $\delta\Phi \lesssim \pi/2$) can be chosen, and $R = \tilde{\omega}_c R_a/R_\phi$.

An expression (48) can be evaluated in closed form but we do not include it here and instead plotted the final spatial probability density spectra $P(S/S_c)$, similarly obtained from the expression for $\delta l_s/L_s$ again substituting $L = \sqrt{S}$ and $dL = dS/(2\sqrt{S})$, for several values of the S_c parameter (Figure 3), as well as for several values of the phase shift Φ (Figure 4). The spectra clearly show again the same power law dependence with -3/2 exponent as was reported in [12, 13] followed by a steep falloff sufficiently close to the critical point. What is interesting, however, is that the spectra for $\Phi = \pi/2$ (and this is the phase shift value used for spiking solutions reported in [1, 2, 39]) recover even the fine structure of the scaling and clearly show the small bump at the end of the scale free part of the spectra where the local probability deflects from the initial -3/2 power exponent and flattens first before turning in to the steep falloff. These small bumps are evident in all experimental spectra [12, 13] shown in Figure 3 as well.

9 Conclusion

Brain activity in general and neuronal avalanches in particular show an abundance of very complex and strangely organized activity patterns. Understanding the nature and the origin of cascades of synchronized activity in the cortex has multiple implications to understanding of organization of cortical functioning. Although originally neuronal avalanches were detected *in vitro* using multi-electrode arrays in 2D slices of cultivated cortex cultures [12, 13], there are now multiple experimental data of *in vivo* avalanche recordings [45–47] involving optical recordings as well [48].

One of the properties of the WETCOW wave modes is that the anisotropy structure of brain conductivity as well as the structure of brain inhomogeneity favors their propagation in the outer regions of the cortex (see, for example, Figure 2 of [1, 2]). Neuronal avalanches are measured in the most external layer of the cortex and, usually, introducing the electrodes deeper in the cortical columns will eliminate the scale-free distributions. Therefore, it seems to be an interesting problem to check the whole-brain scale free distribution in the region of typical propagation of WETCOW wave modes. To do this numerical experiment we generated an ensemble of 10^6 WETCOW modes distributed and randomly propagating through inhomogeneous and anisotropic cortical tissue. Figure 5 shows two randomly selected snapshots of wave mode trajectories that were generated using the procedure described in details in [1] and propagate in the surface-like 2D manner in the external layer of the cortex. Using the same procedure, that replicates the original experimental neuronal avalanche detection method, that is thresholding and then binning the wave signal into dots or "spikes", we again see that the WETCOW modes show scale free behavior as shown in Figure 6.

In summary, in this paper we have presented an analysis of temporal and spatial probability density spectra that are generated due to the critical dynamics of the non-linear weakly evanescent cortical wave (WETCOW) modes [1, 2]. The Hamiltonian framework developed for these WETCOW modes in [39] is advantageous in that it explicitly uncovers the reciprocity of the temporal and the spatial dynamics of the evolutionary equations. Therefore, in the non-linear regime sufficiently close to the critical point the spatial behavior of the wave modes displays features similar to the properties of their non-linear temporal dynamics that can be described as spatial domain spiking, with localized regions of wave activity separated by quiescent areas, with this spatial spiking intermittence increasing near the critical point. Similar spatial behavior was observed experimentally in neuronal avalanches, when activity detected at one electrode was typically skipped over the nearest neighbors. This was interpreted as evidence that avalanche spatial intermittency is not wave-like in nature [12, 13]. Our results demonstrate the contrary, however: the spatial patterns are the direct result of non-linear interactions of weakly evanescent cortical waves.

Both temporal and spatial scaling expressions analytically estimated from the non-linear amplitude/phase evolutionary equations show excellent agreement with the experimental neuronal avalanche probability spectra reproducing not only the general average power law exponent values and falloffs in the vicinity of the critical point, but also finding some very subtle but nevertheless clearly experimentally evident fine details, like bumps in the transition region at the edge of the scale free part of the probability spectra.

In a more general way these results may be applicable not only to neuronal avalanches but to many other physical systems that involve wave processes as they show that a system of wave modes interacting through all possible combinations of the third order non-linear terms described by a general wave Hamiltonian necessarily produces anharmonic wave modes with temporal and spatial scaling properties that follow scale free power laws.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding author.

Author contributions

VG and LF developed the theoretical formalism, performed the analytic calculations and performed the numerical simulations. Both VG and LF contributed to the final version of the manuscript.

Funding

LF and VG were supported by NIH grant R01 AG054049.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

References

1. Galinsky VL, Frank LR. Universal theory of brain waves: From linear loops to nonlinear synchronized spiking and collective brain rhythms. *Phys Rev Res* (2020) 2: 023061. (1–23). doi:10.1103/physrevresearch.2.023061

2. Galinsky VL, Frank LR. Brain waves: Emergence of localized, persistent, weakly evanescent cortical loops. *J Cogn Neurosci* (2020) 32:2178–202. doi:10.1162/jocn_a_01611

3. Buzsaki G. *Rhythms of the brain*. Oxford: Oxford University Press (2006).

4. Hodgkin AL, Huxley AF. A quantitative description of membrane current and its application to conduction and excitation in nerve. *J Physiol (Lond.)* (1952) 117:500–44. doi:10.1113/jphysiol.1952.sp004764

5. Fitzhugh R. Impulses and physiological states in theoretical models of nerve membrane. *Biophys J* (1961) 1:445–66. doi:10.1016/s0006-3495(61)86902-6

6. Nagumo J, Arimoto S, Yoshizawa S. An active pulse transmission line simulating nerve axon. *Proc IRE* (1962) 50:2061–70. doi:10.1109/jrproc.1962.288235

7. Morris C, Lecar H. Voltage oscillations in the barnacle giant muscle fiber. *Biophys J* (1981) 35:193–213. doi:10.1016/s0006-3495(81)84782-0

8. Izhikevich EM. Simple model of spiking neurons. *IEEE Trans Neural Netw* (2003) 14:1569–72. doi:10.1109/tnn.2003.820440

9. Gerstner W, Kistler WM, Naud R, Paninski L. *Neuronal dynamics: From single neurons to networks and models of cognition*. New York, NY, USA: Cambridge University Press (2014).

10. Kulkarni A, Ranft J, Hakim V. Synchronization, stochasticity, and phase waves in neuronal networks with spatially-structured connectivity. *Front Comput Neurosci* (2020) 14:569644. doi:10.3389/fncom.2020.569644

11. Kim R, Sejnowski TJ. Strong inhibitory signaling underlies stable temporal dynamics and working memory in spiking neural networks. *Nat Neurosci* (2021) 24: 129–39. doi:10.1038/s41593-020-00753-w

12. Beggs JM, Plenz D. Neuronal avalanches in neocortical circuits. *J Neurosci* (2003) 23:11167–77. doi:10.1523/jneurosci.23-35-11167.2003

13. Beggs JM, Plenz D. Neuronal avalanches are diverse and precise activity patterns that are stable for many hours in cortical slice cultures. *J Neurosci* (2004) 24:5216–29. doi:10.1523/jneurosci.0540-04.2004

14. Plenz D, Ribeiro TL, Miller SR, Kells PA, Vakili A, Capek EL. Self-organized criticality in the brain. *Front Phys* (2021) 9:639389. doi:10.3389/fphy.2021.639389

15. Friedman N, Ito S, Brinkman BA, Shimono M, DeVille RE, Dahmen KA, et al. Universal critical dynamics in high resolution neuronal avalanche data. *Phys Rev Lett* (2012) 108:208102. doi:10.1103/physrevlett.108.208102

16. Chialvo DR. Emergent complex neural dynamics. *Nat Phys* (2010) 6:744–50. doi:10.1038/nphys1803

17. Beggs JM, Timme N. Being critical of criticality in the brain. *Front Physiol* (2012) 3:163. doi:10.3389/fphys.2012.00163

18. Priesemann V, Wibral M, Valderrama M, Pröpper R, Le Van Quyen M, Geisel T, et al. Spike avalanches *in vivo* suggest a driven, slightly subcritical brain state. *Front Syst Neurosci* (2014) 8:108. doi:10.3389/fnsys.2014.00108

19. Cramer B, Stöckel D, Kreft M, Wibral M, Schemmel J, Meier K, et al. Control of criticality and computation in spiking neuromorphic networks with plasticity. *Nat Commun* (2020) 11:2853. doi:10.1038/s41467-020-16548-3

20. Fontenele AJ, de Vasconcelos NAP, Feliciano T, Aguiar LAA, Soares-Cunha C, Coimbra B, et al. Criticality between cortical states. *Phys Rev Lett* (2019) 122:208101. doi:10.1103/physrevlett.122.208101

21. Allegrini P, Paradisi P, Menicucci D, Laurino M, Piarulli A, Gemignani A. Self-organized dynamical complexity in human wakefulness and sleep: Different critical brain-activity feedback for conscious and unconscious states. *Phys Rev E Stat Nonlin Soft Matter Phys* (2015) 92:032808. doi:10.1103/physreve.92.032808

22. Allegrini P, Menicucci D, Bedini R, Fronzoni L, Gemignani A, Grigolini P, et al. Spontaneous brain activity as a source of ideal1/fnoise. *Phys Rev E Stat Nonlin Soft Matter Phys* (2009) 80:061914. doi:10.1103/physreve.80.061914

23. Teixeira Borges AF, Irrmischer M, Brockmeier T, Smit DJA, Mansvelder HD, Linkenkaer-Hansen K. Scaling behaviour in music and cortical dynamics interplay to mediate music listening pleasure. *Sci Rep* (2019) 9:17700. doi:10.1038/s41598-019-54060-x

24. Fosque LJ, Williams-García RV, Beggs JM, Ortiz G. Evidence for quasicritical brain dynamics. *Phys Rev Lett* (2021) 126:098101. doi:10.1103/physrevlett.126.098101

25. Bak P, Tang C, Wiesenfeld K. Self-organized criticality: An explanation of the 1/f noise. *Phys Rev Lett* (1987) 59:381–4. doi:10.1103/PhysRevLett.59.381

26. Bak P, Tang C, Wiesenfeld K. Self-organized criticality. *Phys Rev A* (1988) 38: 364–74. doi:10.1103/PhysRevA.38.364

27. Zapperi S, Lauritsen KB, Stanley HE. Self-organized branching processes: Mean-field theory for avalanches. *Phys Rev Lett* (1995) 75:4071–4. doi:10.1103/PhysRevLett.75.4071

28. Bækgaard Lauritsen K, Zapperi S, Stanley HE. Self-organized branching processes: Avalanche models with dissipation. *Phys Rev E* (1996) 54:2483–8. doi:10.1103/PhysRevE.54.2483

29. Eurich CW, Herrmann JM, Ernst UA. Finite-size effects of avalanche dynamics. *Phys Rev E Stat Nonlin Soft Matter Phys* (2002) 66:066137. doi:10.1103/physreve.66.066137

30. Bédard C, Kröger H, Destexhe A. Does the 1/*f* frequency scaling of brain signals reflect self-organized critical states? *Phys Rev Lett* (2006) 97:118102. doi:10.1103/PhysRevLett.97.118102

31. Touboul J, Destexhe A. Can power-law scaling and neuronal avalanches arise from stochastic dynamics? *PLoS One* (2010) 5:e8982. doi:10.1371/journal.pone.0008982

32. Touboul J, Destexhe A. Power-law statistics and universal scaling in the absence of criticality. *Phys Rev E* (2017) 95:012413. doi:10.1103/physreve.95.012413

33. Robinson PA, Rennie CJ, Wright JJ. Propagation and stability of waves of electrical activity in the cerebral cortex. *Phys Rev E* (1997) 56:826–40. doi:10.1103/PhysRevE.56.826

34. Yang DP, Robinson PA. Critical dynamics of Hopf bifurcations in the corticothalamic system: Transitions from normal arousal states to epileptic seizures. *Phys Rev E* (2017) 95:042410. doi:10.1103/physreve.95.042410

35. Robinson PA, Rennie CJ, Rowe DL. Dynamics of large-scale brain activity in normal arousal states and epileptic seizures. *Phys Rev E Stat Nonlin Soft Matter Phys* (2002) 65:041924. doi:10.1103/physreve.65.041924

36. di Santo S, Villegas P, Burioni R, Muñoz MA. Landau–Ginzburg theory of cortex dynamics: Scale-free avalanches emerge at the edge of synchronization. *Proc Natl Acad Sci* (2018) 115:E1356–E1365. doi:10.1073/pnas.1712989115

37. Gireesh ED, Plenz D. Neuronal avalanches organize as nested theta- and beta/gamma-oscillations during development of cortical layer 2/3. *Proc Natl Acad Sci* (2008) 105:7576–81. doi:10.1073/pnas.0800537105

38. Buendía V, Villegas P, Burioni R, Muñoz MA. Hybrid-type synchronization transitions: Where incipient oscillations, scale-free avalanches, and bistability live together. *Phys Rev Res* (2021) 3:023224. doi:10.1103/PhysRevResearch.3.023224

39. Galinsky VL, Frank LR. Collective synchronous spiking in a brain network of coupled nonlinear oscillators. *Phys Rev Lett* (2021) 126:158102. doi:10.1103/PhysRevLett.126.158102

40. Ott E, Antonsen TM. Long time evolution of phase oscillator systems. *Chaos* (2009) 19:023117. doi:10.1063/1.3136851

41. Tyulkina IV, Goldobin DS, Klimenko LS, Pikovsky A. Dynamics of noisy oscillator populations beyond the ott-antonsen ansatz. *Phys Rev Lett* (2018) 120:264101. doi:10.1103/PhysRevLett.120.264101

42. Kuramoto Y. *Chemical oscillations, waves, and turbulence*. Heidelberg: Springer Berlin Heidelberg (2013). Dover Books on Chemistry Series (Dover Publications, Incorporated).

43. Daido H. Generic scaling at the onset of macroscopic mutual entrainment in limit-cycle oscillators with uniform all-to-all coupling. *Phys Rev Lett* (1994) 73:760–3. doi:10.1103/physrevlett.73.760

44. Crawford JD. Scaling and singularities in the entrainment of globally coupled oscillators. *Phys Rev Lett* (1995) 74:4341–4. doi:10.1103/physrevlett.74.4341

45. Petermann T, Thiagarajan TC, Lebedev MA, Nicolelis MA, Chialvo DR, Plenz D. Spontaneous cortical activity in awake monkeys composed of neuronal avalanches. *Proc Natl Acad Sci U S A* (2009) 106:15921–6. doi:10.1073/pnas.0904089106

46. Bellay T, Shew WL, Yu S, Falco-Walter JJ, Plenz D. Selective participation of single cortical neurons in neuronal avalanches. *Front Neural Circuits* (2020) 14:620052. doi:10.3389/fncir.2020.620052

47. Hahn G, Petermann T, Havenith MN, Yu S, Singer W, Plenz D, et al. Neuronal avalanches in spontaneous activity *in vivo*. *J Neurophysiol* (2010) 104:3312–22. doi:10.1152/jn.00953.2009

48. Frostig RD, Chen-Bee CH, Johnson BA, Jacobs NS. Imaging cajal's neuronal avalanche: How wide-field optical imaging of the point-spread advanced the understanding of neocortical structure-function relationship. *Neurophotonics* (2017) 4:031217. doi:10.1117/1.nph.4.3.031217

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 06 March 2023
DOI 10.3389/fphy.2023.1160419

OPEN ACCESS

EDITED BY
Hairong Lin,
Hunan University, China

REVIEWED BY
Yan Liang,
Hangzhou Dianzi University, China
Sajad Jafari,
Amirkabir University of Technology, Iran

*CORRESPONDENCE
Quan Xu,
xuquan@cczu.edu.cn

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 07 February 2023
ACCEPTED 21 February 2023
PUBLISHED 06 March 2023

CITATION
Fan W, Chen X, Wang Y, Chen B, Wu H and Xu Q (2023), Hidden firing patterns and memristor initial condition-offset boosting behavior in a memristive Hindmarsh-Rose neuron model. *Front. Phys.* 11:1160419. doi: 10.3389/fphy.2023.1160419

Hidden firing patterns and memristor initial condition-offset boosting behavior in a memristive Hindmarsh-Rose neuron model

Weiwei Fan, Xiongjian Chen, Yiteng Wang, Bei Chen, Huagan Wu and Quan Xu*

School of Microelectronics and Control Engineering, Changzhou University, Changzhou, China

Electromagnetic induction can effectively induce abundant firing patterns in neurons. In modeling a neuron model with the electromagnetic induction effect, an electromagnetic induction current is frequently added to the state equation of membrane potential. To more properly reflect the non-uniform distribution of the ions inside and outside the neuron membrane, an ideal flux-controlled memristor with sinusoidal memductance function and non-linearly modulated input is raised to depict an electromagnetic induction effect on a Hindmarsh–Rose neuron model, and thereby, a three-dimensional (3D) memristive Hindmarsh–Rose (mHR) neuron model is built in this paper. The proposed mHR neuron model possesses no equilibrium point since the involvement of the ideal flux-controlled memristor, which induces the generation of hidden dynamics. Numerical results declare that the mHR neuron model can generate abundant hidden dynamics, i.e., periodic spiking, chaotic spiking, period-doubling bifurcation route, tangent bifurcation, and chaos crisis. These hidden dynamics are much related to the memristor coupling strength and externally applied stimulus. Afterward, the memristor initial condition-offset boosting behavior is revealed. This can trigger the generation of infinite multiple coexisting firing patterns along the memristor variable coordinate. These coexisting firing patterns have identical attractor topology but different locations in the phase plane. Finally, an analog circuit is designed for implementing the mHR neuron model, and PSIM-based circuit simulation is executed. The circuit-simulated results perfectly verify the generation of hidden infinite multiple coexisting initial condition-offset boosting firing patterns in the proposed mHR neuron model.

KEYWORDS

firing pattern, hidden dynamics, electromagnetic induction, memristor, analog implementation, Hindmarsh–Rose neuron model

1 Introduction

The nervous system contains a huge number of biological neurons, which are the basic information handling and integrating units of a biological nervous system [1]. The dynamical properties of these biological neurons are crucial for determining the behaviors of the nervous systems [2, 3]. Thus, modeling of the biological neuron and exploring its dynamical behaviors are research hotspots and attract many researchers' attention. Up to date, numerous neuron models have been constructed to depict different

kinds of biological neurons, and they can roughly be divided into two categories, i.e., the continuous-time neuron model [4–10] and discrete-time map [11–13]. In the literature, some of the continuous-time neuron models were built based on the electrophysiological ion transport mechanism [4–7]. In addition, some continuous-time neuron models [8–10] and discrete-time maps [11, 13] were built on dynamical assumptions to reproduce electrical activities without regard to the neuron structure [14]. No matter which category they are, all these neuron models can effectively reproduce abundant firing patterns in response to the change of the electrophysiological environment. Recently, the electromagnetic induction effect has immersed great scientists' concern, which can greatly affect the neuron dynamics [15–17] and neural network behaviors [18, 19].

Actually, the media of a biological neuron can be magnetized during its polarizing and depolarizing processes [20]. On one hand, periodic firing can control the transition, pumping, and distribution of calcium, potassium, chloride, etc., ions in these processes. On the other hand, the distribution, pumping, and transition of these ions can induce fluctuation of the membrane potential. Meanwhile, the electromagnetic induction effect is induced when the ions pass through the neuron membrane. Thus, ion channel currents and the electromagnetic induction current simultaneously affect the membrane potential. In the literature, flux-controlled memristors were used in various neuron models to depict the dynamic relation between the membrane potential and magnetic flux [21]. In other words, the flux-controlled memristors were used in the Hodgkin–Huxley neuron model [22–24], Izhikevich neuron model [25], FitzHugh–Nagumo neuron model [26], and three-dimensional (3D) [27–31]/two dimensional (2D) [32–34] Hindmarsh–Rose neuron model to depict the electromagnetic induction effect. These memristive neuron models can generate abundant firing patterns since the involvement of the flux-controlled memristor. To explain in detail, these flux-controlled memristors are non-ideal with a quadratic polynomial memductance function [22–30, 32, 33] and ideal with hypertangent/sinusoidal/cosinoidal memductance functions [20, 31, 33]. It is worth noting that the state equations of these memristors are linearly controlled by a membrane potential. Actually, the membrane potential possesses non-linear regulation on magnetization since the non-uniform distribution of the ions inside and outside the neuron membrane. To stress this issue, an ideal flux-controlled memristor with a sinusoidal memductance function and non-linear modulation on a memristor magnetic flux is raised to availably depict the electromagnetic induction effect in this paper.

The Hindmarsh–Rose neuron model is a simple kind of neuron model built on dynamical assumptions, which can reproduce main firing patterns of the biological neuron [9]. In the literature, the memristive Hindmarsh–Rose (mHR) neuron models with ideal memristors can easily generate multistability with coexisting firing patterns. In this case, the mHR neuron model has no equilibrium point, which induces the occurrence of hidden dynamics [20]. In particular, the initial condition-offset boosting behavior is triggered since the involved ideal flux-controlled memristor possessing sinusoidal/cosinoidal memductance functions [20, 34, 35], which is very different from the parameter-offset boosting behavior [36–38]. This induces the occurrence of extreme multistability with infinite multiple coexisting firing patterns. These coexisting firing patterns own attractors having identical topology and boosting along the memristor variable coordinate [20, 34]. Herein, a mHR neuron model with our proposed memristor is tamed for simplicity but without losing generality. The hidden dynamics and initial condition-offset boosting behavior of the mHR neuron model are investigated by numerical simulation and PSIM-based circuit simulation in this paper. A brief comparison between some aforementioned mHR neuron models and the model reported in this paper is demonstrated in Table 1. It is demonstrated that the memristor employed in building the mHR neuron model in this paper is different from the memristors reported in the aforementioned literature works. The electromagnetic induction effect characterized by the memristor is established by considering the periodic magnetization processing and non-uniform distribution of the ions inside and outside the neuron membrane.

The remainder of this paper is formulated as follows: Section 2 explains the building of a mHR neuron model with hidden dynamics. Section 3 explains memristor parameter- and stimulus parameter-related dynamical distributions and bifurcation behaviors by numerical simulations. Section 4 explains the memristor initial condition-offset boosting behavior and infinite multiple coexisting firing patterns. Section 5 explains the analog circuit design and PSIM-based circuit simulation. Finally, Section 6 briefly concludes the main results of this paper.

2 Memristive Hindmarsh–Rose neuron model

Considering the periodic magnetization process and non-linear regulation of electromagnetic induction, a memristor with sinusoidal memductance function and non-linearly modulated input is raised, which is mathematically expressed as follows:

$$\begin{aligned} i_M &= W(\varphi)v_M = \sin(\varphi)v_M, \\ \dot{\varphi} &= \tanh(v_M), \end{aligned} \tag{1}$$

where v_M and i_M represent the terminal voltage and current, respectively. $W(\varphi) = \sin(\varphi)$ is the periodic memductance function, and magnetic flux φ is the memristor inner state variable. Different from the memristor reported in [20], this memristor has a non-linearly modulated input, i.e., a hypertangent function, to reflect the non-uniform distribution of the ions inside and outside the neuron membrane. The hypertangent function is continuously derivable and bounded above and below.

To investigate this kind of electromagnetic induction effect on a neuron, the memristor is introduced into the existing 2D Hindmarsh–Rose neuron model [9]; thereby, a 3D mHR neuron model is built as follows:

$$\begin{cases} \dot{x} = y - ax^3 + bx^2 + I + k\sin(\varphi)x, \\ \dot{y} = c - dx^2 - y, \\ \dot{\varphi} = \tanh(x), \end{cases} \tag{2}$$

where x is the membrane potential and y is the recovery variable. a, b, c, and d are four controllable parameters in the original model [39]. I is the externally applied stimulus, and k is the coupling strength of electromagnetic induction. We mainly consider the

TABLE 1 Comparison of relevant papers on the mHR neuron models.

Paper	Dimensionality	Memristor state equation	Implementation
[31]	3D	$\begin{cases} i_M = \cos(\varphi)v_M, \\ \dot{\varphi} = v_M. \end{cases}$	DSP
[32]	3D	$\begin{cases} i_M = k_2(a + b\varphi + c\varphi^2)v_M, \\ \dot{\varphi} = \tanh(v_M). \end{cases}$	PSpice
[34]	4D	$\begin{cases} i_M = -\tanh(\varphi)v_M, \\ \dot{\varphi} = -v_M. \end{cases}$	—
This paper	3D	$\begin{cases} i_M = \sin(\varphi)v_M, \\ \dot{\varphi} = \tanh(v_M). \end{cases}$	PSIM

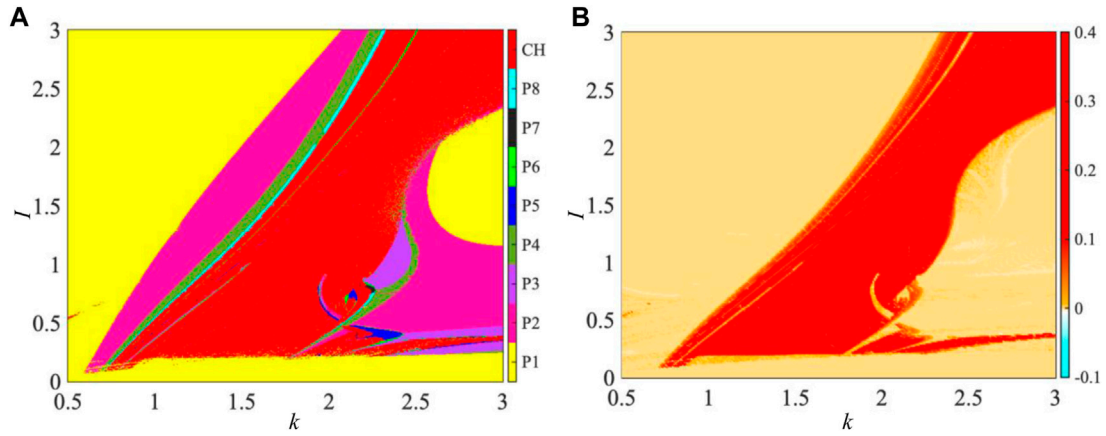


FIGURE 1
Parameter-related dynamical distribution in the k–I parameter plane for $a = 1$, $b = 3$, $c = 1$, and $d = 5$ under the initial conditions $[x(0), y(0), \varphi(0)] = [0, 0, 0]$. **(A)** 2D bifurcation diagram depicted by inspecting periodicities of the membrane potential x and **(B)** 2D dynamical map described by LLE.

dynamical effect of the externally applied stimulus I and coupling strength k on the mHR neuron model in the following sections. Therefore, the four controllable parameters, namely, $a = 1$, $b = 3$, $c = 1$, and $d = 5$, are assigned as the original parameters in [39, 40]; I and k are adjustable parameters with positive values, and their typical values are preset to $I = 1.5$ and $k = 2$.

By setting the left sides of (2) equal to 0, one can obtain the following equation:

$$\begin{cases} 0 = y - x^3 + 3x^2 + I + k\sin(\varphi)x, \\ 0 = 1 - 5x^2 - y, \\ 0 = \tanh(x). \end{cases} \tag{3}$$

Evidently, there is no solution of (3) since I has a positive value. In other words, model (2) has no equilibrium point. Therefore, the dynamical behaviors and firing patterns generated by the 3D mHR neuron model (2) are hidden [41].

3 Parameter-related hidden dynamics

In this section, we mainly focus our concern on the parameter-related hidden dynamics with the two adjustable parameters of the coupling strength k and externally applied stimulus I. The initial conditions $[x(0), y(0), \varphi(0)] = [0, 0, 0]$ are utilized. The MATLAB-based ODE45 algorithm with a fixed time-step duration of 10^{-2} s and time-end duration of 800 s is utilized to calculate the bifurcation diagram, and the Jacobi matrix-based Wolf's method with a time-step duration of 0.1 s and time-end duration of 4,000 s is employed to calculate Lyapunov exponent spectra [42].

3.1 Dynamical distribution

When the two adjustable parameters k and I are varied in $0.5 \leq k \leq 3$ and $0 \leq I \leq 3$, dynamical distributions of the bifurcation diagram and dynamical map in the k–I parameter plane are simulated, as shown in Figure 1. The 2D bifurcation diagram is depicted by checking the periodicities of the membrane potential x, as shown in Figure 1A, that is, the trajectories with different periodicities are painted by different colors. The red marked by CH represents chaos, and the other colors represent period-1 to period-8 marked by P1 to P8, respectively. One can see that the 2D bifurcation diagram possesses a ribbon structure in some regions and the ribbons marked by P1, P2, P4, and P8 appear in sequence.

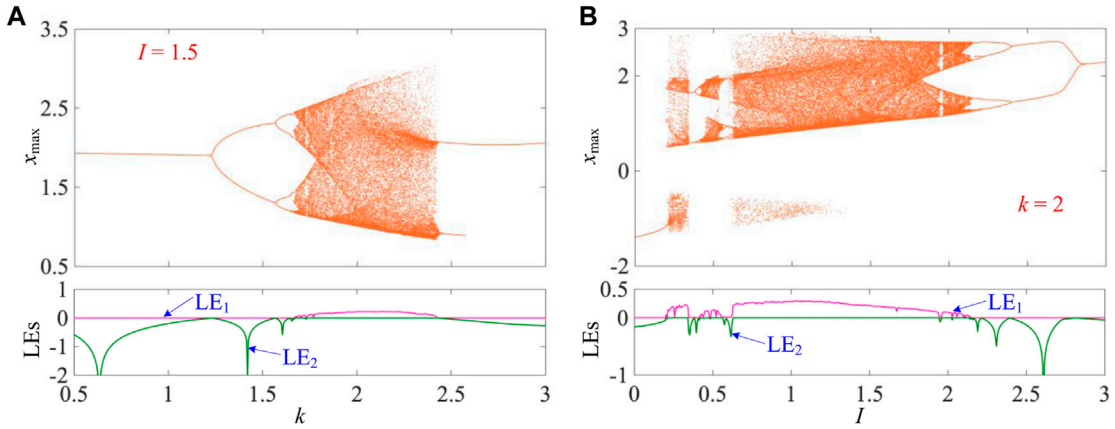


FIGURE 2
Parameter-related bifurcation behaviors as k and I changed in determined ranges for $a = 1$, $b = 3$, $c = 1$, and $d = 5$ under the initial conditions $[x(0), y(0), \varphi(0)] = [0, 0, 0]$. **(A)** k-related bifurcation diagram (top) and LEs (bottom) for $I = 1.5$ and **(B)** I-related bifurcation diagram (top) and LEs (bottom) for $k = 2$.

Also, numerous ribbons marked by P3, P5, and P7 are embedded in the CH (red) region or near the neighborhood of the CH region. These declare that numerous periodic windows generated *via* tangent bifurcations [42] and period-doubling bifurcations [29] are triggered by varying the two parameters. In addition, the 2D dynamical map described by the largest Lyapunov exponent (LLE) is employed to depict the parameter-related dynamical distribution in the k–I parameter plane, as shown in Figure 1B. The colorized domains are painted with different colors according to the values of LLE: red for chaos with positive LLE and other colors for a period with negative LLE.

It is demonstrated that the dynamical behaviors depicted by dynamical distributions of the 2D bifurcation diagram and 2D dynamical map are completely identical. These numerical results demonstrate that the coupling strength and externally applied stimulus can induce abundant dynamical behaviors on neuron properties of the 3D mHR neuron model.

3.2 Bifurcation behavior

To more clearly demonstrate the bifurcation behaviors with the coupling strength k and externally applied stimulus I, the one-dimensional (1D) bifurcation diagram and Lyapunov exponent spectra (LEs) are numerically simulated with the variations of k and I, as shown in Figures 2A, B, respectively. The representations at the top of Figures 2A, B display the 1D bifurcation diagrams of the membrane potential x, while the representations at the bottom exhibit LEs.

Herein, the externally applied stimulus $I = 1.5$ is fixed, and the coupling strength k is varied in $0.5 \leq k \leq 3$. The 1D bifurcation diagram for the maximum value of the membrane potential x (marked as x_{max}) is depicted in Figure 2A. When k increases from 0.5, the trajectory of mHR neuron model (2) starts from period-1, then enters chaos *via* the forward period-doubling bifurcation route [29], returns to period-2 *via* tangent bifurcation [42], and ends up to period-1. It is worth noting that the period-doubling bifurcation route demonstrates the transition of P1-P2-P4-P8-CH. In Figure 2A, only the first Lyapunov exponent LE_1 and partial second Lyapunov exponent LE_2 are shown for better visualization since the third exponent is very small. The LE_1 exponent has a zero value for periodic states with different periodicities and a positive value for the chaotic state. The LE_2 exponent increases to zero and immediately returns to the negative value along with the occurrence of period-doubling bifurcations. It is observed that the bifurcation behaviors revealed by the 1D bifurcation diagram (up) are effectively verified by the LEs (bottom) in Figure 2A.

Then, we fix coupling strength $k = 2$ and change the externally applied stimulus I in $0 \leq I \leq 3$. In Figure 2B, one can see that with the increase of I, the mHR neuron model undergoes period-1 to chaos *via* chaos crisis [43], to period-2 *via* tangent bifurcation, to chaos *via* forward period-doubling bifurcations, to period-3 *via* reverse period-doubling bifurcations, to chaos *via* chaos crisis, and finally to period-1 *via* reverse period-doubling bifurcations. It is worth noting that there exists a small periodic window near the neighborhood of $I = 1.95$. The LE_1 exponent drops to zero and maintains in the narrow parameter range and then returns to a positive value with the appearance of a periodic window. Obviously, the evolution of LEs confirms the occurrence of bifurcation behaviors.

Consequently, these bifurcation behaviors can lead to the occurrence of abundant periodic firing patterns with different periodicities and a chaotic firing pattern in the 3D mHR neuron model and can be regulated by the memristor coupling strength and externally applied stimulus.

