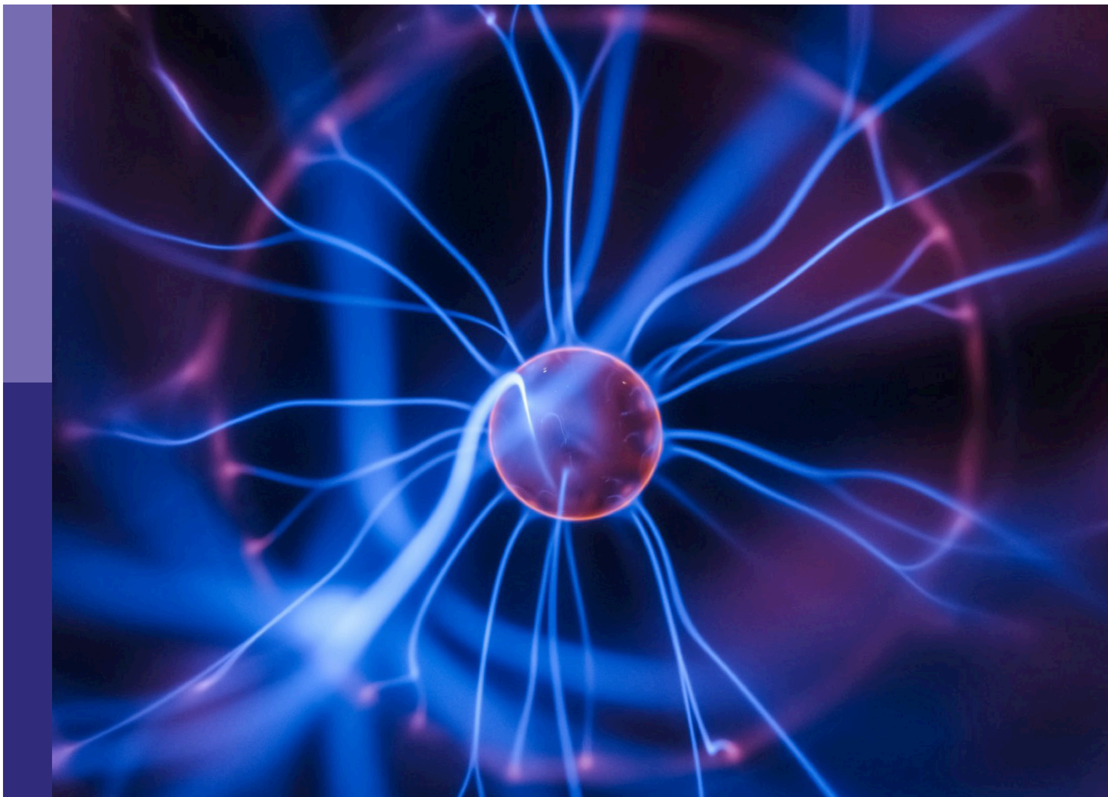
Research Topics

Advances in nonlinear systems and networks, volume III

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

Published in
Frontiers in Physics



February 2026

ISSN 1664-8714
ISBN 978-2-8325-6880-4
DOI 10.3389/978-2-8325-6880-4

Generative AI statement
Any alternative text (Alt text) provided alongside figures in the articles in this ebook has been generated by Frontiers with the support of artificial intelligence and reasonable efforts have been made to ensure accuracy, including review by the authors wherever possible. If you identify any issues, please contact us.

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, volume III

Topic editors
Fei Yu — Changsha University of Science and Technology, China
Ho Ching Iu — University of Western Australia, Australia
Hairong Lin — Central South University, China
Viet-Thanh Pham — Industrial University of Ho Chi Minh City, Vietnam

Citation
Yu, F., Iu, H. C., Lin, H., Pham, V.-T., eds. (2026). *Advances in nonlinear systems and networks, volume III*. Lausanne: Frontiers Media SA.
doi: 10.3389/978-2-8325-6880-4

Table of contents

frontiers | Frontiers in Physics

TYPE Editorial
PUBLISHED 03 September 2025
DOI 10.3389/fphy.2025.1688722

Editorial: Advances in nonlinear systems and networks, volume III

Fei Yu[1]*, Ho-Ching Iu[2]*, Hairong Lin[3]* and Viet-Thanh Pham[4]*

[1]School of Physics and Electronic Science, Changsha University of Science and Technology, Changsha, China, [2]School of Electrical, Electronic and Computer Engineering, University of Western Australia, Perth, WA, Australia, [3]School of Electronic Information, Central South University, Changsha, China, [4]Faculty of Electrical and Electronics Engineering, Ton Duc Thang University, Ho Chi Minh City, Vietnam

KEYWORDS

editorial, nonlinear systems, nonlinear networks, nonlinear device, application

Editorial on the Research Topic
Advances in nonlinear systems and networks, volume III

EDITED AND REVIEWED BY
Alex Hansen,
NTNU, Norway

*CORRESPONDENCE
Fei Yu,
✉ yufeiyfyf@csust.edu.cn
Ho-Ching Iu,
✉ herbert.iu@uwa.edu.au
Hairong Lin,
✉ haironglin@csu.edu.cn
Viet-Thanh Pham,
✉ phamvietthanh@tdtu.edu.vn

RECEIVED 19 August 2025
ACCEPTED 27 August 2025
PUBLISHED 03 September 2025

CITATION
Yu F, Iu H-C, Lin H and Pham V-T (2025) Editorial: Advances in nonlinear systems and networks, volume III.
Front. Phys. 13:1688722.
doi: 10.3389/fphy.2025.1688722

1 Introduction

Nonlinear systems and networks theory is a branch of automatic control theory. It takes systems and networks described by nonlinear differential equations or difference equations as its research objects, focusing on their motion laws and analysis methods, and belongs to the field of physics [1–5]. Its core feature is the failure of the superposition principle, and it mainly studies complex phenomena such as self-excited oscillation, frequency-dependent amplitude, multi-valued response, bifurcation and chaos [6–10].

In recent years, with the development and application of emerging technologies such as big data [11, 12], cloud computing [13, 14], Internet of Things [15–17], and datacentral networks [18–20], nonlinear technologies have gradually shifted from system modeling to intelligent computing [21–24]. At present, nonlinear systems and networks have been deeply studied and applied in the following fields, such as chaotic systems [25–27], neural networks [28–32], memristors [33–35], neural circuits [36–38], system synchronization [39, 40], and related application fields [41–47].

Due to the success of the first and second Research Topic of "Advances in Nonlinear Systems and Networks", we have decided to continue to focus on the ongoing progress of nonlinear systems and networks in the third volume. This Research Topic has published a total of 10 research papers, covering the latest research progress in areas such as adaptive iterative learning control, chaotic system modeling, memristor mathematical model, nonlinear circuit, and their applications.

2 Summary of papers presented in This Research Topic

Zhang et al. in the paper "FSE-RBFNN-based LPF-AILC of finite time complete tracking for a class of time-varying NPNL systems with initial state errors", proposed a low-pass filter adaptive iterative learning control (LPF-AILC) strategy is proposed. The authors combined the Radial basis function neural network (RBFNN) with the Fourier Series expansion (FSE) and proposed a new function approximator (FSE-RBFNN) to model various time-varying nonlinear parametric functions. To mitigate the influence of the initial state error, the article introduces a dynamically changing boundary layer and a series of methods for dealing with the upper bound of unknown errors. Finally, the correctness of the proposed control method was verified through two simulation examples.

Yu et al. in the paper "New discrete memristive hyperchaotic map: modeling, dynamic analysis, and application in image encryption", by coupling the upgraded cosine discrete memristor with the Cubic mapping, a new type of discrete memristor hyperchaotic mapping is constructed. Then, the dynamic characteristics of the system are deeply analyzed. Subsequently, based on the proposed hyperchaotic mapping, the paper presents a new image encryption scheme, effectively scrambling and diffusing the image data. During the diffusion process, a new forward and reverse diffusion strategy is introduced, which improves the encryption efficiency. Finally, through relevant security analysis, it is found that this scheme has high security and practicability.

Wang et al. in the paper "Monophasic and biphasic neurodynamics of bi-S-type locally active memristor", proposed an artificial memristive neuron was proposed to reproduce the function of biological neurons. By using the Chua expansion theorem, the authors established a mathematical model of a double S-type local active memristor with negative differential resistance (NDR). Subsequently, the paper constructed a second-order neural circuit to simulate periodic spikes and excitability. In addition, the constructed neuron circuits generate biphasic action potentials through voltage-symmetric modulation, replicating the bidirectional signal transmission similar to that of biological systems. Finally, hardware simulation verified the neural dynamics under different stimuli.

Gao et al. in the paper "Analysis and Application for the Source-free R M LC Circuits", studied the passive circuit topologies of four types of integrated memristors and energy storage components. Firstly, through mathematical modeling, the authors discovered that all four circuits are controlled by transcendental equations. Secondly, two types of four-component passive circuits were configured and analyzed. It was concluded that the capacitor and inductor provide energy for the system, while the memristor exhibits hysteresis behavior. Finally, the paper configured and discussed the application circuit. Research shows that even within the same circuit, different placement positions of memristors can lead to different topological structures and different nonlinear output behaviors.

Wang et al. in the paper "An echo state network based on enhanced intersecting cortical model for discrete chaotic system prediction", proposed an echo state network framework based on the Enhanced Intersecting Cortical Model (ESN-EICM). This model introduces a neuron model with internal dynamics (including adaptive thresholds and interneuron feedback) into the reservoir structure. The paper compares the performance of the ESN-EICM network with that of the standard ESN and long short-term memory (LSTM) networks. The experimental results show that in the test system, compared with the standard ESN and LSTM models, the ESN-EICM model generates lower error metrics (MSE, RMSE, MAE), and the performance difference is more obvious in multi-step prediction scenarios.

Tu et al. in the paper "Child information protection scheme based on hyperchaotic mapping", proposed an encryption scheme based on hyperchaotic mapping. Firstly, the authors plotted the phase diagrams of the hyperchaotic mapping under different parameter combinations. The changes in the phase trajectories confirmed the sensitivity of the hyperchaotic mapping to the control parameters. Then, the paper iterates on the hyperchaotic mapping to obtain a chaotic sequence and quantizes the chaotic sequence to obtain a pseudo-random sequence. Finally, on this basis, scrambling algorithms and diffusion algorithms were designed to encrypt and protect the images, which are used to protect the information of missing children and can effectively protect the information security of children.

Chen et al. in the paper "A novel image encryption method based on improved two-dimensional logistic mapping and DNA computing", proposed an innovative image encryption method, eliminating the security limitations of traditional one-dimensional logical mapping. Firstly, the article utilizes the improved two-dimensional Logistic-fractional mixed chaotic map (2D-LFHCM) to effectively shuffle the images by merging chaotic sequences. Then, two new algebraic deoxyribonucleic acid (DNA) calculation rules were introduced to enhance diffusion encryption. The experimental results show that this method provides superior security performance.

Zhang et al. in the paper "Grid Image Encryption Based on 4D Memristive Sprott K Chaotic Sequence", proposed an image encryption algorithm for smart grids based on chaotic systems. Firstly, the authors adopted the 4D memristive Sprott K system to generate chaotic sequences as the encryption key stream; Secondly, the article uses a dual encryption mechanism of scrambling and diffusion to scramble the positions of image pixels and replace their values, thereby enhancing the algorithm's anti-attack capability. The simulation results show that this algorithm can effectively protect the security and privacy of smart grid images.

Huang et al. in the paper "HiImp-SMI: an implicit transformer framework with high-frequency adapter for medical image segmentation", studied an implicit transformer framework for medical image segmentation with a high-frequency adapter (HIPP-SMI). The authors have designed a new dual-branch structure that simultaneously processes spatial and frequency information. Experimental evaluations show that on the Kvasir-Sessile and BCV datasets, HiImp-SMI consistently outperforms mainstream models. The framework proposed in the paper can serve as a flexible baseline for future work involving implicit modeling and multimodal representation learning in medical image analysis.

Dmitriev et al. in the paper "Self-organization of the stock exchange to the edge of a phase transition: empirical and theoretical studies", found segments in the hourly stock trading volume sequence of 3,000 listed company stocks, corresponding to the time when the stock exchange remained at the edge of the phase transition. The authors conducted theoretical arguments for this hypothesis and presented phenomenological models of the self-organization of stock exchanges at the first-order phase transition edge and the second-order phase transition edge. The practical significance of this study lies in the possibility of self-organization of stock exchanges to phase transition edge early warning, and it can be expanded in future research using high-frequency data.

3 Concluding remarks

The continuous release of this Research Topic marks that the application and development research of nonlinear systems and networks has entered a brand-new research space. As this field continues to develop, contributions from a broader range of research and applications will play a crucial role in shaping its future direction.

Finally, we would like to express our gratitude to all the authors of the 10 articles in this Research Topic for their outstanding contributions. Their research papers are all highly suitable for the scope of this Research Topic. In addition, we would also like to express our sincere gratitude to all the reviewers, editors and editorial staff of the journal Frontiers in Physics for their support.

Author contributions

FY: Conceptualization, Data curation, Formal Analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review and editing. H-CI: Conceptualization, Data curation, Formal Analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review and editing. HL: Conceptualization, Data curation, Formal Analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review and editing. V-TP: Conceptualization, Data curation, Formal Analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review and editing.

Funding

The author(s) declare that financial support was received for the research and/or publication of this article. This work was supported by the Natural Science Foundation of Hunan Province under grant 2025JJ50368, and by the Scientific Research Fund of Hunan Provincial Education Department under Grant 24A0248, and by the Guiding Science and Technology Plan Project of Changsha City under Grant kzd2501129.

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.

The author(s) declared that they were an editorial board member of Frontiers, at the time of submission. This had no impact on the peer review process and the final decision.

Generative AI statement

The author(s) declare that no Generative AI was used in the creation of this manuscript.

Any alternative text (alt text) provided alongside figures in this article has been generated by Frontiers with the support of artificial intelligence and reasonable efforts have been made to ensure accuracy, including review by the authors wherever possible. If you identify any issues, please contact us.

Publisher's note

References

1. Tennie F, Laizet S, Lloyd S, Magri L. Quantum computing for nonlinear differential equations and turbulence. *Nat Rev Phys* (2025) 7(4):220–30. doi:10.1038/s42254-024-00799-w

2. Jin J, Chen W, Ouyang A, Yu F, Liu H. A time-varying fuzzy parameter zeroing neural network for the synchronization of chaotic systems. *IEEE Trans Emerging Top Comput Intelligence* (2024) 8(1):364–76. doi:10.1109/tetci.2023.3301793

3. Feng W, Zhang K, Zhang J, Zhao X, Chen Y, Cai B, et al. Integrating fractional-order hopfield neural network with differentiated encryption: achieving high-performance privacy protection for medical images. *Fractal and Fractional* (2025) 9(7):426. doi:10.3390/fractalfract9070426

4. 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 Trans Cybernetics* (2020) 50(7):3195–207. doi:10.1109/tcyb.2019.2906263

5. Zhang Z, Zhu M, Ren X. Double center swarm exploring varying parameter neurodynamic network for non-convex nonlinear programming. *Neurocomputing* (2025) 619:129156. doi:10.1016/j.neucom.2024.129156

6. Wu H, Xue G, Bai H, Ren Z. A new modeling methodology for frequency-dependent hysteresis from the perspective of phase lag and amplitude attenuation. *Nonlinear Dyn* (2025) 113:7759–77. doi:10.1007/s11071-024-10531-z

7. Zhang G, Zang C, Friswell MI. Measurement of multivalued response curves of a strongly nonlinear system by exploiting exciter dynamics. *Mech Syst Signal Process* (2020) 140:106474. doi:10.1016/j.ymssp.2019.106474

8. Fang X, Lou J, Wang J, Chuang KC, Wu HM, Huang ZL. A self-excited bistable oscillator with a light-powered liquid crystal elastomer. *Int J Mech Sci* (2024) 271:109124. doi:10.1016/j.ijmecsci.2024.109124

9. Hua C, Cao X, Liao B. Real-time solutions for dynamic complex matrix inversion and chaotic control using ODE-based neural computing methods. *Comput Intelligence* (2025) 41:e70042. doi:10.1111/coin.70042

10. Yu F, Gracia YM, Guo R, Ying Z, Xu J, Yao W, et al. Dynamic analysis and application of 6D multistable memristive chaotic system with wide range of hyperchaotic states. *Axioms* (2025) 14(8):638. doi:10.3390/axioms14080638

11. Dai Z, Guo X. Investigation of E-Commerce security and data platform based on the era of big data of the internet of things. *Mobile Information Systems* (2024) 2022:3023298. doi:10.1155/2022/3023298

12. Vance TC, Huang T, Butler KA. Big data in Earth science: emerging practice and promise. *Science* (2024) 383(6688):eadh9607. doi:10.1126/science.adh9607

13. Lu J, Li W, Sun J, Xiao R, Liao B. Secure and real-time traceable data sharing in cloud-assisted IoT. *IEEE Internet Things J* (2024) 11(4):6521–36. doi:10.1109/jiot.2023.3314764

14. Hu J, Huang J, Li Z, Li Y, Rao S, Jiang W, et al. Proactive transport with high link utilization using opportunistic packets in cloud data centers. *IEEE Trans Mobile Comput (TMC)* (2025) 1–16. doi:10.1109/TMC.2025.3563182

15. Jin J, Wu M, Ouyang A, Li K, Chen C. A novel dynamic hill cipher and its applications on medical IoT. *IEEE Internet Things J* (2025) 12(10):14297–308. doi:10.1109/jiot.2025.3525623

16. Deng Q, Wang C, Yang G, Luo D. Discrete memristive delay feedback rulkov neuron model: chaotic dynamics, hardware implementation and application in secure communication. *IEEE Internet Things J* (2025) 12:25559–67. doi:10.1109/JIOT.2025.3558963

17. Yu F, Wu C, Xu S, Yao W, Xu C, Cai S, et al. Color video encryption transmission in IoT based on memristive hopfield neural network. *Signal Image Video Process.* (2025) 19(1):77. doi:10.1007/s11760-024-03697-x

18. Hu J, He Y, Luo W, Huang J, Wang J. Enhancing load balancing with In-network recirculation to prevent packet reordering in lossless data centers. *IEEE/ACM Trans Networking* (2024) 32(5):4114–27. doi:10.1109/tnet.2024.3403671

19. Hu J, Rao S, Zhu M, Huang J, Wang J, Wang J, et al. SRCC: Sub-RTT Congestion Control for Lossless Datacenter Networks. *IEEE Trans. Ind. Inform (Tii)* (2025) 21:2799–2808. doi:10.1109/TII.2024.3495759

20. Hu J, Zhi R, Wang J. ARS: adaptive routing system for heterogeneous traffic in industrial data centers. *IEEE Trans Ind Inform (Tii)* (2025) 21:6495–504. doi:10.1109/TII.2025.3567376

21. Zhang Z, He Y, Mai W, Luo Y, Li X, Cheng Y, et al. Convolutional dynamically convergent differential neural network for brain signal classification. *IEEE Trans Neural Networks Learn Syst* (2025) 36(5):8166–77. doi:10.1109/tnnls.2024.3437676

22. Hong Q, Jiang H, Xiao P, Du S, Li T. A parallel computing scheme utilizing memristor crossbars for fast corner detection and rotation invariance in the ORB algorithm. *IEEE Trans Comput* (2025) 74(3):996–1010. doi:10.1109/tc.2024.3504817

23. Xiang Q, Gong H, Hua C. A new discrete-time denoising complex neurodynamics applied to dynamic complex generalized inverse matrices. *The J Supercomputing* (2025) 81(1):159. doi:10.1007/s11227-024-06601-z

24. Xiao P, Fang J, Wei Z, Dong Y, Du S, Wen S, et al. A riccati matrix equation solver design based neurodynamics method and its application. *IEEE Trans Automation Sci Eng* (2025) 22:15163–76. doi:10.1109/tase.2025.3567981

25. Yu F, Tan B, He T, He S, Huang Y, Cai S, et al. A wide-range adjustable conservative memristive hyperchaotic system with transient quasi-periodic characteristics and encryption application. *Mathematics* (2025) 13(5):726. doi:10.3390/math13050726

26. Lai Q, Yang L, Chen G. Two-dimensional discrete memristive oscillatory hyperchaotic maps with diverse dynamics. *IEEE Trans Ind Elect* (2025) 72(1):969–79. doi:10.1109/tie.2024.3417974

27. Lai Q, Liu Y. A family of image encryption schemes based on hyperchaotic system and cellular automata neighborhood. *Sci China Technol Sci* (2025) 68(3):1320401. doi:10.1007/s11431-024-2678-7

28. Yu F, He S, Yao W, Cai S, Xu Q. Bursting firings in memristive hopfield neural network with image encryption and hardware implementation. *IEEE Trans Computer-Aided Des Integrated Circuits Syst* (2025) 1. doi:10.1109/TCAD.2025.3567878

29. Yu F, Su D, He S, Wu Y, Zhang S, Yin H. Resonant tunneling diode cellular neural network with memristor coupling and its application in police forensic digital image protection. *Chin Phys B* (2025) 34(5):050502. doi:10.1088/1674-1056/adb8bb

30. He S, Yu F, Guo R, Zheng M, Tang T, Jin J, et al. . Dynamic Analysis and FPGA Implementation of a Fractional-Order Memristive Hopfield Neural Network with Hidden Chaotic Dual-Wing Attractors. *Fractal Fractional* (2025) 9:561. doi:10.3390/fractalfract9090561

31. Wang C, Li Y, Yang G, Deng Q. A review of fractional-order chaotic systems of memristive neural networks. *Mathematics* (2025) 13:1600. doi:10.3390/math13101600

32. Zhang S, Chen C, Zhang Y, Cai J, Wang X, Zeng Z. Multidirectional multidouble-scroll hopfield neural network with application to image encryption. *IEEE Trans Syst Man, Cybernetics: Syst* (2025) 55(1):735–46. doi:10.1109/tsmc.2024.3489226

33. Deng Q, Wang C, Sun Y, Yang G. Discrete memristive conservative chaotic map: dynamics, hardware implementation, and application in secure communication. *IEEE Trans Cybernetics* (2025) 55:3926–34. doi:10.1109/TCYB.2025.3565333

34. Yu F, He T, He S, Tan B, Shi C, Lin H. Influence of memristive activated gradient on chaotic dynamics in discrete neural networks. *Int J Bifurcation Chaos* (2025) 35:2550146. doi:10.1142/S0218127425501469

35. Peng Y, Li M, Li Z, Ma M, Wang M. What is the impact of discrete memristor on the performance of neural network: a research on discrete memristor-based BP neural network. *Neural Networks* (2025) 185:107213. doi:10.1016/j.neunet.2025.107213

36. Xu Q, Wang K, Feng C, Fan W, Wang N. Dynamical effects of low-frequency and high-frequency current stimuli in a memristive morris–lecar neuron model. *Chaos, Solitons Fractals* (2024) 189:115646. doi:10.1016/j.chaos.2024.115646

37. Wang C, Li Y, Deng Q. Discrete-time fractional-order local active memristor-based hopfield neural network and its FPGA implementation. *Solitons Fractal* (2025) 193:116053. doi:10.1016/j.chaos.2025.116053

38. Ding X, Feng C, Wang N, Wu H, Xu Q. Firing activities induced by various stimuli in a memristive ion channel-based bionic circuit. *Chaos, Solitons and Fractals* (2025) 199:116587. doi:10.1016/j.chaos.2025.116587

39. Tan F, Zhou L, Zong G, Wang Z, Zhuang G, Shangguan X. Fixed time control for interlayer synchronism under DDMNNs and application in secure communication. *IEEE Trans Automation Sci Eng* (2025) 22(1):10827–34. doi:10.1109/TASE.2025.3528957

40. Zhou L, Lin H, Tan F. Fixed/predefined-time synchronization of coupled memristor-based neural networks with stochastic disturbance. *Chaos Solitons Fractals* (2023) 173:113643. doi:10.1016/j.chaos.2023.113643

41. Zhu J, Jin J, Chen C, Wu L, Lu M, Ouyang A. A new-type zeroing neural network model and its application in dynamic cryptography. *IEEE Trans Emerging Top Comput Intelligence* (2025) 9(1):176–91. doi:10.1109/tetci.2024.3425282

42. Gao S, Zhang Z, Iu HHC, Ding S, Mou J, Erkan U, et al. A parallel color image encryption algorithm based on a 2-D logistic-rulkov neuron map. *IEEE Internet Things J* (2025) 12(11):18115–24. doi:10.1109/jiot.2025.3540097

43. Li H, Yu S, Feng W, Chen Y, Zhang J, Qin Z, et al. Exploiting dynamic vector-level operations and a 2D-enhanced logistic modular map for efficient chaotic image encryption. *Entropy* (2023) 25(8):1147. doi:10.3390/e25081147

44. Deng Q, Wang C, Sun Y, Cong X, Lin H, Deng Z. Memristor-based brain emotional learning neural network with attention mechanism and its application. *IEEE Trans Computer-Aided Des Integrated Circuits Syst* (2025) 1. doi:10.1109/TCAD.2025.3567534

45. Gao S, Ho-Ching Iu H, Erkan U, Simsek C, Toktas A, Cao Y, et al. A 3D memristive cubic map with dual discrete memristors: design, implementation, and application in image encryption. *IEEE Trans Circuits Syst Video Tech* (2025) 35(8):7706–18. doi:10.1109/tcsvt.2025.3545868

46. Feng W, Zhang J, Chen Y, Qin Z, Zhang Y, Ahmad M, et al. Exploiting robust quadratic polynomial hyperchaotic map and pixel fusion strategy for efficient image encryption. *Expert Syst Appl* (2024) 246:123190. doi:10.1016/j.eswa.2024.123190

47. Deng Q, Wang C, Sun Y, Yang G. Delay difference feedback memristive map: dynamics, hardware implementation, and application in path planning. *IEEE Trans Circuits Syst Regular Pap* (2025) 1–11. doi:10.1109/TCSI.2025.3571961

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 21 August 2024
DOI 10.3389/fphy.2024.1442486

OPEN ACCESS

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

REVIEWED BY
Jinping Jia,
Tianshui Normal University, China
Yichao Yan,
University of Electronic Science and Technology of China, China
Njitacke Tabekoueng Zeric,
University of Buea, Cameroon

*CORRESPONDENCE
Junliang Yao,
✉ yaojunliang@xaut.edu.cn

RECEIVED 02 June 2024
ACCEPTED 02 August 2024
PUBLISHED 21 August 2024

CITATION
Zhang C, Yan L, Gao Y, Yao J and Qian F (2024) FSE-RBFNN-based LPF-AILC of finite time complete tracking for a class of time-varying NPNL systems with initial state errors.
Front. Phys. 12:1442486.
doi: 10.3389/fphy.2024.1442486

FSE-RBFNN-based LPF-AILC of finite time complete tracking for a class of time-varying NPNL systems with initial state errors

Chunli Zhang[1,2], Lei Yan[1,2], Yangjie Gao[1,2], Junliang Yao[1,2]* and Fucai Qian[1,2]

[1]School of Automation and Information Engineering, Xi'an University of Technology, Xi'an, China, [2]Shaanxi Key Laboratory of Complex System Control and Intelligent Information Processing, Xi'an University of Technology, Xi'an, China

The paper proposes a low-pass filter adaptive iterative learning control (LPF-AILC) strategy for unmatched, uncertain, time-varying, non-parameterized nonlinear systems (NPNL systems). To address the difficulty of nonlinear parameterization terms in system models, a new function approximator (FSE-RBFNN), which combines the radial basis function neural network (RBFNN) and Fourier series expansion (FSE), is introduced to model each time-varying nonlinear parameterized function. The adaptive backstepping method is used to design control laws and parameter adaptive laws. In the process of controller design, we may encounter the problem of too many derivatives, which can cause parameter explosions after derivatives. Therefore, we introduce a first-order low-pass filter to solve this problem and simplify the structure of the controller. As the number of iterations increases, the maximum tracking error gradually decreases until it converges to the nearby region, approaching zero within the entire given interval $[0, T]$, according to the Lyapunov-like synthesis. To mitigate the impact of initial state errors, a dynamically changing boundary layer is introduced, along with a series to deal with the unknown error upper bounds. Finally, two simulation examples prove the correctness of the proposed control method.

KEYWORDS

adaptive iterative learning control, time-varying non-parameterized nonlinear systems, backstepping method, Fourier series expansion-radial basis function neural network, initial state errors, low-pass filter

1 Introduction

Adaptive iterative learning control (AILC) is a useful control strategy for solving repetitive tracking control task problems for uncertain nonlinear systems. It continuously adjusts its control algorithm through iterative learning to gradually approach the ideal trajectory of the unknown system. AILC has extensive application value and promising development prospects for practical applications. Repeat systems include uncertain robotic manipulators and uncertain hard disk drivers. The task requirements specify that it can quickly achieve exact tracking as the number of iterations increases [1–4].

A non-parameterized nonlinear (NPNL) system refers to a dynamic characteristic that exhibits a complex nonlinear relationship and unknown parameters, making it difficult to design effective control strategies. It is particularly challenging to achieve high-precision

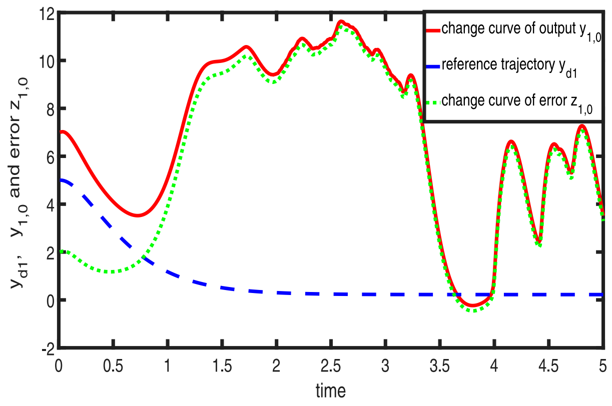


FIGURE 1
Variation in $y_0, y_{d1}, z_{1,0}$ over time without iteration.

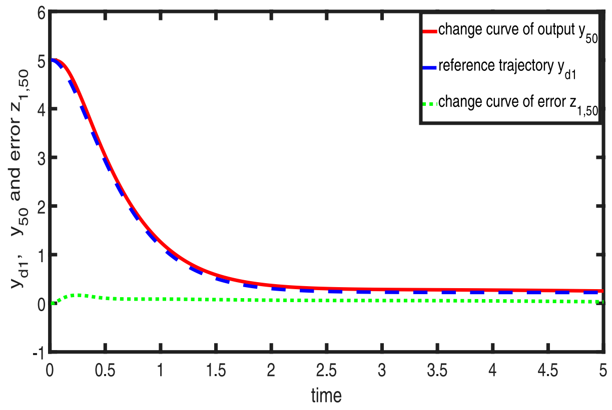


FIGURE 2
Variation in $y_{50}, y_{d1}, z_{1,50}$ over time during the 50th iteration.

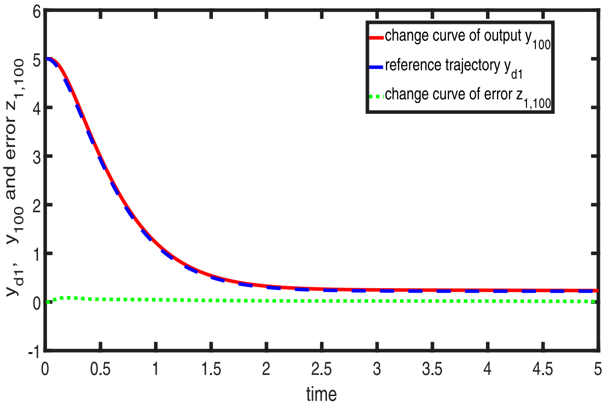


FIGURE 3
Variation in $y_{100}, y_{d1}, z_{1,100}$ over time during the 100th iteration.

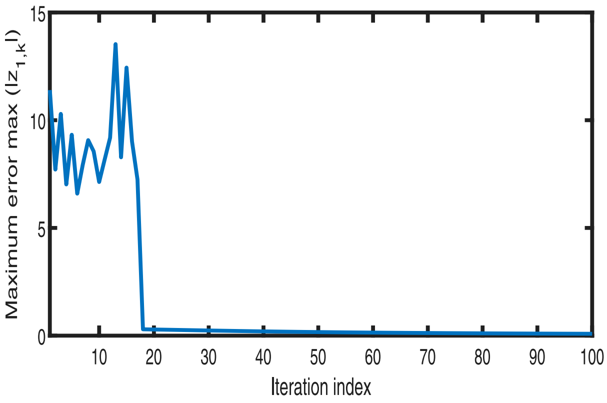


FIGURE 4
Variation in $max(|z_{1,k}|)$ according to the iteration index.

tracking and control within a limited time frame. Traditional control methods often require the establishment of a mathematical model for the system, but for the NPNL system, this step is usually very difficult or even impossible to complete. AILC technology has become an important method for solving these problems [5, 6].

There are many challenging problems in the research of AILC. This paper considers three difficult problems of AILC. The first problem is the processing problem of uncertain nonlinear parameterization terms with time-varying parameters. In the field of control, the control problem of nonlinear systems with uncertain time-varying parameters is very challenging. Adaptive control and robust control are common methods to deal with uncertain problems [7, 8]. Through learning, adaptive control can mitigate the impact of uncertainties. In order to handle uncertain nonlinear terms, adaptive control is often combined with some approximation methods, such as neural networks (NNs) and Fuzzy Logic Systems (FLSs). However, these adaptive controls only solve the uncertain linearly parameterized disturbances and ensure the stability of the system [7–20]. For the uncertain system, a fuzzy AILC was presented [21]. The composite energy function–adaptive iterative learning control (CEF–AILC) is an effective scheme for systems with time-varying disturbances [21–23]. Few AILC research results focus on uncertain, non-parameterized nonlinear systems [24–26]. Specifically, for systems with non-separable time-varying parameters, the tracking control problem on finite time intervals is still an open problem.

The second problem of AILC is ensuring complete tracking over a finite time interval when the initial state has errors. In these studies [27–31], the stability analysis section requires that initial state errors be strictly zero. Although the research on this problem is well done in traditional D-type or P-type ILC [32–41], it has not been well solved based on Lyapunov analysis for AILC. Specifically, in the presence of an initial state error, ensuring the system's completion of accurate tracking tasks within a specified time frame presents a complex challenge. [39] solved the tracking control problem of the unmatched uncertain NPNL systems. [41] solved the tracking problem of a class of high-order nonlinear systems with random initial state shifts, which relaxes the requirement of initial

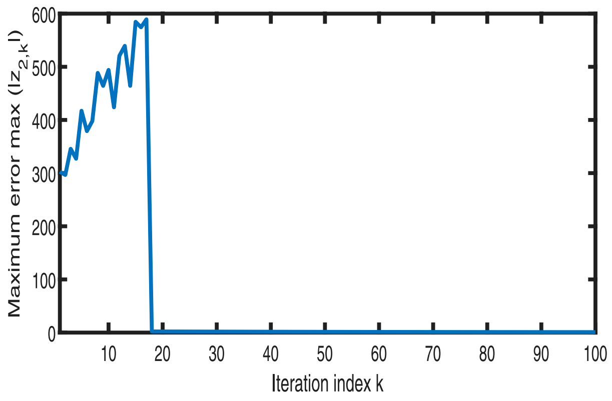


FIGURE 5
Variation in $max(|z_{2,k}|)$ according to the iteration index.

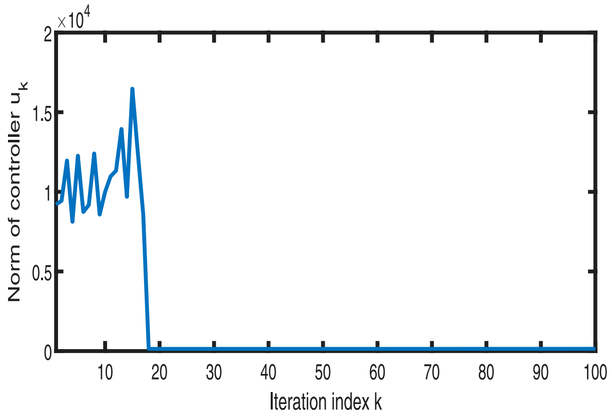


FIGURE 6
Variation in $\|u_k\|$ according to the iteration index.

positioning in ILC. So far, no relevant research results have been found for AILC applied to NPNL systems with uncertain time-varying parameters and initial state errors.

The last problem is parameter explosions after the derivative of the virtual controller. When designing a controller, we may encounter the problem of too many derivatives, which can cause parameter explosions after derivatives. Addressing this issue and streamlining the controller's structure to ensure the effective tracking of the non-parametric, nonlinear, time-varying system is a challenging and crucial problem. [42–44] employed a first-order low-pass filter to address the challenge of parameter explosions and achieve satisfactory performance. Therefore, we introduce a first-order low-pass filter to solve this problem and simplify the structure of the controller.

Motivated by the above discussion, we will use a low-pass filter AILC (LPF-AILC) method for uncertain time-varying NPNL systems. The AILC is given by the adaptive backstepping technique and Lyapunov-like theorem. In response to the difficult issues discussed above, the main contributions of this article are as follows:

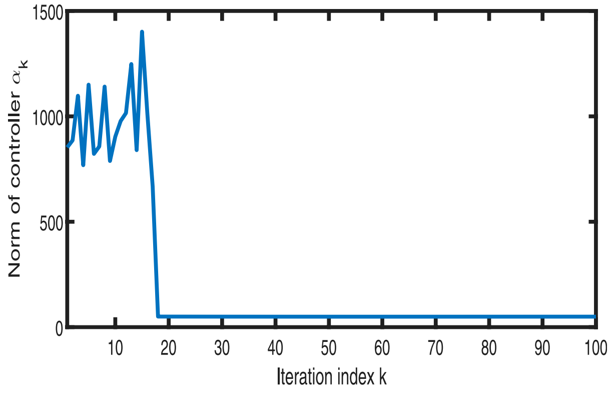


FIGURE 7
Variation in $\|\alpha_k\|$ according to the iteration index.

1) An LPF-AILC strategy is proposed for a class of strongly time-varying, non-parameterized, nonlinear systems combined with a new approximation method.
2) The processing problem of uncertain time-varying nonlinear parameterization terms was solved. This is a very important and difficult problem. Specifically, in the field of AILC, no relevant research results have been found.
3) The difficulty problem of AILC is ensuring complete tracking on a given interval when the initial state has errors.
4) The problem of parameter explosions was solved by applying a derivative to the virtual controller and simplifying its structure.

In this paper, a combination of Fourier series expansion and radial basis function neural network (RBFNN) (FSE-RBFNNs) is used to model the uncertain, time-varying nonlinear dynamics by using their uniform approximation [24, 38]. An updating time-varying boundary layer is used to design the error function to deal with the initial state error. A common convergence series sequence is employed to mitigate the impact of approximation errors on the control performance of the system. A low-pass filter was introduced to solve the problem of parameter explosions resulting from the derivative of the virtual controller and simplify the structure of the controller. Theoretical analysis can demonstrate the bounded nature of all signals within the closed-loop system. The maximum value of errors will gradually converge to a narrow range close to zero as the boundary layer width satisfies the convergence condition with the number of iterations. Finally, two simulation examples are given to prove the effectiveness and correctness of the control method.

2 Problem description and mathematical foundations

2.1 Problem description

Uncertain time-varying NPNL systems are considered:

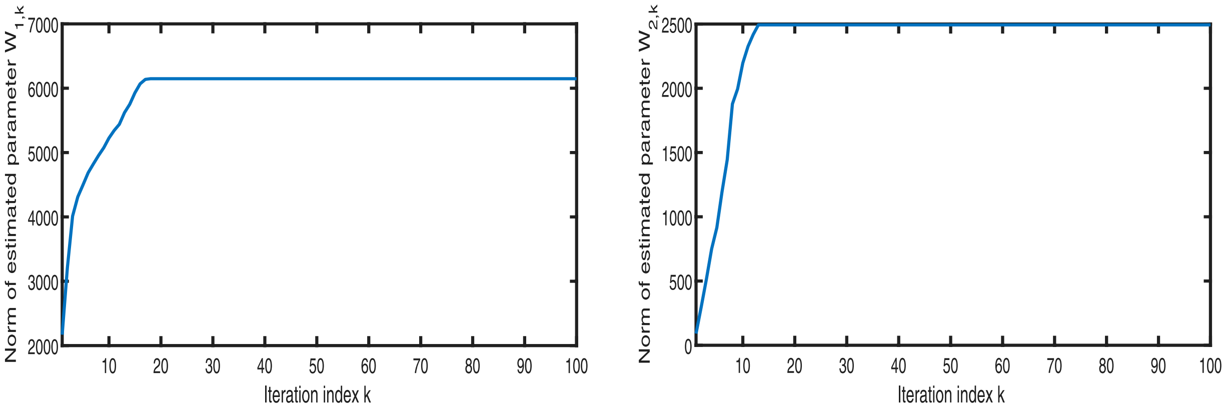


FIGURE 8
Variation in $\|\hat{W}_{1,k}\|$ and $\|\hat{W}_{2,k}\|$ according to the iteration index.

$$\begin{aligned} \dot{x}_{1,k} &= x_{2,k} + f_1\left(\bar{x}_{1,k}, \theta_1(t)\right) + g_1\left(\bar{x}_{1,k}\right) \\ \dot{x}_{i,k} &= x_{i+1,k} + f_i\left(\bar{x}_{i,k}, \theta_i(t)\right) + g_i\left(\bar{x}_{i,k}\right) \\ \dot{x}_{n,k} &= u_k + f_n\left(\bar{x}_{m,k}, \theta_n(t)\right) + g_n\left(\bar{x}_{n,k}\right) \\ y_k &= x_{1,k}, \end{aligned} \tag{1}$$

where $\bar{x}_{i,k} = [x_{1,k}, \ldots, x_{i,k}]^T \in R^i$ and $x = \bar{x}_n$ represents measurable state vectors. $u_k \in R$ is the control input. $y_k \in R$ is the system output. $f_i(\bar{x}_{i,k}, \theta_i(t))$, $g_i(\bar{x}_{i,k})$, and $i = 1, 2, \ldots, n$ are uncertain time-varying functions, and $\theta_i(t)$ represents unknown time-varying parameters. k denotes the iteration time.

The design objective of this article is to find $u_k(t)$ for system (1) to ensure that $y_k(t)$ follows the ideal trajectory $y_{d1}(t)$ on $[0, T]$.

2.2 Mathematical foundations

The mathematical knowledge used in this article is provided with relevant references, and the specific definitions and principles will not be elaborated. Here, we only provide the conclusions that need to be used in this article.

In system (1), the processing of unknown time-varying, nonlinear, parameterized function terms $f(\chi_k, \theta(t))$ is a challenge. Since the function $\theta(t)$ is not known, $\theta(t)$ is expanded using Fourier series as $\theta(t) = M^T\Phi(t) + \delta_\theta(t), \|\delta_\theta(t)\| \le \bar{\delta}_\theta$; based on this, uncertain time-varying nonlinear functions $f(\chi_k, \theta(t))$ can be approximated as

$$f(\chi_k, \theta_k(t)) = W_k^T S(\chi_k, M_k^T\Phi(t) + \delta_{\theta,k}) + \delta_{f,k}. \tag{2}$$

A new FSE-RBFNN approximator is built:

$$G(\chi_k, t) = W_k^T S(\chi_k, M_k^T\Phi(t)), \tag{3}$$

representing $f(\chi_k, \theta_k(t))$ as

$$f(\chi_k, \theta_k(t)) = W_k^T S(\chi_k, M_k^T\Phi(t)) + \delta_k(\chi_k, t), \tag{4}$$

where

$$\delta_k(\chi_k, t) = \delta_{f,k} + W_k^T S(\chi_k, M_k^T\Phi(t) + \delta_{\theta,k}) - W_k^T S(\chi_k, M_k^T\Phi(t)). \tag{5}$$

Assumption 1: In the compact domain Ω_k, the weights W_k and M_k are constrained, and $\|W_k\| \le w_{m,k}$ and $\|M_k\| \le m_{a,k}$ with $w_{m,k}, m_{a,k}$ being unknown positive numbers.

Lemma 1[38]: For $(\chi_k, \theta_k(t)) \in \Omega_k$, $\delta_k(\chi_k, t)$ in (5) is bound, and

$$|\delta_k(\chi_k, t)| \le \overline{\delta_k}, \tag{6}$$

where $\overline{\delta_k}$ represents the supremum of $\delta_k(\chi_k, t)$.

Because W_k and M_k are unknown, we estimate them with $\hat{W}_k$ and $\hat{M}_k$, respectively. $\tilde{W}_k = \hat{W}_k - W_k$ and $\tilde{M}_k = \hat{M}_k - M_k$ are estimation errors.

Lemma 2[38]: In the surrogate model (4), the following conclusion holds:

$$\begin{aligned} &W_k^T S(\chi_k, M_k^T\Phi(t)) - \hat{W}_k^T S\left(\chi_k, \hat{M}_k^T\Phi(t)\right) \\ &\quad = \tilde{W}_k^T\left(S\left(\chi_k, \hat{M}_k^T\Phi(t)\right) - \hat{S}_k'\hat{M}_k^T\Phi(t)\right) + \hat{W}_k^T\hat{S}_k'\tilde{M}_k^T\Phi(t) + d, \end{aligned} \tag{7}$$

where $\hat{S}_k' = [\hat{s}_{1,k}', \hat{s}_{2,k}', \ldots, \hat{s}_{p,k}'] \in \Re^{m\times p}$ with $\hat{s}_{i,k}' = (\partial s_i(\chi_k, \omega_k))/\partial\omega_k|_{\omega_k=\hat{M}_k\Phi(t)}$ and $i = 1, \ldots, p$, and the remainder d_k is bounded by

$$|d_k| \le \|M_k\|_F\|\Phi(t)\hat{W}_k^T\hat{S}_k'\|_F + \|W_k\|\|\hat{S}_k'\hat{M}_k^T\Phi(t)\| + |W_k|_1. \tag{8}$$

For the processing of the supremum of each error term, this article introduces the following typical series sequence:

Lemma 2[39] For a sequence $\Delta_k = \{\frac{1}{k^l}\}$, where $k = 1, 2, \cdots$ and $l \ge 2$, the following result exists:

$$lim_{k\to\infty}\Sigma_{i=1}^k\frac{1}{i^l} \le 2. \tag{9}$$

Assumption 2: The initial error value at the beginning of each iteration should meet $|z_{i,k}(0)| = \epsilon_{i,k}$ with $\epsilon_{i,k}$ being a convergence series sequence, where $i = 1, \ldots, n$.

Considering the initial errors, a new function $z_{\phi,k}^{[34]}$ is accepted:

$$\begin{aligned} z_{\phi,k} &= z_k - \phi_k(t)sat\left(\frac{z_k}{\phi_k(t)}\right) \\ \phi_k(t) &= \epsilon_k e^{-\eta t}, \end{aligned} \tag{10}$$

where *sat* is the saturation function given as

$$sat\left(\frac{z_k}{\phi_k(t)}\right)=\begin{cases}1 & \text{if} \quad z_k>\phi_k(t)\\ \frac{z_k}{\phi_k(t)} & \text{if} \quad z_k\le\phi_k(t)\\ -1 & \text{if} \quad z_k<-\phi_k(t)\end{cases},$$

with $\phi_k(t)$ being an updating time-varying boundary layer. When $lim_{k\to\infty}z_{\phi,k}=0$ and considering assumption 2 again, we have $lim_{k\to\infty}|z_k|=0$.

In order to prevent the problem of gradient explosion, we introduce the first-order low-pass filter β_k, which is given as follows:

$$\dot{\beta}_k=-\xi_k(\beta_k-\alpha_k), \tag{11}$$

where β_k results from filtering an instruction with α_k as its input, with α_k being the virtual controller, $\xi_k>0$, and $\beta_k(0)=\alpha_k(0)$. Because part of α_k $\beta_k-\alpha_k$ cannot pass through the filter, an error compensation mechanism ζ_k is introduced to overcome the influence of the instruction filter. Therefore, a new function Z_k is introduced as follows:

$$Z_k=z_{\phi,k}-\zeta_k. \tag{12}$$

3 AILC design

Based on the above mathematical foundations, we present the specific controller design process.

3.1 Designing the AILC controller

Step 1: Denote $N_1=\omega_{M1}^2$, which will be defined later. $z_{1,k}=x_{1,k}-y_{d1}$ and $z_{2,k}=x_{2,k}-\alpha_{1,k}$, where $\alpha_{1,k}$ is the virtual controller. Because the initial state values of the system have errors and gradient explosion, the new error functions $Z_{1,k}$ and $Z_{2,k}$ are given as

$$\begin{aligned}Z_{1,k}&=z_{1\phi,k}-\zeta_{1,k}\\ z_{1\phi,k}&=z_{1,k}-\phi_{1,k}(t)sat\left(\frac{z_{1,k}}{\phi_{1,k}(t)}\right)\\ z_{1,k}&=x_{1,k}-y_{d1}\\ \phi_{1,k}(t)&=\epsilon_{1,k}e^{-\eta_1 t},\end{aligned} \tag{13}$$

$$\begin{aligned}Z_{2,k}&=z_{2\phi,k}-\zeta_{2,k}\\ z_{2\phi,k}&=z_{2,k}-\phi_{2,k}(t)sat\left(\frac{z_{2,k}}{\phi_{2,k}(t)}\right)\\ z_{2,k}&=x_{2,k}-\beta_{1,k}\\ \phi_{2,k}(t)&=\epsilon_{2,k}e^{-\eta_2 t}.\end{aligned} \tag{14}$$

We recall that

$$\dot{x}_{1,k}=x_{2,k}+f_1(\bar{x}_{1,k},\theta_1(t))+g_1(\bar{x}_{1,k}). \tag{15}$$

Given the derivative of $z_{1\phi,k}$,

$$\begin{aligned}\dot{z}_{1\phi,k}&=\begin{cases}\dot{z}_{1,k}-\dot{\phi}_{1,k} & \text{if} \quad z_{1,k}>\phi_{1,k}(t)\\ 0 & \text{if} \quad z_{1,k}\le\phi_{1,k}(t)\\ \dot{z}_{1,k}+\dot{\phi}_{1,k} & \text{if} \quad z_{1,k}<-\phi_{1,k}(t)\end{cases}\\ &=\dot{z}_{1,k}-sgn\big(z_{1\phi,k}(t)\big)\dot{\phi}_{1,k}\\ &=z_{2,k}+\beta_{1,k}+f_1(\bar{x}_{1,k},\theta_1(t))+g_1(\bar{x}_{1,k})\\ &\quad-\dot{y}_{d1}-sgn\big(z_{1\phi,k}\big)\dot{\phi}_{1,k}.\end{aligned} \tag{16}$$

Therefore, the derivative of $Z_{1,k}$ with respect to time is as follows:

$$\begin{aligned}\dot{Z}_{1,k}=&\ z_{2,k}+\beta_{1,k}+f_1(\bar{x}_{1,k},\theta_1(t))+g_1(\bar{x}_{1,k})\\ &-\dot{y}_{d1}-sgn\big(z_{1\phi,k}\big)\dot{\phi}_{1,k}-\dot{\zeta}_{1,k}.\end{aligned} \tag{17}$$

The error compensation mechanism is considered as follows:

$$\dot{\zeta}_{1,k}=\beta_{1,k}+\zeta_{2,k}-\eta_1\zeta_{1,k}-\alpha_{1,k}. \tag{18}$$

Using Equation 18, we can find the time derivative of the error function as follows:

$$\begin{aligned}\dot{Z}_{1,k}=&\ z_{2,k}-\zeta_{2,k}+\eta_{1,k}\zeta_{1,k}+\alpha_{1,k}-\dot{y}_{d1}\\ &+f_1(\bar{x}_{1,k},\theta_1(t))+g_1(\bar{x}_{1,k})-sgn\big(z_{1\phi,k}\big)\dot{\phi}_{1,k}.\end{aligned} \tag{19}$$

The unknown time-varying, nonlinear functions $f_1(\bar{x}_{1,k},\theta_1(t))$ and $g_1(\bar{x}_{1,k})$ may be approximated by FSE-RBFNN and RBFNN, respectively.

$$\begin{aligned}f_1(\bar{x}_{1,k},\theta_1(t))&=W_{f1}^TS_{f1}(\bar{x}_{1,k},M_1^T\phi_1(t))+\delta_{f1}\\ g_1(\bar{x}_{1,k})&=W_{g1}^TS_{g1}(\bar{x}_{1,k})+\delta_{g1},\end{aligned} \tag{20}$$

where δ_{f1} and δ_{g1} are the truncation errors after approximation and W_{f1} and W_{g1} are weight vectors.

Consider $\Delta_k=\frac{a}{k^l}$, $a>0$, and $l\ge2$. The virtual control law is designed as

$$\begin{aligned}\alpha_{1,k}=&-\hat{W}_{f1,k}^TS_{f1}\left(\bar{x}_{1,k},\hat{M}_{1,k}^T\Phi_1(t)\right)-\hat{W}_{g1,k}^TS_{g1}(\bar{x}_{1,k})\\ &-\hat{N}_{1,k}\frac{1}{\Delta_k}Z_{1,k}+\dot{y}_{d1}-\eta_1 z_{1,k}.\end{aligned} \tag{21}$$

By substituting Equations 20, 21 into Equation 19, we obtain

$$\begin{aligned}\dot{Z}_{1,k}=&\ z_{2,k}-\hat{N}_{1,k}\frac{1}{\Delta_k}Z_{1,k}-\zeta_{2,k}+\eta_{1,k}\zeta_{1,k}\\ &+W_{f1}^TS_{f1}(\bar{x}_{1,k},M_1^T\Phi_1(t))+\delta_{f1}-\hat{W}_{f1,k}^TS_{f1}\left(\bar{x}_{1,k},\hat{M}_{1,k}^T\Phi_1(t)\right)\\ &+W_{g1}^TS_{g1}(\bar{x}_{1,k})+\delta_{g1}-\hat{W}_{g1,k}^TS_{g1}(\bar{x}_{1,k})\\ &-\eta_1 z_{1,k}-sgn\big(z_{1\phi,k}\big)\dot{\phi}_{1,k}(t)\\ =&\ Z_{2,k}-\hat{N}_{1,k}\frac{1}{\Delta_k}Z_{1,k}-\zeta_{2,k}+\eta_{1,k}\zeta_{1,k}\\ &+W_{f1}^TS_{f1}(\bar{x}_{1,k},M_1^T\Phi_1(t))-\hat{W}_{f1,k}^TS_{f1}\left(\bar{x}_{1,k},\hat{M}_{1,k}^T\Phi_1(t)\right)\\ &+W_{g1}^TS_{g1}(\bar{x}_{1,k})-\hat{W}_{g1,k}^TS_{g1}(\bar{x}_{1,k})+\delta_{f1}+\delta_{g1}\\ &+\phi_{2,k}(t)sat\left(\frac{z_{2,k}}{\phi_{2,k}(t)}\right)+\zeta_{2,k}\\ &-\eta_1 z_{1,k}-sgn\big(z_{1\phi,k}\big)\dot{\phi}_{1,k}(t)\\ =&\ Z_{2,k}-\hat{N}_{1,k}\frac{1}{\Delta_k}Z_{1,k}+\phi_{2,k}(t)sat\left(\frac{z_{2,k}}{\phi_{2,k}(t)}\right)\\ &+W_{f1}^TS_{f1}(\bar{x}_{1,k},M_1^T\Phi_1(t))-\hat{W}_{f1,k}^TS_{f1}\left(\bar{x}_{1,k},\hat{M}_{1,k}^T\Phi_1(t)\right)\\ &+W_{g1}^TS_{g1}(\bar{x}_{1,k})-\hat{W}_{g1,k}^TS_{g1}(\bar{x}_{1,k})+\delta_{f1}+\delta_{g1}\\ &-\eta_1 z_{1,k}-sgn\big(z_{1\phi,k}\big)\dot{\phi}_{1,k}(t)+\eta_{1,k}\zeta_{1,k},\end{aligned} \tag{22}$$

where $\hat{W}_{f1,k}$, $\hat{W}_{g1,k}$, $\hat{M}_{1,k}$, and $\hat{N}_{1,k}$ are estimations of W_{f1}, W_{g1}, M_1, and N_1, respectively. $\tilde{W}_{f1,k}=\hat{W}_{f1,k}-W_{f1}$, $\tilde{W}_{g1,k}=\hat{W}_{g1,k}-W_{g1}$, $\tilde{M}_{1,k}=\hat{M}_{1,k}-M_1$, and $\tilde{N}_{1,k}=\hat{N}_{1,k}-N_1$ are the estimation errors. It can be proved that the following result is correct.

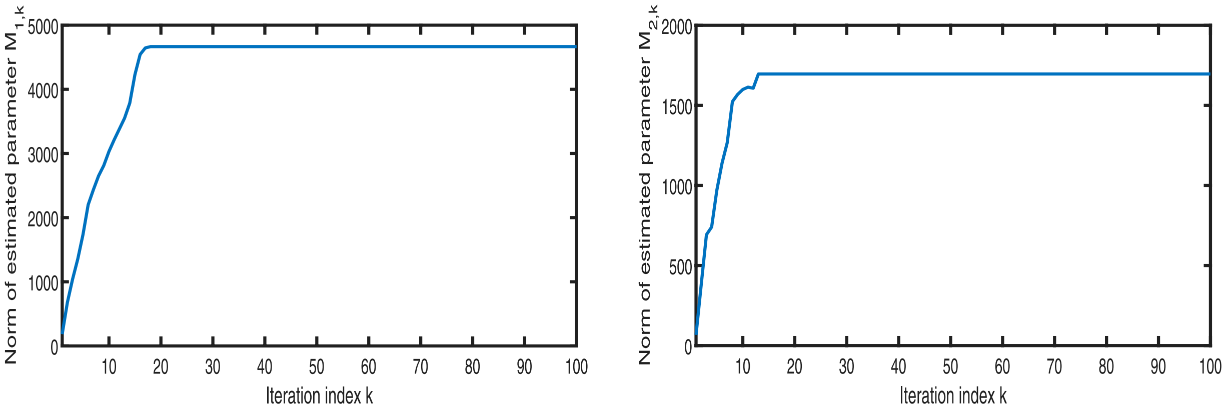


FIGURE 9
Variation in $\|\hat{M}_{1,k}\|$ and $\|\hat{M}_{2,k}\|$ according to the iteration index.

$$
\begin{aligned}
-\eta_1 z_{1,k} - sgn\left(z_{1\phi,k}\right)\dot{\phi}_{1,k}(t) + \eta_1 \zeta_{1,k} &= -\eta_1 z_{1\phi,k} - \eta_1 \phi_{1,k}(t) sat\left(\frac{z_{1,k}}{\phi_{1,k}(t)}\right) \\
&\quad - sgn\left(z_{1\phi,k}\right)\dot{\phi}_{1,k}(t) + \eta_1 \zeta_{1,k} \\
&= -\eta_1 z_{1\phi,k} + \eta_1 \zeta_{1,k} \\
&\quad - sgn\left(z_{1\phi,k}\right)\left(\dot{\phi}_{1,k}(t) + \eta_1 \phi_{1,k}(t)\right) \\
&= -\eta_1\left(z_{1\phi,k} - \zeta_{1,k}\right) \\
&= -\eta_1 Z_{1,k}.
\end{aligned}
\tag{23}
$$

Using Equations 7, 23, Equation 22 can be rewritten as

$$
\begin{aligned}
\dot{Z}_{1,k} = {} & Z_{2,k} - \hat{N}_{1,k}\frac{1}{\Delta_k}Z_{1,k} - \eta_1 Z_{1,k} \\
& + \tilde{W}_{f1}^{T}\left(S_{f1}\left(\bar{x}_{1,k}, \hat{M}_{1,k}^{T}\Phi_1(t)\right) - \hat{S}'_{f1,k}\hat{M}_{1,k}^{T}\Phi_1(t)\right) \\
& + \hat{W}_{f1,k}^{T}\hat{S}'_{f1,k}\tilde{M}_{1,k}^{T}\Phi_1(t) - \tilde{W}_{g1}^{T}S_{g1}(\bar{x}_{1,k}) \\
& + d_1 + \delta_{f1} + \delta_{g1} + \phi_{2,k}(t) sat\left(\frac{z_{2,k}}{\phi_{2,k}(t)}\right).
\end{aligned}
\tag{24}
$$

Let $\omega_1 = d_1 + \delta_{f1} + \delta_{g1} + \phi_{2,k}(t) sat\left(\frac{z_{2,k}(t)}{\phi_{2,k}(t)}\right)$, where d_1 is the remaining term of the estimation error after FSE-RBFNN expansion, and d_i is also the same; then, Equation 24 becomes

$$
\begin{aligned}
\dot{Z}_{1,k} = {} & Z_{2,k} - \hat{N}_{1,k}\frac{1}{\Delta_k}Z_{1,k} - \eta_1 Z_{1,k} + \omega_1 \\
& + \tilde{W}_{f1}^{T}\left(S_{f1}\left(\bar{x}_{1,k}, \hat{M}_{1,k}^{T}\Phi_1(t)\right) - \hat{S}'_{f1,k}\hat{M}_{1,k}^{T}\Phi_1(t)\right) \\
& + \hat{W}_{f1,k}^{T}\hat{S}'_{f1,k}\tilde{M}_{1,k}^{T}\Phi_1(t) - \tilde{W}_{g1}^{T}S_{g1}(\bar{x}_{1,k}).
\end{aligned}
\tag{25}
$$

Assumption 3 The remainder $\omega_i = d_i + \delta_{fi} + \delta_{gi} + \phi_{i+1,k}(t) sat\left(\frac{z_{i+1,k}(t)}{\phi_{i+1,k}(t)}\right)$ $(i = 1, 2, \ldots, n-1)$ is bounded with $|\omega_i| \le \omega_{Mi}$ and $\omega_{Mi} > 0$.

Remark 1: This assumption is easily satisfied because 1) d_i, δ_{fi}, and δ_{gi} are bounded and 2) when η_i is large enough, $\phi_{i,k}(t) sat\left(\frac{z_{i,k}(t)}{\phi_{i,k}(t)}\right)$ is sufficiently small.

The Lyapunov-like function is chosen as follows:

$$
\begin{aligned}
V_{1,k} = {} & \frac{1}{2}Z_{1,k}^2 + \frac{1}{2}\tilde{W}_{f1,k}^{T}\Gamma_{f11}^{-1}\tilde{W}_{f1,k} + \frac{1}{2}\tilde{W}_{g1,k}^{T}\Gamma_{g11}^{-1}\tilde{W}_{g1,k} \\
& + \frac{1}{2}\tilde{M}_{1,k}^{T}\Gamma_{m11}^{-1}\tilde{M}_{1,k} + \frac{1}{2}\Gamma_{n11}^{-1}\tilde{N}_{1,k}^{2},
\end{aligned}
\tag{26}
$$

where Γ_{f11}, Γ_{g11}, Γ_{m11}, and Γ_{n11} are adjustable matrices, each being positive, definite, and symmetric. Consider the derivative of $V_{1,k}$ by system (25), we obtain

$$
\begin{aligned}
\dot{V}_{1,k} = {} & Z_{1,k}Z_{2,k} - \eta_1 Z_{1,k}^2 \\
& + \tilde{W}_{f1,k}^{T}\Gamma_{f11}^{-1}\left(\Gamma_{f11}\left(S_{f1}\left(\bar{x}_{1,k}, \hat{M}_{1,k}^{T}\Phi_1(t)\right) - \hat{S}'_{f1,k}\hat{M}_{1,k}^{T}\Phi_1(t)\right)Z_{1,k}\right. \\
& \left. + \dot{\hat{W}}_{f1,k}\right) - \tilde{W}_{g1,k}^{T}\Gamma_{g11}^{-1}\left(\Gamma_{g11}S_{g1}(\bar{x}_{1,k})Z_{1,k} - \dot{\hat{W}}_{g1,k}\right) \\
& + \tilde{M}_{1,k}^{T}\Gamma_{m11}^{-1}\left(\Gamma_{m11}\Phi_1(t)\hat{W}_{f1,k}^{T}\hat{S}'_{f1,k}Z_{1,k} + \dot{\hat{M}}_{1,k}\right) \\
& - \hat{N}_{1,k}\frac{1}{\Delta_k}Z_{1,k}^2 + \omega_{1,k}Z_{1,k} + \Gamma_{n11}^{-1}\tilde{N}_{1,k}\dot{\hat{N}}_{1,k} \\
\le {} & Z_{1,k}Z_{2,k} - \eta_1 Z_{1,k}^2 \\
& + \tilde{W}_{f1,k}^{T}\Gamma_{f11}^{-1}\left(\Gamma_{f11}\left(S_{f1}\left(\bar{x}_{1,k}, \hat{M}_{1,k}^{T}\Phi_1(t)\right) - \hat{S}'_{f1,k}\hat{M}_{1,k}^{T}\Phi_1(t)\right)Z_{1,k}\right. \\
& \left. + \dot{\hat{W}}_{f1,k}\right) - \tilde{W}_{g1,k}^{T}\Gamma_{g11}^{-1}\left(\Gamma_{g11}S_{g1}(\bar{x}_{1,k})Z_{1,k} - \dot{\hat{W}}_{g1,k}\right) \\
& + \tilde{M}_{1,k}^{T}\Gamma_{m11}^{-1}\left(\Gamma_{m11}\Phi_1(t)\hat{W}_{f1,k}^{T}\hat{S}'_{f1,k}Z_{1,k} + \dot{\hat{M}}_{1,k}\right) \\
& - \hat{N}_{1,k}\frac{1}{\Delta_k}Z_{1,k}^2 + \frac{1}{\Delta_k}\omega_{M1}^2 Z_{1,k}^2 + \frac{1}{4}\Delta_k + \Gamma_{n11}^{-1}\tilde{N}_{1,k}\dot{\hat{N}}_{1,k} \\
= {} & Z_{1,k}Z_{2,k} - \eta_1 Z_{1,k}^2 \\
& + \tilde{W}_{f1,k}^{T}\Gamma_{f11}^{-1}\left(\Gamma_{f11}\left(S_{f1}\left(\bar{x}_{1,k}, \hat{M}_{1,k}^{T}\Phi_1(t)\right) - \hat{S}'_{f1,k}\hat{M}_{1,k}^{T}\Phi_1(t)\right)Z_{1,k}\right. \\
& \left. + \dot{\hat{W}}_{f1,k}\right) - \tilde{W}_{g1,k}^{T}\Gamma_{g11}^{-1}\left(\Gamma_{g11}S_{g1}(\bar{x}_{1,k})Z_{1,k} - \dot{\hat{W}}_{g1,k}\right) \\
& + \tilde{M}_{1,k}^{T}\Gamma_{m11}^{-1}\left(\Gamma_{m11}\Phi_1(t)\hat{W}_{f1,k}^{T}\hat{S}'_{f1,k}Z_{1,k} + \dot{\hat{M}}_{1,k}\right) \\
& - \hat{N}_{1,k}\frac{1}{\Delta_k}Z_{1,k}^2 + \frac{1}{\Delta_k}N_{1,k}Z_{1,k}^2 + \frac{1}{4}\Delta_k + \Gamma_{n11}^{-1}\tilde{N}_{1,k}\dot{\hat{N}}_{1,k} \\
= {} & Z_{1,k}Z_{2,k} - \eta_1 Z_{1,k}^2 \\
& + \tilde{W}_{f1,k}^{T}\Gamma_{f11}^{-1}\left(\Gamma_{f11}\left(S_{f1}\left(\bar{x}_{1,k}, \hat{M}_{1,k}^{T}\Phi_1(t)\right) - \hat{S}'_{f1,k}\hat{M}_{1,k}^{T}\Phi_1(t)\right)Z_{1,k}\right. \\
& \left. + \dot{\hat{W}}_{f1,k}\right) - \tilde{W}_{g1,k}^{T}\Gamma_{g11}^{-1}\left(\Gamma_{g11}S_{g1}(\bar{x}_{1,k})Z_{1,k} - \dot{\hat{W}}_{g1,k}\right) \\
& + \tilde{M}_{1,k}^{T}\Gamma_{m11}^{-1}\left(\Gamma_{m11}\Phi_1(t)\hat{W}_{f1,k}^{T}\hat{S}'_{f1,k}Z_{1,k} + \dot{\hat{M}}_{1,k}\right) \\
& - \tilde{N}_{1,k}\Gamma_{n11}^{-1}\left(\Gamma_{n11}\frac{1}{\Delta_k}Z_{1,k}^2 - \dot{\hat{N}}_{1,k}\right) + \frac{1}{4}\Delta_k,
\end{aligned}
\tag{27}
$$

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

We choose

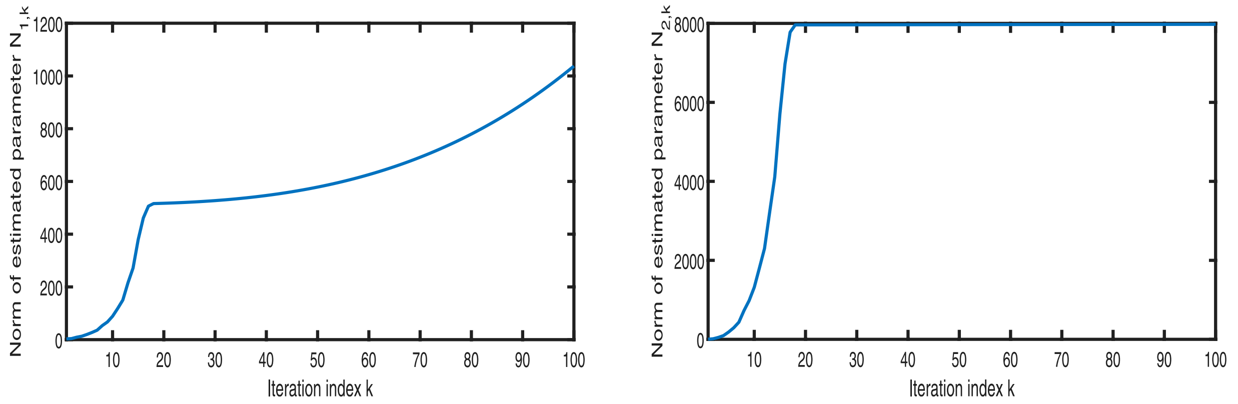


FIGURE 10
Variation in $\|\hat{N}_{1,k}\|$ and $\|\hat{N}_{2,k}\|$ according to the iteration index.

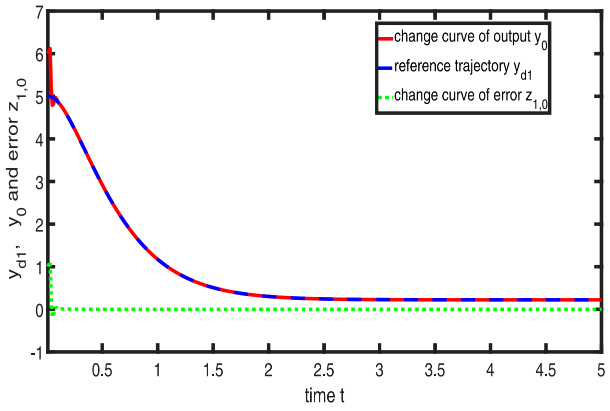


FIGURE 11
Variation in $y_0, y_{d1}, z_{1,0}$ over time without iteration.

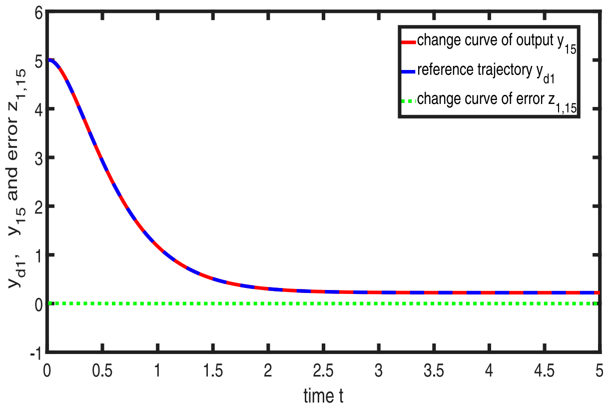


FIGURE 12
Variation in $y_{15}, y_{d1}, z_{1,15}$ over time during the 15th iteration.

$$\begin{aligned}
\dot{\hat{W}}_{f1,k} &= -\Gamma_{f11}\left(S_{f1}\left(\bar{x}_{1,k}, \hat{M}_{1,k}^{T}\Phi_1(t)\right) - \hat{S}'_{f1,k}\hat{M}_{1,k}^{T}\Phi_1(t)\right)Z_{1,k}\\
\dot{\hat{W}}_{g1,k} &= \Gamma_{g11}S_{g1}(\bar{x}_{1,k})Z_{1,k}\\
\dot{\hat{M}}_{1,k} &= -\Gamma_{m11}\Phi_1(t)\hat{W}_{f1,k}^{T}\hat{S}'_{f1,k}Z_{1,k}\\
\dot{\hat{N}}_{1,k} &= \Gamma_{n11}\frac{1}{\Delta_k}Z_{1,k}^2,
\end{aligned} \tag{28}$$

so Equation 27 becomes

$$\dot{V}_{1,k} \le Z_{1,k}Z_{2,k} - \eta_1 Z_{1,k}^2 + \frac{1}{4}\Delta_k. \tag{29}$$

Step 2: Denote $N_2 = \omega_{M2}^2$, which will be defined later. Due to initial state errors and gradient explosion, we introduce the following error function $Z_{3,k}$ as

$$\begin{aligned}
Z_{3,k} &= z_{3\phi,k} - \zeta_{3,k}\\
z_{3\phi,k} &= z_{3,k} - \phi_{3,k}(t)sat\left(\frac{z_{3,k}}{\phi_{3,k}(t)}\right)\\
z_{3,k} &= x_{3,k} - \beta_{2,k}\\
\phi_{3,k}(t) &= \epsilon_{3,k}e^{-\eta_3 t}.
\end{aligned} \tag{30}$$

The derivative of $Z_{2,k}$ is shown as follows:

$$\begin{aligned}
\dot{Z}_{2,k} &= \dot{z}_{2,k} - sgn\left(z_{2\phi,k}(t)\right)\dot{\phi}_{2,k} - \dot{\zeta}_{2,k}\\
&= z_{3,k} + \beta_{2,k} + f_2(\bar{x}_{2,k}, \theta_2(t)) + g_2(\bar{x}_{2,k})\\
&\quad -\dot{\beta}_{1,k} - sgn\left(z_{2\phi,k}\right)\dot{\phi}_{2,k} - \dot{\zeta}_{2,k}.
\end{aligned} \tag{31}$$

Let the error compensation mechanism be defined as follows:

$$\dot{\zeta}_{2,k} = \beta_{2,k} + \zeta_{3,k} - \eta_2\zeta_{2,k} - \zeta_{1,k} - \alpha_{2,k}. \tag{32}$$

Using Equation 32, we can find the time derivative of error function as

$$\begin{aligned}
\dot{Z}_{2,k} = {} & z_{3,k} - \zeta_{3,k} + \eta_{2,k}\zeta_{2,k} + \zeta_{1,k} + \alpha_{2,k} - \dot{\beta}_{1,k}\\
& + f_2(\bar{x}_{2,k}, \theta_2(t)) + g_2(\bar{x}_{2,k}) - sgn\left(z_{2\phi,k}\right)\dot{\phi}_{2,k}.
\end{aligned} \tag{33}$$

The uncertain time-varying, nonlinear functions $f_2(\bar{x}_{2,k}, \theta_2(t))$ and $G_2(\bar{x}_{2,k})$ are approximated by FSE-RBFNN and RBFNN, respectively.

$$\begin{aligned}
f_2(\bar{x}_{2,k}, \theta_2(t)) &= W_{f2}^{T}S_{f2}(\bar{x}_{2,k}, M_2^{T}\Phi_2(t)) + \delta_{f2}\\
G_2(\bar{x}_{2,k}) &= W_{g2}^{T}S_{g2}(\bar{x}_{2,k}) + \delta_{g2},
\end{aligned} \tag{34}$$

where δ_{f2} and δ_{g2} are reconstructed errors and W_{f2} and W_{g2} are optimal weight vectors.