3.3 Hidden firing patterns

In this section, five values of coupling strength k with $I = 1.5$ are selected from Figure 2A to partially display the firing patterns emerged from the 3D mHR neuron model, as shown in Figure 3.

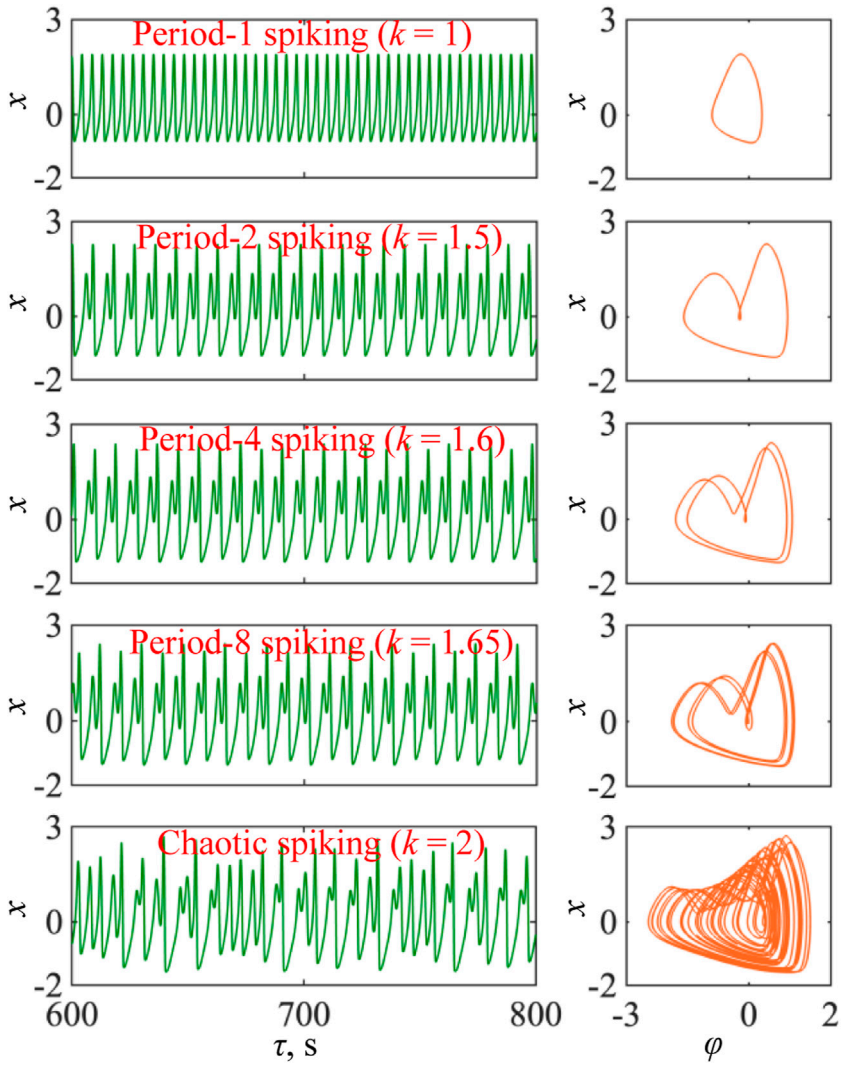


FIGURE 3
Time–domain waveforms (left) and phase portraits in the φ–x phase plane (right) with $a = 1$, $b = 3$, $c = 1$, $d = 5$, and $I = 1.5$ and the initial conditions $[x(0), y(0), \varphi(0)] = [0, 0, 0]$ for $k = 1$ (period-1 spiking), $k = 1.5$ (period-2 spiking), $k = 1.6$ (period-4 spiking), $k = 1.65$ (period-8 spiking), and $k = 2$ (chaotic spiking), respectively.

Time–domain waveforms of the membrane potential x (left) and corresponding phase portraits in the φ–x phase plane (right) are demonstrated; they are period-1 spiking, period-2 spiking, period-4 spiking, period-8 spiking, and chaotic spiking for $k = 1$, 1.5, 1.6, 1.65, and 2, respectively. These numerical results further demonstrate that the 3D mHR neuron model can generate hidden firing patterns of periodic spiking behaviors with different periodicities and a chaotic spiking behavior. Moreover, the state transition (P1-P2-P4-P8-CH) of firing patterns confirms the generation of the period-doubling bifurcation route.

4 Initial condition-related dynamics

Of particular interest, the 3D mHR neuron model (2) can show the initial condition-offset boosting behavior since the involvement of sinusoidal memductance function [20, 44]. This can trigger infinite multiple coexisting firing patterns for a fixed set of model parameters. In this section, we mainly focus our attention on this issue. Herein, we only consider the memristor initial condition-induced dynamical effect and set the initial conditions as $[x(0), y(0), \varphi(0)] = [0, 0, \varphi(0)]$. The numerical simulation settings are identical with those employed in Section 3.

4.1 Initial condition-offset boosting

Herein, two sets of model parameters, i.e., $I = 1.5$, $k = 1.5$ and $I = 1.5$, $k = 2$, are selected as paradigms to demonstrate the memristor initial condition-offset boosting behavior. The memristor initial condition is adjusted in the region [−20, 20]. The bifurcation plots of the 1D bifurcation diagram and Lyapunov exponent spectra for the two sets of model parameters are shown in Figures 4A, B, respectively.

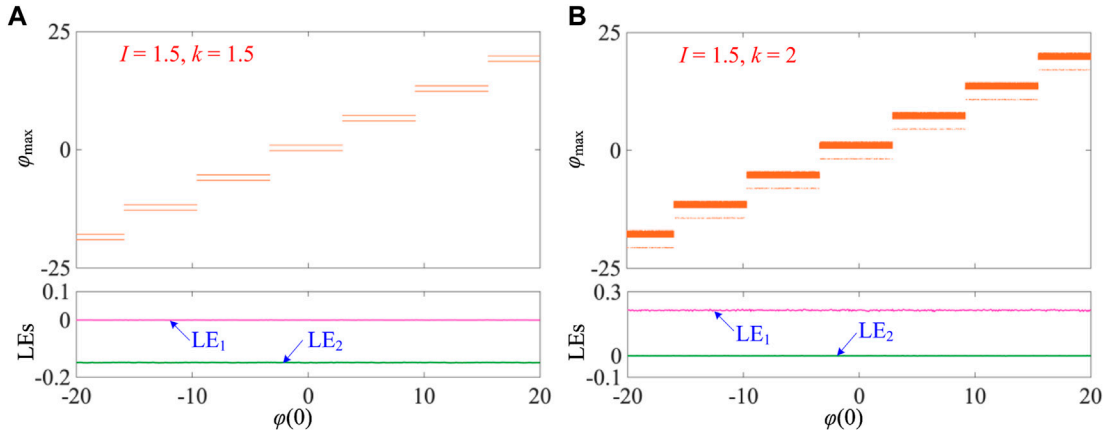


FIGURE 4
Memristor initial condition-offset boosting behaviors illustrated by 1D bifurcation plots of the 1D bifurcation diagram and LEs for the initial conditions [$x(0)$, $y(0)$, $\varphi(0)$] = [0, 0, $\varphi(0)$]. **(A)** I = 1.5 and k = 1.5 and **(B)** I = 1.5 and k = 2.

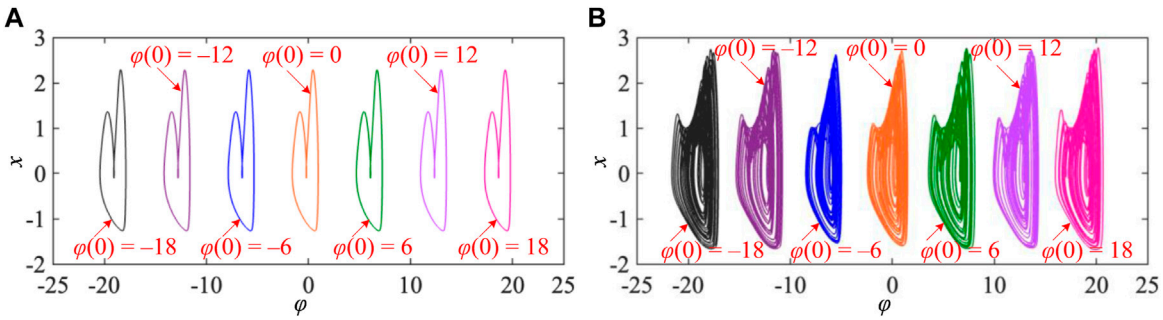


FIGURE 5
Infinite multiple coexisting firing patterns for different memristor initial conditions of $\varphi(0)$ = −18 V, −12 V, −6 V, 0 V, 6 V, 12 V, and 18 V under different sets of model parameters. **(A)** Coexisting period-2 firing patterns for I = 1.5 and k = 1.5 and **(B)** coexisting chaotic firing patterns for I = 1.5 and k = 2.

Figure 4A shows the bifurcation plots for I = 1.5 and k = 1.5, which demonstrates period-2 spiking firing patterns with the increase of the initial condition $\varphi(0)$. One can see that the locations of these firing patterns possess a step-by-step structure and their dynamic amplitude is identical. In addition, the memristor initial condition applies at 2π initial condition-offset, which is the period of the sinusoidal memductance function $\sin(\varphi)$. In this case, the step change happens periodically. In addition, the LE_1 and LE_2 exponents demonstrate constant Lyapunov exponent spectra and are not related to the memristor initial condition, i.e., LE_1 = 0 and LE_2 = −0.15.

As shown in Figure 4B, the bifurcation plots for I = 1.5 and k = 2 are elaborated. The results also display the step-by-step memristor initial condition-offset boosting behavior; thereby, multiple coexisting chaotic firing patterns are generated. Also, the memristor initial condition applies at 2π initial condition-offset, and these steps periodically occur. Identically, the Lyapunov exponent spectra display constant values of LE_1 and LE_2 and possess LE_1 = 0.21 and LE_2 = 0 for the chaotic firing patterns. It is worth noting that the initial condition-offset boosting behavior exists for other firing patterns under different fixed model parameters. In addition, the memristor initial condition-offset boosting behavior can trigger infinite multiple coexisting firing patterns. We only demonstrate this behavior in a finite range of memristor initial conditions in this study.

4.2 Infinite multiple coexisting firing patterns

In the previous bifurcation analysis for the memristor initial condition-offset boosting behavior, the results display the occurrence of hidden infinite multiple coexisting firing patterns in the 3D mHR neuron model. To further demonstrate this striking memristor initial condition-offset boosting behavior, the coexisting firing patterns are displayed by the phase portrait in the φ–x phase plane, as shown in Figure 5. The coexisting period-2 spiking firing patterns for I = 1.5 and k = 1.5 under the initial conditions $\varphi(0)$ = −18, −12, −6, 0, 6, 12, and 18 are shown in Figure 5A. These initial conditions are selected in each step from the bifurcation diagram in Figure 4A. The offsets among each of the two adjacent attractors of

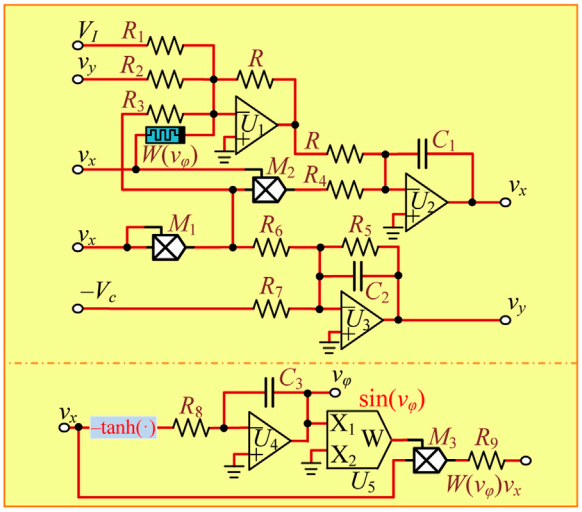


FIGURE 6
Circuit schematic representation of the 3D mHR neuron model.

firing patterns are all 2π, and these attractors are not connected, which implies the emergence of initial condition-offset boosted attractors. The attractors possess identical topology but different locations. Similarly, the memristor initial condition-offset boosted coexisting chaotic firing patterns for $I = 1.5$ and $k = 2$ under the initial conditions $\varphi(0) = -18, -12, -6, 0, 6, 12$, and 18 are also demonstrated in Figure 5B. It is worth noting that we do not select the initial conditions with the interval 2π for their convenient setting in PSIM-based circuit simulation. These results demonstrate the generation of hidden infinite multiple coexisting firing patterns in our proposed 3D mHR neuron model.

5 Analog design and PSIM-based circuit simulation

The analog circuit design of neuron models is crucial for investigating the neuron dynamics and exploring neuron-based engineering applications [45, 46]. The 3D mHR neuron model can be easily designed by utilizing passive circuit components of a capacitor and resistor and the integrated chips of operational amplifier, multiplier, and trigonometric circuit modules. The circuit schematic representation is well-designed and given in Figure 6. The memristor equivalent circuit contains a hypertangent circuit module [47], a sinusoidal function chip U_5, an operational amplifier U_4, a multiplier M_3, a capacitor, and two resistors, as shown in the bottom part of Figure 6. The main circuit involves two integrators, two multipliers, and an inverter, as shown in the top part of Figure 6. Then, the circuit state equations can be correspondingly built as follows:

$$\begin{cases} RC_1\dfrac{dv_x}{dt} = \dfrac{RV_I}{R_1} + \dfrac{Rv_y}{R_2} + \dfrac{g_1 R v_x{}^2}{R_3} - \dfrac{g_1 g_2 R v_x{}^3}{R_4} + \dfrac{g_3 R \sin(v_\varphi) v_x}{R_9}, \\ RC_2\dfrac{dv_y}{dt} = -\dfrac{Rv_y}{R_5} - \dfrac{g_1 R v_x{}^2}{R_6} + \dfrac{RV_c}{R_7}, \\ RC_3\dfrac{dv_\varphi}{dt} = \dfrac{R\tanh(v_x)}{R_8}, \end{cases} \tag{4}$$

where v_x, v_y, and v_φ are three circuit state variables corresponding to the model variables x, y, and φ, respectively. We suppose $C_1 = C_2 = C_3 = C$ and the integral time constant $RC = 1$ ms, i.e., $C = 100$ nF and $R = 10$ kΩ. In addition, the recovery variable v_y is linearly transformed to reduce its dynamic amplitude as follows:

$$(v_x, v_y, v_\varphi) = (v_x, 10v_y, v_\varphi). \tag{5}$$

The linear transformation is conducive to a hardware experiment since the value of the recovery variable approaches the saturation voltage of operational amplifiers without transformation. Thereby, the other circuit parameters are calculated as $R_1 = 10$ kΩ, $R_2 = 1$ kΩ, $R_3 = 3.3$ kΩ, $R_4 = 10$ kΩ, $R_5 = 10$ kΩ, $R_6 = 20$ kΩ, $R_7 = 100$ kΩ, $R_8 = 10$ kΩ, $R_9 = 3.3$ kΩ, $g_1 = g_2 = g_3 = 1$ V^{-1}, and $V_c = 1$ V by comparing (2) with (4). It is worth noting that the two adjustable parameters can be regulated by $I = V_I$ and $k = R/R_9$. The initial states of three capacitors are assigned as $[v_x(0), v_y(0), v_\varphi(0)] = [0\text{ V}, 0\text{ V}, v_\varphi(0)]$.

Employing the circuit schematic representation illustrated in Figure 6, a PSIM-based simulation circuit is built and circuit

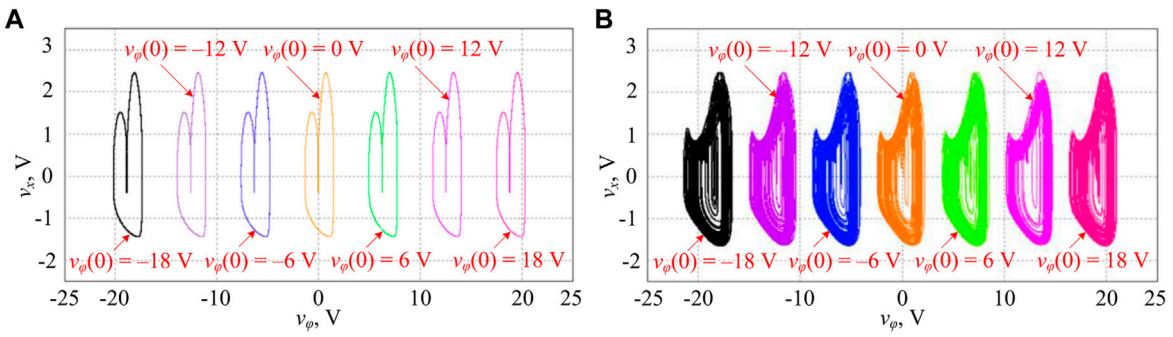


FIGURE 7
PSIM-based circuit simulation of infinite multiple coexisting firing patterns for different memristor initial conditions of $\varphi(0) = -18$ V, −12 V, −6 V, 0 V, 6 V, 12 V, and 18 V under the two sets of circuit parameters. **(A)** Coexisting period-2 firing patterns for $V_I = 1.5$ V ($I = 1.5$) and $R_9 = 6.667$ kΩ ($k = 1.5$) and **(B)** coexisting chaotic firing patterns for $V_I = 1.5$ V ($I = 1.5$) and $R_9 = 5$ kΩ ($k = 2$).

parameters are accurately set. First, the two adjustable circuit parameters are set to V_I = 1.5 V and R_9 = 6.667 kΩ corresponding to I = 1.5 and k = 1.5. By respectively setting the memristor initial condition $v_\varphi(0)$ to −18 V, −12 V, −6 V, 0 V, 6 V, 12 V, and 18 V, PSIM-based circuit simulations are executed and phase trajectories are obtained in the v_φ–v_x phase plane, as shown in Figure 7A. Then, the two adjustable circuit parameters are set to V_I = 1.5 V and R_9 = 5 kΩ corresponding to I = 1.5 and k = 2. The PSIM-based circuit simulations are illustrated in Figure 7B. One can see that the PSIM-based circuit simulations in Figure 7 are very consistent with the numerical results in Figure 5. These circuit-simulated results manifest the occurrence of hidden infinite multiple coexisting initial condition-offset boosting firing patterns in our proposed 3D mHR neuron model. It is worth noting that the power supplies for operational amplifiers and multipliers are, respectively, set to ±30 V and ±15 V in PSIM-based circuit simulation.

6 Conclusion

In this paper, an ideal flux-controlled memristor with sinusoidal memductance and non-linearly modulated input was presented to depict the electromagnetic induction effect in biological neurons. Then, the electromagnetic induction effect on an existing 2D Hindmarsh–Rose neuron model was elaborated. Theoretical analysis and numerical simulation demonstrated that the 3D mHR neuron model can generate the hidden memristor initial condition-offset boosting behavior with infinite multiple coexisting firing patterns. The attractors of these firing patterns have an offset along the memristor variable coordinate, and the offset is identical with the period of memductance function. In addition, PSIM-based circuit simulation further confirmed the validation of the analog circuit design and generation of the initial condition-offset boosting behavior. It is worth noting that the power supplies should be suitably set in PSIM-based circuit simulation to capture the offset boosting firing patterns. The settings can refer to the dynamic range for the attractors of these firing patterns along with the memristor variable coordinate. This hinders the hardware experimental measurement of initial condition-offset boosting firing patterns. In addition, it is not easy to accurately set the initial conditions in each step to acquire corresponding firing patterns from hardware experiments. However, the memristor initial condition-offset boosting firing patterns have potentiality in neuron-based engineering applications [37, 48], i.e., the waveform bias of chaotic signal and random signal generation [49]. These deserve our future concern.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material; further inquiries can be directed to the corresponding author.

Author contributions

All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

Funding

This work was supported by the National Natural Science Foundation of China under grant nos 12172066 and 61801054, the Natural Science Foundation of Jiangsu Province, China, under grants BK20160282 and BK20210850, the Project 333 of Jiangsu Province, the Postgraduate Research and Practice Innovation Program of Jiangsu Province, China, under the grant no. KYCX22_3054, and the College Students' Innovation and Entrepreneurship Training Program of Changzhou University.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Yao Z, Zhou P, Zhu ZG, Ma J. Phase synchronization between a light-dependent neuron and a thermosensitive neuron. *Neurocomputing* (2021) 423:518–34. doi:10.1016/j.neucom.2020.09.083

2. Zhu ZY, Wang RB, Zhu FY. The energy coding of a structural neural network based on the Hodgkin-Huxley model. *Front Neurosci* (2018) 12:122. doi:10.3389/fnins.2018.00122

3. Ji X, Hu XF, Zhou Y, Dong ZK, Duan SK. Adaptive sparse coding based on memristive neural network with applications. *Cogn Neurodyn* (2019) 13:475–88. doi:10.1007/s11571-019-09537-w

4. Hodgkin AL, Huxley AF. A quantitative description of membrane current and its application to conduction and excitation in nerve. *Bull Math Biol* (1990) 52:25–71. doi:10.1016/s0092-8240(05)80004-7

5. Chay TR. Chaos in a three-variable model of an excitable cell. *Physica D* (1985) 16:233–42. doi:10.1016/0167-2789(85)90060-0

6. Xu Q, Tan X, Zhu D, Bao H, Hu YH, Bao BC. Bifurcations to bursting and spiking in the Chay neuron and their validation in a digital circuit. *Chaos Solitons Fractals* (2020) 141:110353. doi:10.1016/j.chaos.2020.110353

7. Morris C, Lecar H. Voltage oscillations in the barnacle giant muscle fiber. *Biophys J* (1981) 35:193–213. doi:10.1016/s0006-3495(81)84782-0

8. FitzHugh R. Impulses and physiological states in theoretical models of nerve membrane. *Biophys J* (1961) 1:445–66. doi:10.1016/s0006-3495(61)86902-6

9. Hindmarsh JL, Rose RM. A model of the nerve impulse using two first-order differential equations. *Nature* (1982) 296:162–4. doi:10.1038/296162a0

10. Xu Q, Ju ZT, Ding ZT, Feng CT, Chen M, Bao BC. Electromagnetic induction effects on electrical activity within a memristive Wilson neuron model. *Cogn Neurodyn* (2022) 16:1221–31. doi:10.1007/s11571-021-09764-0

11. Elson RC, Selverston AI, Huerta R, Rulkov NF, Rabinovich MI, Abarbanel HDI. Synchronous behavior of two coupled biological neurons. *Phys Rev Lett* (1998) 81: 5692–5. doi:10.1103/physrevlett.81.5692

12. Xu Q, Liu T, Ding SK, Wu HG, Huang LP, Chen B. Extreme multistability and phase synchronization in a heterogeneous bi-neuron Rulkov network with memristive electromagnetic induction. *Cogn Neurodyn* (2022). doi:10.1007/s11571-022-09866-3

13. Vivekanandhan G, Natiq H, Merrikhi Y, Rajagopal K, Jafari S. Dynamical analysis and synchronization of a new memristive Chialvo neuron model. *Electronics* (2023) 12: 545. doi:10.3390/electronics12030545

14. Xu Q, Liu T, Feng CT, Bao H, Wu HG, Bao BC. Continuous non-autonomous memristive Rulkov model with extreme multistability. *Chin Phys B* (2021) 30:128702. doi:10.1088/1674-1056/ac2f30

15. Lv M, Ma J. Multiple modes of electrical activities in a new neuron model under electromagnetic radiation. *Neurocomputing* (2016) 205:375–81. doi:10.1016/j.neucom.2016.05.004

16. Lin HR, Wang CH, Deng QL, Xu C, Deng ZQ, Zhou C. Review on chaotic dynamics of memristive neuron and neural network. *Nonlinear Dyn* (2021) 106:959–73. doi:10.1007/s11071-021-06853-x

17. Shen H, Yu F, Wang CH, Sun JR, Cai S. Firing mechanism based on single memristive neuron and double memristive coupled neurons. *Nonlinear Dyn* (2022) 110: 3807–22. doi:10.1007/s11071-022-07812-w

18. Yu F., Shen H., Yu Q. L., Kong X. X., Sharma P. K., Cai S. (2022). Privacy Protection of Medical Data Based on Multi-scroll Memristive Hopfield Neural Network. *IEEE. Trans. Netw. Sci. Eng.* 10 845–858.

19. Yu F, Kong XX, Mokbel AAM, Yao W, Cai S. Complex dynamics, hardware implementation and image encryption application of multiscroll memeristive Hopfield neural network with a novel local active memeristor. *IEEE Trans Circuits Syst Express Briefs* (2023) 70:326–30. doi:10.1109/tcsii.2022.3218468

20. Bao H, Liu WB, Ma J, Wu HG. Memristor initial-offset boosting in memristive HR neuron model with hidden firing patterns. *Int J Bifurc Chaos* (2020) 30:2030029. doi:10.1142/s0218127420300293

21. Du L, Cao ZL, Lei YM, Deng ZC. Electrical activities of neural systems exposed to sinusoidal induced electric field with random phase. *Sci China Technol Sci* (2019) 62: 1141–50. doi:10.1007/s11431-017-9309-9

22. Wu FQ, Wang CN, Jin WY, Ma J. Dynamical responses in a new neuron model subjected to electromagnetic induction and phase noise. *Physica A* (2017) 469:81–8. doi:10.1016/j.physa.2016.11.056

23. Yuan ZX, Feng PH, Du MM, Wu Y. Dynamical response of a neuron-astrocyte coupling system under electromagnetic induction and external stimulation. *Chin Phys B* (2020) 29:030504.

24. Yuan ZX, Feng PH, Fang YC, Yu YY, Wu Y. Astrocytic modulation on neuronal electric mode selection induced by magnetic field effect. *Cogn Neurodyn* (2022) 16: 183–94. doi:10.1007/s11571-021-09709-7

25. Kafraj MS, Parastesh F, Jafari S. Firing patterns of an improved Izhikevich neuron model under the effect of electromagnetic induction and noise. *Chaos Solitons Fractals* (2020) 137:109782. doi:10.1016/j.chaos.2020.109782

26. Jia YB, Lu B, Gu HG. Excitatory electromagnetic induction current enhances coherence resonance of the FitzHugh-Nagumo neuron. *Int J Bifurc Chaos* (2019) 33: 1950242. doi:10.1142/s0217979219502424

27. Tang KM, Wang ZL, Shi XR. Electrical activity in a time-delay four variable neuron model under electromagnetic induction. *Front Comput Neurosci* (2017) 11:105. doi:10.3389/fncom.2017.00105

28. Lu LL, Jia Y, Xu Y, Ge MY, Yang LJ, Zhan X. Energy dependence on modes of electric activities of neuron driven by different external mixed signals under electromagnetic induction. *Sci China Technol Sci* (2019) 62:427–40. doi:10.1007/s11431-017-9217-x

29. An XL, Qiao S. The hidden, period-adding, mixed-mode oscillations and control in a HR neuron under electromagnetic induction. *Chaos Solitons Fractals* (2021) 143: 110587. doi:10.1016/j.chaos.2020.110587

30. Rajagopal K, Jafari S, Karthikeyan A, Srinivasan A. Effect of magnetic induction on the synchronizability of coupled neuron network. *Chaos* (2021) 31:083115. doi:10.1063/5.0061406

31. Xu L, Qi GY, Ma J. Modeling of memristor-based Hindmarsh-Rose neuron and its dynamical analyses using energy method. *Appl Mathemat Model* (2022) 101:503–16. doi:10.1016/j.apm.2021.09.003

32. Hu XY, Wang S, Liu CX. Hidden coexisting firing patterns and bubble-like bifurcation in HR neuron model under electromagnetic induction. *Chin J Phys* (2022) 77:2541–9. doi:10.1016/j.cjph.2022.04.016

33. Njitacke ZT, Doubla IS, Mabekou S, Kengne J. Hidden electrical activity of two neurons connected with an asymmetric electric coupling subject to electromagnetic induction: Coexistence of patterns and its analog implementation. *Chaos Solitons Fractals* (2020) 137:109785. doi:10.1016/j.chaos.2020.109785

34. Zhang S, Zheng JH, Wang XP, Zeng ZG. A novel no-equilibrium HR neuron model with hidden homogeneous extreme multistability. *Chaos Solitons Fractals* (2021) 145:110761. doi:10.1016/j.chaos.2021.110761

35. Wu HG, Ye Y, Bao BC, Chen M, Xu Q. Memristor initial boosting behaviors in a two-memristor-based hyperchaotic system. *Chaos, Solitons Fractals* (2019) 121:178–85. doi:10.1016/j.chaos.2019.03.005

36. Wu HG, Zhou J, Chen M, Xu Q, Bao BC. DC-offset induced asymmetry in memristive diode-bridge-based Shinriki oscillator. *Chaos Solitons Fractals* (2022) 154: 111624. doi:10.1016/j.chaos.2021.111624

37. Li CB, Wang X, Chen GR. Diagnosing multistability by offset boosting. *Nonlinear Dyn* (2017) 90:1335–41. doi:10.1007/s11071-017-3729-1

38. Li CB, Sprott JC. Aninfinite3-Dquasiperiodiclatticeofchaotic attractors. *Phys Lett A* (2018) 382:581–7. doi:10.1016/j.physleta.2017.12.022

39. Lakshmanan S, Lim CP, Nahavandi S, Prakash M, Balasubramaniam P. Dynamical analysis of the Hindmarsh-Rose neuron with time delays. *IEEE Trans Neural Netw Learn* (2017) 28:1953–8. doi:10.1109/tnnls.2016.2557845

40. Kaslik E. Analysis of two- and three-dimensional fractional-order Hindmarsh-Rose type neuronal models. *Fract Calc Appl Anal* (2017) 20:623–45. doi:10.1515/fca-2017-0033

41. Pham VT, Wang X, Jafari S, Volos C, Kapitaniak T. From Wang–Chen system with only one stable equilibrium to a new chaotic system without equilibrium. *Int J Bifurc Chaos* (2017) 27:1750097. doi:10.1142/s0218127417500973

42. Xu Q, Ding SK, Bao H, Chen M, Bao BC. Piecewise-linear simplification for adaptive synaptic neuron model. *IEEE Trans Circuits Syst Express Briefs* (2022) 69: 1832–6. doi:10.1109/tcsii.2021.3124666

43. Bao H, Chen ZG, Cai JM, Xu Q, Bao BC. Memristive cycle three-neuron-based neural network with chaos and global coexisting attractors. *Sci China Technol Sci* (2022) 65:2582–92.

44. Wen JJ, Wang JP. A chaotic system with infinite attractors based on memristor. *Front Phys* (2022) 10:902500. doi:10.3389/fphy.2022.902500

45. He SB, Fu LX, Lu Y, Wu XM, Wang HH, Sun KH. Analog circuit of a simplified Tent map and its application in sensor position optimization. *IEEE Trans Circuits Syst Express Briefs* (2023) 1. doi:10.1109/tcsii.2022.3217674

46. Ding SK, Wang N, Bao H, Chen B, Wu HG, Xu Q. Memristor synapse-coupled piecewise-linear simplified Hopfield neural network: Dynamics analysis and circuit implementation. *Chaos Solitons Fractals* (2023) 166:112899. doi:10.1016/j.chaos.2022.112899

47. Xu Q, Chen XJ, Chen B, Wu HG, Li Z, Bao H. Dynamical analysis of an improved FitzHugh-Nagumo neuron model with multiplier-free implementation. *Nonlinear Dyn* (2023). doi:10.1007/s11071-023-08274-4

48. Kuznetsov AP, Kuznetsov SP, Mosekilde E, Stankevich NV. Coexisting hidden attractors in a radio-physical oscillator system. *J Phys A Math Theor* (2015) 48:125101. doi:10.1088/1751-8113/48/12/125101

49. Oomens W, Maes JHR, Hasselman F, Egger JIM. A time series approach to random number generation: Using recurrence quantification analysis to capture executive behavior. *Front Hum Neurosci* (2015) 9:319. doi:10.3389/fnhum.2015.00319

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 08 March 2023
DOI 10.3389/fphy.2023.1146173

OPEN ACCESS

EDITED BY
Fei Yu,
Changsha University of Science and Technology, China

REVIEWED BY
Kaihui Wang,
Fudan University, China
Jia Lu,
Hebei University of Technology, China

*CORRESPONDENCE
Long Chen,
1040122@hnust.edu.cn

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 17 January 2023
ACCEPTED 20 February 2023
PUBLISHED 08 March 2023

CITATION
Chen L, Yang Z and Yu Q (2023), Pilot optimization for OFDM in the RSSB system.
Front. Phys. 11:1146173.
doi: 10.3389/fphy.2023.1146173

Pilot optimization for OFDM in the RSSB system

Long Chen*, Ziteng Yang and Qian Yu

School of Information and Electrical Engineering, Hunan University of Science and Technology, Xiangtan, China

To increase the receiving sensitivity of the reversely single-sideband (RSSB) system with orthogonal frequency division multiplexing (OFDM) signal transmission, we use the improved pilot interval scheme and pilot power scheme in this system. The improved pilot interval scheme uses more pilots in areas where channel conditions are relatively good to avoid signal–signal beat interference (SSBI) issues for sub-carriers. The improved pilot power scheme compensates for frequency-selective fading by increasing the pilot power in areas where channel conditions are relatively poor. According to the simulation, the generated 60 GHz optical millimeter wave with 2.5 G bandwidth OFDM signal is delivered over 100 km standard single-mode fiber (SSMF). The improved pilot interval scheme and pilot power scheme can increase the system's receiving sensitivity by 2 dB, respectively. These schemes can enhance the system's performance without increasing the complexity of the algorithm and the costs of the RSSB system.