Let the virtual control be defined as follows:

$$\begin{aligned}\alpha_{2,k} = {} & -\hat{W}_{f2,k}^T S_{f2}\left(\bar{x}_{2,k}, \hat{M}_{2,k}^T \Phi_2(t)\right) - \hat{W}_{g2,k}^T S_{g2}(\bar{x}_{2,k}) \\ & -\hat{N}_{2,k}\frac{1}{\Delta_k} Z_{2,k} + \dot{\beta}_{1,k} - \eta_2 z_{2,k} - z_{1\phi,k}.\end{aligned} \tag{35}$$

Substituting Equations 34, 35 into Equation 33, we obtain

$$\begin{aligned}\dot{Z}_{2,k} = {} & -z_{1\phi,k} + \zeta_{1,k} + z_{3,k} - \phi_{3,k}(t) sat\left(\frac{z_{3,k}}{\phi_{3,k}(t)}\right) - \zeta_{3,k} \\ & + W_{f2}^T S_{f2}\left(\bar{x}_{2,k}, M_2^T \Phi_2(t)\right) + \delta_{f2} - \hat{W}_{f2,k}^T S_{f2}\left(\bar{x}_{2,k}, \hat{M}_{2,k}^T \Phi_2(t)\right) \\ & + W_{g2}^T S_{g2}(\bar{x}_{2,k}) + \delta_{g2} - \hat{W}_{g2,k}^T S_{g2}(\bar{x}_{2,k}) \\ & - \hat{N}_{2,k}\frac{1}{\Delta_k} z_{2\phi,k} + \phi_{3,k}(t) sat\left(\frac{z_{3,k}}{\phi_{3,k}(t)}\right) \\ & - \eta_2 z_{2,k} + \eta_2 \zeta_{2,k} - sgn\left(z_{2\phi,k}\right)\dot{\phi}_{2,k} \\ = {} & -Z_{1,k} + Z_{3,k} - \hat{N}_{2,k}\frac{1}{\Delta_k} z_{2\phi,k} + \phi_{3,k}(t) sat\left(\frac{z_{3,k}}{\phi_{3,k}(t)}\right) \\ & + W_{f2}^T S_{f2}\left(\bar{x}_{2,k}, M_2^T \Phi_2(t)\right) + \delta_{f2} - \hat{W}_{f2,k}^T S_{f2}\left(\bar{x}_{2,k}, \hat{M}_{2,k}^T \Phi_2(t)\right) \\ & + W_{g2}^T S_{g2}(\bar{x}_{2,k}) + \delta_{g2} - \hat{W}_{g2,k}^T S_{g2}(\bar{x}_{2,k}) \\ & - \eta_2 z_{2,k} + \eta_2 \zeta_{2,k} - sgn\left(z_{2\phi,k}\right)\dot{\phi}_{2,k},\end{aligned} \tag{36}$$

where $\hat{W}_{f2,k}$, $\hat{W}_{g2,k}$, $\hat{M}_{2,k}$, and $\hat{N}_{2,k}$ are the estimators of W_{f2}, W_{g2}, M_2, and N_2, respectively. $\tilde{W}_{f2,k} = \hat{W}_{f2,k} - W_{f2}$, $\tilde{W}_{g2,k} = \hat{W}_{g2,k} - W_{g2}$, $\tilde{M}_{2,k} = \hat{M}_{2,k} - M_2$, and $\tilde{N}_{2,k} = \hat{N}_{2,k} - N_2$ are estimation errors. It can be proved that the following results are correct.

$$\begin{aligned}-\eta_2 z_{2,k} - sgn\left(z_{2\phi,k}\right)\dot{\phi}_{2,k}(t) + \eta_2 \zeta_{2,k} = {} & -\eta_2 z_{2\phi,k} - \eta_2 \phi_{2,k}(t) sat\left(\frac{z_{2,k}}{\phi_{2,k}(t)}\right) \\ & - sgn\left(z_{2\phi,k}\right)\dot{\phi}_{2,k}(t) + \eta_2 \zeta_{2,k} \\ = {} & -\eta_2 z_{2\phi,k} + \eta_2 \zeta_{2,k} \\ & - sgn\left(z_{2\phi,k}\right)\left(\dot{\phi}_{2,k}(t) + \eta_2 \phi_{2,k}(t)\right) \\ = {} & -\eta_2 z_{2\phi,k} + \eta_2 \zeta_{2,k} \\ = {} & -\eta_2\left(z_{2\phi,k} - \zeta_{2,k}\right) \\ = {} & -\eta_2 Z_{2,k}.\end{aligned} \tag{37}$$

Using Equations 7, 37, Equation 36 can be written as

$$\begin{aligned}\dot{Z}_{2,k} = {} & -Z_{1,k} + Z_{3,k} - \hat{N}_{2,k}\frac{1}{\Delta_k} z_{2\phi,k} - \eta_2 Z_{2,k} \\ & + \tilde{W}_{f2}^T\left(S_{f2}\left(\bar{x}_{2,k}, \hat{M}_{2,k}^T \Phi_2(t)\right) - \hat{S}'_{f2,k}\hat{M}_{2,k}^T \Phi_2(t)\right) \\ & + \hat{W}_{f2,k}^T \hat{S}'_{f2,k}\tilde{M}_{2,k}^T \Phi_2(t) - \tilde{W}_{g2}^T S_{g2}(\bar{x}_{2,k}) \\ & + d_2 + \delta_{f2} + \delta_{g2} + \phi_{3,k}(t) sat\left(\frac{z_{3,k}}{\phi_{3,k}(t)}\right).\end{aligned} \tag{38}$$

Let $\omega_2 = d_2 + \delta_{f2} + \delta_{g2} + \phi_{3,k}(t) sat\left(\frac{z_{3,k}(t)}{\phi_{3,k}(t)}\right)$, then Equation 38 becomes

$$\begin{aligned}\dot{Z}_{2,k} = {} & -Z_{1,k} + Z_{3,k} - \hat{N}_{2,k}\frac{1}{\Delta_k} Z_{2,k} - \eta_2 Z_{2,k} + \omega_2 \\ & + \tilde{W}_{f2,k}^T\left(S_{f2}\left(\bar{x}_{2,k}, \hat{M}_{2,k}^T \Phi_2(t)\right) - \hat{S}'_{f2,k}\hat{M}_{2,k}^T \Phi_2(t)\right) \\ & + \hat{W}_{f2,k}^T \hat{S}'_{f2,k}\tilde{M}_{2,k}^T \Phi_2(t) - \tilde{W}_{g2}^T S_{g2}(\bar{x}_{2,k}).\end{aligned} \tag{39}$$

The Lyapunov-like function was chosen as follows:

$$\begin{aligned}V_{2,k} = {} & V_{1,k} + \frac{1}{2}Z_{2,k}^2 + \frac{1}{2}\tilde{W}_{f2,k}^T \Gamma_{f21}^{-1}\tilde{W}_{f2,k} + \frac{1}{2}\tilde{W}_{g2,k}^T \Gamma_{g21}^{-1}\tilde{W}_{g2,k} \\ & + \frac{1}{2}\tilde{M}_{2,k}^T \Gamma_{m21}^{-1}\tilde{M}_{2,k} + \frac{1}{2}\Gamma_{n21}^{-1}\tilde{N}_{2,k}^2,\end{aligned} \tag{40}$$

where Γ_{f21}, Γ_{g21}, Γ_{m21}, and Γ_{n21} are adjustable, positive, definite, and symmetric matrices. According to Equation 39, Assumption 3, and Remark 1, $V_{2,k}$ can be expressed as

$$\begin{aligned}\dot{V}_{2,k} = {} & \dot{V}_{1,k} + Z_{2,k}\dot{Z}_{2,k} + \tilde{W}_{f2,k}^T \Gamma_{f21}^{-1}\dot{\hat{W}}_{f2,k} \\ & + \tilde{W}_{g2,k}^T \Gamma_{g21}^{-1}\dot{\hat{W}}_{g2,k} + \tilde{M}_{2,k}^T \Gamma_{m21}^{-1}\dot{\hat{M}}_{2,k} + \Gamma_{n21}^{-1}\tilde{N}_{2,k}\dot{\hat{N}}_{2,k} \\ \le {} & Z_{1,k}Z_{2,k} - \eta_1 Z_{1,k}^2 + \frac{1}{4}\Delta_k - Z_{1,k}Z_{2,k} + Z_{2,k}Z_{3,k} - \eta_2 Z_{2,k}^2 \\ & + \tilde{W}_{f2,k}^T \Gamma_{f21}^{-1}\left(\Gamma_{f21}\left(S_{f2}\left(\bar{x}_{2,k}, \hat{M}_{2,k}^T \Phi_2(t)\right) - \hat{S}'_{f2,k}\hat{M}_{2,k}^T \Phi_2(t)\right)Z_{2,k}\right. \\ & \left. + \dot{\hat{W}}_{f2,k}\right) - \tilde{W}_{g2,k}^T \Gamma_{g21}^{-1}\left(\Gamma_{g21}S_{g2}(\bar{x}_{2,k})Z_{2,k} - \dot{\hat{W}}_{g2,k}\right) \\ & + \tilde{M}_{2,k}^T \Gamma_{m21}^{-1}\left(\Gamma_{m21}\Phi_2(t)\hat{W}_{f2,k}^T \hat{S}'_{f2,k}Z_{2,k} + \dot{\hat{M}}_{2,k}\right) \\ & - \hat{N}_{2,k}\frac{1}{\Delta_k}Z_{2,k}^2 + \omega_2 Z_{2,k} + \Gamma_{n21}^{-1}\tilde{N}_{2,k}\dot{\hat{N}}_{2,k} \\ \le {} & -\eta_1 Z_{1,k}^2 + \frac{1}{4}\Delta_k + Z_{2,k}Z_{3,k} - \eta_2 Z_{2,k}^2 \\ & + \tilde{W}_{f2,k}^T \Gamma_{f21}^{-1}\left(\Gamma_{f21}\left(S_{f2}\left(\bar{x}_{2,k}, \hat{M}_{2,k}^T \Phi_2(t)\right) - \hat{S}'_{f2,k}\hat{M}_{2,k}^T \Phi_2(t)\right)Z_{2,k}\right. \\ & \left. + \dot{\hat{W}}_{f2,k}\right) - \tilde{W}_{g2,k}^T \Gamma_{g21}^{-1}\left(\Gamma_{g21}S_{g2}(\bar{x}_{2,k})Z_{2,k} - \dot{\hat{W}}_{g2,k}\right) \\ & + \tilde{M}_{2,k}^T \Gamma_{m21}^{-1}\left(\Gamma_{m21}\Phi_2(t)\hat{W}_{f2,k}^T \hat{S}'_{f2,k}Z_{2,k} + \dot{\hat{M}}_{2,k}\right) \\ & - \hat{N}_{2,k}\frac{1}{\Delta_k}Z_{2,k}^2 + \frac{1}{\Delta_k}\omega_{M2}^2 Z_{2,k}^2 + \frac{1}{4}\Delta_k + \Gamma_{n21}^{-1}\tilde{N}_{2,k}\dot{\hat{N}}_{2,k} \\ = {} & Z_{2,k}Z_{3,k} - \sum_{i=1}^{2}\eta_i Z_{i,k}^2 + \frac{1}{4}\Delta_k \\ & + \tilde{W}_{f2,k}^T \Gamma_{f21}^{-1}\left(\Gamma_{f21}\left(S_{f2}\left(\bar{x}_{2,k}, \hat{M}_{2,k}^T \Phi_2(t)\right) - \hat{S}'_{f2,k}\hat{M}_{2,k}^T \Phi_2(t)\right)Z_{2,k}\right. \\ & \left. + \dot{\hat{W}}_{f2,k}\right) - \tilde{W}_{g2,k}^T \Gamma_{g21}^{-1}\left(\Gamma_{g21}S_{g2}(\bar{x}_{2,k})Z_{2,k} - \dot{\hat{W}}_{g2,k}\right) \\ & + \tilde{M}_{2,k}^T \Gamma_{m21}^{-1}\left(\Gamma_{m21}\Phi_2(t)\hat{W}_{f2,k}^T \hat{S}'_{f2,k}Z_{2,k} + \dot{\hat{M}}_{2,k}\right) \\ & - \hat{N}_{2,k}\frac{1}{\Delta_k}Z_{2,k}^2 + \frac{1}{\Delta_k}N_{2,k}Z_{2,k}^2 + \frac{1}{4}\Delta_k + \Gamma_{n21}^{-1}\tilde{N}_{2,k}\dot{\hat{N}}_{2,k} \\ = {} & Z_{2,k}Z_{3,k} - \sum_{i=1}^{2}\eta_i Z_{i,k}^2 + \frac{2}{4}\Delta_k \\ & + \tilde{W}_{f2,k}^T \Gamma_{f21}^{-1}\left(\Gamma_{f21}\left(S_{f2}\left(\bar{x}_{2,k}, \hat{M}_{2,k}^T \Phi_2(t)\right) - \hat{S}'_{f2,k}\hat{M}_{2,k}^T \Phi_2(t)\right)Z_{2,k}\right. \\ & \left. + \dot{\hat{W}}_{f2,k}\right) - \tilde{W}_{g2,k}^T \Gamma_{g21}^{-1}\left(\Gamma_{g21}S_{g2}(\bar{x}_{2,k})Z_{2,k} - \dot{\hat{W}}_{g2,k}\right) \\ & + \tilde{M}_{2,k}^T \Gamma_{m21}^{-1}\left(\Gamma_{m21}\Phi_2(t)\hat{W}_{f2,k}^T \hat{S}'_{f2,k}Z_{2,k} + \dot{\hat{M}}_{2,k}\right) \\ & - \tilde{N}_{2,k}\Gamma_{n21}^{-1}\left(\Gamma_{n21}\frac{1}{\Delta_k}Z_{2,k}^2 - \dot{\hat{N}}_{2,k}\right).\end{aligned} \tag{41}$$

We choose

$$\begin{aligned}\dot{\hat{W}}_{f2,k} = {} & -\Gamma_{f21}\left(S_{f2}\left(\bar{x}_{2,k}, \hat{M}_{2,k}^T \Phi_2(t)\right) - \hat{S}'_{f2,k}\hat{M}_{2,k}^T \Phi_2(t)\right)Z_{2,k} \\ \dot{\hat{W}}_{g2,k} = {} & \Gamma_{g21}S_{g2}(\bar{x}_{2,k})Z_{2,k} \\ \dot{\hat{M}}_{2,k} = {} & -\Gamma_{m21}\Phi_2(t)\hat{W}_{f2,k}^T \hat{S}'_{f2,k}Z_{2,k} \\ \dot{\hat{N}}_{2,k} = {} & \Gamma_{n21}\frac{1}{\Delta_k}Z_{2,k}^2.\end{aligned} \tag{42}$$

Then, Equation 41 can be changed as

$$\dot{V}_{2,k} \le Z_{2,k}Z_{3,k} - \sum_{i=1}^{2}\eta_i Z_{i,k}^2 + \frac{2}{4}\Delta_k. \tag{43}$$

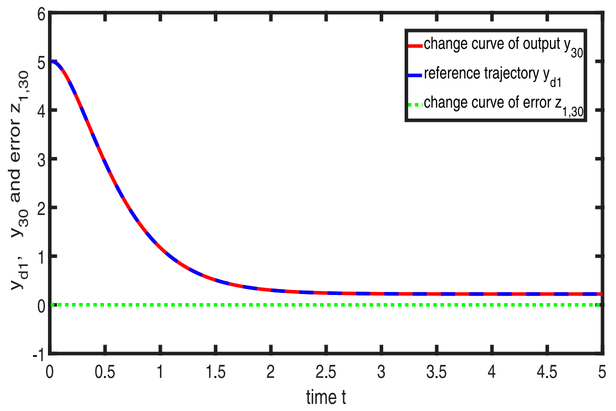


FIGURE 13
Variation in $y_{30}, y_{d1}, z_{1,30}$ over time during the 30th iteration.

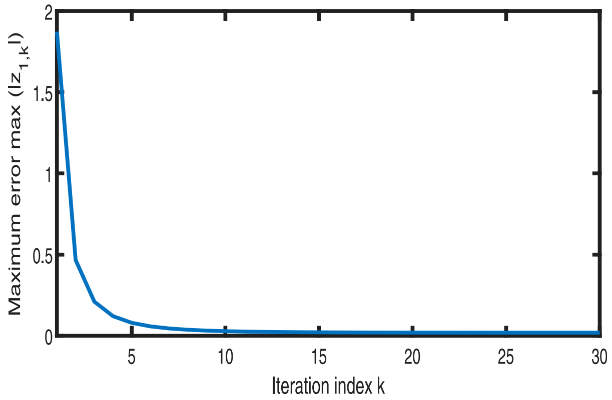


FIGURE 14
Variation in $max(|z_{1,k}|)$ according to the iteration index.

Step i: $(3 \le i \le n-1)$**.** Denote $N_i = \omega_{Mi}^2$, which will be defined later. Because there exist initial state errors and gradient explosion, the error functions $Z_{i,k}$ and $Z_{i+1,k}$ are defined as

$$\begin{aligned} Z_{i,k} &= z_{i\phi,k} - \zeta_{i,k} \\ z_{i\phi,k} &= z_{i,k} - \phi_{i,k}(t) sat\left(\frac{z_{i,k}}{\phi_{i,k}(t)}\right) \\ z_{i,k} &= x_{i,k} - \beta_{i-1,k} \\ \phi_{i,k}(t) &= \epsilon_{i,k} e^{-\eta_i t}, \end{aligned} \tag{44}$$

$$\begin{aligned} Z_{i+1,k} &= z_{i+1\phi,k} - \zeta_{i+1,k} \\ z_{i+1\phi,k} &= z_{i+1,k} - \phi_{i+1,k}(t) sat\left(\frac{z_{i+1,k}}{\phi_{i+1,k}(t)}\right) \\ z_{i+1,k} &= x_{i+1,k} - \beta_{i,k} \\ \phi_{i+1,k}(t) &= \epsilon_{i+1,k} e^{-\eta_{i+1} t}. \end{aligned} \tag{45}$$

Therefore, $\dot{Z}_{i,k}$ can be deduced as follows:

$$\begin{aligned} \dot{Z}_{i,k} &= \dot{z}_{i,k} - sgn\left(z_{i\phi,k}(t)\right)\dot{\phi}_{i,k} - \dot{\zeta}_{i,k} \\ &= z_{i+1,k} + \beta_{i,k} + f_i\left(\bar{x}_{i,k}, \theta_i(t)\right) + g_i\left(\bar{x}_{i,k}\right) \\ &\quad -\dot{\beta}_{i-1,k} - sgn\left(z_{i\phi,k}\right)\dot{\phi}_{i,k} - \dot{\zeta}_{i,k}. \end{aligned} \tag{46}$$

Let the error compensation mechanism be defined as

$$\dot{\zeta}_{i,k} = \beta_{i,k} + \zeta_{i+1,k} - \eta_i \zeta_{i,k} - \zeta_{i-1,k} - \alpha_{i,k}. \tag{47}$$

Using Equation 47, we can find the time derivative of the error function as

$$\begin{aligned} \dot{Z}_{i,k} = \; & z_{i+1,k} - \zeta_{i+1,k} + \eta_{i,k}\zeta_{i,k} + \zeta_{i-1,k} + \alpha_{i,k} - \dot{\beta}_{i-1,k} \\ & + f_i\left(\bar{x}_{i,k}, \theta_i(t)\right) + g_i\left(\bar{x}_{i,k}\right) - sgn\left(z_{i\phi,k}\right)\dot{\phi}_{i,k}. \end{aligned} \tag{48}$$

The uncertain time-varying, nonlinear functions $f_i(\bar{x}_{i,k}, \theta_i(t))$ and $G_i(\bar{x}_{i,k})$ are approximated by FSE-RBFNN and RBFNN, respectively, and reconstruction errors δ_{fi} and δ_{gi} are as given follows:

$$\begin{aligned} f_i\left(\bar{x}_{i,k}, \theta_i(t)\right) &= W_{fi}^T S_{fi}\left(\bar{x}_{i,k}, M_i^T \Phi_i(t)\right) + \delta_{fi} \\ G_i\left(\bar{x}_{i,k}\right) &= W_{gi}^T S_{gi}\left(\bar{x}_{i,k}\right) + \delta_{gi}, \end{aligned} \tag{49}$$

where δ_{fi} and δ_{gi} are the approximation errors and W_{fi} and W_{gi} are ideal weight vectors.

Define $\Delta_k = \frac{a}{k^l}$, where a is any arbitrary number with $a > 0$; meanwhile, $l \ge 2$. Let the virtual control be defined as

$$\begin{aligned} \alpha_{i,k} = \; & -\hat{W}_{fi,k}^T S_{fi}\left(\bar{x}_{i,k}, \hat{M}_{i,k}^T \Phi_i(t)\right) - \hat{W}_{gi,k}^T S_{gi}\left(\bar{x}_{i,k}\right) \\ & -\hat{N}_{i,k}\frac{1}{\Delta_k} Z_{i,k} + \dot{\beta}_{i-1,k} - \eta_i z_{i,k} - z_{i-1\phi,k}. \end{aligned} \tag{50}$$

By substituting Equations 49, 50 into Equation 48, we obtain

$$\begin{aligned} \dot{Z}_{i,k} = \; & -z_{i-1\phi,k} + \zeta_{i-1,k} + z_{i+1,k} - \phi_{i+1,k}(t) sat\left(\frac{z_{i+1,k}}{\phi_{i+1,k}(t)}\right) - \zeta_{i+1,k} \\ & + W_{fi}^T S_{fi}\left(\bar{x}_{i,k}, M_i^T \Phi_i(t)\right) + \delta_{fi} - \hat{W}_{fi,k}^T S_{fi}\left(\bar{x}_{i,k}, \hat{M}_{i,k}^T \Phi_i(t)\right) \\ & + W_{gi}^T S_{gi}\left(\bar{x}_{i,k}\right) + \delta_{gi} - \hat{W}_{gi,k}^T S_{gi}\left(\bar{x}_{i,k}\right) \\ & -\hat{N}_{i,k}\frac{1}{\Delta_k} z_{i\phi,k} + \phi_{i+1,k}(t) sat\left(\frac{z_{i+1,k}}{\phi_{i+1,k}(t)}\right) \\ & -\eta_i z_{i,k} + \eta_i \zeta_{i,k} - sgn\left(z_{i\phi,k}\right)\dot{\phi}_{i,k} \\ = \; & -Z_{i-1,k} + Z_{i+1,k} - \hat{N}_{i,k}\frac{1}{\Delta_k} z_{i\phi,k} + \phi_{i+1,k}(t) sat\left(\frac{z_{i+1,k}}{\phi_{i+1,k}(t)}\right) \\ & + W_{fi}^T S_{fi}\left(\bar{x}_{i,k}, M_i^T \Phi_i(t)\right) + \delta_{fi} - \hat{W}_{fi,k}^T S_{fi}\left(\bar{x}_{i,k}, \hat{M}_{i,k}^T \Phi_i(t)\right) \\ & + W_{gi}^T S_{gi}\left(\bar{x}_{i,k}\right) + \delta_{gi} - \hat{W}_{gi,k}^T S_{gi}\left(\bar{x}_{i,k}\right) \\ & -\eta_i z_{i,k} + \eta_i \zeta_{i,k} - sgn\left(z_{i\phi,k}\right)\dot{\phi}_{i,k}, \end{aligned} \tag{51}$$

where $\hat{W}_{fi,k}$, $\hat{W}_{gi,k}$, $\hat{M}_{i,k}$, and $\hat{N}_{i,k}$ are the estimations of W_{fi}, W_{gi}, M_i, and N_i, respectively. $\tilde{W}_{fi,k} = \hat{W}_{fi,k} - W_{fi}$, $\tilde{W}_{gi,k} = \hat{W}_{gi,k} - W_{gi}$, $\tilde{M}_{i,k} = \hat{M}_{i,k} - M_i$, and $\tilde{N}_{i,k} = \hat{N}_{i,k} - N_i$ are estimation errors. We can rephrase the final three components on the right side of Equation 51 as

$$\begin{aligned} -\eta_i z_{i,k} - sgn\left(z_{i\phi,k}\right)\dot{\phi}_{i,k}(t) + \eta_i \zeta_{i,k} &= -\eta_i z_{i\phi,k} - \eta_i \phi_{i,k}(t) sat\left(\frac{z_{i,k}}{\phi_{i,k}(t)}\right) \\ &\quad - sgn\left(z_{i\phi,k}\right)\dot{\phi}_{i,k}(t) + \eta_i \zeta_{i,k} \\ &= -\eta_i z_{i\phi,k} + \eta_i \zeta_{i,k} \\ &\quad - sgn\left(z_{i\phi,k}\right)\left(\dot{\phi}_{i,k}(t) + \eta_i \phi_{i,k}(t)\right) \\ &= -\eta_i z_{i\phi,k} + \eta_i \zeta_{i,k} \\ &= -\eta_i\left(z_{i\phi,k} - \zeta_{i,k}\right) \\ &= -\eta_i Z_{i,k}, \end{aligned} \tag{52}$$

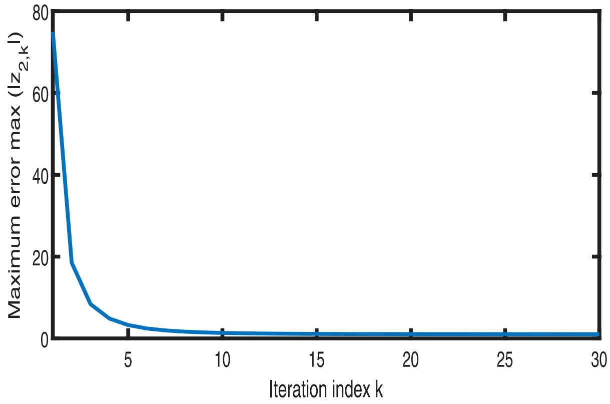


FIGURE 15
Variation in $max(|z_{2,k}|)$ according to the iteration index.

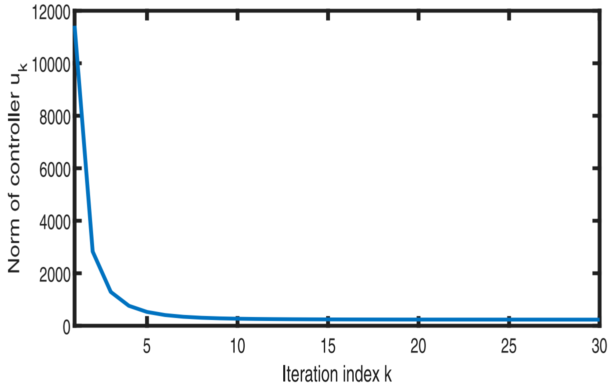


FIGURE 16
Variation in $\|u_k\|$ according to the iteration index.

Using Equations 7, 52, Equation 51 can be reformulated as

$$\begin{aligned}\dot{Z}_{i,k} &= -Z_{i-1,k} + Z_{i+1,k} - \hat{N}_{i,k}\frac{1}{\Delta_k}z_{i\phi,k} - \eta_i Z_{i,k}\\ &+\tilde{W}^T_{fi}\left(S_{fi}\left(\bar{x}_{i,k}, \hat{M}^T_{i,k}\Phi_i(t)\right) - \hat{S}'_{fi,k}\hat{M}^T_{i,k}\Phi_i(t)\right)\\ &+\hat{W}^T_{fi,k}\hat{S}'_{fi,k}\tilde{M}^T_{i,k}\Phi_i(t) - \tilde{W}^T_{gi}S_{gi}(\bar{x}_{i,k})\\ &+d_i + \delta_{fi} + \delta_{gi} + \phi_{i+1,k}(t)sat\left(\frac{z_{i+1,k}}{\phi_{i+1,k}(t)}\right).\end{aligned} \tag{53}$$

Let $\omega_i = d_i + \delta_{fi} + \delta_{gi} + \phi_{i+1,k}(t)sat\,(\frac{z_{i+1,k}(t)}{\phi_{i+1,k}(t)})$, then Equation 53 becomes

$$\begin{aligned}\dot{Z}_{i,k} &= -Z_{i-1,k} + Z_{i+1,k} - \hat{N}_{i,k}\frac{1}{\Delta_k}Z_{i,k} - \eta_i Z_{i,k} + \omega_i\\ &+\tilde{W}^T_{fi,k}\left(S_{fi}\left(\bar{x}_{i,k}, \hat{M}^T_{i,k}\Phi_i(t)\right) - \hat{S}'_{fi,k}\hat{M}^T_{i,k}\Phi_i(t)\right)\\ &+\hat{W}^T_{fi,k}\hat{S}'_{fi,k}\tilde{M}^T_{i,k}\Phi_i(t) - \tilde{W}^T_{gi}S_{gi}(\bar{x}_{i,k}).\end{aligned} \tag{54}$$

Consider the following nonnegative function:

$$\begin{aligned}V_{i,k} &= V_{i-1,k} + \frac{1}{2}Z^2_{i,k} + \frac{1}{2}\tilde{W}^T_{fi,k}\Gamma^{-1}_{fi1}\tilde{W}_{fi,k} + \frac{1}{2}\tilde{W}^T_{gi,k}\Gamma^{-1}_{gi1}\tilde{W}_{gi,k}\\ &+\frac{1}{2}\tilde{M}^T_{i,k}\Gamma^{-1}_{mi1}\tilde{M}_{i,k} + \frac{1}{2}\Gamma^{-1}_{ni1}\tilde{N}^2_{i,k},\end{aligned} \tag{55}$$

where Γ_{fi1}, Γ_{gi1}, Γ_{mi1}, and Γ_{ni1} are adjustable, positive, definite, and symmetric matrices. According to Equation 54, Assumption 3, and Remark 1, $V_{i,k}$ can be expressed as

$$\begin{aligned}\dot{V}_{i,k} = \quad &\dot{V}_{i-1,k} + Z_{i,k}\dot{Z}_{i,k} + \tilde{W}^T_{fi,k}\Gamma^{-1}_{fi1}\dot{\hat{W}}_{fi,k} \qquad + \tilde{W}^T_{gi,k}\Gamma^{-1}_{gi1}\dot{\hat{W}}_{gi,k} + \tilde{M}^T_{i,k}\Gamma^{-1}_{mi1}\dot{\hat{M}}_{i,k}\\ &+\Gamma^{-1}_{ni1}\tilde{N}_{i,k}\dot{\hat{N}}_{i,k} \le Z_{i-1,k}Z_{i,k} - \sum_{j=1}^{i-1}\eta_j Z^2_{j,k} + \frac{i-1}{4}\Delta_k - Z_{i-1,k}Z_{i,k} + Z_{i,k}Z_{i+1,k} - \eta_i Z^2_{i,k}\\ &+\tilde{W}^T_{fi,k}\Gamma^{-1}_{fi1}\left(\Gamma_{fi1}\left(S_{fi}\left(\bar{x}_{i,k}, \hat{M}^T_{i,k}\Phi_i(t)\right) - \hat{S}'_{fi,k}\hat{M}^T_{i,k}\Phi_i(t)\right)Z_{i,k} + \dot{\hat{W}}_{fi,k}\right)\\ &-\tilde{W}^T_{gi,k}\Gamma^{-1}_{gi1}\left(\Gamma_{gi1}S_{gi}(\bar{x}_{i,k})Z_{i,k} - \dot{\hat{W}}_{gi,k}\right)\\ &+\tilde{M}^T_{i,k}\Gamma^{-1}_{mi1}\left(\Gamma_{mi1}\Phi_i(t)\hat{W}^T_{fi,k}\hat{S}'_{fi,k}Z_{i,k} + \dot{\hat{M}}_{i,k}\right) - \hat{N}_{i,k}\frac{1}{\Delta_k}Z^2_{i,k} + \omega_i Z_{i,k}\\ &+\Gamma^{-1}_{ni1}\tilde{N}_{i,k}\dot{\hat{N}}_{i,k} \le -\sum_{j=1}^{i-1}\eta_j Z^2_{j,k} + \frac{i-1}{4}\Delta_k + Z_{i,k}Z_{i+1,k} - \eta_i Z^2_{i,k}\\ &+\tilde{W}^T_{fi,k}\Gamma^{-1}_{fi1}\left(\Gamma_{fi1}\left(S_{fi}\left(\bar{x}_{i,k}, \hat{M}^T_{i,k}\Phi_i(t)\right) - \hat{S}'_{fi,k}\hat{M}^T_{i,k}\Phi_i(t)\right)Z_{i,k} + \dot{\hat{W}}_{fi,k}\right)\\ &-\tilde{W}^T_{gi,k}\Gamma^{-1}_{gi1}\left(\Gamma_{gi1}S_{gi}(\bar{x}_{i,k})Z_{i,k} - \dot{\hat{W}}_{gi,k}\right)\\ &+\tilde{M}^T_{i,k}\Gamma^{-1}_{mi1}\left(\Gamma_{mi1}\Phi_i(t)\hat{W}^T_{fi,k}\hat{S}'_{fi,k}Z_{i,k} + \dot{\hat{M}}_{i,k}\right) - \hat{N}_{i,k}\frac{1}{\Delta_k}Z^2_{i,k} + \frac{1}{\Delta_k}\omega^2_{Mi}Z^2_{i,k}\\ &+\frac{1}{4}\Delta_k + \Gamma^{-1}_{ni1}\tilde{N}_{i,k}\dot{\hat{N}}_{i,k}\\ &= -\sum_{j=1}^{i}\eta_j Z^2_{j,k} + \frac{i-1}{4}\Delta_k + Z_{i,k}Z_{i+1,k} + \tilde{W}^T_{fi,k}\Gamma^{-1}_{fi1}\left(\Gamma_{fi1}\left(S_{fi}\left(\bar{x}_{i,k}, \hat{M}^T_{i,k}\Phi_i(t)\right)\right.\right.\\ &\qquad \left.\left. - \hat{S}'_{fi,k}\hat{M}^T_{i,k}\Phi_i(t))Z_{i,k} + \dot{\hat{W}}_{fi,k}\right) - \tilde{W}^T_{gi,k}\Gamma^{-1}_{gi1}\left(\Gamma_{gi1}S_{gi}(\bar{x}_{i,k})Z_{i,k} - \dot{\hat{W}}_{gi,k}\right)\\ &\qquad + \tilde{M}^T_{i,k}\Gamma^{-1}_{mi1}\left(\Gamma_{mi1}\Phi_i(t)\hat{W}^T_{fi,k}\hat{S}'_{fi,k}Z_{i,k} + \dot{\hat{M}}_{i,k}\right) - \hat{N}_{i,k}\frac{1}{\Delta_k}Z^2_{i,k}\\ &\qquad + \frac{1}{\Delta_k}N_{i,k}Z^2_{i,k} + \frac{1}{4}\Delta_k + \Gamma^{-1}_{ni1}\tilde{N}_{i,k}\dot{\hat{N}}_{i,k} \;=\; -\sum_{j=1}^{i}\eta_j Z^2_{j,k} + \frac{i}{4}\Delta_k + Z_{i,k}Z_{i+1,k}\\ &\qquad +\tilde{W}^T_{fi,k}\Gamma^{-1}_{fi1}\left(\Gamma_{fi1}\left(S_{fi}\left(\bar{x}_{i,k}, \hat{M}^T_{i,k}\Phi_i(t)\right) - \hat{S}'_{fi,k}\hat{M}^T_{i,k}\Phi_i(t)\right)Z_{i,k} + \dot{\hat{W}}_{fi,k}\right)\\ &\qquad -\tilde{W}^T_{gi,k}\Gamma^{-1}_{gi1}\left(\Gamma_{gi1}S_{gi}(\bar{x}_{i,k})Z_{i,k} - \dot{\hat{W}}_{gi,k}\right)\\ &\qquad +\tilde{M}^T_{i,k}\Gamma^{-1}_{mi1}\left(\Gamma_{mi1}\Phi_i(t)\hat{W}^T_{fi,k}\hat{S}'_{fi,k}Z_{i,k} + \dot{\hat{M}}_{i,k}\right)\\ &\qquad -\tilde{N}_{i,k}\Gamma^{-1}_{ni1}\left(\Gamma_{ni1}\frac{1}{\Delta_k}Z^2_{i,k} - \dot{\hat{N}}_{i,k}\right).\end{aligned} \tag{56}$$

We choose

$$\begin{aligned}\dot{\hat{W}}_{fi,k} &= -\Gamma_{fi1}\left(S_{fi}\left(\bar{x}_{i,k}, \hat{M}^T_{i,k}\Phi_i(t)\right) - \hat{S}'_{fi,k}\hat{M}^T_{i,k}\Phi_i(t)\right)Z_{i,k}\\ \dot{\hat{W}}_{gi,k} &= \Gamma_{gi1}S_{gi}(\bar{x}_{i,k})Z_{i,k}\\ \dot{\hat{M}}_{i,k} &= -\Gamma_{mi1}\Phi_i(t)\hat{W}^T_{fi,k}\hat{S}'_{fi,k}Z_{i,k}\\ \dot{\hat{N}}_{i,k} &= \Gamma_{ni1}\frac{1}{\Delta_k}Z^2_{i,k}.\end{aligned} \tag{57}$$

Then, Equation 56 can be written as

$$\dot{V}_{2,k} \le -\sum_{j=1}^{i}\eta_j Z^2_{j,k} + \frac{i}{4}\Delta_k + Z_{i,k}Z_{i+1,k}, \tag{58}$$

Step n: Denote $N_n = \omega^2_{Mn}$, which will be defined later. Because there exist initial state errors and gradient explosion, the function $Z_{n,k}$, denoting the error, is defined as

$$\begin{aligned}Z_{n,k} &= z_{n\phi,k} - \zeta_{n,k}\\ z_{n\phi,k} &= z_{n,k} - \phi_{n,k}(t)sat\left(\frac{z_{n,k}}{\phi_{n,k}(t)}\right)\\ z_{n,k} &= x_{n,k} - \beta_{n-1,k}\\ \phi_{n,k}(t) &= \epsilon_{n,k}e^{-\eta_n t}.\end{aligned} \tag{59}$$

The derivative of $Z_{i,k}$ with respect to time is expressed as

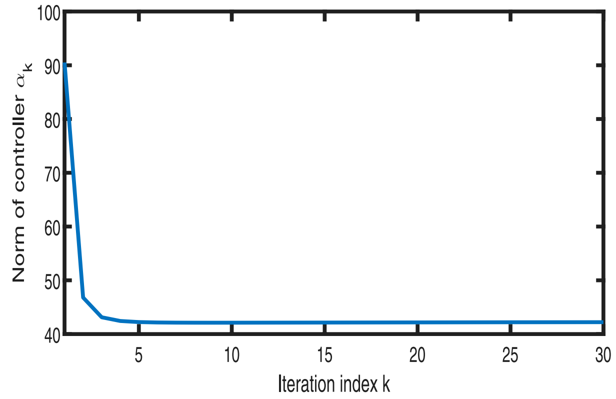


FIGURE 17
Variation in $\|\alpha_k\|$ according to the iteration index.

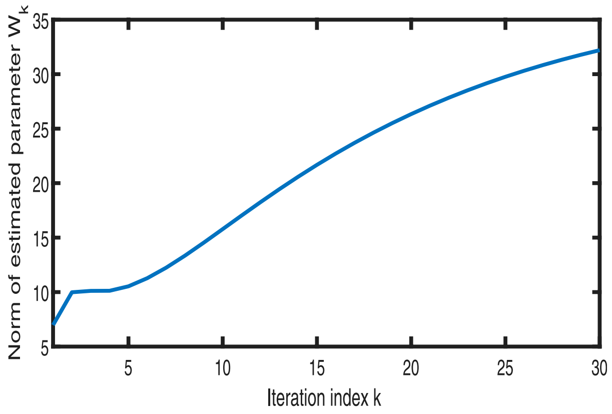


FIGURE 18
Variation of $\|\hat{W}_k\|$ according to the iteration index.

$$\begin{aligned}\dot{Z}_{n,k} &= \dot{z}_{n,k} - sgn\left(z_{n\phi,k}(t)\right)\dot{\phi}_{n,k} - \dot{\zeta}_{n,k}\\ &= u_k + f_n\left(\bar{x}_{n,k}, \theta_n(t)\right) + g_n\left(\bar{x}_{n,k}\right)\\ &\quad -\dot{\beta}_{n-1,k} - sgn\left(z_{n\phi,k}\right)\dot{\phi}_{n,k} - \dot{\zeta}_{n,k}.\end{aligned} \tag{60}$$

Let the error compensation mechanism be defined as

$$\dot{\zeta}_{n,k} = -\eta_n\zeta_{n,k} - \zeta_{n-1,k}. \tag{61}$$

Using Equation 61, we can obtain the time derivative of the error function as

$$\begin{aligned}\dot{Z}_{n,k} &= u_k + \eta_{n,k}\zeta_{n,k} + \zeta_{n-1,k} - \dot{\beta}_{n-1,k}\\ &\quad + f_n\left(\bar{x}_{n,k}, \theta_n(t)\right) + g_n\left(\bar{x}_{n,k}\right) - sgn\left(z_{n\phi,k}\right)\dot{\phi}_{n,k}.\end{aligned} \tag{62}$$

The overall approximation capability of the RBFNN asserts that the unknown nonlinear functions $f_n(\bar{x}_{n,k}, \theta_n(t))$ and $G_n(\bar{x}_{n,k})$ are capable of approximation within a defined scope by FSE-RBFNN and RBFNN, respectively, and reconstruction errors δ_{fn} and δ_{gn} are as follows:

$$\begin{aligned}f_n\left(\bar{x}_{n,k}, \theta_n(t)\right) &= W_{fn}^T S_{fn}\left(\bar{x}_{n,k}, M_n^T\Phi_n(t)\right) + \delta_{fn}\\ G_n\left(\bar{x}_{n,k}\right) &= W_{gn}^T S_{gn}\left(\bar{x}_{n,k}\right) + \delta_{gn},\end{aligned} \tag{63}$$

where δ_{fn} and δ_{gn} are the approximation errors and W_{fn} and W_{gn} are ideal weight vectors.

Define $\Delta_k = \frac{a}{k^l}$,where a is any arbitrary number such that $a > 0$; meanwhile, $l \geq 2$. Let the virtual control be defined as

$$\begin{aligned}u_k &= -\hat{W}_{fn,k}^T S_{fn}\left(\bar{x}_{n,k}, \hat{M}_{n,k}^T\Phi_n(t)\right) - \hat{W}_{gn,k}^T S_{gn}\left(\bar{x}_{n,k}\right)\\ &\quad -\hat{N}_{n,k}\frac{1}{\Delta_k}Z_{n,k} + \dot{\beta}_{n-1,k} - \eta_n z_{n,k} - z_{n-1\phi,k}.\end{aligned} \tag{64}$$

By substituting Equations 63, 64 into Equation 62, we can conclude that

$$\begin{aligned}\dot{Z}_{n,k} &= -z_{n-1\phi,k} + \zeta_{n-1,k} - \eta_n z_{n,k} + \eta_n\zeta_{n,k} - sgn\left(z_{n\phi,k}\right)\dot{\phi}_{n,k}\\ &\quad + W_{fn}^T S_{fn}\left(\bar{x}_{n,k}, M_n^T\Phi_n(t)\right) + \delta_{fn} - \hat{W}_{fn,k}^T S_{fn}\left(\bar{x}_{n,k}, \hat{M}_{n,k}^T\Phi_n(t)\right)\\ &\quad + W_{gn}^T S_{gn}\left(\bar{x}_{n,k}\right) + \delta_{gn} - \hat{W}_{gn,k}^T S_{gn}\left(\bar{x}_{n,k}\right) - \hat{N}_{n,k}\frac{1}{\Delta_k}z_{n\phi,k}\\ &= W_{fn}^T S_{fn}\left(\bar{x}_{n,k}, M_n^T\Phi_n(t)\right) + \delta_{fn} - \hat{W}_{fn,k}^T S_{fn}\left(\bar{x}_{n,k}, \hat{M}_{n,k}^T\Phi_n(t)\right)\\ &\quad + W_{gn}^T S_{gn}\left(\bar{x}_{n,k}\right) + \delta_{gn} - \hat{W}_{gn,k}^T S_{gn}\left(\bar{x}_{n,k}\right) - \hat{N}_{n,k}\frac{1}{\Delta_k}z_{n\phi,k}\\ &\quad - Z_{n-1,k} - \eta_n z_{n,k} + \eta_n\zeta_{n,k} - sgn\left(z_{n\phi,k}\right)\dot{\phi}_{n,k},\end{aligned} \tag{65}$$

where $\hat{W}_{fn,k}$, $\hat{W}_{gn,k}$, $\hat{M}_{n,k}$, and $\hat{N}_{n,k}$ are the estimations of W_{fn}, W_{gn}, M_n, and N_n, respectively. $\tilde{W}_{fn,k} = \hat{W}_{fn,k} - W_{fn}$, $\tilde{W}_{gn,k} = \hat{W}_{gn,k} - W_{gn}$, $\tilde{M}_{n,k} = \hat{M}_{n,k} - M_n$, and $\tilde{N}_{n,k} = \hat{N}_{n,k} - N_n$ are estimation errors. We can rephrase the final three components on the right side of Equation 65 as

$$\begin{aligned}&-\eta_n z_{n,k} - sgn\left(z_{n\phi,k}\right)\dot{\phi}_{n,k}(t) + \eta_n\zeta_{n,k}\\ &\quad = -\eta_n z_{n\phi,k} - \eta_n\phi_{n,k}(t)sat\left(\frac{z_{n,k}}{\phi_{n,k}(t)}\right) - sgn\left(z_{n\phi,k}\right)\dot{\phi}_{n,k}(t) + \eta_n\zeta_{n,k}\\ &\quad = -\eta_n z_{n\phi,k} + \eta_n\zeta_{n,k} - sgn\left(z_{n\phi,k}\right)\left(\dot{\phi}_{n,k}(t) + \eta_n\phi_{n,k}(t)\right)\\ &\quad = -\eta_n z_{n\phi,k} + \eta_n\zeta_{n,k} = -\eta_n\left(z_{n\phi,k} - \zeta_{n,k}\right) = -\eta_n Z_{n,k}.\end{aligned} \tag{66}$$

Using Equations 7, 66, Equation 65 can be reformulated as

$$\begin{aligned}\dot{Z}_{n,k} &= -Z_{n-1,k} - \hat{N}_{n,k}\frac{1}{\Delta_k}z_{n\phi,k} - \eta_n Z_{n,k}\\ &\quad + \tilde{W}_{fn}^T\left(S_{fn}\left(\bar{x}_{n,k}, \hat{M}_{n,k}^T\Phi_n(t)\right) - \hat{S}'_{fn,k}\hat{M}_{n,k}^T\Phi_n(t)\right)\\ &\quad + \hat{W}_{fn,k}^T\hat{S}'_{fn,k}\tilde{M}_{n,k}^T\Phi_n(t) - \tilde{W}_{gn}^T S_{gn}\left(\bar{x}_{n,k}\right)\\ &\quad + d_n + \delta_{fn} + \delta_{gn}.\end{aligned} \tag{67}$$

Let $\omega_n = d_n + \delta_{fn} + \delta_{gn}$, then Equation 67 becomes

$$\begin{aligned}\dot{Z}_{n,k} &= -Z_{n-1,k} - \hat{N}_{n,k}\frac{1}{\Delta_k}Z_{n,k} - \eta_n Z_{n,k} + \omega_n\\ &\quad + \tilde{W}_{fn,k}^T\left(S_{fn}\left(\bar{x}_{n,k}, \hat{M}_{n,k}^T\Phi_n(t)\right) - \hat{S}'_{fn,k}\hat{M}_{n,k}^T\Phi_n(t)\right)\\ &\quad + \hat{W}_{fn,k}^T\hat{S}'_{fn,k}\tilde{M}_{n,k}^T\Phi_n(t) - \tilde{W}_{gn}^T S_{gn}\left(\bar{x}_{n,k}\right).\end{aligned} \tag{68}$$

Assumption 4: The remainder ω_n is bounded with $|\omega_n| \leq \omega_{Mn}$ and $\omega_{Mn} > 0$.

Remark 2: This assumption is reasonable because 1) d_n, δ_{fn}, and δ_{gn} are constrained within the specified area by Equations 6, 8.

Let the following non-negative function be defined as

$$\begin{aligned}V_{n,k} &= V_{n-1,k} + \frac{1}{2}Z_{n,k}^2 + \frac{1}{2}\tilde{W}_{fn,k}^T\Gamma_{fn1}^{-1}\tilde{W}_{fn,k} + \frac{1}{2}\tilde{W}_{gn,k}^T\Gamma_{gn1}^{-1}\tilde{W}_{gn,k}\\ &\quad + \frac{1}{2}\tilde{M}_{n,k}^T\Gamma_{mn1}^{-1}\tilde{M}_{n,k} + \frac{1}{2}\Gamma_{Nn1}^{-1}\tilde{N}_{n,k}^2,\end{aligned} \tag{69}$$

where Γ_{fn1}, Γ_{gn1}, Γ_{mn1}, and Γ_{Nn1} are adjustable, positive, definite, and symmetric matrices. The derivative of $V_{n,k}$ is considered as follows (Equation 68):

$$\begin{aligned}
\dot{V}_{n,k} = & \ \dot{V}_{n-1,k} + Z_{n,k}\dot{Z}_{n,k} + \tilde{W}^T_{fn,k}\Gamma^{-1}_{fn1}\dot{\hat{W}}_{fn,k} \\
& +\tilde{W}^T_{gn,k}\Gamma^{-1}_{gn1}\dot{\hat{W}}_{gn,k} + \tilde{M}^T_{n,k}\Gamma^{-1}_{mn1}\dot{\hat{M}}_{n,k} + \Gamma^{-1}_{Nn1}\tilde{N}_{n,k}\dot{\hat{N}}_{n,k} \\
\leq & \ Z_{n-1,k}Z_{n,k} - \sum_{j=1}^{n-1}\eta_j Z^2_{j,k} + \frac{n-1}{4}\Delta_k - Z_{n-1,k}Z_{n,k} - \eta_i Z^2_{i,k} \\
& +\tilde{W}^T_{fn,k}\Gamma^{-1}_{fn1}\Big(\Gamma_{fn1}\Big(S_{fn}\Big(\bar{x}_{n,k}, \hat{M}^T_{n,k}\Phi_n(t)\Big) \\
& -\hat{S}'_{fn,k}\hat{M}^T_{n,k}\Phi_n(t)\Big)Z_{n,k} + \dot{\hat{W}}_{fn,k}\Big) \\
& -\tilde{W}^T_{gn,k}\Gamma^{-1}_{gn1}\Big(\Gamma_{gn1}S_{gn}(\bar{x}_{n,k})Z_{n,k} - \dot{\hat{W}}_{gn,k}\Big) \\
& +\tilde{M}^T_{n,k}\Gamma^{-1}_{mn1}\Big(\Gamma_{mn1}\Phi_n(t)\hat{W}^T_{fn,k}\hat{S}'_{fn,k}Z_{n,k} + \dot{\hat{M}}_{n,k}\Big) \\
& -\hat{N}_{n,k}\frac{1}{\Delta_k}Z^2_{n,k} + \omega_n Z_{n,k} + \Gamma^{-1}_{Nn1}\tilde{N}_{n,k}\dot{\hat{N}}_{n,k} \\
\leq & -\sum_{j=1}^{n-1}\eta_j Z^2_{j,k} + \frac{n-1}{4}\Delta_k - \eta_n Z^2_{n,k} \\
& +\tilde{W}^T_{fn,k}\Gamma^{-1}_{fn1}\Big(\Gamma_{fn1}\Big(S_{fn}\Big(\bar{x}_{n,k}, \hat{M}^T_{n,k}\Phi_n(t)\Big) \\
& -\hat{S}'_{fn,k}\hat{M}^T_{n,k}\Phi_n(t)\Big)Z_{n,k} + \dot{\hat{W}}_{fn,k}\Big) \\
& -\tilde{W}^T_{gn,k}\Gamma^{-1}_{gn1}\Big(\Gamma_{gn1}S_{gn}(\bar{x}_{n,k})Z_{n,k} - \dot{\hat{W}}_{gn,k}\Big) \\
& +\tilde{M}^T_{n,k}\Gamma^{-1}_{mn1}\Big(\Gamma_{mn1}\Phi_n(t)\hat{W}^T_{fn,k}\hat{S}'_{fn,k}Z_{n,k} + \dot{\hat{M}}_{n,k}\Big) \\
& -\hat{N}_{n,k}\frac{1}{\Delta_k}Z^2_{n,k} + \frac{1}{\Delta_k}\omega^2_{Mn}Z^2_{n,k} + \frac{1}{4}\Delta_k + \Gamma^{-1}_{Nn1}\tilde{N}_{n,k}\dot{\hat{N}}_{n,k} \\
= & -\sum_{j=1}^{n}\eta_j Z^2_{j,k} + \frac{n-1}{4}\Delta_k \\
& +\tilde{W}^T_{fn,k}\Gamma^{-1}_{fn1}\Big(\Gamma_{fn1}\Big(S_{fn}\Big(\bar{x}_{n,k}, \hat{M}^T_{n,k}\Phi_n(t)\Big) \\
& -\hat{S}'_{fn,k}\hat{M}^T_{n,k}\Phi_n(t)\Big)Z_{n,k} + \dot{\hat{W}}_{fn,k}\Big) \\
& -\tilde{W}^T_{gn,k}\Gamma^{-1}_{gn1}\Big(\Gamma_{gn1}S_{gn}(\bar{x}_{n,k})Z_{n,k} - \dot{\hat{W}}_{gn,k}\Big) \\
& +\tilde{M}^T_{n,k}\Gamma^{-1}_{mn1}\Big(\Gamma_{mn1}\Phi_n(t)\hat{W}^T_{fn,k}\hat{S}'_{fn,k}Z_{n,k} + \dot{\hat{M}}_{n,k}\Big) \\
& -\hat{N}_{n,k}\frac{1}{\Delta_k}Z^2_{n,k} + \frac{1}{\Delta_k}N_{n,k}Z^2_{n,k} + \frac{1}{4}\Delta_k + \Gamma^{-1}_{Nn1}\tilde{N}_{n,k}\dot{\hat{N}}_{n,k} \\
= & -\sum_{j=1}^{n}\eta_j Z^2_{j,k} + \frac{n}{4}\Delta_k \\
& +\tilde{W}^T_{fn,k}\Gamma^{-1}_{fn1}\Big(\Gamma_{fn1}\Big(S_{fn}\Big(\bar{x}_{n,k}, \hat{M}^T_{n,k}\Phi_n(t)\Big) \\
& -\hat{S}'_{fn,k}\hat{M}^T_{n,k}\Phi_n(t)\Big)Z_{n,k} + \dot{\hat{W}}_{fn,k}\Big) \\
& -\tilde{W}^T_{gn,k}\Gamma^{-1}_{gn1}\Big(\Gamma_{gn1}S_{gn}(\bar{x}_{n,k})Z_{n,k} - \dot{\hat{W}}_{gn,k}\Big) \\
& +\tilde{M}^T_{n,k}\Gamma^{-1}_{mn1}\Big(\Gamma_{mn1}\Phi_n(t)\hat{W}^T_{fn,k}\hat{S}'_{fn,k}Z_{n,k} + \dot{\hat{M}}_{n,k}\Big) \\
& -\tilde{N}_{n,k}\Gamma^{-1}_{Nn1}\Big(\Gamma_{Nn1}\frac{1}{\Delta_k}Z^2_{n,k} - \dot{\hat{N}}_{n,k}\Big).
\end{aligned} \tag{70}$$

We choose

$$\begin{aligned}
\dot{\hat{W}}_{fn,k} &= -\Gamma_{fn1}\Big(S_{fn}\Big(\bar{x}_{n,k}, \hat{M}^T_{n,k}\Phi_n(t)\Big) - \hat{S}'_{fn,k}\hat{M}^T_{n,k}\Phi_n(t)\Big)Z_{n,k} \\
\dot{\hat{W}}_{gn,k} &= \Gamma_{gn1}S_{gn}(\bar{x}_{n,k})Z_{n,k} \\
\dot{\hat{M}}_{n,k} &= -\Gamma_{mn1}\Phi_n(t)\hat{W}^T_{fn,k}\hat{S}'_{fn,k}Z_{n,k} \\
\dot{\hat{N}}_{n,k} &= \Gamma_{Nn1}\frac{1}{\Delta_k}Z^2_{n,k}.
\end{aligned} \tag{71}$$

Then, Equation 70 can be written as

$$\dot{V}_{n,k} \leq -\sum_{j=1}^{n}\eta_j Z^2_{j,k} + \frac{n}{4}\Delta_k. \tag{72}$$

For the initial state, we rely on the following set of assumed conditions:

Assumption 2: When $t = 0$, $\hat{W}_{fi,k}(0) = \hat{W}_{fi,k-1}(T)$, $\hat{W}_{gi,k}(0) = \hat{W}_{gi,k-1}(T)$, $\hat{N}_{i,k}(0) = \hat{N}_{i,k-1}(T)$, and $\hat{M}_{i,k}(0) = \hat{M}_{i,k-1}(T)$ $(i = 1,\ldots,n)$ holds true for all values of k.

3.2 Stability and convergence analysis

Theorem 1: For nonlinear system (1) with assumptions 2, 3, and 4, if we design virtual controllers (21), (35), (50), controller (64), and parameter updating laws (28), (42), (57), (71),then all signals in the closed-loop system are bounded within the interval [0, T]. We obtain

$$lim_{k\to\infty}Z_{j,k}(t) = 0, \quad j = 1,2,\ldots,n. \tag{73}$$

In other words, $lim_{k\to\infty}|z_{1\phi,k}(t)| = lim_{k\to\infty}\|\zeta_{1,k}(t)\| \leq \frac{\sqrt{2}\aleph_1}{\eta_1}(1-e^{-\eta_1(t-T)})$, and then $lim_{k\to\infty}|z_{1,k}(t)| \leq \phi_{1,\infty}(t) + \frac{\sqrt{2}\aleph_1}{\eta_1}(1-e^{-\eta_1(t-T)})$, where $\aleph_1$ is the boundary of the difference between β_1 and α_1. Let η_1 be chosen sufficiently large, ensuring that $\phi_{1,\infty}(t)$ and $\frac{\sqrt{2}\aleph_1}{\eta_1}(1-e^{-\eta_1(t-T)})$ can be minimized as much as possible throughout the entire time interval [0, T].

Proof: In accordance with Assumption 2, we find that $\|Z_k(0)\|^2 = 0 \leq \|Z_k(T)\|^2$. Consider that $V'_{n,k} = V_{n,k}(Z_k(0), \hat{W}_{fk}(T), \hat{W}_{gk}(T), \hat{N}_k(T), \hat{M}_k(T))$. Using Equation 69, we obtain $Z_k = [Z_{1,k}, Z_{2,k},\ldots,Z_{n,k}]^T$, $\hat{W}_{fk} = [\hat{W}_{f1,k}, \hat{W}_{f2,k},\ldots, \hat{W}_{fn,k}]^T$,$\hat{W}_{gk} = [\hat{W}_{g1,k}, \hat{W}_{g2,k},\ldots,\hat{W}_{gn,k}]^T$, $\hat{M}_k = [\hat{M}_{1,k}, \hat{M}_{2,k},\ldots,\hat{M}_{n,k}]^T$, and$\hat{N}_k = [\hat{N}_{1,k}, \hat{N}_{2,k},\ldots,\hat{N}_{n,k}]^T$. Using Equation 72,

$$\begin{aligned}
V'_{n,k} \leq & \ V_{n,k}\Big(Z_{,k}(0), \hat{W}_{fk}(0), \hat{W}_{gk}(0), \hat{N}_k(0), \hat{M}_k(0)\Big) - \\
& \Sigma^k_{i=1}\Sigma^n_{j=1}\int_0^T \eta_j\Big(Z_{j,i}\Big)^2 dt + n\Big(\frac{1}{4}\Big)T\Big(\Sigma^k_{i=1}\Delta_i\Big).
\end{aligned} \tag{74}$$

Let $V_0(k) = V_{n,1}(Z_1(0),\ \hat{W}_{f1}(0), \hat{W}_{g1}(0), \hat{N}_1(0), \hat{M}_1(0)) + n(\frac{1}{4})T(\Sigma^k_{i=1}\Delta_i)$, then Equation 74 can be rewritten as

$$\Sigma^k_{i=1}\Sigma^n_{j=1}\int_0^T \eta_j\Big(Z_{j,i}\Big)^2 dt \leq V_0(k) - V'_{n,k}. \tag{75}$$

Using Equation 9, we obtain $lim_{k\to\infty}V_0(k) \leq V_{n,1} + 2an(\frac{1}{4})T$ and $V_0(k)$ is bounded. $V_{n,k}(Z_k(0), \hat{W}_{fk}(T), \hat{W}_{gk}(T), \hat{N}_k(T), \hat{M}_k(T)) \geq 0$, so

$$lim_{k\to\infty}\Sigma^n_{j=1}\int_0^T \eta_j\Big(Z_{j,k}\Big)^2 dt = 0. \tag{76}$$

Based on Equation 69, for any given value of k, $V_{n,k}(t) = V_{n,k}(0) + \int_0^t \dot{V}_{m,k}(\tau)d\tau$; substituting Equation 72 obtain

$$V_{n,k}(t) \leq V_{n,k}(0) - \Sigma^n_{j=1}\int_0^t \eta_j\big(Z_{j,k}(\tau)\big)^2 d\tau + tn\Big(\frac{1}{4}\Big)\Delta_k. \tag{77}$$

Based on Equation 76, $\Sigma^n_{j=1}\int_0^t \eta_j(Z_{j,k}(\tau))^2 d\tau$ is bounded. According to definition 1, Δ_k is bounded and $t \in [0,T]$, so $tn(\frac{1}{4})\Delta_k$ is also bounded. In addition, $\hat{W}_{fk}(0) = \hat{W}_{f(k-1)}(T)$, $\hat{W}_{gk}(0) = \hat{W}_{g(k-1)}(T)$, $\hat{M}_k(0) = \hat{M}_{k-1}(T)$, and $\hat{N}_k(0) = \hat{N}_{k-1}(T)$; based on Equation 77, for any given value of k,

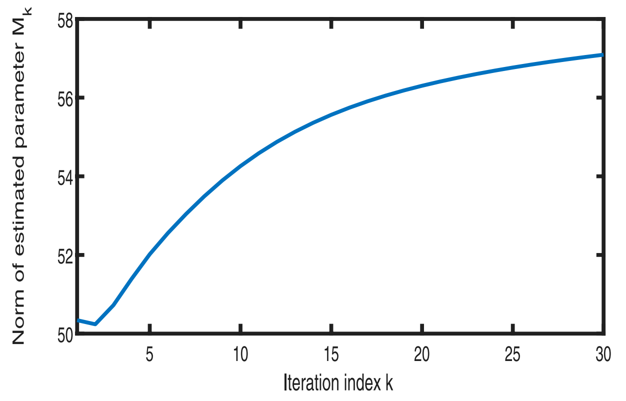


FIGURE 19
Variation in $\|\hat{M}_k\|$ according to the iteration index.

$V_{n,k}(Z_k(0), \hat{W}_{fk}(T), \hat{W}_{gk}(T), \hat{N}_k(T), \hat{M}_k(T))$ is bounded. So, $V_{n,k}(0, \hat{W}_{fk}(0), \hat{W}_{gk}(0), \hat{N}_k(0), \quad \hat{M}_k(0)) = V_{n,k-1}(0, \hat{W}_{f(k-1)}(T), \hat{W}_{g(k-1)}(T), \hat{N}_{k-1}(T), \hat{M}_{k-1}(T))$ is also bounded; from above all, for any given value of k, if $V_{n,k}(t)$ is bounded, then we can deduce that $x_{i,k}$, $\hat{W}_{fk}(t)$, $\hat{W}_{gk}(t)$, $\hat{N}_k(t)$, and $\hat{M}_k(t)$ are bounded. According to Equation 64, u_k is bounded. According to Equation 53, $\dot{Z}_{i,k}$ is bounded, so $Z_{i,k}$ is continuous uniformly. Thus, we can deduce Equation 73.

Then, we need to prove that $\aleph_1$ will converge to a neighborhood that approaches 0. Initially, let $\alpha_{i,k}(t)$ be a signal satisfying $|\alpha_{i,k}(t)| < \bar{\alpha}$ and $|\dot{\alpha}_{i,k}(t)| < \hbar$ for all $t \geq 0$. The compensation error within the compensation system is defined as

$$\varrho_{i,k} = \beta_{i,k} - \alpha_{i,k}. \tag{78}$$

With specified initial conditions, $\beta_{i,0} = \alpha_{i,0}$, i.e., $\varrho_{i,0} = 0, \quad i = 1, 2, \ldots, n-1$. From (11), we obtain

$$\begin{aligned}
\dot{\varrho}_{i,k} &= -\xi_{i,k}\left(\beta_{i,k} - \alpha_{i,k}\right) - \alpha_{i,k} \\
&= -\xi_{i,k}\varrho_{i,k} - \dot{\alpha}_{i,k} \\
\varrho_{i,k}(t) &= -\int_0^t \dot{\alpha}_{i,k} e^{-\xi_{i,k}(t-\tau)} d\tau \\
|\varrho_{i,k}(t)| &= |-\int_0^t \dot{\alpha}_{i,k}(\tau) e^{-\xi_{i,k}(t-\tau)} d\tau| \\
&= |\dot{\alpha}_{i,k}(\tau)| |\int_0^t e^{-\xi_{i,k}(t-\tau)} d\tau| \\
&\leq \max |\dot{\alpha}_{i,k}(\tau)| |\int_0^t e^{-\xi_{i,k}(t-\tau)} d\tau| \\
&\leq \frac{\hbar}{\xi_{i,k}}\left(1 - e^{-\xi_{i,k}t}\right) \\
&\leq \frac{\hbar}{\xi_{i,k}} = \aleph_i.
\end{aligned} \tag{79}$$

As shown in Equation 79, choosing an appropriate value for $\xi_{i,k}$ confines the error $\varrho_{i,k}$ within a narrow range, approximately equating $\alpha_{i,k}$ to $\beta_{i,k}$. In addition, based on the compensation system, the Lyapunov function is defined on the interval $[0, T]$ as follows:

$$V_{\zeta,k} = \sum_{i=1}^{n} \frac{1}{2}\zeta_{i,k}^2. \tag{80}$$

The derivative of $V_{\zeta,k}$ along systems (78) with respect to time is expressed as

$$\begin{aligned}
\dot{V}_{\zeta,k} &= \sum_{i=1}^{n} \zeta_{i,k}\dot{\zeta}_{i,k} \\
&= -\sum_{i=1}^{n} \eta_i \zeta_{i,k}^2 + \sum_{i=1}^{n-1} \zeta_{i,k}\left(\beta_{i,k} - \alpha_{i,k}\right) \\
&\leq -\sum_{i=1}^{n} \eta_i \zeta_{i,k}^2 + \sum_{i=1}^{n-1} |\zeta_{i,k}| |\left(\beta_{i,k} - \alpha_{i,k}\right)| \\
&= -\sum_{i=1}^{n} \eta_i \zeta_{i,k}^2 + \sum_{i=1}^{n-1} |\zeta_{i,k}| |\left(\beta_{i,k} - \alpha_{i,k}\right)| + 0|\zeta_{n,k}| \\
&\leq -\eta_0 \sum_{i=1}^{n} \zeta_{i,k}^2 + \aleph \sum_{i=1}^{n} |\zeta_{i,k}| \\
&\leq -\eta_0 \|\zeta_{i,k}\|^2 + \sqrt{2}\aleph \|\zeta_{i,k}\|,
\end{aligned} \tag{81}$$

where$\aleph = \max \aleph_i, \eta_0 = \min \eta_i$. To ensure the stability of the compensation system, it is sufficient to satisfy

$$\|\zeta_{i,k}\| \leq \frac{\sqrt{2}\aleph}{\eta_0}\left(1 - e^{-\eta_0(t-T)}\right). \tag{82}$$

Equation 82 leads to the conclusion that $\|\zeta_{i,k}\|$ is bounded. Hence, $\zeta_{i,k}$ is also bounded. Moreover, we can choose a parameter $\xi_{i,k} > 0$ to arbitrarily reduce $\aleph_i$, thereby causing the compensation $\zeta_{i,k}$ of the system to approach 0. In this way, by ensuring that the error Z_k approaches 0, $z_{\phi,k}$ will converge to the neighborhood approaching 0. Thus, we conclude Theorem 1.

4 Illustrative examples

4.1 Number simulation

This section includes an example illustrating the effectiveness of the proposed adaptive iterative learning controller.

The second-order pure-feedback nonlinear system described is considered as follows:

$$\begin{aligned}
\dot{x}_{1,k} &= x_{2,k} + \frac{r_1 x_{1,k} + r_1^2 x_{1,k}^2}{1 + r_1^2 x_{1,k}^2} \\
\dot{x}_{2,k} &= u_k + sin\left(r_2 x_{1,k} x_{2,k}\right) e^{-r_2^2 x_{1,k}^2 x_{2,k}^2} \\
y_k &= x_{1,k},
\end{aligned} \tag{83}$$

where $t \in [0, 5]$, $x_{1,k}$, and $x_{2,k}$ are state variables and u_k is the input variable. Utilizing the widely recognized van der Pol oscillator as the reference model, we obtain

$$\begin{aligned}
\dot{x}_{d1} &= x_{d2} \\
\dot{x}_{d2} &= -9x_{d1} - 6x_{d2} + 2 \\
y_{d1} &= x_{d1},
\end{aligned} \tag{84}$$

where x_{d1} and x_{d2} are state variables. The primary control objective is to synchronize the output of systems (82) with the reference trajectory y_{d1} generated by system (84) over the interval [0,5] under the condition $k \to \infty$.

In accordance with Theorem 1, the adaptive iterative learning controller is chosen as

$$\begin{aligned}
\alpha_{1,k} &= -\hat{W}_{1,k}^T S_1\left(\bar{x}_{1,k}, \hat{M}_{1,k}^T \Phi_1(t)\right) - \hat{N}_{1,k}\frac{1}{\Delta_k} z_{1\phi,k} + \dot{y}_r - \eta_1 z_{1,k} \\
u_k &= -\hat{W}_{f2,k}^T S_{f2}\left(\bar{x}_{2,k}, \hat{M}_{2,k}^T \Phi_2(t)\right) - \hat{N}_{2,k}\frac{1}{\Delta_k} Z_{2,k} \\
&\quad + \dot{\beta}_{1,k} - \eta_n z_{2,k} - z_{1\phi,k}.
\end{aligned} \tag{85}$$

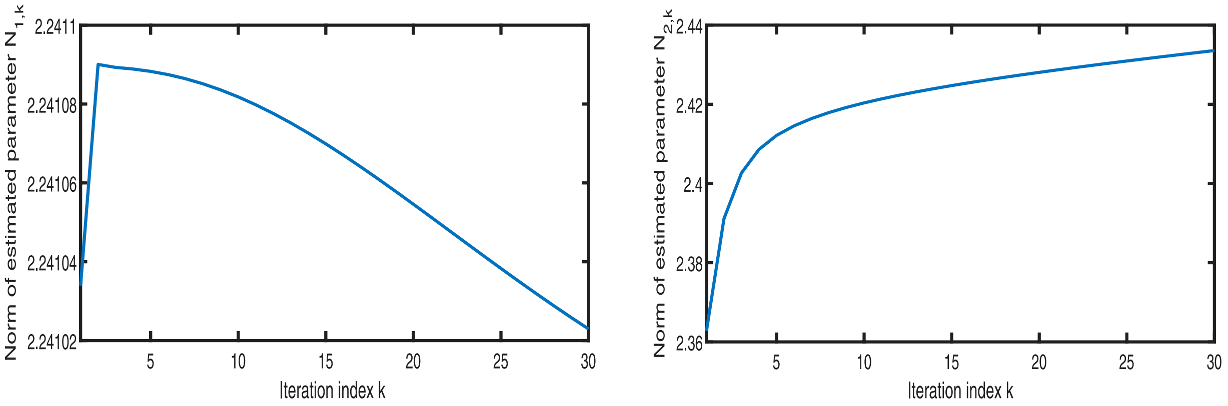


FIGURE 20
Variation in $\|\hat{N}_{1,k}\|$ and $\|\hat{N}_{2,k}\|$ according to the iteration index.

The error compensation mechanism is

$$\begin{aligned}\dot{\zeta}_{1,k} &= \beta_{1,k} + \zeta_{2,k} - \eta_1\zeta_{1,k} - \alpha_{1,k}\\ \dot{\zeta}_{2,k} &= -\eta_2\zeta_{2,k} - \zeta_{1,k},\end{aligned} \tag{86}$$

where $\dot{\beta}_{1,k} = -\xi\left(\beta_{1,k} - \alpha_{1,k}\right)$.

The parameter adaptive iterative learning laws are provided by (57):

$$\begin{aligned}\dot{\hat{W}}_{fi,k} &= -\Gamma_{fi1}\left(S_{fi}\left(\bar{x}_{i,k}, \hat{M}_{i,k}^{T}\Phi_i(t)\right) - \hat{S}'_{fi,k}\hat{M}_{i,k}^{T}\Phi_i(t)\right)Z_{i,k}\\ \dot{\hat{M}}_{i,k} &= -\Gamma_{mi1}\Phi_i(t)\hat{W}_{fi,k}^{T}\hat{S}'_{fi,k}Z_{i,k}\\ \dot{\hat{N}}_{i,k} &= \Gamma_{ni1}\frac{1}{\Delta_k}Z_{i,k}^2,\end{aligned} \tag{87}$$

where $i = 1,2$, $c_1 = 5$, $c_2 = 10, \Delta_k = a/k^2$, $a = 50000$, $\Gamma_{11} = diag\{1,1,1,1,1\}$, $\Gamma_{21} = 10$, $\Gamma_{12} = diag\{1,1,1,1,1\}$, $\Gamma_{22} = 1$, and $\xi = 1$. Figures 1–3 show the tracking performance of the system output and expected output without iteration and at 50th and 100th iterations, respectively. Figures 4, 5 show that as the number of iterations increases, the system error may converge to a small region near the zero point. Furthermore, observations shown in Figures 6–10 confirm that both control signals $\|u_k\|$ and $\|\alpha_k\|$ and estimated parameters, $\|\hat{W}_{1,k}\|$, $\|\hat{W}_{2,k}\|$, $\|\hat{M}_{1,k}\|$, $\|\hat{M}_{2,k}\|$, $\|\hat{N}_{1,k}\|$,and $\|\hat{N}_{2,k}\|$, exhibit bounded behavior within the [0,5] range. The validity of the control strategy presented in this research is reaffirmed by the simulation results shown in Figures 11–20 over the interval $[0, T]$.

4.2 Simulation of a single-joint robotic arm

In this section, we conducted simulation verification on a single degree-of-freedom robotic arm system to assess the performance of the proposed control method. The dynamic equation of a single degree-of-freedom robotic arm is

$$\frac{\partial^2\theta}{\partial t^2} = -10\sin\theta - 2\frac{\partial\theta}{\partial t} + u, \tag{88}$$

where θ is the angle between the robotic arm and the reference frame.u is the input of the DC motor.

$$\frac{\partial^2 y_{d1}}{\partial t^2} = -9y_{d1} - 6\frac{\partial y_{d1}}{\partial t} + 2r, \tag{89}$$

where y_{d1} is the output of the reference model. r is the reference input signal. According to Equations 88, 89, the state equation of the system is derived as

$$\begin{aligned}\dot{x}_{1,k} &= x_{2,k}\\ \dot{x}_{2,k} &= -10\sin\left(x_{1,k}\right) - 2x\left(2,k\right) + u_k\\ y_k &= x_{1,k},\end{aligned} \tag{90}$$

and its reference model is derived as

$$\begin{aligned}\dot{x}_{d1} &= x_{d2}\\ \dot{x}_{d2} &= -9x_{d1} - 6x_{d2} + 2r\\ y_{d1} &= x_{d1},\end{aligned} \tag{91}$$

where $x_{1,k}$ equals to θ can be defined as the angle between the robotic arm and the reference frame. $x_{2,k}$ is the time derivative of θ, i.e., $\dot{\theta}$. The primary control objective is to synchronize the output of systems (88) with the reference trajectory y_{d1} generated by system (89) over the interval [0,5] under the condition $k \to \infty$.

In accordance with Theorem 1, the adaptive iterative learning controller is chosen as

$$\begin{aligned}\alpha_{1,k} &= -\hat{N}_{1,k}\frac{1}{\Delta_k}z_{1\phi,k} + \dot{y}_{d1} - \eta_1 z_{1,k}\\ u_k &= -z_{1\phi,k} - c_2 z_{2,k} - \hat{W}_{2,k}^{T}S_2\left(\bar{x}_{2,k}, \hat{M}_{2,k}^{T}\Phi_2(t)\right) - \hat{N}_{2,k}\frac{1}{\Delta_k}z_{2\phi,k}\\ &\quad +\dot{\beta}_{1,k}.\end{aligned} \tag{92}$$

The error compensation mechanism is

$$\begin{aligned}\dot{\zeta}_{1,k} &= \beta_{1,k} + \zeta_{2,k} - \eta_1\zeta_{1,k} - \alpha_{1,k}\\ \dot{\zeta}_{2,k} &= -\eta_2\zeta_{2,k} - \zeta_{1,k},\end{aligned} \tag{93}$$

where $\beta_{1,k} = -\xi\left(\beta_{1,k} - \alpha_{1,k}\right)$.

The parameter adaptive iterative learning laws are provided by (57).

$$\dot{\hat{W}}_k = \Gamma_f\left(S\left(\bar{x}_{2,k}, \hat{M}_k^{T}\Phi(t)\right) - \hat{S}'_k\hat{M}_k^{T}\Phi_2(t)\right)z_{2\phi,k}, \tag{94}$$

$$\dot{\hat{N}}_{i,k} = \Gamma_{Ni}\frac{1}{\Delta_k}z_{i\phi,k}^2, \quad i = 1,2, \tag{95}$$

$$\dot{\hat{M}}_k = \Gamma_m\Phi(t)\hat{W}_k^{T}\hat{S}'_k z_{2\phi,k}, \tag{96}$$

where $c_1 = 50$, $c_2 = 150, \Delta_k = a/k^2$, $a = 50000$, $\Gamma_{11} = diag\{1,1,1,1,1\}, \Gamma_{21} = 10, \Gamma_{12} = diag\{1,1,1,1,1\}, \Gamma_{22} = 1, and \xi = 10$.

Figures 11–13 show the tracking performance of the system output and expected output without iteration and at 15th and 30th iterations, respectively. Figures 14, 15 show that as the number of iterations increases, the system error may converge to a small region near the zero point. Furthermore, observations from Figures 16–20 confirm that both control signals $\|u_k\|$ and $\|\alpha_k\|$ and estimated parameters, $\|\hat{W}_k\|$, $\|\hat{M}_k\|$, $\|\hat{N}_{1,k}\|$, and $\|\hat{N}_{2,k}\|$, exhibit bounded behavior within the [0,5] range. The validity of the control strategy presented in this research is reaffirmed by the simulation results shown in Figures 11–20 over the interval $[0, T]$.

5 Conclusion

This article presents a solution to the complete trajectory, following challenges within a finite time frame for a category of nonlinearly parameterized systems characterized by time-varying disturbed functions and initial state errors. A new FSE neural network is used to learn the time-varying, nonlinearly parameterized term. Based on this and Lyapunov theory, we proposed the new LPF-AILC method. A low-pass filter is used to solve the problem of parameter explosion after obtaining the derivative of the virtual controller. The unmatched uncertainties, nonlinear parameterization, and initial state errors are well considered. Two simulation examples have proven the feasibility of the control approach. This article does not mention time-delay issues, but they often exist in practical systems. Our future work should consider solving the complete tracking problem on a finite time interval for these complex systems with time delays. This is a more interesting issue. In addition, there are two deficiencies in the controller design process: the assumption of time-varying parameters being periodic and the jitter issues caused by the low-pass filter. These challenges will be carefully considered and addressed in our future work.

Data availability statement

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

Author contributions

CZ: conceptualization, funding acquisition, investigation, methodology, and writing–review and editing. LY: formal analysis, software, writing–original draft, and writing–review and editing. YG: investigation, validation, and writing–original draft. JY: funding acquisition, supervision, and writing–review and editing. FQ: funding acquisition, supervision, and writing–review and editing.

Funding

The authors declare that financial support was received for the research, authorship, and/or publication of this article. This work is supported by the National Natural Science Foundation (NNSF) of China (grants 62073259 and 61973094), the Natural Science Basis Research Plan in Shaanxi Province of China (2023-JC-QN-0752), the Science and Technology Plan Project of Xi'an City (No. 23GXFW0062), and the Shaanxi Provincial Key R& D Program General Project (No. 2024GX-YBXM-106).

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. Arimoto S, Kawamura S, Miyazaki F. Bettering operation of robots by learning. *J Robot Syst* (1984) 1(2):123–40. doi:10.1002/rob.4620010203

2. Moore KL. *Iterative learning control for Deterministic.* London: Springer-Verlag (1993).

3. Ahn HS, Chen YQ, Moore KL. Iterative learning control: brief survey and categorization. *IEEE Trans Syst Man Cybernetics-C, Appl Rev* (2007) 37(6): 1099–121. doi:10.1109/tsmcc.2007.905759

4. Xu J-X, Yan R, Chen Y-Q. "On initial conditions in iterative learning control," in 2006 American Control Conference, Minneapolis, MN, United States (2006). doi:10.1109/ACC.2006.1655358

5. Chen Y, Huang D, Xu C, Dong H. Iterative learning tracking control of high-speed trains with nonlinearly parameterized uncertainties and multiple time-varying delays. *IEEE Trans Intell Transportation Syst* (2022) 23(11):20476–88. doi:10.1109/tits.2022.3183608

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

7. Lu Y. Adaptive-Fuzzy control compensation design for direct adaptive fuzzy control. *IEEE Trans Fuzzy Syst* (2018) 26(6):3222–31. doi:10.1109/tfuzz.2018.2815552

8. Yang Z, Wang S. Adaptive prescribed performance control for nonlinear robotic systems. *J Franklin Inst* (2023) 360(2):1378–94. doi:10.1016/j.jfranklin.2022.10.044

9. Kong X, Yu F, Yao W, Cai S, Zhang J, Lin H. Memristor-induced hyperchaos, multiscroll and extreme multistability in fractional-order HNN: image encryption and FPGA implementation. *Neural Networks* (2024) 171:85–103. doi:10.1016/j.neunet.2023.12.008

10. Yu F, Kong X, Yao W, Zhang J, Cai S, Lin H, et al. Dynamics analysis, synchronization and FPGA implementation of multiscroll Hopfield neural networks with non-polynomial memristor. *Chaos, Solitons & Fractals* (2024) 179:114440. Article ID 114440. doi:10.1016/j.chaos.2023.114440

11. Hu X, Xu B, Hu C. Robust adaptive fuzzy control for HFV with parameter uncertainty and unmodeled dynamics. *IEEE Trans Ind Electronics* (2018) 65(11): 8851–60. doi:10.1109/tie.2018.2815951

12. Liu Z, Shi J, Zhao X, Zhao Z, Li H -X. Adaptive fuzzy event-triggered control of aerial refueling hose system with actuator failures. *IEEE Trans Fuzzy Syst* (2022) 30(8): 2981–92. doi:10.1109/tfuzz.2021.3098733

13. Hu X, Li Y -X, Tong S, Hou Z. Event-triggered adaptive fuzzy asymptotic tracking control of nonlinear pure-feedback systems with prescribed performance. *IEEE Trans Cybernetics* (2023) 53(4):2380–90. doi:10.1109/tcyb.2021.3118835

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

15. Pang N, Wang X, Wang Z. Event-triggered adaptive control of nonlinear systems with dynamic uncertainties: the switching threshold case. *IEEE Trans Circuits Syst Express Briefs* (2022) 69(8):3540–4. doi:10.1109/tcsii.2022.3164572

16. Liu X, Xu B, Cheng Y, Wang H, Chen W. Adaptive control of uncertain nonlinear systems via event-triggered communication and NN learning. *IEEE Trans Cybernetics* (2023) 53(4):2391–401. doi:10.1109/tcyb.2021.3119780

17. Yang X, Zheng X. Gradient descent algorithm-based adaptive NN control design for an induction motor. *IEEE Trans Syst Man, Cybernetics: Syst* (2021) 51(2):1027–34. doi:10.1109/tsmc.2019.2894661

18. Li K, Li Y. Adaptive neural network finite-time dynamic surface control for nonlinear systems. *IEEE Trans Neural Networks Learn Syst* (2021) 32(12):5688–97. doi:10.1109/tnnls.2020.3027335

19. Lin H, Deng X, Yu F, Sun Y. Grid multi-butterfly memristive neural network with three memristive systems: modeling, dynamic analysis, and application in police IoT. *IEEE Internet Things J* (2024) 1. doi:10.1109/jiot.2024.3409373

20. Li J, Wang C, Deng Q. Symmetric multi-double-scroll attractors in Hopfield neural network under pulse controlled memristor. *Nonlinear Dyn* (2024) 112:14463–77. doi:10.1007/s11071-024-09791-6

21. Chien CJ. A combined adaptive law for fuzzy iterative learning control of nonlinear systems with varying control tasks. *IEEE Trans Fuzzy Syst* (2008) 16(1): 40–51. doi:10.1109/tfuzz.2007.902021

22. Xu JX, Yan R. Adaptive learning control for finite interval tracking based on constructive function approximation and wavelet. *IEEE Trans Neural Networks* (2011) 22(6):893–905. doi:10.1109/tnn.2011.2132143

23. Taybi A, Chien CJ. A unified adaptive iterative learning control framework for uncertain nonlinear systems. *IEEE Trans Automatic Control* (2007) 52(10):1907–13. doi:10.1109/TAC.2007.906215

24. Ji H, Hou Z, Zhang R. Adaptive iterative learning control for high-speed trains with unknown speed delays and input saturations. *IEEE Trans Automation Sci Eng* (2016) 13(1):260–73. doi:10.1109/tase.2014.2371816

25. Li JM, Wang YL, Li XM. Adaptive iterative learning control for nonlinear parameterized-systems with unknown time-varying delays. *Control Theor Appl* (2011) 28(6):861–8. doi:10.7641/j.issn.1000-8152.2011.6.ccta091224

26. Zhang C-L, Li J-M. Adaptive iterative learning control for nonlinear time-delay systems with periodic disturbances using FSE-neural network. *Int J Automation Comput* (2011) 8(4):403–10. doi:10.1007/s11633-011-0597-x

27. Park BH, Kuc TY, Lee JS. Adaptive learning of uncertain robotic systems. *Int J Control* (1996) 65(5):725–744. doi:10.1080/00207179608921719

28. Choi JY, Lee JS. Adaptive iterative learning control of uncertain robotic systems. *Proc Inst Elect Eng D* (2000) 147(2):217–23. doi:10.1049/ip-cta:20000138

29. Liu G, Hou Z. Adaptive iterative learning control for subway trains using multiple-point-mass dynamic model under speed constraint. *IEEE Trans Intell Transportation Syst* (2021) 22(3):1388–400. doi:10.1109/tits.2020.2970000

30. Ji H, Hou Z, Zhang R. Adaptive iterative learning control for high-speed trains with unknown speed delays and input saturations. *IEEE Trans Automation Sci Eng* (2016) 13(1):260–73. doi:10.1109/tase.2014.2371816

31. Geng Y, Ruan X, Xu J. Adaptive iterative learning control of switched nonlinear discrete-time systems with unmodeled dynamics. *IEEE Access* (2019) 7:118370–80. doi:10.1109/access.2019.2936763

32. Heinzinger G, Fenwick D, Paden B, Miyazaki F. Stability of learning control with disturbances and uncertain initial conditions. *IEEE Trans Automat Contr* (1992) 37: 110–4. doi:10.1109/9.109644

33. Chien CJ, Liu JS. AP-type iterative learning controller for robust output tracking of nonlinear time-varying systems. *Int J Control* (1996) 64(2):319–34. doi:10.1080/00207179608921630

34. Park KH, Bien Z, Hwang DH. A study on the robustness of a PID-type iterative learning controller against initial state error. *Int J Syst Sci* (1999) 30(1):49–59. doi:10.1080/002077299292669

35. Sun M, Wang D. Iterative learning control with initial rectifying action. *Automatica* (2002) 38:1177–82. doi:10.1016/s0005-1098(02)00003-1

36. Chien CJ, Hsu CT, Yao CY. Fuzzy system based adaptive iterative learning control for nonlinear plants with initial state errors. *IEEE Trans Fuzzy Syst* (2004) 12(5):724–32. doi:10.1109/tfuzz.2004.834806

37. Wu XJ, Wu XL, Ran Zhen XY, Luo XJ, Zhu QM. Adaptive control for time-delay non-linear systems with non-symmetric input non-linearity. *Int J Model Identification Control* (2011) 13(3):152–60. doi:10.1504/ijmic.2011.041302

38. Luo XY, Wu XJ, Guan XP. Adaptive backstepping fault-tolerant controlfor unmatched non-linear systems against actuator dead-zone. *IET Control Theor Appl.* (2010) 4(5):879–88. doi:10.1049/iet-cta.2009.0086

39. Wu X, Wu X, Luo X, Zhu Q, Guan X. Neural network-based adaptive tracking control for nonlinearly parameterized systems with unknown input nonlinearities. *Neurocomputing* (2012) 82:127–42. doi:10.1016/j.neucom.2011.10.019

40. Chen WS. Adaptive backstepping dynamic surface control for systems with periodic disturbances using neural networks. *IET Control Theor Appl* (2009) 3(10): 1383–94. doi:10.1049/iet-cta.2008.0322

41. Zhu S, Ming-Xuan SUN, Xiong-Xiong HE. Iterative learning control of strict-feedback nonlinear time-varying systems. *ACTA AUTOMATICA SINICA* (2010) 36(3): 454–8. doi:10.3724/SP.J.1004.2010.00454

42. Yu H, Li Y. *Adaptive obstacle avoidance control based on first-order low-pass filter for cleaning robots*. Marseille, France: OCEANS 2019 - Marseille (2019). p. 1–6.

43. Zhang C, Tian X, Gao Y, Qian F. Command filter AILC for finite time accurate tracking of aircraft track angle system based on fuzzy logic. *Adv Math Phys* (2023) 2023: 11. Article ID 4744873. doi:10.1155/2023/4744873

44. Huang J -T, Pham T -P. Differentiation-free multiswitching neuroadaptive control of strict-feedback systems. *IEEE Trans Neural Networks Learn Syst* (2017) 29(4): 1095–107. doi:10.1109/tnnls.2017.2651903

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 09 September 2024
DOI 10.3389/fphy.2024.1469418

OPEN ACCESS

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

REVIEWED BY
Jun Mou,
Dalian Polytechnic University, China
Li Xiong,
Hexi University, China
Mengjiao Wang,
Xiangtan University, China

*CORRESPONDENCE
Tianxiu Lu,
✉ lubeeltx@163.com

RECEIVED 23 July 2024
ACCEPTED 19 August 2024
PUBLISHED 09 September 2024

CITATION
Chen Y, Lu T, Chen C and Xiang Y (2024) A novel image encryption method based on improved two-dimensional logistic mapping and DNA computing.
Front. Phys. 12:1469418.
doi: 10.3389/fphy.2024.1469418

A novel image encryption method based on improved two-dimensional logistic mapping and DNA computing

Yuanlin Chen[1], Tianxiu Lu[1]*, Caiwen Chen[1] and Yi Xiang[1,2]

[1]College of Mathematics and Statistics, Sichuan University of Science and Engineering, Zigong, China, [2]South Sichuan Applied Mathematics Research Center, Zigong, China

In the digital era, the significance of cryptographic algorithms has grown significantly within the realm of cybersecurity. This research presents an innovative approach to image encryption that eliminates the security limitations of the conventional one-dimensional logistic mapping. This approach relies on an enhanced two-dimensional logistic-fraction hybrid chaotic mapping (2D-LFHCM) and deoxyribonucleic acid (DNA) computing. Initially, the improved 2D-LFHCM is utilized to effectively scramble the image by incorporating chaotic sequences. Then, two novel algebraic DNA computing rules are introduced to enhance diffusion encryption. Experimental findings show that this approach offers superior security performance, even with renowned attacks.

KEYWORDS

image encryption, chaotic system, DNA computing, logistic mapping, 2D-LFHCM

1 Introduction

Chaos, which refers to complex and unpredictable behavior displayed by nonlinear dynamic systems, is a phenomenon characterized by the inherent unpredictability of deterministic nonlinear systems. The slightest change in the initial state can lead to unforeseen results. Chaos is not restricted to a particular domain but can be observed in various aspects of human society. The profound exploration of chaos has given rise to a natural problem: what are the potential applications of chaos? This query stands as a paramount concern not only in the present world but also in the future. As fundamental and applied sciences progress, chaos theory has evolved into a crucial focal point within the realm of nonlinear science, blossoming into a discipline that has thrived over the past few decades. Contemporary electronic engineering and image processing heavily draw upon chaos theory, utilizing its principles to yield numerous innovative and advantageous advancements in these fields.

The characteristics of chaos systems include nonlinearity, ergodicity, pseudo-random behavior, and a high sensitivity to initial conditions. As a result, chaos theory serves as a solid foundation for the development of excellent image encryption algorithms. However, it has been observed that employing a single chaotic system often leads to a limited range of possible encryption keys, thereby rendering the algorithm susceptible to attacks from malicious entities. Consequently, to ensure the creation of a robust and efficient image encryption algorithm, researchers frequently integrate chaotic systems with other disciplines, including the analysis of deoxyribonucleic acid (DNA) sequences [1–4], the utilization of optical maps [5,6] or cellular automata (CA) [7,8], the application of compressed sensing [9,10], and chaotic circuits [11–17].

Therefore, chaos theory holds immense potential for research and practical significance in the domain of image encryption. Ever since R. Matthews [18] introduced a broader logistic map and relied on it in the data encryption domain, a new era of chaotic systems generating pseudo-random numbers is beginning. Thus, fresh impetus is provided to cryptography. Consequently, chaos and cryptography became intertwined. Subsequently, Alvarez [19] formulated the fundamental requisites and rules of chaotic cryptosystems, gaining recognition from experts in the field of cryptography. Since then, there has been a robust development regarding chaotic digital image encryption. In 2012, Wang [20] invented a novel technique employing a traditional logistic map for the encryption of color images. Nevertheless, the key space induced by one-dimensional chaos is limited, and the algorithm's handling of chaotic sequences is not sufficient, resulting in unsatisfactory robustness of the algorithm. In an attempt to address this issue, Wang [21] put the latest method for creating high-dimensional digital chaotic systems, but the drawback lies in the complexity of the system structure and the inefficiency of the algorithms. More recently, Huang [22] proposed a fine-tuned cubic color image encryption scheme that operates jointly by chaos and hyperchaos. Its core idea is based on an improved logistic-fraction hybrid chaotic mapping (LFHCM) proposed to address the limitations of one-dimensional chaotic mapping and expand the key space. This mapping is then linked with a four-dimensional hyperchaotic system to generate the key stream, which is used to rotate and shift the rows and columns of each component in the red (R), green (G), and blue (B) channels of the color image. Wang [23] attempted to accomplish global scrambling by creating a chaotic sequence using the Lorenz system for binary and Gray code translation. Remarkably, this algorithm exhibits a favorable encryption effect on grayscale images. Building upon these advancements, Gao [24] introduced a multi-image encryption technique founded on single-channel scrambling, diffusion, and chaotic systems. Performance investigation validates that this technique demonstrates exceptional capabilities in ensuring security and achieving efficient encryption speed. Furthermore, in his study [25], Alexan proposes a method for encrypting color images. This approach effectively combines KAA mapping with various chaotic mappings in a synergistic manner. Notably, this approach maximizes the utilization of Shannon's security idea and encrypts the image through bit obfuscation and diffusion.

However, amidst a plethora of algorithms, our specific interest lies in encryption methods rooted in chaotic dynamics and deoxyribonucleic acid (DNA) sequences. The encryption performance of this algorithm, proposed by Chai [26], is not only exceptional but also demonstrates the ability to withstand a range of conventional attacks. In 2018, an image encryption algorithm was introduced by Wu [27], which employed a combination of DNA coding and Henon-Sine mapping. To increase the complexity of the encryption process and strengthen the algorithm's security, XOR operations and DNA coding were added to the diffusion process. In 2020, Patel [28] introduced a novel algorithm for encrypting images, which combined DNA coding and a three-dimensional chaotic mapping technique. In addition to utilizing the idea of eight complementary encodings for picture encryption, this approach employed a chaotic sequence to jumble the image. Both of these algorithms are applicable for encrypting grayscale and color images. Liu [29] then applies an improved Arnold transformation to scramble the three components and uses the DNA sequence generated through the chaotic sequence to conduct diffusion encryption of the color image. Hua [30] presented an innovative dynamic image encryption technique that enhanced the security of image data by utilizing quantum walk and chaos-induced DNA. Inspired by them, a plagiarism detection method is presented utilizing an improved two-dimensional logistic-fraction hybrid chaotic mapping (2D-LFHCM) and DNA computation. This method incorporates DNA chaotic diffusion and scrambling techniques.

The organization of this paper is outlined below. Section 2 delves into the 2D-LFHCM and analyzes its chaotic characteristics. The fundamental principles of encryption and decoding are covered in Section 3. Section 4 presents the devised method for key creation as well as the encryption and decryption methods for DNA images. Section 5 elucidates the numerical simulation findings of the proposed cryptosystem, supplemented by a comprehensive exploration of its security analysis. Ultimately, Section 6 furnishes a thorough recapitulation of the study's content and outlines potential directions for future research.

The main contributions of this paper are highlighted below:

(i) Development of an enhanced two-dimensional logistic-fraction hybrid chaotic mapping (2D-LFHCM) for image encryption.
(ii) Design and implementation of novel deoxyribonucleic acid (DNA) computing techniques in the proposed encryption method, including right shift addition, right shift subtraction, right shift XOR, and other DNA computing methods.
(iii) A comprehensive performance analysis of the encryption algorithm was conducted, including aspects such as encryption speed, key space, histograms, information entropy, and correlation coefficients.

2 An improved 2D-LFHCM

2.1 The definition of 2D-LFHCM

Parabolic mapping is a generic term used to describe a kind of chaotic maps. The classical insect population model (or logistic mapping, shortly, LM) is represented as Equation 1.

$$x_{n+1} = r x_n (1 - x_n), \tag{1}$$

where $r \in (0,4)$, with initial value $x_0 \in (0,1)$. Another classic 2D-LMM (two-dimensional logistic mixing mapping) [31], is a discrete chaotic map in two dimensions derived from the traditional logistic map. The difference equation's mathematical model is represented as Equation 2.

$$\begin{cases} x_{n+1} = t(3y_n + 1)x_n(1 - x_n); \\ y_{n+1} = t(3x_{n+1} + 1)y_n(1 - y_n), \end{cases} \tag{2}$$

where t is a control parameter, x_n and y_n denote the state variables within the iterative process of the difference equation. Compared with the traditional 2D-LMM, the newly proposed 2D-LM (two-dimensional logistic mapping) by Ye [32] is a two-dimensional chaotic mapping with a simpler equation structure. Its model is described below.

$$\begin{cases} x_{n+1} = u x_n \left(1 - x_n\right); \\ y_{n+1} = v x_n \left(1 - y_n\right). \end{cases} \tag{3}$$

In Equation 3, u and v are the control parameters of the proposed 2D-LM, x_n and y_n are the state variables, and n is the number of iteration steps. When $u = 3.99$ and $v = 1.4$, starting from $(0.1, 0.1)$, the 2D-LM demonstrates chaotic behavior.

Based on the original one-dimensional logistic map, the LFHCM (logistic-fraction hybrid chaotic mapping) derived from the logistic map and fraction map is proposed by Huang [22]. The fraction mapping is proposed by Lu et al. [33] to address the practical needs of multi-objective optimization and multi-model issues. The definition equation of fraction mapping is Equation 4.

$$z_{n+1} = F\left(c, z_n\right) = \frac{1}{z_n^2 + 0.1} - c z_n, \tag{4}$$

where $c \in (0, 1]$ is a control parameter, and the output range of all chaotic sequences $z_n \in [-10.0025, 10.0025]$. The definition equation of LFHCM constructed by combining logistic mapping and fraction mapping is Equation 5.

$$x_{n+1} = L\left(a, x_n\right) = a x_n \left(1 - x_n\right)^2 \times \frac{1}{x_n^2 + 1}, \tag{5}$$

where $a \in (0, 11.5]$ is a control parameter, and the sequence output value $x_n \in [0, 1.56]$.

Thanks to their excellent chaotic performance, LM, 2D-LM, 2D-LMM, and LFHCM are often used as pseudo-random signal generators in engineering fields such as cryptography and dynamics. However, LFHCM has not yet been extended to two-dimensional. The traditional logistic-fraction mapping serves as the fundamental basis for the 2D-LFHCM described in this study, and its difference equation is

$$\begin{cases} x_{n+1} = \lambda x_n \left(1 - x_n\right)^2 \times \dfrac{1}{x_n^2 + 1}; \\ y_{n+1} = \mu x_n \left(1 - y_n\right)^2 \times \dfrac{1}{x_n^2 + 1}, \end{cases} \tag{6}$$

where, λ and μ serve as the control parameters, and x_n and y_n stand for the state variables. When $u = 3.99$ and $v = 1.4$, starting from the initial point $(0.1, 0.1)$, chaotic behavior is observed in the 2D-LFHCM.

2.2 Analysis and comparison of chaotic properties of 2D-LFHCM

In the preceding section, different classical maps were defined, and enhancements were made to the two-dimensional map, referred to as 2D-LFHCM. This section assesses and compares the chaotic properties of the following chaotic maps: 2D-LM, 2D-LFHCM, 2D-LMM, and LFHCM. The study is done from the perspectives of the phase trajectory diagrams, Lyapunov exponents, bifurcation diagrams, and chaotic analysis of the iterative sequences. It will be shown that the improved two-dimensional chaotic map 2D-LFHCM has better chaotic characteristics.

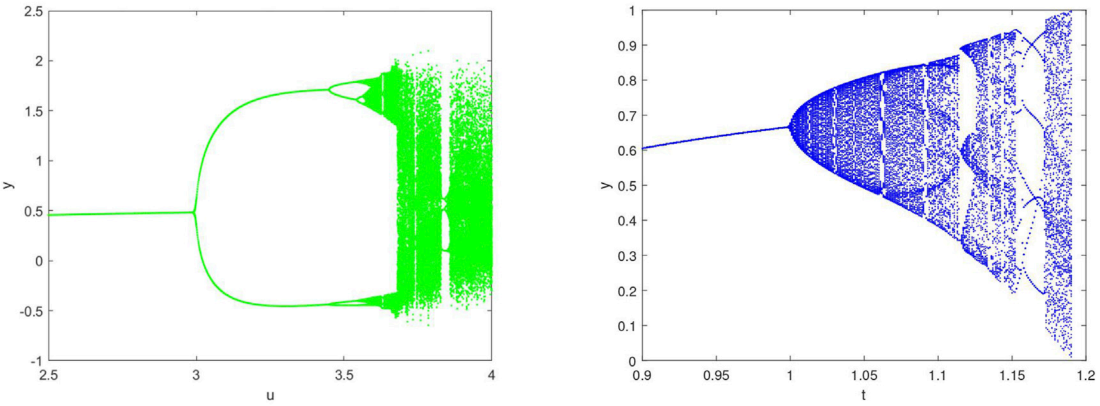
2.3 Bifurcation diagrams

Assume that the initial conditions of the following four mappings are (0.1, 0.1), and their control parameters are a, t, u, and λ, respectively. Then, their bifurcation diagrams are shown in Figure 1. The bifurcation diagram of 2D-LM is shown in Figure 1A. When $u = 2.99$, the system transitions from a period-1 to a period-2 state. At $u = 3.464$, the system enters a period-4 orbit. When $u = 3.554$, the system enters a period-8 orbit and then transitions into a chaotic orbit. The maximum amplitude of 2D-LM is 2.491. The bifurcation diagram for the 2D-LMM is presented in Figure 1B. As the control parameter t increases from 0.9 to 1.19, the trajectory of point y of 2D-LMM undergoes a transition, shifting from a periodic orbit to a chaotic orbit, and the maximum amplitude is 0.995. The bifurcation diagram for LFHCM is displayed in Figure 1C. When $a = 5.9$, LFHCM enters a chaotic state. The 2D-LFHCM model proposed in this paper, as shown in Figure 1D, when the control parameters $\lambda = 5.206$ and $\lambda = 5.509$, the tangent bifurcation of the mapping occurs, and the obvious period-2 window and period-4 window are formed, respectively. Then the mapping forms the period-8 window, and then enters the chaotic state. Changing the parameter λ, it can be observed that the mapping has rich nonlinear dynamic phenomena such as period-doubling bifurcation, tangent bifurcation, periodic window, chaos, and so on. In addition, it can be seen that whether 2D-LMM or 2D-LM, the length of the chaotic interval is less than 1, and there are some glaringly visible blank windows even inside the narrow chaotic region. It is evident from a comparison of the newly proposed 2D-LFHCM with the above chaotic maps that it has a broader chaotic region, a longer chaotic interval, and fewer blank windows. The comparison of their chaotic intervals is shown in Table 1. Where, the chaotic region area of 2D-LM is regarded as unit 1.

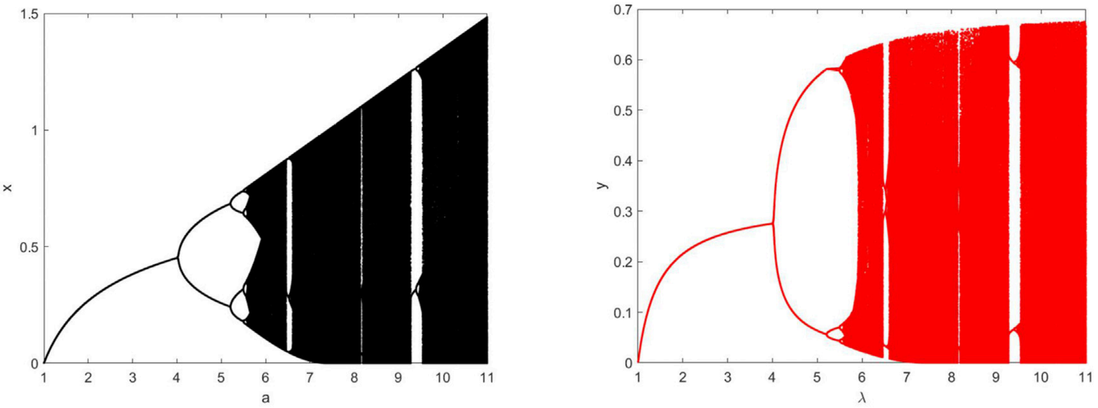
2.4 Lyapunov exponents spectrum

In general, the Lyapunov exponent is a very important statistical feature. It characterizes the stability of dynamic systems and can be used to judge whether the system presents chaotic behavior and the degree of chaos. The Lyapunov exponent describes the exponential growth rate of the system under small changes in initial conditions, which reflects the sensitivity and predictability of the system. To rephrase, determining the Lyapunov exponent spectrum can aid in our comprehension of the system's dynamic behavior, as well as in determining whether or not chaos exists inside the system and to what extent. For a discrete chaotic mapping $L(x)$ of dimension m (see Equation 7),

$$L(x)\colon \begin{cases} x_{n+1}^1 = L_1\left(x_n^1, \ldots x_n^m\right); \\ x_{n+1}^2 = L_2\left(x_n^1, \ldots x_n^m\right); \\ \vdots \\ x_{n+1}^m = L_m\left(x_n^1, \ldots x_n^m\right), \end{cases} \tag{7}$$



a. Bifurcation diagram of the 2D-LM. **b.** Bifurcation diagram of the 2D-LMM.



c. Bifurcation diagram of the LFHCM. **d.** Bifurcation diagram of the 2D-LFHCM.

FIGURE 1
The bifurcation diagrams of 2D-LM **(A)**, 2D-LMM **(B)**, LFHCM **(C)**, and 2D-LFHCM **(D)**.

TABLE 1 Comparison of chaotic regions of four chaotic maps.

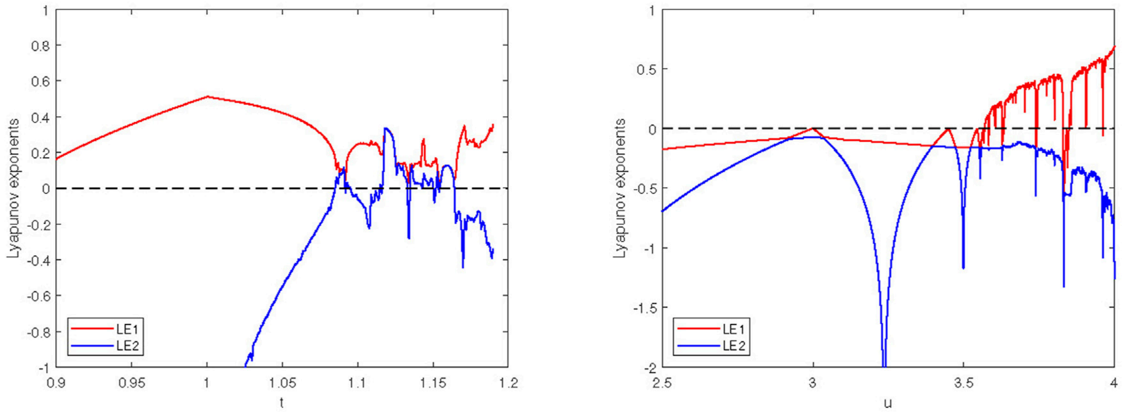
Chaotic map	Chaotic interval	Chaotic region area ratio
2D-LM	$[3.567, 3.738] \cup [3.749, 3.828] \cup [3.848, 4]$	1
2D-LMM	$[0.999, 1.089] \cup [1.089, 1.113] \cup [1.128, 1.152] \cup [1.172, 1.189]$	1.374
LFHCM	$[5.569, 6.464] \cup [6.577, 8.123] \cup [8.214, 9.274] \cup [9.461, 11]$	1.921
2D-LFHCM	$[5.583, 6.454] \cup [6.565, 9.274] \cup [9.481, 11]$	2.441

the Lyapunov exponent can be expressed as Equation 8.

$$LE_j = \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \ln|\lambda_j|, \tag{8}$$

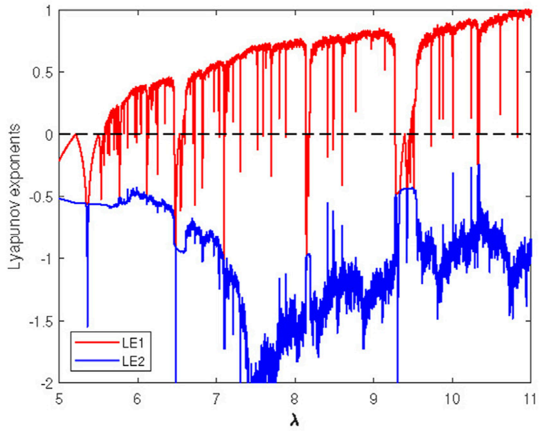
where $j = 1, 2, \ldots, m$, and $\lambda_1, \lambda_2, \ldots, \lambda_m$ are the m eigenvalues of the Jacobian matrix of $L(x)$ at the n-th iteration.

Two Lyapunov exponents, LE_1 and LE_2, correspond to a two-dimensional discrete chaos mapping. In terms of the Lyapunov exponent, a system will only exhibit chaotic properties when it has a positive number of states. Furthermore, the system performs more chaotically the higher the Lyapunov exponent. Selecting the control parameters $\nu = \mu = 4$, the initial point is (0.1, 0.1), Figure 2 shows the Lyapunov exponents spectrum of three two-dimensional chaotic maps. The largest Lyapunov exponent (LE_1) is shown by the red line, while the second Lyapunov exponent (LE_2) is represented by the blue line. The comparative analysis reveals that the average Lyapunov exponent of the 2D-LFHCM introduced in this study surpasses that of both the 2D-LMM and 2D-LM. Consequently, the 2D-LFHCM exhibits superior chaotic performance. Moreover, the Lyapunov exponent values within the parameter range of λ for the 2D-LFHCM are predominantly positive, confirming its heightened suitability for image encryption.



a. Lyapunov exponent of the 2D-LMM.

b. Lyapunov exponent of the 2D-LM.



c. Lyapunov exponent of the 2D-LFHCM.

FIGURE 2
Lyapunov exponents spectrum of 2d-LMM **(A)**, 2D-LM **(B)**, 2D-LFHCM **(C)**.

2.5 Iteration sequence and phase diagram

For the 2D-LFHCM, with fixed parameters $\lambda = 4.5$, $\mu = 1.4$ and $\lambda = 6$, $\mu = 1.4$, the chaotic sequence is obtained after 300 iterations, as shown in Figure 3. The black curve S_1 represents the trajectory starting from the initial value (0.1, 0.1). The green curve S_2 represents the trajectory starting from the initial value $(0.1, -0.1)$. To make the image clear, the curve of S_1 is intentionally translated upward. From Figure 3B, it becomes apparent that upon reaching a specific iteration count, the two running paths become indistinguishable. Indeed, this phenomenon arises when certain conditions are met by the initial value.

Based on diverse parameters, maps in the specified interval can generate chaotic effects, resulting in a chaotic phase diagram. Figure 4 illustrates the chaotic phase portraits of 2D-LMM, 2D-LM, and 2D-LFHCM under specific conditions ($t = 1.19$, $u = 3.99$, $v = 1.4$, and $\lambda = 9$, $\mu = 1.4$, respectively).

By analyzing the numerical simulation results presented in Figure 4, it becomes evident that the 2D-LFHCM proposed in this research exhibits a larger chaotic range in the phase plane compared to 2D-LMM and 2D-LM. This observation indicates that the 2D-LFHCM can generate a more diverse range of chaotic pseudo-random outcomes, thereby enhancing ergodicity. This improvement is valuable for potential applications, including signal generation and the utilization of chaotic systems in image encryption.

3 The basic principles of encryption and decryption

In the field of biology, deoxyribonucleic acid (DNA) stands as a fundamental biomolecule present within the cells of all organisms, serving as the genetic material for the majority of living entities. It is gratifying that DNA also plays an indispensable role in cryptography [34]. If the nucleotide bases in DNA information are matched to the binary digits 00, 01, 10, and 11, there are a total of 8 DNA coding rules [35], each corresponding to its own rules for addition, subtraction, and XOR operations. DNA primarily achieves the genetic code through the arbitrary combination of four bases: adenine (A), cytosine (C), guanine (G), and thymine (T), where A and T are complementary, C and G are complementary. The binary numerals 0 and 1, which complement each other, also serve a

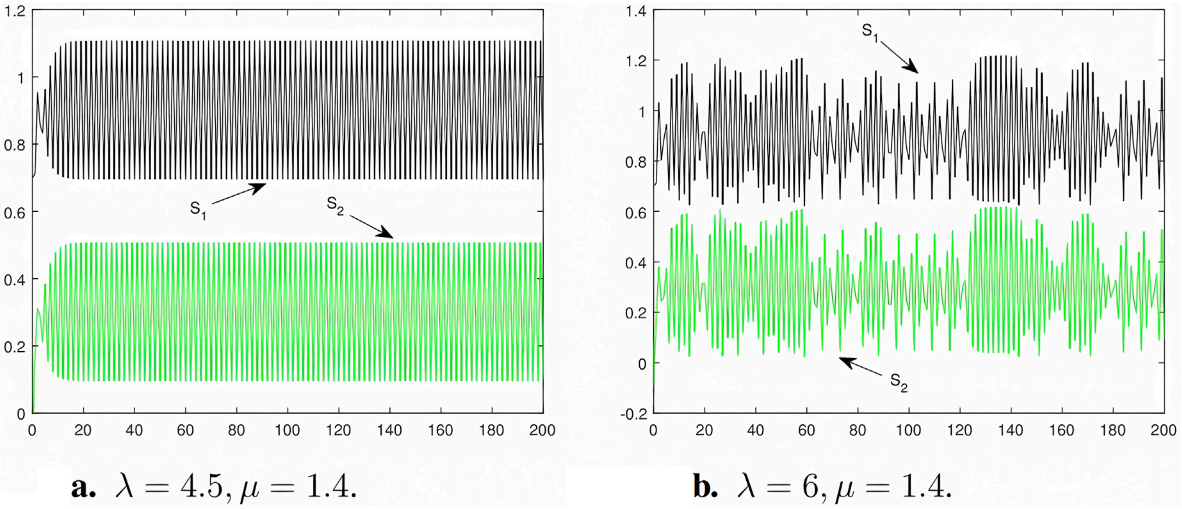


FIGURE 3
Two iterative sequences of the 2D-LFHCM for $\mu = 1.4$ ($\lambda = 4.5$ **(A)** and $\lambda = 6$ **(B)**, respectively).

purpose in computers to store information. By leveraging these characteristics, when implementing DNA's quaternary encoding with four bases, there can be a total of eight pairing rules. The coding table is shown in Table 2.

In a computer, the quaternary system is a digital system based on the number 4. The four numbers 0, 1, 2, 3, and A, T, C, G one-to-one mapping. If four bases in DNA are used for four-image coding, there are a total of eight rules that can be paired with each other. The coding table is shown in Table 2. Following the rules provided in Table 2, a 4-digit quaternary number can be directly represented by a 4-length DNA sequence. As an example, the quaternary number "1320" can be used to represent the decimal 120 Gy value. Since the numbers 0, 1, 2, and 3 are mapped one by one with A, T, C, and G, they are eventually converted into TGCA.

The cryptosystem in modern cryptography can be succinctly denoted as a five-tuple P, C, K, Enc, Dec, where P denotes the plaintext sequence, C represents the ciphertext sequence, K embodies the key system, Enc signifies the encryption algorithm, and Dec denotes the decryption algorithm. The core idea of modern cryptography involves encrypting a sequence of plaintext using a designated encryption algorithm. Subsequently, the encrypted file can be decrypted by the recipient, using a specific decryption key, to retrieve the original plaintext sequence. Table 3 displays the DNA operation rules, when A = 0, C = 1, G = 2, and T = 3, of addition "+," subtraction "–", exclusive or "xor," right shift '→', and left shift "←".

Mathematically, the well-known technique of the right cyclic shift involves rearranging a collection of data sequences. The specific procedure entails relocating the final number to the initial position and shifting all the remaining elements to the right, aligning them with their corresponding positions. On the other hand, the left circulation shift is similar. Throughout the shifting process, the cyclicity is maintained, ensuring that the removed element reappears at the opposite end of the sequence.

Let $R((s_0, s_1, \ldots, s_{n-1}), k)$ represents the k-th right cyclic shift, that is, the right cyclic shift k times. Then,

$$R((s_0, s_1, \ldots, s_{n-1}), k) = \left(s_{\mathrm{mod}(0-k,n)}, s_{\mathrm{mod}(0-k,n)+1} \cdots, s_{\mathrm{mod}(n-1-k,n)}\right).$$

Correspondingly, $L((s_0, s_1, \ldots, s_{n-1}), k)$ represents the kth left cyclic shift. Then,

$$L((s_0, s_1, \ldots, s_{n-1}), k) = \left(s_{\mathrm{mod}(0+k,n)}, s_{\mathrm{mod}(0-k,n)+1} \cdots, s_{\mathrm{mod}(n-1+k,n)}\right).$$

As per the operational guidelines provided in Table 3, DNA left shift and DNA right shift algebraic operators, grounded in DNA sequences, facilitate the definition of six DNA algebraic operations. These include DNA right (left) shift addition, DNA right (left) shift subtraction, and DNA right (left) shift XOR.

As an illustration, for the DNA operation before the shift and the DNA right shift XOR, one can get

$$\begin{aligned}
&((A,C,G,T),\ (A)_4) = ((A,C,G,T),0) \rightarrow (A,C,G,T) \rightarrow (A,A,A,A),\\
&((A,C,G,T),\ (C)_4) = ((A,C,G,T),1) \rightarrow (T,A,C,G) \rightarrow (T,C,T,C),\\
&((A,C,G,T),\ (G)_4) = ((A,C,G,T),2) \rightarrow (G,T,A,C) \rightarrow (G,G,G,G),\\
&((A,C,G,T),\ (T)_4) = ((A,C,G,T),3) \rightarrow (C,G,T,A) \rightarrow (C,T,C,T).
\end{aligned}$$

The DNA right shift addition f_{r+} is expressed as

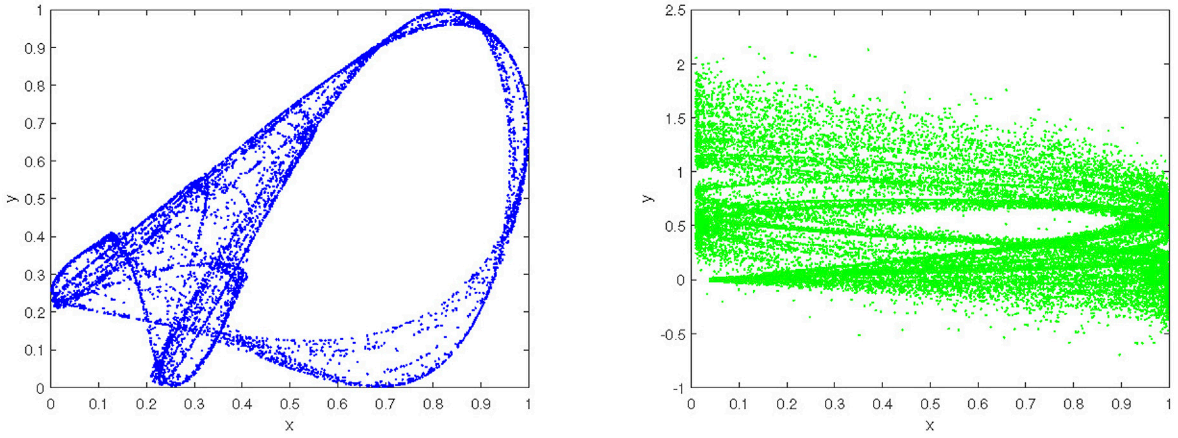
$$f_{r+} = \begin{bmatrix} R_+((A,C,G,T),\ (A)_4) \\ R_+((A,C,G,T),\ (C)_4) \\ R_+((A,C,G,T),\ (G)_4) \\ R_+((A,C,G,T),\ (T)_4) \end{bmatrix}$$

$$= \begin{bmatrix} R_+((A,C,G,T),0) \\ R_+((A,C,G,T),1) \\ R_+((A,C,G,T),2) \\ R_+((A,C,G,T),3) \end{bmatrix} \rightarrow \begin{bmatrix} A & A & C & C \\ T & C & T & A \\ G & G & G & G \\ C & T & A & T \end{bmatrix}$$

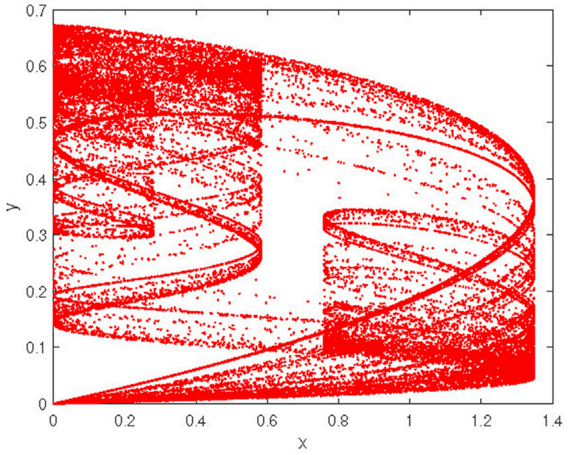
The DNA right shift subtraction f_{r-} is expressed as

$$f_{r-} = \begin{bmatrix} R_-((A,C,G,T),\ (A)_4) \\ R_-((A,C,G,T),\ (C)_4) \\ R_-((A,C,G,T),\ (G)_4) \\ R_-((A,C,G,T),\ (T)_4) \end{bmatrix}$$

$$= \begin{bmatrix} R_-((A,C,G,T),0) \\ R_-((A,C,G,T),1) \\ R_-((A,C,G,T),2) \\ R_-((A,C,G,T),3) \end{bmatrix} \rightarrow \begin{bmatrix} A & A & A & A \\ T & C & G & C \\ G & G & T & T \\ C & T & C & G \end{bmatrix}$$



a. The phase diagram of the 2D-LMM. **b.** The phase diagram of the 2D-LM.



c. The phase diagram of the 2D-LFHCM.

FIGURE 4
The phase diagram of 2D-LMM **(A)**, 2D-LM **(B)**, 2D-LFHCM **(C)** in the $x - y$ plane.

TABLE 2 DNA code table.

Quaternary number	Rule 1	Rule 2	Rule 3	Rule 4	Rule 5	Rule 6	Rule 7	Rule 8
0	A	A	T	T	C	C	G	G
1	C	G	C	G	A	T	A	T
2	G	C	G	C	T	A	T	A
3	T	T	A	A	G	G	C	C

The DNA right shift XOR f_{rX} is expressed as

$$f_{rX} = \begin{bmatrix} R_X\left((A,C,G,T),\ (A)_4\right) \\ R_X\left((A,C,G,T),\ (C)_4\right) \\ R_X\left((A,C,G,T),\ (G)_4\right) \\ R_X\left((A,C,G,T),\ (T)_4\right) \end{bmatrix}$$

$$= \begin{bmatrix} R_X\left((A,C,G,T),0\right) \\ R_X\left((A,C,G,T),1\right) \\ R_X\left((A,C,G,T),2\right) \\ R_X\left((A,C,G,T),3\right) \end{bmatrix} \rightarrow \begin{bmatrix} A & A & A & A \\ T & C & T & C \\ G & G & G & G \\ C & T & C & T \end{bmatrix}$$

4 Proposed image encryption scheme

This section includes a thorough overview of the important parts of the encryption mechanism, such as key creation, chaotic DNA scrambling, and diffusion. Specifically, the scrambling operation exchanges the position and interference of pixels in the ordinary image, minimizing the strong correlation between adjacent pixel values. The pixel data diffusion serves as a critical measure to enhance security. By integrating scrambling and diffusion, both

TABLE 3 DNA operation rules table.

Addition					Subtraction					Exclusive or					Right shift					Left shift				
+	A	C	G	T	–	A	C	G	T	xor	A	C	G	T	A	C	G	T	→	A	C	G	T	←
A	A	C	G	T	A	A	C	T	G	A	A	C	G	T	A	C	G	T	A	A	C	G	T	A
C	C	A	T	G	C	C	A	G	T	C	C	A	T	G	T	A	C	G	C	T	A	C	G	C
G	G	T	C	A	G	G	T	A	C	G	G	T	A	C	G	T	A	C	G	G	T	A	C	G
T	T	G	A	C	T	T	G	C	A	T	T	G	C	A	C	G	T	A	T	C	G	T	A	T

the position and grayscale value of the pixel are simultaneously altered, ensuring that the grayscale information of any pixel is concealed within numerous other pixels.

Let I_1 denote a grayscale image of size $M \times N$, where N represents the number of columns and M is the number of rows. The encryption and decryption process based on the principles outlined in Section 3 is detailed below, and the flow chart of the entire encryption process is shown in Figure 5.

Step 1. Key stream generation.

(i) Let $(x_0^1, y_0^1) = (0.1, 0.1)$, $(x_0^2, y_0^2) = (0.2, 0.2)$, $(x_0^3, y_0^3) = (0.3, 0.3)$, $\lambda = 6$ and $\mu = 1.4$ serve as the initial conditions and control parameters employed for iterating Equation 6.

(ii) After iterating $n_0 + 4MN$ times, three pseudo-random generated sequences $\{y^1_{n_0+4MN}\}$, $\{y^2_{n_0+4MN}\}$, and $\{y^3_{n_0+4MN}\}$ are obtained separately.

(iii) To eliminate transient effects for increased security, the first n_0 or $n_0 + 3MN$ iterations of sequences $\{y^1_{n_0+4MN}\}$, $\{y^2_{n_0+4MN}\}$, and $\{y^3_{n_0+4MN}\}$ (where $n_0 = 800$) are discarded. New sequences y_1, y_2, and y_3, respectively, of length $4MN$, MN, and $4MN$ are obtained.

(iv) The encrypted chaotic sequence s_1 is produced by Equation 9. The y_1 elements are sorted in ascending order, y_{new} is the newly formed sequence after sorting, and s_1 is the index value of y_{new}.

$$[y_{new}, s_1] = sort(y_1), \tag{9}$$

where, the function sort is employed to arrange the data and provide the corresponding index values.

(v) Generation of the encrypted chaotic sequence s_2. By applying the following Equation 10 to compute the obtained pseudo-random sequence y_2, ensuring that the values of y_2 are within $\{0, 1, 2, 3\}$, a new sequence s_2 is obtained.

$$s_2(i) = floor(mod(y_2(i) \times 10^3, 4)), \tag{10}$$

where $i = 1, 2, 3, \ldots, MN$, and $floor(x)$ denotes the function that outputs the largest integer less than x.

(vi) The generation of an encrypted chaotic sequence, referred to as s_3, involves several steps. First, the sequence y_3 is processed using Equation 11 to ensure that the resulting sequence, denoted as y_3^*, only consists of values within the range $\{0, 1, 2, 3\}$. Second, the processed sequence y_3^* undergoes encoding into a DNA sequence following rule 1 presented in Table 2. Finally, Equation 12 is applied to the encoded sequence to obtain the desired chaotic sequence, referred to as s_3. In other words, sequence s_3 is generated sequentially from sequence y_3^*, taking groups of four.

$$y_3^*(i) = mod(floor((y_3(i) + 100)*10^3), 4), \quad i = 1, 2, 3, \ldots, 4MN, \tag{11}$$

$$s_3(j) = y_3^*(4j - 3{:}\ 4j), \quad j = 1, 2, 3, \ldots, MN. \tag{12}$$

Step 2. DNA encoding of the original image.

(i) Let I_1 be a grayscale image with dimensions $M \times N$.

(ii) Reshape the original image I_1 of size $M \times N$ into a $1 \times MN$ vector I_2.

(iii) Encode each pixel value of I_2 into a 4-bit quaternary number, transforming vector I_2 into a quaternary matrix I_3 of size $1 \times 4MN$.

(iv) The DNA image I_4, with a size of $1 \times 4MN$, is produced by encoding each element of the image I_3 into quaternary, which corresponds to the four nucleotides A, C, G, and T depending on rule 1 in Table 2.

Step 3. DNA chaotic confusion and diffusion.

(i) To initiate the initial chaotic confusion, the following Equation 13 is employed to disrupt the positions of I_4.

$$I_5(i) = I_4(s(i)), \ i = 1, 2, 3, \ldots, 4MN. \tag{13}$$

(ii) The sequence I_5 is extracted and grouped consecutively into sets of four. This new sequence is then denoted as I_6, as illustrated in Equation 14.

$$I_6(i) = I_5(4i - 3{:}\ 4i), \ i = 1, 2, 3, \ldots, 4MN. \tag{14}$$

(iii) The implementation of the DNA diffusion operation between the DNA sequence I_6 and the key DNA sequences s_2 and s_3 are conducted using Equation 15.

$$I_7(i) = \begin{cases} R_+(I_6(i), (A)_4) = R_+(I_6(i), 0) \text{ if } s_2(i) = 0; \\ L_-(I_6(i), (C)_4) = L_-(I_6(i), 1) \text{ if } s_2(i) = 1; \\ R_X(I_6(i), (G)_4) = R_X(I_6(i), 2) \text{ if } s_2(i) = 2; \\ L_X(I_6(i), (T)_4) = L_X(I_6(i), 3) \text{ if } s_2(i) = 3, \end{cases} \tag{15}$$

(iv) To further scramble the positions, we employ the method presented in (i), which is Equation 16 in this case, to disrupt

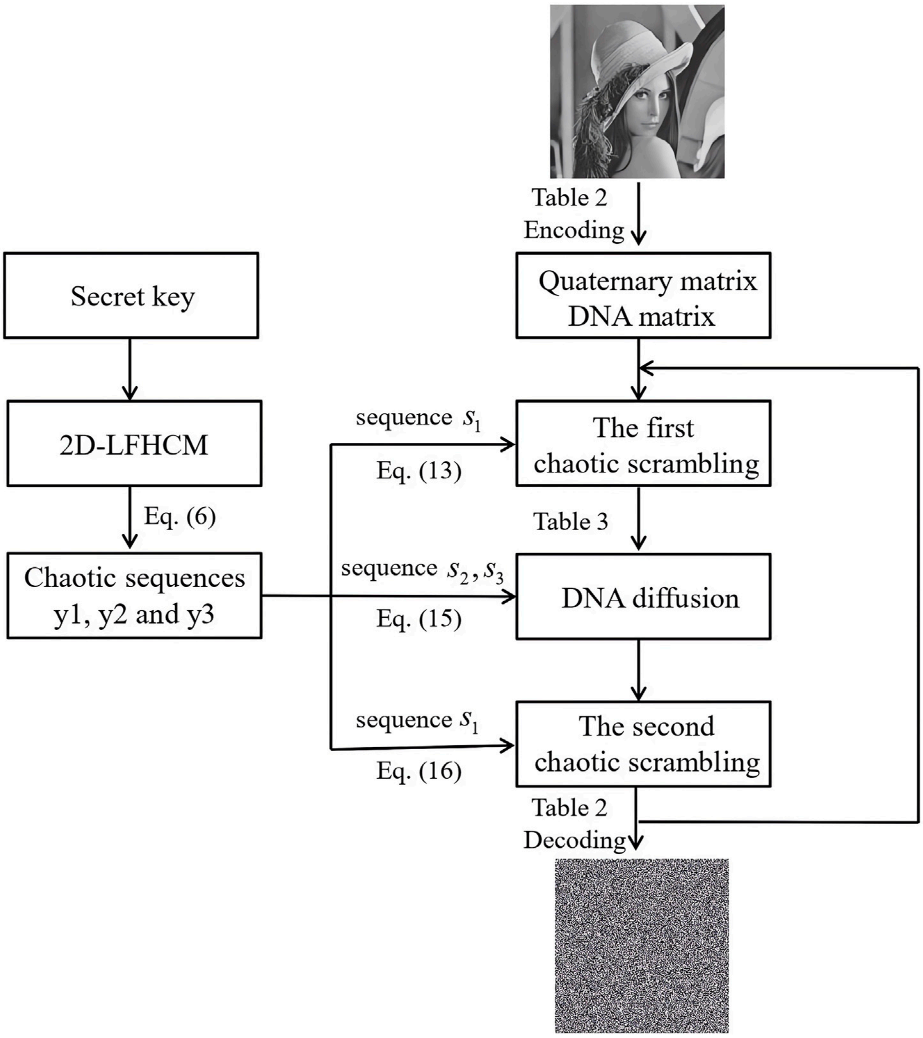


FIGURE 5
Flowchart of the encryption process (where the picture of Lena is sourced from [38]).

the position of I_7, effectively achieving the second chaotic scrambling.

$$I_8(i) = I_7(s(i)),\ i = 1, 2, 3, \ldots, 4MN. \tag{16}$$

(v) Following rule 1 in Table 2, every nucleotide A, C, G, and T in the diffused DNA image I_8 is decoded into a quaternary number, resulting in an encrypted quaternary image I_9 of size $1 \times 4MN$.

(vi) Encoded as integer values in the range of 0–255 for every 4 bits, these values are then transformed into a grayscale cipher image I_{10} with dimensions $1 \times MN$.

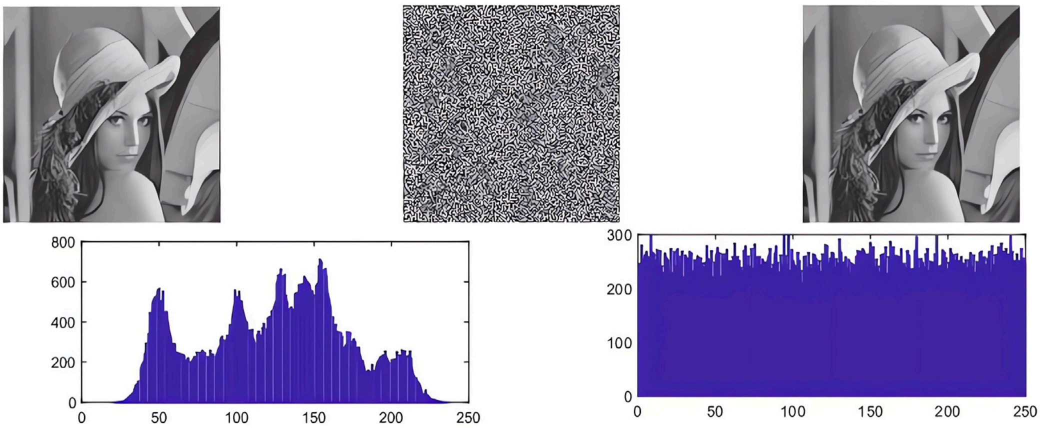
Step 4. Cipher image.

The gray cipher image I_{10}, which is $1 \times MN$ in size, is reshaped into a gray cipher image I_{11} with dimensions $M \times N$.

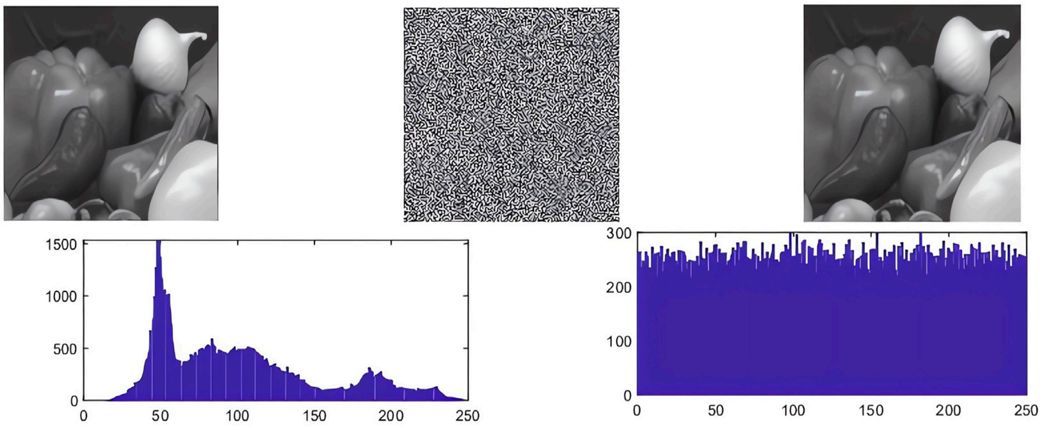
The image decryption process closely mirrors the encryption procedure, involving the sequential inversion of steps utilized in encryption and relying on the application of a cryptographic key. Similarly, if DNA right shift addition is utilized in the encryption phase, it would be reversed in the decryption phase.

5 Performance evaluation

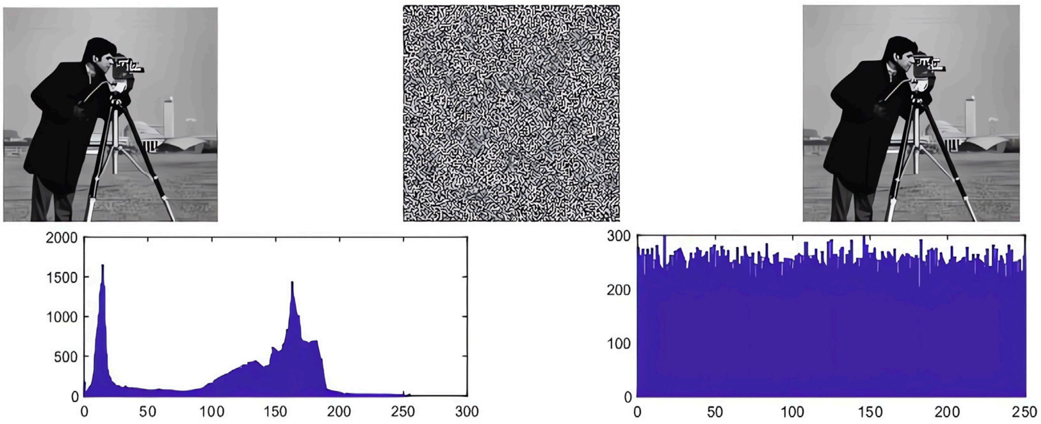
In this section, various images (such as Lena, Onion, and Cameraman) will be utilized to evaluate the performance of the proposed cryptosystem based on image statistical performance and security analysis. All experimental results were calculated using MATLAB 2018b on a compatible computer with Windows 10, 8.00 GB RAM, and Intel (R) Core (TM) i5-7300HQ CPU @ 2.50 GHz. Figure 6 displays the encryption performance of the proposed cryptographic system. Each part begins with a row displaying the plain image, followed by the encrypted image, and concludes with the decrypted image, from left to right. The second line exhibits histograms for both the plain and encrypted images. We have documented all experimental data in a table, which provides evidence of the outstanding capabilities of our cryptographic system in effectively addressing various security and statistical risks.



a. Lena



b. Onion



c. Cameraman

FIGURE 6
The original images, encrypted images, decrypted images, and histograms of the original and encrypted images of Lena **(A)**, Onion **(B),** and Cameraman **(C)**, respectively (where the pictures of Lena, Onion, and Cameraman are sourced from [38], [43], and [27], respectively).

TABLE 4 Comparison of encryption time of different algorithms.

Encryption algorithm	Time cost (units in s)	Encryption speed (units in Kbit/s)
Proposed in this paper	0.4753	1077.2144
Reference [38]	0.4862	1053.0646
Reference [39]	3.6240	141.2804
Reference [40]	0.5683	900.9326

5.1 Key space analysis

The extent of the key space in an image encryption scheme is a pivotal factor in determining its security. The key space encompasses all authorized keys for the scheme. Evidently, an expanded key space augments the scheme's resilience against exhaustive attacks, thereby ensuring an elevated level of security for the encrypted image algorithm. As a general rule, if the key space exceeds $2^{100} \approx 10^{30}$, the encryption mechanism becomes impervious to brute force attempts. In this paper, the encryption scheme's initial key comprises two control parameters, namely, λ and μ, along with two initial values, x_0 and y_0. By adhering to the Institute of Electrical and Electronic Engineers' (IEEE) recommendation of using 64-bit double-precision numbers, the key space for this scheme can amount to $(10^{15})^4 = 10^{60}$. This immense value far surpasses 10^{30}, thereby ensuring that the image encryption scheme presented in this study possesses a suitably extensive key space, affording it robust protection against severe attacks.

5.2 Time cost and speed analysis

A superior encryption scheme should not sacrifice encryption time but instead strive to minimize it while ensuring security. In certain application scenarios, such as image transmission, real-time performance is paramount. This necessitates that encryption algorithms be capable of completing data encryption within a short timeframe to ensure real-time transmission. The average encryption time for the aforementioned grayscale images of size 256×256 were calculated and compared with several established encryption algorithms, including DNA encoding or S-box. The amount of data of a gray image with a size of 256×256 is about 512 Kbit, so the encryption speed can be obtained. All results are presented in Table 4. It can be observed from the table that the proposed solution exhibits the shortest encryption duration, indicating its superior encryption efficiency.

5.3 Histogram analysis

During everyday practical use, there is a potential risk of theft or attack on encrypted images while they are being transmitted. Thus, it becomes crucial to assess both the statistical properties and security of these encrypted images. One of the most basic and intuitive techniques for examining the frequency distribution in plaintext and encrypted images is histogram analysis. Examining the histogram is instrumental in assessing the performance of the encryption algorithm. In case the histogram of the encrypted image exhibits an even or irregular distribution, it indicates that the statistical characteristics have been concealed or destroyed, suggesting that the encryption algorithm might be more efficient. If the histogram of the ciphertext image displays noticeable characteristics or exhibits a notably dissimilar distribution pattern compared to that of the plaintext image, it could indicate potential vulnerabilities in information leakage or the encryption algorithm. Such observations are valuable in identifying encryption issues and enhancing the encryption scheme. In Figure 6, the histograms for various images (Lena, Onion, and Cameraman) can be observed. From an intuitive perspective, it becomes apparent that encrypted images exhibit a uniform histogram, while the histograms of plaintext images vary. If the histogram of encrypted images exhibits an approximately uniform distribution, indicating a lack of discernible regularity in pixel value distribution, it signals the heightened robustness of the encryption scheme against statistical attacks.

5.4 Chi-square analysis

Non-uniformly distributed pixel values can imply that there are some specific features or structures in the image, which may make it easier for the encrypted image to infer some information from the histogram, thereby compromising the encryption's security level. On the contrary, when pixel values are uniformly distributed, potential intruders are prevented from extracting reliable information from the histogram, because the histogram lacks discernible peaks or features, indicating that the image's statistical characteristics are to some extent concealed. Consequently, inferring information about the original image from the histogram becomes challenging.

The χ^2 statistic (one-sided hypothesis test) is frequently employed to quantify the difference between the two in terms of quantity. Chi-square represents a statistical method utilized to measure such differences. If the frequency distribution of a given set of samples is denoted by f_i, $i = 1, 2, \ldots, n$, the theoretical frequency distribution is assumed to be g_i, $i = 1, 2, \ldots, n$. Assumption H_0: The sample comes from the theoretical distribution. When H_0 is assumed to hold Equation 17,

$$\chi^2 = \sum_{i=1}^{n} \frac{(f_i - g_i)^2}{g_i}, \tag{17}$$

is called the Pearson χ^2 statistic and obeys the χ^2 distribution with $n - 1$ degrees of freedom.

Given the image dimensions as $M \times N$, we posit that the pixel frequency f_i associated with each gray value in the histogram

TABLE 5 The entropy values and scores of the original images and encrypted images of Lena, Onion, and Cameraman, respectively.

Images	P/E	Size	Information entropy	Chi-square score
Lena	Plain	256 × 256	7.4508	4.0523×10^4
Lena	Encrypted	256 × 256	7.9997	278.7568
Onion	Plain	256 × 256	7.3426	6.8641×10^4
Onion	Encrypted	256 × 256	7.9971	262.9375
Cameraman	Plain	256 × 256	7.1048	9.8781×10^4
Cameraman	Encrypted	256 × 256	7.9984	264.9377

TABLE 6 Results of the correlation coefficient between original and encrypted images of Lena, Onion, and cameraman in various directions.

Images	Direction	Plain image correlation	Encrypted image correlation
Lena	Horizontal	0.9757	0.0021
Lena	Vertical	0.9552	0.0102
Lena	Main diagonal	0.9229	0.0006
Lena	Secondary diagonal	0.9372	−0.0148
Onion	Horizontal	0.9926	0.0032
Onion	Vertical	0.9934	−0.0179
Onion	Main diagonal	0.9840	0.0056
Onion	Secondary diagonal	0.9892	0.0228
Cameraman	Horizontal	0.9596	0.0339
Cameraman	Vertical	0.9284	0.0003
Cameraman	Main diagonal	0.8921	−0.0164
Cameraman	Secondary diagonal	0.9076	−0.0063

conforms to a uniform distribution. At this time, $g_i = g = MN/256,\ i = 0, 1, 2\ \ldots, 255$, then,

$$\chi^2 = \sum_{i=0}^{255}\frac{(f_i - g_i)^2}{g_i} = \sum_{i=0}^{255}\frac{\left(f_i - \frac{MN}{256}\right)^2}{\frac{MN}{256}} = \frac{1}{256}\sum_{i=0}^{255}\frac{(256 f_i - MN)^2}{MN}. \tag{18}$$

Equation 18 obeys the χ^2 distribution with a degree of freedom of 255. The significance level α is given such that $P\left\{\chi^2 \geqslant \chi^2_\alpha (n-1)\right\} = \alpha$, that is, the null hypothesis H_0 is accepted when $\chi^2 < \chi^2_\alpha (n-1)$. In instances where the level of significance $\alpha = 0.01, \alpha = 0.05$, and $\alpha = 0.1$, the degree of freedom is 255, the χ^2 distribution value $\chi^2_{0.01}(255) = 310.457$, $\chi^2_{0.05}(255) = 293.248$, and $\chi^2_{0.1}(255) = 284.336$.

The generally used significance level is $\alpha = 0.05$. An encrypted image with a chi-square score of $\chi^2_{0.05}(255) = 293.248$ indicates a highly uniform pixel distribution. Table 4 presents the chi-square scores for various encrypted images, namely, Lena, Onion, and Cameraman, demonstrating that the pixel values of our proposed encryption scheme are evenly distributed between 0 and 255 in different rounds of encryption. As a consequence, the ciphertext histogram exhibits an even distribution, suggesting that the image encryption method employed in this study demonstrates increased resilience against statistical attacks. The outcomes of the χ^2 test can be found in the provided Table 5.

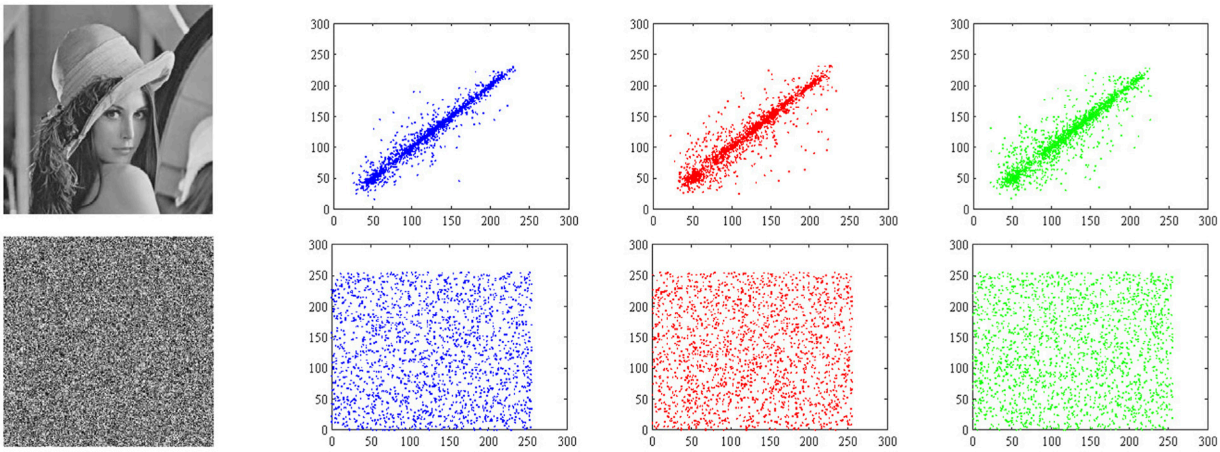
5.5 Information entropy

The unpredictability of image information is reflected in information entropy. It is widely accepted that higher entropy corresponds to increased uncertainty, greater disorder within the information, and reduced visual information. The calculation formula for information entropy can be expressed as Equation 19.

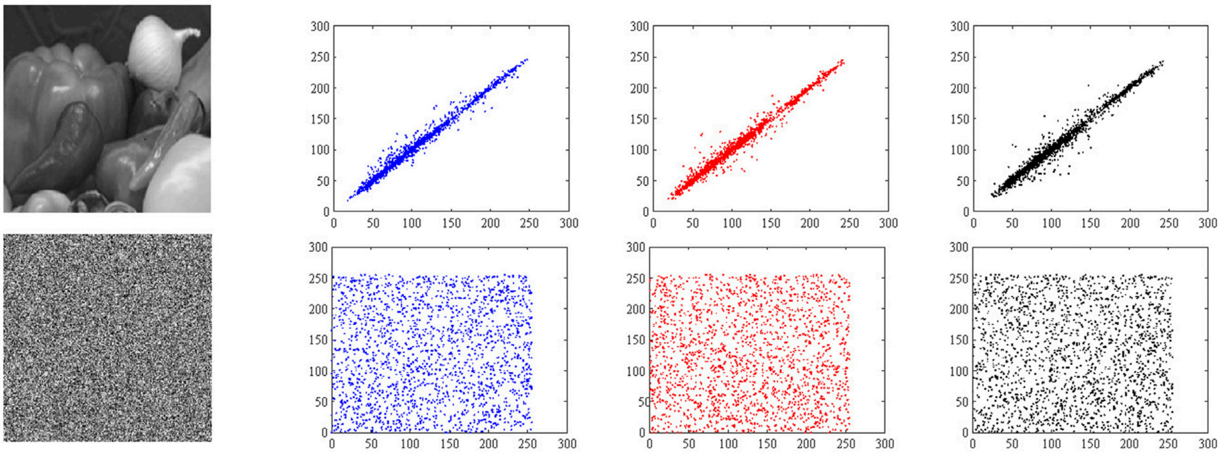
$$H = -\sum_{i=0}^{L} p(i)\log_2 p(i), \tag{19}$$

where, L represents the gray level of the image, and $p(i)$ denotes the probability of gray level i.

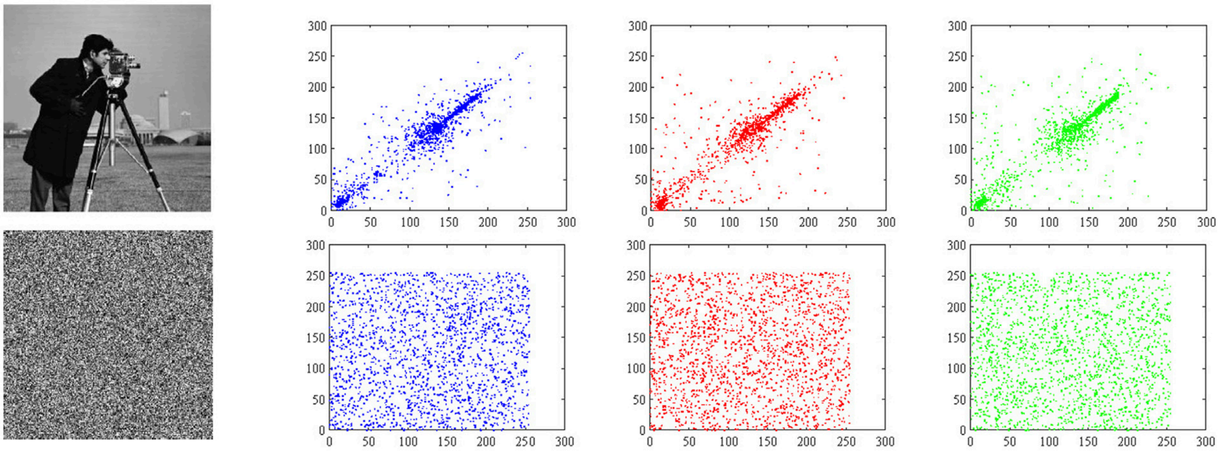
For a randomly generated grayscale image with a gray level of $L = 256$, the theoretical information entropy value H is 8. The information entropy is computed for plain images of Lena, Onion, Cameraman, and their corresponding encrypted



a. Lena



b. Onion



c. Cameraman

FIGURE 7
The first row of the three sets of images Lena **(A)**, Onion **(B)**, and Cameraman **(C)**, from left to right, are the original images and the correlation of adjacent pixels of the original images in the horizontal, vertical, and diagonal direction, respectively. The second row is the same, only for their encrypted images. (The pictures of Lena, Onion, and Cameraman are sourced from [38], [43], and [27], respectively).

versions. The results of the calculations are presented in Table 5, revealing that the information entropies of encrypted images closely approach 8. This suggests that encrypted images exhibit a more advantageous random distribution. Therefore, the encryption method proposed by us exhibits strong resistance to entropy-based attacks.

TABLE 7 Comparison of encryption performance of different algorithms (Lena, and size: 256 × 256).

Algorithm	Adjacent pixel correlation			Information entropy	Time cost	Encryption speed
	Horizontal	Vertical	Diagonal			
Proposed in this paper	0.0021	0.0102	**0.0006**	**7.9997**	**0.4753**	**1077.2144**
Reference [25]	−0.0017	−0.0009	−0.0019	7.9962	0.9170	558.3424
Reference [39]	−0.0031	0.0084	−0.0007	7.9971	—	—
Reference [38]	0.0068	−0.0054	0.0010	7.9967	0.4862	1053.0646
Reference [41]	−0.0036	0.0026	0.0012	7.9995	0.9510	538.3807
Reference [42]	**−0.0006**	−0.0057	0.0009	7.9938	—	—
Reference [43]	0.0013	**0.0002**	0.0033	7.9972	—	—
Reference [44]	0.0105	−0.0023	0.0052	**7.9997**	—	—

Note: Bold font indicates the best result in each column. "-" indicates that the reference did not record this test result for the Lena image.

5.6 Correlation

Evaluating the correlation properties of both the original and encrypted images is essential, complementing the analysis of the image's histogram and information entropy. Neighboring pixels in the horizontal, vertical, main diagonal, and sub-diagonal directions exhibit a strong correlation in the original image. The objective of image encryption algorithms is to minimize the correlation between adjacent pixels in the encrypted image, providing a defense against statistical attacks. A correlation value of zero is ideal. This study randomly samples 2000 pairs of neighboring pixels along the horizontal, vertical, main diagonal, and secondary diagonal directions from both the plain and encrypted images. In this study, 2000 pairs of adjacent pixels are randomly selected from both the original and encrypted images in the horizontal, vertical, main diagonal, and secondary diagonal directions. The correlation coefficient between the two adjacent pixels can be computed by applying Equation 20.

$$\begin{cases} E(x) = \dfrac{1}{N}\sum\limits_{i=1}^{N} x_i; \\ D(x) = \dfrac{1}{N}\sum\limits_{i=1}^{N} (x_i - E(x))^2; \\ cov(x,y) = \dfrac{1}{N}\sum\limits_{i=1}^{N} (x_i - E(x))(y_i - E(y)); \\ r_{xy} = \dfrac{cov(x,y)}{\sqrt{D(x)}\sqrt{D(y)}}, \end{cases} \tag{20}$$

where N is the number of pixel pairs, x and y denote the gray values of two adjacent pixels, $E(x)$ is the mean value, $D(x)$ represents variance, $\text{cov}(x,y)$ stands for covariance, and r_{xy} is correlation coefficient of x and y. The correlation coefficients for both plain and encrypted images of Lena, Onion, and Cameraman in the horizontal, vertical, main diagonal, and secondary diagonal directions are presented in Table 6.

The correlation between adjacent pixels in the original and encrypted images of Lena, Onion, and Cameraman is depicted in Figure 7 for the horizontal, vertical, and main diagonal directions (from left to right). The experimental results indicate a lack of significant correlation between neighboring pixels in the encrypted images, in contrast to the noticeable correlation present in the original images. The efficacy of the encryption system described in this study is highlighted by this conclusion. It's important to note that these results are obtained after only a single round of encryption. If multiple encryptions are performed, the effect may be more significant.

5.7 Comparison and analysis

The algorithm in this paper is used to encrypt and test the performance of image Lena, and compared with other encryption algorithms. The test results of other algorithms are directly quoted from the corresponding papers. The comparison results are shown in Table 7. It can be observed that the performance difference of adjacent pixel correlation analysis of each algorithm is small. In terms of information entropy and other resistance to statistical attacks and encryption speed, the algorithm in this paper has better performance, indicating that the algorithm in this paper has better security.

6 Conclusion

The hybrid image encryption method described in this paper integrates DNA computing theory with the improved 2D-LFHCM. Furthermore, the security, histogram, correlation coefficient, and information entropy aspects of the proposed scheme are examined to demonstrate its rationality. Numerical simulations demonstrate the notable efficacy of the image encryption technique introduced in this study.

A feasible idea for future work is to apply the proposed method to multi-image encryption [24,36], which can improve efficiency while ensuring security. Another possibility is to combine encryption with quantum technology. In light of the advancements in quantum information technology, numerous technologies have been proposed to enhance traditional image encryption algorithms. The exponentially accelerating capabilities of quantum technology, as opposed to traditional computing, are critical for mitigating the vulnerability of encryption algorithms to decipherment. To harness the potential benefits of combining quantum computing with conventional image encryption approaches, Hua Hua et al. [30] came up with dynamic image encryption via quantum walks and chaos-induced DNA to boost image security. Wen Wen and Lin [37] analyzed the security of an existing image encryption algorithm based on quantum chaotic map and DNA coding (QCMDC-IEA), and proposed a low-complexity attack method, which provides some theoretical tips and suggestions for improving the security of the system based on DNA coding and chaotic image encryption. Our upcoming study aims to investigate the potential synergy between quantum walking and the recently proposed DNA computing principles to develop an innovative encryption method. This novel approach is expected to enhance the security measures for image encryption, thus carrying significant implications.

Data availability statement

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Author contributions

YC: Software, Validation, Writing–original draft. TL: Funding acquisition, Supervision, Writing–review and editing. CC: Conceptualization, Formal Analysis, Writing–review and editing. YX: Investigation, Writing–review and editing.

Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. This work was funded by Sichuan Science and Technology Program (No. 2023NSFSC0070), the Opening Project of Sichuan Province University Key Laboratory of Bridge Non-destruction Detecting and the Engineering Computing (No. 2023QYJ06), and Innovation Team Program of Sichuan University of Science and Engineering (No. SUSE652B002).

Acknowledgments

Many thanks to Prof. Simin Yu, Prof. Yong Wang, and Prof. Xiaoyuan Wang for their help in achieving this work.

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. Chen L, Li CQ, Li C. Security measurement of a medical communication scheme based on chaos and dna coding. *J Vis Commun Image Representation* (2022) 83:103424. doi:10.1016/j.jvcir.2021.103424

2. Wang S, Peng Q, Du B. Chaotic color image encryption based on 4d chaotic maps and dna sequence. *Opt Laser Technology* (2022) 148:107753. doi:10.1016/j.optlastec.2021.107753

3. Wen H, Lin Y. Cryptanalyzing an image cipher using multiple chaos and dna operations. *J King Saud University-Computer Inf Sci* (2023) 35:101612. doi:10.1016/j.jksuci.2023.101612

4. 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

5. Huang Z, Cheng S, Gong L, Zhou N. Nonlinear optical multi-image encryption scheme with two-dimensional linear canonical transform. *Opt Lasers Eng* (2020) 124:105821. doi:10.1016/j.optlaseng.2019.105821

6. Wu C, Hu K, Wang Y, Wang J, Wang QH. Scalable asymmetric image encryption based on phase-truncation in cylindrical diffraction domain. *Opt Commun* (2019) 448:26–32. doi:10.1016/j.optcom.2019.05.009

7. Li X, Li C, Lee I-K. Chaotic image encryption using pseudo-random masks and pixel mapping. *Signal Process.* (2016) 125:48–63. doi:10.1016/j.sigpro.2015.11.017

8. Supreeti K, Sukanta D. A study of chaos in non-uniform cellular automata. *Commun Nonlinear Sci Numer Simulation* (2019) 76:116–31. doi:10.1016/j.cnsns.2019.04.020

9. Chai X, Zheng X, Gan Z, Han D, Chen Y. An image encryption algorithm based on chaotic system and compressive sensing. *Signal Process.* (2018) 148:124–44. doi:10.1016/j.sigpro.2018.02.007

10. Gong L, Qiu K, Deng C, Zhou N. An image compression and encryption algorithm based on chaotic system and compressive sensing. *Opt Laser Technology* (2019) 115:257–67. doi:10.1016/j.optlastec.2019.01.039

11. Chen Y, Lu T, Wang Q. The chaotic properties and circuit design of a generalized high-dimensional integer-domain system. *Chaos, Solitons and Fractals* (2024) 181:114610. doi:10.1016/j.chaos.2024.114610

12. Li C, Gao Y, Lei T, Li RYM, Xu Y. Two independent offset controllers in a three-dimensional chaotic system. *Int J Bifurcation Chaos* (2024) 34:2450008. doi:10.1142/s0218127424500081

13. Li Y, Li C, Zhong Q, Liu S, Lei T. A memristive chaotic map with only one bifurcation parameter. *Nonlinear Dyn* (2024) 112:3869–86. doi:10.1007/s11071-023-09204-0

14. Kong X, Yu F, Yao W, Cai S, Zhang J, Lin H. Memristor-induced hyperchaos, multiscroll and extreme multistability in fractional-order hnn: image encryption and fpga implementation. *Neural Networks* (2024) 171:85–103. doi:10.1016/j.neunet.2023.12.008

15. Wang X, Zhang X, Gao M, Iu HHC, Wang C. A color image encryption algorithm based on hash table, halbert curve and hyper-chaotic synchronization. *Mathematics* (2023) 11:567. doi:10.3390/math11030567

16. Yu F, Kong X, Yao W, Zhang J, Cai S, Lin H, et al. Dynamics analysis, synchronization and fpga implementation of multiscroll hopfield neural networks with non-polynomial memristor. *Chaos, Solitons and Fractals* (2024) 179:114440. doi:10.1016/j.chaos.2023.114440

17. Wang MJ, Gu L. Multiple mixed state variable incremental integration for reconstructing extreme multistability in a novel memristive hyperchaotic jerk system with multiple cubic nonlinearity. *Chin Phys B* (2024) 33:020504. doi:10.1088/1674-1056/acddd0

18. Matthews R. On the derivation of a chaotic encryption algorithm. *Cryptologia* (1989) 13:29–42. doi:10.1080/0161-118991863745

19. Alvarez G, Li S. Some basic cryptographic requirements for chaos-based cryptosystems. *Int J Bifurcation Chaos* (2006) 16:2129–51. doi:10.1142/s0218127406015970

20. Wang X, Teng L, Qin X. A novel colour image encryption algorithm based on chaos. *Signal Process.* (2012) 92:1101–8. doi:10.1016/j.sigpro.2011.10.023

21. Wang Q, Yu S, Li C, Lu J, Fang X, Guyeux C, et al. Theoretical design and fpga-based implementation of higher-dimensional digital chaotic systems. *IEEE Trans Circuits Syst Regular Pap* (2016) 63:401–12. doi:10.1109/tcsi.2016.2515398

22. Huang L, Wang S, Xiang J. A tweak-cube color image encryption scheme jointly manipulated by chaos and hyper-chaos. *Appl Sci* (2019) 9:4854. doi:10.3390/app9224854

23. Wang X, Su Y, Zhang H, Zou C. A new hybrid image encryption algorithm based on gray code transformation and snake-like diffusion. *Vis Computer* (2021) 38:3831–52. doi:10.1007/s00371-021-02224-0

24. Gao X, Mou J, Xiong L, Sha Y, Yan H, Gao Y. A fast and efficient multiple images encryption based on single-channel encryption and chaotic system. *Nonlinear Dyn* (2022) 108:613–36. doi:10.1007/s11071-021-07192-7

25. Alexan W, Elkandoz M, Mashaly M, Azab E, Aboshousha A. Color image encryption through chaos and KAA map. *IEEE Access* (2023) 11:11541–54. doi:10.1109/access.2023.3242311

26. 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

27. Wu J, Liao X, Yang B. Image encryption using 2d henon-sine map and dna approach. *Signal Process.* (2018) 153:11–23. doi:10.1016/j.sigpro.2018.06.008

28. Patel S, Bharath K-P, Kumar R. Symmetric keys image encryption and decryption using 3d chaotic maps with dna encoding technique. *Multimedia Tools Appl* (2020) 79:31739–57. doi:10.1016/j.cnsns.2019.04.020

29. Liu Q, Liu L. Color image encryption algorithm based on dna coding and double chaos system. *IEEE Access* (2020) 8:83596–610. doi:10.1109/access.2020.2991420

30. Hua N, Liu H, Xiong X, Wang JL, Liang JQ. A dynamic image encryption scheme based on quantum walk and chaos-induced dna. *Quan Eng* (2023) 2023:1–15. doi:10.1155/2023/3431107

31. Wu Y, Yang G, Jin H, Noonan JP. Image encryption using the two-dimensional logistic chaotic map. *J Electron Imaging* (2012) 21:013014. doi:10.1117/1.jei.21.1.013014

32. Ye X, Wang X. Design a novel image encryption algorithm based on a prng with high performance. *IEEE MultiMedia* (2022) 99:1–11. doi:10.1109/mmul.2022.3232180

33. Lu J, Wu X, Lv J, Kang L. A new discrete chaotic system with rational fraction and its dynamical behaviors. *Chaos, Solitons and Fractals* (2004) 22:311–9. doi:10.1016/j.chaos.2004.01.010

34. Watson JD, Crick FHC. Molecular structure of nucleic acids: a structure for deoxyribose nucleic acid. *Nature* (1953) 171:737–8. doi:10.1038/171737a0

35. Lone PN, Singh D, Mir UH. Image encryption using dna coding and three-dimensional chaotic systems. *Multimedia Tools Appl* (2022) 81:5669–93. doi:10.1007/s11042-021-11802-2

36. Chen X, Mou J, Cao Y, Banerjee S. Chaotic multiple-image encryption algorithm based on block scrambling and dynamic dna coding. *Int J Bifurcation Chaos* (2023) 33:2350190. doi:10.1142/s0218127423501900

37. Wen H, Lin Y. Cryptanalysis of an image encryption algorithm using quantum chaotic map and dna coding. *Expert Syst Appl* (2024) 237:121514. doi:10.1016/j.eswa.2023.121514

38. Sun S. A novel hyperchaotic image encryption scheme based on dna encoding, pixel-level scrambling and bit-level scrambling. *IEEE Photon J* (2018) 10:1–14. doi:10.1109/jphot.2018.2817550

39. Xu L, Li Z, Li J, Hua W. A novel bit-level image encryption algorithm based on chaotic maps. *Opt Lasers Eng* (2016) 78:17–25. doi:10.1016/j.optlaseng.2015.09.007

40. Brindha M, Gounden NA. A chaos based image encryption and lossless compression algorithm using hash table and Chinese remainder theorem. *Appl Soft Comput* (2016) 40:379–90. doi:10.1016/j.cnsns.2019.04.020

41. Zarebnia M, Pakmanesh H, Parvaz R. A fast multiple-image encryption algorithm based on hybrid chaotic systems for gray scale images. *Optik* (2019) 179:761–73. doi:10.1016/j.ijleo.2018.10.025

42. Kaur G, Agarwal R, Patidar V. Chaos based multiple order optical transform for 2d image encryption. *Eng Sci Technol Int J* (2020) 23:998–1014. doi:10.1016/j.jestch.2020.02.007

43. He P, Sun K, Zhu C. A novel image encryption algorithm based on the delayed maps and permutation-confusion-diffusion architecture. *Security Commun Networks* (2021) 2021:1–16. doi:10.1155/2021/6679288

44. Kumar K, Roy S, Rawat U, Malhotra S. Iehc: an efficient image encryption technique using hybrid chaotic map. *Chaos, Solitons and Fractals* (2022) 158:111994. doi:10.1016/j.chaos.2022.111994

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 22 January 2025
DOI 10.3389/fphy.2024.1508465

Self-organization of the stock exchange to the edge of a phase transition: empirical and theoretical studies

Andrey Dmitriev[1,2]*, Andrey Lebedev[1], Vasily Kornilov[3] and Victor Dmitriev[1]

[1]Big Data and Information Retrieval School, HSE University, Moscow, Russia, [2]Cybersecurity Research Center, University of Bernardo O'Higgins, Santiago, Chile, [3]Graduate School of Business, HSE University, Moscow, Russia

Check for updates

OPEN ACCESS

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

REVIEWED BY
Yousef Azizi,
Independent Researcher, Zanjan, Iran
Andrei Khrennikov,
Linnaeus University, Sweden

*CORRESPONDENCE
Andrey Dmitriev,
a.dmitriev@hse.ru

RECEIVED 09 October 2024
ACCEPTED 24 December 2024
PUBLISHED 22 January 2025

CITATION
Dmitriev A, Lebedev A, Kornilov V and Dmitriev V (2025) Self-organization of the stock exchange to the edge of a phase transition: empirical and theoretical studies. *Front. Phys.* 12:1508465.
doi: 10.3389/fphy.2024.1508465

Our study is based on the hypothesis that stock exchanges, being nonlinear, open and dissipative systems, are capable of self-organization to the edge of a phase transition. To empirically support the hypothesis, we find segments in hourly stock volume series for 3,000 stocks of publicly traded companies, corresponding to the time of stock exchange's stay to the edge of a phase transition. We provide a theoretical justification of the hypothesis and present a phenomenological model of stock exchange self-organization to the edge of the first-order phase transition and to the edge of the second-order phase transition. In the model, the controlling parameter is entropy as a measure of uncertainty of information about a share of a public company, guided by which stock exchange players make a decision to buy/sell it. The order parameter is determined by the number of buy/sell transactions by stock exchange players of a public company's shares, i.e., stock's volume. By applying statistical tests and the AUC metric, we found the most effective early warning measures from the set of investigated critical deceleration measures, multifractal measures and reconstructed phase space measures. The practical significance of our study is determined by the possibility of early warning of self-organization of stock exchanges to the edge of a phase transition and can be extended with high frequency data in the future research.