KEYWORDS

radio-over-fiber, reversely single-sideband, OFDM, pilot interval, pilot power

1 Introduction

In recent years, wireless and broadband have become the research trends of the communication industry. Radio-over-fiber (ROF) technology, combining the advantages of high-frequency wireless communication and optical-fiber communication, can achieve high-frequency signal transmission of ultra-wideband wireless access [1–6]. Orthogonal frequency division multiplexing (OFDM) has become a research hot spot owing to the advantages of its high spectral efficiency and its anti-interference capability. The application of the OFDM to the ROF system can cause millimeter-wave resist dispersion impairments in the fiber link and multipath fading in wireless transmission. Therefore, it is regarded as an effective approach to solving broadband wireless access in the future [7–10]. With the high-frequency spectrum resources, many countries have allocated continuous license-free spectrum resources around 60 GHz. It has motivated many companies and research teams to study the signals in this spectrum segment [11–13]. In the research on ROF, three main issues slow down its development: Improving the energy efficiency of the signal modulation and the signal detection is a concern. The second issue is resisting the loss and walk-off effect in optical fiber links. The last issue is the platform transparency for many signal services in different frequency bands. For the three major problems in ROF, a reversely single-sideband ROF system has been proposed [14]. It is an ROF system without an electric mixer, and it meets the optical frequency doubling. The system has three advantages, including high energy efficiency in modulation, no work-off effect, and multi-band service [15]. Due to its superior performance compared to other earlier systems, the system has been widely recognized and researched since it was proposed [15–18].

Channel estimation is an essential problem in OFDM system design. The basic task of channel estimation is to calculate the channel response for the consequent equalization. The

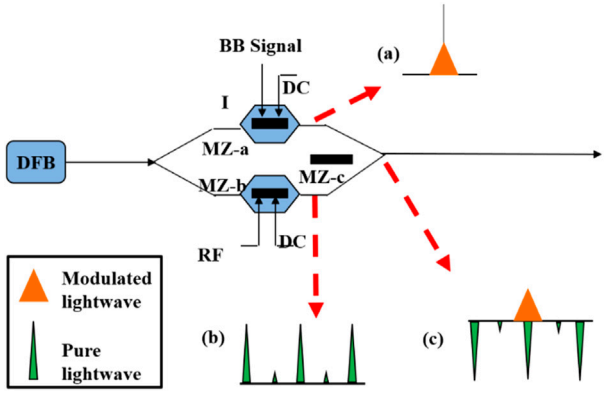


FIGURE 1
Schematic diagram of the RSSB system. **(A)** Modulated lightwave; **(B)** Pure lightwave; **(C)** Combined signal.

channel frequency response or impulse response is often derived from pilot symbols. An efficient pilot scheme is necessary because we need to use the channel information at pilot sub-carriers to estimate the channel at data sub-carriers. There is no research on channel estimation in the reversely single-sideband (RSSB) system. Because of the unique character of the optical system, the pilot scheme that performs well in wireless communication may perform much differently in the RSSB system. In the present RSSB-OFDM system, the interval of the pilots in the OFDM signal is an equal scheme. However, the OFDM signal will be affected by frequency-selective fading and signal–signal mixing interference in the RSSB system, which differs from the impairment in the traditional wireless communication channels. Therefore, estimating the channel through the pilots is an important problem.

In this study, the pilot setting scheme was optimized. According to the features of the RSSB-OFDM system, the improved pilot interval scheme and the improved pilot power scheme are proposed. In the simulation of the pilot interval scheme, we placed more pilots in the area where channel conditions are relatively good, which can avoid signal–signal beat interference (SSBI). The simulation results show that the increased pilot interval scheme can improve the performance of receiving sensitivity. The improved pilot power scheme compensates for frequency-selective fading by increasing the pilot power in areas where channel conditions are relatively poor. The simulation results show that the increased pilot power scheme can improve the performance of receiving sensitivity.

2 Principle of the proposed pilot optimization system

2.1 Principle of the RSSB system

The principle of the RSSB system is shown in Figure 1. The RSSB system is based on a parallel Mach–Zehnder modulator (P-MZM). The P-MZM includes three sub-modulators: up-road Mach–Zehnder modulator-a, down-road Mach–Zehnder modulator-b, and optical-phase modulator-c. The modulator-a that operates in the linear region modulates the baseband OFDM signal. The optical spectrogram of the baseband signal modulated by modulator-a is shown in Figure 1A. The modulator-b that operates in the non-linear region is driven by electrical radio frequency (RF) to produce optical millimeter waves. When the bias voltage is at the highest point, it can obtain a double-frequency modulation in the optical domain. The optical spectrogram after modulation is shown in Figure 1B. The two optical signals are combined in the modulator-c. Because the baseband data are only modulated to a central optical carrier, no data are carried by the sideband. The system reduces the effect of walk-off in the fiber transmission. Modulator-a operates in the linear regions, and modulator-b operates in the non-linear region, which can meet the different modulation requirements of the baseband signal and optical millimeter wave. Therefore, the data format in the system is compatible. When two optical signals are combined in modulator-c, we can adjust the bias voltage in modulator-c to change the phase offset before the coupling of the two optical signals. As the phase shift is set to π, the DC component of the central optical carrier will be suppressed because of the phase cancelation interference, as shown in Figure 1C. Thus, the modulation power efficiency can be improved. Figure 1C shows that the baseband OFDM is reversely modulated onto a light wave owing to the phase shift of π.

2.2 Pilot interval optimization scheme

In the OFDM communication system, some researchers have shown that a pilot distributed in an equal interval can obtain the best performance [19]. In the RSSB-OFDM system, the OFDM signal will be subject to frequency-selective fading and the SSBI during the transmission. Generally, each sub-carrier of the OFDM signal is subject to different intensities of the interference factors. Because the beat interference has a large amplitude near the optical carrier, the sub-carriers at the low frequency are subject to a stronger interference factor. At high frequencies, sub-carriers are subject to strong frequency-selective fading because the electronic devices cannot achieve the ideal spectral characteristics. In addition, the frequency-selective fading factor in optical fiber is directly proportional to the signal frequency. The higher the frequency of the sub-carriers, the more serious the fading, and the greater the power loss.

For the RSSB-OFDM, we can place fewer pilots at high frequency to decrease the loss of the pilots used for channel estimation. It can ensure the accuracy of channel estimation and improve system performance. We set up three pilot interval schemes. First, the equal interval scheme is as follows: keeping the pilot interval constant in the OFDM signal. Second, the increased interval allocation scheme is that the pilot is more distributed in high frequencies. Third, the decreased interval allocation scheme is that the pilot is less distributed in high frequencies. In the three schemes, we keep the total number of inserted pilots the same. We only change the interval distribution.

The three pilot interval schemes shown in Figure 2 are analyzed and simulated. The abscissa axis represents the frequency. The frequency of the intermediate carrier is 0. Frequency increases from the center to both sides. The vertical axis represents the power of the sub-carriers. This graph has three pilot schemes: the

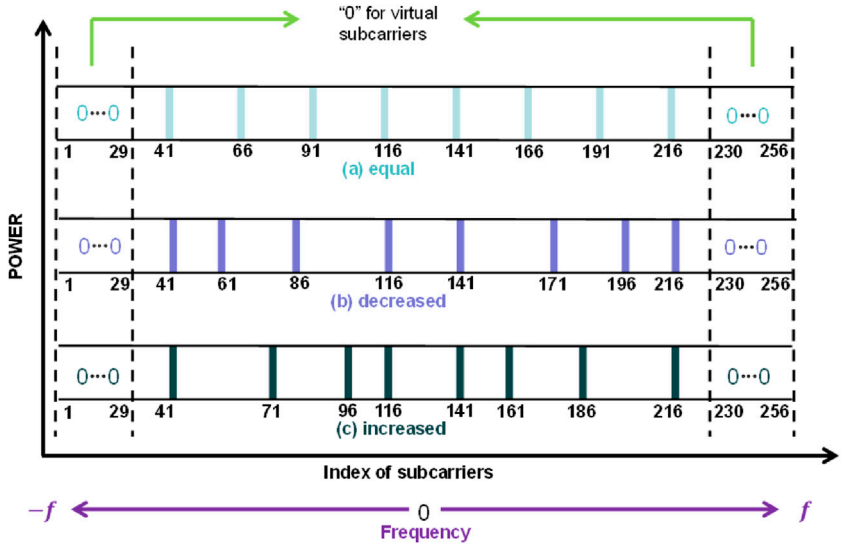


FIGURE 2
Three different pilot interval setting schemes.

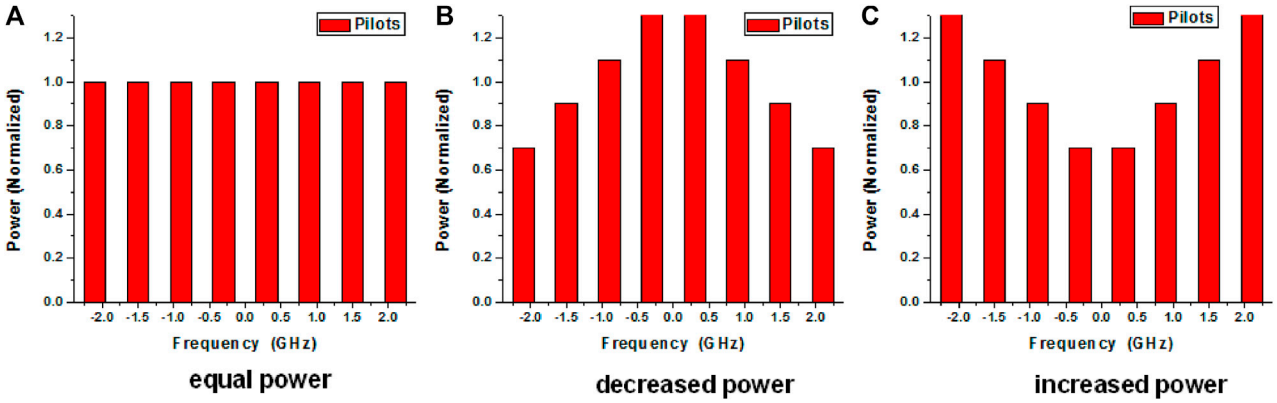


FIGURE 3
Three different pilot power setting schemes. **(A)** The equal-power scheme; **(B)** The decreased-power scheme; **(C)** The increased-power scheme.

equal-interval scheme, the decreased-interval scheme, and the increased-interval scheme, as shown in Figures 2A–C, respectively.

2.3 Pilot power optimization scheme

In the communication system, we can increase the power of the transmitted signal to resist interference under difficult channel conditions. For this consideration, the power of each pilot should be distributed reasonably to improve the accuracy of the channel estimation and reduce the bit error rate (BER) of the system.

Because the sub-carriers are subject to serious interference from frequency-selective fading at high frequencies, we can counter channel interference by increasing the power of the pilot information. Three pilot power schemes are set up in this study. In Figure 3, the abscissa axis represents the frequency, and the frequency at the intermediate carrier is 0. Frequency increases from the center to both sides. The vertical axis represents the normalized power of each sub-carrier. This graph has three different pilot power schemes: the equal-power scheme, the decreased power scheme, and the increased power scheme, as shown in Figures 3A–C, respectively.

3 Simulation design and results

3.1 Pilot interval scheme simulation design and results

We have simulated the pilot-optimized scheme, and the simulation structure diagram is shown in Figure 4. OptiSystem and MATLAB are

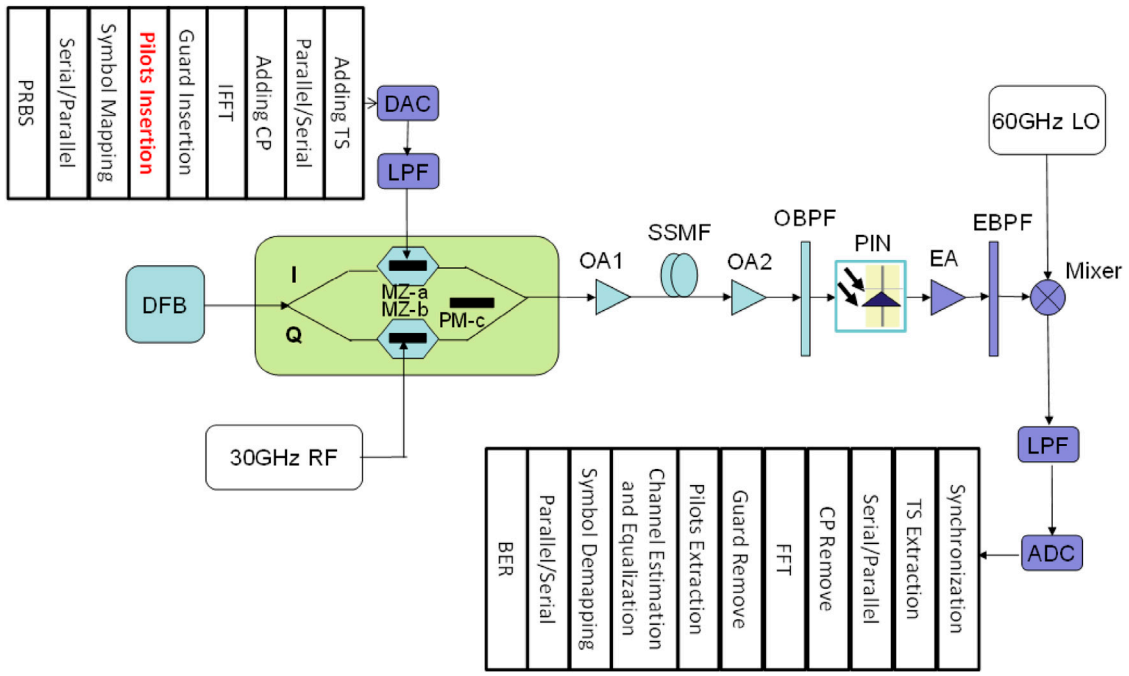


FIGURE 4
Simulation structure diagram of the RSSB-OFDM system with pilot optimization.

used for simulation. Digital baseband OFDM signal is generated using MATLAB. First, the sent data are mapped into QPSK, and each OFDM symbol includes 192 data sub-carriers carrying information. Eight pilots with three different pilot interval schemes are inserted for channel estimation. Fifty-six virtual sub-carriers are used as a guard interval, where zeros are inserted. These virtual sub-carriers are at high frequency, as shown in Figure 2. The cyclic prefix used to reduce inter-symbol interference contains 32 sub-carriers. Therefore, each OFDM symbol consists of 288 sub-carriers. The OFDM signal used in this system is called discrete multi-tone (DMT) modulated signal. The generated OFDM signal is a real-valued signal because of the conjugate symmetric signal used in the IFFT operation. Second, the digital QPSK-mapped OFDM signal is converted into an electrical signal using the digital-to-analog converter (DAC) in the transmitter. Then, a lowpass filter (LPF) is used to filter out the higher-order sidebands and noise from the electrical signal. The continuous light wave with a central wavelength of 1,553.60 nm is generated using a laser, of which the power is 13 dB m and the line width is 10 MHz. The optical signal from the laser is divided into I/Q signals through a power splitter. The power-splitting ratio for the two ports is 1:1. The two optical signals out of the power splitter are the same signal, and both have a power of 6.5 dB m. The electrical OFDM baseband signal with a bandwidth of 2.5 GHz is fed into modulator-a operating in the linear region. The 30-GHz RF signal is fed into modulator-b driving by DC offset at the highest point, thus suppressing the odd sidebands. The optical signals from modulator-a and modulator-b are combined in phase modulator-c. The phase offset between two optical signals changed by modulator-c can adjust the energy efficiency of the signal modulation. In all simulation tests, an optical amplifier (OA) is used to amplify the coupled optical signal to maintain the signal power at 0 dB m before entering the standard single-mode fiber (SSMF). The second OA is used to control the received signal so that it has sufficient optical power after the optical fiber transmission. Then, an optical bandpass filter is used for filtering out the extra sideband. The filtered optical signal, containing the central carrier with data and one of the second-order sidebands, is converted into a high-frequency electrical signal using a photodetector (PD). The electrical 60 GHz mm wave with OFDM signal is through an electrical bandpass filter (EBPF). The EBPF has a center frequency of 60 GHz and a bandwidth of 20 GHz, making sure useful signals get through. For each text, the power is amplified to −14 dB m by an electrical amplifier (EA). Then, the 60 GHz millimeter-wave signal and the 60 GHz RF with a power of 4 dB m are mixed for down-conversion in an electric mixer. An LPF with 3 GHz bandwidth is used to filter out the high-frequency noise after down-conversion. The baseband OFDM signal after ADC is further processed in MATLAB to measure the performance of the link. The demodulation steps are shown in Figure 4. The detailed processes are the same for different pilot schemes. We use three indicators to measure the system performance: BER, error vector magnitude (EVM), and Q-factor. EVM represents the approximate degree between the I/Q components generated by the receiver during the demodulation of the signal and the ideal signal components and is an indicator of the quality of the modulated signal. The specific parameters of each device in this system are shown in Table 1.

It is convenient to set the position of the pilot in the OFDM signal when modulating the OFDM digital baseband signal in MATLAB. Figure 5 shows three electrical spectrum diagrams of OFDM signals in different interval setting schemes. The abscissa axis represents the frequency, and the vertical axis represents the power. The equal-interval scheme, the decreased-interval

TABLE 1 Parameters of the devices in the simulation.

Parameter of the devices		Parameter of the devices	
Laser frequency	193.1 THz	Optical fiber attenuation	0.2 dB/km
Laser line width	10 MHz	Optical fiber dispersion	16.75 ps/nm/km
Laser output power	13 dB m	Fiber dispersion slope	0.075 ps/nm^2/km
Laser initial phase	0	PD sensitivity	1A/W
MZM-a extinction ratio	80 dB	PD dark current	10 nA
MZM-a insertion loss	5 dB	MZM-b extinction ratio	80 dB
MZM-a exchange bias voltage	4 V	MZM-b insertion loss	5 dB
MZM-a exchange RF voltage	4 V	MZM-b exchange bias voltage	4 V
MZM-a modulation voltage 1	2 V	MZM-b exchange RF voltage	4 V
MZM-a modulation voltage 2	2 V	MZM-b modulation voltage	1 4 V
MZM-a bias voltage 1	0 V	MZM-b modulation voltage 2	−4 V
MZM-a bias voltage 2	2 V	MZM-b bias voltage 1	0 V
Modulator-c phase offset	π	MZM-b bias voltage 2	0 V

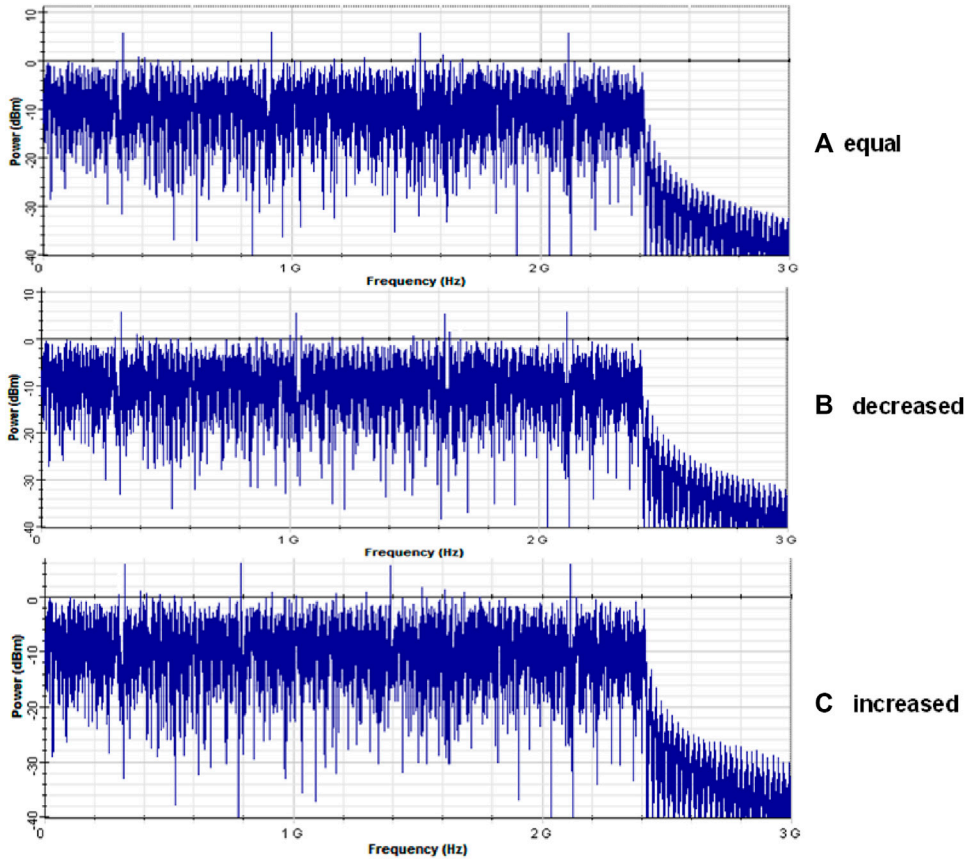


FIGURE 5
OFDM spectrum of different pilot interval schemes in the transmitter. **(A)** The equal-interval scheme; **(B)** The decreased-interval scheme; **(C)** The increased-interval scheme.

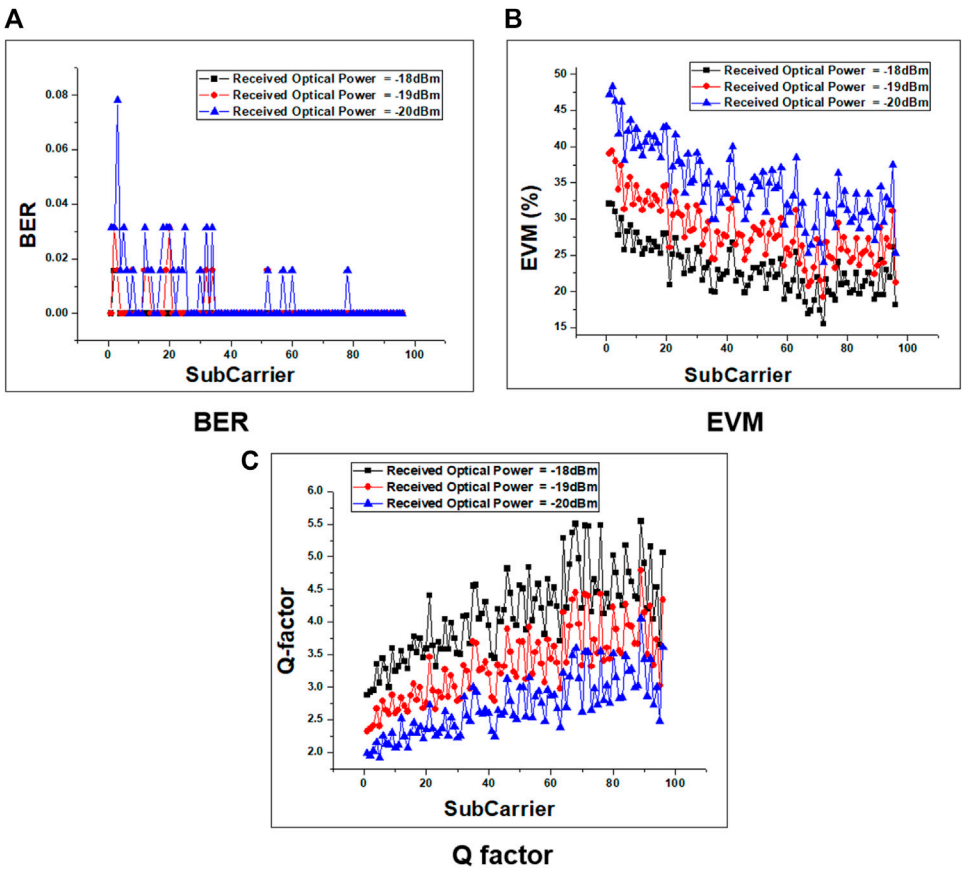


FIGURE 6
BER, EVM, and Q-factor distributions on each data sub-carrier. **(A)** BER distributions; **(B)** EVM distributions; **(C)** Q-factor distributions.

scheme, and the increased-interval scheme are shown in Figures 5A–C, respectively. The larger power amplitude is the pilot information component in each spectrogram. In the three simulation schemes, the power of each pilot information component is equal. The location of the pilots in each scheme can be seen in the spectrogram.

Figure 6 represents the BER, EVM, and Q-factor on each sub-carrier in an OFDM symbol after the signal has been delivered 100 km in the fiber. A total of 192 sub-carriers carry data in the OFDM symbol, and these sub-carriers are set to be conjugate symmetric in the frequency domain. Because the channel characteristics of conjugate symmetric parts in the transmission system are the same, only half of the performance is analyzed and displayed. In Figure 6, the more to the right on the abscissa, the higher the frequency. The sub-carrier with a smaller index is at a high frequency, and the sub-carrier with a bigger index is at a lower frequency. The performance of the system with an optical received power of −18, −19, and −20 dB m is shown in Figure 6, respectively. With the decrease of received optical power, the performance of the system will degrade and the error rate on each sub-carrier will increase. In Figure 6, the results of each sub-graph are consistent and in line with the theory previously described. The data sub-carrier is most influenced by frequency-selective fading at high frequency, leading to higher BER and EVM and lower Q-factor. We can see from the graph that the lower frequency has better performance. However, it is evident from the EVM and Q-factor that the trend disappears at the lowest frequency, mainly because the sub-carrier at low frequency is subject to serious interference from the SSBI, which affects the performance.

Figures 7A–C represent the BER, EVM, and Q-factor curves *versus* different received optical power after delivering 100 km SSMF with different pilot interval schemes. The results of the sub-graphs in Figure 7 are consistent and in line with the theory previously described. The RSSB-OFDM are seriously affected by frequency-selective fading. We can reduce pilot data at high frequency and place more pilot data in the area where the channel condition is relatively good at low frequency, which can improve the accuracy of channel estimation with the pilot. The results show that the increased interval scheme provides the best performance among the three schemes and improves the performance by 2-dB receiving sensitivity compared with the worst scheme.

3.2 Pilot power scheme simulation design and results

The simulation system diagram and device parameters of the pilot power scheme are the same as those of the pilot interval scheme, as shown in Figure 4 and Table 1. The parameters of these pilot scheme simulation cases are the same as those in the OptiSystem. Only the methods of the pilot setting are different in

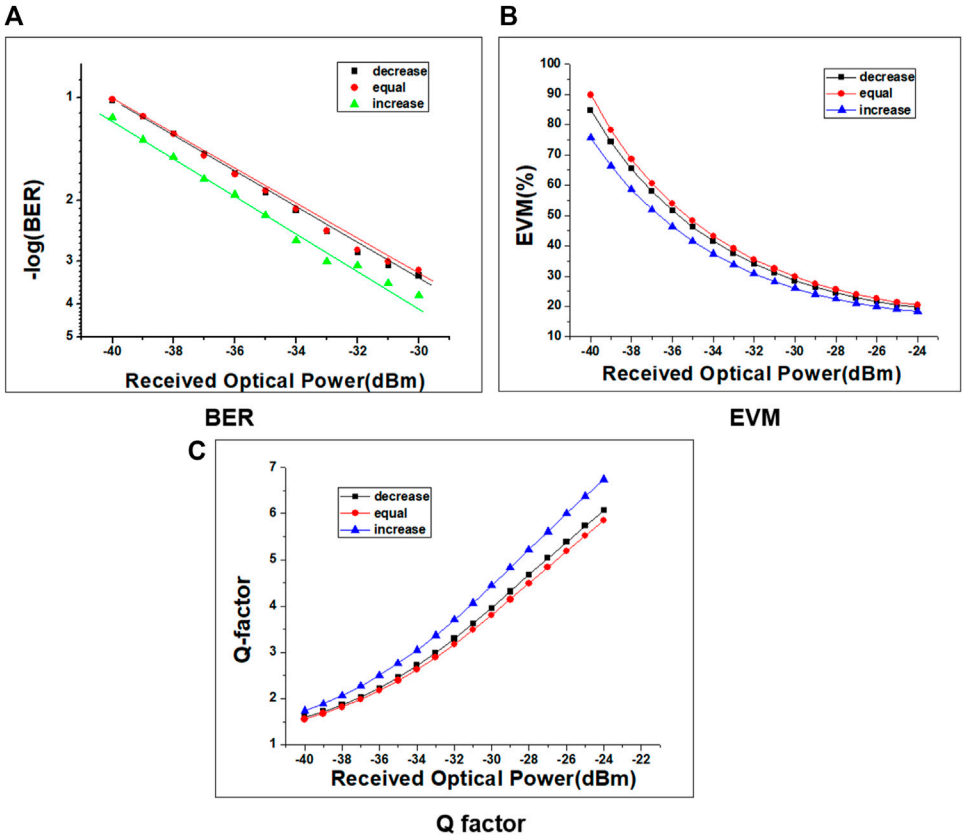


FIGURE 7
BER, EVM, and Q-factor curves under different pilot interval schemes. **(A)** BER curves; **(B)** EVM curves; **(C)** Q-factor curves.

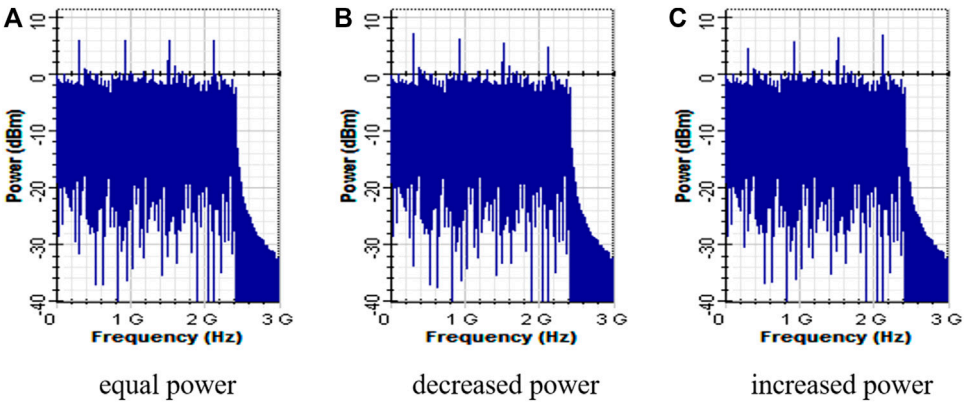


FIGURE 8
OFDM spectrum of different pilot power schemes in the transmitter. **(A)** The equal-power scheme; **(B)** The decreased-power scheme; **(C)** The increased-power scheme.

MATLAB. When modulating the digital OFDM baseband signal in MATLAB, the equal-interval distribution of the pilot position is fixed, and then the power of the pilot data is set separately. There are three different pilot power setting schemes at the transmitter, as shown in Figure 8. The abscissa axis represents the frequency. The vertical axis represents the power. The equal-power scheme, the decreased-power scheme, and the increased-power scheme are shown in Figures 8A–C, respectively. In the three schemes, we keep the total power of the inserted pilots the same and only change the power distribution. In the pilot

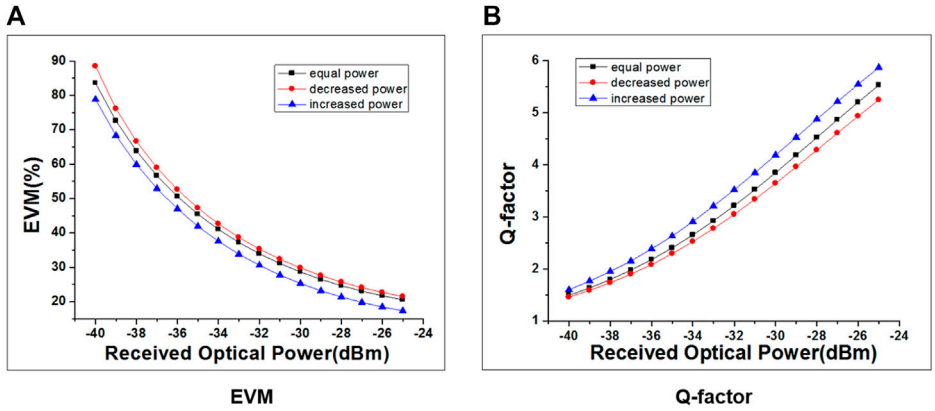


FIGURE 9
EVM and Q-factor curves under different pilot power schemes. **(A)** EVM curves; **(B)** Q-factor curves.

increased-power scheme, the power of the high frequency is increased to resist fading, and the power of the low frequency is decreased to ensure the same total power in each simulation. The larger power amplitude is the pilot information component in each spectrogram. We can see the change in the pilot power in each scheme from the spectrogram.

Figures 9A, B represent the EVM and Q-factor curves *versus* different received power after delivering 100 km of SSMF with different pilot power schemes. In Figure 9, the results of each sub-graph are consistent and in line with the theory previously described. RSSB-OFDM are seriously affected by frequency-selective fading. We can increase the pilot power at a high frequency, which can improve the ability to resist channel interference and the accuracy of channel estimation with the pilot. Therefore, the increased power scheme provides the best performance among the three schemes and improves the performance by 2-dB receiving sensitivity compared with the worst scheme.

4 Conclusion

For the channel characteristics of the RSSB-OFDM, the improved pilot interval scheme and the pilot power scheme are proposed in this study. We built a simulation system to verify the correctness of the theory and the viability of the solution. The improved pilot interval scheme uses the method of placing more pilots in areas where channel conditions are relatively good. The increased interval scheme provides the best performance and improves the performance by 2-dB receiving sensitivity compared with the worst scheme. The improved pilot power scheme uses the method of increasing the pilot power in areas where channel conditions are relatively poor. The increased power scheme provides the best performance and improves the performance by 2-dB receiving sensitivity compared with the worst scheme. These schemes can enhance the system performance without increasing the complexity of the algorithm and the costs of the RSSB system. It is an ideal way to improve the performance of the RSSB system.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material; further inquiries can be directed to the corresponding author.