KEYWORDS

phase transition, self-organized criticality, early warning signals, sandpile cellular automata, stock exchange, econophysical modeling, trading

1 Introduction

More than 35 years ago, P. Bak together with C. Tang suggested that in nonlinear systems far from equilibrium, complex holistic properties may emerge through their self-organization into a critical state [1]. Subsequently, the theory of self-organized criticality (SOC) was formed, the main provisions of which have found application in sociology, biological evolution, seismology, economics and other sciences (e.g., see the papers [2–7, 7–9]). The theory of self-organization at the edge of phase transitions has found applications in cognitive and social science (e.g., see the papers [10, 11]).

The basic model of SOC theory is the sandpile cellular automaton (SCA), which demonstrates how complex holistic properties emerge in a model system with simple rules as a result of self-organization of the automaton into a critical state (e.g., see the papers [12, 13]). The simplest model of SCA is the following model. Suppose that the nodes of the lattice graph are assigned integer numbers (the number of grains of sand in the cells). Then we increase by one the numbers assigned to randomly chosen nodes of the graph (add one grain of sand in the cells). If the number (grains of sand), $z_{k,l}$, for some node (k,l) exceeds some threshold value, z_c, for instance $z_c = 4$, then this node is unstable and its toppling occurs. As a result of node toppling (k,l) numbers for neighbouring nodes, $z_{k\pm1,l\pm1}$, are increased by 1, i.e., $z_{k\pm1,l\pm1} \rightarrow z_{k\pm1,l\pm1} + 4$. Thus $z_{k,l} \rightarrow z_{k,l} - 4$. Collapses occur until the SCA becomes stable, that is, until at each node $z_{k,l} < 4$.

Each iteration of the SCA simulation is followed by its perturbation, by adding one grain of sand to randomly selected cells at a time, and relaxation, by collapsing unstable cells. Starting from some critical iteration, i_c, a single added grain of sand in a randomly selected cell can cause an avalanche of collapses of any size, continuing until all cells regain stability. In the subcritical phase $(i < i_c)$ avalanches rapidly decay in time and space.

In the context of mean-field theory of phase transitions, the control parameter of the SCA is determined by the ratio of the number of particles in the cells to the total number of cells of the SCA, the order parameter is determined by the ratio of the number of unstable cells to the total number of cells of the SCA (e.g., see the paper [14]). The transition of the SCA from the subcritical phase to the critical state corresponding to the critical value of the control parameter occurs as a result of self-organization of the SCA and does not require precise adjustment of the control parameter to the critical value. This is a fundamental difference between self-organization into a critical state and a classical phase transition of the first or second kind, for which precise tuning of the control parameters to critical values is required.

Our study is based on the hypothesis that stock exchanges, being nonlinear, open and dissipative systems, are able to self-organize into a critical state. The theoretical justification of the hypothesis and a phenomenological model of stock exchange self-organization into a critical state are presented in Subsection 3.2. This econophysical model is based on the isomorphism of the SCA model and the stock exchange in the context of systems theory. In the model, the control parameter is defined by entropy as a measure of uncertainty of information about a stock of some public company, based on which the stock exchange traders make a decision to buy/sell it. The order parameter is determined by the number of buy/sell transactions by stock exchange traders of shares of some public company, i.e., stock's volume.

To quantitatively substantiate the hypothesis, we determined time intervals corresponding to the time of the stock exchange's stay in a critical state, Δt_c. The main signs of the system being in a self-organized critical state (in the interval Δt_c or Δi_c for the SCA) are $\rho(1) = 1$, $S(f) = f^{-\beta}$ where $1 \leq \beta \leq 2$ and $p(\xi) = \xi^{-2}$ (e.g., see the paper [13]). Here $\rho(1)$ is the autocorrelation at lag-1, $S(f)$ is the power spectral density, $p(\xi)$ is the probability density function for the values of the dynamic series ξ in the interval Δt_c, corresponding to the order parameter of the system. The identification of Δt_c from the values β requires a significant computational cost in estimating $S(f)$. Recall that we investigated hourly stock volume series for more than 2,600 stocks of publicly traded companies. In addition, the estimation of $p(\xi)$ is obtained only in the intermediate asymptotic region, which is bounded due to the finiteness of the size (number of stock exchange traders and the links between them) of the stock exchange. Therefore, to identify Δt_c in stock volume series, we used the features of 100-hour moving average (MA100) behavior in the vicinity of t_c followed by verification using critical deceleration, multifractal and chaotic measures. Features of MA100 behavior for test series (series of unstable nodes of the SCA) in the vicinity of t_c are presented in Subsection 2.1. Peculiarities of MA100 behavior for stock volume series in the vicinity of t_c and detected Δt_c for stock exchanges and their features are presented in Subsection 3.1.

The practical significance of our study is determined by the possibility of early warning of self-organization of stock exchanges into a critical state (e.g., see the papers [15, 16]). We identified the most effective early warning measures from a wide range of investigated early warning measures (the simplest critical slowing down measures, multifractal measures and chaotic measures). The methods for computing the measures and extracting the most effective early warning measures are presented in Subsection 2.3. The results obtained and their discussion are presented in Subsection 3.3. The detection of a precursor to such self-organization gives investors a reason to pay attention to a stock that is likely to have a large trading volume expected after some time (early warning time). To the stock exchange trading regulator, precursors provide a tool to distinguish between normal market behavior and large one-off manipulations in investigations. We investigated the effectiveness of a wide range of early warning measures: simple critical slowing down measures, multifractal measures and chaotic measures.

The main conclusions, as well as the possibilities and limitations of the empirical results obtained and the proposed model are presented in Conclusion.

Existing studies on the empirical validation of stock market self-organization into a critical state are limited to the analysis of daily world stock indices (e.g., see the papers [17–23]) or daily stock prices of public company shares (e.g., see the papers [24–29]). Studies of financial series with daily intervals allow us to identify time intervals of the critical state only in the case of slow self-organization of the stock exchange into a critical state, when the time interval corresponds to several days. We used a 1 hour interval series, which enabled us to identify a large number of time intervals of several hours corresponding to stock exchange critical states, as well as intervals of several days. We also analyzed stock exchange samples of larger size (dynamic series at 1 hour intervals for stocks of more than 2,600 public companies) and used a larger number of early warning measures. Accordingly, the results we obtain are more reliable and representative than those obtained earlier. In addition, we provide a theoretical justification of the critical behavior of stock exchanges within the framework of the proposed model of self-organization into a critical state with an order parameter corresponding to the number of exchange transactions on shares of a public company.

2 Data set and methods

2.1 Model time series generated by sandpile cellular automata

As test dynamic series, that is, series to determine the required number of iterations in moving average and moving variance in the effective detection of critical iteration, i_c, we used the series of the number of unstable nodes ($i \in [0, n], n \in N$) of the SCA on the Chung-Lu graph with two-parameter degree distribution of graph nodes' degrees (e.g., see the paper [30]) and Manna rule (e.g., see the paper [31]). Series ξ_i demonstrate the exact value i_c, $\rho(1) = 1$, $S(f) = f^{-\beta}$ $(1 \leq \beta \leq 2)$ and $p(\xi) = \xi^{-2}$ in the critical state (at $i > i_c$), which is one of the reasons for their use as test series. The rationale for the choice of the specified graph topology and rule in the context of stock exchanges is presented in Subsection 3.2.

There are two main reasons why we examined the sandpile model and the time series that the model generates. First, on the sandpile model we managed to find out under which conditions we can talk about similarity in critical transitions between model and real financial data, which will be discussed in more detail in Subsection 2.2. Secondly, we used the sandpile model as a model of the stock exchange, which allowed us to theoretically justify the possibility of self-organization of the exchange at the edge of a phase transition (see Subsection 3.2).

Let $z_{k,l}$ be the number of particles (grains of sand) in the node (k, l) of the Chung-Lu graph, z_c be the critical number of grains. If $z_{k,l} \geq z_c$, the node (k, l) is unstable. In general, the self-organization of the SCA into a critical state is determined by perturbation (pumping) and relaxation of the automaton. At the beginning of iteration 0, a perturbation of the automaton takes place in the form of randomly pouring grains of sand into its randomly chosen nodes. If some nodes have $z_{k,l} \geq z_c$, they are considered unstable and their collapse occurs with sand grains moving to neighboring nodes until all nodes are stable ($z_{k,l} < z_c$). In this way the automata are relaxed. The next iteration 1, as well as the iterations following it, also start with perturbation and end with relaxation.

The feature of the Manna rule that distinguishes it from other rules is that each unstable vertex transmits to neighboring (connected) vertices a random number of particles that is equal to the total number of edges of that vertex.

Starting from iteration i_c the SCA self-organizes into a critical state. At that, the dynamical series ξ_i ($i > i_c$) is characterised by the above-mentioned power laws for $\rho(1)$, $S(f)$ and $p(\xi)$.

The considered scenario of self-organization of the automaton to the critical state corresponds to its self-organization to the edge of the second-order phase transition. For self-organization of the automaton to the edge of the first-order phase transition, it is enough to consider in the Manna rule that the collapse of an unstable node (k, l) occurs not only at $z_{k,l} \geq z_c$, but also in the case of transferring to node (k, l) more than one grain of sand from neighbouring nodes (e.g., see the paper [32]).

2.2 Stock volume series and time intervals for critical state

As the source of the real data, we elected to utilize hourly volumes of stock trading for the assets comprising the Russell 3,000 index (exclusive of pre- and post-market data, given their markedly lower liquidity levels), for the preceding 2 years, with the exclusion of companies experiencing data unavailability. This resulted in 2,667 time series, each comprising 3,498 observations. We elected to utilize volumes as they are more conducive to the viability assessment of the model, given that these series are more proximate to the theoretical ones and exhibit a paucity of trends in the data. As an alternative data frequency, 1-minute and 30-minute data were considered. However, both data sets exhibited an issue of mass automatic trade executions close to the astronomical hour end, resulting in a large number of singular spikes. It is possible to mitigate the impact of these automatic spikes to some extent by providing researchers with direct access to the market bids data, rather than statistical aggregates. However, in this case, we were constrained to working with the final time series.

In order to define critical transitions for systems it is necessary to create additional rules that define the criteria for such transitions. The primary criterion is that the moving average of the time series (MA100) increases by 20% in comparison to the volumes of the preceding five iterations. The secondary criterion is that this regime change persists for a minimum of 10 iterations following the transition. It should be noted that the logic described may require modification for systems exhibiting significantly different characteristics. However, in the base case scenario, it should remain equally effective.

The rationale behind the selection of these parameters is as follows:

- MA100 – modification of the first moment of the distribution, which is a well-established early warning measure. Furthermore, 100 iterations were chosen as a highly stringent threshold, enabling the removal of outliers in the data set.
- A 20% increase was selected as it defines the severity of the shift and was chosen based on the simulations with sandpile automaton with Manna rules on the Chung-Lu random graph in comparison to white noise and random walk. The 20% level was deemed appropriate for filtering jumps that occurred in the random time series, while also enabling the identification of transitions from the time series generated by complex systems.
- A comparison to the five iterations preceding the current iteration allows for the filtration of trends and the isolation of actual transitions from the data set.
- A minimum of ten iterations following the transition permits the filtration of sudden outliers that do not result in short- or mid-term changes to the system.

In order to filter time series for modelling purposes, we have elected to employ a further criterion, namely, that there must be a minimum of 800 iterations prior to the critical transition (e.g., see the paper [26]), without the occurrence of other transitions. This threshold was selected on the basis that the majority of early warning measures necessitate the availability of sufficiently wide windows in order to function effectively, without the introduction of artefacts. In this particular case, the initial 500 iterations will be utilized for this purpose, with the remaining 300 employed for prediction purposes, given that all relevant metrics have been duly calculated.

2.3 Early warning measures

In Subsection 2.3 we present a brief description of methods for computing early warning measures (EWMs) for the self-organization into a critical state. The analysis of the behavior of EWMs as the system approaches t_c makes it possible to detect early warning signals for the self-organization of the stock exchange into a critical state. We also introduce the notion of effectiveness of EWMs, using which we determine the most effective EWMs.

Let $\left\{t = \overline{0, n}, n \in \mathbb{N}\right\}$ be the dynamic series for the number of unstable vertices of the SCA on the Chung-Lu graph and Manna rule (see Subsection 2.1), $\left\{t = \overline{t_0, t_f}\right\}$ be the stock volume series with step Δt equal to 1 h. We obtained the dynamic series for EWMs, $\left\{t = \overline{0, n - w_0}\right\}$ for the series ψ_t and $\left\{t = \overline{t_0, t_f - w_0}\right\}$ for the series v_t, computing the measures in a sliding window of width $w_0 = 500$ iterations for the series ψ_t and $w_0 = 500$ hours for the series v_t. For example, for the series v_t, we obtain a sequence of values of some measure m, $m_{t_0}, m_{t_1}, m_{t_2}, \ldots, m_{t_f - w_0}$, the terms of which are calculated in the segments of the series v_t, $[t_0, t_0 + w_0], [t_1, t_1 + w_0], [t_2, t_2 + w_0], \ldots, \left[t_f - w_0, t_f\right]$.

We investigated the behavior of EWMs directly related to the critical slowing down of the system (SCA and stock exchange) as it approaches t_c (e.g., see the paper [33]), as well as multifractal EWMs (e.g., see the papers [25, 34]) and EWMs based on the reconstruction of the phase space of the dynamical system (e.g., see the papers [35, 36]).

2.3.1 Measures of critical slowing down

Computationally, the simplest measures of critical deceleration are variance, σ^2, and autocorrelation at lag-1, ρ, whose series show a sharp increase as the system approaches t_c followed by saturation in the time interval Δt_c, as well as kurtosis, κ, and skewness, γ, whose series are characterised by a sharp switch from increasing to decreasing in the vicinity of t_c. Moreover, the series ρ_t takes values close to 1 in the interval Δt_c.

The power-law scaling exponent, β, of the power spectral density and generalized Hurst exponent, h, are also EWMs, whose significant increase as the system approaches t_c, is an early warning signal of its critical slowing down (e.g., see the papers [22, 33]). Also, the series β_t and h_t, tend to take nearly constant values in the interval Δt_c. In particular, it is shown that $1 \le \beta \le 2$ for $t \in \Delta t_c$ (e.g., see the paper [36]). We computed the β values in all sliding windows by the Welch's method (e.g., see the paper [37]). For each window, the ψ_t and v_t series were segmented using the longest and most overlapping segments, followed by estimating the power spectral density, $S(f)$, for each segment and averaging these estimates. Next, the exponent β for the power law $S(f) = f^{-\beta}$ was calculated. To estimate h we used detrended fluctuation analysis (e.g., see the paper [38]), which gives the most reliable estimate of Hurst exponent for nonstationary series. For the dynamic series under study, e.g., v_t, in each ith sliding window, the profile $V(k) = \sum\limits_{t=t_i}^{t_i+w_0} \left(v_t - \langle v \rangle\right)$ was calculated. Hereinafter, the symbol $\langle \bullet \rangle$ denotes the mean value of some quantity. Next, segmentation of the profile $V(k)$ into non-overlapping segments of length n and determination of the linear trend, $V_n(k)$, for each segment was performed. For different n, the standard deviation of $V(k)$ fluctuations relative to $V_n(k)$, $F(n) = \sqrt{(1/n)\sum_{t=t_i}^{t_i+w_0}\left[V(k) - V_n(k)\right]^2}$, followed by estimation of the exponent h for the power law $F(n) = n^h$.

2.3.2 Multifractal measures

The specific features of the behavior of multifractal EWMs as the system approaches t_c are probably also related to the critical slowing down of the system (e.g., see the paper [34]), but there is no theoretical justification of this connection yet. Full information on the multifractal properties of the dynamical series is given by the multifractal spectrum, $D(h)$, as a dependence of the fractal dimension, D, on the values of Holder exponents, h. The spectrum $D(h)$ cannot be used as an EWM, calculated in a sliding window, but its three main parameters characterising the geometry of the $D(h)$ dependence can be used. Such parameters are the position of the spectrum maximum, h_0, the width of the spectrum, $W = h_{max} - h_{min}$, and the slope of the spectrum, $S = (h_{max} - h_0)/(h_0 - h_{min})$. As the system approaches the edge of the phase transition of the second kind, an increase in h_0, W and S (see the paper [36]).

To calculate the parameters of the multifractal spectrum, we used the wavelet leader method and $D(h) = qh(q) - \tau(q)$, where $\tau(q)$ is the scaling exponents of the structure function $Z(q,s)$ (e.g., see the paper [39]). Following the algorithm of the method, $Z(q,s)$ is represented in the Equation 1 as the sum of qth powers of the largest coefficients, or leaders, of the discrete wavelet transform of the dynamic series v_t, corresponding to the scale s:

$$Z(q,s) = \frac{1}{n_s}\sum_{k=1}^{n_s} L(s,k)^q, \tag{1}$$

where $L(s,k) = |d(s,k)|$ the leaders of wavelet coefficients $d(s,k)$ of scale 2^s and time shift k, $3\lambda_{s,k} = [(k-1)\,2^s, k2^s) \bigcup [k2^s, (k+1)2^s) \bigcup [(k+1)\,2^s, (k+2)2^s)$ is the time neighborhood. If the series v_t is a multifractal series, then the scaling relation $Z(q,s) \simeq s^{\tau(q)}$ is satisfied at all scales s. Decomposing the function $\tau(q)$ into a $\sum\left(C_l q^l\right)/l!$ series allows us to compute the first log-cumulant (C_1), which corresponds to h_0, the second log-cumulant (C_2), which corresponds to W, and the third log-cumulant (C_3), which corresponds to S. Therefore, we used the first three log-cumulants as multifractal EWMs.

2.3.3 Measures of reconstructed phase space

As EWMs, for the calculation of which requires the reconstruction of the phase space of the dynamical system, we used the correlation dimension of the phase space, D_c, and the largest Lyapunov exponent, λ. The dimension of D_c is an estimate of the fractal dimension of the reconstructed attractor of the dynamical system, which increases as the system approaches t_c (e.g., see the paper [36]). The exponent λ, being a measure of the chaotic nature of the dynamical system, increases, taking positive values, as the system approaches t_c (e.g., see the paper [40]).

We used the Takens theorem (see the paper [41]) to reconstruct the phase space of the stock volume series, $P = (P_1, P_2, \ldots, P_M) \in R^M$, over the stock volume series from a sliding window of width w_0, $V = \left(v_1, v_2, \ldots, v_{w_0}\right)$. The phase space P was reconstructed from the series V, using as missing coordinates the l-th state vector, P_l, the series V, taken with some lag τ:

$$P_l = \left(v_l, v_{l+\tau}, \ldots, v_{l+(M-1)\tau}\right), \tag{2}$$

where τ is the time delay, M is the embedding dimension, $l = 1, 2, \ldots, w_0 - (M-1)\tau$. Takens' theorem does not answer the question of how to calculate the value τ and M.

The time τ for the Equation 2 was chosen so that the correlation between v_l and $v_{l+\tau}$ was minimal. The delay τ was chosen equal to the time of the first zero crossing of the autocorrelation function $(w_0 - \tau)^{-1}\sum_{k=1}^{w_0-\tau}(v_k - \langle v\rangle)(v_{k+\tau} - \langle v\rangle)$ (e.g., see the paper [42]).

To estimate the values of M and D_c we calculated the correlation sum (e.g., see the paper [42]):

$$C(\varepsilon) = \frac{1}{p(p-1)}\sum_{i=0}^{p-2}\sum_{j=i+1}^{p-1}\theta\left(\varepsilon - \left|P_i - P_j\right|\right), \tag{3}$$

where $p = w_0 - (M-1)\tau$, $\theta = \begin{cases} 1, \varepsilon - \left|P_i - P_j\right| \geq 0 \\ 0, \varepsilon - \left|P_i - P_j\right| < 0 \end{cases}$. The sum $C(\varepsilon)$ from the Equation 3 was calculated for different values of distances, ε, between vectors P_i and P_j of the reconstructed phase space. This procedure was repeated for several dimensions M. The criterion for stopping the procedure is the fulfillment of the power law $C(\varepsilon) \simeq \varepsilon^{D_c}$. As the value of M increases, the correlation dimension increases. At some M, the value of D_c comes to a constant level. The estimate of the dimensionality of D_c is the tangent of the slope of the straight line approximating the correlation sum $C(\varepsilon)$ in a double logarithmic scale. At the same time, only linear parts of the dependence were investigated.

There exists a spectrum of Lyapunov exponents characterizing the separation rate of infinitely close phase space trajectories (e.g., see the paper [43]). The largest Lyapunov exponent, λ, defines the notion of predictability of the dynamical system. Let $\delta(0)$ be the minimum value of the distances between the vectors of the reconstructed phase space, i.e., $\delta(0) = \left|P_i - P_j\right|$. The distance between vectors after time t is $\delta(t) = \exp(\lambda t)$. The linear regression for λt is an estimate of the largest Lyapunov exponent. Regardless of the dimensionality of the phase space, this procedure was repeated for several dimensions to ensure that λ does not depend on the dimensionality of the space.

Previously (see the paper [44]), we introduced the notion of EWM, defined in terms of the number of false early warning signals, v, for the zero-mean dynamic series of EWM increments, Δm_t, and the early warning time, $\Delta\tau_{EW}$, for the series m_t. For example, EWM1 is more effective than EWM2, if $v_1 < v_2$ and $\Delta\tau_{EW1} > \Delta\tau_{EW2}$. In the context of the presented study, this measure was modified to the AUC (area under curve for all of the combinations of false positive rate and true positive rate for all possible thresholds of separation between predicted classes) as a more stable measure in case of problems with class balance in the sample.

3 Results and their discussion

3.1 Time intervals for critical state of stock exchange

Following the implementation of all filters mentioned in Subsection 2.2, a total of 967 time series were identified as exhibiting critical transitions in accordance with the predefined criteria. For all of the aforementioned time series, metrics were calculated in accordance with the specifications outlined in Subsection 2.3. Additionally, the 8-hour dynamics and variance of these instruments were calculated (as daily trading sessions on the US stock exchanges last for 8 h), which further reduced the sample size. However, the resulting observations still numbered nearly 281.4 thousand. Subsequently, observations in the time series are divided into two categories: those that are close to a critical transition and those that are not. Eight distinct closeness horizons (H) were considered, ranging from 1 to 8 iterations. This allowed for the classification of observations as either predicting a critical transition in not more than H iterations, or otherwise. Given the imbalanced nature of the dataset, we opted to down sample it via bootstrapping (see the book [45]), with positive observation shares of 5%, 10%, 15% and 20% and 500 random separations for each of the H-share combinations, in order to demonstrate the stability of the random sampling and modelling results.

In order to predict the probability of an iteration belonging to the "close to the critical transition" group, the probit model has been selected (see the paper [46]). The simplicity and high interpretability of the model would facilitate the straightforward observation of the efficiency of the measures and their derivatives. Two sets of models were constructed: one using all variables, and another with only one variable at a time. This was done to ascertain whether there were any differences in the final impact on quality prediction. In the first set of models, the importance of each variable was calculated as a share of those where the p-value of the coefficient was less than 5%. In the second set, the metric was the largest time horizon that would still achieve an AUC higher than 0.75. In addition to the AUC, two sample Kolmogorov-Smirnov (KS) tests (see the paper [47]) were employed to measure the capacity of our models to effectively differentiate between positive and negative observations.

Table 1 shows us that all of the variables (white–no statistically significant impact on the quality of the prediction, yellow–significant in some of the modifications of the variable, green–significant in most of the modifications) except for the Hurst exponent, correlation dimension and the second cumulant of wavelet leader can be at least partially useful for the task of critical transition prediction, which mostly follows previous research on this topic and tells us that at least for the financial data classification models can be applied with high level accuracy and interpretability.

3.2 Phenomenological model of stock exchange self-organization into a critical state

As shown in Subsection 3.1, a stock exchange self-organizes into a critical state and stays in this state for a certain number of hours, determined by the share of a public company that is traded on the exchange. In other words, each segment of a stock exchange has a different time duration for it to be in a critical state. By a stock exchange segment we mean a set of trading platforms (world stock exchanges) and market traders involved in buying/selling a share of some public company. Hereinafter we use the term stock exchange and understand it as a segment of the stock exchange.

TABLE 1 Efficiency comparison for EWM and their modifications on the stock market data.

Early warning measure	Share of united models where the p-value of the coefficient was less than 5%			Largest time horizon that would still achieve an AUC higher than 0.75 for models with separated variables (or AUC for horizon 1)		
	Original measure value	Dynamics of measure over 8 iterations	Variance of measure for 8 iterations	Original measure value	Dynamics of measure over 8 iterations	Variance of measure for 8 iterations
Hurst exponent	0.06	0.28	0.12	−(0.52)	−(0.51)	−(0.50)
Correlation dimension	0.01	0.06	0.30	−(0.52)	−(0.49)	−(0.52)
Lyapunov exponent	0.71	0.01	0.14	−(0.65)	−(0.51)	−(0.68)
Variance	0.00	1.00	0.04	−(0.53)	**5**	1
Skewness	0.96	1.00	0.95	−(0.74)	1	5
Kurtosis	0.96	1.00	0.98	−(0.73)	−(0.71)	5
Power-law scaling exponent of power spectral density	0.97	0.15	1.00	−(0.53)	−(0.52)	−(0.52)
Autocorrelation at lag-1	0.06	0.18	0.99	−(0.54)	−(0.53)	**4**
First log-cumulant	0.09	1.00	0.99	−(0.53)	−(0.55)	−(0.52)
Second log-cumulant	0.04	0.00	0.68	−(0.52)	−(0.50)	−(0.52)
Third log-cumulant	0.72	0.01	0.03	−(0.50)	−(0.50)	−(0.50)

A stock exchange in a critical state is characterized by a near-1 autocorrelation for stock's volume and a power law for the power spectral density of stock's volume with degree exponent from 1 to 2. The dynamics of a system with such characteristics is known as the avalanche-like dynamics of the system observed when it is in a critical state, also known as the edge of a phase transition (e.g., see the paper [14]). One of the first and most studied models of self-organization of systems into a critical state is the SCA model, which explains the spontaneous emergence of a system into a critical state with its avalanche-like behavior. Therefore, we used SCA not only as a system generating test dynamical series (see Subsection 2.1), but also as a basic, systemically isomorphic model of SCA in the context of systems theory, the stock market model. In other words, when building a stock exchange model, we use the analogy of structure (Chung-Lu graph of SCA and complex network of exchange transaction network), the nature of elements (stable/unstable vertices of SCA and passive/active stock exchange traders) and links (collapse of unstable vertices of SCA and buy/sell transaction of a public company share) between the elements of SCA and stock exchange.

Let Γ be a planar graph of exchange transactions with nodes (k,m), for which $k,m \in \mathbb{Z}$ are the ultrametric coordinates of the exchange traders. As Γ we used Chung-Lu graphs with two-parameter degree distribution of edges as the most common and empirically validated model determining the topological structure of exchange transactions (e.g., see the papers [48–53]).

Let $h(k,m) \in \mathbb{Z}^{+} \cup \{0\}$ be the entropy as a measure of uncertainty of information about the share of some public company, which is available to the stock exchange trader (k,m). Let $h(k,m)$ be denoted by h_c, which defines the threshold value of entropy for a trader (k,m) to sell a share to its nearest neighbour, for example, $(k+1,m)$, in the graph Γ.

Thus, each exchange trader with some number of shares can be in both an active state, denoted (k_a,m_a), and a passive state, denoted $\left(k_p,m_p\right)$. Trader (k_a,m_a) is in the active state if the corresponding entropy $h\left(k_p,m_p\right)$ is not less than a critical value, h_c. Otherwise, trader (k,m) is in the passive state. Trader (k_a,m_a), having uncertainty about a stock at least h_c, seeks to get rid of such stocks. As a result, trader (k_a,m_a) sells the shares to his nearest neighbour in the graph Γ, e.g., trader $\left(k_p,m_p\right) = (k_a+1,m_a)$, who is in the passive state and has uncertainty about the share less than h_c. In this case, trader $\left(k_p,m_p\right)$ has more information about the tendencies of the price behavior of the bought stock. After the local exchange transaction of buy/sell $(k_a,m_a) \to \left(k_p,m_p\right)$ the trader (k_a,m_a) becomes passive until he receives some information which increases the uncertainty of information about tendencies of price behavior of the share. The source of such information can be a report of a public company, mass media news or some insider information. On the contrary, after the exchange transaction $(k_a,m_a) \to \left(k_p,m_p\right)$

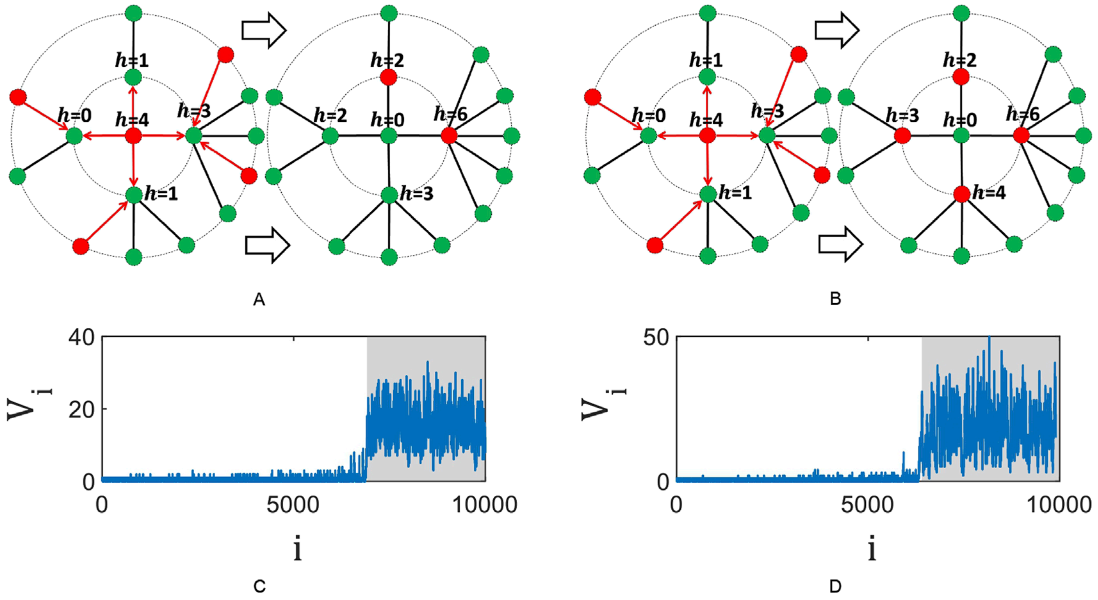


FIGURE 1
Local exchange transactions leading to self-organization of the stock exchange to the edge of the second-order phase transition **(A)** and to the edge of the first-order phase transition **(B)**. The symbol h denotes the entropy. Model series of exchange volume corresponding to self-organization of the stock exchange to the edge of the second-order phase transition **(C)** and to the edge of the first-order phase transition **(D)**. The gray region indicates the edge of a phase transition.

trader $\left(k_p, m_p\right)$ enters an active state in which he is ready to sell the stock to some of his passive nearest neighbours. Figure 1A shows local exchange collapses.

Self-organization of the stock exchange into a critical state occurs as a result of its pumping (perturbation) and relaxation at each iterative step. Each iteration starts with pumping and ends with complete relaxation of the stock exchange. Information pumping of the stock exchange leads to an increase in entropy or to an increase in the volatility of the stock, i.e., to an increase in the possibility of the stock price to change in any direction. Relaxation of the stock exchange occurs as a result of local exchange transactions of buying/selling a share and is formally defined by the following rules:

$$\begin{gathered} h(k,m) \geq h_c(k,m) \\ h(k,m) \rightarrow h(k,m) - h_c(k,m) \\ h(\mathrm{Ne}(k,m)) \rightarrow h(\mathrm{Ne}(k,m)) + \delta_p \\ \sum_{p=1}^{z_c(k,m)} \delta_p = h_c(k,m), \delta_p \geq 0 \end{gathered}, \tag{4}$$

where $h_c(k,m)$ is the critical for trader (k,m) entropy value equal to the number of its nearest neighbors in the graph Γ; $\mathrm{Ne}(k,m)$ is the nearest neighbour of trader (k,m) in the graph Γ; δ_m is a random number taking values from the set $\mathbb{Z}^+ \cup \{0\}$.

The model based on the Equation 4 explains the phenomenon of self-organization of the stock exchange into a critical state starting from some critical iteration i_c. Starting from initial public offering $(i = 0)$ and up to the moment of completion of the subcritical phase $(0 < i < i_c)$, the stock exchange observes a small number of share buy/sell transactions, which quickly decay in ultrametric space and time. The global information pumping of the stock exchange to a critical entropy value H_c brings the stock exchange into the critical state $(i \geq i_c)$. Staying in a small neighbourhood of H_c the stock exchange is unstable to small information perturbations. In such an unstable state, a small entropy increment $(H_c \pm \delta H)$ is sufficient for the stock exchange to experience avalanches of stock buy/sell transactions. The stock volume series, V_i, in the critical state of the stock exchange $(i \geq i_c)$ is characterised by $\rho(1) = 1$, $S(f) = f^{-1}$, and $p(\xi) = \xi^{-2}$. The dynamic series V_i, demonstrating the dynamics of such self-organization, is presented in Figure 1C.

The above described self-organization of the stock exchange corresponds to its self-organization to the edge of the second-order phase transition. To describe the self-organization of the stock exchange to the edge of the first-order phase transition, the following changes in the rules of model (1) are sufficient. Any stock exchange trader (k,m), who is in the passive state $\left(k_p, m_p\right)$, can move to the active state $\left(k_a, m_a\right)$ if $h(k,m) \geq h_c(k,m)$, and if he has purchased a share from at least one of his nearest neighbours. The latter is characteristic of the stock exchange during the period of increased activity of its traders, i.e., when each trader $\left(k_p, m_p\right)$, having bought a share from a neighboring trader, passes to the state $\left(k_a, m_a\right)$ independently of the entropy value $h(k,m)$. Being in the state $\left(k_a, m_a\right)$ a trader immediately tries to sell the bought share. Such a stock exchange is dominated by speculative buy/sell transactions of the stock. Figure 1B demonstrates the corresponding local stock exchange collapses. The dynamic series V_i, demonstrating the dynamics of self-organization of the stock exchange to the edge of the first-order phase transition, is presented in Figure 1D. Local exchange transactions of buying/selling a stock are formally determined by the following rules:

$$\begin{array}{l} h(k,m) \geq h_c(k,m) \vee 2 \leq h(k,m) < h_c(k,m) \\ h(k,m) \geq h_c(k,m) : \begin{cases} h(k,m) \rightarrow h(k,m) - h_c(k,m) \\ h(\mathrm{Ne}(k,m)) \rightarrow h(\mathrm{Ne}(k,m)) + \delta_p \\ \sum\limits_{p=1}^{z_c(k,m)} \delta_p = h_c(k,m), \delta_p \geq 0 \end{cases} \\ 2 \leq h(k,m) < h_c(k,m) : \begin{cases} h(k,m) \rightarrow h(k,m) - h_c(k,m) \\ h(\mathrm{Ne}(k,m)) \rightarrow h(\mathrm{Ne}(k,m)) + \delta_p \\ \sum\limits_{p=1}^{z_c(k,m)} \delta_p = h_c(k,m), \delta_p \geq 0 \\ h(\mathrm{Ne}(k,m)) \rightarrow h(\mathrm{Ne}(k,m)) + 1 \end{cases} \end{array} . \qquad (5)$$

Note that the proposed models which are based on the Equation 5 determine the self-organization of the stock exchange into a critical state, which does not require fine-tuning of the control parameter H to the critical value H_c. Exit to the critical state is achieved as a result of perturbation and relaxation of the stock exchange, as well as the above-described nonlinear interactions between the stock exchange traders.

3.3 Early warning signals for stock exchange self-organization into a critical state

One of the results of our calculations is the independence of the behavior of the series for any of the EWMs in the vicinity of the critical onset from the specific public company for which the EWM series was calculated. The EWMs series differ only in their noise and early warning time (see Subsection 3.1). Apparently, the self-organization of a stock exchange into a critical state is a universal phenomenon. Therefore, we will limit ourselves to discussing the behavior of a series of EWMs for stock exchange transactions of, for example, Ameris Bancorp. This company is a bank holding company that, through its subsidiary Ameris Bank, provides banking services to its retail and commercial customers.

Figure 2 shows the behavior of the moving average smoothed series of EWMs that are obtained for the stock volume series of Ameris Bancorp from 10:30 7 February 2022 to 15:30 p.m. 5 February 2024. The smoothing of these series reduced the number of false early warning signals.

The MA100 series obtained for the stock volume series increases sharply in the vicinity of the critical point, t_c, i.e., the time when the stock exchange starts to self-organize into a critical state (see Figure 2A). The time t_c corresponds to 15:30 10 March 2023. The MA100 series increased by 20% compared to the volumes of the previous 5 hours at 11:30 10 March 2023. Therefore, no more than 4 h are given to take preventive measures to avoid self-organization of the stock exchange into a critical state.

The above described behavior of the MA100 series is a consequence of the critical slowdown of the stock exchange, the manifestation of which is an increase in the average amplitude of stochastic fluctuations of the order parameter (stock volume). Indeed, in the vicinity of t_c there is an increase in the average amplitude of stochastic fluctuations of stock volume, which leads to an increase in MA100.

Other evidence of the critical slowing down of the stock market in the vicinity of t_c is the behavior of window variance (see Figure 2B), kurtosis (see Figure 2C), skewness (see Figure 2D), autocorrelation at lag-1 (see Figure 2E), and power-law scaling exponent of the power spectral density (see Figure 2F) characteristic of the critical slowing down. These measures increase sharply in the neighborhood of t_c. At the same time, kurtosis and skewness take positive values, which is a consequence of the increase in the amplitude of stochastic fluctuations of stock volume. Moreover, autocorrelation at lag-1 and power-law scaling exponent of the power spectral density take values close to 1 in the time interval from 15:30 10 March 2023 to 15:30 p.m. 27 March 2023. Thus, the stock exchange has been in a critical state for 17 trading days. In Figure 2, the interval corresponding to the critical state, or the edge of the phase transition, is shown as a gray region. The stock exchange in this interval is characterized by abnormal fluctuations of the stock volume and strong, close to 1, correlation between neighboring elements of the sequence of values of the stock volume.

Another sign of the stock volume series approaching t_c is a sharp increase of the generalised Hurst exponent to the value of 0.63 in the interval corresponding to the critical state (see Figure 2G). Consequently, if the stock volume series is considered as a real-time series, the sequence of values of the stock volume becomes more correlated as the stock volume series approaches t_c. The stock volume series corresponding to the critical state is a time series with long-term positive autocorrelation. Based on the fact that the position of the center of the multifractal spectrum, $h_0 = C_1$, shifts to the right as the stock approaches t_c (see Figure 2H), the stock volume series becomes more singular in the vicinity of t_c. The width, $W = C_2$, and skewness, $S = C_3$, of the multifractal spectrum increase as the stock volume series approaches t_c (see Figures 2I, J). The multifractal spectrum becomes symmetric, $C_3 = S = 1$, at $t = t_c$ (see Figure 2J). Since $S < 1$ at $t < t_c$,the multifractal spectrum for the subcritical phase, $t < t_c$, is asymmetric with small fluctuations dominating the stock volume. Consequently, in the neighborhood of t_c the stock volume series becomes a more inhomogeneous series with dominance of large fluctuations. Thus, the described behavior of multifractal measures and Hurst exponent are early warning signals for the stock exchange self-organization into a critical state.

Let us consider the behavior of the series of EWMs, the calculation of which is based on the reconstruction of the phase space of the stock exchange. As the stock exchange approaches t_c the correlation dimension of the reconstructed attractor increases (see Figure 2K), hence the fractal structure of the attractor becomes more complex and the chaotic behavior of the stock exchange becomes more complicated. The most complex chaotic behavior of the stock exchange, corresponding to the highest value of the correlation dimension, is observed in its critical state. An indication of the increasing complexity of the chaotic behavior of the stock exchange is also an increase in the largest Lyapunov exponent, which is positive, as the stock volume series approaches t_c (see Figure 2L). The most complex chaotic dynamics of the stock exchange also corresponds to its critical state, since the largest value of the exponent is observed in the time interval corresponding to the critical state.

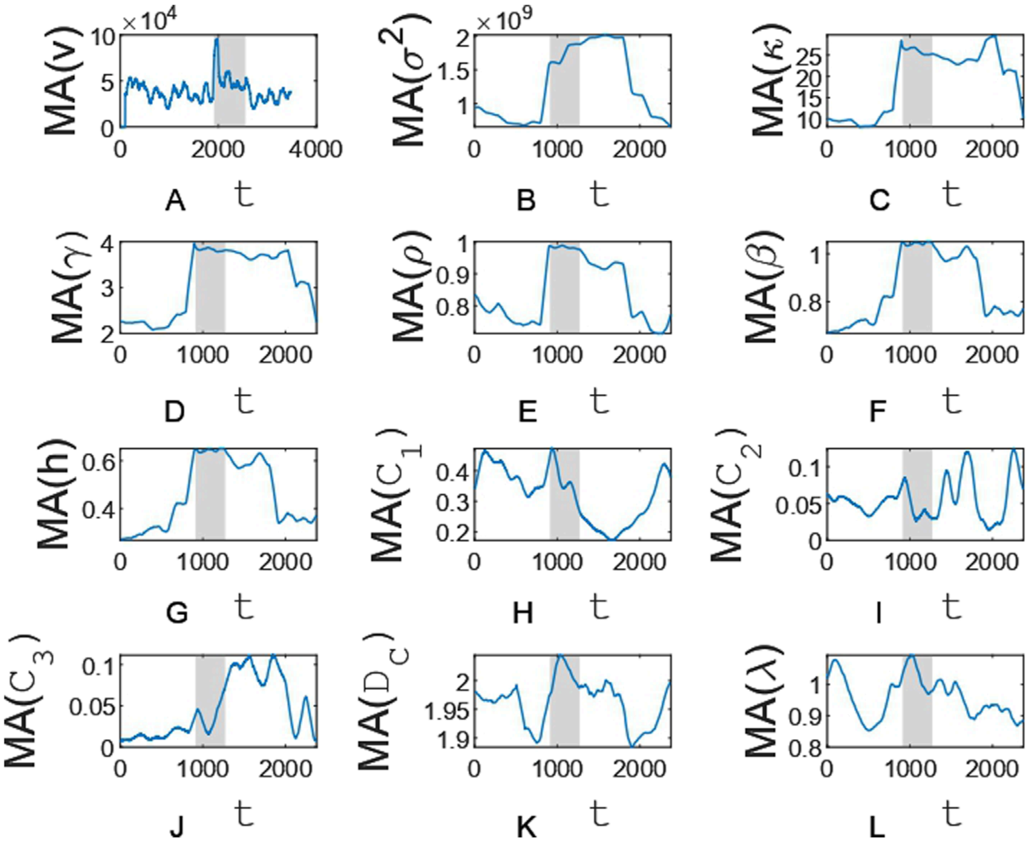


FIGURE 2
Moving average series for the stock volume series **(A)**, variance **(B)**, kurtosis **(C)**, skewness **(D)**, autocorrelation at lag-1 **(E)**, power-law scaling exponent of the power spectral density **(F)**, generalized Hurst exponent **(G)**, position of the multifractal spectrum maximum **(H)**, multifractal spectrum width **(I)**, multifractal spectrum skewness **(J)**, correlation dimension **(K)**, and largest Lyapunov exponent **(L)**. The gray region indicates the edge of a phase transition.

4 Conclusion

The stock exchange self-organizes to the edge of a phase transition. The duration of a stock exchange at the edge ranges from 7 to 19 trading hours and depends on the public company whose shares are traded on the stock exchange. We set such durations for public company stocks from the Russel 3,000 index, which measures the performance of the 3,000 largest US companies by market capitalization. Perhaps the result of finding time intervals corresponding to the edge of a phase transition for more public company stocks would be a longer range of trading day durations. In addition, further research of the time intervals should be focused on the analysis of the stock volume series with higher frequency, such as every second and every minute series, but adjusted for the volumes of pre-planned execution of deals. Analyzing such series will allow you to identify the time intervals that cannot be identified in hourly stock volume. For example, high-frequency trading implies the conclusion of a large number of buy/sell transactions in a fraction of a second and it may take several seconds for the stock exchange to self-organize to the edge of a phase transition. If the duration of the stock exchange on the edge of a phase transition is less than 1 h, the analysis of the hourly stock volume series will not allow to identify the time interval corresponding to the edge. The best identification will be obtained when analyzing the second-by-second series for the stock volume. In addition, the transition to more frequent stock volume series will allow to obtain segments of series corresponding to the edge, of longer length and possibly of sufficient length to obtain a reliable estimate for the power-law scaling exponent of the power spectral density. Comparison of such estimates will allow us to determine which of the critical states, i.e., the edge of the phase transition of the first or second kind, corresponds to the detected time interval.

The sandpile cellular automaton model of self-organization to the edge of a phase transition is based on the idea that information drives stock markets (e.g., see the paper [54]). Self-organization of a stock exchange occurs in a discrete number of steps, each of which begins with an information perturbation of the stock exchange and ends with its relaxation. If the information pumping results in supra-critical uncertainty, or entropy, in the price behavior of a stock for some traders, then the stock exchange relaxation occurs as a result of these traders' execution of stock buy/sell transactions, which reduces the uncertainty in the price behavior of the stock for the traders. We have considered implementations of the model under the assumption that all traders are characterized by a single critical level of uncertainty. In the context of effective market hypothesis such assumption is quite reasonable, but it is not applicable when analyzing the stock market in the context of fractal market hypothesis. Therefore, further improvement of the model should be focused on the study of the influence of the type and parameters of the probability distribution of critical uncertainty on the behaviour of the stock volume series when the stock exchange approaches the edge of a phase transition, as well as on the edge. Another direction of the model improvement is the introduction of an assumption about the existence of some critical uncertainty of price behaviour, which determines the condition of buying a share of a public company. Moreover, the critical uncertainty when buying a share is not equal to the critical uncertainty when selling it.

The studied early warning measures, first of all MA100, variance, kurtosis and skewness as the most effective ones, can be used to detect early warning signals for self-organization of the stock exchange to the edge of a phase transition in real-time early warning systems. Such signals are important for the regulator of trading on the stock exchange, as they allow detecting illegal exchange operations. The volume indicator reflects an increase or decrease in the activity of traders on the stock exchange. Therefore, early detection of the time interval in the stock volume series corresponding to the stock exchange's edge will allow a trader to make reasonable and timely changes in his trading strategy. As a rule, traders correlate the volume indicator with the direction of the stock price movement. If the stock price is rising along with the volume, the price growth is likely to continue. High volume (25% higher than average) when the stock price reaches a new high is a harbinger of a strong increase in the stock price. Traders should refrain from selling existing shares and/or buy shares while they are cheap and sell them when they rise in price. If the share price is declining while volume is rising, the stock market is dominated by stock sellers - the trader should refrain from speculating in the stock.

Data availability statement

The datasets presented in this study can be found in online repositories. The names of the repository/repositories and accession number(s) can be found below: https://github.com/lebedevaale/early_warning_model.

Author contributions

AD: Conceptualization, Formal Analysis, Funding acquisition, Methodology, Validation, Writing–original draft. AL: Data curation, Investigation, Resources, Software, Validation, Visualization, Writing–review and editing. VK: Data curation, Funding acquisition, Project administration, Supervision, Writing–review and editing. VD: Conceptualization, Data curation, Funding acquisition, Investigation, Project administration, Visualization, Writing–review and editing.

Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. The work is an output of a research project implemented as part of the Basic Research Program at the National Research University Higher School of Economics (HSE 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.

Generative AI statement

The author(s) declare that no Generative AI was used in the creation of this manuscript.

Publisher's note

References

1. 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

2. Watkins NW, Pruessner G, Chapman SC, Crosby NB, Jensen HJ. 25 Years of self-organized criticality: concepts and controversies. *Space Sci Rev* (2016) 198:3–44. doi:10.1007/s11214-015-0155-x

3. Walter N, Hinterberger T. Self-organized criticality as a framework for consciousness: a review study. *Front Psychol* (2022) 13:911620. doi:10.3389/fpsyg.2022.911620

4. 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

5. Zhukov D. How the theory of self-organized criticality explains punctuated equilibrium in social systems. *Methodological Innov* (2022) 15:163–77. doi:10.1177/20597991221100427

6. Tadic B, Melnik R. Self-organized critical dynamics as a key to fundamental features of complexity in physical, biological, and social networks. *Dynamics* (2021) 1:181–97. doi:10.3390/dynamics1020011

7. Zhukov D. Personality and society in the theory of self-organized criticality. *Changing Societies and Personalities* (2023) 7:10–33. doi:10.15826/csp.2023.7.2.229

8. Smyth WD, Nash JD, Moum JN. Self-organized criticality in geophysical turbulence. *Scientific Rep* (2019) 9:3747. doi:10.1038/s41598-019-39869-w

9. Mikaberidze G, Plaud A, D'Souza RM. Dragon kings in self-organized criticality systems. *Phys Rev Res* (2023) 5:L042013. doi:10.1103/PhysRevResearch.5.L042013

10. Alodjants AP, Bazhenov AY, Khrennikov AY, Bukhanovsky AV. Mean-field theory of social laser. *Scientific Rep* (2022) 12:8566. doi:10.1038/s41598-022-12327-w

11. Alodjants AP, Tsarev DV, Avdyushina AE, Khrennikov AY, Boukhanovsky AV. Quantum-inspired modeling of distributed intelligence systems with artificial

intelligent agents self-organization. *Scientific Rep* (2024) 14:15438. doi:10.1038/s41598-024-65684-z

12. Jarai AA. The sandpile cellular automaton. *Complexity Comput* (2018) 27:79–88. doi:10.1007/978-3-319-65558-1_6

13. Shapoval A, Shnirman M. Explanation of flicker noise with the Bak-Tang-Wiesenfeld model of self-organized criticality. *Phys Rev E* (2024) 110:014106. doi:10.1103/PhysRevE.110.014106

14. Buendia V, di Santo S, Bonachela JA, Muñoz MA. Feedback mechanisms for self-organization to the edge of a phase transition. *Front Phys* (2020) 8:333. doi:10.3389/fphy.2020.00333

15. Song S, Li H. Early warning signals for stock market crashes: empirical and analytical insights utilizing nonlinear methods. *EPJ Data Sci* (2024) 13:16. doi:10.1140/epjds/s13688-024-00457-2

16. Ran M, Tang Z, Chen Y, Wang Z. Early warning of systemic risk in stock market based on EEMD-LSTM. *PLoS One* (2024) 19:e0300741. doi:10.1371/journal.pone.0300741

17. Bartolozzi M, Leinweber D, Thomas A. Self-organized criticality and stock market dynamics: an empirical study. *Physica A: Stat Mech its Appl* (2005) 350:451–65. doi:10.1016/j.physa.2004.11.061

18. Stanley HE, Amaral LAN, Buldyrev SV, Gopikrishnan P, Plerou V, Salinger MA. Self-organized complexity in economics and finance. *PNAS* (2002) 99:2561–5. doi:10.1073/pnas.022582899

19. Goncalves CP. Chaos-induced self-organized criticality in stock market volatility-an application of smart topological data analysis. *Int J Swarm Intelligence Evol Comput* (2023) 12:331. doi:10.35248/2090-4908.22.12.331

20. Chandra A, Reinstein A. A study of production, stock prices and self-organized criticality. *Rev Account Finance* (2004) 3:20–39. doi:10.1108/eb043406

21. Kaki B, Farhang N, Safari H. Evidence of self-organized criticality in time series by the horizontal visibility graph approach. *Scientific Rep* (2022) 12:16835. doi:10.1038/s41598-022-20473-4

22. Diks C, Hommes C, Wang J. Critical slowing down as an early warning signal for financial crises? *Empirical Econ* (2019) 57:1201–28. doi:10.1007/s00181-018-1527-3

23. Kang BS, Park C, Ryu D, Song W. Phase transition phenomenon: a compound measure analysis. *Physica A: Stat Mech its Appl* (2015) 428:383–95. doi:10.1016/j.physa.2015.02.044

24. Biondo AE, Pluchino A, Rapisarda A. Order book, financial markets, and self-organized criticality. *Chaos, Solitons and Fractals* (2016) 88:196–208. doi:10.1016/j.chaos.2016.03.001

25. Schmidhuber C. Financial markets and the phase transition between water and steam. *Physica A* (2022) 592:126873. doi:10.1016/j.physa.2022.126873

26. Dmitriev A, Lebedev A, Kornilov V, Dmitriev V. Multifractal early warning signals about sudden changes in the stock exchange states. *Complexity* (2022) 2022:8177307. doi:10.1155/2022/8177307

27. Biondo AE, Pluchino A, Rapisarda A. Modeling financial markets by self-organized criticality. *Phys Rev E* (2015) 92:042814. doi:10.1103/PhysRevE.92.042814

28. Rao B, Yi D, Zhao C. Self-organized criticality of individual companies: an empirical study. *Third Int Conf Nat Comput (ICNC 2007)* (2007) 481–7. doi:10.1109/ICNC.2007.654

29. Zhang D, Zhuang Y, Tang P, Peng H, Han Q. Financial price dynamics and phase transitions in the stock markets. *The Eur Phys J B* (2023) 96:35. doi:10.1140/epjb/s10051-023-00501-6

30. Chung F, Lu L. Connected components in random graphs with given expected degree sequences. *Ann Combinatorics* (2002) 6:125–45. doi:10.1007/pl00012580

31. Manna SS. Sandpile models of self-organized criticality. *Curr Sci* (1999) 77:388–93. Available from: https://www.jstor.org/stable/24102958. (Accessed December 24, 2024).

32. di Santo S, Burioni R, Vezzani A, Muñoz MA. Self-organized bistability associated with first-order phase transitions. *Phys Rev Lett* (2016) 116:240601. doi:10.1103/PhysRevLett.116.240601

33. George SV, Kachhara S, Ambika G. Early warning signals for critical transitions in complex systems. *Physica Scripta* (2023) 98:072002. doi:10.1088/1402-4896/acde20

34. Zhao L, Li W, Yang C, Han J, Su Z, Zou Y. Multifractality and network analysis of phase transition. *PLoS ONE* (2017) 12:e0170467. doi:10.1371/journal.pone.0170467

35. Hasselman F. Early warning signals in phase space: geometric resilience loss indicators from multiplex cumulative recurrence networks. *Front Physiol* (2022) 13:859127. doi:10.3389/fphys.2022.859127

36. Dmitriev A, Lebedev A, Kornilov V, Dmitriev V. Twitter self-organization to the edge of a phase transition: discrete-time model and effective early warning signals in phase space. *Complexity* (2023) 2023:1–13. doi:10.1155/2023/3315750

37. Astfalck LC, Sykulski AM, Cripps EJ. Debiasing Welch's method for spectral density estimation. *Biometrika* (2024) 111:1313–29. doi:10.1093/biomet/asae033

38. Shanshan Z, Jiang Y, He W, Mei Y, Xie X, Wan S. Detrended fluctuation analysis based on best-fit polynomial. *Front Environ Sci* (2022) 10. doi:10.3389/fenvs.2022.1054689

39. Achour R, Li Z, Selmi B, Wang T. A multifractal formalism for new general fractal measures. *Chaos, Solitons and Fractals* (2024) 181:114655. doi:10.1016/j.chaos.2024.114655

40. Nazarimehr F, Jafari S, Hashemi Golpayegani SMR, Sprott JC. Can Lyapunov exponent predict critical transitions in biological systems? *Nonlinear Dyn* (2017) 88:1493–500. doi:10.1007/s11071-016-3325-9

41. Takens F. Detecting strange attractors in turbulence. Dynamical systems and turbulence, warwick 1980. *Lecture Notes Mathematics* (1981) 898:366–81. doi:10.1007/BFb0091924

42. Weber I, Oehrn CR. NoLiTiA: an open-source toolbox for non-linear time series analysis. *Front Neuroinformatics* (2022) 16:876012. doi:10.3389/fninf.2022.876012

43. Rosenstein MT, Collins JJ, De Luca CJ. A practical method for calculating largest Lyapunov exponents from small data sets. *Physica D: Nonlinear Phenomena* (1993) 65:117–34. doi:10.1016/0167-2789(93)90009-P

44. Dmitriev A, Lebedev A, Kornilov V, Dmitriev V. Effective precursors for self-organization of complex systems into a critical state based on dynamic series data. *Front Phys* (2023) 11. doi:10.3389/fphy.2023.1274685

45. Mooney CZ, Duval RD. *Bootstrapping: a nonparametric approach to statistical inference*. Newbury Park, CA, USA: Sage Publications (1993).

46. Chib S. Analysis of multivariate probit models. *Biometrika* (1998) 85:347–61. doi:10.1093/biomet/85.2.347

47. Berger VW, Zhou Y. *Kolmogorov-Smirnov test: overview*. Wiley StatsRef: Statistics Reference Online (2024). doi:10.1002/9781118445112.stat06558

48. Tao B, Dai HN, Wu J, Ho IWH, Zheng Z, Cheang CF. Complex network analysis of the bitcoin transaction network. *IEEE Trans Circuits Syst Express Briefs* (2022) 69:1009–13. doi:10.1109/TCSII.2021.3127952

49. Jihun P, Cho CH, Lee JW. A perspective on complex networks in the stock market. *Front Phys* (2022) 10. doi:10.3389/fphy.2022.1097489

50. Huang W, Wang H, Wei Y, Chevallier J. Complex network analysis of global stock market co-movement during the COVID-19 pandemic based on intraday open-high-low-close data. *Financial Innovation* (2024) 10:7. doi:10.1186/s40854-023-00548-5

51. Moghadam HE, et al. Complex networks analysis in Iran stock market: the application of centrality. *Physica A* (2019) 531:121800. doi:10.1016/j.physa.2019.121800

52. Wang Z, Zhang G, Ma X, Wang R. Study on the stability of complex networks in the stock markets of key industries in China. *Entropy* (2024) 26:569. doi:10.3390/e26070569

53. Liu XF, Tse CK. A complex network perspective of world stock markets: synchronization and volatility. *Int J Bifurcation Chaos* (2012) 22:1250142. doi:10.1142/S0218127412501428

54. Szczygielski JJ, Charteris A, Bwanya PR, Brzeszczyński J. Google search trends and stock markets: sentiment, attention or uncertainty? *Int Rev Financial Anal* (2024) 91:102549. doi:10.1016/j.irfa.2023.102549

Frontiers in Physics

TYPE Original Research
PUBLISHED 30 May 2025
DOI 10.3389/fphy.2025.1622487

OPEN ACCESS

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

REVIEWED BY
Huihai Wang,
Central South University, China
Minglin Ma,
Xiangtan University, China
Ning Wang,
Changzhou University, China

*CORRESPONDENCE
Yujiao Dong,
✉ yjdong@hdu.edu.cn

RECEIVED 03 May 2025
ACCEPTED 19 May 2025
PUBLISHED 30 May 2025

CITATION
Wang X, Dong Y, Wang G, Zhou Z, Mao Y, Jin P and Liang Y (2025) Monophasic and biphasic neurodynamics of bi-S-type locally active memristor.
Front. Phys. 13:1622487.
doi: 10.3389/fphy.2025.1622487

Monophasic and biphasic neurodynamics of bi-S-type locally active memristor

Xinyi Wang[1], Yujiao Dong[1]*, Guangyi Wang[1,2], Ziyu Zhou[1], Yidan Mao[1], Peipei Jin[1] and Yan Liang[1]

[1]Zhejiang Key Laboratory of Intelligent Vehicle Electronics Research, Hangzhou Dianzi University, Hangzhou, China, [2]College of Computer and Information Engineering, Qilu Institute of Technology, Jinan, Shandong, China

Inspired by the energy-efficient information processing of biological neural systems, this paper proposes an artificial memristive neuron to reproduce biological neuronal functions. By leveraging Chua's unfolding theorem, we establish a bi-S-type locally active memristor mathematical model exhibiting negative differential resistance (NDR), which serve as fingerprints for local activity. A second-order neuronal circuit is constructed to emulate periodic spiking and excitability, while a third-order circuit extends functionality to chaotic oscillations and bursting behaviors. Besides, the constructed neuronal circuit generates biphasic action potential through voltage symmetry modulation, replicating bidirectional signal transmission akin to biological systems. Hardware emulation validates neurodynamics under varying stimuli from theoretical analyses, offering a unit module and theoretical reference for energy-efficient neuromorphic computing network.

KEYWORDS

memristor, local activity, neuron, spikes, neuromorphic behaviors

1 Introduction

As information technology rapidly advances, traditional computing architectures face growing limitations in energy efficiency and computational complexity. Against this backdrop, neuromorphic computing has emerged as a novel computing paradigm [1, 2]. Its core concept is to emulate information processing mechanisms of biological systems by constructing brain-like computing structures to achieve energy-efficient computation [3, 4]. This brain-inspired approach demonstrates superior capabilities in adaptive learning, positioning it as a cornerstone for next-generation intelligent systems [5–7]. Central to this technology are neuroelectronic devices that emulate neuronal functions, which are fundamental units in the construction of neuromorphic computing systems [8, 9].

Current neuroelectronic implementations primarily employ Complementary Metal-Oxide-Semiconductor (CMOS) circuits, leveraging mature fabrication techniques to simulate membrane potential dynamics and action potential generation [10, 11]. However, CMOS-based neurons suffer from inherent limitations including complex circuit topologies and elevated power consumption hinder scalability in large neural networks [12]. Memristive devices present an alternative solution through their intrinsic nonlinearity and low-power operation [13], yet conventional passive memristors require auxiliary negative impedance converters to achieve neuronal dynamics, compromising system integration efficiency [14, 15]. These challenges have driven the exploration of locally active memristors (LAMs) [16, 17], whose negative differential resistance (NDR) enables weak signal amplification and action potential generation without external circuitry [18–20].

Recent advancements in LAM-based neuronal modeling demonstrate promising results. The FitzHugh-Nagumo circuit modifications using N-type LAMs successfully replicate biological spiking patterns [21, 22]. Enhanced LAM designs with ultra-robust NDR characteristics further enable hardware implementation of nine distinct neuronal firing modes [4]. However, these advancements remain primarily confined to monophasic action potential emulation. Emerging experimental evidence from sciatic nerve electrophysiology and myocardial fiber studies demonstrates that biphasic potentials constitute fundamental encoding mechanisms in neural systems [23–25], enabling sophisticated information processing [26]. Current neuromorphic platforms predominantly neglect this biphasic paradigm, impeding hardware-level implementation of biologically plausible neural networks.

Memristive neurons exhibit broad application potential in neuromorphic systems, including frequency-based classifiers for animal sound recognition [27], image protection systems [28, 29], and cyclic neural networks with self-adaptive synapses [30]. Notably, their implementation in artificial neural networks has achieved high-precision MNIST digit recognition and effective edge detection in image processing [31].

The structure of this work is as follows: section 2 characterizes the proposed bi-S-type LAM's nonlinear dynamics; section 3 constructed a second-order neuronal circuit and demonstrates spiking regimes; Section 4 illustrates various monophasic neurodynamics, biphasic spikes, and symmetry behaviors in the third-order memristive neuron. Section 5 gives the circuit simulated validation.

2 Bi-S-type locally active memristor

Most nanoscale memristors fabricated using various materials exhibit characteristics of generic or extended memristors. Chua's unfolding theorem provides a systematic method to construct generic memristor models [32]. A generic current-controlled memristor can be defined as

$$\begin{cases} v = R_\mathrm{m}(x)i = \left(\sum\limits_{k=0}^{r} d_k x^k\right) i \\ \dfrac{dx}{dt} = f(x,i) = \sum\limits_{k=0}^{n} \alpha_k x^k + \sum\limits_{k=0}^{m} \beta_k i^k + \sum\limits_{k=0}^{p}\sum\limits_{l=0}^{q} \delta_{kl} i^k x^l \end{cases} \tag{1}$$

where v, i, and x are the voltage, current, and state variable of the memristor, respectively; $R_\mathrm{m}(x)$ represents memristance; $f(x, i)$ is the state-controlled equation; α_k, β_k, δ_{kl}, and d_k are tunable parameters.

Using Equation 1, we derive a memristor model characterized by

$$\begin{cases} v_\mathrm{m} = R_\mathrm{M}(x) i_\mathrm{m} = (d_2 x^2 + d_0) i_\mathrm{m} = f_1(x, i_\mathrm{m}) \\ \dfrac{dx}{dt} = \delta_0 + \alpha_1 x + \beta_2 {i_\mathrm{m}}^2 = f_2(x, i_\mathrm{m}) \end{cases} \tag{2}$$

with parameters: $\delta_0 = 3 \times 10^4$, $\alpha_1 = -3 \times 10^3$, $\beta_2 = -8 \times 10^7$, $d_2 = 2$, $d_0 = 20$.

2.1 Fingerprints of locally active memristor

Chua indicates that a pinched hysteresis loop in the voltage-current plane constitutes a definitive memristor signature [33]. The negative differential resistance (NDR) regions on the DC V-I curve serve as critical indicators of local activity in one-port memristors [34]. These are fingerprints of LAMs.

2.1.1 fingerprint 1: pinched hysteresis loop

Let us apply a sinusoidal voltage $v = A\sin(2\pi ft)$ with amplitude $A = 5$ V and frequencies $f = 2$ kHz, 5 kHz, 200 kHz to the proposed model. The characteristics of input voltage v_m and response current i_m are depicted in Figure 1A. It shows that the loci plotted on the v_m-i_m plane is a pinched hysteresis loop at $f = 2$ kHz (dark red curve). The lobe area decreases progressively with increasing frequency (black curve: $f = 5$ kHz), collapsing to a linear blue line at $f = 200$ kHz, confirming memristive behavior.

2.1.2 fingerprint 2: negative differential resistance (NDR) regions

The DC V-I curve (Figure 1B), obtained by sweeping DC currents from −25 mA to 25 mA with the step size of 0.1 mA, reveals two NDR regions (yellow shading) corresponding to local activity. Signal amplification occurs at these operating points where $V \in [0.385\text{ V}, 1.287\text{ V}]$ ($I \in [9.2\text{ mA}, 19.1\text{ mA}]$) and $V \in [-1.287\text{ V}, -0.385\text{ V}]$ ($I \in [-19.1\text{ mA}, -9.2\text{ mA}]$).

However, the operating points Q (V, I) of the LAM exhibit instability when biased at $V \in [-1.287\text{ V}, -0.385\text{ V}] \cup [0.385\text{ V}, 1.287\text{ V}]$. As shown in Figure 2 (left inset), three intersections (M_0, M_1, M_2) or (M_3, M_4, M_5) emerge at $V = \pm1$ V: two stable (M_1, M_2 or M_4, M_5) and one unstable (M_0 or M_3), which is verified by the dynamic route with x-dx/dt. Then, stabilization was achieved by adding an appropriate resistor $R_0 = 1$ kΩ, and the obtained locally active voltages are $V \in [R_0 I_\mathrm{D} + V_\mathrm{D}, R_0 I_\mathrm{C} + V_\mathrm{C}] \cup [R_0 I_\mathrm{B} + V_\mathrm{B}, R_0 I_\mathrm{A} + V_\mathrm{A}]$, i.e., $V \in [-19.485\text{ V}, -10.487\text{ V}] \cup [10.487\text{ V}, 19.485\text{ V}]$ (Figure 2, right inset). Observe that single stable equilibria emerge under these two operating points.

2.2 Small-signal equivalent circuit of LAM

The small-signal equivalent circuit enables nonlinear dynamics prediction at arbitrary operating points. By applying Taylor series expansion to Equation 2 at operating point Q (V_Q, I_Q) under sufficiently small signals (ignoring higher-order terms), we obtain

$$\begin{cases} \triangle v = a_{11}(Q) \triangle x + a_{12}(Q) \triangle i \\ \triangle \dot{x} = b_{11}(Q) \triangle x + b_{12}(Q) \triangle i \end{cases} \tag{3}$$

where Δv, Δi, and Δx denote small perturbations; $a_{11} = (\delta f_1/\delta x)|_Q = 2d_2 I_Q X_Q$, $a_{12} = (\delta f_1/\delta i)|_Q = d_2 {X_Q}^2 + d_0 = R_\mathrm{M}(X)$, $b_{11} = (\delta f_2/\delta x)|_Q = \alpha_1$, $b_{12} = (\delta f_2/\delta i)|_Q = 2\beta_2 I_Q$. Here, the differential resistance $R_\mathrm{D} = (\Delta v/\Delta i)|_Q = a_{11}(\Delta x/\Delta i) + a_{12} = a_{12} - (a_{11}b_{12})/b_{11}$, the equivalent resistance $R_\mathrm{M} = a_{12}$.

Taking the Laplace transform of (Equation 3) yeilds

$$\begin{aligned} Z(s,Q) &= \frac{\hat{v}(s)}{\hat{i}(s)} = \frac{a_{11}(Q) b_{12}(Q)}{s - b_{11}(Q)} + a_{12}(Q) \\ &= \frac{1}{\frac{s}{b_{11}(R_\mathrm{M} - R_\mathrm{D})} + \frac{1}{R_\mathrm{D} - R_\mathrm{M}}} + R_\mathrm{M} = \frac{1}{C_\mathrm{s} s + \frac{1}{R_\mathrm{s}}} + R_\mathrm{M} \end{aligned} \tag{4}$$

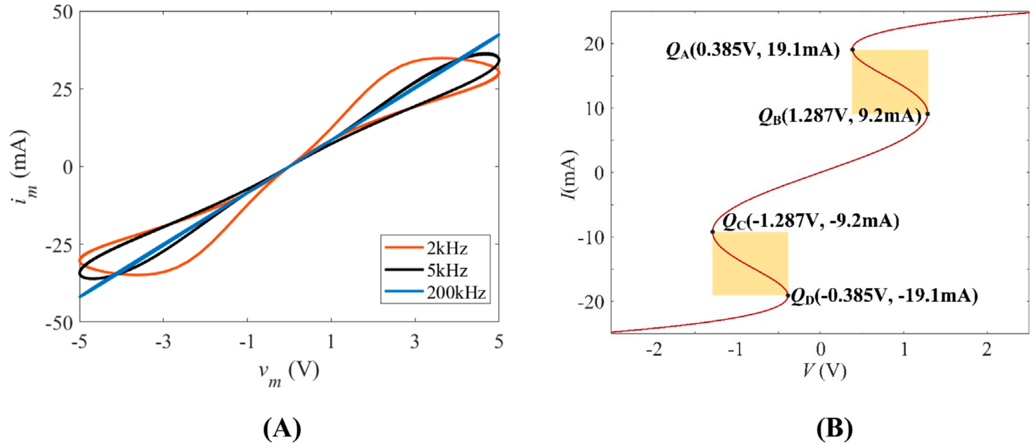


FIGURE 1
(A) Pinched hysteresis loops measured from Equation 2 on v_m-i_m plane for input voltages v_m with amplitude A = 5 V and frequencies f = 2 kHz, 5 kHz, 200 kHz; **(B)** DC V-I curve of the LAM with the shaded NDR regions.

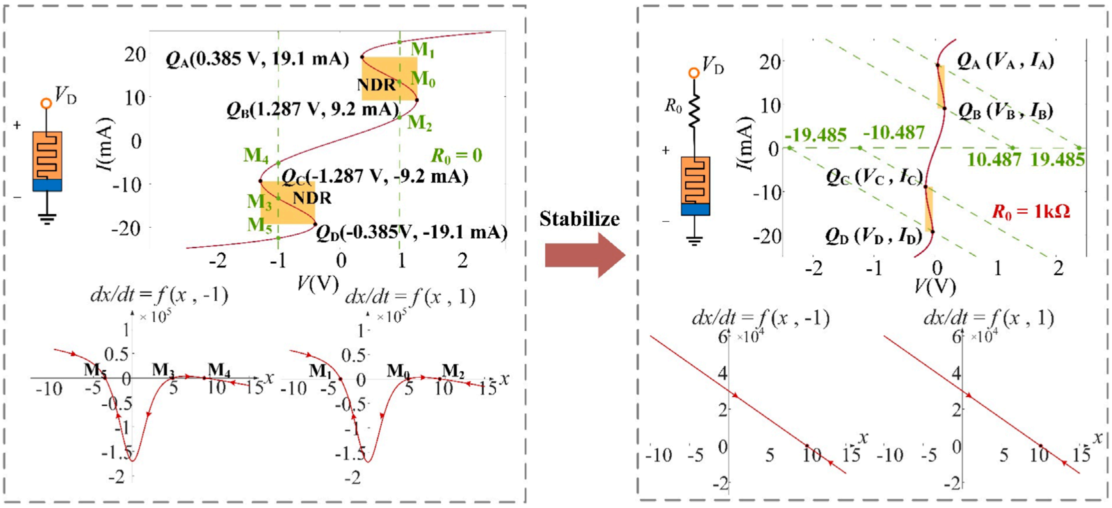


FIGURE 2
Stabilization mechanism: (left) unstable equilibria in the memristor without resistors; (right) stabilized operation with series resistor R_0.

Figure 3A illustrates the small-signal equivalent circuit of the LAM about an operating point: a parallel R_s-C_s network in series with R_M. Figure 3B shows parameter variations under $V \in$ [10 V, 20 V]. Notably, negative capacitances (C_s < 0) occur at locally active voltages $V \in$ [10.487 V, 19.485 V], which critically determine memristive characteristics for neuronal circuit design. Similar trends hold for $V \in$ [−19.485 V, −10.487 V].

3 LAM-based second-order neuron

To construct a second-order neuronal circuit using the LAM, an external capacitor C_0 is required to compensate for the inductive behavior of the LAM in locally active domains (LADs). The proposed circuit includes excitation and response signals (v_{in} and v_{out} = v_C = v_m), a biasing resistor R_0, and capacitor C_0, as depicted in Figure 4.

Frequency-domain analysis determines C_0. Substituting $s = i\omega$ into the impedance function Z (s, Q) in Equation 4 yields

$$\begin{aligned} Z(i\omega,Q) &= \mathrm{Re}Z(i\omega,Q)+\mathrm{iIm}Z(i\omega,Q) \\ &= \left(R_M+\frac{R_s}{1+{R_s}^2{C_s}^2\omega^2}\right)+\mathrm{i}\frac{-{R_s}^2C_s\omega}{1+{R_s}^2{C_s}^2\omega^2} \end{aligned} \tag{5}$$

The resonant frequency ω_0 occurs when Re [Z ($i\omega$, Q)] = 0. From Equation 5, the corresponding imaginary part can be calculated. For oscillation initiation, the critical capacitance satisfies:

$$C_0 = \frac{1}{\omega_0 \mathrm{Im}(i\omega_0, Q)} = -\frac{R_s C_s}{R_s + R_M} \tag{6}$$

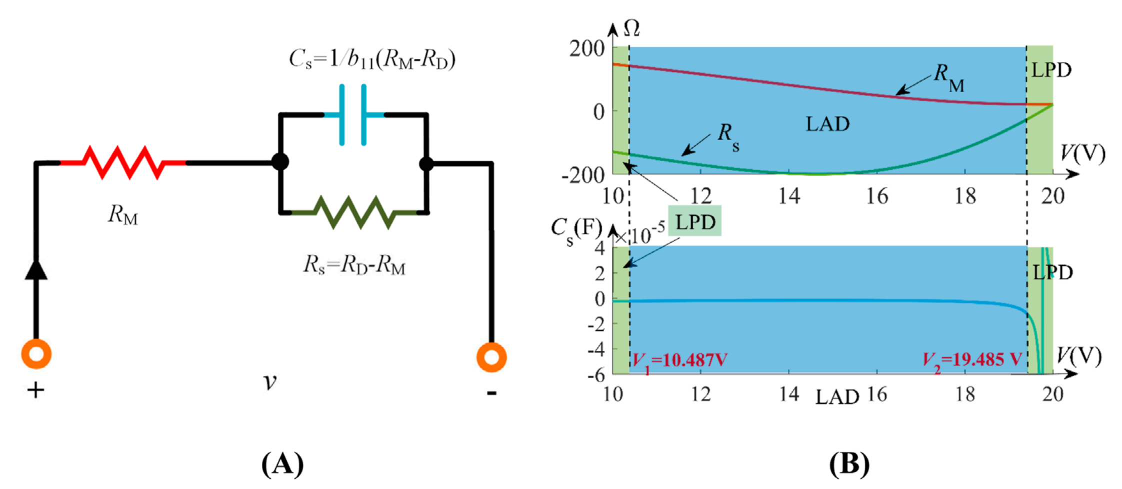


FIGURE 3
Small-signal analysis of the LAM: **(A)** the small-signal equivalent circuit of the LAM; **(B)** the values of R_M, C_s, and R_s varying with operating voltages over the range of $V \in [10\text{ V}, 20\text{ V}]$.

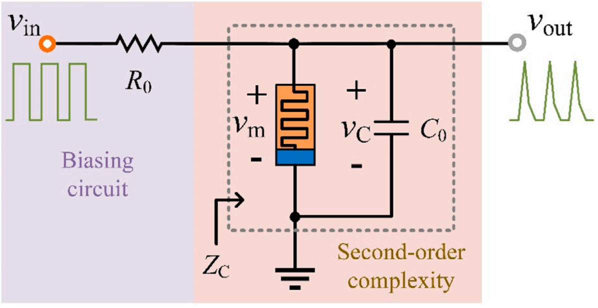


FIGURE 4
LAM-based second-order neuronal circuit.

3.1 Composite impedance function

The oscillation condition for the composite neuronal circuit is derived from its impedance function:

$$Z_C(s,Q) = \frac{1}{\frac{1}{Z(s,Q)} + C_0 s} = \frac{R_M s - R_D b_{11}}{R_M C_0 s^2 + (1 - R_D C_0 b_{11})s - b_{11}} \tag{7a}$$

with two poles

$$\begin{cases} p_1 = \dfrac{R_D C_0 b_{11} - 1 + \sqrt{(1 - R_D C_0 b_{11})^2 + 4 R_M C_0 b_{11}}}{2 R_M C_0} \\ p_2 = \dfrac{R_D C_0 b_{11} - 1 - \sqrt{(1 - R_D C_0 b_{11})^2 + 4 R_M C_0 b_{11}}}{2 R_M C_0} \end{cases} \tag{7b}$$

Figure 5A maps three operational domains: Locally Passive Domains (LPD, yellow), Unstable Locally Active Domains (RHP, right-half plane, cyan), Stable Locally Active Domains (EOC, edge of chaos, green) based on Equations 6, 7a, 7b. These domains align with the memristor's LAD and LPD characteristics. RHP requires simultaneous local activity and instability, while EOC demands local activity with asymptotic stability.

The pole evolution analysis in Equation 7b under bias voltages $v_{in} \in [-20.5\text{ V}, -9.5\text{ V}] \cup [9.5\text{ V}, 20.5\text{ V}]$ and $C = 10\,\mu\text{F}$ reveals dynamic stability transitions, as depicted in Figure 5B. Red and blue curves represent the trajectories of p_1 and p_2, respectively, with oscillation occurring when $v_{in} \in [-19.03\text{ V}, -11.4\text{ V}] \cup [11.4\text{ V}, 19.03\text{ V}]$ (orange region). In this region, at least one pole is in the right-half plane (RHP). However, stability persists when $p_{1,2} \in$ LHP (left-half plane). Particularly, Hopf bifurcation emerges at $v_{in} = \pm 11.4$ V and ± 19.03 V, characterized by conjugate complex pole pairs.

3.2 Periodic spikes

The state equations of the second-order neuronal circuit in Figure 4 are governed by

$$\begin{cases} \dfrac{dx}{dt} = \delta_0 + \alpha_1 x + \beta_2 \left(\dfrac{v_m}{d_2 x^2 + d_0} \right)^2 \\ \dfrac{dv_m}{dt} = \dfrac{1}{C_0} \left(\dfrac{v_{in} - v_m}{R_0} - \dfrac{v_m}{d_2 x^2 + d_0} \right) \end{cases} \tag{8}$$

where x and v_m represent memristor state and membrane potential, respectively.

With $C_0 = 10\,\mu\text{F}$ and initial condition $[x(0), v_m(0)] = (0, 0)$, distinct neuromorphic behaviors emerge under varying stimuli v_{in} according to Equation 8. For stimuli $v_{in} = 9.5$ V (LPD) and 10.8 V (EOC), the trajectories converge from the initial point (0, 0) into $(v_C, x) = (1.27, 8.19)$ and $(1.29, 7.58)$, respectively, maintaining resting states as shown in Figure 6A. Increasing v_{in} to 18 V (see RHP domain in Figure 5A) triggers sustained periodic spikes with frequency $f = 204$ Hz, demonstrated by time-domain waveform of v_C and limit cycles in the x-v_{out} phase portrait (Figure 6B). We conclude that the neuron maintains quiescence when operating in the LPD or EOC domains, while inducing spikes under locally

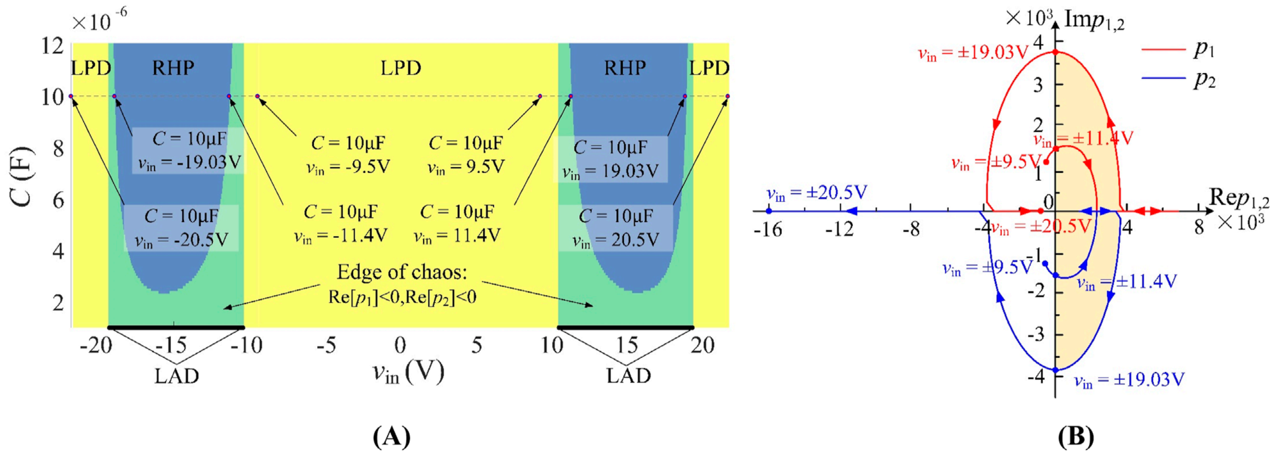


FIGURE 5
(A) The parameter distributions of LAD, LPD, and EOC on V_{in}-C plane; **(B)** poles trajectories with the change of voltage V_{in} at C = 10 μF.

active operating points. Notably, spiking frequency modulation under varying v_{in} = 12 V, 14.5 V, 16.5 V, and 18.5 V replicates biological neural encoding mechanisms (Figure 6C). Besides, the neuron emulates excitatory and inhibitory response transitions, as illustrated in Figure 6D.

4 LAM-based third-order neuron

Second-order neurons cannot simulate complex neurodynamics such as chaos and bursting, then we construct a memristive neuron with third-order complexity, as shown in Figure 7, including an LAM, a capacitor, an inductor, a resistor, and a voltage source.

4.1 Stability condition

The impedance function Z_T (s, Q) of third-order memristive neuron circuit is written as:

$$Z_T(s,Q)=\frac{1}{1/(Z(s,Q)+sL)+sC}=\frac{Ls^2+(R_M-b_{11}L)s-b_{11}R_D}{LCs^3+(R_M-Lb_{11})Cs^2+(1-CR_Db_{11})s-b_{11}} \tag{9a}$$

whose three poles are

$$\begin{cases} p_1=-\dfrac{R_M-b_{11}L}{3L}+\sqrt[3]{-\dfrac{q}{2}+\sqrt{\Delta}}+\sqrt[3]{-\dfrac{q}{2}-\sqrt{\Delta}} \\ p_2=-\dfrac{R_M-b_{11}L}{3L}+\left(\dfrac{-1+\sqrt{3}\mathrm{i}}{2}\right)\sqrt[3]{-\dfrac{q}{2}+\sqrt{\Delta}}+\left(\dfrac{-1+\sqrt{3}\mathrm{i}}{2}\right)^2\sqrt[3]{-\dfrac{q}{2}-\sqrt{\Delta}} \\ p_3=-\dfrac{R_M-b_{11}L}{3L}+\left(\dfrac{-1+\sqrt{3}\mathrm{i}}{2}\right)^2\sqrt[3]{-\dfrac{q}{2}+\sqrt{\Delta}}+\omega\sqrt[3]{-\dfrac{q}{2}-\sqrt{\Delta}} \end{cases} \tag{9b}$$

where $\Delta=\left(\frac{(R_M-b_{11}L)\cdot(1-CR_Db_{11})}{6L^2C}-\frac{(R_M-b_{11}L)^3}{27L^3}+\frac{b_{11}}{2LC}\right)^2+\left(\frac{1-CR_Db_{11}}{3LC}-\frac{(R_M-b_{11}L)^2}{9L^2}\right)^3$, $q=-\frac{(R_M-b_{11}L)(1-CR_Db_{11})}{3L^2C}+\frac{2(R_M-b_{11}L)^3}{27L^3}-\frac{b_{11}}{LC}$.

Based on Equation 9, the trajectory diagram of poles $p_{1,2,3}$ within the range of 8.2 V ≤ V_{in} ≤ 20 V is depicted in Figure 8A, where blue, red, and yellow curves correspond to the trajectories of p_1, p_2, and p_3, respectively, with arrows denoting directionality as v_{in} increases. Oscillatory behavior occurs when Re $p > 0$, particularly within $v_{in} \in$ [11.12 V, 19.38 V] where Hopf bifurcation emerges at v_{in} = 11.12 V and v_{in} = 19.38 V, characterized by conjugate complex poles (Im p = 0). For $v_{in} \le$ 11.12 V, all poles reside in the left-half plane (LHP), driving the circuit to stable equilibrium. Conversely, right-half plane (RHP) poles dominate in the oscillatory regime, enabling sustained dynamics. Figures 8B,C confirm this operational range through Lyapunov exponent and bifurcation diagram analysis, demonstrating consistent periodic and chaotic domains in this range under L = 20 mH and C = 10 μF, where the chaotic ranges are $v_{in} \in$ [−18.98 V, −18.89 V] ∪[18.89 V, 18.98 V].

4.2 Monophasic neurodynamics

The third-order LAM-based neuronal circuit in Figure 7 is described by

$$\begin{cases} \dfrac{dx}{dt}=\delta_0+\alpha_1x+\beta_2{i_L}^2 \\ \dfrac{dv_C}{dt}=\dfrac{1}{C}\left(\dfrac{v_{in}-v_{out}}{R_0}-i_L\right) \\ \dfrac{di_L}{dt}=\dfrac{1}{L}\left(v_{out}-(d_2x^2+d_0)i_L\right) \end{cases} \tag{10}$$

where x (memristor state), i_L (inductor current), and v_C (output voltage) define the neuron dynamics.

Under L = 20 mH and C = 10 μF, six monophasic neuromorphic behaviors emerge through parametric control of v_{in} based on Equation 10. At v_{in} = 19.3 V (RHP domain), subthreshold oscillations occur (Figure 9A). Reducing v_{in} to 18.5 V and 18.9 V within the RHP domain induces periodic spikes (Figure 9B) and chaotic dynamics (Figure 9C), respectively. Time-varying stimulation v_{in} = 9.9 t V (t ∈[1.2 s, 2.2 s]) triggers Class II excitability, maintaining constant spiking frequency despite voltage modulation (Figure 9D). For periodic square-wave inputs (T = 0.0625 s, A = 15 V), the neuron exhibits bursting patterns

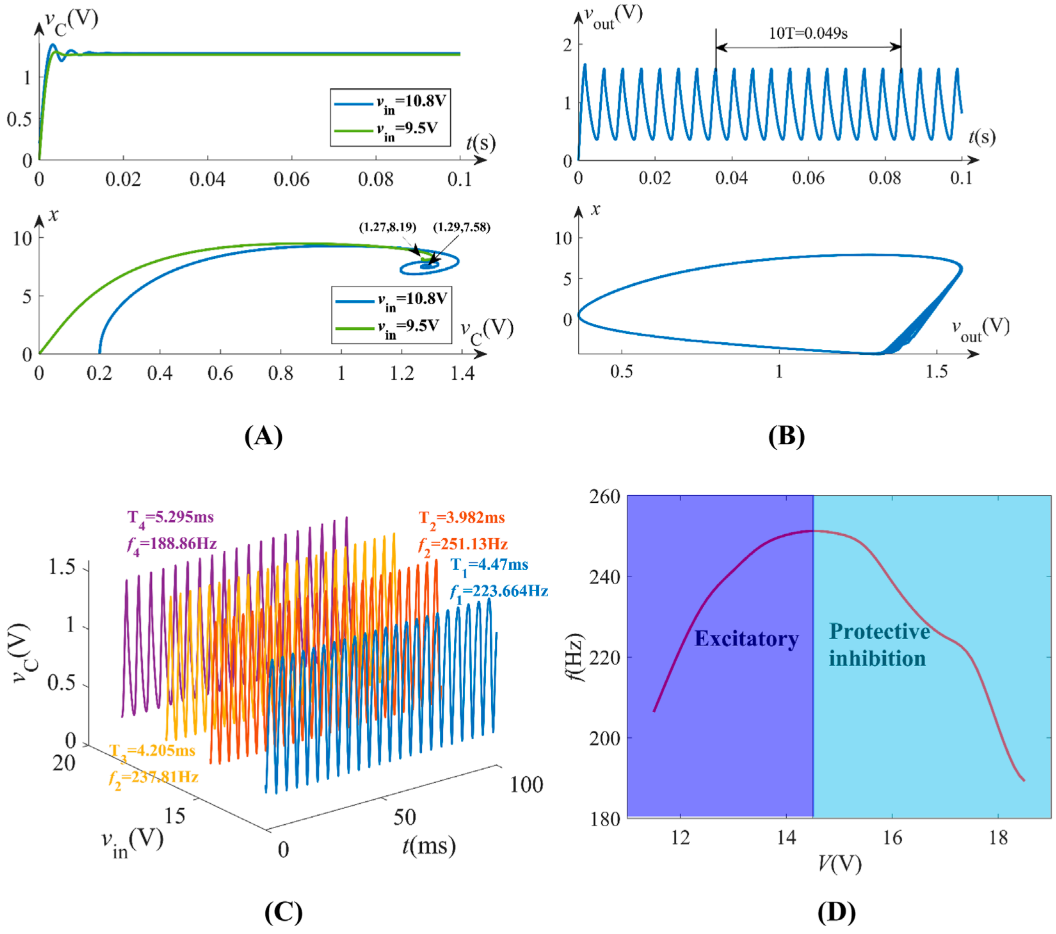


FIGURE 6
Neuromorphic behaviors under some typical voltages: **(A)** v_{in} = 9.5 V, 10.8 V, resting states; **(B)** v_{in} = 18 V, periodic spikes; **(C)** spiking frequency modulation; **(D)** excitatory and protective inhibition behaviors.

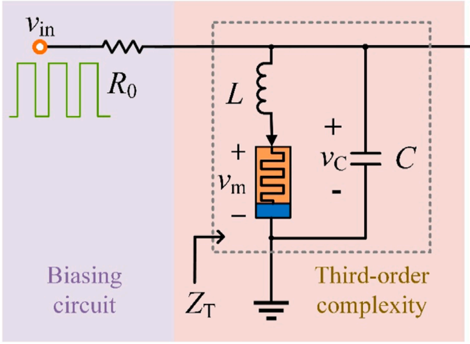


FIGURE 7
The circuit schematic of the third-order memristive neuron model.

(Figure 9E). Besides, depolarizing after-potentials emerge under the parameter set of C = 0.5 μF, L = 20 mH and v_{in} = 15 V, mimicking post-spike membrane potential modulation (Figure 9F).

These results demonstrate voltage-controlled emulation of biological neuronal encoding.

4.3 Biphasic spikes

The neuronal circuit in Figure 7 generates biphasic action potentials when driven by bipolar square-wave inputs (v_{in} = 16 V, D = 50%). As shown in Figures 10A a T= 10 ms periodic stimulus (blue waveform) induces single-cycle bidirectional spiking, characterized by counterphase positive and negative pulses in the inductor current i_L (red waveform). When we increase the period T of the input periodic square wave to 22.22 ms, 33.33 ms, 43.48 ms, 55.56 ms and 66.67 ms, the output waveform changes into two spikes, three spikes, four spikes, five spikes and six spikes in the upward direction and down direction in one period, as shown in Figures 10B–F.

4.4 Symmetric behaviors

The third-order memristive neuron demonstrates voltage-polarity-dependent symmetry in neurodynamic behaviors,

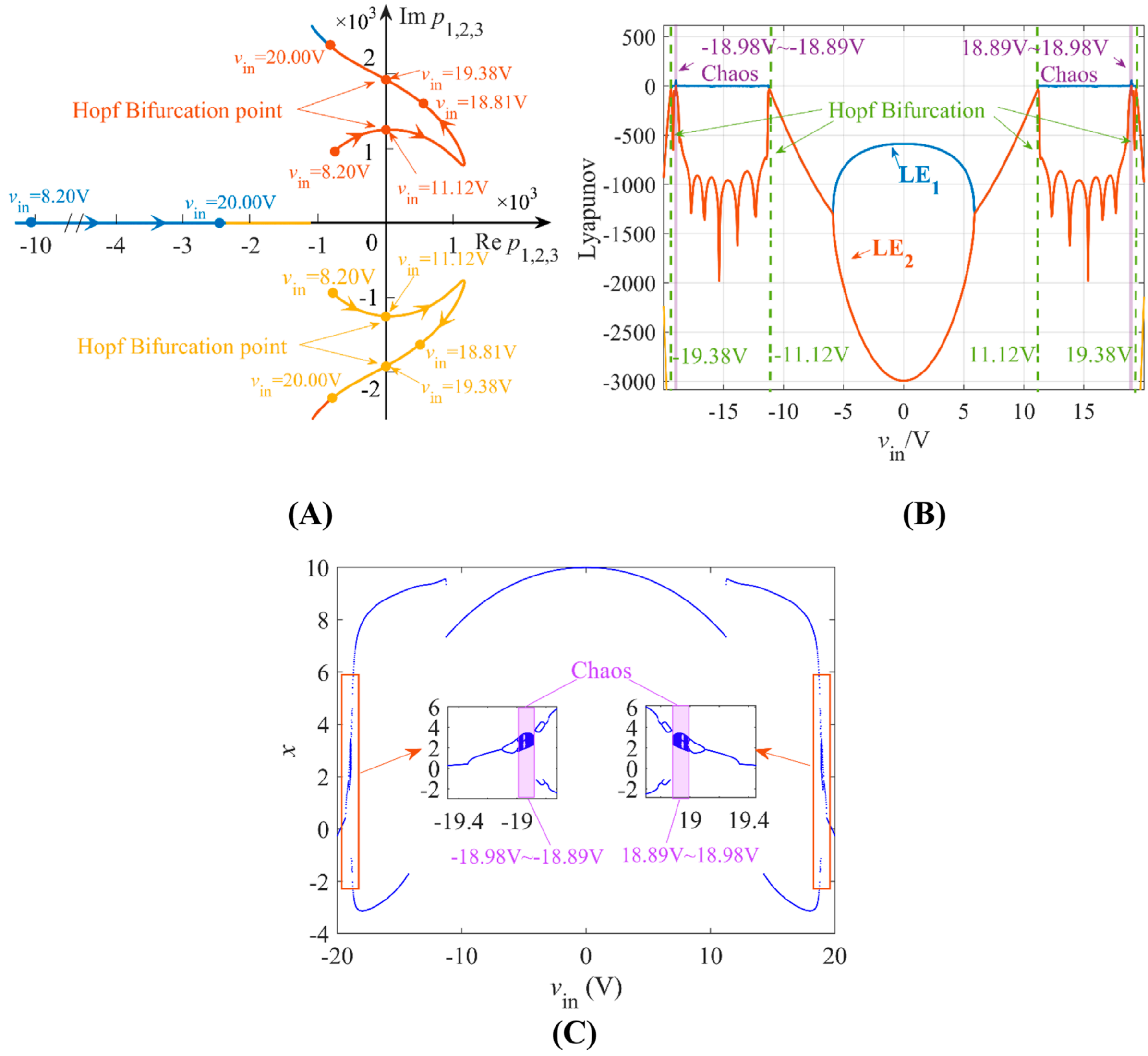


FIGURE 8
(A) Poles diagram with respect to biasing voltages v_{in} when L = 20 mH, C = 10 μF; **(B)** Lyapunov exponents under L = 20 mH, C = 10 μF; **(C)** bifurcation diagram with respect to v_{in} under L = 20 mH, C = 10 μF.

originating from the voltage symmetry in Figure 5A. This nonlinear symmetry allows symmetrical action potential generation: positive DC voltages ($v_{in} > 0$) induce upward-polarized spikes, while negative inputs ($v_{in} < 0$) produce downward-polarized counterparts, as depicted in Figure 11.

Under voltage excitation v_{in} = ±18.5 V, the inductor current i_L exhibits mirror-symmetric periodic spiking, i.e., upward polarization for positive bias (orange curve) versus downward polarization for negative bias (blue curve) in Figure 11A. Voltage modulation to ±18.9 V induces symmetrical chaotic dynamics with identical Lyapunov exponents but opposing phase-space trajectories, as shown in Figure 11B. Transient behavior analysis reveals bidirectional spike initiation: v_{in} = 19.4 V triggers upward spikes while v_{in} = −19.4 V generates downward equivalents, both returning to symmetrical resting potentials after undergoing 5 m (Figure 11C). The corresponding phase portraits of these three nonlinear behaviors are depicted in Figures 11D–F.

5 Circuit emulator

The circuit emulator of the memristive neuron with third-order complexity is constructed, as shown in Figure 12, which consists of two operational amplifiers (U1A and U1B), three analog multipliers (U1, U2, and U3), two capacitors (C_0 and C_1), one inductor L, and some resistors.

As shown in Figure 12, the equivalent circuit of the locally active memristor comprises four functionally integrated modules: (1) Current sensing module ① monitors emulator input current in real-time, generating proportional output v_i; (2) analog multiplier arrays ② and ④ implement nonlinear term computations; (3)

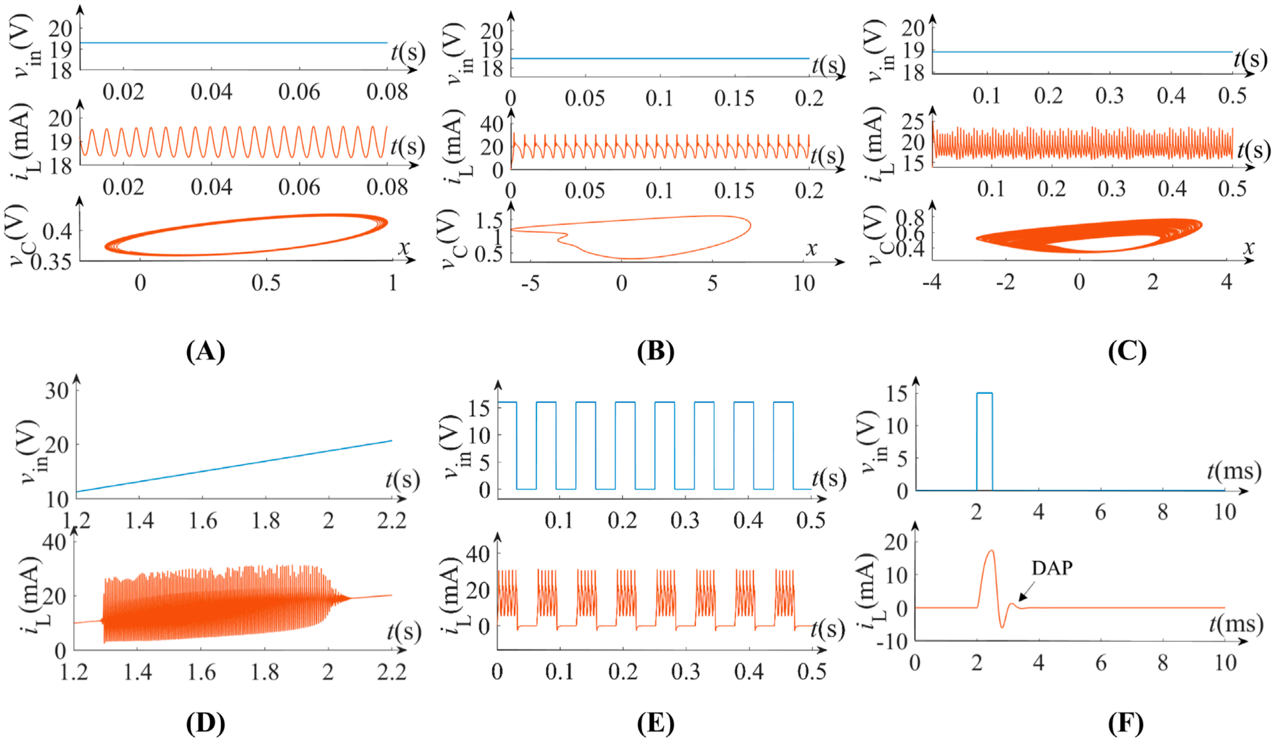


FIGURE 9
Monophasic neurodynamics under different input voltage v_{in} with C = 10 μF and L = 20 mH: **(A)** subthreshold oscillation; **(B)** periodic spiking; **(C)** chaos; **(D)** Class II excitability; **(E)** periodic bursting. **(F)** depolarizing after-potential with C = 0.5 μF, L = 20 mH and v_{in} = 15 V.

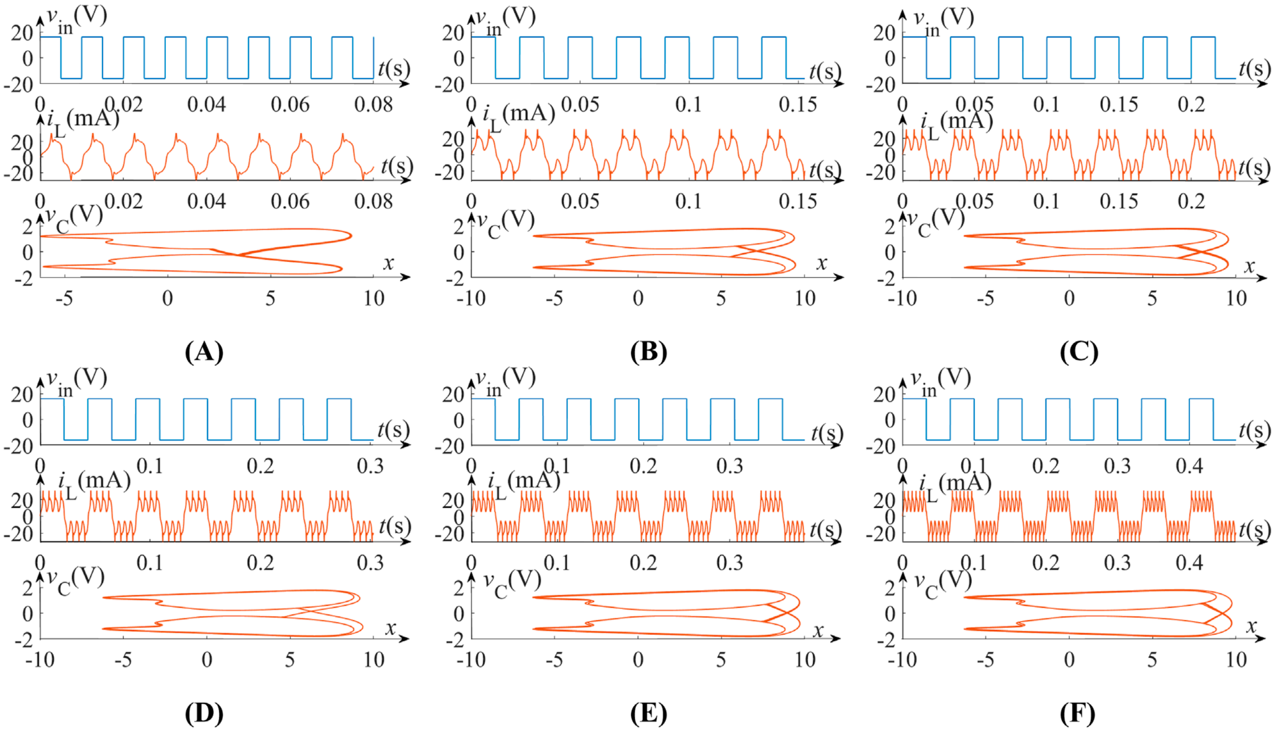


FIGURE 10
Biphasic action potentials generated by the neuron circuit, when driven by a bipolar periodic square wave with the amplitude v_{in} = 16 V, duty cycles D = 50% and various period T. **(A)** T = 10.00 ms; **(B)** T = 22.22 ms; **(C)** T = 33.33 ms; **(D)** T = 43.48 ms; **(E)** T = 55.56 ms; **(F)** T = 66.67 ms.

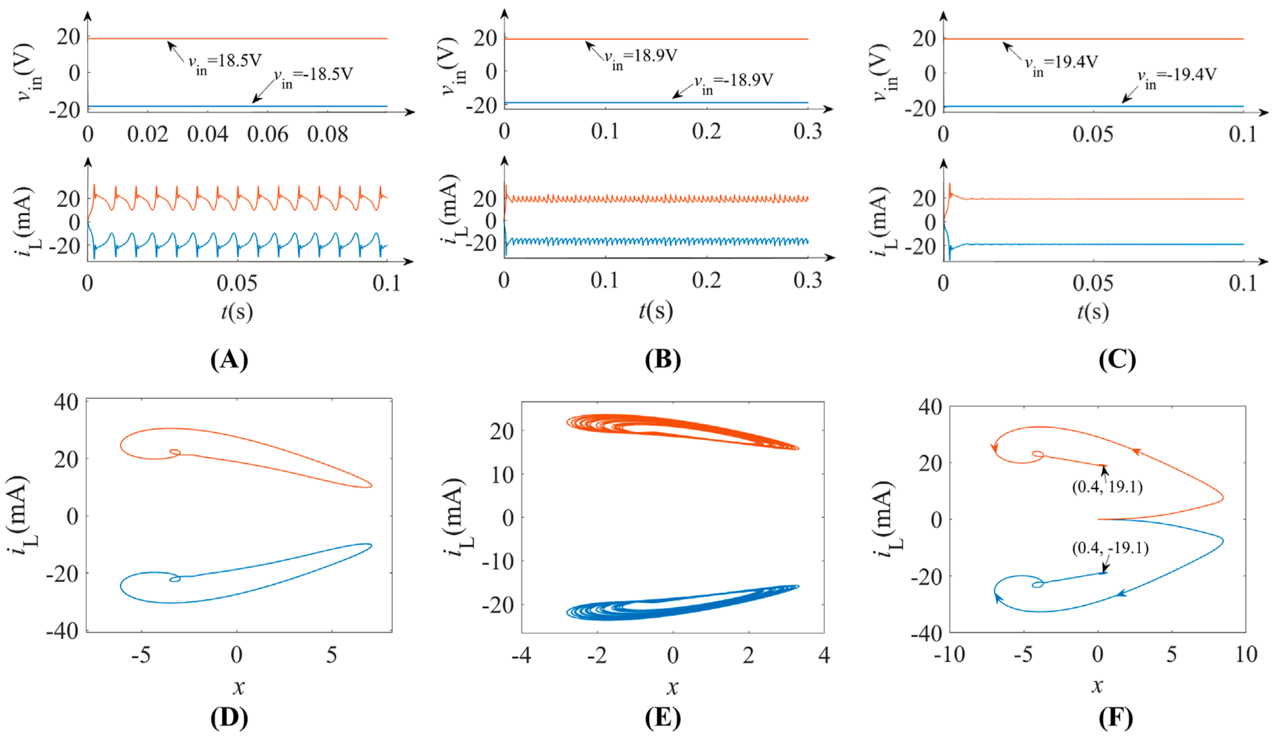


FIGURE 11
Voltage-polarity-modulated symmetric behaviors: **(A)** periodic spikes; **(B)** chaos; **(C)** resting states; **(D)** phase portraits of periodic spikes; **(E)** phase portraits of chaos; **(F)** phase portraits of resting states.

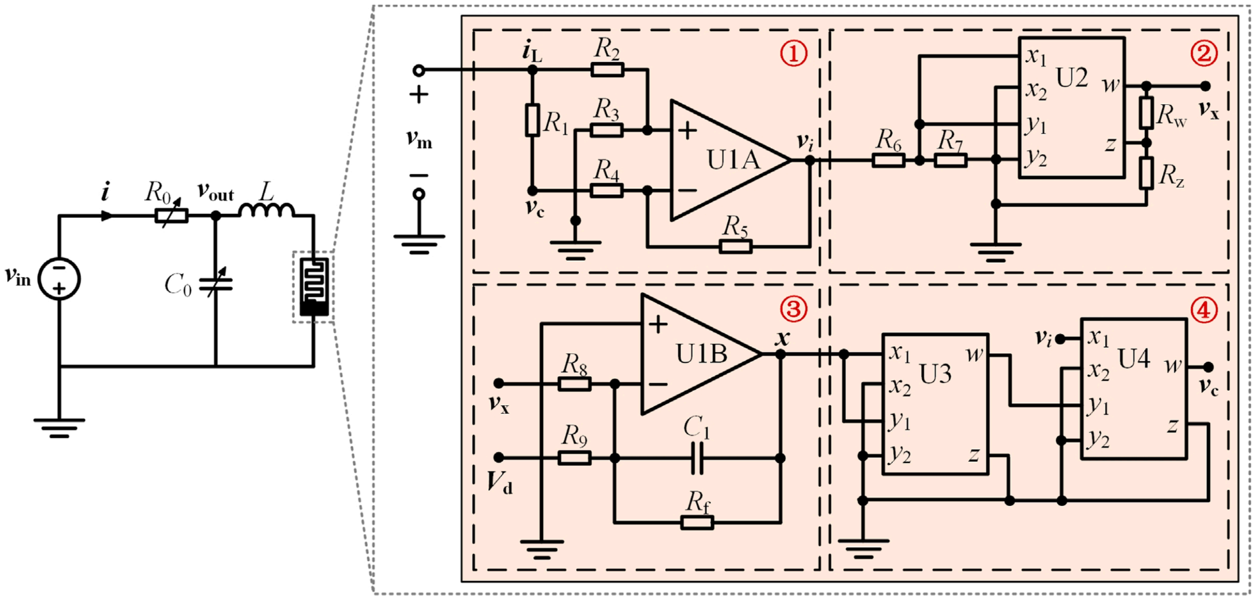


FIGURE 12
Circuit emulator schematic of the third-order memristive neuron.

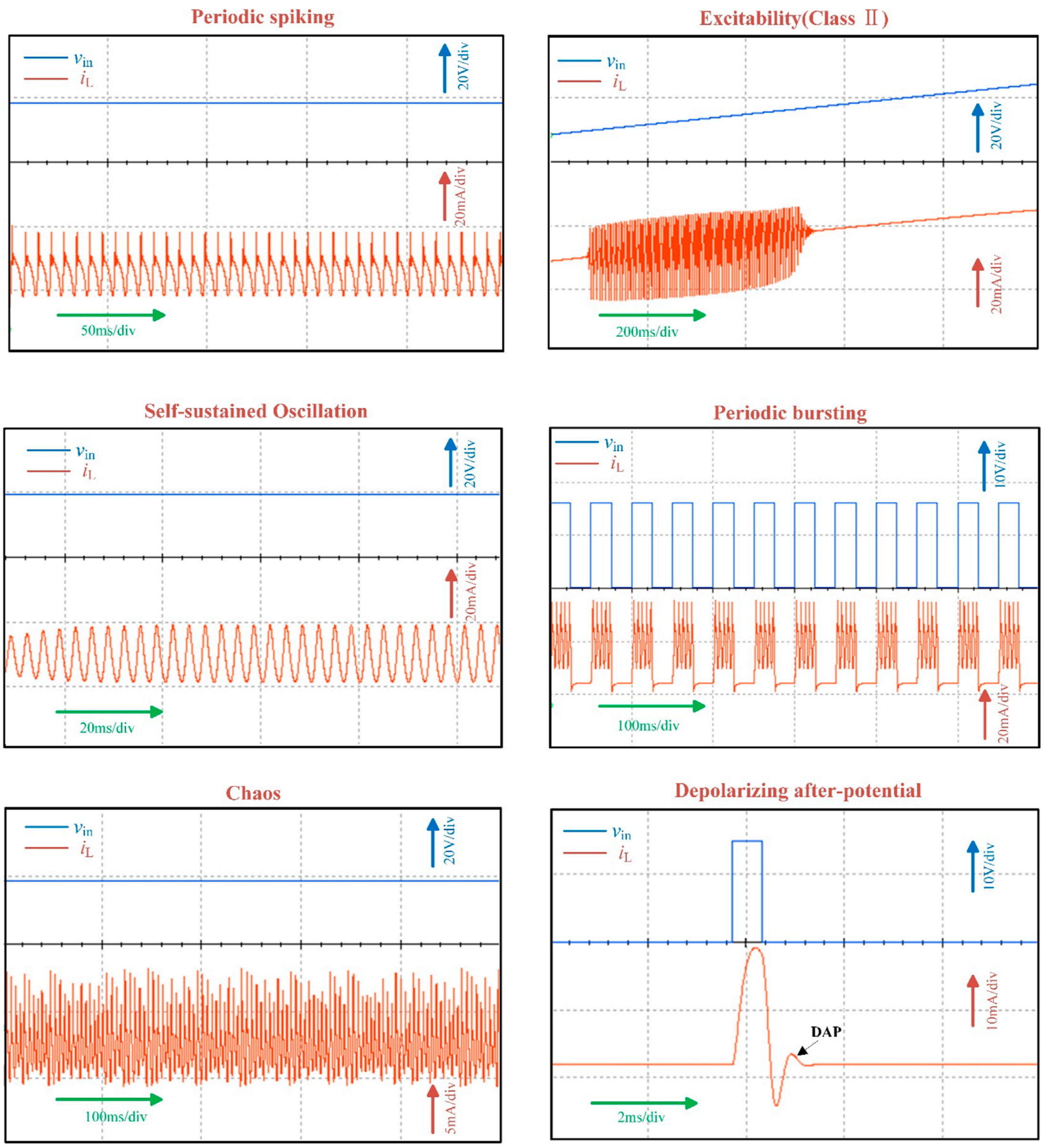


FIGURE 13
Circuit simulated neuromorphic dynamics under various biasing voltages.

State equation solver ③ converts DC bias V_d into memristor state variable x through differential integration, that is, $v_x = x$; (4) Feedback integration completes the loop via R_1. Kirchhoff's voltage and current laws govern this circuit architecture, yielding three coupled differential equations that mathematically describe electrophysiological dynamics of the memristive neuron, as

$$\begin{cases} C_1 \dfrac{dx}{dt} = -\left(\dfrac{V_d}{R_9} + \dfrac{x}{R_f} + \dfrac{R_w + R_z}{10R_w R_8} \left(\dfrac{R_7 R_5 R_1}{R_4 (R_6 + R_7)} i_L \right)^2 \right) \\ C_0 \dfrac{dv_C}{dt} = \dfrac{v_{in} - v_C}{R_0} - i_L \\ L \dfrac{di_L}{dt} = v_C - \left(\dfrac{R_1 R_5 R_{11}}{10 R_4 R_{10}} x^2 + R_1 \right) i_L \end{cases} \tag{11}$$

where the circuit parameters are $R_1 = 20\ \Omega$, $R_0 = R_2 = R_4 = R_6 = R_7 = R_w = 1\ \text{k}\Omega$, $R_3 = R_5 = R_8 = R_9 = 10\ \text{k}\Omega$, $R_z = 7\ \text{k}\Omega$, $R_f = 33.3\ \text{k}\Omega$, $C_1 = 10$ nF, $C_0 = 9.5\ \mu$F, $L = 20$ mH, and $V_d = -3$ V.

The circuit simulated results calculated via Equation 11 are shown in Figure 13, reproducing key neurodynamics including periodic spikes, class II excitability, self-sustained oscillations, bursting, chaos, and depolarizing after-potential. These results demonstrate quantitative agreement with theoretical predictions.

6 Conclusion

This work constructs neuronal circuits leveraging a bi-S-type locally active memristor that amplifies weak signals through intrinsic local activity. The designed second-order circuit achieves voltage-modulated periodic spiking and adaptive inhibition, while the third-order extension emulates biological neural dynamics including monophasic and biphasic action potentials, chaos, and bursting, which are driven by memristive symmetry. The study of memristive neurons not only offers essential building blocks for neuromorphic computing architectures but also lays a theoretical reference for the development of more advanced and bio-realistic neural processing systems.

Data availability statement

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

Author contributions

XW: Software, Writing – original draft, Formal Analysis, Validation. YD: Methodology, Writing – review and editing, Validation, Software, Funding acquisition. GW: Writing – review and editing, Supervision, Formal Analysis. ZZ: Writing – original draft, Formal Analysis, Software. YM: Writing – review and editing, Visualization. PJ: Investigation, Resources, Writing – review and editing. YL: Formal Analysis, Writing – review and editing, Funding acquisition.

Funding

The author(s) declare that financial support was received for the research and/or publication of this article. This work was supported in part by the Zhejiang Provincial Natural Science Foundation of China under Grant LQ23F010018 and Y24F010007; in part by the National Natural Science Foundation of China under Grant 62301202 and 62171173.

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.

Generative AI statement

The author(s) declare that no Generative AI was used in the creation of this manuscript.

Publisher's note

References

1. Kendall JD, Kumar S. The building blocks of a brain-inspired computer. *Appl Phys Rev* (2020) 7(1):011305. doi:10.1063/1.5129306

2. Yuan R, Tiw PJ, Cai L, Yang ZY, Liu C, Zhang T, et al. A neuromorphic physiological signal processing system based on VO$_2$ memristor for next-generation human-machine interface. *Nat Commun* (2023) 14(1):3695. doi:10.1038/s41467-023-39430-4

3. Park SO, Jeong H, Park J, Bae J, Choi S. Experimental demonstration of highly reliable dynamic memristor for artificial neuron and neuromorphic computing. *Nat Commun* (2022) 13(1):2888. doi:10.1038/s41467-022-30539-6

4. Pei YF, Yang B, Zhang XM, He H, Sun Y, Zhao JH, et al. Ultra robust negative differential resistance memristor for hardware neuron circuit implementation. *Nat Commun* (2025) 16(1):48. doi:10.1038/s41467-024-55293-9

5. Markovic D, Mizrahi A, Querlioz D, Grollier J. Physics for neuromorphic computing. *Nat Rev Phys* (2020) 2(9):499–510. doi:10.1038/s42254-020-0208-2

6. Schuman CD, Kulkarni SR, Parsa M, Mitchell JP, Date P, Kay B. Opportunities for neuromorphic computing algorithms and applications. *Nat Comput Sci* (2022) 2(1):10–9. doi:10.1038/s43588-021-00184-y

7. Sun B, Guo T, Zhou GD, Ranjan S, Jiao YX, Wei L, et al. Synaptic devices based neuromorphic computing applications in artificial intelligence. *Mater Today Phys* (2021) 18:100393. doi:10.1016/j.mtphys.2021.100393

8. Kumar S, Williams RS, Wang ZW. Publisher Correction: third-order nanocircuit elements for neuromorphic engineering. *Nature* (2020) 592(7853):E9. doi:10.1038/s41586-021-03349-x

9. Yi W, Tsang KK, Lam SK, Bai XW, Crowell JA, Flores EA. Biological plausibility and stochasticity in scalable VO$_2$ active memristor neurons. *Nat Commun* (2018) 9:4661. doi:10.1038/s41467-018-07052-w

10. Wu XY, Saxena V, Zhu KH, Balagopal S. A CMOS spiking neuron for brain-inspired neural networks with resistive synapses and *in situ* learning. *IEEE Trans Circuits Syst Exp Briefs* (2015) 62(11):1088–92. doi:10.1109/TCSII.2015.2456372

11. Frenkel C, Lefebvre M, Legat JD, Bol D. A 0.086-mm^2 12.7-pJ/SOP 64k-synapse 256-neuron online-learning digital spiking neuromorphic processor in 28-nm CMOS. *IEEE Trans Biomed Circuits Syst* (2019) 13(1):145–58. doi:10.1109/TBCAS.2018.2880425

12. Natarajan A, Hasler J. Hodgkin-Huxley neuron and FPAA dynamics. *IEEE Trans Biomed Circuits Syst* (2018) 12(4):918–26. doi:10.1109/TBCAS.2018.2837055

13. Strukov DB, Snider GS, Stewart DR, Williams RS. The missing memristor found. *Nature* (2008) 453(7250):80–3. doi:10.1038/nature08166

14. Li YX, Wang SQ, Yang K, Yang YC, Sun Z. An emergent attractor network in a passive resistive switching circuit. *Nat Commun* (2024) 15(1):7683. doi:10.1038/s41467-024-52132-9

15. Xie Q, Pan XQ, Wang Y, Luo WB, Zhao ZB, Tong JD, et al. Passive LiNbO$_3$ memristor with multilevel states for neuromorphic computing. *IEEE Trans Electron Devices* (2024) 71(10):6049–54. doi:10.1109/TED.2024.3450437

16. Jin P, Wang G, Liang Y, Lu HHC, Chua LO. Neuromorphic dynamics of Chua corsage memristor. *IEEE Trans Circuits Syst Reg Pap* (2021) 68(11):4419–32. doi:10.1109/tcsi.2021.3121676

17. Qin MH, Lai Q, Fortuna L, Zhao XW. Rich dynamics induced by memristive synapse in Chialvo neuron network. *Nonlinear Dyn* (2025) 113(13):17095–109. doi:10.1007/s11071-025-10996-6

18. Kumar S, Strachan JP, Williams RS. Chaotic dynamics in nanoscale NbO$_2$ Mott memristors for analogue computing. *Nature* (2017) 548(7667):318–21. doi:10.1038/nature23307

19. Tang D, Wang CH, Lin HR, Yu F. Dynamics analysis and hardware implementation of multi-scroll hyperchaotic hidden attractors based on locally active memristive Hopfield neural network. *Nonlinear Dyn* (2024) 112(2):1511–27. doi:10.1007/s11071-023-09128-9

20. Shen H, Yu F, Kong XX, Mokbel AAM, Wang CH, Cai S. Dynamics study on the effect of memristive autapse distribution on Hopfield neural network. *Chaos* (2022) 32(8):083133. doi:10.1063/5.0099466

21. Xu Q, Fang YJ, Wu HG, Bao H, Wang N. Firing patterns and fast-slow dynamics in an N-type LAM-based FitzHugh-Nagumo circuit. *Chaos Solitons and Fractals* (2024) 187:115376. doi:10.1016/j.chaos.2024.115376

22. Xu Q, Wang YT, Lu HHC, Wang N, Bao H. Locally active memristor-based neuromorphic circuit: firing pattern and hardware experiment. *IEEE Trans Circuits Syst Reg Pap* (2023) 70(8):3130–41. doi:10.1109/TCSI.2023.3276983

23. Yao L, Yu HM, Xiong QP, Sun W, Xu Y, Meng W, et al. Cordycepin decreases compound action potential conduction of frog sciatic nerve *in vitro* involving Ca^{2+}-dependent mechanisms. *Neural Plast* (2015) 2015:1–7. doi:10.1155/2015/927817

24. Dumitru D, King JC, van der Rijt W, Stegeman DF. The biphasic morphology of voluntary and spontaneous single muscle fiber action potentials. *Muscle Nerve* (1994) 17:1301–7. doi:10.1002/mus.880171109

25. Someck S, Levi A, Sloin HE, Spivak L, Gattegno R, Stark E. Positive and biphasic extracellular waveforms correspond to return currents and axonal spikes. *Commun Biol* (2023) 6:950. doi:10.1038/s42003-023-05328-6

26. Watanabe M, Otani NF, Gilmour RF, Jr. Biphasic restitution of action potential duration and complex dynamics in ventricular myocardium. *Circ Res* (1995) 76:915–21. doi:10.1161/01.res.76.5.915

27. Chen P, Xiong X, Zhang B, Ye Y, Pan G, Lin P. Neuromorphic auditory classification based on a single dynamical electrochemical memristor. *Neuromorphic Comput Eng* (2024) 4:014012. doi:10.1088/2634-4386/ad33cc

28. Yu F, He S, Yao W, Cai S, Xu Q. Quantitative characterization system for macroecosystem attributes and states. *IEEE Trans Comput-aided Des Integr Circuits Syst* (2025) 36:1–12. doi:10.13287/j.1001-9332.202501.031

29. Yu F, Dan S, He S, Wu Y, Zhang S, Yin H. Resonant tunneling diode cellular neural network with memristor coupling and its application in police forensic digital image protection. *Chin Phys B* (2025) 34:050502. doi:10.1088/1674-1056/adb8bb

30. Lai Q, Guo S. Heterogeneous coexisting attractors, large-scale amplitude control and finite-time synchronization of central cyclic memristive neural networks. *Neural Networks* (2024) 178:106412. doi:10.1016/j.neunet.2024.106412

31. Wu Y, Li S, Ji Y, Weng Z, Xing H, Arauz L, et al. Enhancing memristor performance with 2D SnOx/SnS2 heterostructure for neuromorphic computing. *Sci China Mater* (2025) 68:581–9. doi:10.1007/s40843-024-3208-3

32. Chua L. Resistance switching memories are memristors. *Appl Phys A, Sol Surf.* (2011) 102(4):765–83. doi:10.1007/s00339-011-6264-9

33. Chua L. If it's pinched it's a memristor. *Semicond Sci Technol* (2013) 29(10):104001. doi:10.1088/0268-1242/29/10/104001

34. Chua L. Everything you wish to know about memristors but are afraid to ask. *Radioengineering* (2015) 24(2):319–68. doi:10.13164/re.2015.0319

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 05 June 2025
DOI 10.3389/fphy.2025.1617964

Check for updates

OPEN ACCESS

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

REVIEWED BY
Suo Gao,
Harbin Institute of Technology, China
Xinlei An,
Lanzhou Jiaotong University, China

*CORRESPONDENCE
Jie Jin,
✉ jj67123@hnust.edu.cn

RECEIVED 25 April 2025
ACCEPTED 12 May 2025
PUBLISHED 05 June 2025

CITATION
Yu F, Wu Y, Wang X, He T, Zhang S and Jin J (2025) New discrete memristive hyperchaotic map: modeling, dynamic analysis, and application in image encryption.
Front. Phys. 13:1617964.
doi: 10.3389/fphy.2025.1617964

New discrete memristive hyperchaotic map: modeling, dynamic analysis, and application in image encryption

Fei Yu[1], Yiya Wu[1], Xuqi Wang[1], Ting He[1], ShanKou Zhang[1] and Jie Jin[2]*

[1]School of Computer Science and Technology, Changsha University of Science and Technology, Changsha, China, [2]School of Information Engineering, Changsha Medical University, Changsha, China

With the rapid development of information technology, the demand for ensuring data security and privacy protection has become increasingly urgent. The purpose of this study is to address the limitations of existing image encryption methods and develop a more secure and efficient image encryption scheme. To achieve this, we adopt a research method that involves constructing a new type of discrete memristor hyperchaotic map by coupling an upgraded cosine discrete memristor with the Cubic map, and then conducting in-depth analysis of the system's dynamic characteristics using phase diagrams, Lyapunov exponential spectra, and bifurcation diagrams to confirm its ability to reach a hyperchaotic state. Based on this hyperchaotic map, we propose a new image encryption scheme, generating high-quality chaotic sequences through its excellent chaotic characteristics to effectively scramble and diffuse image data, and also introducing a novel forward and reverse diffusion strategy in the diffusion process to enhance encryption efficiency. Through experiments on various images, we verify the algorithm's effectiveness in improving encryption strength, reducing information leakage risks, and ensuring data security. Finally, the results of keyspace analysis, histogram analysis, correlation analysis, and information entropy demonstrate that the scheme has high security and practicability, along with good application prospects and practical value.

KEYWORDS

discrete memristors, hyperchaotic map, dynamical analysis, image encryption, data security

1 Introduction

Chaos is a non-linear kinematic system widely used in the biological and social sciences of nature [1–5]. The application of chaotic systems due to their randomness, unpredictability, and initial state sensitivity brings many advantages [6–10], and hyperchaotic systems further extend this complexity [11–14]. Hyperchaotic systems are oscillators with two positive Lyapunov exponents, but chaotic systems have only one, so hyperchaotic systems have more complex dynamical behaviors than general chaotic systems [15–18]. In a continuous system, at least four dimensions or more are required to produce hyperchaos, while in discrete systems, it is possible to produce

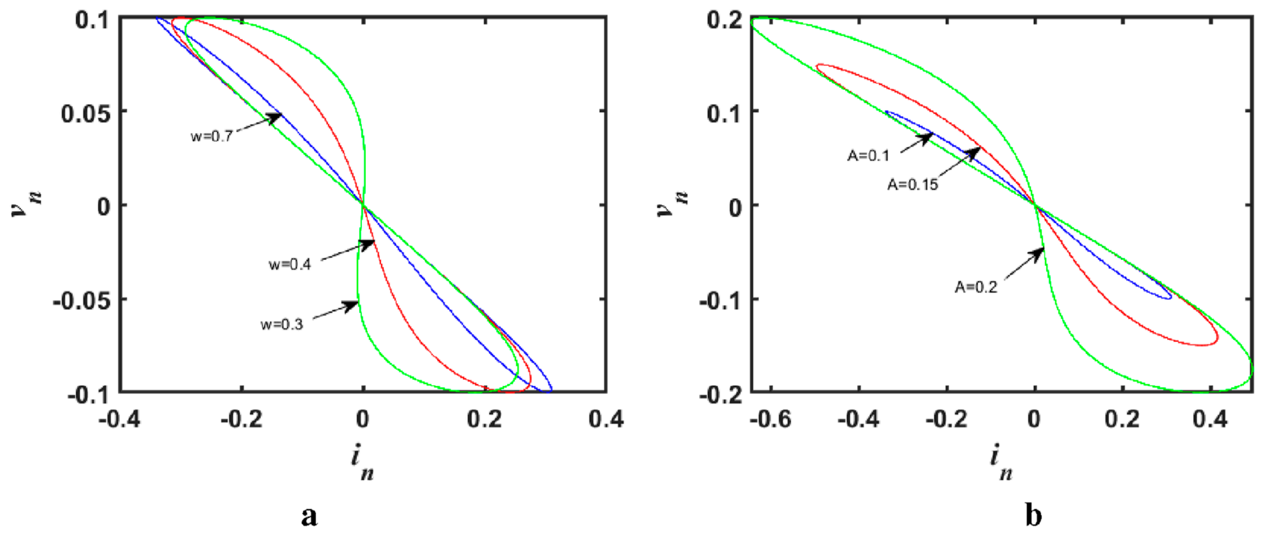


FIGURE 1
Properties of discrete memristors when $I(n) = Asin(\omega n)$. **(a)** Hysteresis loop at $A = 0.1$, $\omega = 0.3, 0.4, 0.7$. **(b)** Hysteresis loop at $\omega = 0.7$, $A = 0.1, 0.15, 0.2$.

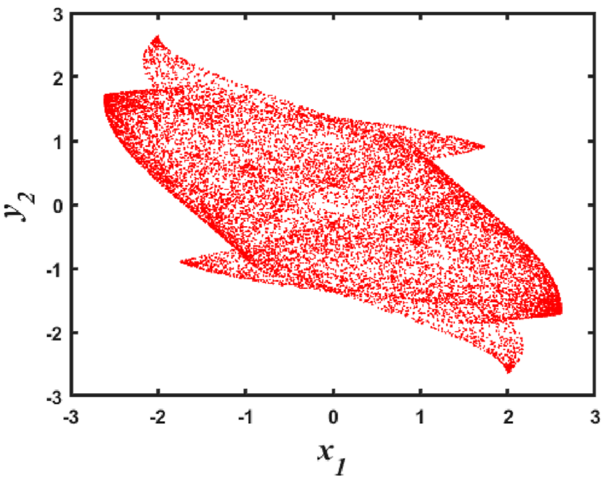


FIGURE 2
Hyperchaotic phase diagram.

a hyperchaotic state in two dimensions and have abundant dynamic behaviors [19, 20]. Mostafaee et al. proposed a novel exponential hyperchaotic system with complex dynamics and analyzed the dynamic behaviors of chaotic attractors, bifurcation graphs, and equilibrium points [21].

Based on the principle of symmetry and completeness of circuit variables, Chua proposed a mathematical model to describe the relationship between charge and magnetic flux, namely, a memristor [22]. As a non-linear resistive element, memristor can adjust the resistance or conductance value through charge or magnetic flux due to its small size and low power consumption [23, 24], and its unique non-linear electrical transport characteristics similar to neural synapses have attracted much attention in many fields [25–28]. In addition, memristors are widely used in chaotic systems to improve nonlinear dynamic behavior [4, 29–31]. It should be noted that most of the research on memristive chaotic systems is limited to the continuous-time domain [32–36], but the common application of continuous memristors will lead to problems such as high computational cost and poor controllability, so the concept of discrete memristors is introduced. In addition, discrete maps have simpler iterative equations and higher computational efficiency than continuous systems [37–41].

Discrete memristor-fusion chaos mapping can generate rich dynamic behaviors such as hyperhybrid and coexisting attractors [42–45], and can also enhance sequence complexity and chaos range [46–48]. Pan et al. [49] proposed a discrete memristor model based on difference theory, describing in detail the process of constructing a discrete memristor by difference theory. Peng et al. [50] established a Simulink model of discrete memristor chaos mapping and verified the feasibility of discrete memristors. Liu et al. [38] reported a discrete two-dimensional memristive map and observed the coexistence of its hidden attractors. Bao et al. [51] reported a new two-dimensional discrete memristive hyperchaotic map.

With the continuous advancement of image encryption technology, researchers have found that it is difficult to improve image encryption with a single chaotic system [52, 53]. In order to ensure people's privacy, Banu S et al. studied traditional encryption algorithms such as AES, RSA, and DES [54], but this algorithm is more suitable for text message encryption. Therefore, it is necessary to develop an efficient encryption scheme to solve the security problem of image encryption [55]. Researchers have investigated a variety of image encryption schemes, such as the application of chaos theory [56], optical methods [57], and compressive sensing [58] to image encryption algorithms. Among them, the characteristics of chaos theory are extremely consistent with the requirements of image encryption schemes, which also promotes the development of chaotic digital image encryption. An image encryption scheme based on double chaotic cyclic shift and Joseph's problem uses the complexity and unpredictability of chaotic systems to enhance

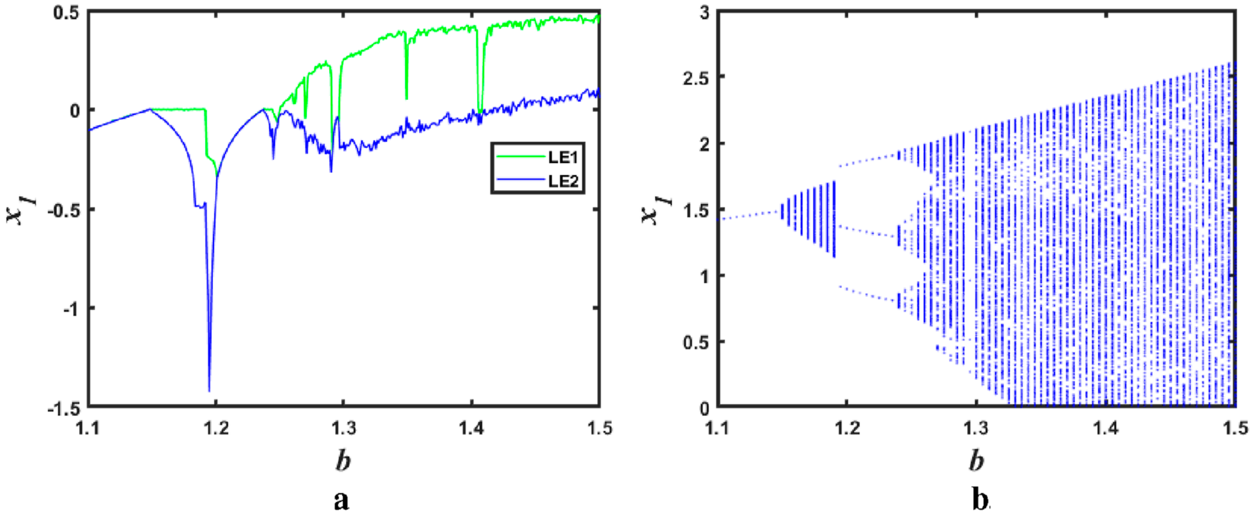


FIGURE 3
The Lyapunov exponent spectrum and bifurcation diagram of $b \in [1.1,1.5]$. **(a)** Lyapunov exponent spectrum. **(b)** Bifurcation Graph.

the encryption effect. Xu et al. proposed a fast image encryption algorithm based on compressed sensing and hyperchaotic map [59], which uses the sparse representation of compressed sensing and the randomness of chaotic map to realize image encryption and decryption. Chen et al. proposed an optical multiimage encryption method based on multiplane phase retrieval and interference [60], which significantly enhances encryption security and robustness of encryption through the complexity and unpredictability of optical components. However, when applied to image encryption, the image encryption algorithm is inefficient due to weaknesses such as discontinuous chaotic regions and narrow chaotic ranges of the chaotic map [61, 62]. In short, the structural defects of the image encryption algorithms and the low performance of chaos theory will lead to the inefficiency of the encryption algorithms, and it is difficult to resist ordinary security attacks.

With the rapid development of information technology, data security and privacy protection are confronted with unprecedented challenges. Traditional image encryption technologies are gradually showing their limitations when dealing with complex and changeable security threats, and there is an urgent need to explore more efficient and secure encryption solutions. In this context, this paper conducts in-depth research and achieves a series of innovative results. Firstly, a new two-dimensional hyperchaotic map is proposed. By skillfully combining the classical cubic map with the improved cosine discrete memristor, a new discrete memristive map is constructed, which provides new ideas for the research of chaotic systems. Secondly, the encryption method is innovated. The chaotic sequence generated by the new chaotic map is integrated into the encryption algorithm. Through operations such as index scrambling and forward and reverse diffusion of images, the image encryption process is optimized. Thirdly, the characteristics of the system are analyzed in multiple dimensions. By studying the parameter-dependent phase diagram, Lyapunov exponential spectrum, bifurcation diagram, and coexisting attractors, and verifying the pseudo-randomness, the chaotic characteristics of the system suitable for image encryption are revealed. Through simulation and analysis of the dynamic characteristics of the chaotic system, it is verified that the system is highly sensitive to parameters, thus providing a new approach for image encryption. Moreover, the chaotic sequence is incorporated into the encryption algorithm. Through operations like index scrambling and diffusion on images and security analysis, it is confirmed that the proposed scheme has extremely high security and anti-interference capabilities, indicating that the chaotic characteristics of the system possess great application value in the field of image encryption.

The general structure of this paper is as follows. Section 2 mainly introduces the Cubic map and the proposed discrete memristor, and then constructs the proposed discrete memristor hyperchaotic map and analyzes its performance; Section 3 shows the rich dynamics of a new discrete memristive hyperchaotic map; Section 4 details the image encryption algorithm; Section 5 summarizes the work of this paper and illustrates the prospects for future research directions.

2 Design of a new discrete hyperchaotic map model

2.1 Mathematical model of discrete hyperchaotic map

Discrete hyperchaotic map is a unique dynamic system, and the key point is to improve the complexity and security of the system with the help of high-dimensional chaos characteristics. Based on the Cubic map, a new discrete hyperchaotic map model can be constructed. Memristors, as the fourth fundamental circuit element, relate charge to magnetic flux and possess unique memory characteristics. In this paper, a cosine-type discrete memristor is proposed, in which the relationship between current and voltage

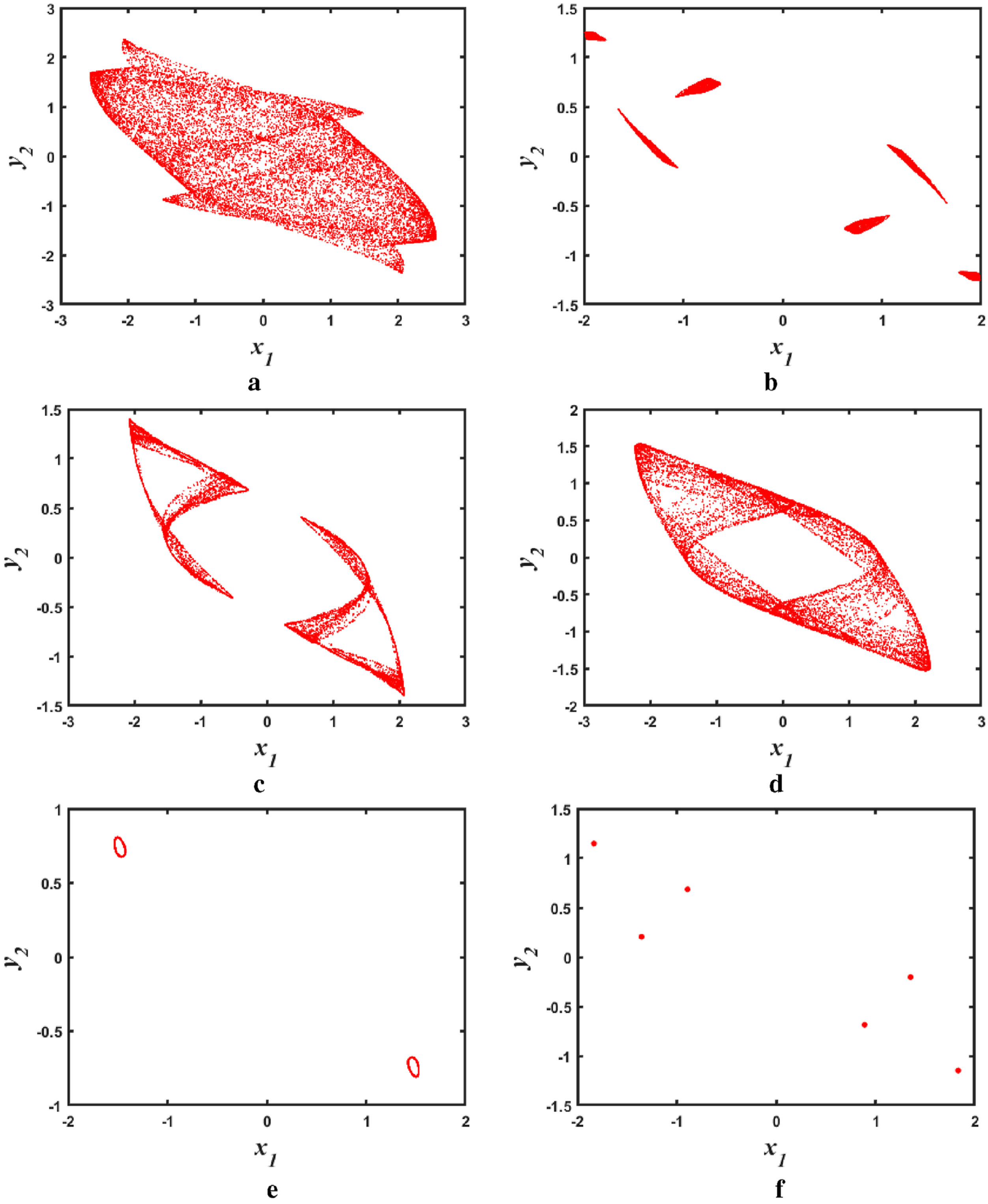


FIGURE 4
Phase diagram of a discrete memristive chaotic system as a function of parameter b. **(a)** $b = 1.48$. **(b)** $b = 1.26$. **(c)** $b = 1.29$. **(d)** $b = 1.35$. **(e)** $b = 1.15$. **(f)** $b = 1.20$.

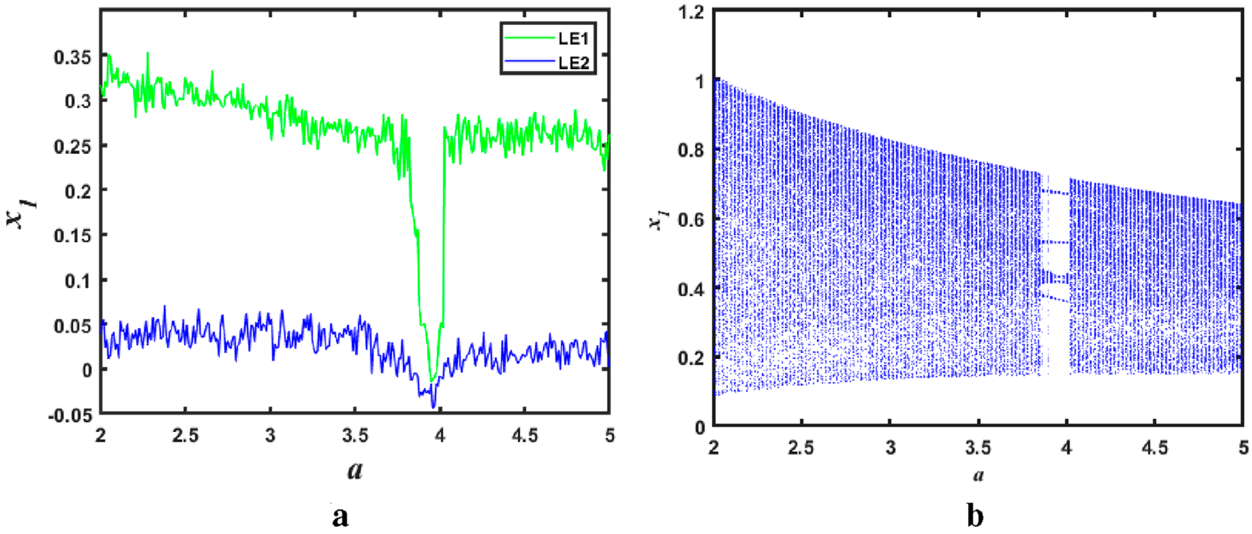


FIGURE 5
The Lyapunov exponent spectrum and bifurcation diagram of a ϵ[2,5]. **(a)** Lyapunov exponential spectrum. **(b)** Bifurcation diagram.

and the relationship between internal charge variables is described as Equation 1:

$$\begin{cases} V(n) = M(q(n)) \cdot I(n) \\ q(n+1) = q(n) + I(n) \end{cases} \tag{1}$$

where $M(q(n)) = cos(q(n))$ is the periodically varying discrete memory resistance, q is the internal charge variable of the memristor, V and I are the voltage and current of the memristor respectively, and an improved class of discrete memristors can be obtained by increasing the parameters g and constant k of the coupling strength of the cosine discrete memristor, the memristor model is shown in Equation 2:

$$\begin{cases} V(n) = M(q(n)) \cdot I(n) = [k + g \cdot cos(q(n))] \cdot I(n) \\ q(n+1) = q(n) + I(n) \end{cases} \tag{2}$$

Adding the power supply $I(n) = Asin(\omega n)$ (ω is the radian frequency) as input to the discrete memristor produces a characteristic volt-ampere graph as shown in Figure 1. The fixed parameters $A = 0.1$, $g = 1$, $k = 1$ and $q0 = 0.1$, it can be seen from the figure that the volt-ampere characteristic curve of the discrete memristor is a diasteretic loop diagram in the shape of "8" through the origin point, when $A = 0.1$ and $\omega = 0.3$, 0.4, 0.7 are taken, Figure 1a is the volt-ampere characteristic curve of the frequency-dependent tight hysteresis loop shape, from which it can be seen that with the increase of radian frequency, the area of the hysteresis loop gradually decreases, and finally tends to a straight line. When $\omega = 0.7$ and $A = 0.1$, 0.15, 0.2, the characteristic curve of the volt-ampere of the discrete memristor is shown in Figure 1b. As the amplitude A decreases, the area of the tight hysteresis loop gradually decreases and finally tends to a straight line. Therefore, its volt-ampere characteristics fully meet the requirements of generalized memristor characteristics.

Through the analysis of numerical simulation, the trajectory distribution and dynamic behavior characteristics of the model can be clearly observed under different initial conditions. It can be seen that the discrete hyperchaotic map not only opens up a new perspective for the research of chaos theory, but also lays a solid theoretical foundation for practical application in related fields.

2.2 Application of discrete memristor in hyperchaotic map

Compared with traditional chaotic maps, discrete hyperchaotic maps exhibit richer dynamic characteristics in parameter space, including irregular periodicity and extreme sensitivity to initial conditions [46]. In the hyperchaotic mapping system, the memristor interacts with other maps. The nonlinear characteristics of the memristor will be coupled with the characteristics of other components, thus generating more complex nonlinear dynamic behaviors, so as to improve the complexity and robustness of their chaotic behavior.

The cubic map is a discrete chaotic map with a simple structure. Its iterative equation is shown in Equation 3:

$$x(n+1) = ax(n)^3 - bx(n) \tag{3}$$

By introducing the memristor model (Equation 2) into the Cubic map, a new two-dimensional discrete memristive chaotic system can be obtained:

$$\begin{cases} x(n+1) = ax(n)^3 - b(1 + gcos(y(n)))x(n) \\ y(n+1) = y(n) + x(n) \end{cases} \tag{4}$$

When the parameters are $a = 0.30$, $g = 0.6$, $b = 1.50$, the two indices are $LE1 = 0.506156$ and $LE2 = 0.0820328$, respectively, the system (Equation 4) has two positive Lyapunov exponents, and the system is in a hyperchaotic state at this time. The phase diagram of its hyperchaotic attractor is shown in Figure 2. As can be seen in Figure 2, the structure of the map is simple, but the dynamic behavior is complex.

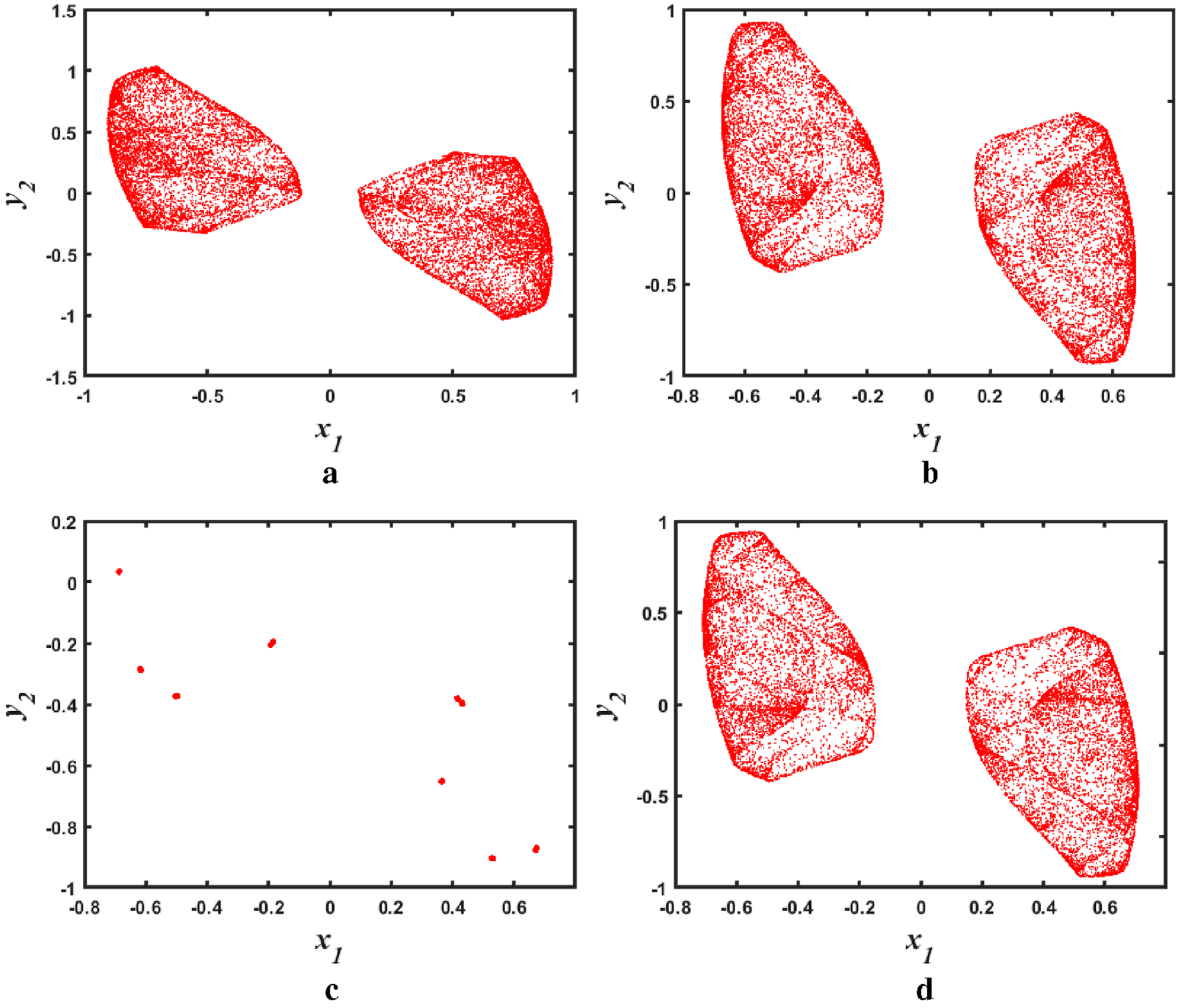


FIGURE 6
Phase diagram of a discrete memristive chaotic system as a function of parameter *a*. **(a)** $a = 2.5$. **(b)** $a = 4.5$. **(c)** $a = 3.96$. **(d)** $a = 4.07$.

where *a*, *b*, and *g* are the control parameters, and in practical applications, the discrete memristor realizes the real-time adjustment of the dynamic behavior in the hyperchaotic map model through its variable resistance characteristics. This mechanism not only improves the adaptability of the system, but also expands the application range of the hyperchaotic map in the field of information encryption.

3 Construction and dynamic analysis of a new discrete hyperchaotic map

3.1 Fixed point

In the study of chaotic systems, an immobile point is one of the important features of a dynamical system, denoting a state that remains unchanged during the evolution of the system. For a new type of discrete hyperchaotic map, it is of great theoretical and practical significance to determine the location and properties of its fixed points. The fixed point of 2D-DMC is the solution to Equation 5.

$$\begin{cases} x* = a(x*)^3 - b\left(1 + g\cos\left(y*\right)\right)x* \\ y* = y* + x* \end{cases} \tag{5}$$

From Equation 5, it follows that 2D-DMC has infinite fixed points, which can be expressed as $F = (x*, y*) = (0, Q)$, where Q is an arbitrary constant. The characteristic equation of the system can be obtained using the Jacobian matrix of fixed points F as shown in Equation 6.

$$P(\lambda) = (\lambda - 1)\left[\lambda + b\left(1 + g\cos Q\right)\right] \tag{6}$$

It can be seen that the eigenvalue $\lambda_1 = 1$ always lies in the unit circle, and whether λ_2 lies inside or outside the unit circle depends on the parameters b, g and the internal initial condition Q of the memristor. Therefore, by

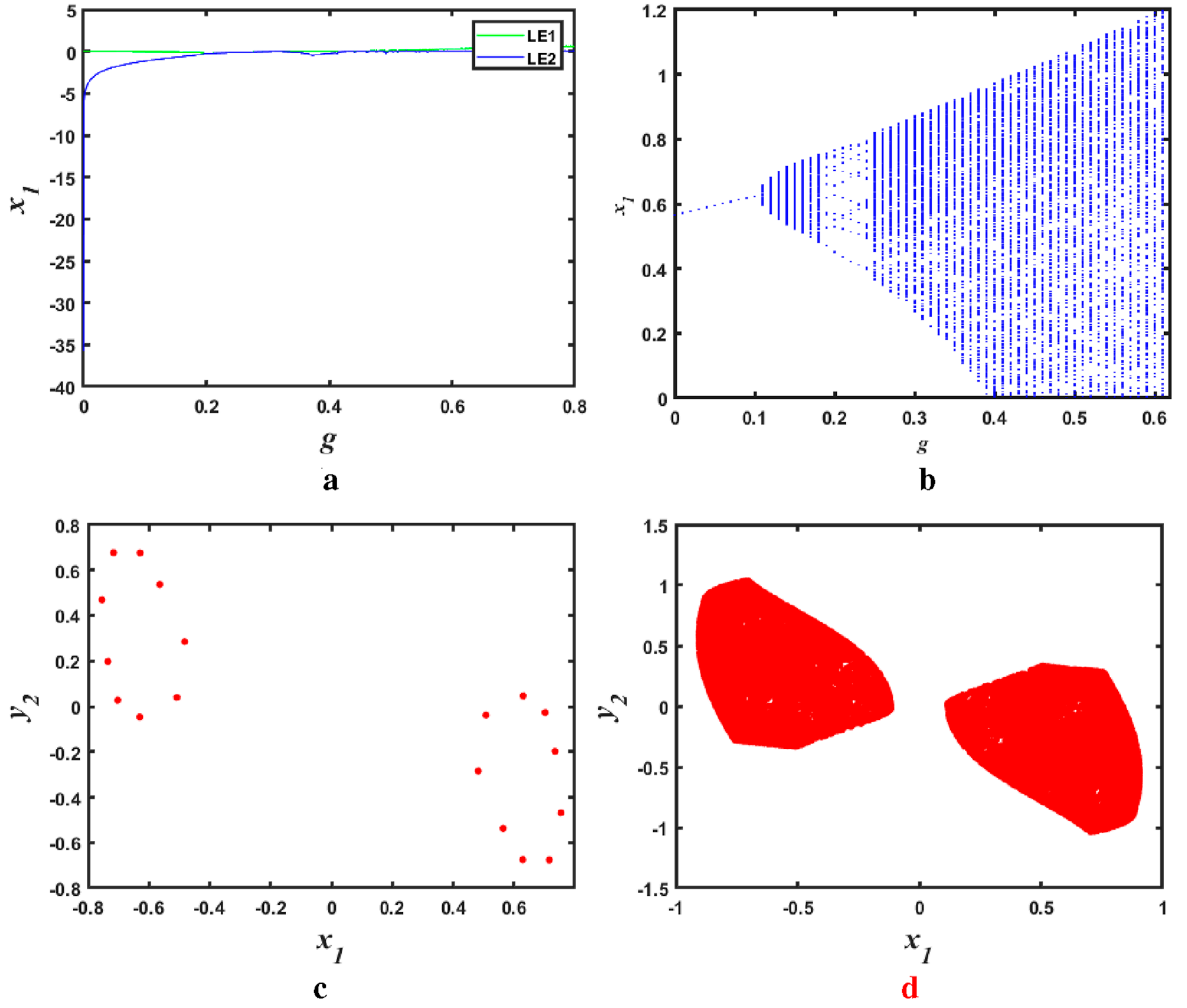


FIGURE 7
The Lyapunov exponent spectrum, bifurcation diagram and phase diagram of $g \in [0,0.6]$. **(a)** Lyapunov exponential spectrum. **(b)** Bifurcation diagram. **(c)** $g = 0.42$. **(d)** $g = 0.6$.

adjusting the parameters b,g and Q, the fixed point of the 2D-DMC can be placed in an unstable or critical stable state.