Author contributions

LC contributed to the conception and design of the study. ZY wrote the first draft of the manuscript. QY wrote sections of the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding

This paper is supported by the National Natural Science Foundation of China (Grant No. 61905074); the Scientific Research Fund of Hunan Provincial Education Department (22B0479, 21B0476); the Hunan Provincial Natural Science Foundation of China (Grant No. 2022JJ30264).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Zhou H, Fei C, Zeng Y, Tan Y, Chen M. A ROF system based on 18-tuple frequency millimeter wave generation using external modulator and SOA. *Opt Fiber Tech* (2021) 61:102402. doi:10.1016/j.yofte.2020.102402

2. Jan B, Komanec M, Jan S, Slavik R, Zvanovec S. Transmitters for combined radio over a fiber and outdoor millimeter-wave system at 25 GHz. *IEEE Photon J* (2020) 12(3): 1–14. doi:10.1109/jphot.2020.2997976

3. Vallejo L, Ortega B, Nguyen D-N, Jan B, Almenar V, Zvanovec S. Usability of a 5G fronthaul based on a DML and external modulation for M-QAM transmission over photonically generated 40 GHz, *IEEE Access* (2020) 8:223730–42. doi:10.1109/access.2020.3042756

4. Chen L, Luo J, Tang Z, Xi Z. Integration system of photon-assisted W-band millimeter wave wireless communication and free space optical communication based on OFDM signal transmission. *Opt Fiber Tech* (2019) 52:101973. doi:10.1016/j.yofte.2019.101973

5. Bohata J, Komanec M, Spáčil J, Hazdra P, Lonský T, Hradecký Z, et al. Experimental demonstration of a microwave photonic link using an optically phased antenna array for a millimeter wave band. *Appl Opt* (2021) 60(4):1013. doi:10.1364/ao.414069

6. Jain V, Bhatia R. Analysis of XPM induced crosstalk in radio over fiber system including the effect of higher-order dispersion parameters. *Opt Quan Elect* (2022) 54(4): 264–16. doi:10.1007/s11082-022-03625-8

7. Kurbatska I, Ostrovskis A, Spolitis S, Bobrovs V. Impact of the receiver and transmitter bandwidth on the performance of high-speed OFDM signal in Radio-over-Fiber Communication Systems. In: 2021 13th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT); October 25-27, 2021; Brno, Czech Republic (2021). p. 282–8.

8. Chen L, Liu M, Tang Z. Nonlinear noise compensation in an integration system of fiber-wireless communication and free space optical communication. *Adv Condensed Matter Phys* (2020) 2020:1–7. doi:10.1155/2020/9605607

9. Deng L, Pang X, Monroy IT, Tang M, Shum P, Liu D. Experimental demonstration of nonlinearity and phase noise tolerant 16-QAMOFDMW-band (75-110GHz) signal over fiber system. *J Lightwave Tech* (2014) 32(8):1442–8. doi:10.1109/jlt.2014.2307054

10. Wang K, Zhao L, Yu JJ. 200 Gbit/s Photonics-aided MMW PS-OFDM signals transmission at W-band enabled by hybrid time-frequency domain equalization. *J Lightwave Tech* (2021) 39(10):3137–44. doi:10.1109/JLT.2021.3062387

11. Saha RK. Spectrum allocation and reuse in 5G new radio on licensed and unlicensed millimeter-wave bands in indoor environments. *Mobile Inf Syst* (2021) 2021:1–21. doi:10.1155/2021/5538820

12. Wang N, Jiao L, Wang P, Li W, Zeng K, "Exploiting beam features for spoofing attack detection in mmWave 60-GHz IEEE 802.11ad networks", *IEEE Trans Wireless Commun* (2021) 20:3321–35. doi:10.1109/twc.2021.3049160

13. Du K, Mujumdar O, Ozdemir O, Ozturk E, Guvenc I, Sichitiu ML, et al. 60 GHz outdoor propagation measurements and analysis using facebook terragraph radios. In: 2022 IEEE Radio and Wireless Symposium (RWS); 16-19 January 2022; Las Vegas, NV, USA (2022). p. 156–9.

14. Cao Z, Yu J, Chen L, Shu Q, "Reversely modulated optical single sideband scheme and its application in a 60-GHz full duplex ROF system," *IEEE Photon Tech Lett* (2012) 24:827–9. doi:10.1109/lpt.2012.2188853

15. Cao Z, Yu J, Li F, Chen L, Shu Q, Tang Q, et al. Energy efficient and transparent platform for optical wireless networks based on reverse modulation. *IEEE J Selected Areas Commun* (2013) 31(12):804–14. doi:10.1109/jsac.2013.sup2.12130013

16. Chen L, He ZW, Zhao L, Zhong B. 60 GHz reverse modulation fiber-wireless system with two jointed DSP algorithms, *Adv Condensed Matter Phys* (2018) 2018:1–6. doi:10.1155/2018/7256162

17. Mao MZ, Giddings RP, Cao BY, Xu YT, Wang M, Tang JM. DSP-enabled reconfigurable and transparent spectral converters for converging optical and mobile fronthaul/backhaul networks. *Opt Express* (2017) 25:13836–56. doi:10.1364/oe.25.013836

18. Bhatia R, Prakash S, Saini E. Performance improvement of 60-GHz wireless optical systems with reverse-parallel hybrid modulation scheme. *Int J Commun Syst* (2019) 32(2):e3848. doi:10.1002/dac.3848

19. Negi R, Cioffi J. Pilot tone selection for channel estimation in a mobile OFDM system. *IEEE Trans Consumer Elect* (1998) 44(3):1122–8. doi:10.1109/30.713244

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 13 March 2023
DOI 10.3389/fphy.2023.1159212

OPEN ACCESS

EDITED BY
Hairong Lin,
Hunan University, China

REVIEWED BY
Yujiao Dong,
Hangzhou Dianzi University, China
Sen Zhang,
Nanjing Normal University, China

*CORRESPONDENCE
Xiangyu Lan,
1299464560@qq.com
Jie Jin,
jj67123@hnust.edu.cn

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 05 February 2023
ACCEPTED 17 February 2023
PUBLISHED 13 March 2023

CITATION
Lan X, Jin J and Liu H (2023), Towards non-linearly activated ZNN model for constrained manipulator trajectory tracking.
Front. Phys. 11:1159212.
doi: 10.3389/fphy.2023.1159212

Towards non-linearly activated ZNN model for constrained manipulator trajectory tracking

Xiangyu Lan[1]*, Jie Jin[1,2]* and Haiyan Liu[2]

[1]School of Information and Electrical Engineering, Hunan University of Science and Technology, Xiangtan, China, [2]Hunan key laboratory of the research and development of novel pharmaceutical preparations, Changsha Medical University, Changsha, China

As a powerful method for time-varying problems solving, the zeroing neural network (ZNN) is widely applied in many practical applications that can be modeled as time-varying linear matrix equations (TVLME). Generally, existing ZNN models solve these TVLME problems in the ideal no noise situation without inequality constraints, but the TVLME with noises and inequality constraints are rarely considered. Therefore, a non-linear activation function is designed, and based on the non-linear activation function, a non-linearly activated ZNN (NAZNN) model is proposed for solving constrained TVLME (CTVLME) problems. The convergence and robustness of the proposed NAZNN model are verified theoretically, and simulation results further demonstrate the effectiveness and superiority of the NAZNN model in dealing with CTVLME and the constrained robot manipulator trajectory tracking problems. In addition, the wheeled robot trajectory tracking fault problems with physical constraints are also analyzed theoretically, and the proposed NAZNN model is also applied to the manipulator trajectory tracking fault problem, and the experimental results prove that the NAZNN model also deal with the manipulator trajectory tracking fault problem effectively.

KEYWORDS

zeroing neural network, activation function, robot trajectory tracking, trajectory tracking fault problem, ABB simulation

1 Introduction

With the acceleration of industrialization, manipulator has been widely used in industrial production. However, it is important to realize accurate control of robotic manipulators because as there are various disturbances and constraints in the production environment. In general, the behavioural control of a robot manipulator can be modelled as a TVLME problem [1–3]. When designing behavioural controllers for robot manipulators, there are necessary to consider robot dynamics [4,5], torque saturation [6] and obstacle avoidance [7,8] of the problem. For example, wheeled robots are non-linear systems and their controllers do not consider sliding, so it is difficult for these controllers to have good control when wheeled robots work with some complex environments, such as wet and icy, uneven surfaces. Moreover, the problems of dynamic coupling, dynamic limitations caused by the environments, and delay problems of the controller are also should be considered, and they complicate the control process of manipulator trajectory tracking. Therefore, researchers have proposed the PID control [9,10], feedback control [11], finite-time control [12–14], fuzzy control [15–18] and neural network control [19–25] to solve the above problems.

In recent years, with the continuous improvement and development of deep learning [26–30], neural networks [31–36] have become an efficient solution for various time-varying problems. For example, Jin et al. build the interference-tolerant fast convergence zeroing neural network (ITFCZNN) model [37] based on a new activation function, which exhibit excellent time-varying Robustness and convergence. One of the things that must be mentioned is that the activation function heavily influences the performance of the model. Therefore, the researchers have proposed a corresponding novel activation function in order to achieve the desired experimental results. For example [38], proposed zero-tuned neural networks (ZTNN) to solve the Stein matrix equation based on several new activation functions. In [39], it is proposed that the PSAF-based ZNN model is applied to the secondary programming problem. A ZNN model based on the activation function (HSAF) is designed in [40] to solve the time-varying square root of the matrix. In [41,42], it is proposed to apply the SBPAF-based ZNN model to solve the time-varying Sylvester equation. In [43,44], a ZNN model with adjustable convergence rate is designed based on the new activation function and applied to the control process of a robotic arm. However, most practical problems can be modelled as time-varying non-linear systems of equations with constraints. In contrast, the ZNN model described above only considers the unconstrained ideal environment and may not work as well as expected in practical applications, especially for the control of robot behaviour.

The physical constraints of industrial robots must be taken into account in the actual control process, otherwise specific tasks cannot be performed and even the hardware of the industrial robot is damaged. Considering the physical constraints of industrial robots are less studied. In [45], the motion process of an industrial robot is modelled as a multilayer time-varying problem in order to solve the problem of joint angle constraints in the control process [46]. proposed a physical limit constrained minimum velocity parametric coordination scheme to solve the constraints of a wheeled robot. In [47], the analytical solution of the robot controller is solved based on a parametric approach, which in turn yields the joint angle range of the industrial robot. However, the above methods only consider the angle overrun fault of the robotic manipulator, and they ignore other faults, such as speed overrun, noises, and the robotic manipulator does not stop in the case of angle overrun, which is possible in practical situations.

It is clear that the usage of neural networks to solve constrained manipulator trajectory tracking is still far from being investigated, and the potential of neural networks in this area remains to be exploited. Therefore, this paper develops the NAZNN for manipulator trajectory tracking with joint angle and joint velocity constraints in noisy environments.

The remainder of the paper is organized as follows. The modeling process and theoretical proof of the NAZNN model are analyzed in Section 2. Examples of the NAZNN model for solving CTVLME problems are presented in Section 3. Modeling of a wheeled robot is provided in Section 4, and simulation experiments of the NAZNN model for failure case of manipulator trajectory tracking with physical constraints are provided in Section 4. Finally, The conclusions of the paper are given in Section 5.

2 CTVLMA problem description and NAZNN model

2.1 The CTVLMA problem

In mathematics, a constrained time-varying linear matrix equation (CTVLMA) can be expressed as

$$\begin{cases} \mathbf{A}(t)\mathbf{X}(t) = \mathbf{C}(t) \\ \mathbf{B}(t)\mathbf{X}(t) \le \mathbf{D}(t) \end{cases} \tag{1}$$

where $\mathbf{A}(t) \in R^{m\times n}$ and $\mathbf{B}(t) \in R^{p\times n}$ are smoothed full rank matrices, and $m < n$. $\mathbf{C}(t) \in R^m$ and $\mathbf{D}(t) \in R^p$ are smooth vectors. And $\mathbf{X}(t) \in R^n$ is the unknown vector to be solved. The time-varying solution $\mathbf{X}(t)$ of $\mathbf{A}(t)\mathbf{X}(t) = \mathbf{C}(t)$ exists only if it satisfies the restriction $\mathbf{B}(t)\mathbf{X}(t) \le \mathbf{D}(t)$.

For CTVLMA (1), the following equation is obtained by introducing a non-negative variable $v^2 \in R^p$.

$$\begin{cases} \mathbf{A}(t)\mathbf{X}(t) = \mathbf{C}(t) \\ \mathbf{B}(t)\mathbf{X}(t) + v^2(t) = \mathbf{D}(t) \end{cases} \tag{2}$$

Here $v^2(t)$ is also a unknown vector, and the superscript .2 denotes the square operation of each element of $v(t) \in R^p$. Defining the logarithmic matrix $\mathfrak{R}(t) = diag[v_1(t), v_2(t), \cdots v_p(t)] \in R^{p\times p}$ leads to $v^2(t) = \mathfrak{R}(t)v(t)$. Thus, Equation 2 can be represented in a matrix form below.

$$\begin{bmatrix} \mathbf{A}(t) & 0 \\ \mathbf{B}(t) & \mathfrak{R}(t) \end{bmatrix} \begin{bmatrix} \mathbf{X}(t) \\ v(t) \end{bmatrix} = \begin{bmatrix} \mathbf{C}(t) \\ \mathbf{D}(t) \end{bmatrix} \tag{3}$$

Then, we define $\mathbf{V}(t) = \begin{bmatrix} \mathbf{A}(t) & 0 \\ \mathbf{B}(t) & \mathfrak{R}(t) \end{bmatrix} \in R^{(m+p)(n+p)}$, $\mathbf{w}(t) = [\mathbf{C}(t) \ \mathbf{D}(t)]^T \in R^{m+p}$ and $\mathbf{u}(t) \in [\mathbf{X}(t)^T, v(t)^T]^T \in R^{n+p}$ where the upper label T denotes permutation.

Then, the following equation is obtained based on the above equivalent substitution.

$$\mathbf{V}(t)\mathbf{u}(t) = \mathbf{w}(t) \tag{4}$$

Obviously, Eq. 4 is equivalent to Eq. 2, and solving CTVLMA (1) is transformed into solving the matrix-vector Equation 4.

2.2 NAZNN model for CTVLMA problem solving

In ordered to find the solution of Eq. 4, the NAZNN is designed according to the following steps.

Firstly, denote the following error function $\mathbf{E}(t)$.

$$\mathbf{E}(t) = \mathbf{V}(t)\mathbf{u}(t) - \mathbf{w}(t) \tag{5}$$

Then, the time derivatives of $\mathbf{E}(t)$ are deduced below.

$$\dot{\mathbf{E}}(t) = \lambda\mathbf{F}(\mathbf{E}(t)) \tag{6}$$

where $\lambda > 0$ is the convergence factor, and $\mathbf{F}(\bullet)$ is an activation function. In order to construct our model for solving Eq. 4, the following theorem 1 is introduced in advance.

Lemma 1: The time derivatives of $v^2(t)$ can be written as

$$\dot{v}^2(t) = \frac{dv^2(t)}{dt} = 2\mathfrak{R}(t)\dot{v}(t) \tag{7}$$

where $\dot{v}(t)$ is the time derivatives of $v(t)$.

Proof: Let $\dot{\mathfrak{R}}(t)$ represent the temporal derivatives of $\mathfrak{R}(t)$, and we can obtain the following equation.

$$\dot{v}^2(t) = \dot{\mathfrak{R}}(t)v(t) + \mathfrak{R}(t)\dot{v}(t) \tag{8}$$

Then,

$$\begin{aligned}\dot{\mathfrak{R}}(t)\mathrm{v}(t) &= \begin{bmatrix} \dot{v}_1(t) & 0 & \cdots & 0 \\ \vdots & \dot{v}_2(t) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \dot{v}_p(t) \end{bmatrix}\begin{bmatrix} v_1(t) \\ v_2(t) \\ \vdots \\ v_p(t) \end{bmatrix} \\ &= \begin{bmatrix} v_1(t) & 0 & \cdots & 0 \\ \vdots & v_2(t) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & v_p(t) \end{bmatrix}\begin{bmatrix} \dot{v}_1(t) \\ \dot{v}_2(t) \\ \vdots \\ \dot{v}_p(t) \end{bmatrix} \\ &= \mathfrak{R}(t)\dot{v}(t)\end{aligned} \tag{9}$$

Then, Eq. 8 can be expressed in the following form.

$$v^2(t) = dv^2(t)/dt = 2\mathfrak{R}(t)\dot{v}(t) \tag{10}$$

The proof is completed.

Based on Eq. 5, Eq. 6 and Lemma 1, the following NAZNN model is obtained.

$$\begin{aligned}\begin{bmatrix} \mathbf{A}(t) & 0 \\ \mathbf{B}(t) & 2\mathfrak{R}(t) \end{bmatrix}\begin{bmatrix} \dot{\mathbf{X}}(t) \\ \dot{v}(t) \end{bmatrix} &= -\begin{bmatrix} \dot{\mathbf{A}}(t) & 0 \\ \dot{\mathbf{B}}(t) & 0 \end{bmatrix}\begin{bmatrix} \mathbf{X}(t) \\ v(t) \end{bmatrix} + \begin{bmatrix} \dot{\mathbf{C}}(t) \\ \dot{\mathbf{D}}(t) \end{bmatrix} \\ &\quad -\lambda\left(\begin{bmatrix} \mathbf{A}(t) & 0 \\ \mathbf{B}(t) & \mathfrak{R}(t) \end{bmatrix}\begin{bmatrix} \mathbf{X}(t) \\ v(t) \end{bmatrix} - \begin{bmatrix} \mathbf{C}(t) \\ \mathbf{D}(t) \end{bmatrix}\right)\end{aligned} \tag{11}$$

where $\dot{\mathbf{X}}(t)$, $\dot{\mathbf{A}}(t)$, $\dot{\mathbf{B}}(t)$, $\dot{\mathbf{C}}(t)$ and $\dot{\mathbf{D}}(t)$ are the time-derivatives of $\mathbf{X}(t)$, $\mathbf{A}(t)$, $\mathbf{B}(t)$, $\mathbf{C}(t)$ and $\mathbf{D}(t)$.

Let us set $\mathbf{N}(t) = \begin{bmatrix} \mathbf{A}(t) & 0 \\ \mathbf{B}(t) & 2\mathfrak{R}(t) \end{bmatrix}$, $\dot{\mathbf{N}}(t) = \begin{bmatrix} \dot{\mathbf{A}}(t) & 0 \\ \dot{\mathbf{B}}(t) & 0 \end{bmatrix}$, $\dot{\mathbf{u}}(t) = [\,\dot{\mathbf{X}}(t) \;\; \dot{v}(t)\,]^T$, $\dot{\mathbf{M}}(t) = [\,\dot{\mathbf{C}}(t) \;\; \dot{\mathbf{D}}(t)\,]^T$, and Eq. 11 can be transformed to the following simplified NAZNN (12).

$$\mathbf{N}(t)\dot{\mathbf{u}}(t) = -\dot{\mathbf{N}}(t)\mathbf{u}(t) + \dot{\mathbf{M}}(t) - \lambda\mathbf{F}(\mathbf{N}(t)\mathbf{u}(t) - \mathbf{M}(t)) \tag{12}$$

The NAZNN model with noise $\mathbf{Y}(t)$ can be expressed in the following form.

$$\mathbf{N}(t)\dot{\mathbf{u}}(t) = -\dot{\mathbf{N}}(t)\mathbf{u}(t) + \dot{\mathbf{M}}(t) - \lambda\mathbf{F}(\mathbf{N}(t)\mathbf{u}(t) - \mathbf{M}(t)) + \mathbf{Y}(t) \tag{13}$$

where $\lambda > 0$ and $\mathbf{F}(\bullet)$ are the same definitions above.

Actually, the performances of the ZNN models are closely related with the activation functions $\mathbf{F}(\bullet)$, and the existing activation functions are listed in Table 1. Additionally, $f(\bullet)$ denotes the element of $\mathbf{F}(\bullet)$.

In order to enhance the performances of the NAZNN (13), the following non-linear activation function (14) is designed.

$$f(x) = \left[(k_1|x|^p + k_2|x|^q)^m + k_3x + k_4e^{|x|^z}|x|^{1|z}\right]sgn(x) \tag{14}$$

where $m > 0$, $p > 0$, $q > 0$, $k_1 > 0$, $k_2 > 0$, $k_3 > 0$, $k_4 > 0$, $0 < z < 1$ and sgn(x) denotes the signum function.

2.3 Convergence and robustness analysis of the NAZNN model

2.3.1 Convergence analysis

The following Lemma 2 is provided in advance for the convergence analysis of the proposed NAZNN model.

Lemma 2: [48,49] Considering a non-linear dynamic system as follows.

$$G = f(G(t), t) \tag{15}$$

where $f(\bullet)$ is a continuously non-linear function. Assumed that there is a continuous function $L(b)$ satisfying both of the following conditions.

(1) $L(b)$ is a radial unconstrained function intersecting with $L(\xi) = 0$.
(2) $L \le -(pL^a(\xi(t))) + qL^b(\xi(t)))^l$, where $p > 0$, $q > 0$, $a > 0$, $b > 0$, $l > 0$ and $al > 1$, $bl < 1$.

Then, the above dynamic system (15) is fixed time stable, and the upper bound of its stable time is $T_{\max} \le 1/(p^l(al - 1)) + 1/(q^l(1 - bl))$.

The following Theorem 1 and 2 guarantee the convergence and robustness of the proposed NAZNN model (12), respectively.

Theorem 1: If the NAZNN model (12) is not polluted by noise, and Eq. 1 is smooth at all times. For any initial system state $\mathbf{X}(t)$, the state solution of model (12) converges to the theoretical solution of Eq. 1 at a predetermined time t_s.

$$t_s = 1/\left(\lambda{k_1}^m(pm - 1)\right) + 1/\left(\lambda{k_2}^m(1 - qm)\right) \tag{16}$$

where $k_1 > 0$, $k_2 > 0$, $\lambda > 0$, $q > 0$, $m > 0$, $mp > 1$, $qm < 1$.

Proof: If the error function $\mathbf{E}(t)$ in Eq. 5 converges to 0, the neural state solution of NAZNN (12) will be equal to the theoretical solution of Eq. 1, and the evolutionary formulation (6) assures the convergence of the error function $\mathbf{E}(t)$. Furthermore, Eq. 6 consists of n^2 independent subsystems. Therefore, we only need to show these subsystems are stable at a fixed time.

$$\dot{e}_{ij}(t) = -\lambda\mathbf{F}\left(e_{ij}(t)\right); i, j \in \{1, 2, \cdots n\} \tag{17}$$

where the scalars $\dot{e}_{ij}(t)$ and $e_{ij}(t)$ mean the elements of the ith row and jth column of $\dot{\mathbf{E}}(t)$ and $\mathbf{E}(t)$ respectively. The flowing Lyapunov function is adopted for the convergence validation.

$$h(t) = \left|e_{ij}(t)\right| \tag{18}$$

Derive for $h(t)$:

$$\dot{h}(t) = \dot{e}_{ij}sgn\left(\dot{e}_{ij}(t)\right) = -\lambda\mathbf{F}\left(e_{ij}(t)\right)sgn\left(\dot{e}_{ij}(t)\right) \tag{19}$$

Then, substituting the non-linear activation function (14) into the above formula yields

$$\begin{aligned}\dot{h}(t) &= -\lambda\left[\left(k_1\left|e_{ij}(t)\right|^p + k_2\left|e_{ij}(t)\right|^q\right)^m + k_3\exp\left(\left|e_{ij}(t)\right|^z\right)\left|e_{ij}(t)\right|^{1-z}(t)/z + k_4\right] \\ &\le -\lambda\left(k_1\left|e_{ij}(t)\right|^p + k_2\left|e_{ij}(t)\right|^q\right)^m \quad = -\left(\lambda^{1/m}k_1h^p + \lambda^{1/q}k_2h^q\right)^m\end{aligned} \tag{20}$$

Based on Lemma 2, the upper bound of convergence time t_{ij} for the subsystem in row i and column j of $\mathbf{E}(t)$ is obtained.

TABLE 1 Classical activation function and ZNN model.

AFs	Expressions	ZNN model
LAF	$f(x) = x$	IZNN-1
LAF	$f(x) = kx,\ k \neq 1$	IZNN-2
HSAF	$f(x) = (e^{(kx)} - e^{(-kx)})/2$ with $k \geq 1$	IZNN-3
PSAF	$f(x) = kx^{v};\ k > 0,\quad v = 1, 3, 5 \ldots$	IZNN-4
SBPAF	$f(x) = (k_1\lvert x\rvert^{p} + k_2\lvert x\rvert^{\frac{1}{q}})sgn(x);\ p > 0, q > 0$	IZNN-5

$$t_s = \max\left(t_{ij}\right) \leq 1\Big/\left(\lambda {k_1}^m\left(mp-1\right)\right) + 1\Big/\left(\lambda {k_2}^m\left(1-qm\right)\right) \tag{21}$$

On basis of the above analysis, it can be drawn the conclusion that the state solution of the ZNN model (12) converges to the theory solutions of Eq. 1 in a predetermined time.

2.3.2 Robustness analysis

The robustness of NAZNN model (13) in noisy environment is discussed in this subsection. In order not to lose generality, the noise in Eq.13 is chosen to be $\mathbf{Y}(t) = 0.1exp(0.5t)$, and the following Theorem 2 guarantees the robustness of NAZNN model (13).

Theorem 2: Assume the theoretical solution of Eq. 1 exists, and each element of $\mathbf{Y}(t)$ satisfies the conditions $|y_{ij}(t)| \leq \delta|e_{ij}(t)|$ and $0 < \delta < \lambda k_3 e/z$. The NAZNN model (13) has the following fixed convergence time t_s with noise $\mathbf{Y}(t)$.

$$t_s = 1/\left(\lambda {k_1}^m\left(pm-1\right)\right) + 1/\lambda {k_2}^m\left(1-qm\right) \tag{22}$$

Proof: Similar with the analysis of Theorem 1, Eq. 6 also contains n^2 mutually independent subsystems with noise $\mathbf{Y}(t)$.

$$\dot{e}_{ij} = -\lambda\mathbf{F}\left(e_{ij}(t)\right) + y_{ij}(t) \tag{23}$$

The following Lyapunov function is selected for the stable validation of $\mathbf{E}(t)$.

$$h(t) = \left|e_{ij}(t)\right|^2 \tag{24}$$

The derived of $h(t)$ is obtained below.

$$\begin{aligned}\dot{h}(t) &= 2e_{ij}(t)\dot{e}_{ij}(t) = 2e_{ij}(t)\left(-\lambda\mathbf{F}\left(e_{ij}(t)\right) + y_{ij}(t)\right)\\ &= 2e_{ij}(t)\left(-\lambda\left(k_1\left|e_{ij}(t)\right|^{p} + k_2\left|e_{ij}(t)\right|^{q}\right)^{m}\right.\\ &\quad \left. + \mathrm{k}_3\exp\left(\left|\mathrm{e}_{\mathrm{ij}}(\mathrm{t})\right|^{z}\right)\left|\mathrm{e}_{\mathrm{ij}}(\mathrm{t})\right|^{1-z}(\mathrm{t})\Big/\mathrm{z} + k_4\dot{\mathrm{s}}\mathrm{gn}\left(e_{ij}(t)\right) + y_{ij}(t)\right)\end{aligned} \tag{25}$$

According to the conditions $|y_{ij}(t)| \leq \delta|e_{ij}(t)|$, we can obtain the following inequalities.

$$\begin{aligned}\dot{h}(t) \leq\ & -2\lambda\left|e_{ij}(t)\right|\left(\left(k_1\left|e_{ij}(t)\right|^{p} + k_2\left|e_{ij}(t)\right|^{q}\right)^{m}\right.\\ &\left. + 2\delta\left|e_{ij}(t)\right|^2 + 2\lambda k_3\Big/z\exp\left(\left|e_{ij}(t)\right|^{z}\right)\left|e_{ij}(t)\right|^{2-z}(t)\Big/z\right)\end{aligned} \tag{26}$$

In order to simplify the approach, setting up the following equation

$$R(t) = 2\delta\left|e_{ij}(t)\right|^2 + 2\lambda k_3\Big/z\exp\left(\left|e_{ij}(t)\right|^{z}\right)\left|e_{ij}(t)\right|^{2-z} \tag{27}$$

Then,

$$\begin{aligned}R(t) &= 2\delta\left|e_{ij}(t)\right|^2 + 2\lambda k_3\Big/z\exp\left(\left|e_{ij}(t)\right|^{z}\right)\left|e_{ij}(t)\right|^{2-z}\\ &= 2\left|e_{ij}(t)\right|^2\left[\delta - \lambda k_3\exp\left(\left|e_{ij}(t)\right|^{z}\right)\Big/\left(z\left|e_{ij}(t)\right|^{z}\right)\right]\end{aligned} \tag{28}$$

Let $u_1(t) = \delta - \lambda k_3\exp(|e_{ij}(t)|^z)/(z|e_{ij}(t)|^z)$, $f(t) = \lambda k_3\exp(|e_{ij}(t)|^z)/(z|e_{ij}(t)|^z)$ and $\varpi = |e_{ij}(t)|^z$, $f(t)$ can be simplified to the following form.

$$f(\varpi) = \lambda k_3 e^{\varpi}/(z\varpi) \tag{29}$$

The derivative of (29) is provided in Eq. 30.

$$\dot{f}(\varpi) = e^{\varpi}(\varpi - 1)\Big/\varpi^2 \tag{30}$$

According to the above formula, it is clear that when $0 < \varpi < 1$, the derivative of $f(\varpi)$ is less than 0, and $f(\varpi)$ decreases monotonically; when $\varpi > 1$, $f(\varpi) > 0$ and the derivative of $f(\varpi)$ rises monotonically. Obviously, on basis of the above monotonic analysis, $\varpi = 1$ is equivalent to z being close to 0 and $f(\varpi)$ achieving a minimum, which indicates $f(t)$ reaches its minimum value.

$$f(t)_{\min} = \lambda k_3 e/z \tag{31}$$

Since $\delta \leq \lambda k_3 e/z$, δ is less than the minimum value of $f(t)$, and $u_1(t)$ is always less than 0. In summary, inequality (26) can be further simplified to the following form.

$$\begin{aligned}\dot{h}(t) &\leq -2\lambda\left|e_{ij}(t)\right|\left(k_1\left|e_{ij}(t)\right|^{p} + k_2\left|e_{ij}(t)\right|^{q}\right)^{m} + R(t)\\ &\leq -2\lambda\left|e_{ij}(t)\right|\left(a_1\left|e_{ij}(t)\right|^{p} + k_2\left|e_{ij}(t)\right|^{q}\right)^{m}\\ &= -2\lambda\left|e_{ij}(t)\right|^{1/m}\left(k_1\left|e_{ij}(t)\right|^{p} + k_2\left|e_{ij}(t)\right|^{q}\right)^{m}\\ &= -\left[(2\lambda)^{1/m}\left(k_1\left|e_{ij}(t)\right|^{(p+1/m)} + k_2\left|e_{ij}(t)\right|^{(q+1/m)}\right)\right]^{m}\\ &= -\left[(2\lambda)^{1/m}\left(k_1 h^{(1+pm)/2m} + (2\lambda)^{1/m}\left(k_2 h^{(1+pm)/2m}\right)\right]m\end{aligned} \tag{32}$$

Then, based on Lemma 2, the convergence time of the ijth element of $\mathbf{E}(t)$ is obtained.

$$\begin{aligned}t_{ij} &\leq 1\Big/\left(2\lambda {k_1}^m\left(m\left(pm+1\right)/2m-1\right)\right) + 1/2\lambda {k_2}^m\left(1-m\left(m\left(pm+1\right)\right)\right)\\ &= 1/2\lambda {k_1}^m\left(\left(pm+1\right)/2-1\right) + 1/2\lambda {k_2}^m\left(1-\left(qm+1\right)/2\right)\\ &= 1/\left(\lambda {k_1}^m\left(qm-1\right)\right) + 1/\left(\lambda {k_2}^m\left(1-qm\right)\right)\end{aligned} \tag{33}$$

The maximal convergence time t_s of the ZNN model (13) is

$$t_s = \max\left(t_{ij}\right) \leq 1\Big/\left(\lambda {k_1}^m\left(pm-1\right)\right) + 1\Big/\left(\lambda {k_2}^m\left(1-qm\right)\right) \tag{34}$$

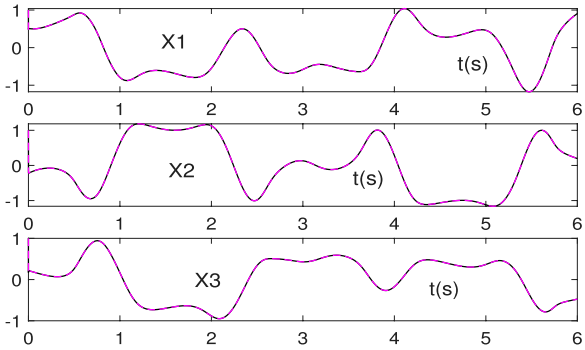


FIGURE 1
Neural state solution of the NAZNN model when $m = 5$ and $\lambda = 1$.

From the above analyses, we can draw the conclusion that the proposed NAZNN model has the fixed convergence time t_s in noiseless and noisy environments.

3 Numerical simulations for CTVLME solving

In this section, an illustrative example of the NAZNN model for solving CTVLME (1) is introduced to demonstrate its efficiency and accuracy.