The properties of these fixed points determine the complexity of hyperchaotic maps and their potential applications in image encryption. Further studies show that appropriate initial values and parameters can make fixed points exhibit rich dynamic behaviors, thus enhancing the security of chaotic systems.

In general, the study of fixed points provides an important theoretical basis for the application of new discrete hyperchaotic maps. Through in-depth analysis of the properties of fixed points, we can understand the behavior characteristics of chaotic systems and provide valuable guidance for the design and implementation of image encryption algorithms.

3.2 Parametric bifurcation graphs and lyapunov exponents

In the dynamic analysis, the chaotic characteristics of the model can be evaluated by using tools such as the Lyapunov exponent and the bifurcation diagram. In order to explore the sensitivity of the system to different parameters, the dynamic behavior of the system is analyzed in detail by a bifurcation diagram and the Lyapunov exponential spectrum.

(1) The influence of the parameter b on the system: set the parameter $a = 0.3$, $g = 0.6$ to explore the influence of the system parameter b on the discrete memristor system. The initial state is $x1 = 0.1$ and $y1 = 0.1$. In the range of $b \in [1.1, 1.5]$, *LEs*

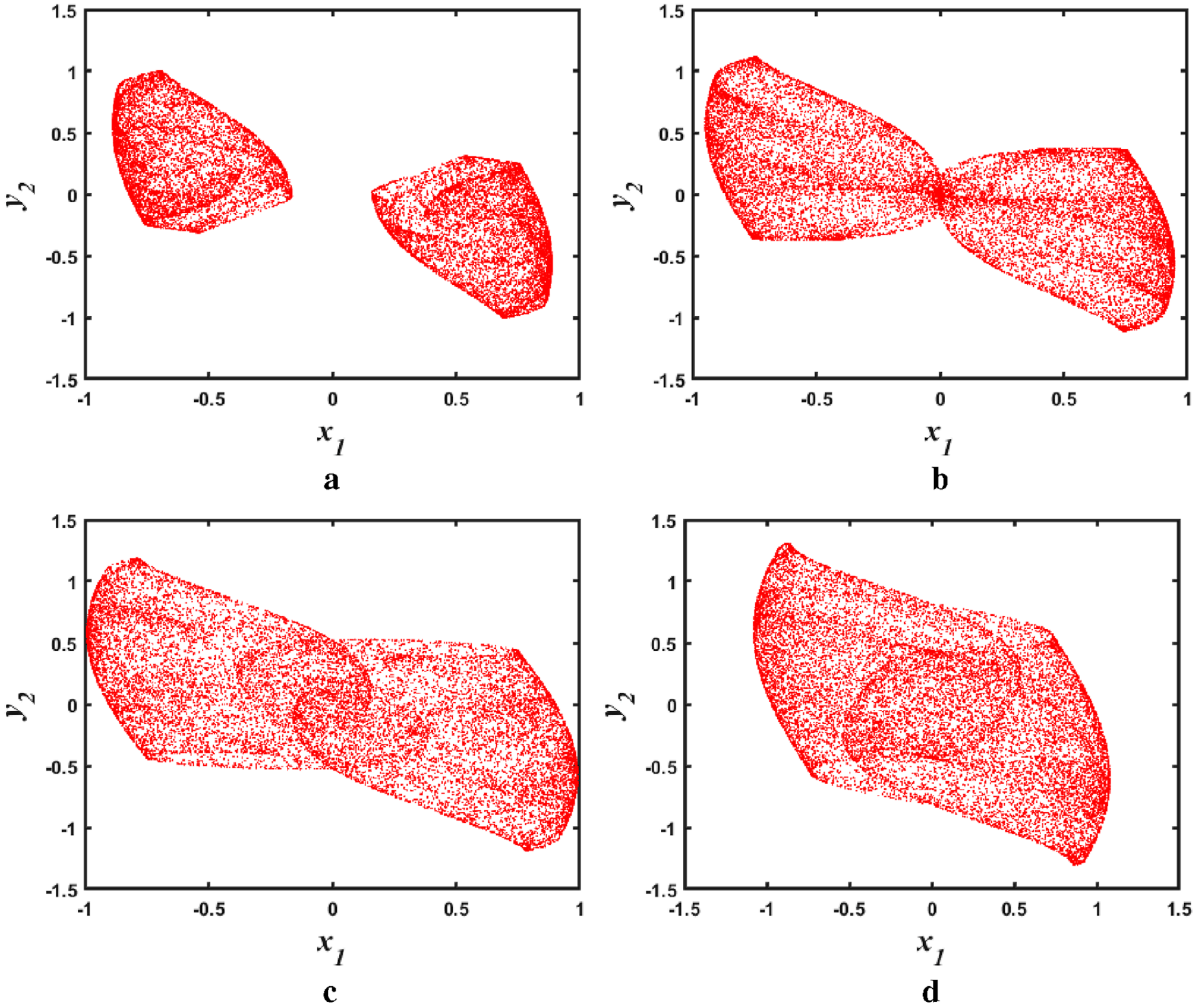


FIGURE 8
Phase diagram of $g \in[0.57,0.8]$ discrete memristive hyperchaotic map. **(a)** $g = 0.58$. **(b)** $g = 0.65$. **(c)** $g = 0.70$. **(d)** $g = 0.80$.

and their bifurcation plots of the discrete memristor chaos map are shown in Figures 3a, b.

As can be seen in the figure, when the parameter $b \in[1.43, 1.54]$ range, the system has two positive Lyapunov exponents, indicating that the system is in a hyperchaotic state in this range. For example, when b = 1.48, the phase diagram of the hyperchaotic attractor is shown in Figure 4a. In the range of parameters $b \in (1.25,1.27)$, $b \in[1.28,1.29)$ and $b \in[1.3,1.41)$, the system has a positive Lyapunov index, indicating that the system is in a chaotic state in this range. For example, when $b = 1.26$, 1.29, and 1.35, the chaotic attractor of the system is shown in Figures 4b–d. When the parameters are in the range of $b \in[1.1, 1.25]$, the system is in a periodic state. For example, when $b = 1.15$ and $b = 1.20$, the chaotic attractor of the system is shown in Figures 4e, f. Through numerical simulation of the model, the rich trajectory behavior and dynamic behavior of the discrete memristor system can be observed under different initial conditions.

(2) Impact of the parameter a on the system: Similarly, to explore the impact of parameter a on the system, parameter b is set to 1.5, g to 0.6, and parameter a varies within the range [2, 5]. The *LEs* of the discrete memristor chaotic map and its bifurcation diagram are shown in Figure 5.

As can be seen in Figure 5, when $a \in[2, 3.95)$ and $a \in(4.08, 5)$, the discrete memristor chaotic system exhibits hyperchaotic behavior. For example, when $a = 2.5$ and 4.5, it can be seen that the system has two positive Lyapunov exponents, and the chaotic attractors of the system are shown in Figures 6a, b. When $a \in[3.95, 3.98]$, the system has a periodic attractor. For example, when $a = 3.96$, the chaotic attractor of the system is shown in Figure 6c. When $a \in[4.06, 4.08]$, the system has a positive Lyapunov exponent and is in a chaotic state. For example, when $a = 4.07$, the chaotic attractor of the system is shown in Figure 6d. The analysis of the comprehensive parameter bifurcation diagram and the Lyapunov exponent can provide a solid theoretical basis for the application of new discrete hyperchaotic maps.

TABLE 1 The NIST test results.

Number	Statistical test terms	P-values	Result
1	Frequency	0.987	Success
2	Intra-block frequency	0.978	Success
3	Cumulative sums	0.765	Success
4	Runs	0.654	Success
5	Longest run	0.543	Success
6	Binary matrix order	0.432	Success
7	FFT	0.21	Success
8	Non-overlapping module matching	0.821	Success
9	Overlapping module matching	0.109	Success
10	General statistics	0.098	Success
11	Approximate entropy	0.087	Success
12	Random deviations	0.076	Success
13	Random excursions variant	0.065	Success
14	Serial	0.054	Success
15	Linear complexity	0.043	Success

TABLE 2 The 0-1 test results.

Sequence	S1	S2	S3	S4
xn	0.9984	0.9994	0.9981	0.9972
yn	0.9981	0.9982	0.9965	0.9971

(3) The influence of the parameter g on the system: In addition, to explore the influence of parameter g on the discrete memristor system, parameters $a = 2.5$ and $b = 1.5$ are set to make the parameter g change in the range of $[0, 0.61]$, and the *LEs* and their bifurcation diagrams of the chaotic map of the discrete memristor are shown in Figure 7.

When parameters $a = 2.5$, $b = 1.5$ and initial values $(x1, y1) = (0.1, 0.1)$ are selected, the bifurcation plot and LE exponential spectra for parameter g are shown in Figures 7a, b. As can be seen in Figure 7, with the change of parameter g, the discrete memristive chaotic system enters the chaotic state from the typical periodic bifurcation, and a complex window period appears in the chaotic region. When $g \in [0.31,\ 0.32]$, the system has a positive LE exponent and presents a chaotic state, and at $g \in [0,\ 0.31)$ and $g \in (0.32,\ 0.32]$, the discrete memristic chaotic system behaves periodically. For example, when $g = 0.42$, the periodic attractor of the system is shown in Figure 7c. When $g \in [0.57,\ 0.6]$, there are two positive LE exponents, and the discrete memristic chaotic system exhibits hyperchaotic behavior. For example, when $g = 0.6$, the discrete memristive chaotic attractor of the system is shown in Figure 7d.

The discrete memristive chaotic attractors corresponding to the different parameter values g are shown in Figure 8. It can be observed that the chaotic attractor has a complex fractal structure and with increasing parameter g, the originally separated chaotic attractor, as shown in Figure 8a, gradually merges with the adjustment of system parameters to form a more complex and unique global attractor, as shown in Figure 8d. The synthesis process of chaotic attractors increases the dimension and complexity of the state space of the system, so that the discrete memristive chaotic system can be flexibly applied to the field of information security. In addition, in the discrete memristive chaotic system, the chaotic sequence generated by the composite attractor has better randomness and non-repeatability, and this complex dynamic behavior makes the output sequence of the system difficult to predict, which provides a high degree of nonlinear characteristics for the encryption process and increases the difficulty of cracking.

3.3 Random analysis

Discrete memristive hyperchaoticmap has been widely used to improve the credibility of data analysis, random number generation, and encrypted communication. In these areas, randomness is a critical requirement, as the resulting chaotic sequences that do not have sufficient randomness can easily be cracked or predicted, compromising the security of the application. Through the randomness test, the randomness and safety of chaotic sequences generated by the discrete memristive hyperchaotic map can be evaluated. To test the randomness of chaotic sequences, we performed two statistical tests, NIST and 0-1. NIST tests are a series of standardized tests that are used to evaluate and verify the security of random number generators and cryptographic algorithms to check whether the generated data are random. The test results are shown in Table 1, from which it can be found that all P values are greater than 0.01, indicating that the key system has successfully passed the test and the generated data have sufficient security and randomness.

The “0-1 test” generally refers to a statistical test performed on a random number or chaotic sequence, mainly to evaluate whether the resulting sequence is sufficiently random to meet specific statistical requirements and application needs. Firstly, a chaotic sequence with a duration of more than 2000 was randomly selected, and the values were selected at a certain step interval for testing, and the test results are shown in Table 2. As you can see from the results in the table, the test result value is close to 1. This indicates that the discrete memristive hyperchaotic map exhibits a high degree of randomness.

Based on the results shown in Tables 1, 2, it can be concluded that the key system derived from the discrete memristive hyperchaos map has excellent randomness. Chaotic randomness testing is of great significance in application security, data analysis, and simulation, which can ensure the security of the application and meet the security and reliability requirements required for image encryption.

FIGURE 9
Encryption and decryption process diagram. **(a)** Encryption Flow Diagram. **(b)** Decryption Process Diagram.

4 Design and implementation of image encryption algorithm

4.1 An image encryption scheme based on hyperchaotic map

In this section, a novel image encryption scheme based on a two-dimensional hyperchaotic map based on cyclic shift, forward and reverse diffusion, and global displacement is introduced. By combining key steps such as pixel diffusion and displacement, hyperchaotic sequences are used to reorder the pixel positions of the original image and disrupt the overall structure of the image. On the one hand, the displacement process ensures randomness, while diffusion further enhances the complexity of image encryption. The processed chaotic sequence and pixel value are used to perform XOR operations to further improve the encryption strength and reduce the risk of information leakage. The steps are as follows:

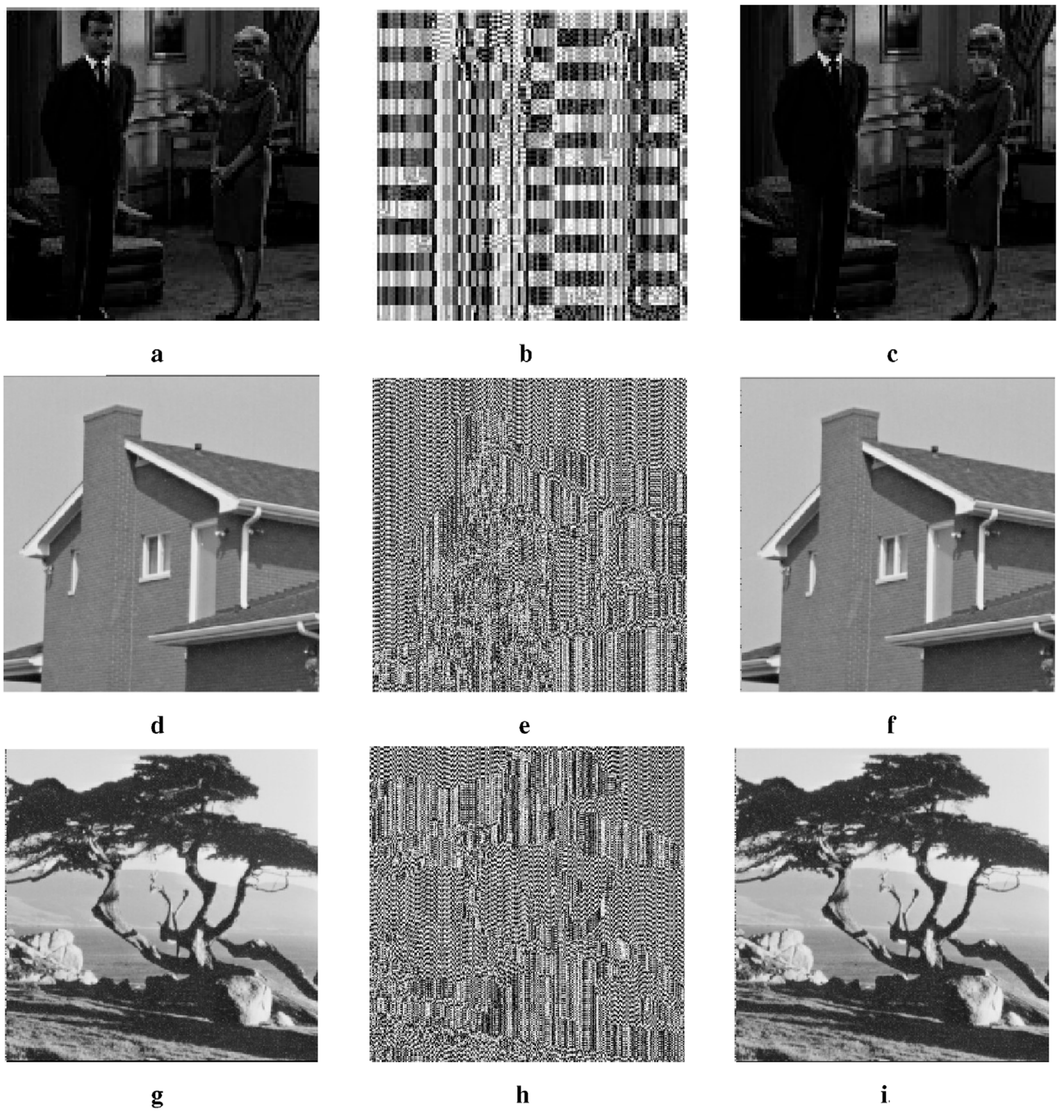


FIGURE 10
Encryption and decryption effects of images. **(a)** Plaintext image. **(b)** Encryption-image. **(c)** Decryption-image. **(d)** Plaintext image. **(e)** Encryption-image. **(f)** Decryption-image. **(g)** Plaintext image. **(h)** Encryption-image. **(i)** Decryption-image.

1. Select the original image and perform channel separation, and select a grayscale image of $m \times n$ as the original image. $m \times n$ chaotic sequences $X(m), Y(n)$ are generated from the state variables $x0, y0$, and two chaotic sequences $X(m)$ and $Y(n)$ are generated using a Gaussian chaotic neural network, which is used for row and column shifts, respectively.

2. Generate $m \times n$ chaotic sequences from the state variables $x0, y0$ The chaotic sequences $Z(m,n)$ and the scrambled image are added pixel by pixel to achieve positive diffusion. Regenerate $m \times n$ chaotic sequences $W(m,n)$ from state variables $x0, y0$. The chaotic sequence $W(m,n)$ and the image after forward diffusion are subtracted pixel by pixel to achieve reverse diffusion.
3. Based on the chaotic characteristics of the hyperchaotic sequences $X(m)$ and $Y(n)$, a permutation index matrix is generated. According to the permutation index matrix, the position of the image is rearranged after forward and reverse diffusion processing. For each pixel position (i,j) in the image, determine its corresponding displacement position (i',j') in the permutation index matrix and move the pixel value from position (i,j) to position (i',j'). In this way, all pixels of the image are rearranged in the order determined by the chaotic sequence, which completely changes the pixel distribution of the image and hides the structure and information of the original image.

The image encryption algorithm of the new discrete memristive chaotic system provides a secure and powerful encryption scheme for grayscale image encryption. In the encryption stage, the plaintext image undergoes cyclic shift, forward and reverse diffusion, and global substitution operations, combined with the

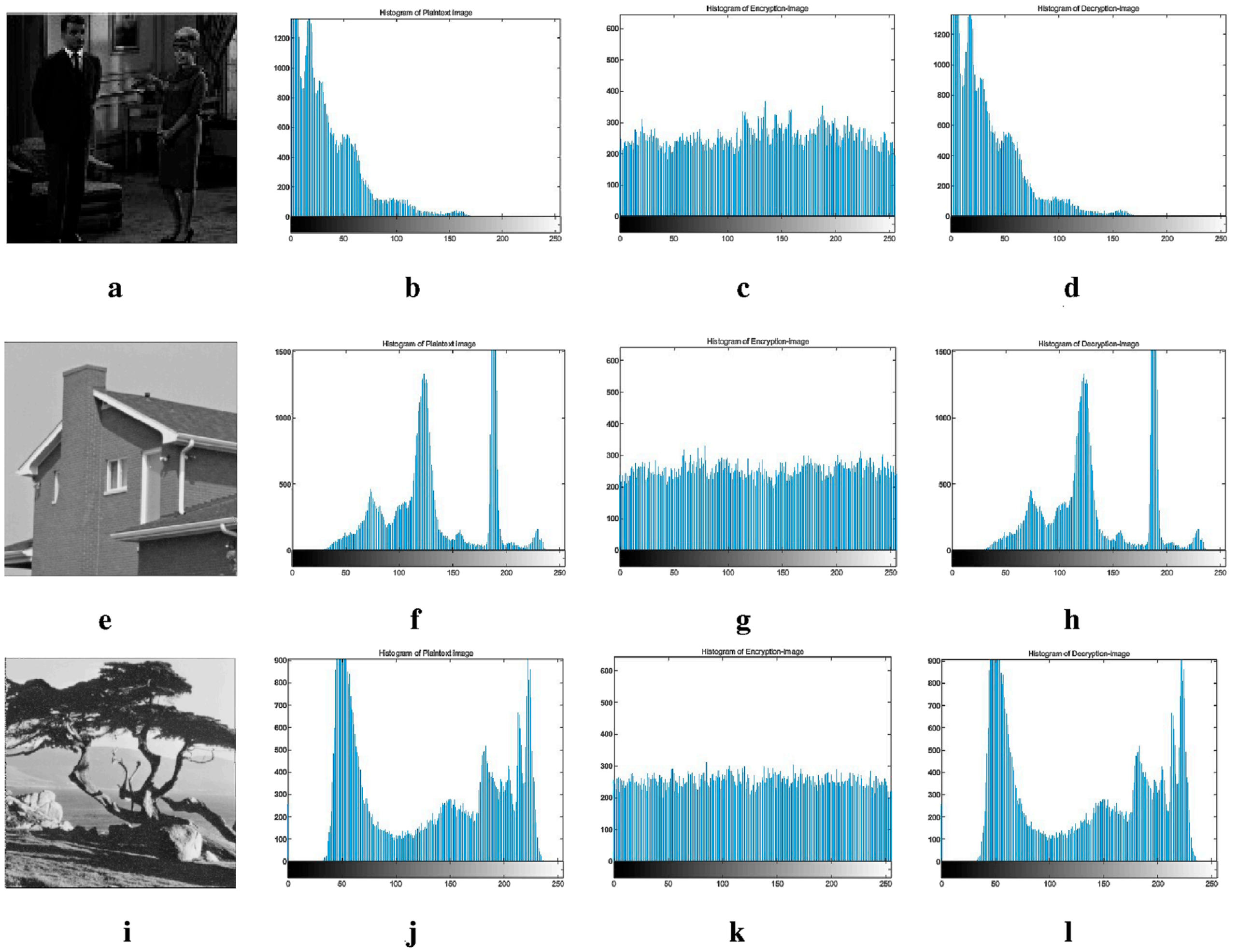


FIGURE 11
Histograms of plaintext and ciphertext images. **(a)** Plaintext-image. **(b)** Histogram of Plaintext-image. **(c)** Histogram of Encryption-image. **(d)** Histogram of Decryption-image. **(e)** Plaintext-image. **(f)** Histogram of Plaintext-image. **(g)** Histogram of Encryption-image. **(h)** Histogram of Decryption-image. **(i)** Plaintext image. **(j)** Histogram of Plaintext image. **(k)** Histogram of Encryption-image. **(l)** Histogram of Decryption-image.

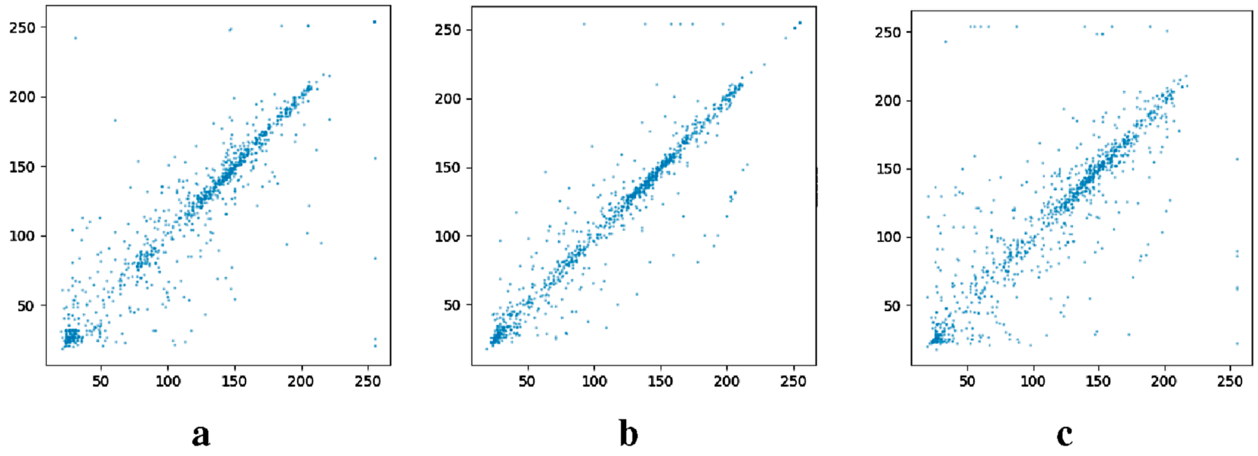


FIGURE 12
Correlation between adjacent pixels in a plaintext image. **(a)** Horizontal. **(b)** Vertical. **(c)** Diagonal.

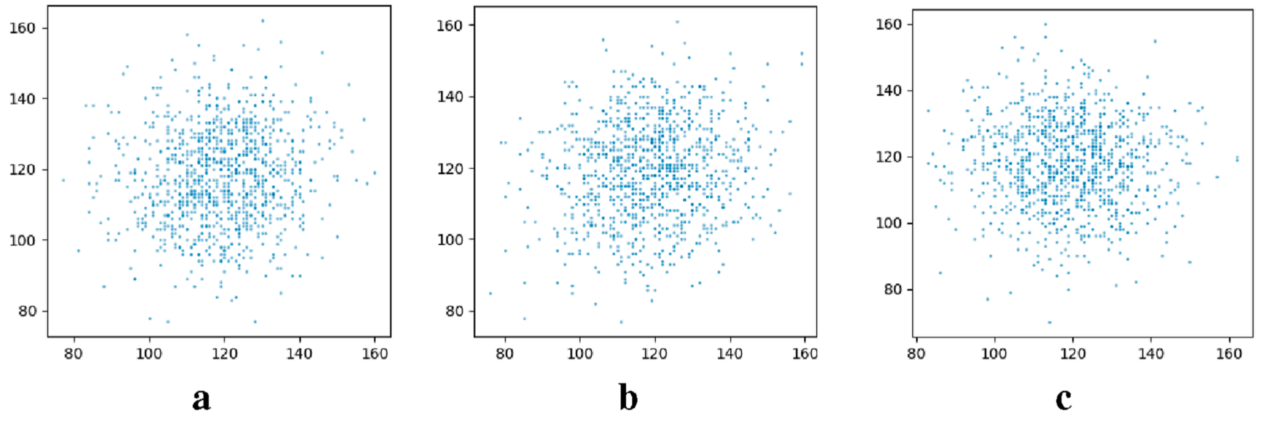


FIGURE 13
Encrypted image adjacent pixel correlation. **(a)** Horizontal. **(b)** Vertical. **(c)** Diagonal.

TABLE 3 Correlation coefficient of ciphertext images.

Encryption scheme	Horizontal	Vertical	Diagonal
Ref. [64]	0.0055	−0.0068	−0.0032
Ref. [65]	−0.0158	−0.0042	−0.0039
Ref. [38]	−0.0066	−0.0089	0.0424
This article	−0.0036	0.0032	0.0010

TABLE 4 Information entropy of ciphertext images with different encryption schemes.

Scheme	Ref. [38]	Ref. [40]	This article
Information entropy	7.9909	7.9971	7.9993

dynamic key generated by the chaotic system, and finally generates an irreversible ciphertext image,as shown in Figure 9a. During decryption, global substitution, forward and reverse diffusion, and cyclic shift are performed in reverse, and the original pixel value and position are restored by the same chaos key to achieve lossless decryption,as shown in Figure 9b.

4.2 Performance analysis of encryption algorithms

In image encryption algorithms, the size of the key space directly determines the security of the encryption. To verify the feasibility and effectiveness of the proposed algorithm, simulation tests were performed using Matlab 2023b, with the key set as $g = 0.6$, $a = 3, b = 1.8$ and $(x0, y0) = (0.1, 0.1)$. A chaotic sequence required for encryption was generated using a discrete memristive chaotic system, and then the image was encrypted through the encryption algorithm. When establishing a new discrete hyperchaotic map, the selection of the key depends on multiple parameters, such as initial conditions and the dynamic characteristics of the system. As shown in Figures 10a, d, g is the original image before the above-mentioned algorithm is encrypted, Figures 10b, e, h are the encrypted image after using the above-mentioned encryption algorithm, and Figures 10c, f, i are the decryption image after using the above-mentioned algorithm. In order to verify the security of the encryption effect of the system, this paper conducted performance analysis, mainly including key space analysis, histogram analysis, correlation analysis, and information entropy analysis.

4.2.1 Key space analysis

In image encryption algorithms, the size of the key space directly determines the security of the encryption. Studies have shown that the larger the key space, the more difficult it is for attackers to crack. It is generally accepted that the size of the key space should be greater than 2,128 [45] to ensure security. The keys of IES-CTG are $a, b, g, x0$ and $y0$, and the parameter intervals $a \in [2,3]$, $b \in [1.6,1.8]$, $g \in [0.5, 0.6]$ and the initial value range $x0 \in [0.1, 0.3]$, $y0 \in [0.1, 0.3]$, and the results of image encryption and decryption are shown in Figure 10. Therefore, the key space S of the IES-CTG is shown in Equation 7:

$$S = S_1 \times S_2 \times S_3 \times S_4 \times S_5 = 8 \times 10^{71} \approx 2^{238} \tag{7}$$

where $S_1 = (3-2) \times 10^{15}, S_2 = (1.8-1.6) \times 10^{15}, S_3 = (0.6-0.5) \times 10^{15}, S_4 = (0.3-0.1) \times 10^{15}, S_5 = (0.3-0.1) \times 10^{15}$.

Brute-force attack refers to the situation where an attacker tries all possible key combinations until the correct key for decrypting the information is found. The size of the key space determines the number of possible key combinations. The larger the key space, the more difficult it is for the attacker to find the correct key through brute-force attempts. The key space designed for the novel discrete hyperchaotic map is 2^{238}, significantly larger than the recommended minimum of 2^{128} for the key space, which can effectively resist brute-force attacks. Therefore, the algorithm has larger scale and complexity, and the proposed image encryption scheme can effectively resist external attacks and provide greater security.

4.2.2 Histogram analysis

In image encryption, histogram analysis is an important method to evaluate the encryption effect [63]. Figures 11a, c, e are the original images, and their corresponding image histograms are shown in Figures 11b, d, f, and their pixel value distribution can be visually seen. By comparing the encrypted histograms, as shown in Figures 11g, i, k, it can be observed that the encrypted image histograms should show more uniform distribution characteristics. The histogram of the decrypted image is obtained by the decryption algorithm as shown in Figures 11h, j, l. This balance indicates that confusing and dispersing the pixel information of the original image reduces its recognizability and improves security. That is, the attacker cannot obtain the histogram information of the plaintext image by statistically analyzing the histogram of the ciphertext image, indicating that the proposed algorithm has good diffusion and resistance to statistical attacks.

The entropy of the histogram is also a key indicator to evaluate the effectiveness of image encryption. The higher the entropy value, the higher the complexity of the encrypted image information and the stronger the ability to resist various attacks, as shown in Figures 11d, h, i. In this study, the encrypted image generated by the new discrete hyperchaotic map has a high histogram entropy value, which shows the effectiveness and security of the encryption algorithm in practical applications.

4.2.3 Relevance analysis

In the design and implementation of image encryption algorithms, correlation analysis is an important performance index. Low correlation means that there is almost no linear or non-linear relationship between the pixel values of the encrypted image, which effectively increases the difficulty of cracking. In order to evaluate the performance of the new discrete hyperchaotic map proposed in the process of image encryption, it is necessary to analyze the correlation of the images before and after encryption. Figures 12, 13 illustrate the correlation between the adjacent pixels of the plaintext image before image encryption and the ciphertext image after encryption, respectively.

The image is very strong, as shown in Figures 12a–c, and there is usually a correlation close to 1; However, the correlation between adjacent pixels in a ciphertext image is close to zero, as shown in Figures 13a–c. For different test images, different chaotic sequences are used for encryption, and the correlation difference between the encryption results can be observed, which further verifies the randomness of the new discrete hyperchaotic map, thus improving the security of the encrypted images.

As can be seen from the correlation coefficient of the ciphertext image in Table 3, the correlation between adjacent pixels in the ciphertext image is close to 0, and they are almost uncorrelated. The experimental results show that the designed image encryption algorithm maintains a high encryption strength under the condition of low correlation. Compared with traditional chaotic encryption algorithms, this novel discrete hyperchaotic mapping effectively reduces the correlation between different pixels, thereby enhancing the security of the encrypted image. When compared with more advanced chaotic encryption algorithms, this algorithm also has obvious advantages in terms of processing speed. It can complete the encryption and decryption processes of images more rapidly. Moreover, when facing common attack methods such as differential attacks and statistical attacks, it demonstrates stronger attack resistance, providing a more reliable guarantee for the security of image data.

4.2.4 Information entropy analysis

Information entropy is a basic concept of information theory. It is an important index for measuring the randomness and uncertainty of information. Generally, it is around 8.0, indicating that the encrypted image has good randomness in the pixel intensity distribution. In this study, a new encryption algorithm based on a discrete hyperchaotic map is used to compare the entropy of the original image and the encrypted image. It can be seen from Table 4 that after IES - CTG encryption processing, the information entropy of the ciphertext image is very close to the ideal value of 8, and compared to some existing schemes, it has certain advantages.

Furthermore, the variation law of the information entropy under different chaotic parameters is analyzed, and the information entropy performance of the encryption results is affected by adjusting the parameters of the chaotic system. Under the corresponding parameter settings, the increase in the entropy value shows significant sensitivity, which further verifies the effectiveness of chaos characteristics in enhancing the security of image encryption.

5 Conclusion

In this paper, we conduct in-depth research and discussion on a new type of discrete hyperchaotic map and its application in image encryption. By designing and analyzing a novel discrete hyperchaotic map model, we not only clarify its dynamic characteristics, but also reveal its advantages in generating high-quality chaotic sequences. Then, an image encryption algorithm based on a novel discrete hyperchaotic map design is implemented on the MATLAB platform. The key is used to scramble and diffuse the digital image to be encrypted at the pixel level to improve the security of the image. The experimental results show that the proposed encryption algorithm has significant performance advantages. By comparing images with different encryption effects, the security of encrypted images was evaluated using methods such as histogram analysis, information entropy calculation, and adjacent pixel correlation detection. The experimental results show that the encrypted image presents a good degree of visual chaos and the information entropy value is significantly improved, indicating that its security is better than that of traditional image encryption methods.

The new discrete hyperchaotic map and its application in image encryption have important theoretical value and practical significance. Future research can further explore the application potential of other chaos map models in different information security fields, to promote the progress and innovation of overall information encryption technology.

Data availability statement

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

Author contributions

FY: Supervision, Writing – review and editing, Project administration, Resources, Writing – original draft. YW: Conceptualization, Data curation, Formal Analysis, Methodology, Software, Writing – original draft, Writing – review and editing. XW: Formal Analysis, Software, Validation, Writing – original draft. TH: Investigation, Supervision, Validation, Writing – original draft. SZ: Software, Supervision, Writing – original draft. JJ: Funding acquisition, Supervision, Writing – review and editing.

Funding

The author(s) declare that financial support was received for the research and/or publication of this article. This work was supported by National Natural Science Foundation of China under Grant 62273141, and by the Guiding Science and Technology Plan Project of Changsha City under Grant kzd2501129.

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.

Generative AI statement

The authors declare that no Generative AI was used in the creation of this manuscript.

Publisher's note

References

1. Deng Q, Wang C, Yang Q. Chaotic dynamics of memristor-coupled tabu learning neuronal network. *International Journal of Bifurcation and Chaos* (2025) 35:2550053. doi:10.1142/S0218127425500531

2. Yuan Y, Yu F, Tan B, Huang Y, Yao W, Cai S, et al. A class of n-d Hamiltonian conservative chaotic systems with three-terminal memristor: modeling, dynamical analysis, and fpga implementation. *Chaos* (2025) 35:013121. doi:10.1063/5.0238893

3. Gao S, Zhang Z, Iu HHC, Ding S, Mou J, Erkan U, et al. A parallel color image encryption algorithm based on a 2d logistic-rulkov neuron map. *IEEE Internet Things J* (2025) 1. doi:10.1109/JIOT.2025.3540097

4. Yu F, Wu C, Xu S, Yao W, Xu C, Cai S, Wang C. Color video encryption transmission in IoT based on memristive hopfield neural network. *Signal, Image and Video Processing* (2025) 19:77. doi:10.1007/s11760-024-03697-x

5. Jin J, Wu M, Ouyang A, Li K, Chen C. A novel dynamic hill cipher and its applications on medical iot. *IEEE Internet Things J* (2025) 12:14297–308. doi:10.1109/JIOT.2025.3525623

6. Yuan Z, Wu Y, Ou C, Zhong P, Zhao X, Ma M. Dynamical behavior of sw-sw neural networks. *Chin J Phys* (2025) 94:108–20. doi:10.1016/j.cjph.2024.12.031

7. Luo D, Wang C, Deng Q, Sun Y. Dynamics in a memristive neural network with three discrete heterogeneous neurons and its application. *Nonlinear Dyn* (2025) 113:5811–24. doi:10.1007/s11071-024-10513-1

8. Hua C, Cao X, Liao B. Real-time solutions for dynamic complex matrix inversion and chaotic control using ode-based neural computing methods. *Comput Intelligence* (2025) 41:e70042. doi:10.1111/coin.70042

9. Yu F, Zhang S, Su D, Wu Y, Gracia YM, Yin H. Dynamic analysis and implementation of fpga for a new 4d fractional-order memristive hopfield neural network. *Fractal and Fractional* (2025) 9:115. doi:10.3390/fractalfract9020115

10. Deng Q, Wang C, Sun Y, Yang G. Memristive multi-wing chaotic hopfield neural network for lidar data security. *Nonlinear Dyn* (2025) 113:17161–76. doi:10.1007/s11071-025-10982-y

11. Feng W, Zhang J, Chen Y, Qin Z, Zhang Y, Ahmad M, et al. Exploiting robust quadratic polynomial hyperchaotic map and pixel fusion strategy for efficient image encryption. *Expert Syst Appl* (2024) 246:123190. doi:10.1016/j.eswa.2024.123190

12. Lai Q, Yang L, Chen G. Design and performance analysis of discrete memristive hyperchaotic systems with stuffed cube attractors and ultraboosting behaviors. *IEEE Trans Ind Electronics* (2024) 71:7819–28. doi:10.1109/tie.2023.3299016

13. Liu X, Mou J, Zhang Y, Cao Y. A new hyperchaotic map based on discrete memristor and meminductor: dynamics analysis, encryption application, and dsp implementation. *IEEE Trans Ind Electronics* (2023) 71:5094–104. doi:10.1109/tie.2023.3281687

14. Yu F, Tan B, He T, He S, Huang Y, Cai S, et al. A wide-range adjustable conservative memristive hyperchaotic system with transient quasi-periodic characteristics and encryption application. *Mathematics* (2025) 13:726. doi:10.3390/math13050726

15. Chen L, Yu M, Luo J, Mi J, Shi K, Tang S. Dynamic analysis and fpga implementation of a new linear memristor-based hyperchaotic system with strong complexity. *Mathematics* (2024) 12:1891. doi:10.3390/math12121891

16. Ma X, Wang Z, Wang C. An image encryption algorithm based on tabu search and hyperchaos. *Int J Bifurcation Chaos* (2024) 34:2450170. doi:10.1142/s0218127424501700

17. Zhu S, Deng X, Zhang W, Zhu C. Construction of a new 2d hyperchaotic map with application in efficient pseudo-random number generator design and color image encryption. *Mathematics* (2023) 11:3171. doi:10.3390/math11143171

18. 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

19. Hua Z, Zhou Y, Bao B. Two-dimensional sine chaotification system with hardware implementation. *IEEE Trans Ind Inform* (2019) 16:887–97. doi:10.1109/tii.2019.2923553

20. Feng W, Wang Q, Liu H, Ren Y, Zhang J, Zhang S, et al. Exploiting newly designed fractional-order 3d lorenz chaotic system and 2d discrete polynomial hyper-chaotic map for high-performance multi-image encryption. *Fractal and Fractional* (2023) 7:887. doi:10.3390/fractalfract7120887

21. Mostafaee J, Mobayen S, Vaseghi B, Vahedi M, Fekih A. Complex dynamical behaviors of a novel exponential hyper–chaotic system and its application in fast synchronization and color image encryption. *Sci Prog* (2021) 104:00368504211003388. doi:10.1177/00368504211003388

22. Liu J, Li Z, Tang Y, Hu W, Wu J. 3d convolutional neural network based on memristor for video recognition. *Pattern Recognition Lett* (2020) 130:116–24. doi:10.1016/j.patrec.2018.12.005

23. Hong Q, Jiang H, Xiao P, Du S, Li T. A parallel computing scheme utilizing memristor crossbars for fast corner detection and rotation invariance in the orb algorithm. *IEEE Trans Comput* (2025) 74:996–1010. doi:10.1109/tc.2024.3504817

24. Yu F, He S, Yao W, Cai S, Xu Q. Bursting firings in memristive hopfeld neural network with image encryption and hardware implementation. *IEEE Trans Computer-Aided Des Integrated Circuits Syst* (2025) 1–13. doi:10.1109/TCAD.2025.3567878

25. Zhang S, Yao W, Xiong L, Wang Y, Tang L, Zhang X, et al. A hindmarsh–rose neuron model with electromagnetic radiation control for the mechanical optimization design. *Chaos, Solitons and Fractals* (2024) 187:115408. doi:10.1016/j.chaos.2024.115408

26. Sun J, Zhai Y, Liu P, Wang Y. Memristor-based neural network circuit of associative memory with overshadowing and emotion congruent effect. *IEEE Trans Neural Networks Learn Syst* (2024) 36:3618–30. doi:10.1109/TNNLS.2023.3348553

27. Xu Q, Fang Y, Feng C, Parastesh F, Chen M, Wang N. Firing activity in an n-type locally active memristor-based hodgkin–huxley circuit. *Nonlinear Dyn* (2024) 112:13451–64. doi:10.1007/s11071-024-09728-z

28. Peng Y, Li M, Li Z, Ma M, Wang M, He S. What is the impact of discrete memristor on the performance of neural network: a research on discrete memristor-based bp neural network. *Neural Networks* (2025) 185:107213. doi:10.1016/j.neunet.2025.107213

29. Yao W, Fang J, Yu F, Xiong L, Tang L, Zhang J, et al. Electromagnetic radiation control for nonlinear dynamics of hopfield neural networks. *Chaos* (2024) 34:073149. doi:10.1063/5.0194928

30. Deng W, Ma M. Analysis of the dynamical behavior of discrete memristor-coupled scale-free neural networks. *Chin J Phys* (2024) 91:966–76. doi:10.1016/j.cjph.2024.08.033

31. Xu Q, Ding X, Chen B, Parastesh F, Ho-Ching IH, Wang N. A universal configuration framework for mem-element-emulator-based bionic firing circuits. *IEEE Trans Circuits Syst Regular Pap* (2024) 71:4120–30. doi:10.1109/tcsi.2024.3428857

32. Lin H, Deng X, Yu F, Sun Y. Diversified butterfly attractors of memristive hnn with two memristive systems and application in iomt for privacy protection. *IEEE Trans Computer-Aided Des Integrated Circuits Syst* (2025) 44:304–16. doi:10.1109/tcad.2024.3429410

33. Hong Q, Xiao P, Fan R, Du S. Memristive neural network circuit design based on locally competitive algorithm for sparse coding application. *Neurocomputing* (2024) 578:127369. doi:10.1016/j.neucom.2024.127369

34. Gao S, Iu HHC, Erkan U, Simsek C, Toktas A, Cao Y, et al. A 3d memristive cubic map with dual discrete memristors: design, implementation, and application in image encryption. *IEEE Trans Circuits Syst Video Technology* (2025) 1. doi:10.1109/TCSVT.2025.3545868

35. Yu F, Su D, He S, Wu Y, Zhang S, Yin H. Resonant tunneling diode cellular neural network with memristor coupling and its application in police forensic digital image protection. *Chin Phys B* (2025) 34:050502. doi:10.1088/1674-1056/adb8bb

36. Wan Q, Yang Q, Liu T, Chen C, Shen K. Single direction, grid and spatial multi-scroll attractors in hopfield neural network with the variable number memristive self-connected synapses. *Chaos, Solitons and Fractals* (2024) 189:115584. doi:10.1016/j.chaos.2024.115584

37. Yu F, Xu S, Lin Y, Gracia YM, Yao W, Cai S. Dynamic analysis, image encryption application and fpga implementation of a discrete memristor-coupled neural network. *Int J Bifurcation Chaos* (2024) 34:2450068. doi:10.1142/s0218127424500688

38. Liu X, Sun K, Wang H, He S. A class of novel discrete memristive chaotic map. *Chaos, Solitons and Fractals* (2023) 174:113791. doi:10.1016/j.chaos.2023.113791

39. Tang Z, Zhang Y. Continuous and discrete gradient-zhang neuronet (gzn) with analyses for time-variant overdetermined linear equation system solving as well as mobile localization applications. *Neurocomputing* (2023) 561:126883. doi:10.1016/j.neucom.2023.126883

40. Demirtaş M. A novel multiple grayscale image encryption method based on 3d bit-scrambling and diffusion. *Optik* (2022) 266:169624. doi:10.1016/j.ijleo.2022.169624

41. Xiang Q, Gong H, Hua C. A new discrete-time denoising complex neurodynamics applied to dynamic complex generalized inverse matrices. *The J Supercomputing* (2025) 81:159–25. doi:10.1007/s11227-024-06601-z

42. Peng Y, He S, Sun K. A higher dimensional chaotic map with discrete memristor. *AEU-International J Electronics Commun* (2021) 129:153539. doi:10.1016/j.aeue.2020.153539

43. Liu X, Mou J, Yan H, Bi X. Memcapacitor-coupled Chebyshev hyperchaotic map. *Int J Bifurcation Chaos* (2022) 32:2250180. doi:10.1142/s0218127422501802

44. Wang C, Li Y, Deng Q. Discrete-time fractional-order local active memristor-based hopfield neural network and its fpga implementation. *Chaos, Solitons and Fractals* (2025) 193:116053. doi:10.1016/j.chaos.2025.116053

45. Di Marco M, Forti M, Pancioni L, Tesi A. New class of discrete-time memristor circuits: first integrals, coexisting attractors and bifurcations without parameters. *Int J Bifurcation Chaos* (2024) 34:2450001. doi:10.1142/s0218127424500019

46. Zhong H, Li G, Xu X. A generic voltage-controlled discrete memristor model and its application in chaotic map. *Chaos, Solitons and Fractals* (2022) 161:112389. doi:10.1016/j.chaos.2022.112389

47. Wang C, Luo D, Deng Q, Yang G. Dynamics analysis and fpga implementation of discrete memristive cellular neural network with heterogeneous activation functions. *Chaos, Solitons and Fractals* (2024) 187:115471. doi:10.1016/j.chaos.2024.115471

48. Hamadneh T, Abbes A, Al-Tarawneh H, Gharib GM, Salameh WMM, Al Soudi MS, et al. On chaos and complexity analysis for a new sine-based memristor map with commensurate and incommensurate fractional orders. *Mathematics* (2023) 11:4308. doi:10.3390/math11204308

49. Pan X. Research on discrete differential solution methods for derivatives of chaotic systems. *AIMS Mathematics* (2024) 9:33995–4012. doi:10.3934/math.20241621

50. Peng S, Shi H, Li R, Xiang Q, Dai S, Li Y. Simulink modeling and analysis of a three-dimensional discrete memristor map. *Symmetry* (2024) 16:990. doi:10.3390/sym16080990

51. Bao BC, Li H, Wu H, Zhang X, Chen M. Hyperchaos in a second-order discrete memristor-based map model. *Electronics Lett* (2020) 56:769–70. doi:10.1049/el.2020.1172

52. Zheng J, Luo Z, Tang Z. An image encryption algorithm based on multichaotic system and dna coding. *Discrete Dyn Nat Soc* (2020) 2020:1–16. doi:10.1155/2020/5982743

53. Lai Q, Lai C, Zhang H, Li C. Hidden coexisting hyperchaos of new memristive neuron model and its application in image encryption. *Chaos, Solitons and Fractals* (2022) 158:112017. doi:10.1016/j.chaos.2022.112017

54. Banu SA, Amirtharajan R. A robust medical image encryption in dual domain: chaos-dna-iwt combined approach. *Med and Biol Eng and Comput* (2020) 58:1445–58. doi:10.1007/s11517-020-02178-w

55. Li Y, You X, Lu J, Lou J. A joint image compression and encryption scheme based on a novel coupled map lattice system and dna operations. *Front Inf Technology and Electron Eng* (2023) 24:813–27.

56. Ahmad I, Shin S. A novel hybrid image encryption–compression scheme by combining chaos theory and number theory. *Signal Processing: Image Commun* (2021) 98:116418. doi:10.1016/j.image.2021.116418

57. Wei H, Wang X. Optical multiple-image authentication and encryption based on phase retrieval and interference with sparsity constraints. *Opt and Laser Technology* (2021) 142:107257. doi:10.1016/j.optlastec.2021.107257

58. Hua Z, Zhang K, Li Y, Zhou Y. Visually secure image encryption using adaptive-thresholding sparsification and parallel compressive sensing. *Signal Process.* (2021) 183:107998. doi:10.1016/j.sigpro.2021.107998

59. Xu Q, Sun K, Cao C, Zhu C. A fast image encryption algorithm based on compressive sensing and hyperchaotic map. *Opt Lasers Eng* (2019) 121:203–14. doi:10.1016/j.optlaseng.2019.04.011

60. Chen W, Chen X. Optical multiple-image encryption based on multiplane phase retrieval and interference. *J Opt* (2011) 13:115401. doi:10.1088/2040-8978/13/11/115401

61. Wei Z, Sixing X, Guilin W, Yonghong L, Qichang J. Multi-image optical encryption method of jtc system combining cgh and frequency shift. *Infrared Laser Eng* (2022) 51:20220175–1. doi:10.3788/irla20220175

62. Zhang B, Liu L. A novel fast image encryption algorithm based on coefficient independent coupled exponential chaotic map. *Physica Scripta* (2024) 99:025249. doi:10.1088/1402-4896/ad1fc3

63. Ning Y, Jin J, Li Z, Chen C, Ouyang A. A time-varying hill cipher for dynamic image cryptography. *Tsinghua Sci Technology* (2025). doi:10.26599/TST.2024.9010213

64. Wu X, Wang D, Kurths J, Kan H. A novel lossless color image encryption scheme using 2d dwt and 6d hyperchaotic system. *Inf Sci* (2016) 349:137–53. doi:10.1016/j.ins.2016.02.041

65. Xu M, Tian Z. A novel image cipher based on 3d bit matrix and Latin cubes. *Inf Sci* (2019) 478:1–14. doi:10.1016/j.ins.2018.11.010

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 26 June 2025
DOI 10.3389/fphy.2025.1614983

EDITED BY
Hairong Lin,
Central South University, China

REVIEWED BY
Feifei Yang,
Xi'an University of Science and Technology, China
Shuang Zhou,
Chongqing Normal University, China

*CORRESPONDENCE
Feng Peng,
✉ pengfeng@dlpu.edu.cn
Yinghong Cao,
✉ caoyinghong@dlpu.edu.cn

RECEIVED 23 May 2025
ACCEPTED 11 June 2025
PUBLISHED 26 June 2025

CITATION
Huang L, Peng F, Huang B and Cao Y (2025) HiImp-SMI: an implicit transformer framework with high-frequency adapter for medical image segmentation.
Front. Phys. 13:1614983.
doi: 10.3389/fphy.2025.1614983

HiImp-SMI: an implicit transformer framework with high-frequency adapter for medical image segmentation

Lianchao Huang[1], Feng Peng[2]*, Binghao Huang[1] and Yinghong Cao[3]*

[1]School of Information Science and Engineering, Dalian Polytechnic University, Dalian, China, [2]Information Technology Center, Dalian Polytechnic University, Dalian, China, [3]School of Biological Engineering, Dalian Polytechnic University, Dalian, China

Accurate and generalizable segmentation of medical images remains a challenging task due to boundary ambiguity and variations across domains. In this paper, an implicit transformer framework with a high-frequency adapter for medical image segmentation (HiImp-SMI) is proposed. A new dual-branch architecture is designed to simultaneously process spatial and frequency information, enhancing both boundary refinement and domain adaptability. Specifically, a Channel Attention Block selectively amplifies high-frequency boundary cues, improving contour delineation. A Multi-Branch Cross-Attention Block facilitates efficient hierarchical feature fusion, addressing challenges in multi-scale representation.Additionally, a ViT-Conv Fusion Block adaptively integrates global contextual awareness from Transformer features with local structural details, thereby significantly boosting cross-domain generalization. The entire network is trained in a supervised end-to-end manner, with frequency-adaptive modules integrated into the encoding stages of the Transformer backbone. Experimental evaluations show that HiImp-SMI consistently outperforms mainstream models on the Kvasir-Sessile and BCV datasets, including state-of-the-art implicit methods. For example, on the Kvasir-Sessile dataset, HiImp-SMI achieves a Dice score of 92.39%, outperforming I-MedSAM by 1%. On BCV, it demonstrates robust multi-class segmentation with consistent superiority across organs. These quantitative results demonstrate the framework's effectiveness in refining boundary precision, optimizing multi-scale feature representation, and improving cross-dataset generalization. This improvement is largely attributed to the dual-branch design and the integration of frequency-aware attention mechanisms, which enable the model to capture both anatomical details and domain-robust features. The proposed framework may serve as a flexible baseline for future work involving implicit modeling and multi-modal representation learning in medical image analysis.

KEYWORDS

nonlinear system, medical image segmentation, high-frequency adapter, cross-attention, feature fusion

1 Introduction

Medical image segmentation plays a crucial role in assisting disease diagnosis and guiding clinical treatment. Traditional discrete methods based on convolutional neural networks (CNNs), such as U-Net [1], nnU-Net [2], and PraNet [3], effectively integrate multi-scale features but remain highly sensitive to variations in data distribution, thus limiting cross-domain generalization. Although boundary-aware methods, such as Boundary-aware U-Net [4], WM-DOVA [5], Hausdorff distance-based approaches [6], dropout-based calibration [7], and neural network calibration [8], have improved localization precision and feature representation, these methods still face challenges when dealing with complex medical structures and achieving consistent segmentation performance across different domains. Additionally, multi-scale residual architectures like Res2Net [9] further enhance feature representation but are still limited in boundary preservation.

Recent developments have introduced Transformer-based architectures, such as TransUNet [10] and UNETR [11], leveraging global contextual awareness through self-attention mechanisms [12]. Despite superior global feature capture capabilities, these approaches often underperform in local boundary refinement and require extensive training data for effective generalization. Further advancements, such as LoRA [13], aim to improve Transformer efficiency and generalization but do not explicitly optimize for boundary segmentation accuracy. Furthermore, adaptations based on the Segment Anything Model (SAM) [14], including MedSAM [15], SAM-based 3D extensions [16], and customized SAM models [17], generally improve generalization capabilities but typically neglect fine-grained feature integration, resulting in limited boundary segmentation accuracy. Additional SAM-related studies, such as NTo3D [18], Customized SAM [19], SAM-Med2D [20], DiffDP [21], spatial prior-based approaches [22], and mask-enhanced SAM models [23], have explored further improvements but continue to face challenges with boundary precision.

Beyond conventional deep learning approaches, emerging research spans several interdisciplinary directions that address these challenges. For instance, memristor- and memcapacitor-based neural network models have been proposed to enable neuromorphic hardware implementations [24, 25]; such analog in-memory circuits have demonstrated improved image segmentation speed and accuracy via parallel high-efficiency computations [26, 27]. Recent studies have further explored Hamiltonian conservative chaotic systems integrated with memristors for modeling and FPGA implementation, enhancing the physical interpretability and stability of neuromorphic designs [28]. Similarly, chaotic and hyperchaotic dynamical systems have been exploited in image encryption, leveraging their high-dimensional unpredictability to enhance security. In particular, memristor-coupled cellular neural networks based on resonant tunneling diodes have been applied in forensic digital image protection, offering a secure hardware foundation for sensitive applications [29]. Some studies even integrate memristive chaotic circuits to strengthen resistance against differential attacks [30], and in general hyper-chaos offers greater randomness and key space than lower-dimensional maps [31], yielding encryption schemes with robust immunity to cryptanalytic attacks [32]. Other researchers have implemented novel hyperchaotic systems in FPGA to support audio encryption, demonstrating the practical deployment of such dynamics on low-power reconfigurable hardware [33, 34]. In IoT contexts, researchers have developed lightweight image encryption and steganography techniques to secure multimedia data with minimal computational overhead [35, 36], addressing the limitations of earlier cryptosystems on resource-constrained devices [37]. Moreover, discrete n-dimensional hyperchaotic maps with customizable Lyapunov exponents have been proposed to expand the design space for secure communications and embedded cryptography [38]. Additionally, integrating multi-modal information has become crucial for improving diagnostic accuracy, prompting new architectures that effectively fuse heterogeneous medical data streams [39, 40]. Equally important, domain-generalization strategies are being pursued to ensure models remain robust across disparate imaging domains, tackling the severe performance degradation caused by cross-modality shifts without requiring retraining on target data [41]. Finally, a concerted effort is underway to translate these advances into practical deployments: specialized DSP-based accelerators and other hardware implementations are achieving real-time image processing with low power consumption [42, 43], and even complex neuromorphic networks are being prototyped on DSP platforms [25, 26]. These developments across hardware design, secure encryption, lightweight algorithms, and multi-modal learning collectively strengthen the foundation for next-generation medical image segmentation systems.

Implicit neural representation methods represent another advancement, employing continuous mappings from coordinate spaces to representation spaces, exemplified by OSSNet [44], IOSNet [45], and SWIPE [46]. These models exhibit improved segmentation robustness across resolutions but remain constrained by their reliance on traditional convolutional encoders, limiting their capacity to simultaneously capture detailed boundary information and global contextual features. Further implicit methods, including NeRF [47], NUDF [48], NISF [49], ImplicitAtlas [50], implicit neural representations survey [51], shape reconstruction from sparse measurements [52], implicit functions for 3D reconstruction [53], MRI super-resolution [16], and volumetric SAM adaptations [54], have significant potential but share similar limitations. Frequency-domain adapters, like those in I-MedSAM [55], have enhanced boundary delineation, but single-adapter designs remain insufficient for comprehensive multi-scale feature integration.

To address these challenges, this study introduces HiImp-SMI, an implicit Transformer-based medical image segmentation framework incorporating three key innovations: (1) a Channel Attention Block to explicitly enhance high-frequency boundary information, (2) a Multi-Branch Cross-Attention Block to facilitate efficient hierarchical feature fusion across different scales, and (3) a ViT-Conv Fusion Block designed to integrate global context from Transformer-based architectures with local fine-grained features extracted by convolutional networks. Experimental validations conducted on the Kvasir-Sessile and BCV datasets demonstrate that HiImp-SMI outperforms existing segmentation methods, highlighting its effectiveness in boundary precision, multi-scale feature representation, and cross-dataset generalization capabilities.

The remainder of this paper is organized as follows: Section 2 details the proposed HiImp-SMI framework; Section 3 presents the experimental setup and results; and Section 4 concludes the study, providing directions for future research.

2 Materials and methods

The overall architecture of the proposed HiImp-SMI framework is depicted in Figure 1. It comprises a dual-branch encoder structure that jointly exploits spatial-domain and frequency-domain information. Given an input image I, a Fast Fourier Transform (FFT) is applied to derive its frequency representation I_{FFT}, which highlights high-frequency components corresponding to anatomical boundaries and texture transitions. By integrating I_{FFT} into the encoder, our Channel Attention Block can selectively amplify boundary-sensitive features, enhancing fine-grained localization and generalization to unseen domains. These embeddings are then processed by three key modules: a Channel Attention Block, which selectively enhances high-frequency boundary details; a Multi-Branch Cross Attention Block, designed to enable effective feature exchange across hierarchical levels; and a ViT-Conv Fusion Block, which adaptively integrates global contextual information from the Transformer branch and local structural features from the convolutional branch. Through this architecture, HiImp-SMI aims to achieve more precise boundary segmentation, stronger multi-scale representation, and enhanced cross-domain generalization.

2.1 Channel attention block

In this study, SAM employs a Vision Transformer (ViT) as the image encoder, pretrained on a large-scale natural image dataset. To preserve the strong feature representation capability of the pretrained ViT, its weights are kept frozen during training. Instead, a local adapter module is introduced to incorporate localized inductive biases into the model, as illustrated in Figure 2.

The Channel Attention Block enhances the domain-specific feature extraction capability of the pretrained Vision Transformer (ViT) without fine-tuning its weights. The procedure involves the following steps:

Step 1: Obtain the input embedding F_{vit} from the ViT attention block. This embedding carries high-level semantic features. It serves as the input to the channel attention block.

Step 2: Apply layer normalization (LN) to stabilize feature distributions. LN normalizes each channel to reduce internal covariate shift. This improves training stability and convergence.

Step 3: Perform a pointwise convolution ($\mathrm{Conv}_{1\times1}$) to adjust channel dimensions. This operation projects features into a latent space. It preserves spatial structure while enabling channel-wise transformation.

Step 4: Execute a depthwise convolution ($\mathrm{DWConv}_{3\times3}$) to capture spatial information. Each channel is convolved independently to extract local patterns. This enhances spatial modeling without increasing parameter count significantly.

Step 5: Apply a Squeeze-and-Excitation (SE) block to model channel-wise dependencies. Specifically, the SE block performs global average pooling followed by two fully connected layers and non-linear activations to generate a channel attention vector s, which is then applied to recalibrate the feature map, as shown in Equation 1:

$$\begin{cases} z = \dfrac{1}{H\times W}\displaystyle\sum_{i=1}^{H}\sum_{j=1}^{W}F_{ij} \\ s = \sigma\left(W_2\cdot\delta\left(W_1\cdot z\right)\right) \\ \mathrm{SE}\left(F\right) = F\otimes s \end{cases} \tag{1}$$

Here, $F\in\mathbb{R}^{C\times H\times W}$ denotes the input feature map, and $z\in\mathbb{R}^{C}$ is the channel-wise descriptor obtained by global average pooling. W_1 and W_2 are learnable weight matrices of two fully connected layers. $\delta(\cdot)$ and $\sigma(\cdot)$ denote the ReLU and sigmoid activation functions, respectively. The resulting attention vector s is used to rescale each channel of F via element-wise multiplication, enabling adaptive channel emphasis.

Step 6: Integrate the processed features using another pointwise convolution (Conv1 × 1) to obtain refined embedding $\widehat{F}$vit, as defined in Equation 2:

$$\widehat{F}\mathrm{vit} = \mathrm{Conv}1\times1\left(\mathrm{SE}\left(\mathrm{DWConv}3\times3\left(\mathrm{Conv}1\times1\left(\mathrm{LN}\left(F_{\mathrm{vit}}\right)\right)\right)\right)\right) \tag{2}$$

Step 7: Merge the refined features with the original features through a residual connection, as formulated in Equation 3:

$$F_{\mathrm{out}} = F_{\mathrm{vit}} + \widehat{F}_{\mathrm{vit}} \tag{3}$$

2.2 Multi-branch Cross Attention Block

Figure 3 illustrates the structure of the Multi-branch Cross Attention Block, which integrates deep features from the ViT branch with shallow features from a convolutional branch. The procedure involves the following steps:

Step 1: Extract shallow features (F_s) from the resized input image using a lightweight convolutional block. This step captures low-level visual patterns such as edges and textures. The convolutional block is designed to be efficient for early-stage feature extraction.

Step 2: Generate queries, keys, and values for the ViT branch and convolutional branch separately, as described in Equation 4:

$$\begin{cases} Q_d = W_q^dF_d, \quad K_d = W_k^d\left[F_b;F_s\right], \quad V_d = W_v^dF_s \\ Q_s = W_q^sF_s, \quad K_s = W_k^s\left[F_b;F_d\right], \quad V_s = W_v^sF_d \end{cases} \tag{4}$$

Here, F_d and F_s denote deep features from the ViT branch and shallow features from the convolutional branch, respectively. F_b represents bottleneck features shared across branches. W_q, W_k, and W_v are learnable linear projection matrices used to obtain queries (Q), keys (K), and values (V) for attention computation.

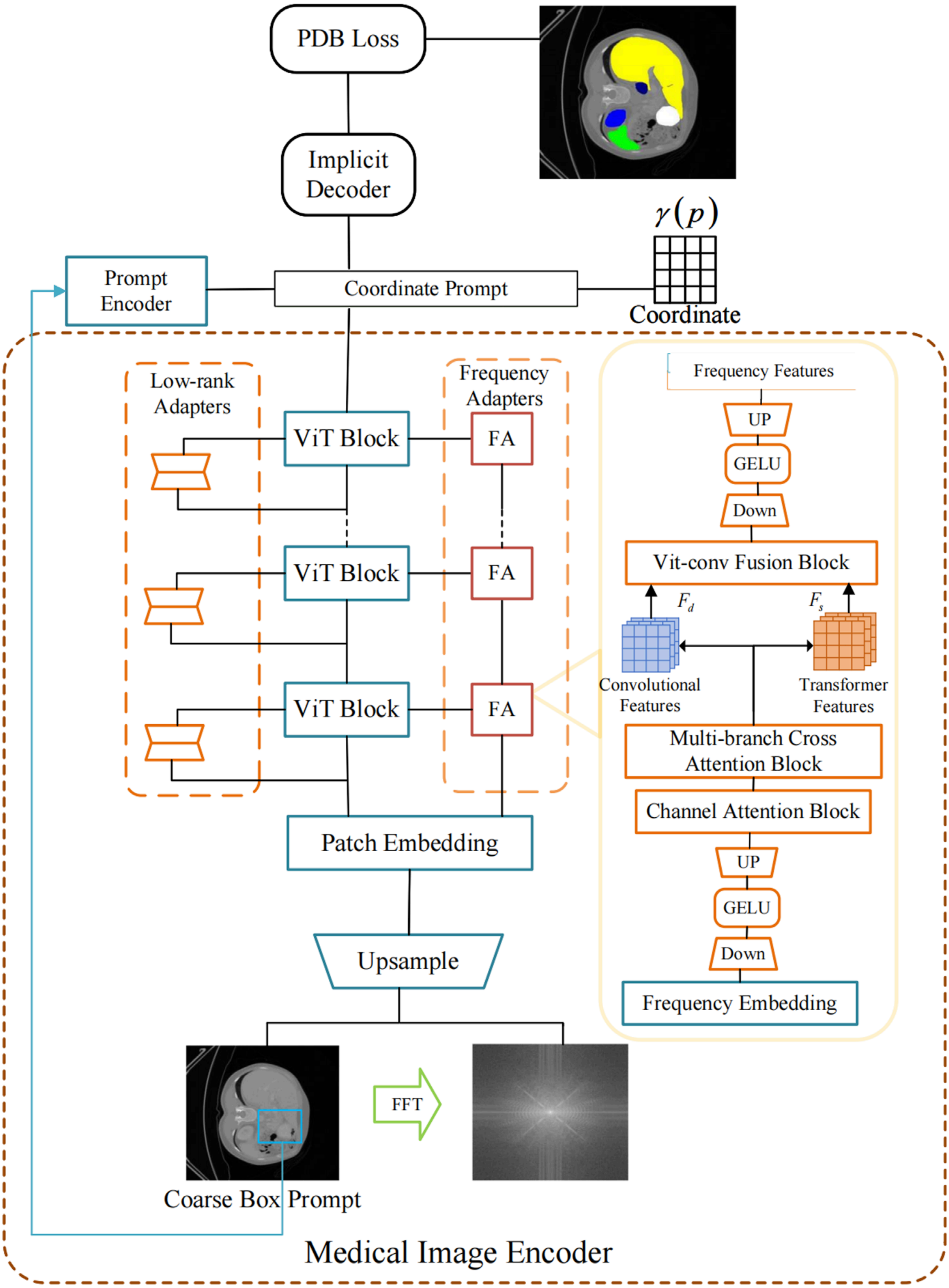


FIGURE 1
Overall architecture of our proposed model.

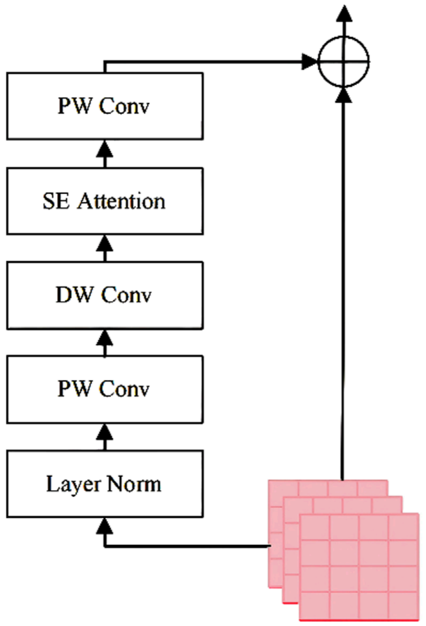


FIGURE 2
The Channel Attention Block for domain-specific feature enhancement in the ViT encoder.

Step 3: Fuse features across branches using deformable attention, detailed in Equation 5:

$$\begin{cases} F_d^c = \mathrm{DeformAttn}\left(Q_d, K_d, V_d\right) \\ F_s^c = \mathrm{DeformAttn}\left(Q_s, K_s, V_s\right) \end{cases} \tag{5}$$

Here, F_d^c and F_s^c represent the cross-attended features refined via deformable attention in the ViT and convolutional branches, respectively. Deformable attention adaptively samples spatial locations, enabling the model to focus on semantically relevant regions. This mechanism facilitates more effective feature alignment across the two branches.

Step 4: Refine the fused features with residual feedforward networks (FFN) and layer normalization (LN)—this refinement is formalized in Equation 6:

$$\begin{cases} F_d^1 = \mathrm{FFN}\left(\mathrm{LN}\left(F_d + F_d^c\right)\right) + \left(F_d + F_d^c\right) \\ F_s^1 = \mathrm{FFN}\left(\mathrm{LN}\left(F_s + F_s^c\right)\right) + \left(F_s + F_s^c\right) \end{cases} \tag{6}$$

Here, F_d^1 and F_s^1 denote the updated deep and shallow features after refinement. The FFN enhances non-linear representation capacity, while LN improves training stability. The residual connection facilitates efficient information preservation and gradient flow.

2.3 ViT-Conv fusion block

A fusion block equipped with an automatic selection mechanism is constructed to integrate the diverse information provided by convolutional features and Transformer features. The architectural details of this module are illustrated in Figure 4.

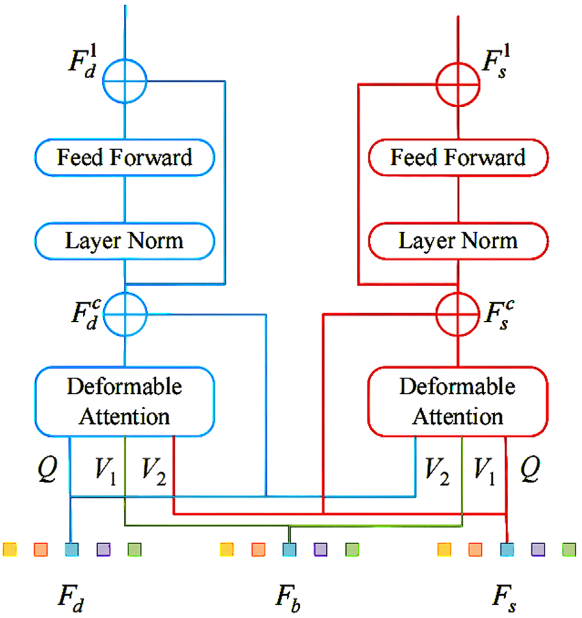


FIGURE 3
The Multi-branch Cross Attention Block for fusing ViT and convolutional features via cross-attention.

The ViT-Conv Fusion Block adaptively integrates convolutional and Transformer features through these steps:

Step 1: Process deep (F_d) and shallow (F_s) features individually with a channel attention layer to obtain logits (φ_d, φ_s). Channel attention highlights informative channels in each branch. This yields two attention logits representing the feature importance.

Step 2: Aggregate logits from both branches to compute an element-wise selection mask using a sigmoid function. Equation 7 defines this aggregation process.

$$\omega = \mathrm{Sigmoid}\left(\varphi_d + \varphi_s\right) \tag{7}$$

Here, ω denotes the attention-based selection mask used to balance feature contributions from the two branches. The summed logits $\varphi_d + \varphi_s$ capture joint channel importance. The sigmoid function constrains the mask values between 0 and 1, enabling soft feature weighting.

Step 3: Compute the final fused output via element-wise multiplication, as specified in Equation 8:

$$F_{\mathrm{output}} = F_d^o \otimes \omega + F_s^o \otimes (1 - \omega) \tag{8}$$

Here, F_d^o and F_s^o represent the output features from the Transformer and convolutional branches, respectively. F_{output} denotes the final fused representation. The selection mask ω adaptively controls the

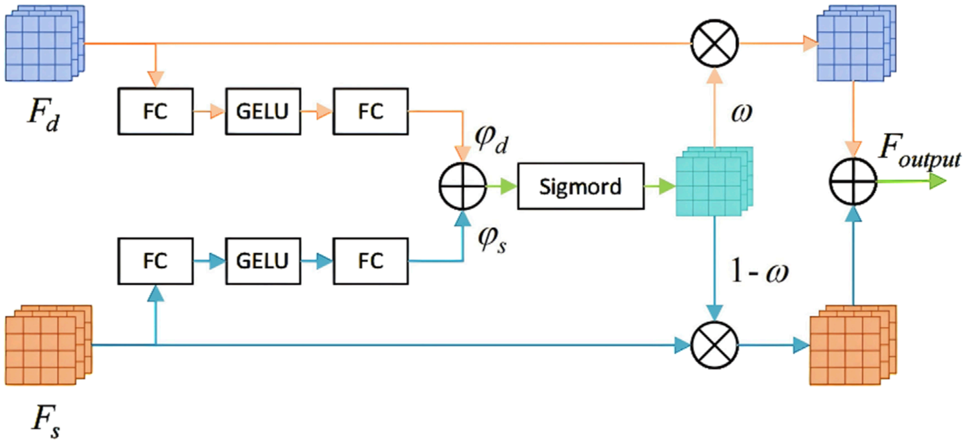


FIGURE 4
The ViT-Conv Fusion Block for adaptive integration of Transformer and convolutional features.

contribution of each branch, enabling dynamic integration of global and local information.

2.4 Loss function

To supervise both the coarse and fine segmentation branches during training, a Progressive Dual-Branch Loss (PDB Loss) is proposed. This loss function dynamically adjusts the supervision weights between the coarse and fine predictions over training epochs. The total training loss is precisely defined by Equation 9:

$$\mathcal{L}_{\text{PDB}} = \frac{1}{B}\sum_{i=1}^{B}\left[(1-\alpha)\cdot\mathcal{L}_{\text{DiceCE}}\left(\hat{y}^{(i)}_{\text{coarse}}, y^{(i)}\right) + \alpha\cdot\mathcal{L}_{\text{DiceCE}}\left(\hat{y}^{(i)}_{\text{fine}}, y^{(i)}\right)\right] \tag{9}$$

Here, $\hat{y}^{(i)}_{\text{coarse}}$ and $\hat{y}^{(i)}_{\text{fine}}$ are the predicted masks from the coarse and fine branches for the i-th sample, and $y^{(i)}$ is the corresponding ground truth. B denotes the batch size. $\alpha \in [0,1]$ is a progressive weight that determines the relative contribution of the fine branch.

For each prediction, a hybrid loss combining Dice and binary cross-entropy (BCE) is used, aspresented in Equation 10:

$$\mathcal{L}_{\text{DiceCE}}(\hat{y}, y) = \lambda_{\text{dice}}\cdot\mathcal{L}_{\text{Dice}}(\hat{y}, y) + \lambda_{\text{ce}}\cdot\mathcal{L}_{\text{CE}}(\hat{y}, y) \tag{10}$$

The loss weights were set as $\lambda_{dice} = 0.8$ and $\lambda_{ce} = 0.2$. To shift the learning focus from coarse to fine predictions over time, the coefficient α was scheduled according to the current epoch t as given in Equation 11:

$$\alpha = \min\left(\frac{t+1}{5}, 1.0\right) \tag{11}$$

This progressive weighting strategy encourages the model to learn global structural features in early epochs via the coarse branch and gradually refine local boundaries and details through the fine branch.

3 Experiments

In this section, a series of comprehensive experiments is performed to evaluate the effectiveness of the proposed HiImp-SMI on medical image segmentation tasks. Initially, the experimental setup is detailed, including dataset selection and training configurations. Subsequently, the performance of HiImp-SMI is quantitatively and qualitatively compared with state-of-the-art implicit and discrete segmentation approaches, specifically addressing binary polyp segmentation on the Kvasir-Sessile dataset [13] and multi-class organ segmentation on the BCV dataset [56]. Additionally, robustness analyses under various data distributions are presented. Finally, a systematic ablation study is conducted to elucidate the contributions of individual modules within HiImp-SMI.

The quantitative comparison results are summarized in Table 1, highlighting mean Dice and IoU scores alongside corresponding standard deviations. The best-performing methods are emphasized in bold, illustrating that HiImp-SMI consistently achieves superior segmentation performance compared to existing state-of-the-art methods.

3.1 Experimental setup

The model's performance is evaluated on two distinct medical image segmentation tasks: binary polyp segmentation and multi-class abdominal organ segmentation.

For polyp segmentation, experiments are conducted on the challenging Kvasir-Sessile dataset [13], which contains 196 RGB images of small sessile polyps. To assess the generalization capability of HiImp-SMI, the pretrained model is further evaluated on the CVC-ClinicDB dataset [13], which consists of 612 images extracted from 31 colonoscopy sequences.

For multi-organ segmentation, the model is trained on the BCV dataset [56], which includes 30 CT scans with annotations for 13 organs, and is further evaluated on the AMOS dataset [57], which contains 200 CT training samples, following the same

TABLE 1 Overall segmentation results compared to state-of-the-art discrete and implicit methods. The last two columns present the mean Dice and IoU scores with standard deviation. The best results are highlighted in bold.

Method type	Method	Kvasir-sessile		BCV	
		Dice(%)	IoU(%)	Dice(%)	IoU(%)
Discrete	U-Net [1]	63.89±1.30	46.94±0.65	74.47±1.57	59.32±0.79
	PraNet [3]	82.56±1.08	70.3±0.54	N/A	N/A
	UNETR [11]	N/A	N/A	81.14±0.85	68.27±0.43
	Res2UNet [9]	81.62±0.97	68.95±0.49	79.23±0.66	65.6±0.33
	NnUNet [2]	82.97±0.89	70.9±0.45	85.15±0.67	74.14±0.34
	MedSAM [15]	82.88±0.55	70.77±0.28	85.85±0.81	75.21±0.41
Implicit	OSSNet [44]	76.11±1.14	61.43±0.57	73.38±1.65	57.95±0.83
	IOSNet [45]	78.37±0.76	64.43±0.38	76.75±1.37	62.27±0.69
	SWIPE [46]	85.05±0.82	73.99±0.41	81.21±0.94	68.36±0.47
	I-MedSAM [55]	91.49±0.52	84.31±0.26	89.91±0.68	81.67±0.34
	HiImp-SMI (Ours)	**92.39±0.36**	**85.86±0.18**	**91.21±0.31**	**83.84±0.16**

Note. "N/A" indicates that the corresponding experiment was not conducted.
Bold values indicate the best performance for each metric.

experimental setup as [22]. Since this study focuses on 2D medical image segmentation, slice-wise segmentation is performed on CT images. Following the data preprocessing strategy of SWIPE [46], all datasets are split into training, validation, and test sets in a 6:2:2 ratio, and the reported Dice scores are based on test set results.