Consider CTVLME (1) with the following coefficients.

$$\begin{cases} \mathbf{A}(t)\mathbf{X}(t) = \mathbf{C}(t) \\ \mathbf{B}(t)\mathbf{X}(t) \le \mathbf{D}(t) \end{cases} \tag{35}$$

where $\mathbf{A}(t) = [\sin(t)+2 \quad \cos(t)+3 \quad -\sin(t)+5]$, $\mathbf{B}(t) = \begin{bmatrix} \sin(2t) & -\cos(3t) \\ \cos(3t) & \sin(3t) \\ -\sin(3t) & \cos(3t) \end{bmatrix}^T$, $\mathbf{C}(t) = \sin(2t) + \cos(2t), \mathbf{D}(t) = \begin{bmatrix} \sin(4t) \\ \cos(4t) \end{bmatrix}$.

Here, we set all the parameters $m, p, q, z, k_1, k_2, k_3, k_4$ in the nonlinear activation function (14) to 5, and the state solutions generated by the proposed NAZNN model are shown in Figure 1.

According to Figure 1, it can be observed that the black curves quickly coincide with the red curves, which indicate that the state solutions generated by the NAZNN model converge to the theoretical solutions of Eq. 1 in a very brief period of time.

In addition, the residual errors of the NAZNN model (12) and the IZNN model activated by LAF, PSAF, HSAF and SBPAF for solving the above same equation are compared in Figure 2. Figure 2A shows only the residual error of the NAZNN model (12) converges rapidly to 0, while the IZNN model activated by the activation functions converges slowly.

Furthermore, it can be observed that the convergence speed becomes faster as the value of m increases in Figure 2B. This phenomenon indicates that the time required for the NAZNN model (12) to solve inequality (1) becomes shorter as m increases. Finally, from Figure 2C, we can see that the convergence time of the NAZNN model becomes smaller as the convergence coefficient λ increases, which indicates that the convergence speed of the NAZNN model is also closely related to its convergence coefficient. Therefore, we can set these parameter values according to the accuracy requirements.

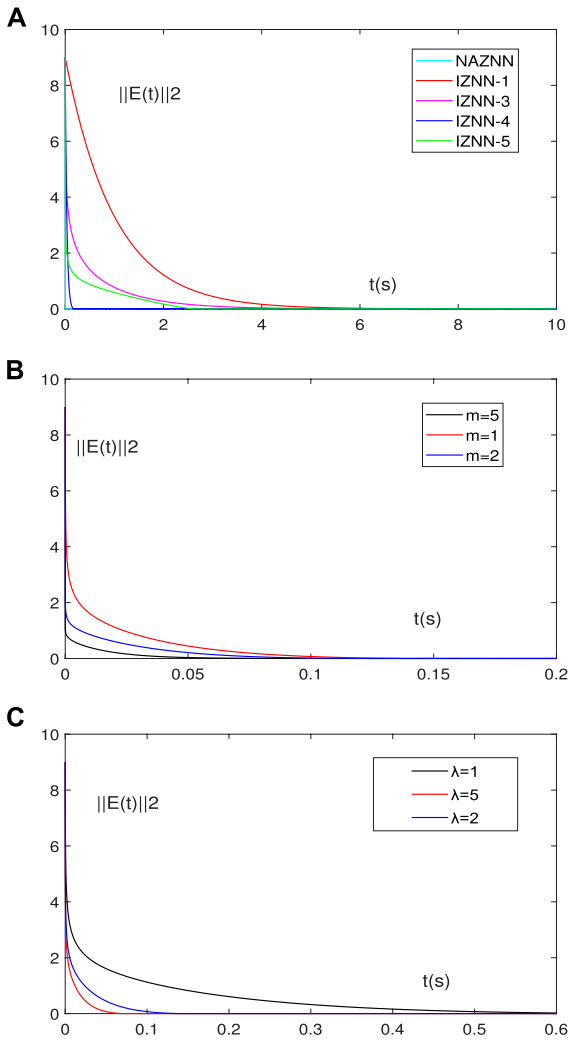


FIGURE 2
Comparative experiments of the ZNN models for solving Eq. 1. **(A)** is the residual errors of all the models, **(B)** is the residual errors of the NAZNN model with different parameter m, **(C)** is the residual errors of the NAZNN model with different parameter λ.

It is worth mentioning that the system environment in reality is complex and variable. Therefore, the NAZNN model (13) is used to solve inequality (1) with the different noises to highlight its robustness, and the simulation result are shown in Figure 3.

As can be seen from Figure 3A-D, even under various noise conditions, all the black curves still coincide exactly with the red dashed curves, while the other curves do not coincide quickly and accurately with the red dashed line. This indicates that only the transient solution of the NAZNN model quickly tracks the theoretical solution of inequality (1), while the transient solutions of other models cannot. Therefore, it can be concluded that the NAZNN model solves the time-varying linear Equation 1 perfectly in a predetermined time, and it has better convergence and greater robustness than the existing ZNN models. To further compare the errors of the NAZNN model (13) with the IZNN model for solving the inequality system (1) under different noise conditions, the residual errors of all the model are shown in Figure 4.

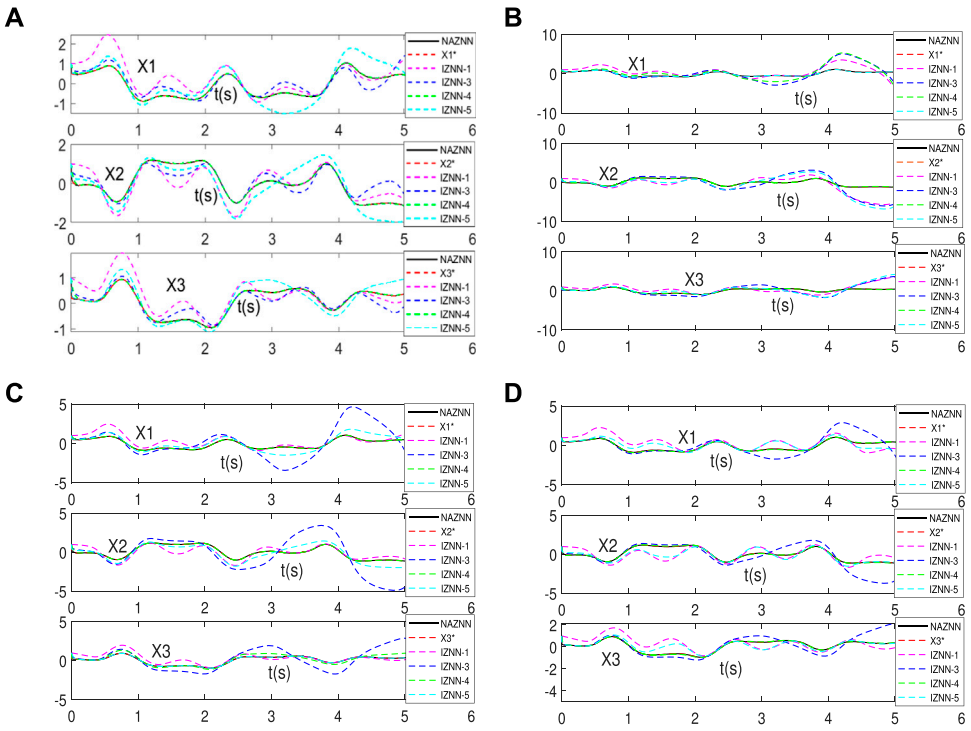


FIGURE 3
Neural state solution of NAZNN model and IZNN model for solving Eq.1 with different noises, **(A)** is add Y(t)=0.5, **(B)** is add Y(t)=0.5t, **(C)** is add Y(t) =cost, **(D)** is add exponential interference.

As can be seen in Figure 4A-D, the residual errors of the NAZNN model (13) drops to 0 in a very short time in the presence of four noises. However, the residual errors of the IZNN model fluctuate greatly or converge very slowly, or even fail to converge to 0 due to noises.

4 Applications on wheeled robots

In this section, the NAZNN model is applied to solve the wheeled robot trajectory tracking problems [50,54].

4.1 The construction of NAZNN model for wheeled robot trajectory tracking with physical constrains

4.1.1 Modeling of the physical constrained manipulator

In this subsection, the modeling of a movable six-joint three-wheel manipulator is introduced and its 3D model is shown in Figure 5A.

The equations of motion are analyzed according to the wheeled robot mobile device in Figure 5B, and the parameters are shown below.

W_0: the midpoint of the drive axis, expressed in the world coordinate system as (x_0, y_0, z_0).
W_s: the position of the manipulator, expressed in the world coordinate system (x_s, y_s, z_s).
d: distance from W_0 to W_s, $d = 0.1m$.
b: distance from the midpoint of the drive axis to the left and right drive wheels, $b = 0.32m$.
r: radius of each driving wheel, $r = 0.1025m$.
θ: The heading angle of the mobile device, expressed as the angle of the mobile device from the X-axis to the symmetry axis; its time derivative is the heading speed.
P: the mobile device rotates around the point P.
R: the distance of point P from the left driving wheel.
ω: the rotation speed of the mobile device around point P; and $\omega = \dot{\theta}$.

$\dot{\beta}_1$ and $\dot{\beta}_r$: rotation speed of the left and right wheels.

The next step is to establish robot model of the mobile device. Assume that each link of the wheeled robot is rigid and there is no relative sliding between the mobile device and the robot arm. Based on Figure 5B, the kinematic equations for the mobile device are established as follows, and the detailed derivation can be found in Supplementary Appendix S1A.

$$\begin{cases} \dot{x}_s = \left(\frac{r}{2}\cos\theta + \frac{dr}{2b}\sin\theta\right)\dot{\beta}_1 + \left(\frac{r}{2}\cos\theta - \frac{dr}{2b}\sin\theta\right)\dot{\beta}_r \\ \dot{y}_s = \left(\frac{r}{2}\sin\theta - \frac{dr}{2b}\cos\theta\right)\dot{\beta}_1 + \left(\frac{r}{2}\sin\theta + \frac{dr}{2b}\cos\theta\right)\dot{\beta}_r \end{cases} \tag{36}$$

Rewrite Eq. 36 into matrix form

$$\begin{bmatrix} \dot{x}_s \\ \dot{y}_s \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} \frac{r}{2} & \frac{r}{2} \\ -\frac{dr}{2b} & \frac{dr}{2b} \end{bmatrix} \begin{bmatrix} \dot{\beta}_1 \\ \dot{\beta}_r \end{bmatrix} \tag{37}$$

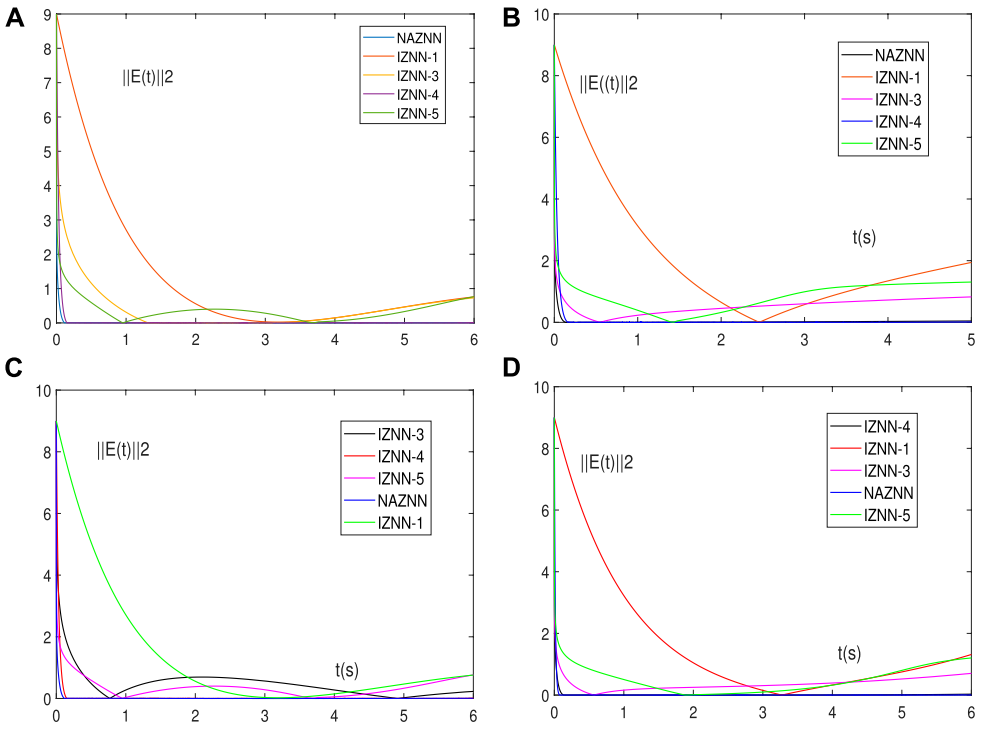


FIGURE 4
Residual errors of the NAZNN model and the IZNN models for solving Eq. 1 when disturbed by various noises. The noises considered in **(A-D)** are the same ones with **(A-D)**.

where $\dot{x}_s$ and $\dot{y}_s$ denote the x-axis velocity component and y-axis velocity of the point $\mathbf{W}_s$, respectively. Figure 5A shows a 3D model of a wheeled robot. The origin of the moving platform coordinate system (x_s, y_s, z_s) is defined at the point w_s in Figure 5B, while x_s is the positive direction of the moving device. Let (x_s, y_s, z_s) indicates the reference points in the coordinate system of the moving platform, and for computational convenience we set z_s to 0. Similarly, the coordinate system of the ith joint of the six-joint robot is defined as ($\mathbf{X}_i$, $\mathbf{Y}_i$, $\mathbf{Z}_i$), as shown in Figure 5C ($i = 1, 2, \ldots, 6$). The articulation of each joint with the previous joint is defined as the origin of each joint's coordinate system, and the direction along this joint is the Z-axis of this joint's coordinate system, which conforms to the left-hand rule to find the X-axis and Y-axis. The coordinate system of the end-effector ($\mathbf{X}_6$, $\mathbf{Y}_6$, $\mathbf{Z}_6$) position vector ${}^6\mathbf{W}_{end}$ belongs to R^m, and the homogeneity condition shows that ${}^6\mathbf{W}_{end} = [{}^6\mathbf{W}_{end}{}^T, 1]^T$ belongs to R^{m+1}. (The superscript T denotes the transpose of the vector.) Based on the platform coordinate system (x_s, y_s, z_s), we can represent the end-effector vector as follows.

$$ {}^s\mathbf{W}_{end} = {}^s_1\mathbf{T}\,{}^1_2\mathbf{T}\,{}^2_3\mathbf{T}\,{}^3_4\mathbf{T}\,{}^4_5\mathbf{T}\,{}^5_6\mathbf{T}\,{}^6\mathbf{W}_{end} \tag{38} $$

where ${}^s_1\mathbf{T}$, ${}^1_2\mathbf{T}$, ${}^2_3\mathbf{T}$, ${}^3_4\mathbf{T}$, ${}^4_5\mathbf{T}$, ${}^5_6\mathbf{T}$ denote the flush transformation matrices. The position coordinate vector of the end-effector in world coordinates can be expressed as

$$ \begin{bmatrix} \delta_z & 1 \end{bmatrix}^T = {}^z_s\mathbf{T}\,{}^s\mathbf{W}_{end} \tag{39} $$

where $\delta_z \in R^n$ indicates the position relative to the world coordinate system, and ${}^z_s\mathbf{T} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & x_s \\ \sin\theta & \cos\theta & 0 & y_s \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$.

According to the above analysis, the coordinates of the end-effector in the world coordinate system can be expressed as (x_z, y_z, z_z), and the kinematic relations of the end-effector can be obtained as follows:

$$ \delta_z = \begin{bmatrix} x_z & y_z & z_z \end{bmatrix}^T = \begin{bmatrix} x_s & y_s & 0 \end{bmatrix}^T + \delta_s \tag{40} $$

where δ_s is a three-dimensional vector, and the detailed expressions and calculation procedure for δ_s are shown in Supplementary Appendix S1B. Besides, Eq. 40 can be obtained by the time derivative of the speed level equation.

$$ \dot{\delta}_z = \begin{bmatrix} \dot{x}_z & \dot{y}_z & \dot{z}_z \end{bmatrix}^T = \begin{bmatrix} \dot{x}_s & \dot{y}_s & \dot{z}_s \end{bmatrix}^T + \dot{\delta}_s \tag{41} $$

It can be reformulated as:

$$ \dot{\delta}_z = \mathbf{J}_z\dot{\phi} \tag{42} $$

where $\dot{\phi} = [\dot{\theta}_1, \dot{\theta}_r, \dot{\gamma}_1, \dot{\gamma}_2, \dot{\gamma}_3, \dot{\gamma}_4, \dot{\gamma}_5, \dot{\gamma}_6]$. In three-dimensional space, the position of the end-effector is calculated with $n = 3$ (n is the number of variables), and the position and direction of the end-effector is calculated with $n = 6$. To control the wheeled robot, we set $n = 8$ (including two rotation variables and six joint variables). $\dot{\theta}_1$ and $\dot{\theta}_r$ denote the rotational speed of the left and right drive wheels, and $\dot{\gamma}_i$ ($i = 1, 2, \ldots, 6$) denotes the joint velocity vector of the robotic arm. The combined velocity vector $\dot{\phi}$ is a derivation of the combined angular vector ϕ with respect to t. In addition, $\mathbf{J}_z \in R^{m\times n}$ denotes a generalized Jacobi matrix and $\mathbf{J}_z \in R^{m\times n}$. The detailed expressions and derivations are given in Supplementary Appendix S1C. The physical constraints of the robot are shown in Table 2.

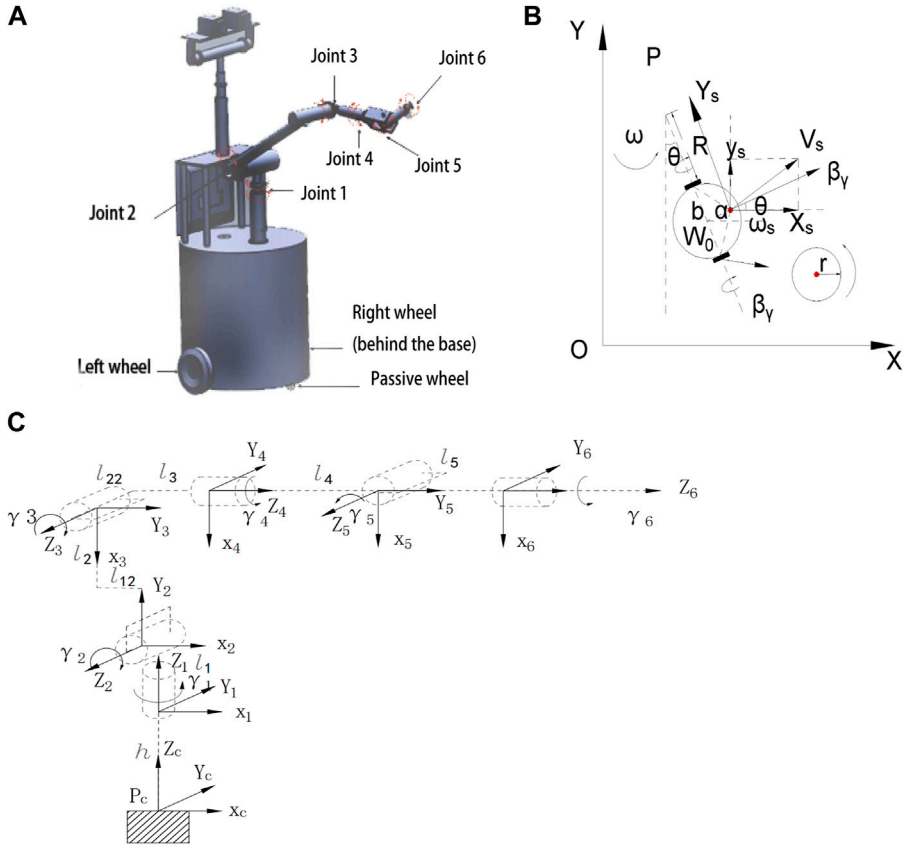


FIGURE 5
Wheeled robot modeling process. **(A)** is 3D model, **(B)** is the model of mobile platform, **(C)** is Wheeled robot world coordinate system and axis coordinate system.

Conventional control approaches have difficulty in handling trajectory tracking with constraints, and physical constraints are rarely considered before. However, almost all controllers have physical constraints (angular constraints, speed constraints, etc.). Therefore, it is very realistic and necessary to consider the failure problem of trajectory tracking of wheeled robots subject to constraints. The physical constraints of wheeled robot trajectory tracking considered in this work are listed below.

$$\begin{cases} \mathbf{J}_z\dot{\phi} = \dot{\delta}_{zd} + \nu(\delta_{zd} - \delta_z) \\ \phi^- \le \phi \le \phi^+ \\ \dot{\phi}^- \le \phi \le \dot{\phi}^+ \end{cases} \tag{43}$$

where $\nu > 0$ is the feedback gain, $\dot{\delta}_{zd}$ and δ_{zd} denote the position vector and velocity of the desired end position of the tracking path. ϕ^- and ϕ^+ denote the lower and upper limits of the joint angle ϕ, $\dot{\phi}^-$ and $\dot{\phi}^+$ denote the lower and upper limits of the joint velocity $\dot{\phi}$, respectively. For the range of motion of the wheeled robot, we theoretically set the upper and lower limits of ϕ_1 and ϕ_2 to converge to infinity.

To deal with the problem at the joint velocities level, the range of active wheel rotation angles of the wheeled robot and the range of robot joint angles are transformed into the corresponding velocities as follows.

$$\eta(\phi^- - \phi) \le \phi \le \eta(\phi^+ - \phi) \tag{44}$$

where the coefficient $\eta > 0$ is used to calculate the range of joint velocities, and the coefficient has η special points. The value of η may lead to a sudden deceleration of the joint velocity when the wheeled robot approaches the joint limit, in numerical terms, $\eta \ge 2\max_{1\le i\le n}\{\dot{\phi}_i^+/(\phi_i^+ - \phi_i^-), -\dot{\phi}_i^-/(\phi_i^+ - \phi_i^-)\}$. By calculating and setting the next simulation experiment in $\eta = 4$; the constraints of angle and velocity can be combined into constraint $\psi^- \le \dot{\phi} \le \psi^+$, where the κ th element of ψ^- and ψ^+ can be expressed as

$$\begin{cases} \psi^- = \max\{\phi_\kappa^- \eta(\phi^- - \phi)\} \\ \psi^+ = \min\{\phi_\kappa^+ \eta(\phi^+ - \phi)\} \end{cases} \tag{45}$$

In summary, the velocity control of wheeled robot trajectory tracking with physical constraints can be formulated as

$$\begin{cases} \mathbf{J}\mathbf{q} = \dot{\delta}, \\ \psi^- \le \mathbf{q} \le \psi^+, \end{cases} \tag{46}$$

where $\mathbf{q} \in R^m$ denotes the combined velocity vector $\dot{\phi}$, $\mathbf{J} = \mathbf{J}_z$, and the equation restriction represents the linear relationship between the velocities. Besides, $\dot{\delta} = \dot{\delta}_{zd} + \nu(\delta_{zd} - \delta_z) \in R^m$, where $\nu > 0$ is the feedback gain. $\dot{\delta}_{zd}$ and δ_{zd} denote the position vector and velocity at the desired end position of the tracking path. ψ denotes the range of active wheel rotation angles of the wheeled robot and the range of robot joint angles transformed into the corresponding velocities. ψ^+ and ψ^- represent the upper and lower bounds of the transformed

TABLE 2 Physical constraints of the robotic arm.

Joint angle	Upper limit (rad)	Lower limit (rad)
Left wheel angle	+∞	−∞
Right wheel angle	+∞	−∞
Joint angle 1	+ π/5	−π/5
Joint angle 2	+π/9	−π/3
Joint angle 3	+π/2	−π/6
Joint angle 4	+π/6	−π/9
Joint angle 5	+π/5	−π/3
Joint angle 6	+π/9	−π/9
velocity	**Upper limit (rad/s)**	**Lower limit (rad/s)**
Rotation speed	+90	−90
Rotation speed	+90	−90
Joint angular velocity 1	+3	−3
Joint angular velocity 2	+3	−3
Joint angular velocity 3	+3	−3
Joint angular velocity 4	+3	−3
Joint angular velocity 5	+3	−3
Joint angular velocity 6	+3	−3

velocities, respectively. In order to keep the joint variables within the set range, they can be constrained with restriction conditions.

4.1.2 NAZNN model for trajectory tracking control of wheeled robots

Equation 46 with the above conditions can be transformed as

$$\begin{cases} \mathbf{Jq} = \dot{\delta} \\ \mathbf{q} \le \psi^{+} \\ \mathbf{q} \ge \psi^{-} \end{cases} \tag{47}$$

Convert (47) to matrix form by referring to the CTVLMA calculation process.

$$\begin{bmatrix} \mathbf{J} & 0 & 0 \\ 1 & \mathbf{D_1}(t) & 0 \\ 1 & 0 & -\mathbf{E_1}(t) \end{bmatrix} \begin{bmatrix} \mathbf{q} \\ \mathbf{y}(t) \\ \mathbf{o}(t) \end{bmatrix} = \begin{bmatrix} \dot{\delta} \\ \psi^{+} \\ \psi^{-} \end{bmatrix} \tag{48}$$

$\mathbf{D_1}(t)$, $\mathbf{E_1}(t)$, $\mathbf{y}(t)$ and $\mathbf{o}(t)$ are process quantities generated by the calculation process with reference to CTVLMA. Let $\mathbf{M}(t) = \begin{bmatrix} J & 0 & 0 \\ 1 & \mathbf{D_1}(t) & 0 \\ 1 & 0 & -\mathbf{E_1}(t) \end{bmatrix} \in R^{(m+p)\times(n+p)}$ and $\mathbf{N}(t) = [\dot{\delta}; \psi^{+}; \psi^{-}]^{T} \in R^{(m+p)}$.

Then, define $\mathbf{u}(t) = [\mathbf{q} \quad \mathbf{y}(t) \quad \mathbf{o}(t)]^{T}$, the following matrix vector form of (45) is obtained

$$\mathbf{M}(t)\mathbf{u}(t) = \mathbf{N}(t) \tag{49}$$

The NAZNN model for trajectory tracking control of wheeled robots with physical constraints is obtained.

$$\mathbf{M}(t)\dot{\mathbf{u}}(t) = -\dot{\mathbf{M}}(t)\mathbf{u}(t) + \dot{\mathbf{N}}(t) - \lambda\mathbf{F}(\mathbf{M}(t)\mathbf{u}(t) - \mathbf{N}(t)) \tag{50}$$

4.2 Simulation results on physically constrained wheeled robots

In this section, the above NAZNN model (50) and other related ZNN models are all used to solve the same wheeled robotic arm trajectory tracking problem with physical constraints for the purpose of comparison.

The initial heading angle of the wheeled robot is set to zero, and the motion direction of the wheeled robot is parallel to the x-axis of the world coordinate system, and its initial position is $(x_s(0), y_s(0), z_s(0)) = (0, 0, 0)$. The initial state of the robot arm joint is $[\pi/5; \pi/3; \pi/6; \pi/3; \pi/6; \pi/3]^{T}$, and the rotation angle of the drive wheel set to $\beta_1 = \beta_r = 0rad$. In summary, the initial value of the combined angular vector is $\phi(0) = [0,0; \pi/5; \pi/3; \pi/6; \pi/3; \pi/6; \pi/3]^{T}$.

4.2.1 NAZNN model simulation results

In this subsection, the proposed NAZNN model is applied to control the wheeled robot to perform trajectory tracking, and the corresponding simulation results are represented in Figure 6. As observed in Figure 6, the wheeled robot controlled by the NAZNN model completes the tracking task effectively.

Additionally, the joint angle variation of the robot for spiral trajectory tracking is represented in Figure 7, and the joint velocity of the robot for the tracking process is represented in Figure 8.

It can be observed in Figures 7, 8 that the joint angles and joint speeds of the NAZNN model-controlled wheeled robotic arm are within the physical limits until the end of the trajectory tracking, which indicates that the NAZNN model-controlled wheeled robotic arm successfully completes the tracking task.

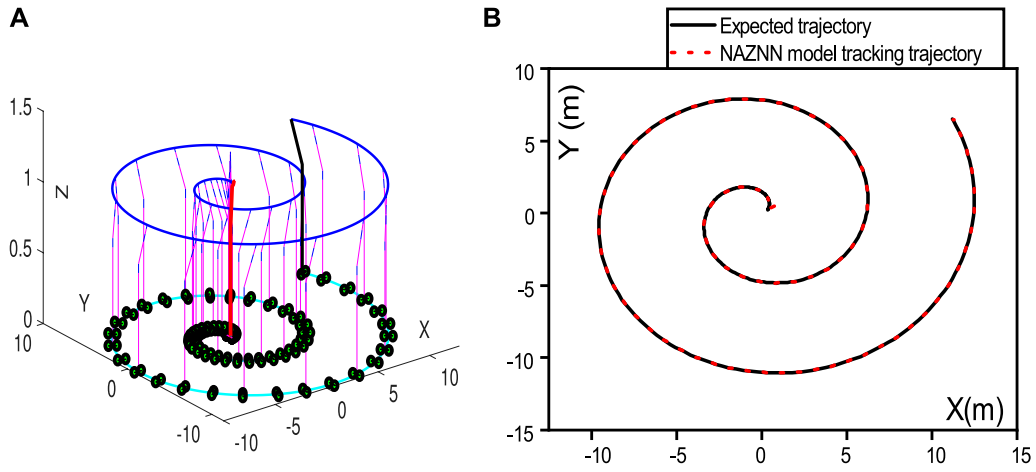


FIGURE 6
NAZNN model-controlled wheeled robot trajectory and the end-effector trajectory. **(A)** is Robot operation trajectory, **(B)** is Robotic arm end-effector trajectory.

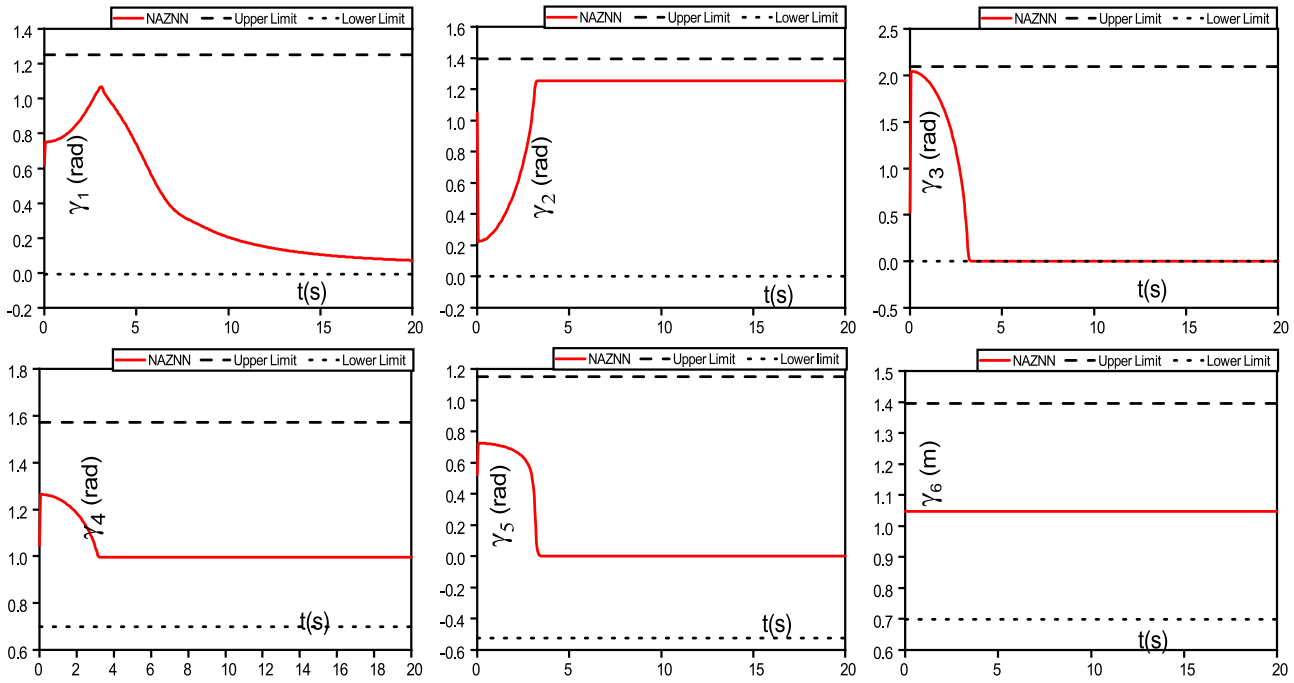


FIGURE 7
The NAZNN model-controlled wheeled robots trajectory tracking with 6 joint angle changes.

4.2.2 IZNN models simulation results

In this subsection, the IZNN models controlled robots are also used for the same tracking task for the purpose of comparison, and the corresponding simulation results are represented in Figures 9–12.

Figure 9 shows the difference between the IZNN-1 and IZNN-2 models. Both the IZNN-1 and IZNN-2 models are activated by the linear activation function. Because the parameter size has a great impact on the models, the actual experimental results are also different.

As observed in Figure 9, the IZNN-1 model controlled robot fails the tracking task due to robot joint angle $\dot{\lambda}_1$ exceeds limitation; besides, the IZNN-2 model controlled robot also fails the tracking task due to robot joint speed $\dot{\lambda}_5$ exceeds limitation. Figure 9A shows that joint angle 1 of the IZNN-1 model exceeds the upper limit at 3.6 s, and it lead to the trajectory tracking task fail; Figure 9B shows that joint velocity 5 of the IZNN-2 model falls below the lower limit at 3.6 s, and it lead to the trajectory tracking task pause.