The training process involves fine-tuning the SAM encoder [7] with ViT-B as the backbone network. The LoRA rank is set to 4, with amplitude information incorporated in the frequency adapter. The MLP dimensions for the implicit segmentation decoder are [1,024, 512] for Decc and [512, 256, 256, 128] for Decf. During training, 12.5% of the most uncertain points are sampled for refinement, and the dropout probability is set to 0.5. For the multi-organ segmentation task, the final layer of Decc and Decf is adjusted to match the number of target segmentation classes. HiImp-SMI is optimized using AdamW [58] with $\alpha = 0.5, \beta = 0.1$, a learning rate of $\lambda_{ada} = 5 \times 10^{-5}$ for the encoder adapter, and $\lambda_{dec} = 1 \times 10^{-3}$ for the decoder.

To ensure fair comparison, all methods are trained for 1,000 epochs under the same experimental setup. During testing, Dice scores and Hausdorff distances [6] are reported based on the best validation epoch. The input image resolutions are set to 384×384 (Sessile dataset) and 512×512 (BCV dataset slices).

The baseline approaches are categorized into discrete methods and implicit (continuous) methods. The discrete methods include U-Net [1], PraNet [3], Res2UNet [9], nnUNet [2], UNETR [11], and MedSAM [15]. Among these, MedSAM [15] is also a SAM-based approach, where the original decoder is directly fine-tuned. The implicit methods include OSSNet [44], IOSNet [45], and SWIPE [46] and I-MedSAM [55].

3.2 Quantitative comparison

A Dice score comparison is first presented against baseline methods. Subsequently, experiments are conducted across different resolutions and domains to evaluate the model's cross-domain generalization ability under data distribution shifts. Finally, Hausdorff Distance (HD) [6] is computed to compare the segmentation boundary quality across different experimental settings.

Discrete methods and implicit methods are compared in terms of trainable parameters and Dice scores (including standard deviation). Specifically, binary segmentation is performed on the Kvasir-Sessile dataset, while multi-class segmentation is conducted on the CT BCV dataset, with results detailed in Table 2. Leveraging the proposed frequency adapter, SAM generates richer feature representations, leading to improved segmentation boundary quality. In contrast, SwIPE, which employs Res2Net-50 [9] as its backbone, exhibits weaker feature extraction capability, resulting in lower segmentation quality.

The adaptability of binary polyp segmentation across different resolutions and domains is assessed by comparing it with the best-performing discrete and implicit methods. To adapt to different target resolutions (e.g., low resolution 128×128 and high resolution 896×896), the pretrained HiImp-SMI model, initially trained at 384×384 standard resolution, is modified by scaling the input coordinates to match the target resolution, and the corresponding Dice scores are computed. For discrete methods, the output resolution remains consistent with the input resolution. Input images at the original resolution of 384×384 are provided, and the generated segmentation results are rescaled to the target resolution

TABLE 2 Cross-resolution evaluation from 384 × 384 to 128 × 128 and from 384 × 384 to 896 × 896.

Method type	Method	384 × 384→128 × 128		384 × 384→896 × 896	
		Dice(%)	IoU(%)	Dice(%)	IoU(%)
Discrete	PraNet [3]	72.64	57.04	74.95	59.94
	PraNet* [3]	68.79	52.43	43.92	28.14
	nnUNet [2]	73.97	58.69	83.56	71.76
	nnUNet* [2]	65.34	48.52	76.36	61.76
	MedSAM [15]	82.39	70.05	83.56	71.76
Implicit	IOSNet [45]	78.37	64.43	78.01	63.95
	SWIPE [46]	81.26	68.44	84.33	72.91
	I-MedSAM [55]	91.45	84.25	91.33	84.04
	HiImp-SMI (ours)	**92.52**	**86.08**	**92.28**	**85.67**

Bold values indicate the best performance for each metric.

TABLE 3 Cross-domain results for binary polyp segmentation and multi-class abdominal organ segmentation.

Method type	Method	Kvasir→CVC		BCV→AMOS	
		Dice(%)	IoU(%)	Dice(%)	IoU(%)
Discrete	PraNet [3]	68.37	51.94	N/A	N/A
	UNETR [11]	N/A	N/A	81.75	69.13
	nnUNet [2]	84.91	73.78	79.63	66.15
	MedSAM [15]	74.59	59.48	71.98	56.23
Implicit	IOSNet [45]	59.42	42.27	79.48	65.95
	SWIPE [46]	70.1	53.96	82.81	70.66
	I-MedSAM [55]	88.83	79.9	86.28	75.87
	HiImp-SMI (ours)	**91.58**	**84.47**	**88.17**	**78.84**

Note.“N/A″ indicates that the corresponding experiment was not conducted.
Bold values indicate the best performance for each metric.

TABLE 4 HD distance (↓) for different methods and datasets.

Method	Kvasir-sessile	Kvasir→CVC	384→128	384→896	BCV	BCV→AMOS
nnUNet [2]	31.30	82.31	13.69	72.31	6.50	80.39
MedSAM [15]	21.53	30.15	8.04	51.82	10.62	52.14
IOSNet [45]	51.72	81.60	35.33	87.86	21.46	61.19
I-MedSAM [55]	11.59	**19.76**	7.91	32.77	5.95	**37.53**
HiImp-SMI (ours)	**10.48**	20.30	**3.60**	**24.52**	**4.97**	38.12

Bold values indicate the best performance for each metric.

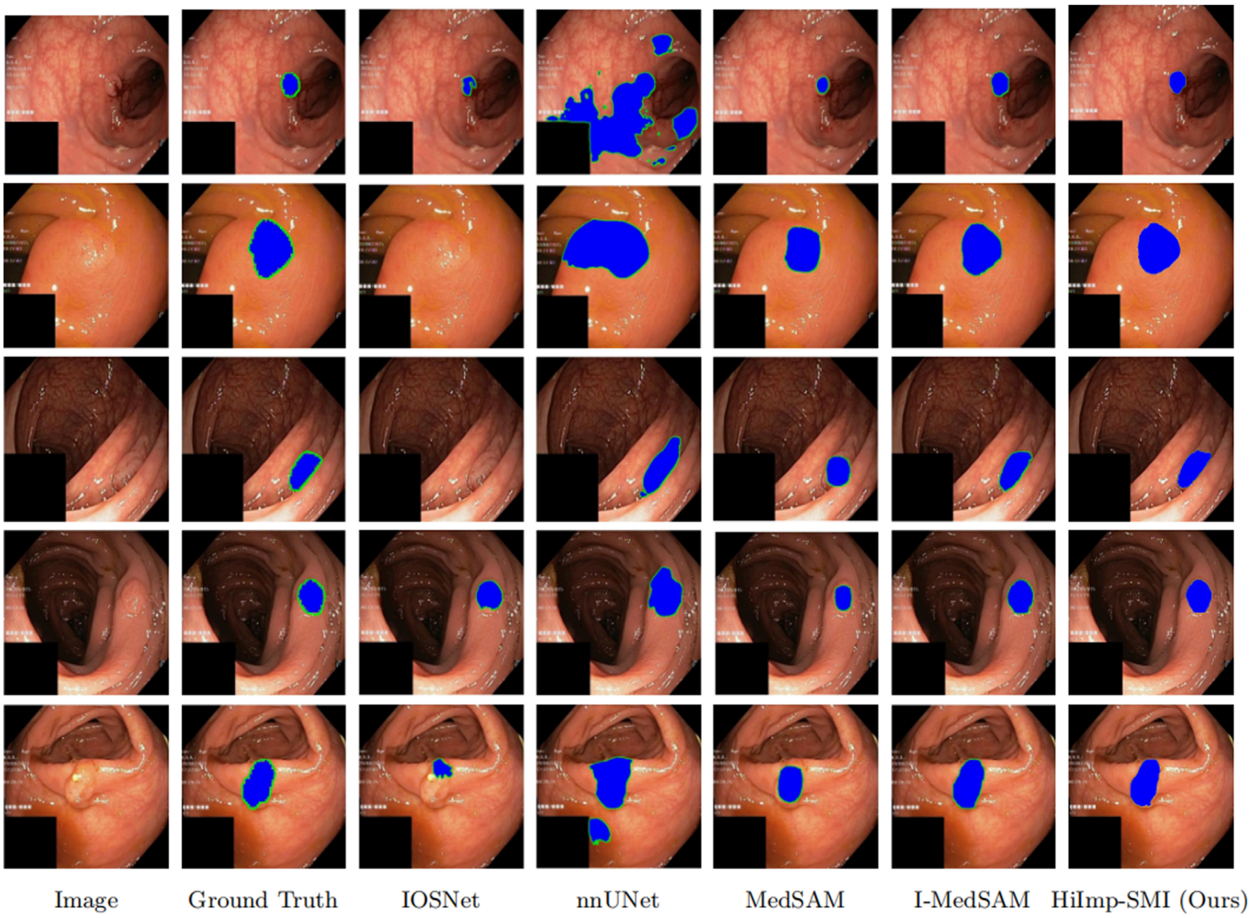


FIGURE 5
Qualitative comparisons on five representative samples. The last row indicates the method names corresponding to each column.

TABLE 5 Ablation study on the integration of different modules: Channel Attention Block (CAB), Multi-branch Cross Attention Block (MCAB), and ViT-Conv Fusion Block (VCFB). Evaluation is conducted on the Kvasir-Sessile dataset and its cross-domain transfer to the CVC dataset.

Modules			Kvasir-sessile			Kvasir-sessile → CVC		
CAB	MCAB	VCFB	Dice (%) ↑	HD ↓	IoU (%) ↑	Dice (%) ↑	HD ↓	IoU (%) ↑
			91.81	11.80	84.86	89.07	24.06	80.29
✓			92.02	11.28	85.22	88.94	24.66	80.08
✓	✓		92.42	11.50	85.91	88.87	22.12	79.97
✓	✓	✓	**92.51**	**9.98**	**86.06**	**91.46**	**21.03**	**84.26**

Bold values indicate the best performance for each metric.

for evaluation. Additionally, the suffix (*) is used to mark discrete baselines, where the original medical images are resized to the target resolution before being fed into the models, allowing these methods to directly generate segmentation results at the target resolution.

As shown in Table 2, implicit methods exhibit stronger adaptability to spatial resolution changes and consistently outperform discrete methods. Among implicit methods, HiImp-SMI achieves the highest performance across different output resolutions, which can be attributed to the proposed frequency adapter, enhancing HiImp-SMI's predictive capability across resolutions.

Model performance across different datasets is examined. In binary polyp segmentation, all methods are pretrained on the Kvasir-Sessile dataset and directly evaluated on the CVC dataset. Similarly, in multi-class abdominal organ segmentation, all methods are pretrained on the BCV dataset and evaluated on the AMOS dataset, focusing exclusively on the liver class.

As shown in Table 3, leveraging SAM's generalization ability, HiImp-SMI outperforms the best discrete method, achieving Dice scores of 91.58% on the CVC dataset and 88.17% on the AMOS dataset.

Segmentation boundary quality is further assessed using Hausdorff Distance (HD) [19]. As shown in Table 4, HiImp-SMI achieves lower HD scores, indicating superior boundary precision compared to existing methods.

3.3 Qualitative comparison

As shown in Figure 5, a qualitative comparison is conducted on the Kvasir-Sessile dataset. Additionally, the input medical images and their corresponding ground truth segmentation masks are provided, where segmentation boundaries are highlighted in green in Figure 5. The sharpness of boundaries in the visual results may be attributed in part to the frequency-domain information introduced via FFT.

From the results, it is evident that HiImp-SMI produces more precise segmentation boundaries. By leveraging the proposed modules, HiImp-SMI effectively aggregates high-frequency information from the input, leading to improved segmentation accuracy in the final output.

3.4 Ablation study

An ablation study is conducted to evaluate the effectiveness of each module within the high-frequency adapter. The results are summarized in Table 5.

In the baseline model, the single frequency adapter module consists of a linear down-projection layer, a GELU activation function, and a linear up-projection layer. On the Kvasir-Sessile dataset [8], the baseline model achieves a Dice score of 91.81% and an HD of 11.80. When transferred to the CVC dataset, the Dice score drops to 89.07%, with an HD of 24.06.

As the channel attention block, bi-directional cross-attention block, and ViT-Conv fusion block are incrementally added, model performance exhibits a significant improvement. When all three modules are incorporated, the Dice score on the Kvasir-Sessile dataset improves to 92.51%, while HD decreases to 9.98. Similarly, on the CVC dataset, the Dice score improves to 91.46%, and HD decreases to 21.03, highlighting the necessity and effectiveness of the proposed modules.

4 Conclusion

In this study, a novel implicit Transformer-based framework, HiImp-SMI, was proposed to overcome key limitations in medical image segmentation, such as poor boundary refinement, weak feature fusion, and limited cross-domain generalization. High-frequency information and multi-scale features were incorporated through three main components: a Channel Attention Block for frequency-domain feature adaptation, a Multi-Branch Cross Attention Block for hierarchical feature exchange, and a ViT-Conv Fusion Block for adaptive context integration. Additionally, a Progressive Dual-Branch Loss was introduced to guide the training process from coarse to fine segmentation. Extensive experiments conducted on the Kvasir-Sessile and BCV datasets demonstrated that HiImp-SMI consistently outperformed state-of-the-art methods, particularly in cross-domain and cross-resolution tasks. Ablation studies further confirmed the effectiveness of each proposed module.

However, the current framework has not yet been validated in clinical or multi-center settings. Future research will aim to evaluate its applicability in real-world clinical workflows.

Overall, HiImp-SMI provided a unified and adaptive solution for precise and generalizable medical image segmentation.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: Kvasir-Sessile Dataset: https://datasets.simula.no/kvasir/ Repository: Simula Research Laboratory Accession Number: Not applicable (open access dataset) CVC-ClinicDB: https://github.com/CVC-ClinicDB Repository: GitHub Accession Number: Not applicableBCV (Beyond Cranial Vault) Dataset: https://www.synapse.org/#!Synapse:syn3193805 Repository: Synapse Accession Number: syn3193805AMOS Dataset: https://amos22.grand-challenge.org/ Repository: Grand Challenge Accession Number: Not applicable.

Author contributions

LH: Conceptualization, Methodology, Validation, Writing – original draft, Supervision, Writing – review and editing, Data curation. FP: Supervision, Writing – review and editing. BH: Data curation, Writing – review and editing, Investigation. YC: Supervision, Writing – review and editing, Methodology, Project administration.

Funding

The author(s) declare that financial support was received for the research and/or publication of this article. This work was supported by the Liaoning Provincial Science and Technology Plan Joint Project (Grant No. 2024-MSLH-033).

Acknowledgments

Many thanks to Yinghong Cao and Feng Peng for their help in achieving this work.

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.

Generative AI statement

The author(s) declare that no Generative AI was used in the creation of this manuscript.

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. Ronneberger O, Fischer P, Brox T. U-net: convolutional networks for biomedical image segmentation. In: N Navab, J Hornegger, WM Wells, AF Frangi, editors. Proceedings of the 18th International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI 2015), 9351. Cham, Switzerland: Springer (2015). p. 234–41. doi:10.1007/978-3-319-24574-4_28

2. Isensee F, Jäger PF, Kohl SAA, Petersen J, Maier-Hein KH. Automated design of deep learning methods for biomedical image segmentation. *arXiv preprint arXiv:1904* (2019):08128. doi:10.48550/arXiv.1904.08128

3. Fan DP, Ji GP, Zhou T, Chen G, Fu H, Shen J, et al. Pranet: parallel reverse attention network for polyp segmentation. In: Proceedings of the 23rd International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI 2020), 12266. Cham, Switzerland: Springer (2020). p. 263–73. doi:10.1007/978-3-030-59725-2_26*Lecture Notes in Computer Sci*

4. Alahmadi MD. Boundary aware u-net for medical image segmentation. *Arabian J Sci Eng* (2023) 48:9929–40. doi:10.1007/s13369-022-07431-y

5. Bernal J, Sánchez FJ, Fernández-Esparrach G, Gil D, Rodríguez C, Vilariño F. Wm-dova maps for accurate polyp highlighting in colonoscopy: validation vs. saliency maps from physicians. *Comput Med Imaging Graphics* (2015) 43:99–111. doi:10.1016/j.compmedimag.2015.02.007

6. Huttenlocher DP, Klanderman GA, Rucklidge WJ. Comparing images using the hausdorff distance. *IEEE Trans Pattern Anal Machine Intelligence* (1993) 15:850–63. doi:10.1109/34.232073

7. Gal Y, Ghahramani Z. Dropout as a bayesian approximation: representing model uncertainty in deep learning. In: MF Balcan, KQ Weinberger, editors. *Proceedings of the 33rd international conference on machine learning*, New York, NY, USA: PMLR (2016). p. 48. 1050–9.*Proc Machine Learn Res*

8. Guo C, Pleiss G, Sun Y, Weinberger KQ. On calibration of modern neural networks. In: D Precup, YW Teh, editors. *Proceedings of the 34th international conference on machine learning*, Sydney, Australia: PMLR (2017). p. 1321–30.*Proc Machine Learn Res*

9. Gao SH, Cheng MM, Zhao K, Zhang XY, Yang MH, Torr P. Res2net: a new multi-scale backbone architecture. *IEEE Trans Pattern Anal Machine Intelligence* (2021) 43:652–62. doi:10.1109/TPAMI.2019.2938758

10. Chen J, Mei J, Li X, Lu Y, Yu Q, Wei Q, et al. Transunet: rethinking the u-net architecture design for medical image segmentation through the lens of transformers. *Med Image Anal* (2024) 84:103280. doi:10.1016/j.media.2024.103280

11. Hatamizadeh A, Tang Y, Nath V, Yang D, Myronenko A, Landman B, et al. UNETR: transformers for 3D medical image segmentation. In: *Proceedings of the IEEE/CVF winter Conference on Applications of computer vision (WACV) (waikoloa, HI, USA: ieee)* (2022). p. 574–84. doi:10.1109/WACV51458.2022.00181

12. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. *Adv in Neural Inf Process Syst* (2017). p. 5998–6008.

13. Hu EJ, Shen Y, Wallis P, Allen-Zhu Z, Li Y, Wang S, et al. LoRA: low-rank adaptation of large language models. In: *Proceedings of the international conference on learning representations (ICLR)*. Virtual Event: OpenReview.netAvailable online at: https://openreview.net/forum?id=nZeVKeeFYf9 (2022).

14. Kirillov A, Mintun E, Ravi N, Mao H, Rolland C, Gustafson L, et al. Segment anything. In: *Proceedings of the IEEE/CVF international Conference on computer vision (ICCV) (paris, France: ieee)* (2023). p. 4015–26. doi:10.1109/ICCV51070.2023.00371

15. Ma J, He Y, Li F, Han L, You C, Wang B. Segment anything in medical images. *Nat Commun* (2024) 15:654. doi:10.1038/s41467-024-44824-z

16. McGinnis J, Shit S, Li HB, Sideri-Lampretsa V, Graf R, Dannecker M, et al. Single-subject multi-contrast MRI super-resolution via implicit neural representations, *Med Image Comput Computer Assisted Intervention – MICCAI 2023*. (2023). 14230. 173–83. doi:10.1007/978-3-031-43993-3_17

17. Wu J, Wang Z, Hong M, Ji W, Fu H, Liu Y, et al. Segment anything model for medical image analysis: an experimental study. *Med Image Anal* (2023) 89:102918. doi:10.1016/j.media.2023.102918

18. Wei X, Zhang R, Wu J, Liu J, Lu M, Guo Y, et al. NTO3D: neural target object 3d reconstruction with segment anything. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*. Seattle, WA, USA: IEEE (2024). p. 20352–62. doi:10.1109/CVPR52733.2024.01924

19. Zhang K, Liu D. Customized segment anything model for medical image segmentation. *arXiv preprint arXiv:2304.13785* (2023).

20. Cheng J, Ye J, Deng Z, Chen J, Li T, Wang H, et al. SAM-Med2D. *arXiv preprint arXiv:2308.16184* (2023). doi:10.48550/arXiv.2308.16184

21. Bui NT, Hoang DH, Tran MT, Doretto G, Adjeroh D, Patel B, et al. SAM3D: segment anything model in volumetric medical images. *arXiv preprint arXiv:2309.03493* (2023). doi:10.48550/arXiv.2309.03493

22. Zhang Y, Sapkota N, Gu P, Peng Y, Zheng H, Chen DZ. Keep your friends close and enemies farther: debiasing contrastive learning with spatial priors in 3d radiology images. In: Proceedings of the 2022 IEEE international Conference on Bioinformatics and biomedicine (BIBM) Las Vegas, NV, USA: (IEEE) (2022). p. 1824–9. doi:10.1109/BIBM55620.2022.9995481

23. Shi H, Han S, Huang S, Liao Y, Li G, Kong X, et al. Mask-enhanced segment anything model for tumor lesion semantic segmentation. In: *Proceedings of the international conference on medical image computing and computer-assisted intervention (MICCAI 2024)* (2024). p. 403–13. doi:10.1007/978-3-031-72111-3_38

24. Wang X, Mou J, Cao Y, Jahanshahi H. Modeling and analysis of cellular neural networks based on memcapacitor. *Int J Bifurcation Chaos* (2025) 35. doi:10.1142/S0218127425300101

25. Cao H, Cao Y, Lei Q, Mou J. Dynamical analysis, multi-cavity control and dsp implementation of a novel memristive autapse neuron model emulating brain behaviors. *Chaos, Solitons and Fractals* (2025) 191:115857. doi:10.1016/j.chaos.2024.115857

26. Ma Y, Mou J, Jahanshahi H, Alkhateeb AF, Bi X. Design and dsp implementation of a hyperchaotic map with infinite coexisting attractors and intermittent chaos based on a novel locally active memcapacitor. *Chaos, Solitons and Fractals* (2023) 173:113708. doi:10.1016/j.chaos.2023.113708

27. Chen Y, Cao Y, Mou J, Sun B, Banerjee S. A simple photosensitive circuit based on a mutator for emulating memristor, memcapacitor, and meminductor: light illumination effects on dynamical behaviors. *Int J Bifurcation Chaos* (2024) 34:2450069. doi:10.1142/S021812742450069X

28. Yuan Y, Yu F, Tan B, Huang Y, Yao W, Cai S, et al. A class of n-d Hamiltonian conservative chaotic systems with three-terminal memristor: modeling, dynamical analysis, and fpga implementation. *Chaos* (2025) 35:013121. doi:10.1063/5.0238893

29. Yu F, Su D, He S, Wu Y, Zhang S, Yin H. Resonant tunneling diode cellular neural network with memristor coupling and its application in police forensic digital image protection. *Chin Phys B* (2025) 34:050502. doi:10.1088/1674-1056/adb8bb

30. Yu F, He S, Yao W, Cai S, Xu Q. Quantitative characterization system for macroecosystem attributes and states. *IEEE Trans Computer-Aided Des Integrated Circuits Syst* (2025) 36:1–12. doi:10.13287/j.1001-9332.202501.031

31. Han Z, Cao Y, Banerjee S, Mou J. Hybrid image encryption scheme based on hyperchaotic map with spherical attractors. *Chin Phys B* (2025) 34:030503. doi:10.1088/1674-1056/ada7db

32. Liu Z, Li P, Cao Y, Mou J. A novel multimodal joint information encryption scheme based on multi-level confusion and hyperchaotic map. *Int J Mod Phys C* (2025). doi:10.1142/S012918312550038X

33. Zhou S, Yin Y, Erkan U, Toktaş A, Zhang Y. Novel hyperchaotic system: implementation to audio encryption. *Chaos, Solitons and Fractals* (2025) 193:116088. doi:10.1016/j.chaos.2025.116088

34. Yu F, Zhang S, Su D, Wu Y, Gracia YM, Yin H. Dynamic analysis and implementation of fpga for a new 4d fractional-order memristive hopfield neural network. *Fractal and Fractional* (2025) 9:115. doi:10.3390/fractalfract9020115

35. Mou J, Zhang Z, Zhou N, Zhang Y, Cao Y. Mosaic tracking: lightweight batch video frame awareness multi-target encryption scheme based on a novel discrete tabu learning neuron and yolov5. *IEEE Internet Things J* (2024) 12:4038–49. doi:10.1109/JIOT.2024.3482289

36. Mou J, Tan L, Cao Y, Zhou N, Zhou Y. Multi-face image compression encryption scheme combining extraction with stp-cs for face database. *IEEE Internet Things J* (2025) 12:19522–31. doi:10.1109/JIOT.2025.3541228

37. Mou J, Zhang Z, Banerjee S, Zhang Y. Combining semi-tensor product compressed sensing and session keys for low-cost encryption of batch information in wbans. *IEEE Internet Things J* (2024) 11:33565–76. doi:10.1109/jiot.2024.3429349

38. Zhou S, Liu H, Iu HHC, Erkan U, Toktas A. Novel n-dimensional nondegenerate discrete hyperchaotic map with any desired lyapunov exponents. *IEEE Internet Things J* (2025) 12:9082–90. doi:10.1109/JIOT.2025.3541229

39. Shi F, Cao Y, Xu X, Mou J. A novel memristor-coupled discrete neural network with multi-stability and multiple state transitions. *Eur Phys J Spec Top* (2025). doi:10.1140/epjs/s11734-024-01440-8

40. Zhang Z, Cao Y, Zhou N, Xu X, Mou J. Novel discrete initial-boosted tabu learning neuron: dynamical analysis, dsp implementation, and batch medical image encryption. *Appl Intelligence* (2025) 55:61. doi:10.1007/s10489-024-05918-9

41. Ma T, Mou J, Banerjee S, Cao Y. Analysis of the functional behavior of fractional-order discrete neuron under electromagnetic radiation. *Chaos, Solitons and Fractals* (2023) 176:114113. doi:10.1016/j.chaos.2023.114113

42. Mou J, Cao H, Zhou N, Cao Y. A fhn-hr neuron network coupled with a novel locally active memristor and its dsp implementation. *IEEE Trans Cybernetics* (2024) 54:7333–42. doi:10.1109/TCYB.2024.3471644

43. Zhang Z, Mou J, Zhou N, Banerjee S, Cao Y. Multi-cube encryption scheme for multi-type images based on modified klotski game and hyperchaotic map. *Nonlinear Dyn* (2024) 112:5727–47. doi:10.1007/s11071-024-09292-6

44. Reich C, Prangemeier T, Cetin Ö, Koeppl H. Oss-net: memory efficient high resolution semantic segmentation of 3d medical data. In: *Proceedings of the British machine vision conference (BMVC)* (2021). p. 429.

45. Khan MO, Fang Y. Implicit neural representations for medical imaging segmentation. Medical image computing and computer assisted intervention – MICCAI 2022 springer. *Lecture Notes in Computer Sci* (2022) 13431:433–43. doi:10.1007/978-3-031-16443-9_42

46. Zhang Y, Gu P, Sapkota N, Chen DZ. SwIPE: efficient and robust medical image segmentation with implicit patch embeddings. In: *Medical image computing and computer-assisted intervention – MICCAI 2023*. Springer Nature Switzerland (2023). p. 315–26. doi:10.1007/978-3-031-43904-9_31

47. Mildenhall B, Srinivasan PP, Tancik M, Barron JT, Ramamoorthi R, Ng R. Nerf: representing scenes as neural radiance fields for view synthesis. *Commun ACM* (2022) 65:99–106. doi:10.1145/3503250

48. Sørensen K, Camara O, Backer OD, Kofoed KF, Paulsen RR. NUDF: neural unsigned distance fields for high resolution 3d medical image segmentation. In: *Proceedings of the 19th IEEE international symposium on biomedical imaging (ISBI)* (2022). p. 1–5. doi:10.1109/ISBI52829.2022.9761610

49. Stolt-Ansó N, McGinnis J, Pan J, Hammernik K, Rueckert D. Nisf: neural implicit segmentation functions. In: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2023, 14231. Vancouver, BC, Canada: Springer (2023). p. 734–44. doi:10.1007/978-3-031-43901-8_70

50. Yang J, Wickramasinghe U, Ni B, Fua P. Implicitatlas: learning deformable shape templates in medical imaging. In: *Proceedings of the IEEE/CVF Conference on computer Vision and pattern recognition (CVPR) (new Orleans, LA, USA: IEEE)* (2022). p. 15861–71.

51. Molaei A, Aminimehr A, Tavakoli A, Kazerouni A, Azad B, Azad R, et al. Implicit neural representation in medical imaging: a comparative survey. In: *Proceedings of the IEEE/CVF international Conference on computer vision workshops (ICCVW) (paris, France: IEEE)* (2023). p. 2381–91. doi:10.1109/ICCVW60793.2023.00252

52. Amiranashvili T, Lüdke D, Li HB, Menze B, Zachow S. Learning shape reconstruction from sparse measurements with neural implicit functions. In: *Proceedings of the 5th international Conference on medical Imaging with deep learning (MIDL) (Zürich, Switzerland: PMLR)*, 172 (2022). p. 22–34.

53. Chibane J, Alldieck T, Pons-Moll G. Implicit functions in feature space for 3d shape reconstruction and completion. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*. Seattle, WA, USA: IEEE (2020). p. 6968–79. doi:10.1109/CVPR42600.2020.00698

54. Bui NT, Hoang DH, Tran MT, Doretto G, Adjeroh D, Patel B, et al. Sam3d: segment anything model in volumetric medical images. In: Proceedings of the 2024 IEEE International Symposium on Biomedical Imaging (ISBI). Athens, Greece: IEEE (2024). p. 1–4.

55. Wei X, Cao J, Jin Y, Lu M, Wang G, Zhang S. I-medsam: implicit medical image segmentation with segment anything. In: *Proceedings of the European Conference on computer vision (ECCV) (Milan, Italy: Springer*, 15068 (2024). p. 90–107. doi:10.1007/978-3-031-72684-2_6

56. Landman B, Xu Z, Iglesias J, Styner M, Langerak T, Klein A. Miccai multi-atlas labeling beyond the cranial vault—workshop and challenge. *Proc MICCAI Multi-Atlas Labeling Beyond Cranial Vault—Workshop Challenge (Munich, Germany)* (2015) 5:12.

57. Ji Y, Bai H, Yang J, Ge C, Zhu Y, Zhang R, et al. Amos: a large-scale abdominal multi-organ benchmark for versatile medical image segmentation. *Adv in Neural Inf Process Syst* (2022) 35:36722–32.

58. Loshchilov I, Hutter F. Decoupled weight decay regularization. In: Proceedings of the 7th international Conference on learning representations (ICLR) (New Orleans, LA, USA) (2019).

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 21 July 2025
DOI 10.3389/fphy.2025.1636357

OPEN ACCESS

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

REVIEWED BY
Liu Jie,
Northwestern Polytechnical University, China
Tianxiu Lu,
Sichuan University of Science and Engineering, China

*CORRESPONDENCE
Yide Ma,
✉ ydma@lzu.edu.cn

RECEIVED 27 May 2025
ACCEPTED 17 June 2025
PUBLISHED 21 July 2025

CITATION
Wang X, Ma P, Lian J, Liu J and Ma Y (2025) An echo state network based on enhanced intersecting cortical model for discrete chaotic system prediction.
Front. Phys. 13:1636357.
doi: 10.3389/fphy.2025.1636357

An echo state network based on enhanced intersecting cortical model for discrete chaotic system prediction

Xubin Wang[1], Pei Ma[1], Jing Lian[2], Jizhao Liu[1] and Yide Ma[1]*

[1]School of Information Science and Engineering, Lanzhou University, Lanzhou, Gansu, China, [2]School of Electronics and Information Engineering, Lanzhou Jiaotong University, Lanzhou, Gansu, China

Introduction: The prediction of chaotic time series is a persistent problem in various scientific domains due to system characteristics such as sensitivity to initial conditions and nonlinear dynamics. Deep learning models, while effective, are associated with high computational costs and large data requirements. As an alternative, Echo State Networks (ESNs) are more computationally efficient, but their predictive accuracy can be constrained by the use of simplistic neuron models and a dependency on hyperparameter tuning.

Methods: This paper proposes a framework, the Echo State Network based on an Enhanced Intersecting Cortical Model (ESN-EICM). The model incorporates a neuron model with internal dynamics, including adaptive thresholds and inter-neuron feedback, into the reservoir structure. A Bayesian Optimization algorithm was employed for the selection of hyperparameters. The performance of the ESN-EICM was compared to that of a standard ESN and a Long Short-Term Memory (LSTM) network. The evaluation used data from three discrete chaotic systems (Logistic, Sine, and Ricker) for both one-step and multi-step prediction tasks.

Results: The experimental results indicate that the ESN-EICM produced lower error metrics (MSE, RMSE, MAE) compared to the standard ESN and LSTM models across the tested systems, with the performance difference being more pronounced in multi-step forecasting scenarios. Qualitative analyses, including trajectory plots and phase-space reconstructions, further support these quantitative findings, showing that the ESN-EICM's predictions closely tracked the true system dynamics. In terms of computational cost, the training phase of the ESN-EICM was faster than that of the LSTM. For multi-step predictions, the total experiment time, which includes the hyperparameter optimization phase, was also observed to be lower for the ESN-EICM compared to the standard ESN. This efficiency gain during optimization is attributed to the model's intrinsic stability, which reduces the number of divergent trials encountered by the search algorithm.

Discussion: The results indicate that the ESN-EICM framework is a viable method for the prediction of the tested chaotic time series. The study shows that enhancing the internal dynamics of individual reservoir neurons can be an effective strategy for improving prediction accuracy. This approach of modifying neuron-level complexity, rather than network-level architecture, presents a potential direction for the design of future reservoir computing models for complex systems.

KEYWORDS

ESN-EICM, time-series prediction, reservoir computing, complex system, brain-inspired computing

1 Introduction

Time series prediction is a critical task across diverse scientific and engineering domains, including economics, meteorology, and industrial process control [1]. Among various types of time series, chaotic systems pose a unique and formidable challenge due to their deterministic yet highly unpredictable nature, extreme sensitivity to initial conditions (the butterfly effect), and complex, aperiodic dynamics [2]. Accurately modeling and predicting such systems is crucial for understanding their underlying mechanisms and for making informed decisions in applications.

In recent years, deep learning (DL) methodologies have played a important role in time series prediction. Recurrent Neural Networks (RNNs) and their variants, such as Long Short-Term Memory (LSTM) [2] and Gated Recurrent Units (GRU) [3], are designed to capture temporal dependencies. More recently, Transformer-based architectures [4] have demonstrated success in sequence modeling tasks. While these DL models can learn complex nonlinear relationships from data, they often entail significant drawbacks. These include high computational expense for training, the need for large datasets to avoid overfitting, and a "black-box" nature that hinders interpretability and deployment in critical domains requiring decision transparency [5]. Specialized architectures like WaveNet [6] and DeepAR [5] also face challenges such as resource consumption or limitations with sparse data.

Reservoir Computing (RC) has emerged as an alternative paradigm that offers a compelling balance between performance and computational efficiency [7]. Echo State Networks (ESNs), a principal RC model, utilize a fixed, randomly generated recurrent neural network (the "reservoir") to project input signals into a high-dimensional state space, with only a linear output layer being trained. This drastically reduces training complexity compared to DL models. However, standard ESNs are not without limitations. Their performance is highly sensitive to the initialization of reservoir hyperparameters, which typically requires extensive manual tuning or grid search [7]. Moreover, traditional ESNs often employ simplistic neuron activation functions, which may not adequately capture the rich dynamics inherent in complex chaotic systems. While advancements like Leaky ESNs, Deep Reservoir Computing [8], and multi-reservoir ESNs have been proposed, they can introduce further complexities or still rely on fundamentally simple neuronal dynamics.

The limitations of existing DL and RC approaches motivate the development of novel prediction models that can combine the training efficiency of RC with more sophisticated, adaptive internal dynamics and a systematic approach to hyperparameter optimization. Indeed, current research trends emphasize that network models with internal complexity can bridge artificial intelligence and neuroscience, offering pathways to more robust and capable systems [9]. Drawing inspiration from neuroscience, the Intersecting Cortical Model (ICM) [10] simulates neuronal behaviors like adaptive thresholds and feedback, but its original formulation is primarily suited for image processing and has limitations for continuous time series tasks.

This work proposes a novel framework, the Echo State Network Based on Enhanced Intersecting Cortical Model (ESN-EICM), for discrete chaotic time series prediction, the overall structure of which is illustrated in Figure 1. The ESN-EICM integrates a modified EICM neuron model into the ESN reservoir. The main contributions of this study are as follows:

1. The neural network model EICM can exhibit complex dynamic characteristics, is incorporated into reservoir computing. This novel neuron model is tailored for time series by incorporating features such as continuous sigmoid activation, global mean-driven adaptive thresholds, and introduces mechanisms for inter-neuron coupling and dynamic threshold regulation within each neuron, thereby enhancing the nonlinear representation capability of reservoir computing and forming a reservoir computing model based on biological neurons. The design leverages principles of how biological neural systems integrate information and utilize internal neuronal dynamics for complex computations, such as feature binding through dendritic networks [11] or learning multi-timescale dynamics [12]. While traditional RC research often focuses on optimizing reservoir topology or simplifying dynamics, our work explores a complementary direction: enhancing the computational power of individual neurons within the reservoir. We hypothesize that by equipping neurons with more sophisticated, adaptive dynamics inspired by the cortex, the reservoir can more effectively capture the intricate, non-linear patterns of chaotic systems without requiring complex topological design.
2. The application of a Bayesian Optimization strategy for the automated and efficient tuning of ESN-EICM hyperparameters, mitigating the traditional RC challenge of manual parameter selection.
3. A comprehensive empirical evaluation on three discrete chaotic systems (Logistic, Sine, and Ricker), demonstrating the ESN-EICM's superior predictive accuracy and stability in both one-step and multi-step prediction scenarios compared to standard ESN and LSTM models.

The remainder of this paper is organized as follows. Section 2 reviews related work in deep learning and reservoir computing for time series prediction. Section 3 details the proposed ESN-EICM model, including the EICM neuron design and the Bayesian optimization strategy. Section 4 describes the experimental setup and presents a thorough analysis of the results, covering prediction performance, hyperparameter sensitivity, and training time comparisons. Section 5 then discusses the broader implications

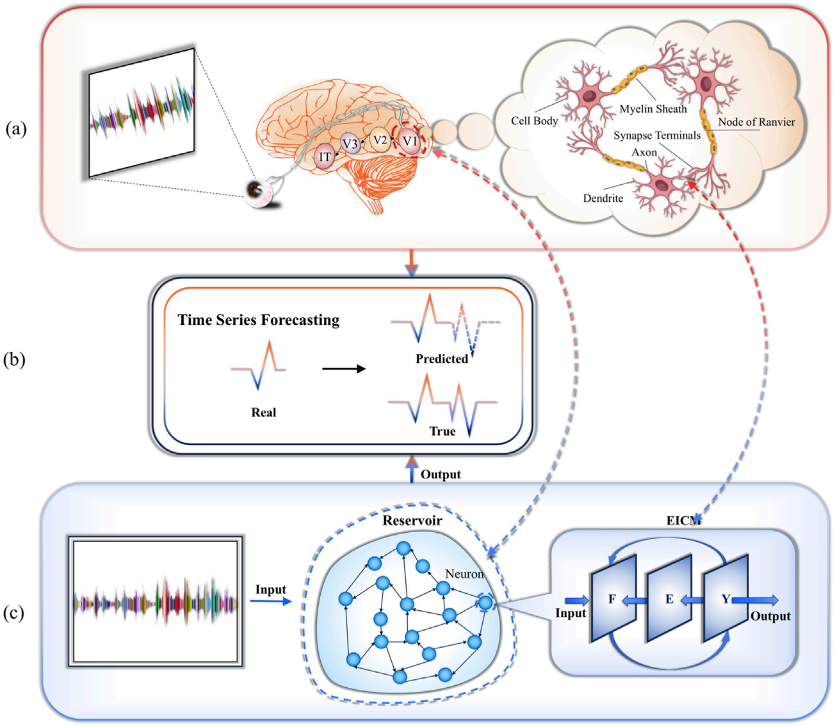


FIGURE 1
Proposed framework is based on the EICM neuron model of the mammalian visual cortex and uses it to construct a reservoir for performing time-series prediction tasks in chaotic systems.The EICM neuron model used in this framework is inspired by the dynamic characteristics of neurons in the mammalian primary visual cortex. Its core mechanism aims to more accurately simulate the real behavior of these biological neurons. By integrating the EICM neuron model (inspired by the V1 area of the primary visual cortex) into the reservoir, the constituent neurons are randomly interconnected via the weight matrix. **(a)**Visual pathway of the brain: Visual information from the retina is relayed via the lateral geniculate nucleus to the primary visual cortex (V1) and then processed in V2, V3, and V4, ultimately yielding patterns in the inferior temporal cortex. **(b)**Real vs. predicted time series. **(c)**ESN-EICM framework incorporates a V1 neuron model into reservoir computing to effectively forecast complex time-series sequences.

of our findings, including the model's design philosophy and its robustness against chaotic dynamics. Subsequently, Section 6 outlines the limitations of the current study and potential avenues for future research. Finally, Section 7 concludes the paper, summarizing the main contributions.

2 Related works

2.1 Deep learning-based time series prediction methods

With the continuous advancement of deep learning techniques, time series prediction has increasingly shifted toward neural network-based modeling strategies. The Recurrent Neural Network (RNN), proposed by Rumelhart et al. in 1986 [1], models sequential data through recurrent connections, effectively encoding historical information into hidden states. However, RNNs face challenges such as gradient vanishing and exploding gradients when handling long sequences, limiting their ability to capture long-term dependencies [2].

To address these limitations, Hochreiter and Schmidhuber introduced the Long Short-Term Memory (LSTM) network in 1997 [2]. LSTMs are specifically designed to retain long-range temporal information via sophisticated gating mechanisms (input, forget, and output gates), which control the flow of information through the cell. This architecture significantly improved the modeling of nonlinear data and has seen widespread application in diverse fields such as financial market prediction and climate modeling. Owing to their capacity to learn complex temporal dependencies and approximate highly nonlinear functions, LSTMs have also become a prominent benchmark for prediction chaotic time series, where accurately capturing long-range, intricate patterns is essential [13]. Indeed, studies have demonstrated LSTMs' potential in predicting various chaotic systems, leveraging their ability to learn from historical data without explicit knowledge of the system's underlying equations [14]. Nevertheless, despite their utility as a powerful baseline, the application of LSTMs, particularly to sensitive chaotic dynamics, is not without its difficulties. LSTM training demands substantial computational resources and a considerable amount of data to prevent overfitting, which can be a significant constraint in scenarios where data is scarce or computationally expensive to generate. They also exhibit high overall computational complexity [3], and their performance can be sensitive to hyperparameter choices, often requiring extensive tuning.

In contrast, the Gated Recurrent Unit (GRU) [3] simplifies the LSTM's gating mechanism (employing update and reset gates) to reduce model complexity and the number of parameters. GRUs often demonstrate comparable or, in some cases, superior performance to LSTMs, especially in scenarios with limited data volume. However, they may still exhibit higher prediction errors when processing large-scale, highly complex datasets compared to more specialized architectures [4].

More recently, the Transformer architecture [4], originally developed for natural language processing, has transcended traditional recurrent networks through its self-attention mechanism. This allows for powerful parallel computation and has led to outstanding performance in large-scale sequence modeling tasks. However, the standard Transformer's quadratic complexity with respect to sequence length and its potential sensitivity to noise in high-frequency or irregular time series can compromise its effectiveness for certain types of chaotic data without specific adaptations [15].

Despite ongoing methodological developments, deep learning models exhibit inherent limitations that are particularly pertinent to chaotic time series prediction: 1) Complex architectures generally lead to increased computational costs for training and inference. 2) Their "black-box" nature often weakens interpretability, hindering their deployment in domains requiring decision transparency or a deeper understanding of the model's predictive reasoning (e.g., finance, healthcare, scientific discovery) [5]. For instance, while WaveNet [6] can model long sequences through dilated convolutions, it consumes excessive resources and is not easily parallelized. DeepAR [5], a probabilistic prediction model, may struggle with very sparse data scenarios sometimes encountered in chaotic systems. Furthermore, hybrid models like LSTM-FCN (LSTM Fully Convolutional Network) [16], while effective for classification, can face efficiency bottlenecks in feature fusion for regression tasks. Additionally, modifications aimed at reducing complexity in Transformers, such as the ProbSparse attention mechanism in the Informer model [15], can often discard critical subtle temporal patterns vital for chaotic systems, potentially degrading prediction stability. Beyond these established deep learning architectures, Spiking Neural Networks (SNNs), which more closely mimic biological neuronal dynamics through event-driven spike-based communication, are also being actively investigated for their potential in efficient temporal processing and learning, with research exploring aspects such as advanced training methodologies like adaptive smoothing gradient learning [17], effective parameter initialization techniques [18], and the role of noise [19]. SNNs are also finding applications in complex learning paradigms like brain-inspired reinforcement learning [20], and are also being developed for energy-efficient applications such as speech enhancement [21].

2.2 Reservoir computing for time series prediction

Diverging from deep learning approaches, Reservoir Computing (RC) offers novel insights through nonlinear dynamical systems. Echo State Networks (ESNs), introduced by Lukoševičius and Jaeger [7], map inputs into high-dimensional dynamic spaces via randomly connected reservoirs. Their efficiency stems from training only the output layer, yet performance critically depends on reservoir initialization and hyperparameter selection [7]. To overcome these limitations, researchers proposed innovations: Leaky ESN balances short-term dynamics and long-term memory through leakage parameters; Adaptive Elastic ESN optimizes reservoir weights using sparse Bayesian learning, dynamically adjusting sparsity to enhance multi-scale feature adaptation, though suffering from high training complexity and hyperparameter sensitivity [22]. Multi-reservoir ESN improves complex dynamic capture by parallelizing multiple reservoirs processing distinct frequency bands, but increases training complexity without unified state-fusion protocols. Deep Reservoir Computing [8] extracts hierarchical features via cascaded reservoirs, achieving excellence in long-period modeling while risking state inflation and overfitting. Recently, the SEP framework advanced lossless byte-stream prediction through semantic-enhanced compression [23], opening new directions for complex temporal modeling.

Current RC methods predominantly rely on simplistic neuron models, failing to simulate mammalian brain structures. This restricts generalization capabilities and robustness–simple reservoirs perform poorly on complex systems, while intricate designs induce overfitting and instability. Furthermore, although RC reduces RNN training costs, its fixed critical parameters necessitate manual tuning, lacking dynamic adaptability [7]. These constraints motivate the integration of biologically inspired neuron models (e.g., ICM) with reservoir computing, aiming to enhance chaos sequence prediction robustness through dynamic weight initialization strategies. The broader field of neuromorphic computing also explores various mechanisms for temporal processing in SNNs, including specialized modules designed to capture temporal shifts [24], build sequential memory [25], or adapt temporal characteristics [26].

3 Methods

3.1 Problem statement and challenges

The existing methods face the following challenges:

(1) The performance of reservoir computing models highly depends on critical hyperparameters such as reservoir size, spectral radius, and input scaling. These parameters not only influence dynamic characteristics (e.g., memory capacity and nonlinear mapping ability) but also directly determine prediction accuracy. In practical applications, extensive experiments and manual tuning are required to identify optimal parameter combinations, leading to significant time costs. Prediction errors vary widely under different configurations, particularly for high-dimensional and long-term sequences, where parameter sensitivity becomes more pronounced. Some parameter combinations even cause training divergence [27], [28].

(2) Most reservoir models rely on basic neuron designs that fail to simulate the complex connectivity and information-processing mechanisms of mammalian cortical neurons. While this simplification reduces implementation complexity, it limits expressive power for tasks involving long-term dependencies or abrupt feature detection. Traditional ESN models maintain reasonable accuracy in short-term predictions [29] but suffer rapid performance degradation with increasing sequence length and dynamic complexity. Model states decay over time, and sensitivity to abrupt changes diminishes [22].

(3) Although expanding reservoir size or introducing multi-layer structures can enhance model expressiveness and achieve low training errors, these modifications introduce new challenges. Increased complexity improves training fit but severely harms generalization on unseen data. Prediction errors fluctuate significantly during testing, indicating overfitting and poor robustness under noise or input distribution shifts. This highlights that boosting model complexity alone cannot resolve generalization issues in time series prediction [8].

3.2 Echo state network based on enhanced intersecting cortical model framework

3.2.1 Input layer

The input layer transforms raw time series data into feature representations suitable for subsequent processing. Given a time series input $u_t \in \mathbb{R}^D$, where D denotes the input dimensionality, the input layer performs the following operation (Equation 1):

$$S_t = W_{\text{in}} \cdot [1; u_t] \tag{1}$$

where $[1; u_t]$ Adds a bias term to the input vector, and it allows the linear regression to learn an offset in the predictions in the output layers; $W_{in} \in \mathbb{R}^{N \times (D+1)}$ is input weight matrix, randomly initialized from a normal distribution, scaled by *input_scale*, and subsequently its elements are clipped to the range $[-2, 2]$; (N) is Reservoir size.

This approach ensures preliminary nonlinear mapping of input data while introducing an adjustable scaling factor *input_scale* to enhance adaptability to sequences with varying magnitudes [27].

3.2.2 Reservoir layer

The reservoir layer, the core component of ESN-EICM, comprises neurons governed by the Enhanced Intersecting Cortical Model (EICM). This design simulates biological feedback mechanisms and adaptive responses observed in mammalian cortical neurons.

The internal reservoir connectivity matrix $W \in \mathbb{R}^{N \times N}$ is constructed through the following steps:

- Elements of W are drawn from a standard normal distribution and then multiplied by the scaling factor *w_scale*.
- Sparsity is applied: a binary mask is generated where each element has a probability *w_sparsity* of being 1 (retaining the connection). The matrix W is multiplied element-wise by this mask, effectively setting a fraction of connections to zero. Thus *w_sparsity* represents the desired connection density.
- The elements of the resulting sparse matrix are then clipped to the range [-1,1].
- Finally, the spectral radius of this processed matrix is normalized to the target *spectral_radius* value to help ensure Echo State Property and dynamic stability, as shown in Equation 2:

$$W = W \cdot \left(\frac{\text{spectral\_radius}}{\max(|\lambda_i|)} \right) \tag{2}$$

The EICM neurons maintain internal states: F (feeding input), E (dynamic Threshold), and Y (Output term). Before the simulation begins (at time t = 0), these states are initialized. Specifically, F and E are initialized with random values drawn from a uniform distribution over [0, 0.1] and then their elements are clipped to the range $[-1, 1]$. The initial output states Y are set to zeros. Each neuron updates its state using the following equations:

Each neuron updates its state using the EICM dynamics, which are detailed in Section 3.3.

- The feeding input F_t is updated based on its prior value $F_{t=1}$, weighted feedback from other neurons in the reservoir $(W \cdot Y_{t=1})$, and the external stimulus S_t. This F_t is then clipped.
- The neuron's output Y_t is generated using a Sigmoid activation function. The input to the sigmoid is the clipped difference between the current F_t and the previous threshold $E_{t=1}$. Gaussian noise is added to the sigmoid's output, and the final Y_t is clipped to [0,1].
- The dynamic threshold E_t adapts based on its previous value $E_{t=1}$ and the mean of the intermediate neuron activations before noise. This E_t is also clipped.

For numerical stability, the primary state variables F_t and E_t, as well as the difference term $F_t = E_{t=1}$, are clipped to the range [−50, 50] during their update.

The EICM neuron model introduces critical modifications to the original ICM framework [10]. Feeding input F incorporates $W \cdot Y_{t-1}$ to enable cross-neuron interactions. This replaces the original ICM's local dynamics f, g, h with parameterized decay rates f, g, h. Threshold E is updated using the population mean of Y, diverging from the original ICM's local update rule. This prevents over-activation of individual neurons. Gaussian noise with standard deviation 0.001 is added to Y for regularization and exploration enhancement. The exploitation of noise as a computational resource is also a recognized concept in other neuromorphic models such as SNNs [19]. After an initial period (of length initLen steps, where neuron states stabilize), the augmented state vectors are collected for training the output layer. Each augmented state vector is formed as $\Phi(Y_t, u_t) = [1; u_t; Y_t]$, where u_t is the external input vector at time t, Y_t is the reservoir's neuron output vector, and 1 represents a bias term. These augmented vectors form the columns of a matrix $X_{collected} = [\Phi(Y_{initLen}, u_{initLen}), \ldots, \Phi(Y_T, u_T)]$. This $X_{collected}$ (referred to simply as X in the context of the output weight computation equation) is then used for training the output weights W_{out}.

3.2.3 Output layer

The output layer trains weights via regularized linear regression to produce predictions. Its equation is given by Equation 3:

$$\hat{y}_t = W_{\text{out}}^{\top} \cdot \Phi(Y_t, u_t) \tag{3}$$

where $\Phi(Y_t, u_t) = [1; u_t; Y_t]$ represents the concatenated feature vector; $W_{\text{out}} \in \mathbb{R}^{(D+N+1) \times K}$ is output weight matrix, solved using Tikhonov-regularized least squares (Equation 4):

$$W_{\text{out}} = (XX^{\top} + \lambda I)^{-1} XY \tag{4}$$

where the regularization coefficient $\lambda \in [1e-8, 1e-2]$.

TABLE 1 ESN-EICM parameters optimization space.

Parameter	Symbol	Range	Function
Reservoir Size	*res_size*	[300, 1500]	Balances model complexity and computational cost
Input Scale	*input_scale*	[0.2, 2.0]	Adjusts input mapping strength for scale adaptation
Sparsity	*w_sparsity*	[0.1, 0.3]	Reduces computation while preserving nonlinearity
Weight Scale	*w_scale*	[0.2, 2.0]	Controls the strength of internal weight connections
Spectral Radius	*spectral_radius*	[0.3, 0.99]	Ensures dynamic stability via eigenvalue normalization
Feedback Decay Rate	f	[0.1, 0.99]	Regulates historical input decay with clipping
Threshold Decay Rate	g	[0.1, 0.99]	Prevents threshold oscillation
Excitation Gain	h	[0.5, 2.0]	Amplifies global activation impact for robustness
Nonlinearity Control	β	[1.0, 10.0]	Adjusts sensitivity to input differences
Regularization Coefficient	λ	10^{-8}, 10^{-2}	Stabilizes weight inversion and improves generalization

During inference, future states are recursively generated using historical inputs and reservoir states (Equation 5):

$$\hat{y}_{t+1:T} = f\left(W_{\text{out}}, \{u_t, Y_t\}\right) \tag{5}$$

3.2.4 Bayesian optimization strategy

To address the time-consuming manual hyperparameter tuning and susceptibility to local optima in traditional reservoir computing models, we introduce Bayesian Optimization (BO) within the ESN-EICM framework. BO constructs a surrogate function (e.g., Gaussian Process) and an acquisition function to efficiently balance exploration (sampling unexplored regions of the hyperparameter space) and exploitation (focusing on promising regions identified by prior evaluations). This approach rapidly converges to globally optimal configurations by leveraging information from prior experiments [30], a significant advancement over simpler strategies like grid or random search [31]. In our experiments, we employ the gp_minimize function (based on Gaussian Process Regression) for iterative parameter search. The optimization objective is defined as minimizing the mean squared error (MSE) on the validation set. To guide the optimization, the training data (the first 16,000 steps) was further partitioned: the first 14,000 steps were used to train the ESN-EICM's output weights for a given hyperparameter set, and the subsequent 2,000 steps served as the validation set for calculating the MSE. The final reported test performance is evaluated on the held-out test set, which was never seen during training or optimization. Specify ranges and types (continuous/integers) for all parameters. Generate candidate hyperparameter combinations at each iteration and evaluate their MSE performance. Terminate the search process when the optimization objective (minimizing MSE) shows no significant improvement over consecutive iterations. Apply the optimal hyperparameter combination to train and test the final model. This strategy significantly reduces manual tuning costs while enhancing generalization capabilities for chaotic system prediction. The optimization space for ESN-EICM parameters is detailed in Table 1.

3.3 Enhanced intersecting cortical model

The Enhanced Intersecting Cortical Model (EICM) neuron model (Figure 2 proposed in this work is built upon the original Intersecting Cortical Model (ICM) framework. The ICM was first introduced by Ekblad et al. [10], and was originally designed for image processing tasks—particularly for extracting features with indistinct boundaries. It simulates the behavioral characteristics of neurons in the mammalian primary visual cortex, including feedback mechanisms and adaptive threshold regulation.

The original ICM neuron model consists of three key state variables: Feeding input F is a feedback term representing historical input memory at the current time step; Dynamic threshold E is an adaptive threshold that modulates neuron activation; Output term Y is a binary output or activation state.

Its update equations are defined as follows (Equation 6):

$$\begin{cases} F_t & = f \cdot F_{t-1} + S_t \\ Y_t & = \begin{cases} 1, & \text{if } F_t > E_t \\ 0, & \text{otherwise} \end{cases} \\ E_t & = g \cdot E_{t-1} + h \cdot Y_t \end{cases} \tag{6}$$

where feedback decay rate f controls the temporal decay of the feeding input; threshold decay rate g Prevents threshold oscillation; Excitation gain h regulates the strength of threshold updates based on neuron output; S is External stimulus input; output term Y equals 1 when the feeding input exceeds the dynamic threshold, and 0 otherwise.

This model has demonstrated strong edge detection and noise resistance capabilities in image segmentation applications. However, its binary output mechanism limits its expressiveness in time series

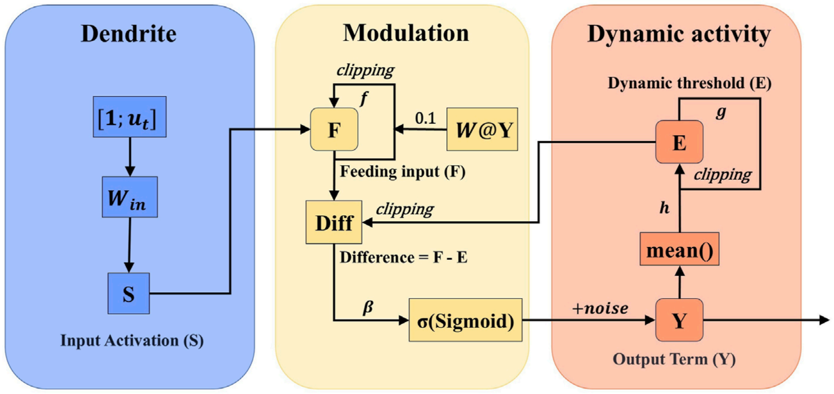


FIGURE 2
Enhanced Intersecting cortical model.

modeling. Despite the biologically inspired structure and nonlinear mapping advantages of ICM, several critical limitations arise when applying it to time series prediction tasks such as chaotic system prediction:

(1) The model exhibits a lack of neuron-to-neuron coupling. The feeding input solely takes into account individual history and external input, without incorporating interactions across the reservoir.
(2) The local adaptation mechanism is limited as the dynamic threshold updates based merely on the current output of neuron, failing to reflect global network activity. The static parameter settings without range constraints also pose an issue, causing hyperparameters to remain fixed or loosely defined, which in turn leads to instability during training.
(3) The binary output limitation of the original ICM restricts its applicability to continuous-value regression tasks, as it only employs a binary pulse output of 0 or 1.

These issues significantly impair the ability of ICM to capture long-term dependencies and abrupt changes in complex nonlinear systems, resulting in suboptimal performance in chaotic time series modeling.

To enhance the modeling capability of the original ICM for time series prediction, we propose the EICM neuron design. The key improvements are outlined as follows:

(1) Improved sensitivity to long-range dependencies and abrupt changes. The original ICM model struggles with capturing long-term dependencies and detecting sudden signal transitions due to its local update mechanism. In EICM, we introduce global coupling through reservoir connectivity $(W \cdot Y_{t-1})$ and a mean-driven threshold adaptation strategy [32], enabling neurons to respond more sensitively to abrupt changes in chaotic systems [27].
(2) Enhanced Expressiveness for Continuous-Value Prediction. Unlike the binary pulse output in standard ICM, output term Y EICM employs a continuous Sigmoid activation function. This modification allows the model to perform regression-based time series prediction tasks effectively, significantly expanding its applicability compared to the original image segmentation-oriented design.
(3) Integration of data-driven adaptation mechanisms. We redesign the dynamic threshold E update rule by using the global mean activation of all neurons. This approach improves generalization and prevents local over-activation or under-activation, ensuring better consistency across the network during long-term prediction. This principle of integrating global network context to modulate local neuronal behavior is an active area of research, with concepts like context gating being explored in SNNs to achieve robust and adaptive learning, such as in lifelong learning scenarios [33].
(4) During the implementation, we imposed numerical range constraints on the parameters to enhance training stability, performed numerical clipping on f, g, h, β, and added Gaussian noise perturbations with a standard deviation of 0.001 after each activation to prevent overfitting and strengthen exploration capabilities.

These enhancements address the limitations of ICM in temporal modeling while preserving its biologically inspired structure. The EICM neuron model operates on three key state variables: the feeding input F, the dynamic threshold E, and the output term Y. To ensure a consistent starting point, these states are initialized at t = 0 as follows: F and E are populated with random values drawn from a uniform distribution $U(0, 0.1)$, which are then clipped element-wise to the range [−1, 1]. This small, positive initialization range was chosen to ensure that neurons start in a responsive, non-saturated state, close to the linear region of the sigmoid activation function, which promotes stable initial dynamics as the reservoir settles. The update dynamics of the EICM neuron from time $t = 1$ to t proceed in the following equations (Equation 7):

$$\begin{cases} F_t &= f \cdot F_{t-1} + 0.1 \cdot (W \cdot Y_{t-1}) + S_t \\ Y_t &= \dfrac{1}{1 + e^{-\beta(F_t - E_t)}} \\ E_t &= g \cdot E_{t-1} + h \cdot mean(Y_t) \end{cases} \tag{7}$$

These coupled dynamics allow the EICM neuron to maintain and process historical information over varying time scales, which is crucial for predicting chaotic systems. Effectively modeling such temporal dependencies is a key challenge in neural computation, with various architectures exploring mechanisms like dedicated delay units or gates to manage temporal information flow [34]. The core parameters intuitively govern the neuron's behavior: f (Feedback Decay Rate) controls the neuron's short-term memory of its own past state; g (Threshold Decay Rate) stabilizes the adaptive threshold, preventing overly rapid fluctuations; h (Excitation Gain) determines how strongly the global network activity influences a neuron's excitability; and β (Nonlinearity Control) adjusts the steepness of the sigmoid activation, controlling the neuron's sensitivity to the difference between its feeding input and its threshold. A higher *beta* leads to a more switch-like, saturating behavior where the neuron's output quickly approaches 0 or 1, while a lower *beta* results in a smoother, more graded response across a wider range of inputs.

The enhanced performance of the ESN-EICM stems from the synergistic interaction between its two primary modifications: the global coupling feedback ($W \cdot Y_{t-1}$) and the global mean-driven adaptive threshold ($h \cdot \mathrm{mean}(Y_t)$). These mechanisms work in concert to regulate the reservoir's dynamics. The global coupling term ensures a rich and diverse set of inputs to each neuron, promoting complex, high-dimensional state representations and preventing the network from falling into simple, synchronized activity patterns. Concurrently, the adaptive threshold acts as a homeostatic, or self-regulating, mechanism. By adjusting each neuron's excitability based on the *average* activity of the entire reservoir, it prevents runaway activation or quiescence. This homeostatic regulation keeps the reservoir in a critical "edge of chaos" regime, where it is most sensitive to input perturbations and possesses maximal memory capacity, which is crucial for stabilizing long-term predictions and effectively modeling chaotic dynamics.

3.3.1 Feeding input *F*

In the design of the feeding input F, we retain the exponential decay mechanism from the original ICM model, while introducing a dynamic coupling mechanism through the reservoir connectivity matrix W to enable each neuron to perceive the overall state of the network. Where F is first computed and then clipped to produce F. The coefficient 0.1 serves as a normalization factor for the reservoir feedback term. S is the external driving stimulus. This enhancement significantly improves. The updated equation is defined as (Equation 8):

$$F_t = f \cdot F_{t-1} + 0.1 \cdot \left(W \cdot Y_{t-1}\right) + S_t \tag{8}$$

where f denotes the feedback decay rate, which controls the temporal decay of the historical feedback term and is constrained within a reasonable range to improve training stability; $W \cdot Y_{t-1}$ represents the influence from other neurons in the reservoir on the current neuron's feedback input; The coefficient 0.1 serves as a normalization factor to prevent gradient explosion; S is the external driving stimulus.

This enhancement significantly improves the suitability of the model for time series modeling by increasing inter-neuron information flow and cross-neuron coordination, thereby enhancing its nonlinear mapping capability compared to the original ICM framework.

3.3.2 Output term *Y*

To improve the expressiveness and robustness of the model, we replace the binary output mechanism in the original ICM with a multi-step process yielding a continuous output. First, the input to the sigmoid, $Delta_t$, is calculated and clipped. The sigmoid function produces an intermediate output. Gaussian noise $\mathcal{N}(0, 0.001)$ is then added, and finally, the output Y is clipped to the range [0,1] to maintain stability and a consistent output scale. The updated output equation is given as (Equation 9):

$$Y_t = \frac{1}{1 + e^{-\beta(F_t - E_t)}} + \mathcal{N}\left(0, 0.001\right) \tag{9}$$

where β controls the steepness of the activation function. The output values are no longer restricted to binary pulses (0 or 1), but instead fall within the continuous range [0,1] due to sigmoid activation and subsequent clipping. The standard deviation of 0.001 was chosen empirically as a value large enough to provide a regularizing effect and prevent overfitting, yet small enough not to disrupt the underlying learned dynamics of the system.

This improvement allows the model to more accurately capture subtle changes in input dynamics. The addition of small-scale noise injection further enhances exploration during training and improves generalization performance, particularly under noisy or uncertain conditions.

3.3.3 Dynamic threshold *E*

For the threshold update mechanism, we modify the original ICM approach which updates based on individual neuron output to a global mean-driven adaptation strategy. The updated equation is defined as (Equation 10):

$$E_t = g \cdot E_{t-1} + h \cdot \mathrm{mean}\left(Y_t\right) \tag{10}$$

where g is the threshold decay rate that governs the temporal decay of the dynamic threshold; h determines the gain factor of threshold adjustment; $\mathrm{mean}(Y_t)$ represents the average activation across all neurons at time t.

This global adaptive thresholding strategy enables each neuron to adjust its response threshold according to the overall network activity, preventing certain neurons from being overly activated or suppressed. As a result, the model achieves greater stability and consistency across the reservoir.

4 Experiment

4.1 Dataset generation

To evaluate the predictive capabilities of the proposed ESN-EICM, ESN, and LSTM models, we generate three representative discrete chaotic system: Logistic system, Sine system, and Ricker system. These datasets are chosen for their distinct dynamic characteristics:

- Logistic System: A discrete-time chaotic system with strong nonlinearity.
- Sine System: A smooth periodic system with limited chaotic behavior.

- Ricker System: A biological population model exhibiting complex oscillatory patterns.

4.1.1 Data generation Process

Each dataset is generated using the following equations (Equations 11–13):

$$\text{Logistic System}: x_{t+1} = 3.8 \cdot x_t \cdot (1 - x_t) \tag{11}$$

$$\text{Sine System}: x_{t+1} = 0.9 \cdot \sin(\pi x_t) \tag{12}$$

$$\text{Ricker System}: x_{t+1} = x_t \cdot \exp(4 \cdot (1 - x_t)) \cdot 0.5 \tag{13}$$

The initial value x_0 is set to 0.1 for Logistic/Sine Maps and 0.5 for Ricker Map. Each system is iterated for $T = 20000$ steps. We then construct 3D feature vectors to capture nonlinear dependencies:

- Logistic System: $[x_t, x_t^2, x_t^3]$
- Sine System: $[x_t, \sin(x_t), \cos(x_t)]$
- Ricker System: $[x_t, \log(x_t + 10^{-6}), \sqrt{x_t}]$

4.1.2 Data preprocessing

All datasets undergo the following preprocessing pipeline:

(1) Standardization: Data is standardized using Scikit-learn's StandardScaler (Equation 14):

$$x_{\text{scaled}} = \frac{x - \mu}{\sigma} \tag{14}$$

where μ and σ are computed on the training split.

(2) Input-Target alignment: The input-output relationship is defined as (Equation 15):

$$\text{inputs} = \mathbf{X}_{1:T-1}, \quad \text{targets} = \mathbf{X}_{2:T} \tag{15}$$

This ensures the model predicts x_{t+1} given x_t.

(3) Train/Test Split: All systems use the same split (Equation 16):

$$\text{trainLen} = 16000, \quad \text{testLen} = 2000 \tag{16}$$

The dataset was split chronologically to ensure strict temporal ordering and prevent information leakage from the test set into the training set. The first 16,000 steps were used for training and hyperparameter optimization, while the subsequent 2,000 steps were reserved exclusively for final testing. This partitioning is consistent across all systems to avoid introducing bias.

4.1.3 Dataset properties

A summary of the dataset configurations and properties is provided in Table 2.

TABLE 2 Dataset configurations and properties.

Dataset	Length	Features	Train/Test split
Logistic System	20,000	3	16,000/2000
Sine System	20,000	3	16,000/2000
Ricker System	20,000	3	16,000/2000

4.2 Evaluation metrics

In our experiments, we compute the following evaluation metrics to quantify prediction performance. Let y_i denote the ground truth value and $\hat{y}_i$ the predicted value at time i, where $n = 2000$ is the number of test samples.

4.2.1 Mean squared error (MSE)

The Mean Squared Error (MSE) measures the average squared difference between predicted and actual values. It's a common metric for regression problems, penalizing larger errors more heavily. The equation is as follows (Equation 17):

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{17}$$

4.2.2 Root mean squared error (RMSE)

The Root Mean Squared Error (RMSE) is simply the square root of the MSE. It reflects the standard deviation of prediction errors and is in the same units as the target variable, making it more interpretable than MSE. The equation is as follows (Equation 18):

$$\text{RMSE} = \sqrt{\text{MSE}} \tag{18}$$

4.2.3 Mean absolute error (MAE)

The Mean Absolute Error (MAE) measures the average absolute difference between predicted and actual values. Unlike MSE, MAE gives equal weight to all errors, making it more robust to outliers. The equation is as follows (Equation 19):

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{19}$$

4.2.4 Coefficient of determination (R^2)

The Coefficient of Determination (R^2) quantifies the proportion of variance in the dependent variable that can be predicted from the independent variables. A value closer to 1 indicates that the model explains a larger proportion of the variance in the ground truth values. The equation is as follows (Equation 20):

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{20}$$

4.2.5 Explained variance score (EVS)

The Explained Variance Score (EVS) evaluates how well the model captures the variance in the target variable. It's similar to R^2

but can be more informative in cases where the model has a bias. The equation is as follows (Equation 21):

$$\mathrm{EVS} = 1 - \frac{\mathrm{Var}(y_i - \hat{y}_i)}{\mathrm{Var}(y_i)} \tag{21}$$

4.2.6 Max error (ME)

The Max Error (ME) reports the maximum residual error between any predicted and actual value. This metric highlights the worst-case prediction scenario. The equation is as follows (Equation 22):

$$\mathrm{ME} = \max_{i=1,\ldots,n} |y_i - \hat{y}_i| \tag{22}$$

4.3 Model configuration

This section details the configuration of the models employed in our comparative study: the proposed ESN-EICM, the baseline ESN, and the LSTM network. For the ESN-EICM and ESN models, hyperparameters were primarily determined through Bayesian Optimization, aiming to minimize Mean Squared Error on a validation set. For the LSTM model, key architectural and training hyperparameters were also optimized using Bayesian Optimization, while others were set based on common practices in time series forecasting. The specific search ranges and fixed values for each model are presented in the subsequent subsections. All models were trained and evaluated on the datasets described in Section 2 to ensure fair comparison.

4.3.1 ESN-EICM model configuration

Table 3 presents the key configuration parameters for the ESN-EICM model determined through Bayesian optimization in our experiments. A washout period initLen of 1000 steps was used for all experiments. This length was determined through preliminary observations to be sufficiently long to allow the reservoir's internal state to become independent of its initial zero state and synchronize with the dynamics of the input signal across the range of tested hyperparameters.

TABLE 3 ESN-EICM model Configuration parameters.

Parameter	Description	Search range
res_size	Reservoir Size	[300, 1500]
input_scale	Input Scale	[0.2, 2.0]
w_sparsity	Sparsity	[0.1, 0.3]
w_scale	Weight Scale	[0.2, 2.0]
spectral_radius	Spectral Radius	[0.3, 0.99]
f	Feedback Decay Rate	[0.1, 0.99]
g	Threshold Decay Rate	[0.1, 0.99]
h	Excitation Gain	[0.5, 2.0]
β	Nonlinearity Control	[1.0, 10.0]
λ	Regularization Coefficient (λ)	10^{-8}, 10^{-2}
n_calls	Total Bayesian Optimization Iterations	50
initLen	Washout Period Length	1000
trainLen	Training Data Length	16,000
testLen	Test Data Length	2000

4.3.2 ESN model configuration

Table 4 presents the key configuration parameters for the Echo State Network (ESN) model determined through Bayesian optimization in our experiments.

TABLE 4 ESN model Configuration parameters.

Parameter	Description	Search range
res_size	Reservoir Size	[300, 700]
input_scale	Input Scale	[0.5, 1.0]
w_sparsity	Sparsity	[0.1, 0.3]
w_scale	Weight Scale	[0.5, 1.0]
λ	Regularization Coefficient (λ)	10^{-8}, 10^{-3}
n_calls	Total Bayesian Optimization Iterations	50
initLen	Washout Period Length	1000
trainLen	Training Data Length	16,000
testLen	Test Data Length	2000

4.3.3 LSTM model configuration

Table 5 outlines the key configuration parameters for the Long Short-Term Memory (LSTM) network model. The hyperparameters were optimized using Bayesian optimization, while other parameters were set based on common practices.

4.4 Hyperparameter optimization results

To ensure optimal performance, critical hyperparameters for both the ESN-EICM and the baseline ESN models were determined using Bayesian Optimization. This process, guided by minimizing Mean Squared Error on a validation set as described in Section 3, yielded task-specific parameter configurations. The best-found parameters for each model across the different chaotic systems and prediction horizons are presented below.

4.4.1 ESN-EICM best parameters

The optimal hyperparameters identified for the proposed ESN-EICM model through Bayesian Optimization are

TABLE 5 LSTM model Configuration parameters.

Parameter	Description	Search range/Value
hidden_size	Number of units in LSTM hidden layers	[128, 256]
num_layers	Number of LSTM layers	[1, 6]
lr	Learning rate for Adam optimizer	10^{-5}, 10^{-1}
batch_size	Number of samples per gradient update	[128, 256]
dropout	Dropout rate for LSTM layers	[0.1, 0.4]
sequence_length	Number of time steps in input sequences	[5, 20]
epochs	Number of training epochs per optimization trial/final model	70
clip_grad_norm	Gradient clipping threshold	1.0
n_calls	Total Bayesian Optimization function evaluations	20
n_initial_points	Initial random points for Bayesian Optimization	10
trainLen	Training data length (original time steps before sequencing)	16,000
testLen	Test data length (original time steps before sequencing)	2000
input_size	Number of features per time step (data-dependent)	3
output_size	Number of features to predict (data-dependent)	3

summarized in Table 6. These parameters cover aspects of reservoir architecture, input processing, EICM neuron dynamics, and output regularization.

4.4.2 ESN best parameters

For the baseline ESN model, the key hyperparameters tuned via Bayesian Optimization are detailed in Table 7. This allows for a direct comparison with the ESN-EICM model under similarly optimized conditions.

4.4.3 LSTM best parameters

The Long Short-Term Memory (LSTM) network, serving as another important baseline, also underwent hyperparameter optimization using Bayesian Optimization. Key architectural and training parameters were tuned to achieve its best performance on each specific task. The optimized values for parameters such as hidden size, number of layers, learning rate, batch size, dropout rate, and input sequence length are presented in Table 8. These results reflect the optimal configurations found for the LSTM model across the different chaotic systems and prediction steps.

4.5 Hyperparameter sensitivity analysis

AS shown in Figure 3 (Logistic System), Figure 4 (Sine System), and Figure 5 (Ricker System), this section presents the hyperparameter sensitivity analysis for the ESN-EICM model. The analysis investigates the Mean Squared Error (MSE) response to variations in individual hyperparameters, while other parameters are held at their globally optimized values (from Table 6, for one-step prediction). This provides insights into each parameter's influence on model performance and highlights the complexity of the hyperparameter landscape. The vertical dashed line in each plot marks the globally optimal value found by Bayesian Optimization.

For the Logistic system, several parameters show high sensitivity. The `res_size` has a local minimum around the globally optimized value of 1116. Both `input_scale` and `w_sparsity` display V-shaped curves, with their local minima being slightly lower than their respective globally optimized values (marked at 1.01 and 0.10). The `spectral_radius` is critical, with its lowest MSE point aligning perfectly with the global optimum of 0.69. The EICM neuron parameters also show distinct patterns: `f` has a local minimum near 0.35, while its global optimum is 0.44. Both `g` and `h` exhibit U-shaped curves. Notably, `beta` is extremely sensitive, with its MSE sharply decreasing to a minimum that coincides with its global optimum of 3.17.

In the Sine system, the parameter sensitivities differ. `res_size` shows that larger reservoirs in the tested range yield better performance, with the global optimum at 569. For `w_scale`, a clear trend of decreasing MSE with smaller values is observed. Both `spectral_radius` and the neuron parameter `f` are highly sensitive, with sharp V-shaped curves where the local minima are very close to their global optima (0.65 and 0.10, respectively). The parameters `g` and `h` have broader optimal regions. For `beta`, the MSE is lowest at

TABLE 6 Best ESN-EICM parameters for different chaotic systems and prediction steps.

Logistic system				
Parameter	One-step	Two-step	Three-step	Four-step
res_size	1116.0000	1443.0000	1444.0000	1409.0000
input_scale	1.0109	1.6158	1.9879	1.7625
w_sparsity	0.1027	0.1297	0.1533	0.2525
w_scale	1.8960	1.5453	1.9671	0.6222
spectral_radius	0.6887	0.7746	0.8073	0.5201
f	0.4430	0.4703	0.1002	0.1525
g	0.1142	0.6869	0.1031	0.7266
h	0.8463	1.4669	0.5000	0.5845
β	3.1692	2.0944	6.0793	4.2441
λ	0.0068	0.0033	0.0081	0.0015
Sine System				
Parameter	One-step	Two-step	Three-step	Four-step
res_size	569.0000	473.0000	1500.0000	1284.0000
input_scale	0.4049	1.8084	1.5526	1.2281
w_sparsity	0.1258	0.1143	0.2953	0.3000
w_scale	1.1507	1.1761	0.8319	0.9361
spectral_radius	0.6493	0.6754	0.3000	0.3223
f	0.1000	0.1908	0.3129	0.2806
g	0.6548	0.6750	0.9900	0.5462
h	2.0000	1.8889	0.5000	1.5949
β	8.0238	7.9411	4.2171	5.3005
λ	0.0006	0.0095	0.0026	0.0027
Ricker System				
Parameter	One-step	Two-step	Three-step	Four-step
res_size	1058.0000	473.0000	1483.0000	1500.0000
input_scale	2.0000	1.8084	1.7422	1.2621
w_sparsity	0.2619	0.1143	0.2525	0.1926
w_scale	0.2000	1.1761	0.7928	1.5747
spectral_radius	0.7054	0.6754	0.4555	0.3000
f	0.1000	0.1908	0.1229	0.1000
g	0.1000	0.6750	0.6453	0.4710

(Continued on the following page)

TABLE 6 (*Continued*) Best ESN-EICM parameters for different chaotic systems and prediction steps.

Ricker System				
Parameter	One-step	Two-step	Three-step	Four-step
h	0.9920	1.8889	0.6528	0.5000
β	2.1395	7.9411	4.4978	9.7544
λ	0.0000	0.0095	0.0021	0.0021

TABLE 7 Best ESN parameters for different chaotic systems and prediction steps.

Logistic system				
Parameter	One-step	Two-step	Three-step	Four-step
res_size	776.0000	1000.0000	500.0000	500.0000
input_scale	1.4807	2.0000	1.8349	1.4689
w_sparsity	0.1000	0.1000	0.1000	0.1000
w_scale	0.2000	0.2000	0.2362	0.2000
λ	0.0006	0.0000	0.0100	0.0100
Sine System				
Parameter	One-step	Two-step	Three-step	Four-step
res_size	512.0000	568.0000	500.0000	500.0000
input_scale	1.2216	1.6574	1.7777	1.8088
w_sparsity	0.1274	0.1000	0.1709	0.1000
w_scale	0.2000	0.2535	0.2000	0.2595
λ	0.0005	0.0003	0.0100	0.0000
Ricker System				
Parameter	One-step	Two-step	Three-step	Four-step
res_size	500.0000	500.0000	559.0000	500.0000
input_scale	1.6450	1.8691	2.0000	1.6485
w_sparsity	0.1000	0.1000	0.1000	0.1000
w_scale	0.2000	0.2000	0.2000	0.2000
λ	0.0033	0.0100	0.0100	0.0100

the higher end of the tested range, aligning with the global optimum of 8.02.

The Ricker system presents another unique sensitivity profile. Here, `res_size` has a relatively flat response curve, suggesting less sensitivity within this range compared to other systems. `w_scale` is highly critical, with a sharp V-shaped minimum. The `spectral_radius` plot shows that the global optimum of 0.71 is located on the slope of a broader minimum. For the EICM neuron parameters, `f` shows a preference for smaller values. The parameter `h` is highly sensitive, with a distinct local minimum. Finally, `beta` again demonstrates high sensitivity, with its local minimum near the globally optimized value of 2.14.