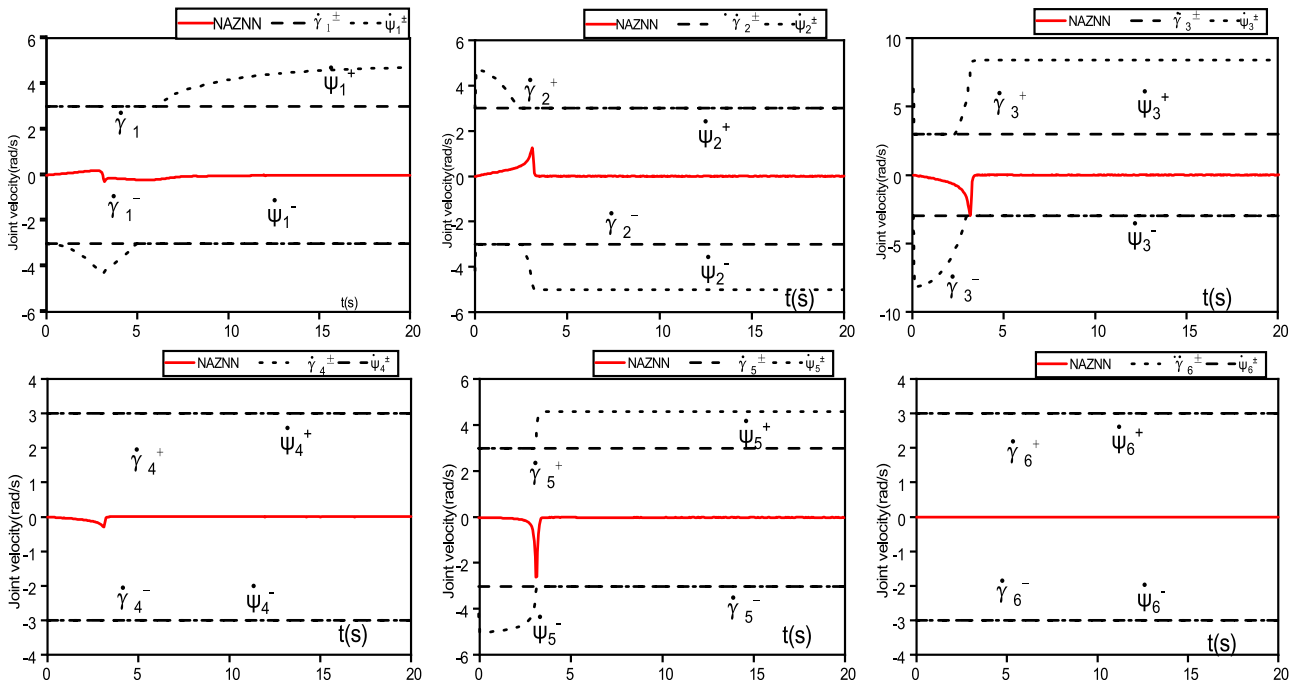


FIGURE 8
The NAZNN model-controlled wheeled robot successfully finishes the trajectory tracking with 6 joint velocities change.

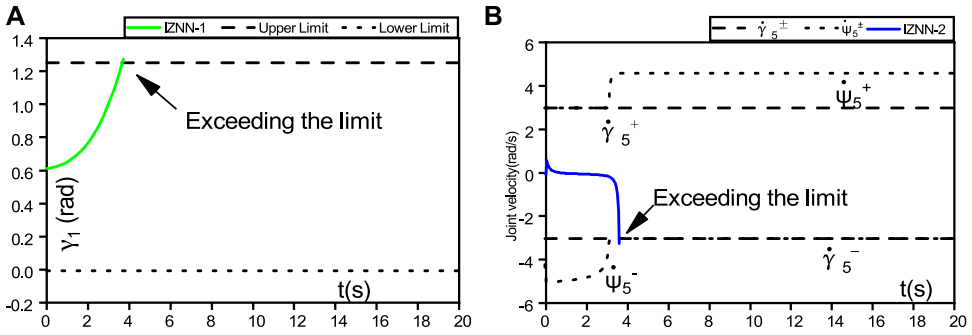


FIGURE 9
(A) represent joint angle fault for IZNN-1, **(B)** represent joint speed fault for IZNN-2.

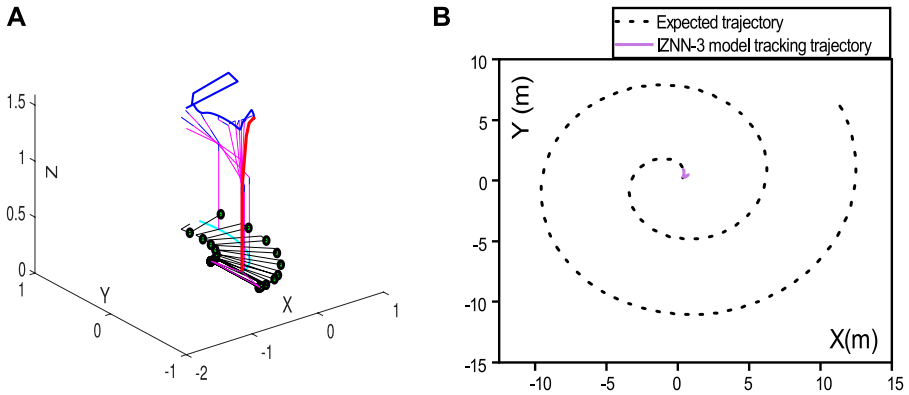


FIGURE 10
(A) represent IZNN-3 model-controlled wheeled robot's joint trajectory, **(B)** represent the end-effector trajectory.

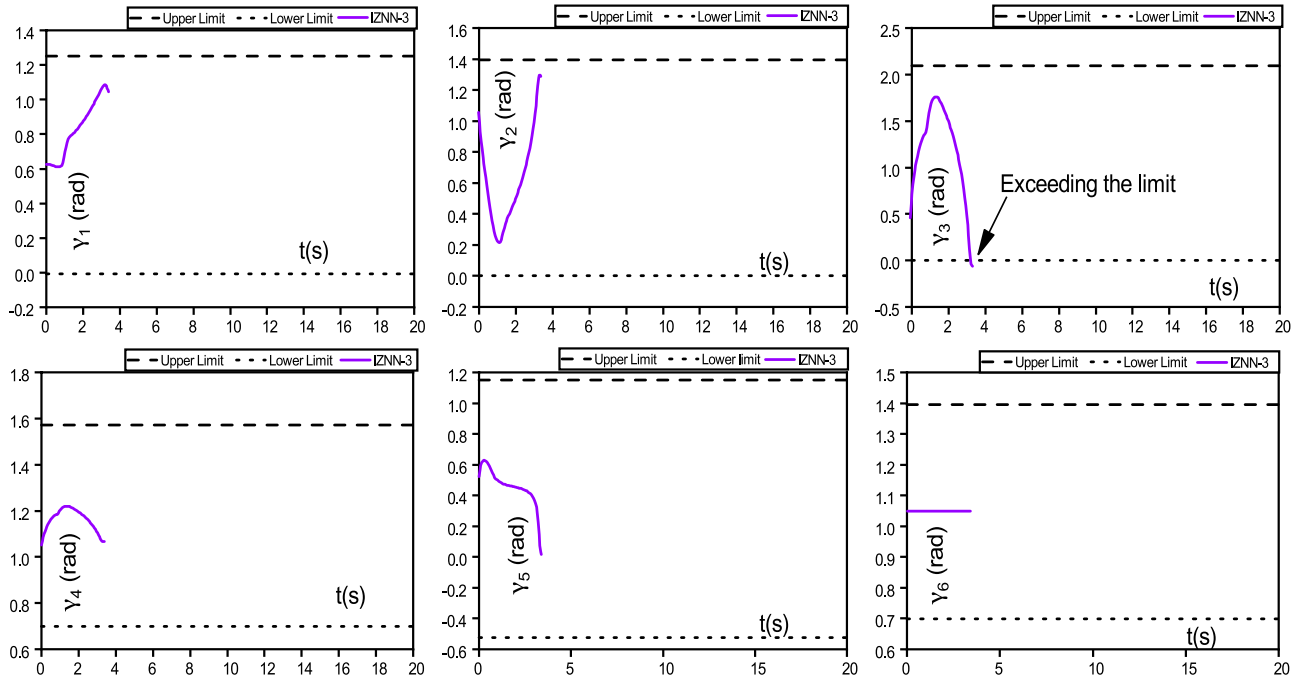


FIGURE 11
The IZNN-3 model-controlled wheeled robot fails trajectory tracking with joint angles exceed limits.

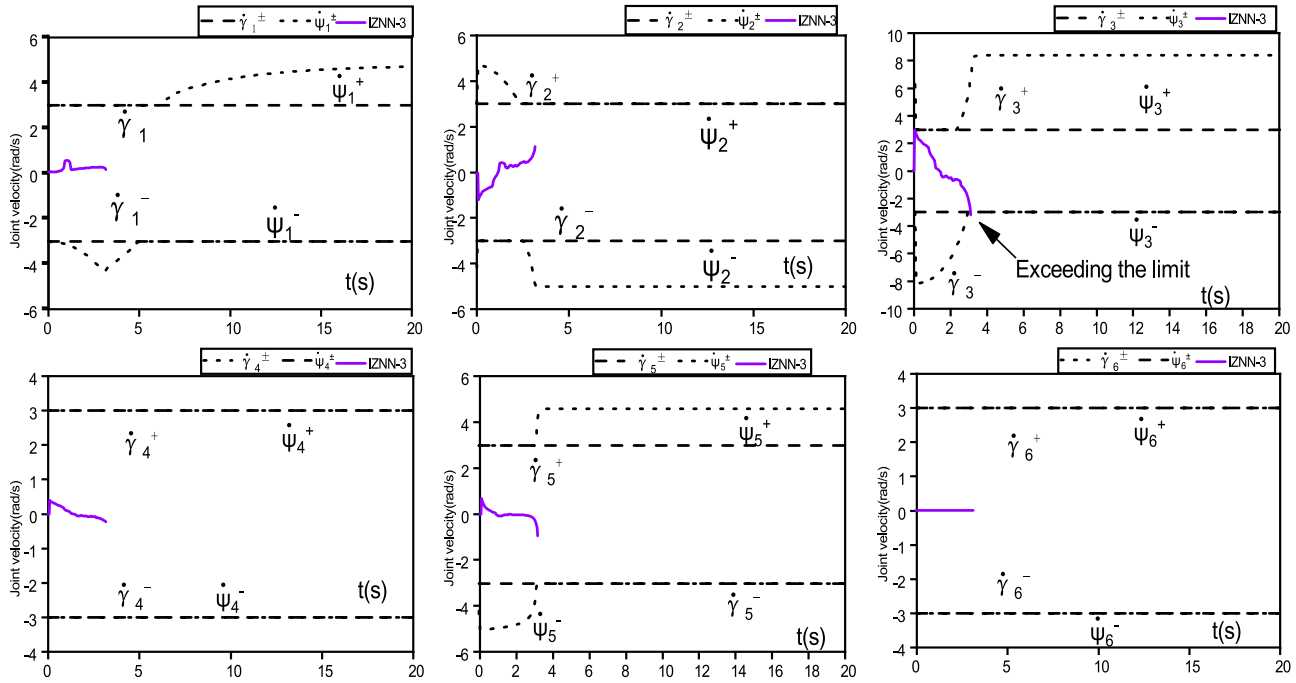


FIGURE 12
The IZNN-3 model-controlled wheeled robot fails trajectory tracking with Joint velocities exceed limits.

The wheeled robot controlled by the IZNN-3 model for the same tracking task is presented in Figures 10–12. As observed in Figure 10, the wheeled robot controlled by the IZNN-3 model also fails the tracking task. Additionally, the joint angle variation of the robot controlled by the IZNN-3 model for spiral trajectory tracking is represented in Figure 11, and the joint velocity of the robot

TABLE 3 Comparisons of ZNN models for trajectory tracking results.

Model	AFs	Stop time (s)	Reason for stopping
IZNN-1	LAF	3.6	Joint angle (λ_1) exceeds limitation
IZNN-2	LAF	3.6	Joint speed ($\dot{\lambda}_5$) exceeds limitation
IZNN-3	PSAF	3.3	Joint angle (λ_3) and joint speed ($\dot{\lambda}_3$) exceed the limitation at the same time
NAZNN	NAF	20	None

controlled by the IZNN-3 model for the tracking process is represented in Figure 12.

As observed in Figure 10, Figures 11, 12, the tracking task is interrupted, with the joint speed ($\dot{\lambda}_3$) exceeds the physical limit (3 rad/s) and the joint angle (λ_3) exceeds the physical limit (exceeded the lower limit - π/6). Speed 3 ($\dot{\lambda}_3$) in Figure 10 exceeds the limit at 3.3 s and angle 3 (λ_3) in Figure 11 exceeds the limit at 3.3 s, which stop the tracking task. The above analysis indicates that the wheeled robotic arm controlled by the IZNN-3 model does not complete the tracking task due to exceediton of the joint speed and joint angle constrains.

In summary, the proposed NAZNN model has better control performances compared to other models for physically constrained wheeled robot trajectory tracking, and it completes the physically constrained trajectory tracking task smoothly and accurately without exceeding the joint angle and joint speed. The detailed comparative results of the models are presented in Table 3.

5 Conclusion

In this paper, a NAZNN model is proposed and effective applied to solve CTVLME problems. It is theoretically demonstrated that the NAZNN model can obtain the exact solutions of CTVLME problems. The validity and superiority of the NAZNN model is further verified by two numerical examples. Besides, the proposed NAZNN model is also applied to the failure problem of trajectory tracking of wheeled robots with physical constraints. Finally, the NAZNN model is used to control the wheeled robot to complete trajectory tracking under restricted conditions to prove the feasibility of this control method.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding authors.

Author contributions

All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

Funding

This work was supported by the National Natural Science Foundation of China (62273141), Natural Science Foundation of Hunan Province (2020JJ4315), Scientific Research Fund of Hunan Provincial Education Department (20B216).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Supplementary material

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fphy.2023.1159212/full#supplementary-material

References

1. Katsikis VN, Mourtas SD, Stanimirović PS, Zhang Y. Solving complex-valued time-varying linear matrix equations via qr decomposition with applications to robotic motion tracking and on angle-of-arrival localization. *IEEE Trans Neural Networks Learn Syst* (2021) 33(8):3415–24. doi:10.1109/tnnls.2021.3052896

2. Kayastha S, Katupitiya J, Pearce G, Rao A. Comparative study of post-impact motion control of a flexible arm space robot. *Eur J Control* (2022) 69:100738. doi:10.1016/j.ejcon.2022.100738

3. Yang S, Zhang Y, Wen H, Jin D. Coordinated control of dual-arm robot on space structure for capturing space targets. *Adv Space Res* (2023) 71(5):2437–48. doi:10.1016/j.asr.2022.10.027

4. Park FC, Kim B, Jang C, Hong J. Geometric algorithms for robot dynamics: A tutorial review. *Appl Mech Rev* (2018) 70. doi:10.1115/1.4039078

5. Yang B, Yang W. Modular approach to kinematic reliability analysis of industrial robots. *Reliab Eng Syst Saf* (2023) 229:108841. doi:10.1016/j.ress.2022.108841

6. Gyebrószki G, Csernák G, Milton JG, Insperger T. The effects of sensory quantization and control torque saturation on human balance control. *Chaos: Interdiscip J Nonlinear Sci* (2021) 31:033145. doi:10.1063/5.0028197

7. Khan AT, Li S, Li Z. Obstacle avoidance and model-free tracking control for home automation using bio-inspired approach. *Adv Control Appl Eng Ind Syst* (2022) 4(1):e63. doi:10.1002/adc2.63

8. Li ZF, Li JT, Li XF, Yang YJ, Xiao J, Xu BW. Intelligent tracking obstacle avoidance wheel robot based on arduino. *Proced Comput Sci* (2020) 166:274–8. doi:10.1016/j.procs.2020.02.100

9. Liu D, Wang J, Lei T, Wang S. Active suspension control with consensus strategy for dynamic posture tracking of wheel-legged robotic systems on uneven surfaces. *ISA Trans* (2022) 131:628–38. doi:10.1016/j.isatra.2022.05.021

10. Carlucho I, De Paula M, Acosta GG. An adaptive deep reinforcement learning approach for mimo pid control of mobile robots. *ISA Trans* (2020) 102:280–94. doi:10.1016/j.isatra.2020.02.017

11. Singhal K, Kumar V, Rana K. Robust trajectory tracking control of non-holonomic wheeled mobile robots using an adaptive fractional order parallel fuzzy pid controller. *J Franklin Inst* (2022) 359(9):4160–215. doi:10.1016/j.jfranklin.2022.03.043

12. Li Y, Li Y-X, Tong S. Event-based finite-time control for nonlinear multi-agent systems with asymptotic tracking. *IEEE Trans Autom Control* (2022) 1–8. doi:10.1109/tac.2022.3197562

13. Rosas-Vilchis A, de Loza AF, Aguilar LT, Cieslak J, Henry D, Montiel-Ross O. Trajectory tracking control for an autonomo vehicle using a decoupling approach. In: 2020 28th Mediterranean Conference on Control and Automation (MED). IEEE (2020). p. 375–80.

14. Lu Q, Chen J, Wang Q, Zhang D, Sun M, Su C-Y. Practical fixed-time trajectory tracking control of constrained wheeled mobile robots with kinematic disturbances. *ISA Trans* (2022) 129:273–86. doi:10.1016/j.isatra.2021.12.039

15. Yu F, Shen H, Liu L, Zhang Z, Huang Y, He B, et al. Ccii and fpga realization: A multistable modified fourth-order autonomous chua's chaotic system with coexisting multiple attractors. *Complexity* (2020) 2020:1–17. doi:10.1155/2020/5212601

16. Pan Y, Wu Y, Lam H-K. Security-based fuzzy control for nonlinear networked control systems with dos attacks via a resilient event-triggered scheme. *IEEE Trans Fuzzy Syst* (2022) 30(10):4359–68. doi:10.1109/tfuzz.2022.3148875

17. Dai J, Chen Y, Xiao L, Jia L, He Y. Design and analysis of a hybrid gnn-znn model with a fuzzy adaptive factor for matrix inversion. *IEEE Trans Ind Inform* (2021) 18(4):2434–42. doi:10.1109/tii.2021.3093115

18. Katsikis VN, Stanimirović PS, Mourtas SD, Xiao L, Karabašević D, Stanujkić D. Zeroing neural network with fuzzy parameter for computing pseudoinverse of arbitrary matrix. *IEEE Trans Fuzzy Syst* (2021) 30(9):3426–35. doi:10.1109/tfuzz.2021.3115969

19. Kong Y, Chen S, Jiang Y, Wang H, Chen H. Zeroing neural network with fuzzy parameter for cooperative manner of multiple redundant manipulators. *Expert Syst Appl* (2023) 212:118735. doi:10.1016/j.eswa.2022.118735

20. Gong J, Jin J. A better robustness and fast convergence zeroing neural network for solving dynamic nonlinear equations. *Neural Comput Appl* (2021) 35:77–87. doi:10.1007/s00521-020-05617-9

21. Xiao L, Liu S, Wang X, He Y, Jia L, Xu Y. Zeroing neural networks for dynamic quaternion-valued matrix inversion. *IEEE Trans Ind Inform* (2021) 18(3):1562–71. doi:10.1109/tii.2021.3090063

22. Yu F, Yu Q, Chen H, Kong X, Mokbel AAM, Cai S, et al. Dynamic analysis and audio encryption application in iot of a multi-scroll fractional-order memristive hopfield neural network. *Fractal and Fractional* (2022) 6(7):370. doi:10.3390/fractalfract6070370

23. Barajas CA, Gobbert M, Polf JC. (2021). Deep residual fully connected neural network classification of compton camera based prompt gamma imaging for proton radiotherapy. UMBC Student Collection.

24. Jin J, Zhao L, Chen L, Chen W. A robust zeroing neural network and its applications to dynamic complex matrix equation solving and robotic manipulator trajectory tracking. *Front Neurorobotics* (2022) 16:1065256. doi:10.3389/fnbot.2022.1065256

25. Jin J, Zhu J, Zhao L, Chen L. A fixed-time convergent and noise-tolerant zeroing neural network for online solution of time-varying matrix inversion. *Appl Soft Comput* (2022) 130:109691. doi:10.1016/j.asoc.2022.109691

26. He B, Dai C, Lang J, Bing P, Tian G, Wang B, et al. A machine learning framework to trace tumor tissue-of-origin of 13 types of cancer based on dna somatic mutation. *Biochim Biophys Acta (BBA)-Molecular Basis Dis* (2020) 1866(11):165916. doi:10.1016/j.bbadis.2020.165916

27. He B, Zhang Y, Zhou Z, Wang B, Liang Y, Lang J, et al. A neural network framework for predicting the tissue-of-origin of 15 common cancer types based on rna-seq data. *Front Bioeng Biotechnol* (2020) 8:737. doi:10.3389/fbioe.2020.00737

28. Jin J, Gong J. A noise-tolerant fast convergence znn for dynamic matrix inversion. *Int J Comput Math* (2021) 98(11):2202–19. doi:10.1080/00207160.2021.1881498

29. Jin J, Xiao L, Lu M, Li J. Design and analysis of two ftrnn models with application to time-varying sylvester equation. *IEEE Access* (2019) 7:58945–50. doi:10.1109/access.2019.2911130

30. Ma M, Xiong K, Li Z, Sun Y. Dynamic behavior analysis and synchronization of memristor-coupled heterogeneous discrete neural networks. *Mathematics* (2023) 11(2):375. doi:10.3390/math11020375

31. Jin J, Chen W, Chen C, Chen L, Tang Z, Chen L, et al. A predefined fixed-time convergence znn and its applications to time-varying quadratic programming solving and dual-arm manipulator cooperative trajectory tracking. *IEEE Trans Ind Inform* (2022) 1–12. doi:10.1109/tii.2022.3220873

32. Jin J, Chen W, Zhao L, Chen L, Tang Z. A nonlinear zeroing neural network and its applications on time-varying linear matrix equations solving, electronic circuit currents computing and robotic manipulator trajectory tracking. *Comput Appl Math* (2022) 41(7):319. doi:10.1007/s40314-022-02031-w

33. Jin J, Qiu L. A robust fast convergence zeroing neural network and its applications to dynamic sylvester equation solving and robot trajectory tracking. *J Franklin Inst* (2022) 359(7):3183–209. doi:10.1016/j.jfranklin.2022.02.022

34. Ma M, Lu Y, Li Z, Sun Y, Wang C. Multistability and phase synchronization of rulkov neurons coupled with a locally active discrete memristor. *Fractal Fractional* (2023) 7(1):82. doi:10.3390/fractalfract7010082

35. Jin J, Chen W, Qiu L, Zhu J, Liu H. A noise tolerant parameter-variable zeroing neural network and its applications. *Math Comput Simul* (2023) 207:482–98. doi:10.1016/j.matcom.2023.01.012

36. Jin J, Zhu J, Zhao L, Chen L, Chen L, Gong J. A robust predefined-time convergence zeroing neural network for dynamic matrix inversion. *IEEE Trans Cybernetics* (2022) 1–14. doi:10.1109/tcyb.2022.3179312

37. Jin J, Gong J. An interference-tolerant fast convergence zeroing neural network for dynamic matrix inversion and its application to mobile manipulator path tracking. *Alexandria Eng J* (2021) 60(1):659–69. doi:10.1016/j.aej.2020.09.059

38. Hu Z, Xiao L, Dai J, Xu Y, Zuo Q, Liu C. A unified predefined-time convergent and robust znn model for constrained quadratic programming. *IEEE Trans Ind Inform* (2020) 17(3):1998–2010. doi:10.1109/tii.2020.2996215

39. Luo J, Yang H, Yuan L, Chen H, Wang X. Hyperbolic tangent variant-parameter robust znn schemes for solving time-varying control equations and tracking of mobile robot. *Neurocomputing* (2022) 510:218–32. doi:10.1016/j.neucom.2022.08.066

40. Yang Y, Zhang Y. Superior robustness of power-sum activation functions in zhang neural networks for time-varying quadratic programs perturbed with large implementation errors. *Neural Comput Appl* (2013) 22(1):175–85. doi:10.1007/s00521-011-0692-5

41. Li S, Chen S, Liu B. Accelerating a recurrent neural network to finite-time convergence for solving time-varying sylvester equation by using a sign-bi-power activation function. *Neural Process Lett* (2013) 37(2):189–205. doi:10.1007/s11063-012-9241-1

42. Xiao L. Accelerating a recurrent neural network to finite-time convergence using a new design formula and its application to time-varying matrix square root. *J Franklin Inst* (2017) 354(13):5667–77. doi:10.1016/j.jfranklin.2017.06.012

43. Zhu J, Jin J, Chen W, Gong J. A combined power activation function based convergent factor-variable znn model for solving dynamic matrix inversion. *Math Comput Simul* (2022) 197:291–307. doi:10.1016/j.matcom.2022.02.019

44. Xiao L, He Y, Liao B. A parameter-changing zeroing neural network for solving linear equations with superior fixed-time convergence. *Expert Syst Appl* (2022) 208:118086. doi:10.1016/j.eswa.2022.118086

45. Li J, Zhu X, Shi Y, Wang J, Guo H. Real-time robot manipulator tracking control as multilayered time-varying problem. *Appl Math Model* (2021) 96:355–66. doi:10.1016/j.apm.2021.03.021

46. He W, Xue C, Yu X, Li Z, Yang C. Admittance-based controller design for physical human–robot interaction in the constrained task space. *IEEE Trans Autom Sci Eng* (2020) 17(4):1937–49. doi:10.1109/tase.2020.2983225

47. Dou R, Yu S, Li W, Chen P, Xia P, Zhai F, et al. Inverse kinematics for a 7-dof humanoid robotic arm with joint limit and end pose coupling. *Mech Machine Theor* (2022) 169:104637. doi:10.1016/j.mechmachtheory.2021.104637

48. Aouiti C, Miaadi F. A new fixed-time stabilization approach for neural networks with time-varying delays. *Neural Comput Appl* (2020) 32(8):3295–309. doi:10.1007/s00521-019-04586-y

49. Tutsoy O, Polat A. Linear and non-linear dynamics of the epidemics: System identification based parametric prediction models for the pandemic outbreaks. *ISA Trans* (2022) 124:90–102. doi:10.1016/j.isatra.2021.08.008

50. Xing H, Torabi A, Ding L, Gao H, Li W, Tavakoli M. Enhancing kinematic accuracy of redundant wheeled mobile manipulators via adaptive motion planning. *Mechatronics* (2021) 79:102639. doi:10.1016/j.mechatronics.2021.102639

51. Wang D, Wei W, Wang X, Gao Y, Li Y, Yu Q, et al. Formation control of multiple mecanum-wheeled mobile robots with physical constraints and uncertainties. *Appl Intell* (2022) 52:2510–29. doi:10.1007/s10489-021-02459-3

52. Ding L, Xiao L, Zhou K, Lan Y, Zhang Y. A New RNN Model With a Modified Nonlinear Activation Function Applied to Complex–Valued Linear Equations. IEEE. *Access* (2018) 6:62954–62962. doi:10.1109/ACCESS.2018.2876665

53. Li W, Xiao L, Liao B. A Finite-time convergent and noise-rejection recurrent neural network and its discretization for dynamic nonlinear equations solving. IEEE. *Transactions on Cybernetics* (2020) 50(7):3195–3207. doi:10.1109/TCYB.2019.2906263

54. Jin L, Li S, Xiao L, Lu R, Liao B. Cooperative motion generation in a distributed network of redundant robot manipulators with noises. IEEE Transactions on Systems, Man, and Cybernetics: *Systems*, (2018) 48(10):1715–1724. doi:10.1109/TSMC.2017.693400

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 22 March 2023
DOI 10.3389/fphy.2023.1162887

OPEN ACCESS

EDITED BY
Viet-Thanh Pham,
Ton Duc Thang University, Vietnam

REVIEWED BY
Qiang Lai,
East China Jiaotong University, China
Jun Mou,
Dalian Polytechnic University, China

*CORRESPONDENCE
Wei Yao,
yaowei@csust.edu.cn

SPECIALTY SECTION
This article was submitted to Interdisciplinary Physics, a section of the journal Frontiers in Physics

RECEIVED 10 February 2023
ACCEPTED 03 March 2023
PUBLISHED 22 March 2023

CITATION
Yao W, Gao K, Zhang Z, Cui L and Zhang J (2023), An image encryption algorithm based on a 3D chaotic Hopfield neural network and random row–column permutation. *Front. Phys.* 11:1162887. doi: 10.3389/fphy.2023.1162887

An image encryption algorithm based on a 3D chaotic Hopfield neural network and random row–column permutation

Wei Yao[1,2]*, Kai Gao[2], Zhihao Zhang[2], Li Cui[3] and Jin Zhang[2]

[1]Engineering Laboratory of Spatial Information Technology of Highway Geological Disaster Early Warning in Hunan Province, Changsha University of Science and Technology, Changsha, Hunan, China, [2]School of Computer and Communication Engineering, Changsha University of Science and Technology, Changsha, China, [3]Hunan University of Science and Technology, Xiangtan, Hunan, China

This study proposes a novel color image encryption algorithm based on a 3D chaotic Hopfield neural network and random row–column permutation. First, a 3D chaotic Hopfield neural network is proposed to produce the random sequence for generating the diffusion and permutation keys. Then, the rows and columns of the original image are randomly arranged according to the permutation key in the permutation process. Three subgraphs are formed by separating the R, G, and B components of the color image in the diffusion process. Each of the three subgraphs is split along the columns to form three parts; the left and middle parts are exchanged. Three diffusion keys are used to encrypt each of the three parts. Finally, the individually encrypted subgraphs are stitched together to obtain the final encrypted image. Simulation results using MATLAB and FPGA and security analysis demonstrate that the encryption scheme has good performance.

KEYWORDS

image encryption, non-linear dynamics, diffusion and permutation, chaotic Hopfield neural network, random row–column permutation

1 Introduction

In recent years, communication technology has made significant progress. At the same time, the security of information distribution has been raised to a new level. Digital images are an important means of multimedia expression [1, 2], which are widely used in clinical medicine, astronomy, inspection, and other regions. In conclusion, image information transmission urgently needs a set of new, more stable, not-easy-to-be-cracked image encryption algorithms.

Chaos is a non-linear dynamical phenomenon that exists in a wide variety of natural fields [3–6], such as biology, meteorology, and economics. Interestingly, chaos is not a pure disorder but rather an ordered state that does not possess periodic changes and other notable symmetrical features. One distinctive feature of chaotic systems is that they are extremely sensitive to the initial values and parameters, and the dynamics and values of the system can vary considerably for different initial values of the same parameters. These characteristics of chaotic systems are well suited to the needs of image encryption algorithms [7, 8], and that is the reason why many researchers have applied chaotic systems to image encryption in recent years. Wang et al. [9] proposed a new image-encryption algorithm based on iterating chaotic maps. Using the pseudorandom sequence generated by a group of one-dimensional chaotic maps, Li et al. [10] used a 1-D chaotic tent map to generate a chaos-based key stream for

image encryption. Lai et al. [11] proposed a novel image encryption based on the 2D Salomon map.

Many image-encryption studies now combine chaotic systems with other methods, such as DNA sequences [12–15] and diffusion-permutation [16–18]. Enayatifar et al. [15]proposed a novel image-encryption algorithm based on the deoxyribonucleic acid (DNA) masking hybrid model, genetic algorithm (GA), and logical map. Chai et al. [19]designed an encryption algorithm based on a chaotic system and DNA sequence operations. Liu et al. proposed an image-encryption algorithm based on one-time keys and robust chaotic maps and designed a novel encryption algorithm based on the spatial bit-level permutation and high-dimension chaotic system in Refs. 20, 21, respectively. Chen et al. [22] proposed a complete cryptosystem, which is built by using Bake maps for image permutations, and Diab et al. [23] improved it.

In recent years, some research studies on artificial neural networks and their applications [24–29] have been widely discussed. With the creation of the first memristor [30], many researchers have used memristors to simulate synapses [31–33] between neurons in the human brain and to analyze the dynamical behavior [34–38] of artificial neuronal networks. The combination of chaotic systems and artificial neural networks has become a hot research topic nowadays [39, 40], and due to the nonlinear characteristic of the Hopfield neural network model, this model is capable of generating abundant chaotic behavior and is often used by researchers to simulate the various dynamic behaviors of neurons in the brain [41]. Using the Hopfield chaotic neural network model, a random sequence can be generated, and the more random the generated random sequence, the better the encryption of the image. There are many works in the field of random number generation using chaotic models [42, 43]. Wang et al. [44] proposed a novel encryption algorithm based on a new fractional-order chaotic system.

Before our work, Wang et al. [45] proposed a new color image encryption, which uses Hopfield chaotic neural networks to generate the self-diffusion chaotic matrix. Chen et al. [46] proposed a three-dimensional fractional-order discrete Hopfield neural network. Wu et al. [47] applied the Hopfield chaotic neural network together with the novel hyperchaotic system to propose a new color image encryption algorithm. The purpose of this study is to investigate a simple but efficient image-encryption algorithm based on a chaotic Hopfield neural network model.

This study proposed an image encryption based on a 3D chaotic Hopfield neural network and random row–column permutation. In this diffusion process, we separate the RGB components of the color image. Each component is split into three equal parts along the columns and then the middle and left parts are swapped. Three different random sequences are obtained by the proposed chaotic Hopfield neural network to encrypt the three parts of each component. Finally, the R, G, and B components of the ciphertext image are combined into the ciphertext image. Most of the previous image-encryption algorithms [48–51] either encrypted the RGB components using the same set of random sequences or encrypted the RGB using three different sets of random sequences separately. The association of each element of RGB in the relative position of the image is ignored, which would make the image easier to crack. Therefore, this study splits the RGB subgraphs separately and then uses different encryption sequences for each part of the subgraph. The experimental results are obtained by using MATLAB and FPGA. The extensive security analysis shows that the proposed algorithm improves encryption efficiency and has good security performance.

The rest of this paper is organized as follows. Section 2 describes the Hopfield neural network system. Section 3 presents the image-encryption and -decryption algorithm. Section 4 shows the simulation result of the image encryption and decryption. Section 5 analyzes the safety of this algorithm. Section 6 concludes this paper.

2 Hopfield neural network systems

The Hopfield neural network was proposed by the American physicist J. Hopfield [52]. In this study, the Hopfield neural network model of three neurons with self-feedback is adopted. We can see that the three neurons have the ability to connect together and influence each other. The Hopfield neural network model is shown as follows:

$$\dot{x}_i(i) = -x_i(i) + \sum_{j=1}^{n} w_{ij} U_i, \tag{1}$$

where $x_i(t)$ represents the state variable of the $i-th$ neuron. $i = 1,2,\ldots,n$. U_i is the activation function, and $w = [w_{ij}]_{nxn}$ is the weight matrix.

In this study, the weight matrix [53] is expressed as follows:

$$w = \begin{pmatrix} 1 & -600 & 4 \\ 980 & 3.5 & -1 \\ -1 & 4 & 1.5 \end{pmatrix}. \tag{2}$$

The weight matrix w is obtained through constant exploration and verification. The initial states of the system are $x_1(0)$, $x_2(0)$, and $x_3(0)$. The system state changes continuously under the action of the weight matrix and the excitation function. After continuous iteration, the system gradually enters a chaotic state.

The 3D chaotic Hopfield neural network of this study can be presented as follows:

$$\begin{cases} \dot{x}_1(t) = -x_1(t) + U_1 - 600U_2 + 4U_3, \\ \dot{x}_2(t) = -x_2(t) + 980U_1 + 3.5U_2 - U_3, \\ \dot{x}_3(t) = -x_3(t) - U_1 + 4U_2 + 1.5U_3, \end{cases} \tag{3}$$

where $x_1(t)$, $x_2(t)$, and $x_3(t)$ represent the state variables of the three neurons, respectively, and the activation function U_i is represented as follows:

$$U_i = \frac{|x_i(t)+1| - |x_i(t)-1|}{2}. \tag{4}$$

Equation 3 represents first-order ordinary differential equations. With the forementioned weight matrix (2) and the activation Eq. 4, we find that the system is chaotically optimal at initial values around [−0.215, −0.127, and 1.530] after continuous experimentation and verification using MATLAB. To use different keys for each encrypted image, we have changed the initial value to a variable in the neighborhood of [−0.215, −0.127, and 1.53], with two decimal places tending to be constant. Such changes to the initial values are minimal, and the system still eventually enters a chaotic state.

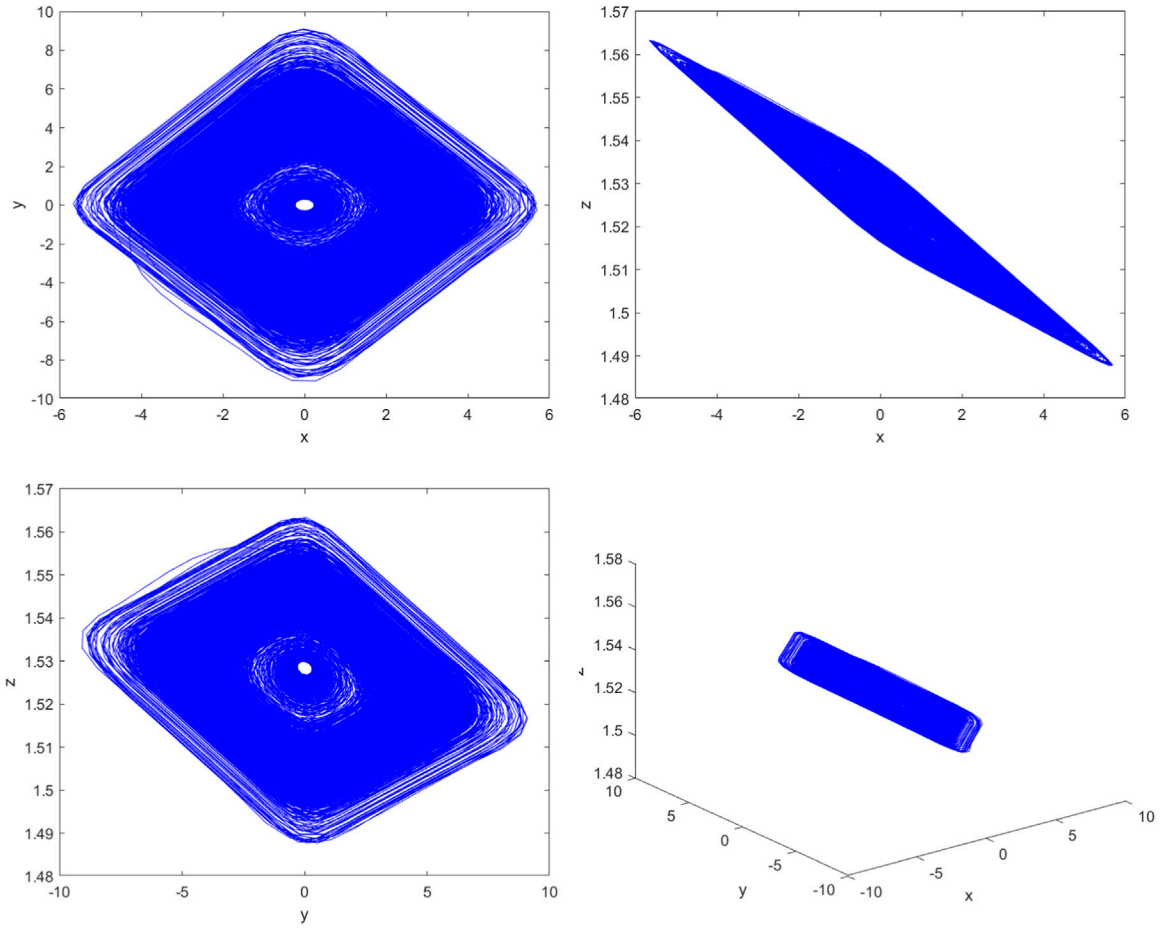


FIGURE 1
Phase diagrams of the chaotic Hopfield neural network system.

The initial values are chosen as follows:

$$\begin{cases} x_1(0) = -0.215, \\ x_2(0) = -0.127, \\ x_3(0) = \dfrac{mod\left(sum\left(p', all'\right), 10\right) + 1530}{1000}, \end{cases} \tag{5}$$

where *sum* represents the function of summing over pixel values, p represents the 3D array of plaintext images, and *mod* represents the residual function.

The phase diagrams of the system are shown in Figure 1, and the initial values of the system are $x_1(0) = -0.123$, $x_2(0) = -0.127$, and $x_3(0) = 1.530$, respectively. The neural network system consisting of three neurons enters a chaotic state with the presence of chaotic attractors.

To further confirm whether the system is in a chaotic state, we take the method of whether there exists a positive Lyapunov exponent [54] to judge. As the weights are determined and the initial values of the system are changed slightly, the Lyapunov exponent is always larger than zero. So, we can consider the proposed Hopfield neural network as a chaotic system [55].

3 Image-encryption and -decryption algorithms

In this section, the encryption process and decryption process of images are introduced.

3.1 Image encryption

Images are made up of pixels in electronics and are divided into color and grayscale images. Each pixel of a grayscale image contains only one pixel value, while every pixel of a color image consists of three pixel values of RGB. The grayscale image can be regarded as a two-dimensional array that contains the horizontal coordinates, vertical coordinates, and pixel value information for every pixel. The color image can be considered a three-dimensional array that contains the horizontal coordinates, vertical coordinates, and the RGB pixel values of every pixel. This study proposes a method to separate the RGB components of the color image and then perform the encryption operation. Equation 6 is used to separate the RGB component of the color image, which is shown as follows:

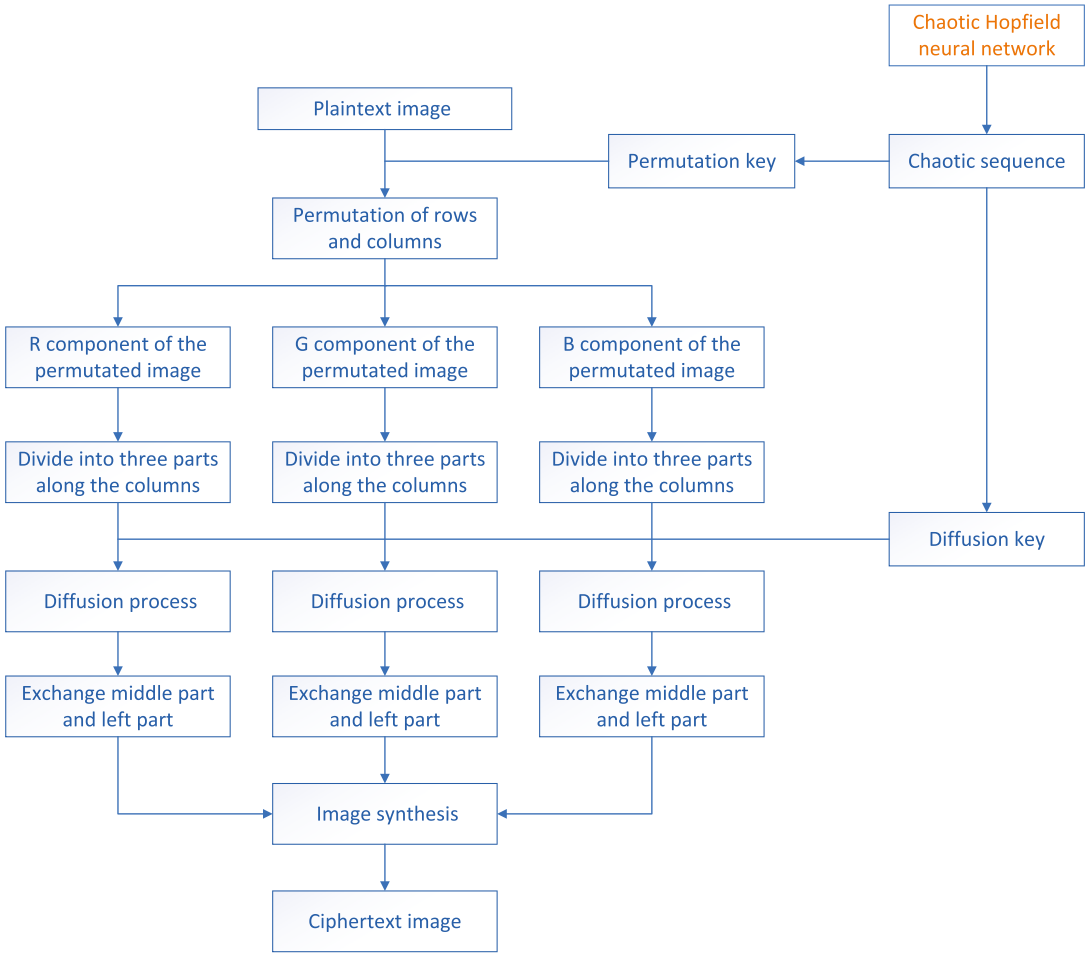


FIGURE 2
Flowchart of image encryption.

$$\begin{cases} PR = P(:,:,1), \\ PG = P(:,:,2), \\ PB = P(:,:,3), \end{cases} \tag{6}$$

where P represents the original image. PR, PG, and PB represent the R, G, and B components of the original image, respectively.

This study proposes an image-encryption algorithm based on a 3D chaotic Hopfield neural network and random row–column permutation. Row–column permutation is the process of changing the position of the pixel without changing the value. The chaotic sequences are generated by the proposed chaotic Hopfield neural network for producing the permutation keys and diffusion keys. In the permutation process, the positions of the ranks of the pixels will be changed according to the permutation keys. Image diffusion is the process of changing the pixel values of RGB with the diffusion keys.

The chaotic sequences are obtained through continuously iterating the proposed 3D chaotic neural network system. The length of the intercepted chaotic sequences depends on the number of pixels in the image. Here, the first 3,000 numbers should be removed from the sequences, which are generated before the chaotic neural network system is stabilized with some error and poor randomness. The intercepted sequences will stay within the interval [0,255] by taking the absolute value, expanding by 10^{15}, and taking the remainder. The permutation keys and diffusion keys are both obtained from the chaotic sequences.

The encryption algorithm in this study consists of two main steps: random row–column permutation and image diffusion. Details of the process can be described as follows: First, the rows and columns of the image are distorted through the permutation keys. Next, the RGB components of the image are separated by the method shown in Eq. 6. Each subgraph is formed by dividing the RGB components into three parts along columns and exchanging the middle and left part. The three parts of the subgraphs are encrypted with three different sets of diffusion keys. Finally, the subgraphs are synthesized to yield the encrypted images. The image encryption flowchart is shown in Figure 2.

3.1.1 Permutation process

The process of image permutation is to change the position of the image pixels without changing the values. First of all, the chaotic sequences are generated by the 3D chaotic Hopfield neural

network (3) for producing the permutation keys, which are combined with the increasing sequences of length and width of the corresponding image to form $2 \times M$ and $2 \times N$ key pairs, respectively. Then, the permutation process is achieved by the determinant transformation of the original image through the key pairs.

Step 1: The length M and width N of the image are obtained at first. Then, the chaotic sequences $x_1(i)$ and $x_2(j)$ ($i = 1,2,\ldots,M$ and $j = 1,2,\ldots,N$) are generated through the 3D chaotic Hopfield network system, respectively.

Step 2: The permutation keys are obtained through the chaotic sequences, and the expression of the generated function is as follows:

$$\begin{cases} RandM(i) = mod\left(floor\left(x_1(i)*10^{15}\right), M\right) + 1, \\ RandN(j) = mod\left(floor\left(x_2(j)*10^{15}\right), N\right) + 1, \end{cases} \tag{7}$$

where $RandM(i)$ and $RandN(j)$ ($i = 1,2,\ldots,M$ and $j = 1,2,\ldots,N$) represent the permutation keys of the row and column, respectively. The *floor* is the downward-rounding function. $x_1(i)$ and $x_2(j)$ are the chaotic sequences.

Step 3: Duplicate numbers are discarded when the permutation keys are obtained. For each number stored in $RandM(i)$ or $RandN(j)$, the corresponding position variables i or j increases by 1. This step is repeated until i and j have reached the values of M and N, or the chaotic sequences have been taken.

Step 4: The numbers in $RandM(i)$ and $RandM(j)$ are complemented because there is no guarantee that all non-repeating numbers from 1 to M or 1 to N will be taken; we need to find numbers that do not exist in the arrays and add them to the arrays until they are filled

Step 5: The key pairs are formed through the permutation keys and increasing sequences. The generating function of the key pairs is shown as follows:

$$\begin{cases} Mchange = [1{:}\ 1{:}\ M; RandM], \\ Nchange = [1{:}\ 1{:}\ N; RandN], \end{cases} \tag{8}$$

Specifically, *Mchange* and *Nchange* are $2 \times M$ and $2 \times N$ arrays, respectively. The first row is an increasing sequence of $1 \sim M$ or $1 \sim N$, and the second row is the sequence of $RandM(i)$ or $RandM(j)$, respectively. Ultimately, the mappings are formed by key pairs to perform permutation.

Step 6: The rows and columns of pixels are randomly permutated to get the permuted image. The permutation method can be shown as follows:

$$\begin{cases} P\left(Mchange(1,:),:\right) = P\left(Mchange(2,:),:\right), \\ P\left(:,Nchange(1,:)\right) = P\left(:,Nchange(2,:)\right), \end{cases} \tag{9}$$

The outline and main information of the permuted image are obscured after this process. However, there is still some plaintext image information that can be captured by illegal hackers. At the same time, the correlation between the pixel points of the adjacent is still at a high level. To make the encryption system works better, the diffusion process is performed.

3.1.2 Segmentation and diffusion process

The difference between image permutation and image diffusion is that image diffusion needs to change the original pixel value, which will completely distort the information of the whole image. The main information and details of the image are completely invisible, and the cryptographer cannot find any useful information.

Three subgraphs are generated by separating the R, G, and B components of the color image in the diffusion process. Each subgraph is split along the columns to form three equal parts. The primary information of the daily photos or machine-made images is usually centered. So, the left and middle parts are exchanged in this process. Three different chaotic sequences y_1, y_2, y_3 are obtained through the 3D chaotic Hopfield neural network system to produce the diffusion keys, which are used to encrypt the three parts, respectively. The experimental results obtained using MATLAB and FPGA prove that our proposed encryption algorithm has a good encryption effect. The detailed steps are as follows:

Step 1: First, RGB components of the permuted image are separated to form three subgraphs, which are divided along columns into three parts to exchange the left part and middle part. Three parts of the RGB components are generated by Eq. 10:

$$\begin{cases} LP(i, j_1) = PR\left(i, 1 \sim floor\left(\frac{N}{3}\right)\right), \\ CP(i, j_2) = PR\left(i, floor\left(\frac{N}{3}\right) + 1 \sim floor\left(\frac{2N}{3}\right)\right), \\ RP(i, j_3) = PR\left(i, floor\left(\frac{2N}{3}\right) + 1 \sim N\right), \end{cases} \tag{10}$$

where $j_1 = 1,3\ldots, floor\left(\frac{N}{3}\right)$, $j_2 = 1,2,\ldots, floor\left(\frac{2N}{3}\right) - floor\left(\frac{N}{3}\right)$, and $j_3 = 1,2,\ldots, N - floor\left(\frac{2N}{3}\right)$ $i = 1,2, \ldots, M$. *PR* represents the R component of the permuted image. *LP*, *CP*, and *RP* represent the left , middle , and right parts of the R component, respectively. The G and B components can be obtained in the same way.

Step 2: The diffusion keys are obtained by Eq. 11:

$$\begin{cases} k_1(n_1) = mod\left(floor\left(y_1(n_1)*10^{15}\right), M\right), \\ k_2(n_2) = mod\left(floor\left(y_2(n_2)*10^{15}\right), M\right), \\ k_3(n_3) = mod\left(floor\left(y_3(n_3)*10^{15}\right), M\right), \end{cases} \tag{11}$$

where $n_1 = 1,2,\ldots, floor\left(\frac{N}{3}\right)$, $n_2 = 1, 2, \ldots, N - floor\left(\frac{2N}{3}\right)$, and $n_3 = 1, 2, \ldots, floor\left(\frac{2N}{3}\right) - floor\left(\frac{N}{3}\right)$.

Step 3: Three different diffusion keys are used to encrypt the three parts of the subgraph, which is shown as follows:

$$\begin{cases} CLR(i, j_1) = bitxor\left(LP(i, j_1), k_1(n_1)\right), \\ CCP(i, j_2) = bitxor\left(CP(i, j_2), k_2(n_2)\right), \\ CRR(i, j_3) = bitxor\left(RP(i, j_3), k_3(n_3)\right), \end{cases} \tag{12}$$

where *CLR*, *CCR*, and *CRR* represent the left, middle, and right parts of the encrypted subgraph R, respectively. $bixor(P, k)$ represents the XOR operation. The pixel value of the cipher image is formed by the XOR operation between the original pixel value and the diffusion key. The information in the plaintext image is completely hidden.

Step 4: Splicing the three encrypted parts of the subgraph together, we get

$$CR = [CCR, CLR, CRR], \tag{13}$$

where *CR* represents the R component of the ciphertext image. The positions of *CCR* and *CLR* are exchanged to obtain a better encryption effect.

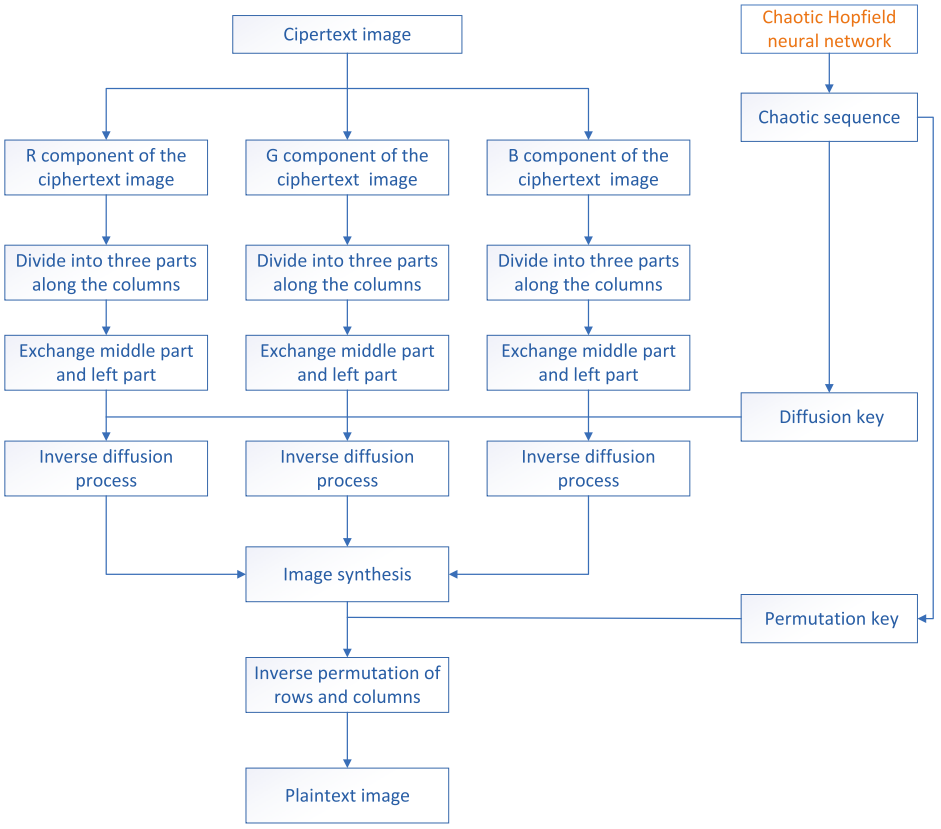


FIGURE 3
Flowchart of the decryption algorithm.

Step 5: Step 1 to Step 4 is repeated to realize the encryption of the G and B components.

Step 6: The encrypted RGB components are combined to form the final encrypted image. Furthermore, three two-dimension arrays are merged into one three-dimensional array, which is shown as follows:

$$C = cat\,(3, CR, CG, CB), \tag{14}$$

where C represents the ciphertext images after the diffusion process and *cat* represents the splicing function.

3.2 Image decryption

The image decryption is the inverse process of the image encryption. In this process, the ciphertext image is obtained at first. Then, the R, G, and B components of the ciphertext image are divided into three parts by columns, and then the middle part and left part are exchanged. After that, the inverse diffusion process is performed, and the RGB components are stitched to obtain the complete images. In image encryption, the column-transformed operation happens behind the row-transformed operation in the permutation process. So, the inverse operation is done in the inverse permutation process. Finally, the plaintext image is obtained. The flowchart of the image decryption is shown in Figure 3.

4 Simulation results

In this study, we selected four color images with a resolution of 512 × 512 for encryption.

4.1 Simulation results using MATLAB

The images with a clear outline, dense pixel distribution, uniform color, and uniform light and dark are selected as the test images because they are representative and better reflect the performance of the encryption algorithm. From Figure 4, we can see that the outline of the ciphertext image is invisible, and the pixels are equally distributed. So, it is almost impossible to obtain plaintext image information from the ciphertext image. The decrypted image is exactly the same as the plaintext image. It can be said that the encryption algorithm has excellent encryption performance.

4.2 Simulation results in FPGA

In this section, FPGA-based implementation of the proposed image cryptosystem is introduced. We have implemented FPGA debugging by using a Xilinx Zynq-7000 series XC7Z020 FPGA chip and an AN9767 dual-port parallel 14-bit digital-to-analog converter module with a maximum conversion rate of 125MHz, Vivado17.4,

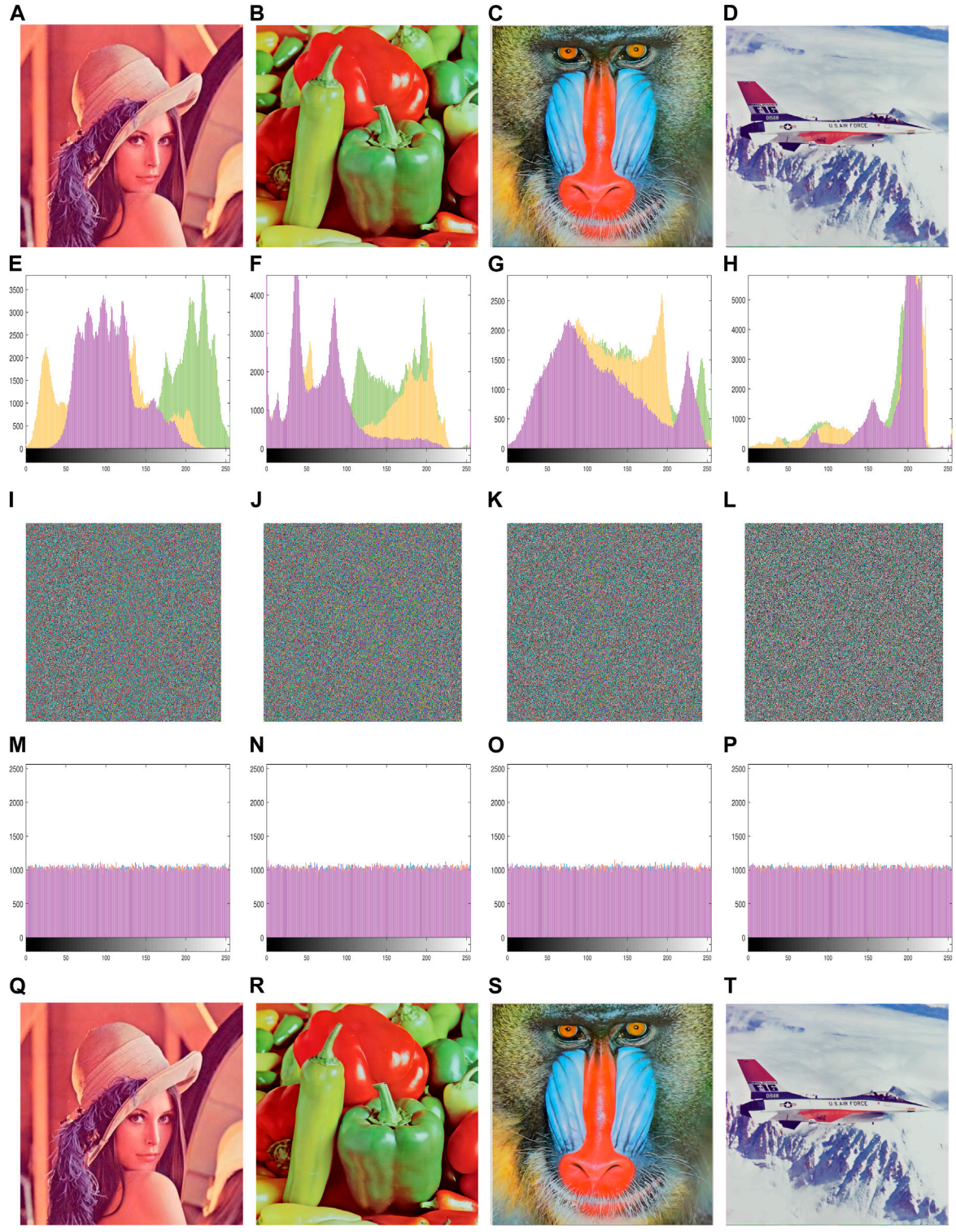


FIGURE 4
Simulation result: **(A–D)** plaintext images, **(E–H)** histograms of the plaintext images, **(I–L)** ciphertext images, **(M–P)** histograms of the ciphertext images, and **(Q–T)** decrypted images of ciphertext images.

and a system generator. The proposed image encryption scheme is implemented on an FPGA platform with the hardware design shown in Figure 5, consisting of five components: image memory, image encryption, image decryption, image display controller, and chaotic sequence controller. The image memory module is used to store plaintext images. The keys are obtained from the chaotic sequence controller. The plaintext image and the chaotic sequences are transferred to the image encryption

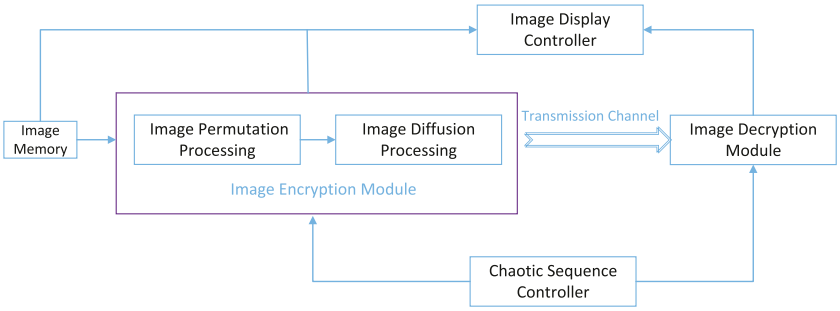


FIGURE 5
Structural diagram of FPGA-based image encryption and decryption.

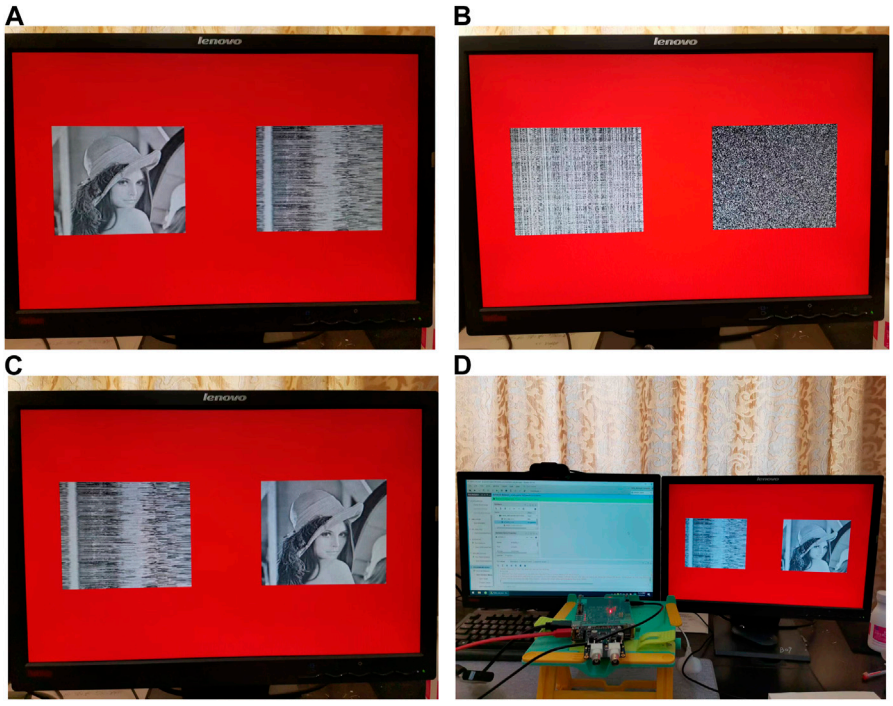


FIGURE 6
FPGA-based implementation results of the proposed image cryptosystem: **(A, B)** process of image encryption; **(C)** process of image decryption and **(D)** experimental equipment and environment.

module, which is used to generate the ciphertext image. The image encryption module consists of the image permutation process and the image diffusion process module. The image decryption module is the inverse of the encryption module and decrypts the ciphertext image into a plaintext image. The image display controller module is used to display both plaintext and ciphertext images.

FPGA-based implementation result of the proposed image cryptosystem is shown in Figure 6. From Figures 6A, B, the plaintext image and the encrypted image are shown on the screen. The images on the right in Figure 6A and on the left in Figure 6B are the permutated images, and the image on the right in Figure 6B is the ciphertext image. Figure 6C is the decryption result of the FPGA-based implementation. Experiments have demonstrated no significant difference between the FGPA platform and MATLAB regarding the effectiveness of image encryption and decryption.

5 Performance analysis

This section is to verify the security and efficiency of the proposed encryption algorithm. The simulation test is performed on a computer using MATLAB R2020b.

5.1 Histogram analysis

The histogram can reflect the distribution of the overall pixel values of the image accurately and intuitively. There is only one component of the pixel value in a grayscale image, so the greyscale image has only one histogram. However, the pixel value of the color image consists of R, G, and B components. Therefore, a color image has three histograms, representing the occurrence of R, G, and B pixel values, respectively. The histogram is a two-dimensional statistical map, where the abscissa represents each pixel value in the color image and the ordinate indicates the frequency of each pixel value appearing in the color image. The analysis of the histogram can capture information about the images. The encryption system with high security should make the histograms of the ciphertext image as uniform as possible. Four color images are selected for histogram analysis; the histograms of plaintext images are shown in Figures 4E–H, and the histograms of the ciphertext images are shown in Figures 4M–P.

The histograms of R, G, and B components of the plaintext images show a mountainous pattern with an uneven distribution of the pixels, while the histograms of the ciphertext images are very uniform and the characteristics of the distribution of the image pixels are well hidden. It is difficult for a cracker to obtain any useful information from the histograms. It can be inferred that this encryption scheme has great security.

5.2 Correlation analysis

Correlation analysis reflects the degree of correlation of pixel values at adjacent positions in the image. The size of the correlation coefficient of adjacent pixel values in the ciphertext image can better reflect the effects of the encryption algorithm. The lower the correlation, the better the encryption effects of the ciphertext image obtained by the proposed encryption algorithm. The correlation coefficient of a good color image-encryption algorithm should be close to zero. The correlation analysis equation is as follows:

$$\begin{cases} E(u) = \frac{1}{n}\sum_{i=1}^{n} u_i, \\ D(u) = \frac{1}{n}\sum_{i=1}^{n} [u_i - E(u)]^2. \\ cov(u,v) = \frac{1}{n}\sum_{1}^{n} [u_i - E(u)][v_i - E(v)], \\ r = \frac{cov(u,v)}{\sqrt[2]{D(u)D(v)}}, \end{cases} \tag{15}$$

where u_i and v_i represent the adjacent pixel values in the image and n is the number of pixels sampled. $E(u)$ and $E(v)$ represent the expectation of u and v. $cov(u, v)$ represents the covariance, and r is the correlation coefficient.

The correlations should be analyzed in the R, G, and B components separately in the color images, and we need to analyze the four directions of the image: horizontal, vertical, positive-diagonal, and negative angles. Here, 10,000 pixels of the components R, G, and B are randomly taken. If the coordinate point of u_i is (x_i, y_i), then the adjacent coordinate point in the horizontal direction is set to v_i $(x_i + 1, y_i)$. Similarly, the adjacent coordinate point we set in the vertical direction is v_i $(x_i, y_i + 1)$, in the positive-diagonal direction; we set v_i $(x_i + 1, y_i + 1)$, and the adjacent coordinate points in the opposite angular direction were set as v_i $(x_i - 1, y_i + 1)$.

The correlation coefficient of the plaintext image is almost close to 1, which indicates that the correlation of the pixels in the plaintext image is extremely strong. However, the ciphertext image is almost close to 0, indicating the adjacent pixels in the ciphertext image have almost no correlation. The correlation test in plaintext and ciphertext images is shown in Figure 7, which contains the distributions in those four directions, respectively. The ciphertext image shows the irregular distribution in four directions, and the pixel values around each pixel point are arbitrarily random. However, most of the points in Figures 7A, C, E, G are around a straight line, indicating there is a significant correlation in the plaintext image.

The results of the correlation coefficient in different directions are shown in Table 1. We can see that the proposed scheme has a remarkable performance. Only one of the correlation coefficients obtained by the proposed algorithm is higher than others.

5.3 Analysis of information entropy

Information entropy is an index to evaluate the performance of the encryption algorithm. The higher the information entropy index, the better the performance of this encryption algorithm. The information entropy equation is as follows:

$$H = -\sum_{i=0}^{L} P(i) log_2 P(i), \tag{16}$$

where $P(i)$ represents the probability of the occurrence of the pixel value of i. The ideal entropy for the R, G, and B components of the color image should be equal to 8. The color Lena graph, Baboon graph, Pepper graph, and plane graph are chosen as the test images, which are encrypted by the proposed encryption algorithm. The information entropy of R, G, and B components are analyzed by Eq. 16. The test results are shown in Table 2.

The table clearly shows the information entropies of the R, G, and B components of the encrypted images Cipher-Lena, Cipher-Baboon, Cipher-Pepper, and Cipher-plane. The entropies obtained by the proposed algorithm are close to the ideal value. From Table 3, we can see that most of the entropies are larger than those obtained by other algorithms. This feature prevents information leakage during the encryption process. So, we can infer that the proposed algorithm is significantly secure.

5.4 Analysis of PSNR and MSE

PSNR and MSE are used to describe the difference between the original and encrypted images. The greater the difference between the plaintext image and the ciphertext image, the better the performance of the encryption algorithm. PSNR is the peak signal-to-noise ratio, which is an index of distortion

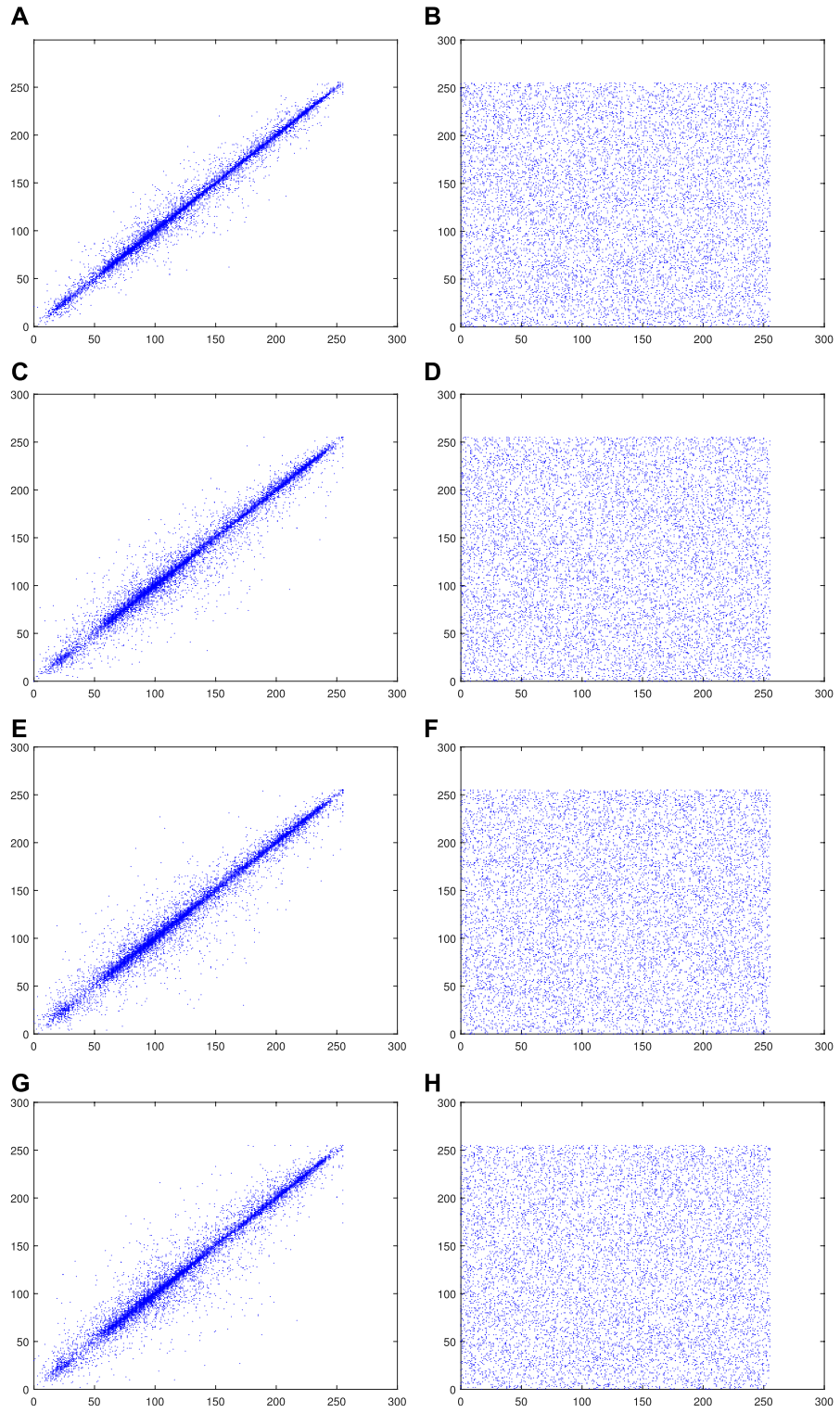


FIGURE 7
Correlation test result of Lena: **(A)** Horizontal diagonal of the plaintext image, **(B)** horizontal diagonal of the ciphertext image, **(C)** vertical diagonal of the plaintext image, **(D)** vertical diagonal of the ciphertext image, **(E)** positive diagonal of the plaintext image, **(F)** positive diagonal of the ciphertext image, **(G)** opposition diagonal of the plaintext image, and **(H)** opposition diagonal of the ciphertext image.

between the plaintext and ciphertext images. The lower the PSNR value, indicating that the greater the difference between plaintext and ciphertext images, the better the encryption algorithm is.

MSE is the mean squared error, which is used to calculate the cumulative squared error between plaintext and ciphertext images. The larger the MSE, the better the encryption effect. The PSNR and MSE are defined as follows:

TABLE 1 Correlation coefficient of the plain image and Cipher image.

Test image	Direction	Plain image	Cipher image				
			Proposed	Reference [56]	Reference [57]	Reference [58]	Reference [23]
Lena	Horizontal	0.99097	−0.00036043	0.0020	−0.004223	−0.0036	0.0030
	Vertical	0.98199	0.0072923	0.0006	0.000551	−0.0045	0.0101
	Positive	0.97328	−0.0022938	0.0055	−0.003665	−0.0041	0.0037
	Diagonal						
	Negative	0.98019	−0.0009774				
	Diagonal						
Baboon	Horizontal	0.89991	−0.00057325	0.0032	0.002188	−0.0036	0.0021
	Vertical	0.92837	0.0032677	0.0034	0.001276	−0.0014	0.0082
	Positive	0.86561	0.0025915	0.0014	0.002372	−0.0065	0.0095
	Diagonal						
	Negative	0.8554	0.00051994				
	Diagonal						
Pepper	Horizontal	0.98899	0.00026825	−0.0078	−0.001830	0.0004	0.0012
	Vertical	0.99003	0.00063181	0.0010	0.002380	0.0013	0.0037
	Positive	0.97805	−0.00062647	−0.0014	−0.00310	−0.0007	0.0005
	Diagonal						
	Negative	0.97989	0.0014451				
	Diagonal						
Average	Horizontal	—	0.000400643	0.0078	0.002747	0.0025333	0.0021
	Vertical	—	0.0032677	0.0040666	0.0014023	0.0024	0.0037
	Positive	—	0.001837357	0.0027667	0.003045667	0.0037667	0.0045667
	Diagonal						
	Negative	—	0.000980813				
	Diagonal						

TABLE 2 Analysis of information entropy.

Image	R	G	B	Average
Lena	7.9993	7.9993	7.9993	7.99926
Baboon	7.9993	7.9992	7.9993	7.99926
Plane	7.9993	7.9994	7.9993	7.99933
Pepper	7.9993	7.9992	7.9992	7.99923

TABLE 3 Information entropy comparison.

Image	Lena	Baboon	Pepper
Proposed	7.99926	7.99926	7.99923
Reference [12]	7.9993	7.9994	—
Reference [59]	7.9975	7.9975	—
Reference [56]	7.9992436	7.993865	7.9994558
Reference [57]	7.9992495	7.9992635	7.9993617
Reference [58]	7.9977	7.9976	7.9992

$$\begin{cases} PSNR = 10 \times \lg \dfrac{MAX^2}{MSE}, \\ MSE = \dfrac{1}{M \times N}[P(i,j) - C(i,j)]^2, \end{cases} \tag{17}$$

where P and C represent the plaintext and the ciphertext images, respectively. In addition, (i,j) stands for the position of each pixel.

The comparison of PSNR and MSE among the proposed and other algorithms is shown in Tables 4, 5, respectively. Most of the PSNR and MSE indices of our proposed algorithm are superior compared to those of others. The results show that the proposed encryption scheme has better performance.

TABLE 4 Result of PSNR analysis.

Image	Proposed	Reference [56]	Reference [57]	Reference [59]	Reference [23]
Lena	8.6485	9.2238	9.2267	8.5979	9.3494
Baboon	8.81	9.5161	9.7296	9.0304	9.5334
Pepper	8.0439	8.4531	8.8792	—	9.0214
Average	8.0439	9.0643	9.2785	8.81415	9.33014

TABLE 5 Result of MSE analysis.

Image	Proposed	Reference [56]	Reference [57]	Reference [59]	Reference [23]
Lena	8876.1674	7775.0	31078.8827	8980.4	7553.4
Baboon	8552.2752	7269.0	6920.1784	8129.1	7240
Pepper	10245.3434	9284.7	33667.8582	—	8145.9
Average	9924.59533	8109.567	23888.9731	8554.75	7646.433

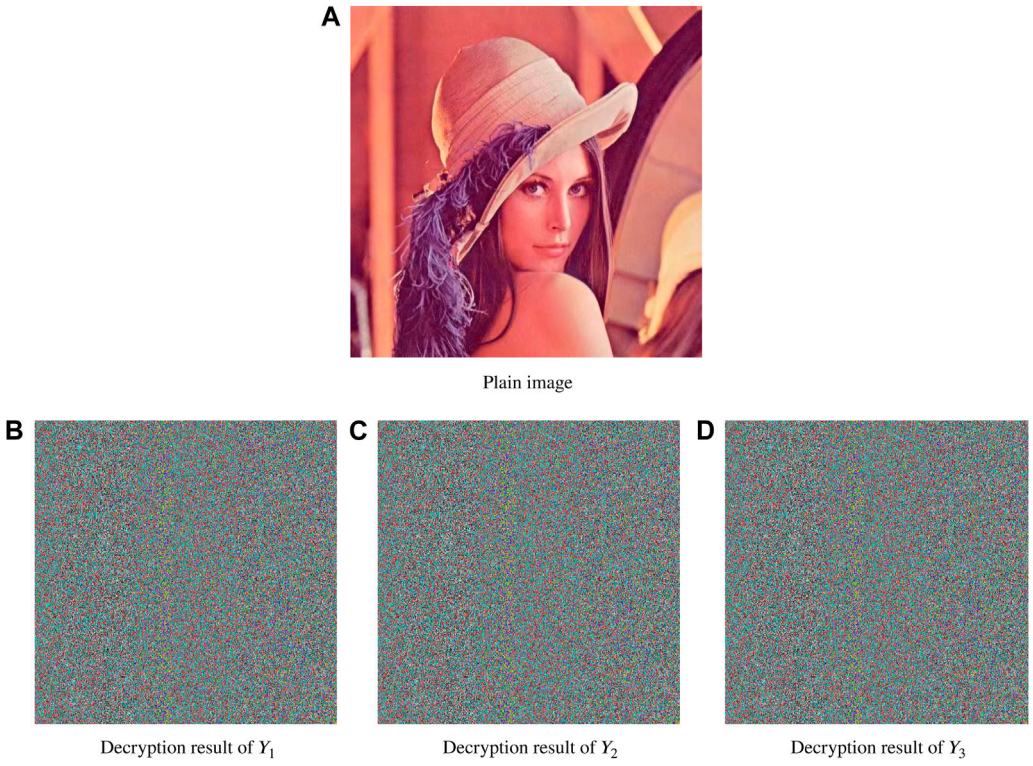


FIGURE 8
Test results of sensitivity analysis. **(A)** Plaintext image. **(B)** Decryption result of Y_1. **(C)** Decryption result of Y_2 **(D)** Decryption result of Y_3.

5.5 Sensitivity analysis

Sensitivity analysis is decrypting the encrypted image with the keys whose initial values are slightly different from the original keys to see if the encrypted image can be decrypted correctly. The Lena plaintext image is encrypted by using the keys. Then, the ciphertext image is decrypted with the pseudo-keys of three very close key values Y_1, Y_2, and Y_3, respectively. The initial values of the pseudo-keys are as follows:

$$\begin{cases} Y_1\colon x_1(0) = x_1(0) + 10^{-10},\ x_2(0) = x_2(0),\ x_3(0) = x_3(0), \\ Y_2\colon x_1(0) = x_1(0),\ x_2(0) = x_2(0) + 10^{-10},\ x_3(0) = x_3(0), \\ Y_3\colon x_1(0) = x_1(0),\ x_2(0) = x_2(0),\ x_3(0) = x_3(0) + 10^{-10}, \end{cases} \tag{18}$$

where $x_1(0)$, $x_2(0)$, and $x_3(0)$ are the initial values of the keys. The three aforementioned sets of pseudo-keys with slightly different initial values are used to decrypt the Lena ciphertext image. The results of decrypting are shown in Figure 8. The plaintext image cannot be recovered correctly. Therefore, the encryption system proposed in this study satisfies the requirement of key sensitivity.

5.6 Analysis of key space

A good image-encryption algorithm must have the ability to withstand outside attacks. Therefore, the key space must be large enough to ensure the security of the encryption algorithm. The key space of an ideal image encryption is larger than 2^{100}.

The computer computational accuracy is about 10^{15}, and the compression rate CR is 10^5. In this study, the key generation process consists of the following: 1) the initial values of $x_1(0)$, $x_2(0)$, and $x_3(0)$ are used for the chaotic Hopfield system iteration and the sampling time point, and 2) chaotic sequences $y_1(t)$, $y_2(t)$, and $y_3(t)$ are also used. So the key space is calculated by Eq. 19:

$$10^2 \times 10^{15} \times 10^{14} \times 10^{14} \times 10^{14} \times 10^{14} \times 10^{14} \times 10^{14} \times 10^{14} \gg 2^{100}. \quad (19)$$

This shows that the encryption algorithm has a large enough key space to resist exhaustive attacks.

6 Conclusion

This study proposes a color image-encryption algorithm based on random row–column permutation and a 3D chaotic Hopfield neural network. The 3D chaotic Hopfield neural network is used to generate chaotic sequences to ensure the randomness of keys. After the permutation process, three subgraphs are formed by separating the R, G, and B components of the color image, and then, the subgraphs are cut along the columns for swapping the middle part and the left part. Three diffusion keys are produced through the chaotic sequence, and then the three parts of the subgraphs are encrypted separately. In this study, we consider the interrelationship of the pixel values of the RGB components in the plaintext image, so three sets of diffusion keys are used to encrypt the three parts of the split RGB subgraphs. This measure effectively reduces the interconnection of pixel values. Through extensive simulations and security analysis, the simulation results in MATLAB and FPGA show that the encryption algorithm has superior performance and high security.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material; further inquiries can be directed to the corresponding author.

Ethics statement

Ethical review and approval were not required for the study on human participants in accordance with the local legislation and institutional requirements. Written informed consent for participation was not required for this study in accordance with the national legislation and the institutional requirements.

Author contributions

Conceptualization: WY, KG, ZZ, LC, and JZ. Methodology: WY. Hardware: LC. Validation: KG, ZZ, LC, and JZ. Formal analysis: KG, ZZ, LC, and JZ. Investigation: ZZ. Resources: WY, KG, LC, and JZ. Data curation: JZ and ZZ. Writing—original draft preparation: KG. Writing—review and editing: WY, JZ, and LC. Visualization: JZ and ZZ. Supervision: JZ and LC. Project administration: JZ and WY. Funding acquisition: JZ and WY. All authors have read and agreed to the published version of the manuscript.

Funding

This work was supported in part by Open Fund of Engineering Laboratory of Spatial Information Technology of Highway Geological Disaster Early Warning in Hunan Province, Changsha University of Science & Technology under Grant No. kfj220603, Hunan Provincial Natural Science Foundation of China, under Grant No. 2022JJ40514, the Scientific research project of Hunan Provincial Department of Education, China under Grant No. 21C0200.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Gui Y, Zeng G. Joint learning of visual and spatial features for edit propagation from a single image. *Vis Comput* (2019) 36:469–82.

2. Long M, Peng F, Li H. Separable reversible data hiding and encryption for hevc video. *J Real Time Image Process* (2018) 14:171–82.

3. Hirata Y, Oku M, Aihara K. Chaos in neurons and its application: Perspective of chaos engineering. *Chaos: Interdiscip J Nonlinear Sci* (2012) 22(4):047511. doi:10.1063/1.4738191

4. Mcneal P, Petcovic H, Bals-Elsholz T, Ellis T. Seeing weather through chaos: A case study of disembedding skills in undergraduate meteorology

students. *Bull Am Meteorol Soc* (2019) 100(6):997–1010. doi:10.1175/bams-d-18-0015.1

5. Wan Q, Li F, Chen S, Yang Q. Symmetric multi-scroll attractors in magnetized hopfield neural network under pulse controlled memristor and pulse current stimulation. *Chaos Solitons Fractals* (2023) 169:113259.

6. Spelta A, Pecora N, Pagnottoni P. Chaos based portfolio selection: A nonlinear dynamics approach. *Expert Syst Appl* (2022) 188:116055. doi:10.1016/j.eswa.2021.116055

7. Lai Q, Yang L, Liu Y. Design and realization of discrete memristive hyperchaotic map with application in image encryption. *Chaos, Solitons & Fractals* (2022) 165: 112781. doi:10.1016/j.chaos.2022.112781

8. Han X, Mou J, Jahanshahi H, Cao Y, Bu F. A new set of hyperchaotic maps based on modulation and coupling. *The Eur Phys J Plus* (2022) 137:523. doi:10.1140/epjp/s13360-022-02734-3

9. Wang X, Zhao J, Liu H. A new image encryption algorithm based on chaos. *Opt Commun* (2012) 285(5):562–6. doi:10.1016/j.optcom.2011.10.098

10. Li C, Luo G, Qin K, Li C. An image encryption scheme based on chaotic tent map. *Nonlinear Dyn* (2017) 87(1):127–33. doi:10.1007/s11071-016-3030-8

11. Lai Q, Hu G, Erkan U, Toktas A. 2023, A novel pixel-split image encryption scheme based on 2D salomon map. *Expert Syst Appl*, 213:118845, doi:10.1016/j.eswa.2022.118845

12. Chai X, Gan Z, Yuan K, Chen Y, Liu X. A novel image encryption scheme based on DNA sequence operations and chaotic systems. *Neural Comput Appl* (2017) 31(1): 219–37. doi:10.1007/s00521-017-2993-9

13. Yu J, Xie W, Zhong Z, Wang H. Image encryption algorithm based on hyperchaotic system and a new DNA sequence operation. *Chaos, Solitons & Fractals* (2022) 162:112456. doi:10.1016/j.chaos.2022.112456

14. Wu J, Liao X, Bo Y. Image encryption using 2D hénon-sine map and DNA approach. *Signal Process.* (2018) 153:11–23. doi:10.1016/j.sigpro.2018.06.008

15. Enayatifar R, Abdullah A, Isnin I. Chaos-based image encryption using a hybrid genetic algorithm and a DNA sequence. *Opt Lasers Eng* (2014) 56:83–93. doi:10.1016/j.optlaseng.2013.12.003

16. Chen J, Zhu Z, Zhang L, Zhang B, Yang Y. Exploiting self-adaptive permutation–diffusion and DNA random encoding for secure and efficient image encryption. *Signal Process.* (2018) 142:340–53. doi:10.1016/j.sigpro.2017.07.034

17. Li M, Lu D, Xiang Y, Zhang Y, Ren H. Cryptanalysis and improvement in a chaotic image cipher using two-round permutation and diffusion. *Nonlinear Dyn* (2019) 96: 31–47. doi:10.1007/s11071-019-04771-7

18. Wang M, Wang X, Zhang Y, Gao Z. A novel chaotic encryption scheme based on image segmentation and multiple diffusion models. *Opt Laser Technol* (2018) 108: 558–73. doi:10.1016/j.optlastec.2018.07.052

19. Chai X, Chen Y, Broyde L. A novel chaos-based image encryption algorithm using DNA sequence operations. *Opt Lasers Eng* (2017) 88:197–213. doi:10.1016/j.optlaseng.2016.08.009

20. Liu H, Wang X. Color image encryption based on one-time keys and robust chaotic maps. *Comput Maths Appl* (2010) 59(10):3320–7. doi:10.1016/j.camwa.2010.03.017

21. Liu H, Wang X. Color image encryption using spatial bit-level permutation and high-dimension chaotic system. *Opt Commun* (2011) 284(16-17):3895–903. doi:10.1016/j.optcom.2011.04.001

22. Chen J, Zhu Z, Chong F, Hai Y, Zhang Y. Reusing the permutation matrix dynamically for efficient image cryptographic algorithm. *Signal Process.* (2015) 111: 294–307. doi:10.1016/j.sigpro.2015.01.003

23. Diab H, El-Semary A. Cryptanalysis and improvement of the image cryptosystem reusing permutation matrix dynamically. *Signal Process. Official Publ Eur Assoc Signal Process* (2018) 148:172–92. doi:10.1016/j.sigpro.2018.02.011

24. Yao W, Wang C, Sun Y, Zhou C, Lin H. Exponential multistability of memristive cohen-grossberg neural networks with stochastic parameter perturbations. *Appl Maths Comput* (2020) 386:125483. doi:10.1016/j.amc.2020.125483

25. Yao W, Yu F, Zhang J, Zhou L. Asymptotic synchronization of memristive cohen-grossberg neural networks with time-varying delays via event-triggered control scheme. *Micromachines* (2022) 13:726. doi:10.3390/mi13050726

26. Yu F, Shen H, Yu Q, Kong X, Sharma P, Cai S. Privacy protection of medical data based on multi-scroll memristive hopfield neural network. *IEEE Trans Netw Sci Eng* (2023) 10(2):845–58. doi:10.1109/tnse.2022.3223930

27. Wang J, Zou Y, Lei P, Simon Sherratt R, Wang L. Research on recurrent neural network based crack opening prediction of concrete dam. *J Internet Technol* (2020) 21(4):1161–9.

28. Wang J, Wu Y, He S, Sharma P, Yu X, AlFarraj O, et al. Lightweight single image super-resolution convolution neural network in portable device. *KSII Trans Internet Inf Syst* (2021) 115:4065–83.

29. Long M, Zeng Y. Detecting iris liveness with batch normalized convolutional neural network. *Comput Mater Continua* (2019) 58(2):493–504. doi:10.32604/cmc.2019.04378

30. Strukov D, Snider G, Stewart D, Williams R. The missing memristor found. *Nature* (2008) 453:80–3. doi:10.1038/nature06932

31. Yao W, Wang C, Sun Y, Zhou C. Robust multimode function synchronization of memristive neural networks with parameter perturbations and time-varying delays. *IEEE Trans Syst Man, Cybernetics: Syst* (2022) 52(1):260–74. doi:10.1109/tsmc.2020.2997930

32. Lai Q, Wan Z, Zhang H, Chen G. Design and analysis of multiscroll memristive hopfield neural network with adjustable memductance and application to image encryption. *IEEE Trans Neural Networks Learn Syst* (2022)(99) 1–14. doi:10.1109/tnnls.2022.3146570

33. Lai Q, Wan Z, Kuate P. Generating grid multi-scroll attractors in memristive neural networks. *IEEE Trans Circuits Syst Regular Pap* (2023) 70(3):1324–36. doi:10.1109/tcsi.2022.3228566

34. Yu F, Shen H, Zhang Z, Huang Y, Cai S, Du S. A new multi-scroll chua's circuit with composite hyperbolic tangent-cubic nonlinearity: Complex dynamics, hardware implementation and image encryption application. *Integration, VLSI J* (2021) 81:71–83. doi:10.1016/j.vlsi.2021.05.011

35. Cui L, Luo W, Ou Q. Analysis and implementation of new fractional-order multi-scroll hidden attractors. *Chin Phys B* (2021) 30(2):020501. doi:10.1088/1674-1056/abbbe4

36. Cui L, Luo W, Ou Q. Analysis of basins of attraction of new coupled hidden attractor system. *Chaos, Solitons & Fractals* (2021) 146:110913. doi:10.1016/j.chaos.2021.110913

37. Yu F, Liu L, Xiao L, Li K, Cai S. A robust and fixed-time zeroing neural dynamics for computing time-variant nonlinear equation using a novel nonlinear activation function. *Neurocomputing* (2019) 350:108–16. doi:10.1016/j.neucom.2019.03.053

38. Ren L, Mou J, Banerjee S, Zhang Y. A hyperchaotic map with a new discrete memristor model: Design, dynamical analysis, implementation and application. *Chaos, Solitons & Fractals* (2023) 167:113024. doi:10.1016/j.chaos.2022.113024

39. Wang X, Zhao Y, Zhang H, Guo K. A novel color image encryption scheme using alternate chaotic mapping structure. *Opt Lasers Eng* (2016) 82:79–86. doi:10.1016/j.optlaseng.2015.12.006

40. Zhu H, Ge J, Qi W, Zhang X, Lu X. Dynamic analysis and image encryption application of a sinusoidal-polynomial composite chaotic system. *Mathematics Comput Simulation* (2022) 198:188–210. doi:10.1016/j.matcom.2022.02.029

41. Lin H, Wang C, Deng Q, Xu C, Deng Z, Zhou C. Review on chaotic dynamics of memristive neuron and neural network. *Nonlinear Dyn* (2021) 106(1):959–73. doi:10.1007/s11071-021-06853-x

42. Sang L, Guo Y, Liu H, Zhang J, Wang Y. Real-time all-optical random numbers based on optical Boolean chaos. *Opt Express* (2021) 29(5):7100–9. doi:10.1364/oe.420010

43. Li S, Liu Y, Ren F, Yang Z. Design of a high throughput pseudo-random number generator based on discrete hyper-chaotic system. *IEEE Trans Circuits Syst Express Briefs* (2023) 70(2):806–10. doi:10.1109/tcsii.2022.3178103

44. Xu S, Wang X, Ye X. A new fractional-order chaos system of Hopfield neural network and its application in image encryption. *Chaos, Solitons & Fractals* (2022) 157: 111889. doi:10.1016/j.chaos.2022.111889

45. Wang X, Li Z. A color image encryption algorithm based on hopfield chaotic neural network. *Opt Lasers Eng* (2019) 115:107–18. doi:10.1016/j.optlaseng.2018.11.010

46. Chen L, Hao Y, Huang T, Yuan L, Yin L. Chaos in fractional-order discrete neural networks with application to image encryption. *Neural Networks* (2020) 125:174–84. doi:10.1016/j.neunet.2020.02.008

47. Wu Y, Zeng J, Dong W, Li X, Qin D, Ding Q. A novel color image encryption scheme based on hyperchaos and hopfield chaotic neural network. *Entropy* (2022) 24(10):1474. doi:10.3390/e24101474

48. Zhang Z, Tang J, Zhang F, Ni H, Chen J, Huang Z. Color image encryption using 2D sine-cosine coupling map. *IEEE Access* (2022) 10:67669–85. doi:10.1109/access.2022.3185229

49. Rani N, Sharma S, Mishra V. Grayscale and colored image encryption model using a novel fused magic cube. *Nonlinear Dyn* (2022) 108:1773–96. doi:10.1007/s11071-022-07276-y

50. Hosny K, Kamal S, Darwish M. A novel color image encryption based on fractional shifted gegenbauer moments and 2D logistic-sine map. *Vis Comput* (2023) 39:1027–44. doi:10.1007/s00371-021-02382-1

51. Chen L, Hao Y, Yuan L, Machado J, Alam Z. Double color image encryption based on fractional order discrete improved henon map and rubik's cube transform. *Signal Processing: Image Commun* (2021) 97:116363. doi:10.1016/j.image.2021.116363

52. Hopfield J. Neural networks and physical systems with emergent collective computational abilities. *Proc Natl Acad Sci* (1982) 79:2554–8. doi:10.1073/pnas.79.8.2554

53. Wang Y, Li J, Wang Y. Hyperchaotic image encryption algorithm based on 4D of hopfield-type neural network and AES algorithm. *Comput Eng Appl* (2018) 54:202–7.

54. Eckhardt B, Yao D. Local lyapunov exponents in chaotic systems. *Physica D Nonlinear Phenomena* (1993) 65(1-2):100–8. doi:10.1016/0167-2789(93)90007-n

55. Koçak H, Palmer K. Lyapunov exponents and stability in interval maps. *SeMA J* (2010) 51:79–82. doi:10.1007/bf03322557

56. Liu L, Zhang Z, Chen R. Cryptanalysis and improvement in a plaintext-related image encryption scheme based on hyper chaos. *IEEE Access* (2019) 7:126450–63. doi:10.1109/access.2019.2938181

57. Zhang W, Yu H, Zhao Y, Zhu Z. Image encryption based on three-dimensional bit matrix permutation. *Signal Process.* (2016) 118:36–50. doi:10.1016/j.sigpro.2015.06.008

58. Benaissi S, Chikouche N, Hamza R. A novel image encryption algorithm based on hybrid chaotic maps using a key image. *Optik* (2023) 272:170316. doi:10.1016/j.ijleo.2022.170316

59. Lone P, Singh D, Mir U. Image encryption using DNA coding and three-dimensional chaotic systems. *Multimedia Tools Appl* (2022) 81:5669–93. doi:10.1007/s11042-021-11802-2

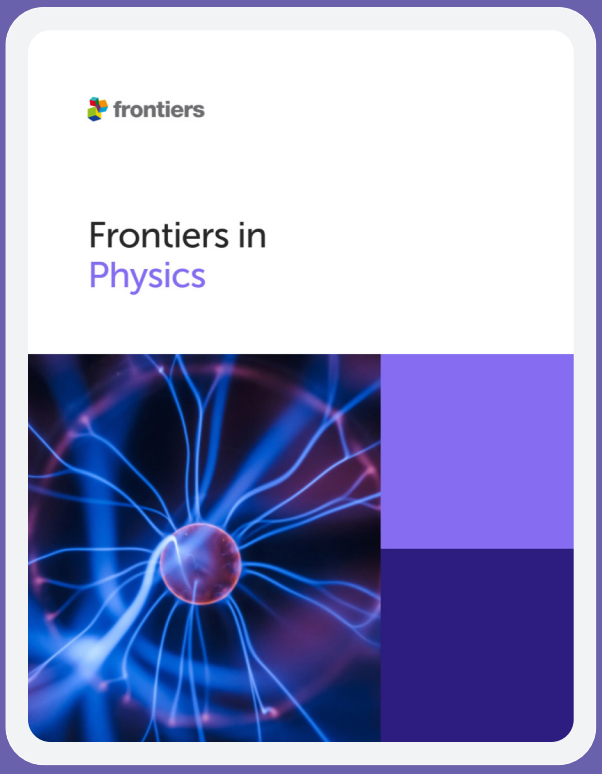
Frontiers in Physics

Investigates complex questions in physics to understand the nature of the physical world

Addresses the biggest questions in physics, from macro to micro, and from theoretical to experimental and applied physics.

Discover the latest Research Topics

See more →



Frontiers

Avenue du Tribunal-Fédéral 34
1005 Lausanne, Switzerland
frontiersin.org

Contact us

+41 (0)21 510 17 00
frontiersin.org/about/contact