Across all three systems, parameters defining the EICM neuron's core dynamics, such as `f` and `beta`, along with reservoir properties like `spectral_radius` and `w_scale`, consistently emerge as highly influential. Small deviations from their optimal values can

TABLE 8 Best LSTM parameters for different chaotic systems and prediction steps.

Logistic system				
Parameter	One-step	Two-step	Three-step	Four-step
hidden_size	174.0000	201.0000	216.0000	201.0000
num_layers	1.0000	1.0000	1.0000	1.0000
lr	0.0140	0.0234	0.0114	0.0234
batch_size	133.0000	150.0000	71.0000	150.0000
dropout	0.1461	0.2185	0.3184	0.2185
sequence_length	16.0000	19.0000	6.0000	19.0000
Sine System				
Parameter	One-step	Two-step	Three-step	Four-step
hidden_size	136.0000	245.0000	201.0000	201.0000
num_layers	1.0000	1.0000	1.0000	1.0000
lr	0.0001	0.0104	0.0234	0.0234
batch_size	128.0000	229.0000	150.0000	150.0000
dropout	0.3747	0.2605	0.2185	0.2185
sequence_length	10.0000	6.0000	19.0000	19.0000
Ricker system				
Parameter	One-step	Two-step	Three-step	Four-step
hidden_size	201.0000	128.0000	194.0000	244.0000
num_layers	1.0000	1.0000	1.0000	1.0000
lr	0.0234	0.0001	0.0048	0.0001
batch_size	186.0000	83.0000	64.0000	143.0000
dropout	0.2185	0.3453	0.3323	0.1000
sequence_length	19.0000	12.0000	19.0000	15.0000

lead to a significant increase in MSE, indicating that precise tuning of these parameters is crucial. In contrast, other parameters like `res_size` can exhibit broader optimal regions or system-dependent behaviors.

The analysis also reveals that the optimal hyperparameter configurations are distinct for each chaotic system, underscoring the necessity of system-specific optimization. While general trends can be observed, the precise values that minimize MSE vary considerably, highlighting the unique dynamic complexity of each system. This systematic analysis is fundamental for understanding the ESN-EICM's behavior and validating the configurations found by our optimization strategy.

It is noteworthy that the optimal parameter values marked by the vertical dashed lines (representing the global optimum found by Bayesian Optimization) do not always coincide with the minimum MSE in each one-dimensional sensitivity plot. This is an expected and insightful result. Bayesian Optimization finds the best set of hyperparameters in a high-dimensional space where all parameters interact. In contrast, our sensitivity analysis examines one-dimensional slices of this space by varying a single parameter while keeping others fixed at their global optimal values. The discrepancy between the global optimum and the local minima in these plots highlights the strong coupling and interdependencies among the hyperparameters. It demonstrates that the ideal value for one parameter is contingent on the values of others, reinforcing the necessity of using a multi-dimensional optimization strategy like Bayesian Optimization rather than relying on one-at-a-time parameter tuning.

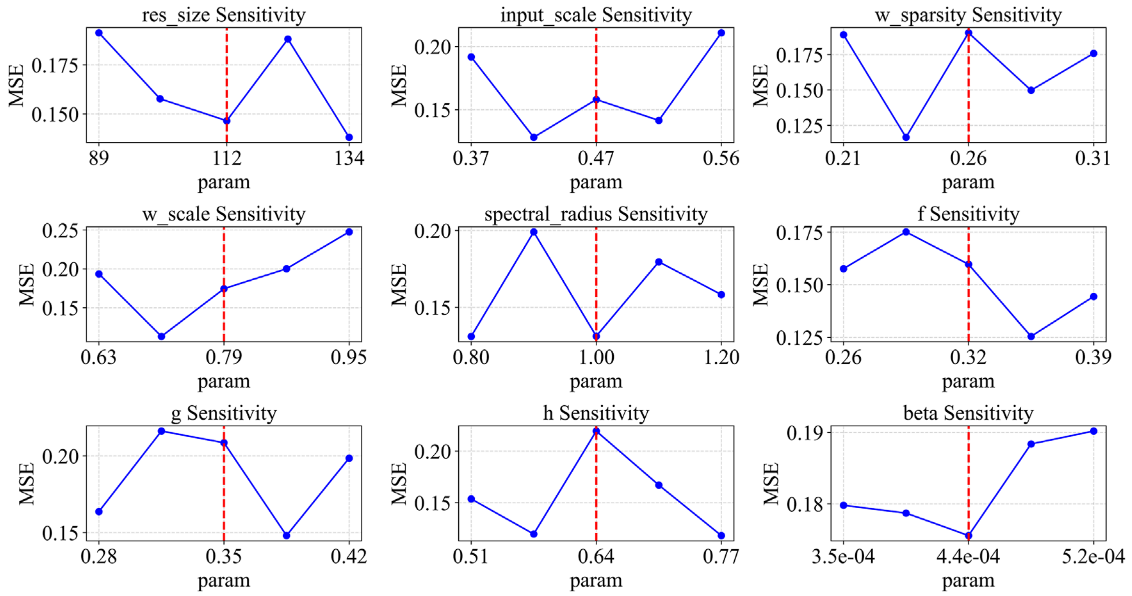


FIGURE 3
Parameter sensitivity analysis of ESN-EICM in logistic system.

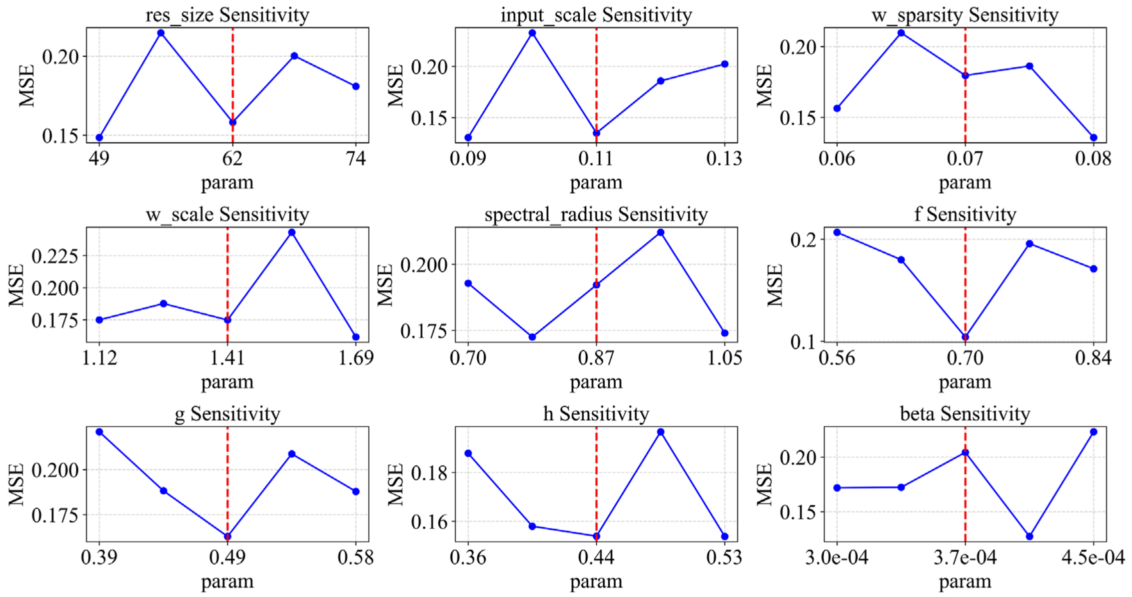


FIGURE 4
Parameter sensitivity analysis of ESN-EICM in sine system.

4.6 Prediction performance evaluation

4.6.1 One-step prediction performance

The efficacy of the ESN-EICM model for one-step prediction was rigorously evaluated on three canonical chaotic systems: the Logistic system, the Sine system, and the Ricker system. Its performance was benchmarked against both traditional ESN and LSTM architectures. The comprehensive results, encompassing both quantitative metrics and qualitative visualizations, consistently underscore the superior predictive accuracy and robustness of the proposed ESN-EICM.

Quantitative analysis, detailed in Table 9, reveals that ESN-EICM generally achieves lower error metrics compared to ESN and LSTM. For instance, in predicting the Logistic system, ESN-EICM recorded a MSE of 5.3281×10^{-8} and a RMSE of 2.3083×10^{-4}. This trend of superior accuracy was also observed for the Sine system, with ESN-EICM yielding an MSE of 2.5966×10^{-8} and an RMSE of 1.6114×10^{-4}. These figures are notably lower

TABLE 9 One-step prediction performance by different chaotic system.

Logistic system			
Metric	ESN-EICM	ESN	LSTM
MSE	5.3281×10^{-8}	1.9481×10^{-7}	2.1895×10^{-7}
RMSE	2.3083×10^{-4}	4.4137×10^{-4}	4.6792×10^{-4}
MAE	1.1417×10^{-4}	1.0409×10^{-4}	3.5956×10^{-4}
R^2	0.9999	0.9999	0.9999
Expl. Var	0.9999	0.9999	0.9999
Max Error	6.6633×10^{-3}	1.8522×10^{-2}	1.3392×10^{-3}
Sine System			
Metric	ESN-EICM	ESN	LSTM
MSE	2.5966×10^{-8}	2.8248×10^{-8}	1.9621×10^{-7}
RMSE	1.6114×10^{-4}	1.6807×10^{-4}	4.4296×10^{-4}
MAE	7.6498e-05	5.6082×10^{-5}	3.4348e-04
R^2	0.9999	0.9999	0.9999
Expl. Var	0.9999	0.9999	0.9999
Max Error	5.3483×10^{-3}	6.4665×10^{-3}	1.4344×10^{-3}
Ricker System			
Metric	ESN-EICM	ESN	LSTM
MSE	1.9910e-07	4.2969×10^{-6}	3.5563e-05
RMSE	4.4621e-04	2.0729×10^{-3}	5.9635e-03
MAE	3.0509e-04	9.7816×10^{-4}	5.0354e-03
R^2	0.9999	0.9999	0.9999
Expl. Var	0.9999	0.9999	0.9999
Max Error	1.0754×10^{-2}	6.1586×10^{-2}	1.6457×10^{-2}

than those of the benchmark models, indicating a more precise alignment between ESN-EICM's predictions and the ground truth. Even for the Ricker system, the MSE of ESN-EICM of 1.9910×10^{-7} demonstrated a substantial improvement over the traditional ESN (4.2969×10^{-6}). Furthermore, ESN-EICM exhibited competitive MAE values across all systems, achieving the lowest MAE for the Sine map (7.6498×10^{-5}) and demonstrating MAEs comparable to or better than ESN and LSTM for the Logistic and Ricker system. While all models displayed high R^2 and Explained Variance scores, ESN-EICM distinguished itself by coupling this high explanatory power with consistently lower prediction errors and well-contained Maximum Errors, as seen for the Logistic (6.6633×10^{-3}) and Sine system (5.3483×10^{-3}), highlighting its predictive stability.

The qualitative visualizations further reinforce these quantitative findings. The one-step prediction trajectories, illustrated in Figure 6 (assuming this figure shows predicted vs. true time series plots), demonstrate the ESN-EICM's capability to closely track the actual system dynamics for the Logistic system (a), Sine map (b), and Ricker model (c), even through complex behavioral regimes. The temporal evolution of absolute prediction error, as depicted in Figure 7, confirms the stability of ESN-EICM's predictions, with errors remaining at consistently low levels (typically on the order of 10^{-3} or less) without significant accumulation or divergence for all three systems. Moreover, the phase space reconstructions presented in Figure 8 show a remarkable congruence between the attractors generated from ESN-EICM's predictions (red markers) and those of the true systems (blue markers). The model accurately reproduces the characteristic geometries of Logistic map's phase space plot (a), the Sine map's phase space plot (b), and the Ricker model's phase space plot (c), indicating to its proficiency in capturing the underlying nonlinear dynamics. Finally, the scatter plots in Figure 9, which compare predicted values against true values, show data points tightly clustered around the ideal $y = x$ diagonal for all systems (a, b, c). This high degree of linearity and consistency provides direct visual evidence of ESN-EICM's superior predictive precision.

In conclusion, the combined evidence from quantitative metrics and qualitative visualizations strongly supports the enhanced performance of the ESN-EICM model in one-step prediction tasks for chaotic time series. It consistently outperforms or matches established models like ESN and LSTM in accuracy and robustness, while also demonstrating a strong capability to learn and replicate the intricate dynamics inherent in these complex systems. These results firmly establish ESN-EICM as a promising and effective tool for nonlinear time series prediction.

4.6.2 Multi-step prediction performance

To further assess the predictive capabilities of the proposed ESN-EICM model, comprehensive multi-step prediction experiments were conducted for two-step, three-step, and four-step ahead forecasts. These predictions were performed on the Logistic, Sine, and Ricker chaotic systems, and the performance of ESN-EICM was benchmarked against standard ESN and LSTM models. The quantitative results for these multi-step predictions are detailed in Table 10 (two-step), Table 11 (three-step), and Table 12 (four-step). Visualizations of the ESN-EICM's multi-step prediction trajectories, corresponding absolute errors, phase space reconstructions, and scatter plots of predicted *versus* true values are presented in Figures 10–13, respectively.

Visually, Figures 10–13 collectively demonstrate the robust performance of the ESN-EICM model in multi-step prediction. Figure 10 shows that the predicted trajectories for all three chaotic systems ((a) Logistic, (b) Sine, (c) Ricker) closely follow the true system dynamics even over extended horizons. The absolute errors, as depicted in Figure 11, remain consistently low and bounded over the 2000 time steps, indicating the stability and accuracy of the ESN-EICM. The fidelity of the model in capturing the underlying dynamics of these chaotic systems is further highlighted by the phase space reconstructions in Figure 12, where the predicted attractors exhibit excellent agreement with the true attractors. Moreover, the scatter plots in Figure 13 show data points tightly clustered around the ideal diagonal line (Predicted Value = True Value),

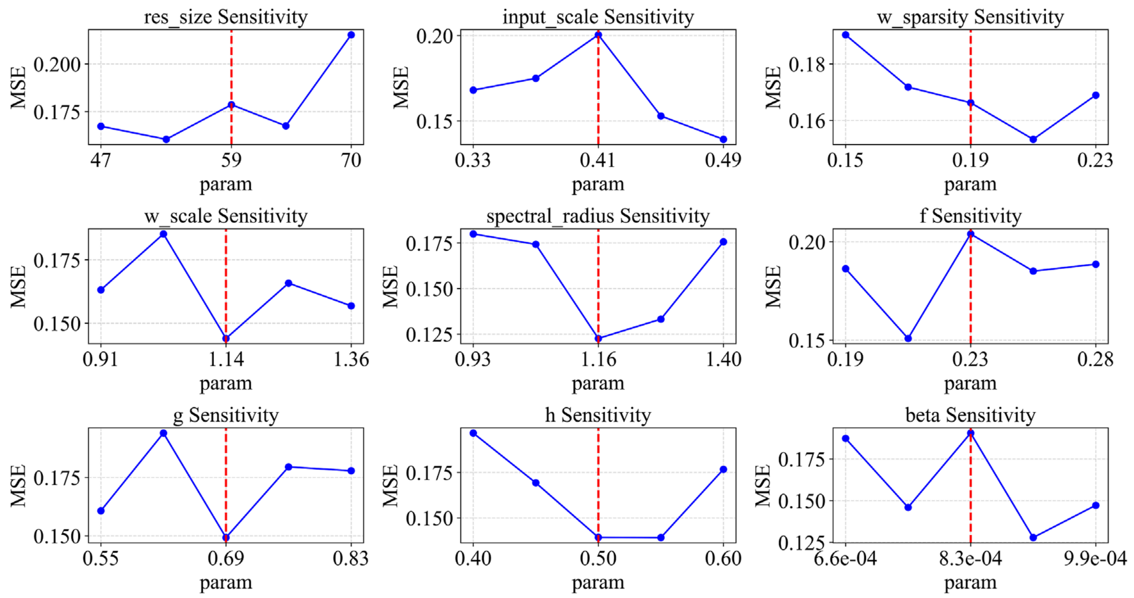


FIGURE 5
Parameter sensitivity analysis of ESN-EICM in ricker system.

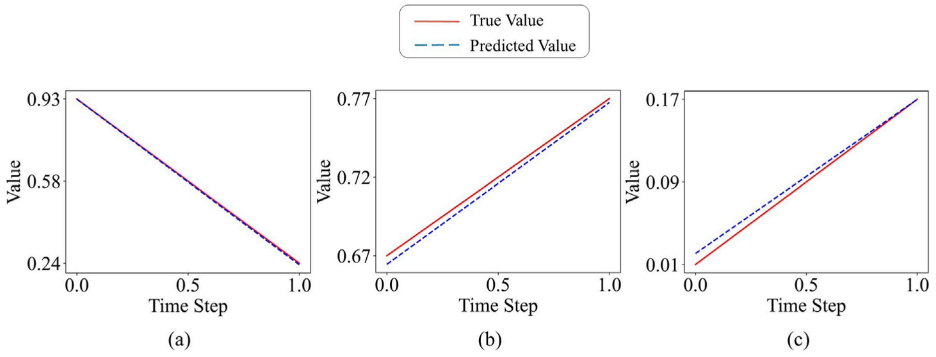


FIGURE 6
ESN-EICM One-step Prediction in Different Chaotic Systems: **(a)** the Logistic system, **(b)** the Sine system, and **(c)** the Ricker system.

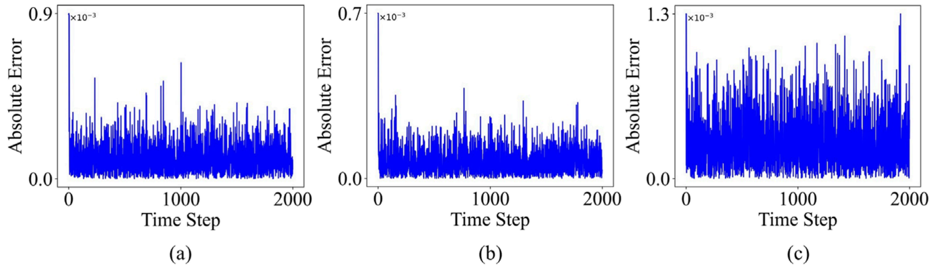
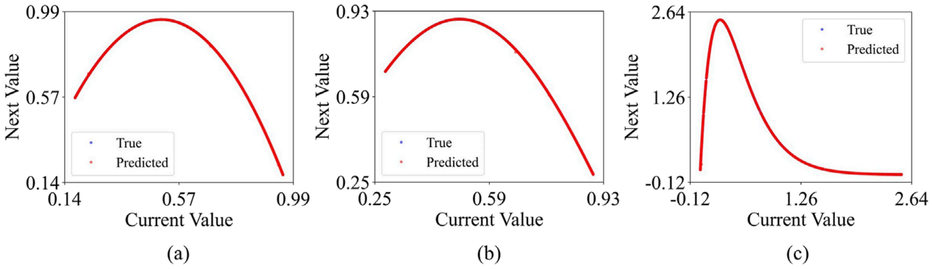
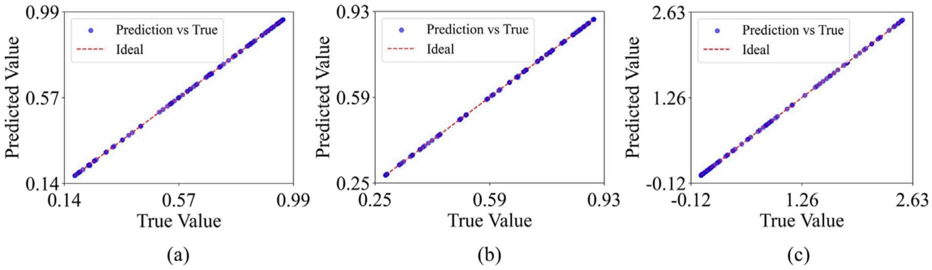


FIGURE 7
ESN-EICM One-step Prediction Absolute Error Over Time in Different Chaotic Systems: **(a)** the Logistic system, **(b)** the Sine system, and **(c)** the Ricker system.



(a) (b) (c)

FIGURE 8
ESN-EICM One-step Prediction Phase Space Reconstruction in Different Chaotic Systems: **(a)** the Logistic system, **(b)** the Sine system, and **(c)** the Ricker system.



(a) (b) (c)

FIGURE 9
ESN-EICM One-step Prediction Accuracy: Predicted vs. True Values in Different Chaotic Systems: **(a)** the Logistic system, **(b)** the Sine system, and **(c)** the Ricker system.

underscoring the high point-wise accuracy of the ESN-EICM in multi-step prediction scenarios.

Quantitatively, the ESN-EICM model consistently outperforms both ESN and LSTM across nearly all metrics and prediction horizons for the three chaotic systems.

For the Logistic system, in 2-step predictions (Table 10), ESN-EICM achieved an MSE of 3.3125×10^{-7}, markedly lower than ESN (7.0259×10^{-7}) and LSTM (1.3184×10^{-5}). This superiority in terms of MSE, RMSE, and MAE was maintained and often accentuated as the prediction horizon increased. For instance, in 4-step predictions (Table 12), ESN-EICM's MSE was 5.2171×10^{-7} and MAE was 4.8916×10^{-5}, significantly better than ESN (MSE: 1.8035×10^{-6}, MAE: 8.2818×10^{-4}) and LSTM (MSE: 4.8622×10^{-6}, MAE: 1.5299×10^{-3}).

In the case of the Sine system, ESN-EICM also demonstrated consistently lower MSE, RMSE, and MAE. For 2-step predictions (Table 10), ESN-EICM's MSE (6.6947×10^{-8}) was superior to both ESN (6.6527×10^{-7}) and LSTM (1.1673×10^{-7}). While LSTM occasionally yielded a lower Max Error (e.g., 9.7656×10^{-4} for 2-steps), ESN-EICM's average error metrics remained dominant. This trend persisted for 4-step predictions (Table 12), where ESN-EICM's MAE of 6.3992×10^{-5} was substantially lower than ESN's 5.6653×10^{-4} and LSTM's 7.1513×10^{-4}.

The Ricker system results particularly highlight the strength of the ESN-EICM. For 2-step predictions (Table 10), ESN-EICM's MSE (3.3589×10^{-7}) was already an order of magnitude better than LSTM (7.2480×10^{-6}) and significantly better than ESN (2.1188×10^{-6}). This advantage became even more pronounced at longer horizons. For 3-step predictions (Table 11), ESN-EICM achieved an exceptionally low MSE of 4.2735×10^{-8}, two orders of magnitude smaller than ESN (7.9467×10^{-6}) and LSTM (1.0065×10^{-5}). It also recorded the lowest Max Error (8.4453×10^{-3}) in this scenario. This pattern continued for 4-step predictions (Table 12), where ESN-EICM's MSE (1.5326×10^{-7}) and Max Error (1.5122×10^{-2}) were notably superior to the comparator models.

Across all tested scenarios, R^2 and Explained Variance values were consistently close to 0.9999 for all models, indicating a good general fit. However, the significant differences in MSE, RMSE, and MAE clearly underscore the enhanced precision and robustness of the ESN-EICM model for multi-step chaotic time series prediction. The sustained low error levels, even as the prediction horizon extends, suggest that ESN-EICM effectively captures the complex underlying dynamics and is less prone to error accumulation compared to standard ESN and LSTM approaches in these multi-step prediction tasks.

4.7 Training time comparison

In this section, we describe the measurement of the execution times of the three models for the same prediction task. The computer configuration is as follows:

- RAM: 32.0 GB (31.2 GB available)

TABLE 10 Two-step prediction performance metrics by different chaotic systems.

Logistic system			
Metric	**ESN-EICM**	**ESN**	**LSTM**
MSE	3.3125×10^{-7}	7.0259×10^{-7}	1.3184×10^{-5}
RMSE	5.7554×10^{-4}	8.3821×10^{-4}	3.6310×10^{-3}
MAE	1.5960×10^{-4}	4.3627×10^{-4}	2.7226×10^{-3}
R^2	0.9999	0.9999	0.9998
Expl. Var	0.9999	0.9999	0.9999
Max Error	1.7849×10^{-2}	7.7281×10^{-3}	9.5704×10^{-3}
Sine System			
Metric	**ESN-EICM**	**ESN**	**LSTM**
MSE	6.6947×10^{-8}	6.6527×10^{-7}	1.1673×10^{-7}
RMSE	2.5874×10^{-4}	8.1564×10^{-4}	3.4165×10^{-4}
MAE	1.2735×10^{-4}	4.6493×10^{-4}	2.7756×10^{-4}
R^2	0.9999	0.9999	0.9999
Expl. Var	0.9999	0.9999	0.9999
Max Error	4.6815×10^{-3}	4.7926×10^{-3}	9.7656×10^{-4}
Ricker System			
Metric	**ESN-EICM**	**ESN**	**LSTM**
MSE	3.3589×10^{-7}	2.1188×10^{-6}	7.2480×10^{-6}
RMSE	5.7956×10^{-4}	1.4556×10^{-3}	2.6922×10^{-3}
MAE	1.1633e-04	9.0384×10^{-4}	2.1195e-03
R^2	0.9999	0.9999	0.9999
Expl. Var	0.9999	0.9999	0.9999
Max Error	2.2803×10^{-2}	1.3637×10^{-2}	1.6870×10^{-2}

TABLE 11 Three-step prediction performance metrics by different chaotic systems.

Logistic system			
Metric	**ESN-EICM**	**ESN**	**LSTM**
MSE	1.6931×10^{-7}	1.2063×10^{-6}	5.9313×10^{-6}
RMSE	4.1147×10^{-4}	1.0983×10^{-3}	2.4354×10^{-3}
MAE	$4.5822e\times10^{-5}$	7.2919×10^{-4}	1.8768×10^{-3}
R^2	0.9999	0.9999	0.9999
Expl. Var	0.9999	0.9999	0.9999
Max Error	1.4288×10^{-2}	8.0371×10^{-3}	6.5480×10^{-3}
Sine System			
Metric	**ESN-EICM**	**ESN**	**LSTM**
MSE	1.3017×10^{-7}	6.0365×10^{-7}	3.4272×10^{-7}
RMSE	3.6078×10^{-4}	7.7695×10^{-4}	5.8542×10^{-4}
MAE	2.0243×10^{-4}	5.3188×10^{-4}	4.2670×10^{-4}
R^2	0.9999	0.9999	0.9999
Expl. Var	0.9999	0.9999	0.9999
Max Error	3.2730×10^{-3}	4.2700×10^{-3}	2.8723×10^{-3}
Ricker System			
Metric	**ESN-EICM**	**ESN**	**LSTM**
MSE	4.2735×10^{-8}	7.9467×10^{-6}	1.0065×10^{-5}
RMSE	2.0672×10^{-4}	2.8190×10^{-3}	3.1726×10^{-3}
MAE	4.7811×10^{-5}	1.4569×10^{-3}	2.1353×10^{-3}
R^2	0.9999	0.9999	0.9999
Expl. Var	0.9999	0.9999	0.9999
Max Error	8.4453×10^{-3}	3.2162×10^{-2}	1.9023×10^{-2}

- Processor: AMD Ryzen 9 7945HX with Radeon Graphics, 2.50 GHz
- System: 64-bit operating system, x64-based processor
- Operating System: Windows 11 Pro, version 24H2
- Graphics Card: NVIDIA GeForce RTX 4060 Laptop GPU, 8 GB GPU VRAM, NVIDIA
- Python: 3.12.0
  - NumPy version: 1.26.4
  - SciPy version: 1.14.1
  - scikit-learn version: 1.5.2
  - Matplotlib version: 3.9.2
  - scikit-optimize version: 0.10.2
  - tqdm version: 4.66.5
  - torch version: 2.7.0
  - Pandas version: 2.2.3

The computational efficiency of the proposed ESN-EICM model was evaluated against traditional ESN and LSTM architectures, with total experiment times recorded in Table 13. A key advantage of reservoir computing models, including ESN and our ESN-EICM, lies in their training efficiency compared to deep learning models like LSTM. This is primarily because the reservoir's internal weights are fixed after initialization, and only the output weights are trained, typically through a computationally inexpensive linear regression. In contrast, LSTMs require iterative backpropagation through time and gradient descent over many epochs (70 epochs

TABLE 12 Four-step prediction performance metrics by different chaotic systems.

Logistic system			
Metric	ESN-EICM	ESN	LSTM
MSE	5.2171×10^{-7}	1.8035×10^{-6}	4.8622×10^{-6}
RMSE	7.2229×10^{-4}	1.3429×10^{-3}	2.2050×10^{-3}
MAE	4.8916×10^{-5}	8.2818×10^{-4}	1.5299×10^{-3}
R^2	0.9999	0.9999	0.9999
Expl. Var	0.9999	0.9999	0.9999
Max Error	2.4936×10^{-2}	1.4678×10^{-2}	1.2481×10^{-2}
Sine System			
Metric	ESN-EICM	ESN	LSTM
MSE	$4.4554 \times {}^{1}0\text{-}7$	8.6754×10^{-7}	$1.0796 \times {}^{1}0\text{-}6$
RMSE	6.6748×10^{-4}	9.3142×10^{-4}	1.0390×10^{-3}
MAE	6.3992×10^{-5}	5.6653×10^{-4}	7.1513×10^{-4}
R^2	0.9999	0.9999	0.9999
Expl. Var	0.9999	0.9999	0.9999
Max Error	2.3050×10^{-2}	7.2835×10^{-3}	6.6527×10^{-3}
Ricker System			
Metric	ESN-EICM	ESN	LSTM
MSE	1.5326×10^{-7}	2.3259×10^{-5}	2.2010×10^{-5}
RMSE	3.9148×10^{-4}	4.8227×10^{-3}	4.6914×10^{-3}
MAE	9.0672×10^{-5}	2.4204×10^{-3}	2.6845×10^{-3}
R^2	0.9999	0.9999	0.9999
Expl. Var	0.9999	0.9999	0.9999
Max Error	1.5122×10^{-2}	5.8588×10^{-2}	5.1520×10^{-2}

in our setup, as per 4.3.3), leading to significantly longer training durations. This fundamental difference is evident across all prediction steps and chaotic systems, where both ESN-EICM and ESN consistently outperform LSTM in terms of speed, often by an order of magnitude. For instance, in one-step prediction for the Logistic system, ESN-EICM took 424.3 s, ESN took 398.8 s, while LSTM required 1649.0 s. This pattern persists and often magnifies in multi-step scenarios; for example, in four-step prediction for the Ricker system, ESN-EICM completed in 570.5 s, ESN in 4006.0 s, and LSTM in 2092.1 s.

When comparing ESN-EICM specifically with the standard ESN, the time performance presents a nuanced but ultimately favorable picture for ESN-EICM, particularly as prediction horizons extend. In one-step and two-step predictions, the ESN-EICM's runtime is generally comparable to that of the standard ESN, occasionally slightly higher. This marginal increase can be attributed to the more complex neuron dynamics within the ESN-EICM reservoir (as described in Section 3), which involve updates for feeding input F, output Y with sigmoid activation and noise, and a dynamic threshold E based on mean population activity. These richer per-neuron computations, while enhancing predictive power, incur a slight overhead per time step during reservoir state generation compared to the simpler activation function of a traditional ESN.

However, a significant advantage for ESN-EICM emerges in longer multi-step predictions, particularly at the four-step horizon. Here, ESN-EICM demonstrates substantially better time efficiency than the standard ESN. For example, in four-step prediction for the Logistic system, ESN-EICM took only 517.6 s, whereas ESN's time escalated to 3183.1 s. Similar substantial speed-ups for ESN-EICM over ESN were observed for the Sine (526.5 s vs. 2136.5 s) and Ricker (570.5 s vs. 4006.0 s) systems at four steps. This pronounced improvement in efficiency for ESN-EICM in more challenging, longer-term prediction tasks can be directly attributed to how its enhanced stability impacts the Bayesian Optimization process. The inherent stability of the EICM neurons—stemming from features like adaptive thresholds and bounded activations—creates a "smoother" hyperparameter landscape for the optimizer to explore. This means that fewer parameter combinations lead to divergent or numerically unstable models, which would otherwise result in extremely high error values (penalties) and waste optimization calls. For the standard ESN, finding a stable parameter set for long-term iterative prediction can be more difficult, leading the optimizer to spend more time evaluating poorly performing or unstable regions. In contrast, the ESN-EICM's robustness means that a larger proportion of the hyperparameter space yields valid, stable models, allowing the Bayesian optimizer to more quickly identify near-optimal configurations in fewer iterations. Therefore, the "faster convergence" mentioned in the abstract is not about the speed of a single training run, but the efficiency of the entire hyperparameter search process, which is significantly accelerated by the model's intrinsic stability.

5 Discussion

5.1 On model complexity and the design philosophy

A central tenet of traditional ESNs is the use of a simple, fixed reservoir to reduce training complexity. Our ESN-EICM model, by incorporating a more complex neuron, appears to diverge from this principle. This is a deliberate design choice motivated by the specific challenge of chaotic system prediction. Instead of seeking complexity through architectural modifications like deep or multi-reservoir structures, we pursue "internal complexification" at the neuronal level. The rationale is that the rich, adaptive dynamics of the EICM neuron—with its coupled feedback and adaptive thresholds—can generate a more expressive variety of temporal patterns. This allows a reservoir of a given size to map the input into a higher-quality, more dynamically rich state space. The

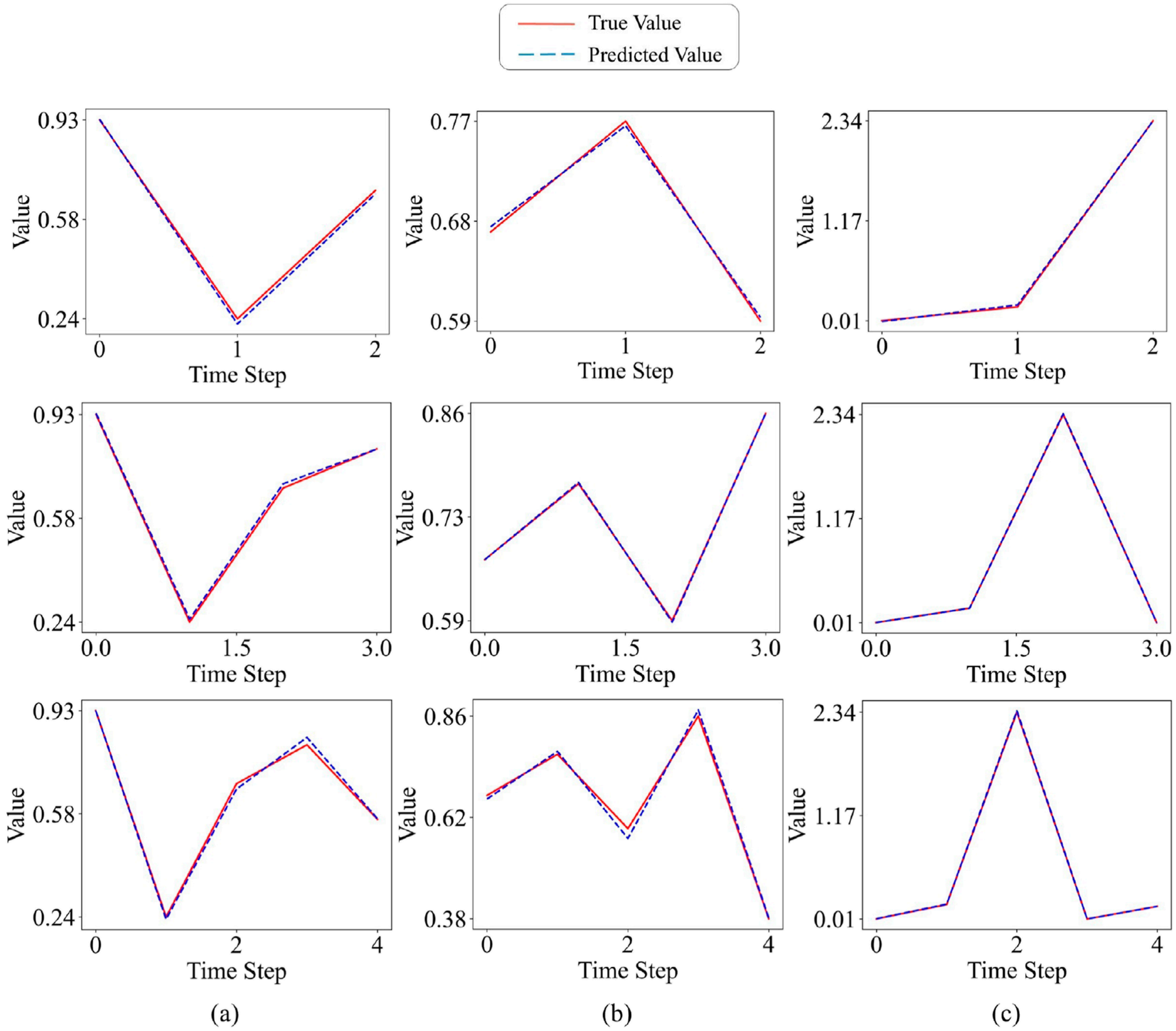


FIGURE 10
ESN-EICM multi-step Prediction in Different Chaotic Systems: **(a)** the Logistic system, **(b)** the Sine system, and **(c)** the Ricker system.

performance gains observed, particularly in multi-step prediction, suggest that for highly complex and sensitive systems like the ones studied, the benefits of enhanced neuronal dynamics outweigh the modest increase in per-neuron computational cost. This approach offers a valuable alternative to topological optimization, focusing instead on the intrinsic computational capabilities of the reservoir's constituent elements.

5.2 Robustness against sensitivity in chaotic systems

The introduction mentions the "butterfly effect," the extreme sensitivity of chaotic systems to initial conditions. The ESN-EICM's strong performance in multi-step prediction suggests an inherent robustness against this sensitivity. This can be attributed to several design features. The adaptive threshold mechanism (E_t) acts to normalize the network's overall activity, preventing small initial errors from being catastrophically amplified and causing state divergence. The internal feedback ($f \cdot F_{t-1}$) and global coupling ($W \cdot Y_{t-1}$) create a rich, stable attractor dynamic within the reservoir that is resistant to minor perturbations. Finally, the injection of a small amount of noise can be seen as a form of regularization that prevents the model from overfitting to a specific trajectory, thereby improving its ability to generalize and remain on the true system's attractor for longer during iterative prediction.

6 Limitations and future work

While the proposed ESN-EICM model has demonstrated significant advantages in prediction chaotic time series, certain limitations and avenues for future research warrant discussion.

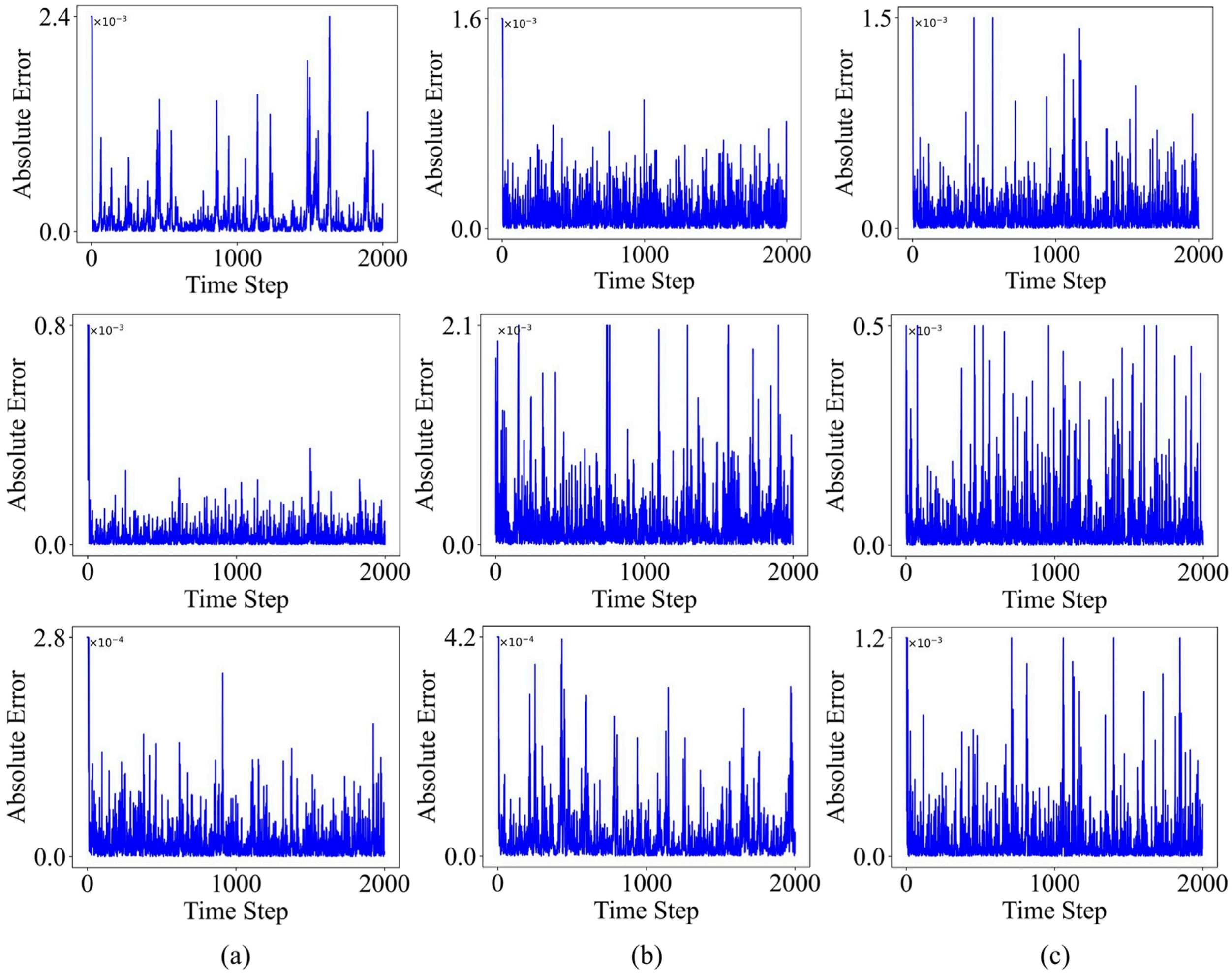


FIGURE 11
ESN-EICM multi-step Prediction Absolute Error Over Time in Different Chaotic Systems: **(a)** the Logistic system, **(b)** the Sine system, and **(c)** the Ricker system.

6.1 Limitations

1. Although Bayesian Optimization (BO) is more efficient than grid search or random search, optimizing a relatively large number of hyperparameters (10 in this study for ESN-EICM, as shown in Table 1) can still be computationally intensive, especially if each evaluation (training and validating the model) is time-consuming due to large reservoir sizes or long time series. The 50 calls to BO used in this study represent a trade-off between search thoroughness and computational budget.
2. The computational complexity of standard ESN training involves matrix operations that scale with reservoir size (N). While the EICM neuron introduces a constant factor overhead per neuron, the fundamental scaling properties of RC remain. For extremely large reservoirs, the memory and computational demands for storing and operating on the reservoir weight matrix W and collecting states could become a bottleneck.
3. The ESN-EICM was evaluated on three discrete chaotic systems, which are well-defined and exhibit specific types of chaos. Real-world time series often contain multiple sources of noise, non-stationarities, and varying types of underlying dynamics that were not explicitly addressed or modeled in this study beyond the inherent learning capacity of the reservoir. The model's performance on such diverse and potentially more complex real-world datasets remains to be extensively validated.
4. While the EICM neuron model is biologically inspired and its mechanisms (adaptive threshold, feedback) are more transparent than the internal workings of an LSTM cell, the collective dynamics of a large reservoir of interconnected EICM neurons can still be complex to analyze and interpret fully. Understanding precisely how the EICM parameters (f, g, h, β) contribute to specific dynamic properties like memory capacity or nonlinearity at the network level requires further investigation.
5. The EICM neuron parameters (f, g, h, β) are optimized via BO and then fixed during training and inference. For highly non-stationary time series, dynamically adapting these internal

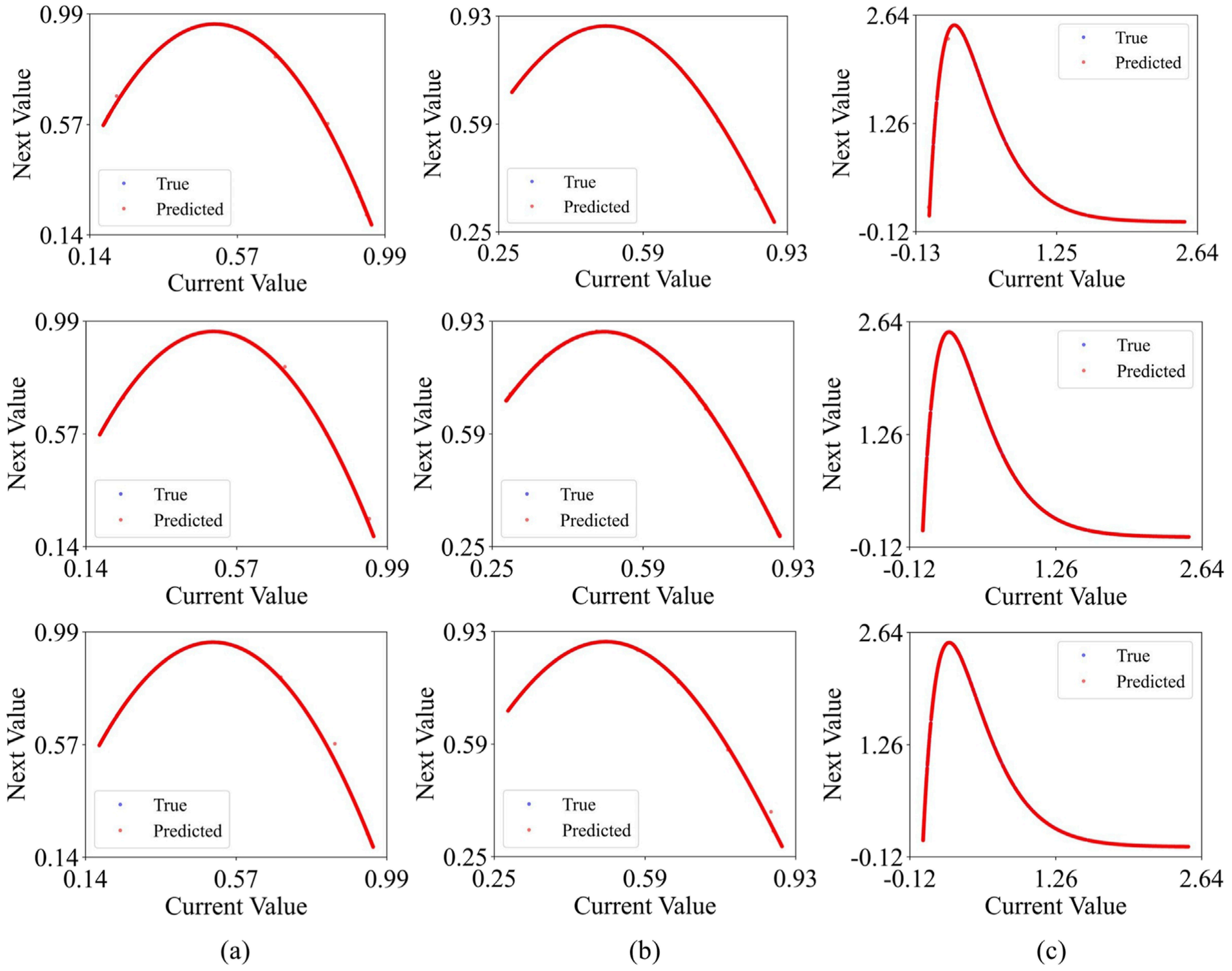


FIGURE 12
ESN-EICM multi-step Prediction Phase Space Reconstruction in Different Chaotic Systems: **(a)** the Logistic system, **(b)** the Sine system, and **(c)** the Ricker system.

neuron parameters online could potentially offer further performance improvements.

6. While the overall training is efficient, the EICM neuron itself is computationally more demanding than a standard tanh or sigmoid neuron due to the multiple state updates (F, E, Y) required at each time step. This introduces a constant factor overhead in the reservoir state generation phase, which could become noticeable for very large reservoirs or extremely long time series.

6.2 Future work

Based on the promising results and current limitations, several directions for future research can be pursued:

1. Exploring more sophisticated or parallelized Bayesian optimization techniques, or meta-learning approaches to warm-start BO, could further reduce the hyperparameter tuning cost. Investigating gradient-based optimization for certain EICM parameters, if feasible, might also be an avenue.
2. Developing mechanisms for online adaptation of key EICM parameters (f, g, h, β) based on the input statistics or prediction error could enhance the model's adaptability to changing dynamics in non-stationary environments.
3. Combining ESN-EICM with other machine learning techniques could yield synergistic benefits. For example, using attention mechanisms in the output layer or employing ESN-EICM as a feature extractor for a subsequent shallow neural network could be explored.
4. A more in-depth theoretical analysis of the ESN-EICM, focusing on its memory capacity, echo state property conditions with EICM neurons, and stability criteria, would provide a stronger foundational understanding.
5. Extending the application of ESN-EICM to a wider range of challenging real-world chaotic and complex time series from domains such as finance (stock market prediction), climate science (weather prediction), engineering (system

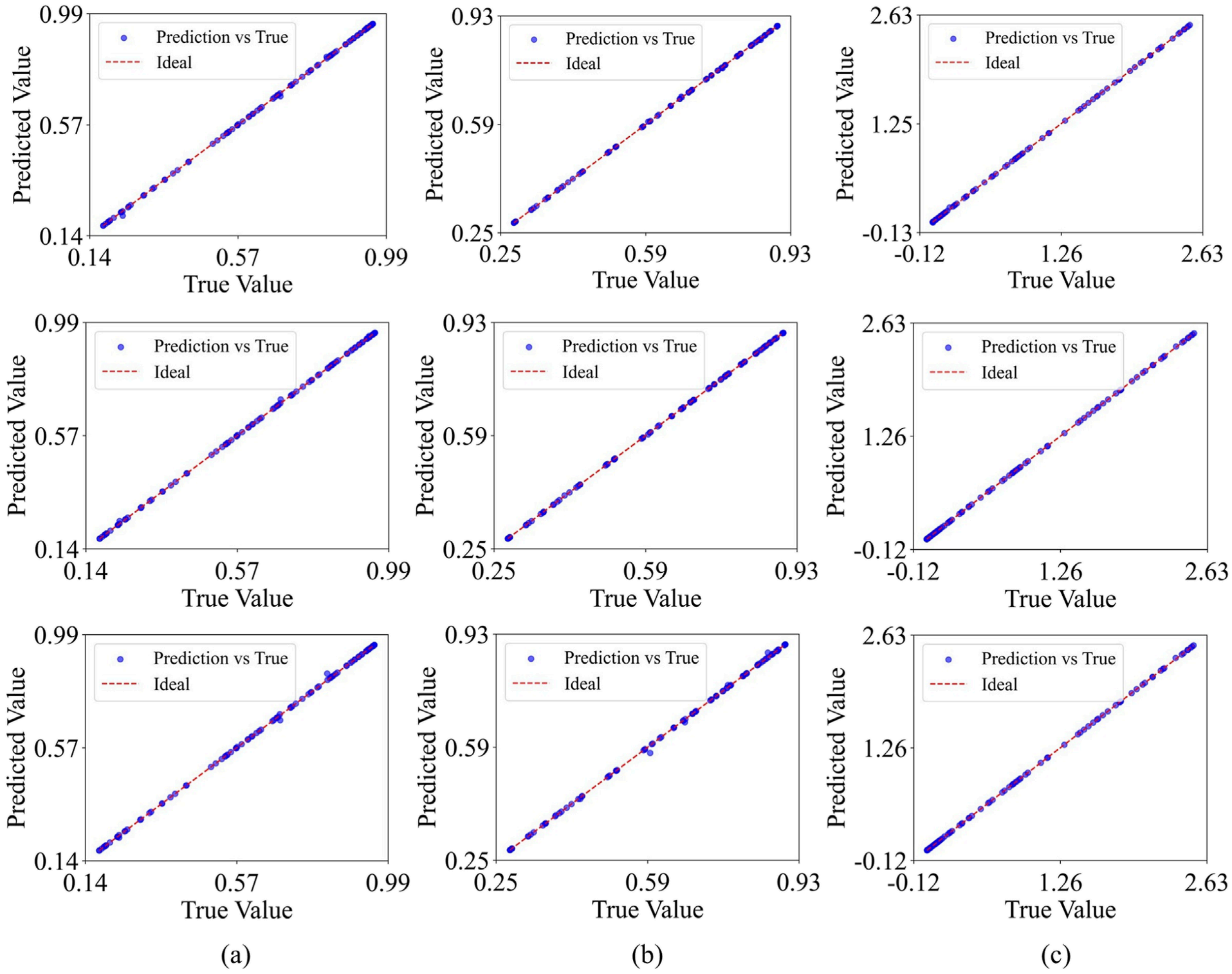


FIGURE 13
ESN-EICM multi-step Prediction Accuracy: Predicted vs. True Values in Different Chaotic Systems: **(a)** the Logistic system, **(b)** the Sine system, and **(c)** the Ricker system.

identification), and neuroscience (EEG signal analysis) would be crucial for demonstrating its practical utility.

6. Investigating the integration of other sophisticated, biologically plausible neuron models (e.g., Izhikevich neurons, adaptive exponential integrate-and-fire models) within the RC framework could lead to further advancements in time series prediction.Further exploration into neuromorphic hardware implementations could also be beneficial, drawing insights from ongoing research into memristive systems and their complex dynamics for specialized tasks [35]. Similarly, advancements in cellular neural networks coupled with novel devices like memristors also contribute to the broader landscape of hardware-oriented neural computation [36]. Exploring efficient hardware avenues, such as FPGA implementations for complex and novel neural architectures, remains an important direction [37].
7. Beyond hyperparameter optimization, exploring techniques for optimizing the reservoir's topology (e.g., using pruning or growing methods guided by EICM neuron activity) could lead to more efficient and specialized reservoir structures.

Addressing these limitations and exploring these future research directions will contribute to advancing the field of reservoir computing and its application to complex time series analysis.

7 Conclusion

In this work, we introduced the Echo State Network Based on Enhanced Intersecting Cortical Model (ESN-EICM), a novel reservoir computing framework designed for accurate and efficient prediction of dicrete chaotic systems. Recognizing the limitations of traditional deep learning models in terms of computational cost and interpretability, and the constraints of standard ESNs concerning simplistic neuron dynamics and hyperparameter sensitivity, the ESN-EICM offers a compelling alternative. The core innovation lies in the integration of biologically inspired EICM neurons into the reservoir, characterized by continuous sigmoid activation, global mean-driven adaptive thresholds, and explicit inter-neuron feedback. This design endows the reservoir with richer internal dynamics, better suited for capturing the complex patterns inherent

TABLE 13 Total experiment time for different prediction steps, chaotic systems, and models.

One-step prediction			
System	ESN-EICM	ESN	LSTM
Logistic	424.3 (s)	398.8 (s)	1649.0 (s)
Sine	378.5 (s)	433.1 (s)	1761.0 (s)
Ricker	376.5 (s)	460.9 (s)	1700.0 (s)
Two-step Prediction			
System	ESN-EICM	ESN	LSTM
Logistic	513.0 (s)	486.3 (s)	2737.3 (s)
Sine	521.1 (s)	417.8 (s)	2518.2 (s)
Ricker	583.7 (s)	461.2 (s)	2578.1 (s)
Three-step Prediction			
System	ESN-EICM	ESN	LSTM
Logistic	1034.4 (s)	434.6 (s)	2406.2 (s)
Sine	1108.3 (s)	440.5 (s)	2711.0 (s)
Ricker	926.5 (s)	456.4 (s)	2533.5 (s)
Four-step Prediction			
System	ESN-EICM	ESN	LSTM
Logistic	517.6 (s)	3183.1 (s)	2797.9 (s)
Sine	526.5 (s)	2136.5 (s)	2904.1 (s)
Ricker	570.5 (s)	4006.0 (s)	2092.1 (s)

in chaotic systems. Furthermore, the adoption of a Bayesian Optimization strategy systematically addresses the challenge of hyperparameter tuning, leading to robust and near-optimal model configurations.

Our comprehensive experimental evaluation on the Logistic, Sine, and Ricker chaotic systems unequivocally demonstrated the ESN-EICM's superiority. In both one-step and challenging multi-step prediction tasks (up to four steps ahead), the ESN-EICM consistently outperformed both standard ESN and LSTM models, as evidenced by significantly lower Mean Squared Error, Root Mean Squared Error, and Mean Absolute Error. Qualitative analyses, including prediction trajectory plots, error distributions, phase space reconstructions, and scatter plots, further visually corroborated the enhanced accuracy and stability of the ESN-EICM. Notably, while maintaining the characteristic training efficiency of RC models over LSTMs, the ESN-EICM often exhibited comparable or even superior total experiment times (including optimization) compared to standard ESNs in multi-step scenarios, attributed to the increased stability and expressiveness of the EICM neurons facilitating a more efficient hyperparameter search.

The successful application of EICM neurons within an ESN framework, coupled with automated hyperparameter optimization, highlights the potential of integrating more sophisticated, biologically plausible mechanisms into reservoir computing. The ESN-EICM stands as a robust, accurate, and computationally viable tool for modeling and predicting chaotic time series, paving the way for further research into neuro-inspired computing paradigms for complex dynamical systems. Future work will focus on extending its application to diverse real-world problems, exploring dynamic adaptation of neuron parameters, and conducting further theoretical analysis of its properties.

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

XW: Conceptualization, Writing – review and editing, Writing – original draft, Visualization, Project administration, Data curation, Methodology. PM: Resources, Conceptualization, Funding acquisition, Writing – review and editing, Writing – original draft, Validation, Software. JnL: Resources, Investigation, Writing – original draft, Funding acquisition, Writing – review and editing, Data curation, Project administration, Supervision. JzL: Project administration, Funding acquisition, Supervision, Writing – review and editing, Writing – original draft, Data curation, Resources, Validation. YM: Resources, Funding acquisition, Formal Analysis, Writing – review and editing, Project administration, Supervision, Conceptualization, Writing – original draft, Data curation.

Funding

The author(s) declare that financial support was received for the research and/or publication of this article. Some experiments are supported by the Supercomputing Center of Lanzhou University. Additionasupport was provided in part by the Gansu Computing Center.

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.

Generative AI statement

The author(s) declare that no Generative AI was used in the creation of this manuscript.

Publisher's note

References

1. Rumelhart DE, Hinton GE, Williams RJ. Learning representations by back-propagating errors. *nature* (1986) 323:533–6. doi:10.1038/323533a0

2. Hochreiter S, Schmidhuber J. Long short-term memory. *Neural Comput* (1997) 9:1735–80. doi:10.1162/neco.1997.9.8.1735

3. Chung J, Gulcehre C, Cho K, Bengio Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555* (2014). doi:10.48550/arXiv.1412.3555

4. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. *Adv Neural Inf Process Syst* (2017) 30. doi:10.48550/arXiv.1706.03762

5. Salinas D, Flunkert V, Gasthaus J, Januschowski T. Deepar: probabilistic forecasting with autoregressive recurrent networks. *Int J Forecast* (2020) 36:1181–91. doi:10.1016/j.ijforecast.2019.07.001

6. Van Den Oord A, Dieleman S, Zen H, Simonyan K, Vinyals O, Graves A, et al. Wavenet: a generative model for raw audio. *arXiv preprint arXiv:1609.03499 12* (2016). doi:10.48550/arXiv.1609.03499

7. Lukoševičius M, Jaeger H. Reservoir computing approaches to recurrent neural network training. *Computer Sci Rev* (2009) 3:127–49. doi:10.1016/j.cosrev.2009.03.005

8. Gallicchio C, Micheli A, Pedrelli L. Deep reservoir computing: a critical experimental analysis. *Neurocomputing* (2017) 268:87–99. doi:10.1016/j.neucom.2016.12.089

9. He L, Xu Y, He W, Lin Y, Tian Y, Wu Y, et al. Network model with internal complexity bridges artificial intelligence and neuroscience. *Nat Comput Sci* (2024) 4:584–99. doi:10.1038/s43588-024-00674-9

10. Ekblad U, Kinser JM, Atmer J, Zetterlund N. The intersecting cortical model in image processing. *Nucl Instr Methods Phys Res Section A: Acc Spectrometers, Detectors Associated Equipment* (2004) 525:392–6. doi:10.1016/j.nima.2004.03.102

11. Tang Y, Jia S, Huang T, Yu Z, Liu JK. Implementing feature binding through dendritic networks of a single neuron. *Neural Networks* (2025) 189:107555. doi:10.1016/j.neunet.2025.107555

12. Zheng H, Zheng Z, Hu R, Xiao B, Wu Y, Yu F, et al. Temporal dendritic heterogeneity incorporated with spiking neural networks for learning multi-timescale dynamics. *Nat Commun* (2024) 15:277. doi:10.1038/s41467-023-44614-z

13. Shi X, Chen Z, Wang H, Yeung D-Y, Wong W-K, Woo W-C., et al. Convolutional lstm network: a machine learning approach for precipitation nowcasting. *Adv Neural Inf Process Syst* (2015) 28. doi:10.48550/arXiv.1506.04214

14. Awad M. Forecasting of chaotic time series using rbf neural networks optimized by genetic algorithms. *Int Arab J Inf Technology (Iajit)* (2017) 14.

15. Zhou H, Zhang S, Peng J, Zhang S, Li J, Xiong H, et al. Informer: beyond efficient transformer for long sequence time-series forecasting. *Proc AAAI Conf Artif intelligence* (2021) 35:11106–15. doi:10.1609/aaai.v35i12.17325

16. Karim F, Majumdar S, Darabi H, Chen S. Lstm fully convolutional networks for time series classification. *IEEE access* (2017) 6:1662–9. doi:10.1109/access.2017.2779939

17. Wang Z, Jiang R, Lian S, Yan R, Tang H. Adaptive smoothing gradient learning for spiking neural networks. In: *International conference on machine learning*. New York, NY, USA: Proceedings of Machine Learning Research (2023). p. 35798–816.

18. Ding J, Zhang J, Huang T, Liu JK, Yu Z. Assisting training of deep spiking neural networks with parameter initialization. *IEEE Trans Neural Networks Learn Syst* (2025) 1–14. doi:10.1109/tnnls.2025.3547774

19. Ma G, Yan R, Tang H. Exploiting noise as a resource for computation and learning in spiking neural networks. *Patterns* (2023) 4:100831. doi:10.1016/j.patter.2023.100831

20. Yang Z, Guo S, Fang Y, Yu Z, Liu JK. Spiking variational policy gradient for brain inspired reinforcement learning. *IEEE Trans Pattern Anal Machine Intelligence* (2024) 47:1975–90. doi:10.1109/tpami.2024.3511936

21. Hao X, Ma C, Yang Q, Wu J, Tan KC. Toward ultralow-power neuromorphic speech enhancement with spiking-fullsubnet. *IEEE Trans Neural Networks Learn Syst* (2025) 1–15. doi:10.1109/tnnls.2025.3566021

22. Xu M, Han M. Adaptive elastic echo state network for multivariate time series prediction. *IEEE Trans cybernetics* (2016) 46:2173–83. doi:10.1109/tcyb.2015.2467167

23. Qin Z, Tao X, Lu J, Tong W, Li GY. Semantic communications: principles and challenges. *arXiv preprint arXiv:2201.01389* (2021). doi:10.48550/arXiv.2201.01389

24. Yu K, Zhang T, Xu Q, Pan G, Wang H. Ts-snn: temporal shift module for spiking neural networks. *arXiv preprint arXiv:2505.04165* (2025).

25. Zhang M, Luo X, Wu J, Belatreche A, Cai S, Yang Y, et al. Toward building human-like sequential memory using brain-inspired spiking neural models. *IEEE Trans Neural Networks Learn Syst* (2025) 36:10143–55. doi:10.1109/tnnls.2025.3543673

26. Zhang J, Zhang M, Wang Y, Liu Q, Yin B, Li H, et al. Spiking neural networks with adaptive membrane time constant for event-based tracking. *IEEE Trans Image Process* (2025) 34:1009–21. doi:10.1109/tip.2025.3533213

27. Lian J, Yang Z, Liu J, Sun W, Zheng L, Du X, et al. An overview of image segmentation based on pulse-coupled neural network. *Arch Comput Methods Eng* (2021) 28:387–403. doi:10.1007/s11831-019-09381-5

28. Lukoševičius M. A practical guide to applying echo state networks. In: *Neural networks: tricks of the trade*. 2nd ed. Springer (2012). p. 659–86.

29. Jaeger H, Haas H. Harnessing nonlinearity: predicting chaotic systems and saving energy in wireless communication. *science* (2004) 304:78–80. doi:10.1126/science.1091277

30. Snoek J, Larochelle H, Adams RP. Practical bayesian optimization of machine learning algorithms. *Adv Neural Inf Process Syst* (2012) 25. doi:10.48550/arXiv.1206.2944

31. Bergstra J, Bengio Y. Random search for hyper-parameter optimization. *J machine Learn Res* (2012) 13:281–305.

32. Carandini M, Heeger DJ. Normalization as a canonical neural computation. *Nat Rev Neurosci* (2012) 13:51–62. doi:10.1038/nrn3136

33. Shen J, Ni W, Xu Q, Pan G, Tang H. Context gating in spiking neural networks: achieving lifelong learning through integration of local and global plasticity. *Knowledge-Based Syst* (2025) 311:112999. doi:10.1016/j.knosys.2025.112999

34. Sun P, Wu J, Zhang M, Devos P, Botteldooren D. Delayed memory unit: modeling temporal dependency through delay gate. *IEEE Trans Neural Networks Learn Syst* (2024) 36:10808–18. doi:10.1109/tnnls.2024.3490833

35. Yu F, He S, Yao W, Cai S, Xu Q. Quantitative characterization system for macroecosystem attributes and states. *IEEE Trans Computer-Aided Des Integrated Circuits Syst* (2025) 36:1–12. doi:10.13287/j.1001-9332.202501.031

36. Yu F, Su D, He S, Wu Y, Zhang S, Yin H. Resonant tunneling diode cellular neural network with memristor coupling and its application in police forensic digital image protection. *Chin Phys B* (2025) 34:050502. doi:10.1088/1674-1056/adb8bb

37. Yu F, Zhang S, Su D, Wu Y, Gracia YM, Yin H. Dynamic analysis and implementation of fpga for a new 4d fractional-order memristive hopfield neural network. *Fractal and Fractional* (2025) 9:115. doi:10.3390/fractalfract9020115

Frontiers in Physics

TYPE Original Research
PUBLISHED 10 September 2025
DOI 10.3389/fphy.2025.1640293

Analysis and application for the source-free $R_M LC$ circuits

Xiang Gao[1], Yuhan Qian[1], Shengpeng Li[1], Wenjuan Li[1], Yao Su[2] and Yue Liu[3]*

[1]Aerospace Times FeiHong Technology Company Limited, Beijing, China, [2]Chinese Academy of Sciences Institute of Automation, Beijing, China, [3]College of Electrical and Electronic Engineering, Changchun University of Technology, Changchun, China

Check for updates

OPEN ACCESS

EDITED BY
Viet-Thanh Pham,
Industrial University of Ho Chi Minh City, Vietnam

REVIEWED BY
Yujiao Dong,
Hangzhou Dianzi University, China
Fang Yao,
The University of Western Australia, Australia

*CORRESPONDENCE
Yue Liu,
✉ liuyue0423@ccut.edu.cn

RECEIVED 03 June 2025
ACCEPTED 11 August 2025
PUBLISHED 10 September 2025

CITATION
Gao X, Qian Y, Li S, Li W, Su Y and Liu Y (2025) Analysis and application for the source-free $R_M LC$ circuits.
Front. Phys. 13:1640293.
doi: 10.3389/fphy.2025.1640293

Introduction: Memristor systems and their application circuits have attracted growing research interest. When a memristor circuit/network is designed, both memristors and conventional electronic components are inevitably required, particularly energy storage elements (e.g., capacitors and inductors). It has found that most existing studies focus on oscillatory phenomena generated by memristive systems, such as chaotic attractors, period-doubling oscillations, spiking and bursting oscillations. However, there is a notable lack of literature exploring and analyzing the energy exchange between these components, as well as the resulting oscillatory behaviors and outcomes arising from such interactions. It is well known that the unit of a memristor, like that of a resistor, is the ohm (Ω). In general circuits, the energy exchange between resistors and energy storage elements can induce nonlinear behaviors such as step functions, damping phenomena, both of which stem from the energy exchange between resistors and capacitors/inductors. So, when a memristor (though physical implementations are rare, several classic mathematical models exist) exchanges energy with energy storage elements, will similar behaviors emerge?

Methods: In this paper, to advance the theoretical completeness of memristive systems and take the classical HP memristor model as an example, four source-free circuit topologies integrating memristors with energy-storage elements are investigated deeply. They are categorized into two types: $R_M C/R_M L$ circuits and series/parallel $R_M LC$ circuits. Firstly, through mathematical modeling, the four circuits are all found to be governed by transcendental equations. Secondly, two types of four-component source-free circuits are configured and analysis. Finally, the application circuits comprising four fundamental components was configured and explored.

Results and Discussion: Simulation results for the mathematical models of the four circuits demonstrate memristor states (R_0, kR_d) and energy-storage elements collectively regulate response characteristics, damped oscillatory and decay behavior. The active power and apparent power curves reveal distinct energy exchange behaviors between components, differing fundamentally from conventional *RL*, *RC*, and *RLC* circuits. These findings demonstrate that due to the presence of memristors, such circuits cannot be employed for step response generation, but are exclusively applicable for energy memorization and dissipation. Then, the following conclusion on two types of source-free circuits are demonstrated: (1) capacitor and inductor provide energy (i.e., ϕ and q) to the system, while memristors exhibit hysteretic behavior, collectively and fundamentally co-modulating oscillation modes and attractor phenomenon; (2) The dual characteristics of memristors—memory capability and energy dissipation—endow them with the potential to break the von Neumann bottleneck, making them essential candidates for implementing next-generation neural networks and AI systems. Finally, the application circuits reveal that even within the same circuit, varying memristor placements can lead to distinct topological configurations and divergent nonlinear output behaviors. This phenomenon further validates the unique characteristics of memristors as an emerging field. These findings establish a solid theoretical and experimental foundation for future exploration and development of memristive systems, including next-generation neural networks, artificial intelligence applications, and aerospace technologies.

KEYWORDS

memristor, R M C circuits, R M L circuits, Kirchhoff's circuit laws, energy exchange

1 Introduction

The memristor has been hypothesized as the fourth fundamental circuit component [1] and named. Its fingerprint is a pinched hysteresis loop [2], which is the recovery of pure resistance (no hysteresis) for high frequencies [1, 2]. Subsequently, the HP-memristor was proposed and fabricated as a canonical model. Due to the special electrical properties of nonvolatile memory and extraordinary nonlinearity, the memristor is usually adopted to design the artificial neural networks, memristive circuits, oscillation circuits and employed for unmanned aerial vehicles and motors. Currently, the discussion is focused not only on the application to computation and memory storage, but also on the fundamental role in nonlinear circuit theory. For instance, real synaptic circuits [3, 4]. biological neurons [5–8], behaviors of some neural network models [9, 10], and even some complex systems [11, 12] with memristors or memristor emulators [11, 13–16, 29]. Also, some meaning and interesting nonlinear behaviors and application have also been discovered and published [17], integration to mention just a few.

Totally, all above involved results contributed to improving the circuit theory and exploring related applications in the fields of circuit engineering, such as mathematics, physics, and aerospace circuits. According to the definition of the memristor, whose value depends on its internal parameter, which in turn has to evolve dynamically according either to current and voltage [2]. In other words, when the memristor was configured into one real circuit, the relationship ($d\Phi = R_M dq$) between its resistance and the state variable is the essence of characterizing the memristor [15, 18], which have been considered as the basic information to analyze the nonlinear and oscillation behaviors [15, 18], such as chaotic circuits [19], damping circuits [20], Bessel filter [21], diode bridge rectifier [22], and oscillation memristive circuit [23, 30], etc. Some of them addressed and studied the dynamics, and the other showed the complicated chaotic phenomenon [24–26]. Furthermore, there are some literatures focused on the memristive oscillators, chaotic attractors [12, 24, 31], and application in synaptic [3], neuron networks [4–7, 10, 11, 24, 27, 28], and oscillation phenomenon [14, 16, 17], and so on.

Furthermore, it has found that most existing studies focus on oscillatory phenomena generated by memristive systems, such as chaotic attractors, period-doubling oscillations, spiking and bursting oscillations. However, there is a notable lack of literature exploring and analyzing the energy exchange between these components, as well as the resulting oscillatory behaviors and outcomes arising from such interactions. It is well known that the unit of a memristor, like that of a resistor, is the ohm (Ω). In general circuits, the energy exchange between resistors and energy storage elements can induce nonlinear behaviors such as step functions, damping phenomena, both of which stem from the energy exchange between resistors and capacitors/inductors. So, when a memristor (though physical implementations are rare, several classic mathematical models exist) exchanges energy with energy storage elements, will similar behaviors emerge? In this paper, to advance the theoretical completeness of memristive systems and take the classical HP memristor model as an example, four source-free circuit topologies integrating memristors with energy-storage elements are investigated deeply. They are categorized into two types: $R_M C/R_M L$ circuits and series/parallel $R_M LC$ circuits. Firstly, through mathematical modeling, the four circuits are all found to be governed by transcendental equations. Simulation results demonstrate memristor states R_0, kR_d and energy-storage elements collectively regulate response characteristics, damped oscillatory and decay behavior. The active power and apparent power curves reveal distinct energy exchange behaviors between components, differing fundamentally from conventional RL, RC, and RLC circuits. These findings demonstrate that due to the presence of memristors, such circuits cannot be employed for step response generation, but are exclusively applicable for energy memorization and dissipation. Secondly, two types of four-component source-free circuits are configured and the following conclusion are demonstrated: (1) energy-storage elements provide energy (i.e., ϕ and q): to the system, while memristors exhibit hysteretic behavior, collectively and fundamentally co-modulating oscillation modes and attractor phenomenon; (2) The dual characteristics of memristors—memory capability and energy dissipation—endow them with the potential to break the von Neumann bottleneck, making them essential candidates for implementing next-generation neural networks and AI systems. Finally, the application circuits comprising four fundamental components was configured and explored. The study reveals that even within the same circuit, varying memristor placements can lead to distinct topological configurations and divergent nonlinear output behaviors. This phenomenon further validates the unique

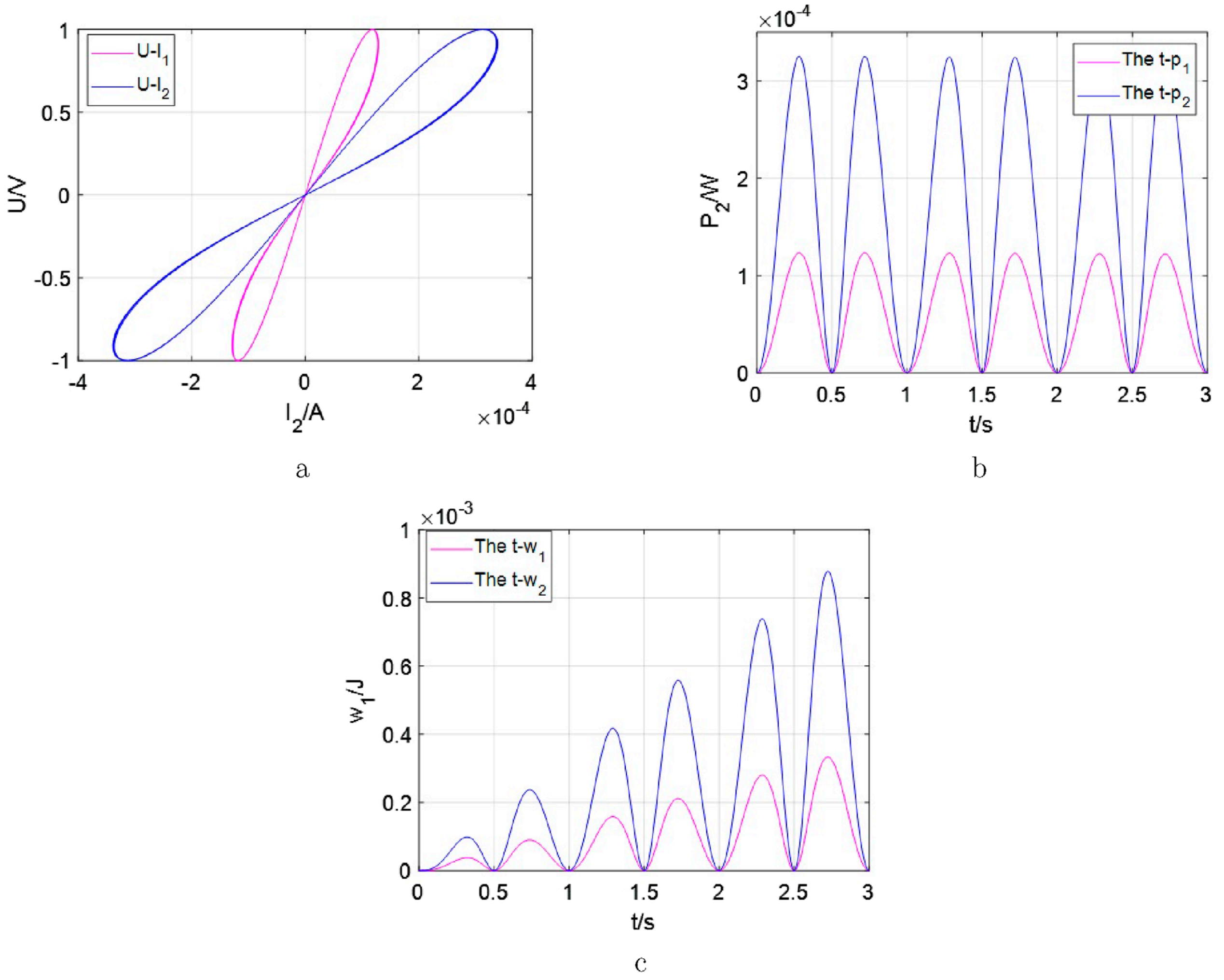


FIGURE 1
Several important relationships between different variants for the memristors, $kR_{d1} = 10^6$, $R_{01} = 16k$ in R_{M1} and $R_{02} = 6k$, $kR_{d1} = 10^6$ in R_{M2}, $dq(t)/dt = i(t) = sin(2t)$. **(a)** the curves of $v - i$ phase. **(b)** $p(t)$ by the memristor. **(c)** $w(t)$ by the memristor.

characteristics of memristors as an emerging field. These findings establish a solid theoretical and experimental foundation for future exploration and development of memristive systems, including next-generation neural networks, artificial intelligence applications, and aerospace technologies. Moreover, once the fundamental rules are improved, more and more foundations could be refined and continual applications in the theories and overall design process, such as nonlinear circuits, the avionics for unmanned aerial vehicle systems, as we shall see.

The remainder of this paper is organized as follows: In Sec II, the information on HP memristive system and two types of general source-free circuits are presented. In Section III, both source-free circuits are introduced, that is, R_MC and R_ML circuits. Then, their mathematical models, novel time constant, the response curves, the trajectories of the power dissipated and energy absorbed are performed, respectively. In Section IV, both series and parallel source-free R_MLC circuits are analyzed. In Section V, the application circuits with four components are provided and demonstrate the influence of energy storage elements or memristors on the frequency and oscillatory behaviors. Finally, the paper is summarized in Section VI.

Notably, all the curves in this paper are tested by the software MATLAB R2018a Version, which is a programming and numeric computing platform used by millions of engineers and scientists to analyze data, develop algorithms, and create models.

2 HP memristor and gerenal source-free circuits

As both one fundamental 2-port electric component and the classical model, HP-memristor (R_M) has been proposed and manufactured as the charge-controlled memristor [2–4, 20–22]. Its model could be given as follows

$$R_M = \frac{d\varphi}{dq} = R_{off} - \frac{\mu_v R_{off}}{D^2} R_{on} \cdot q(t) \tag{1}$$

where, there are two regions: one region with a high dopant concentration with low resistance R_{on}, the other region has a

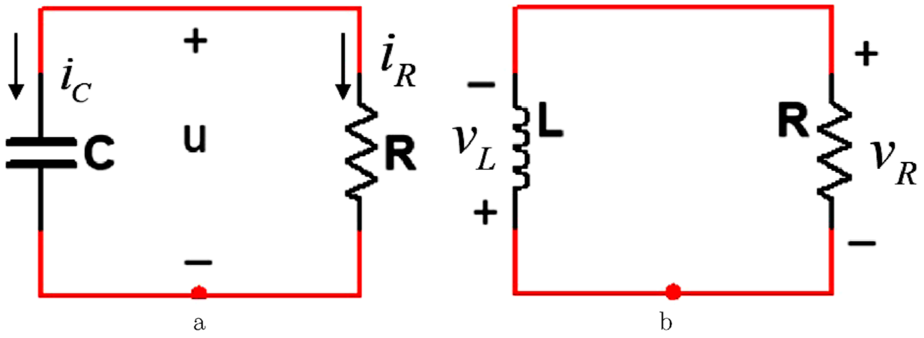


FIGURE 2
Two types of the classical source-free circuits. **(a)** *RC* circuit. **(b)** *RL* circuit.

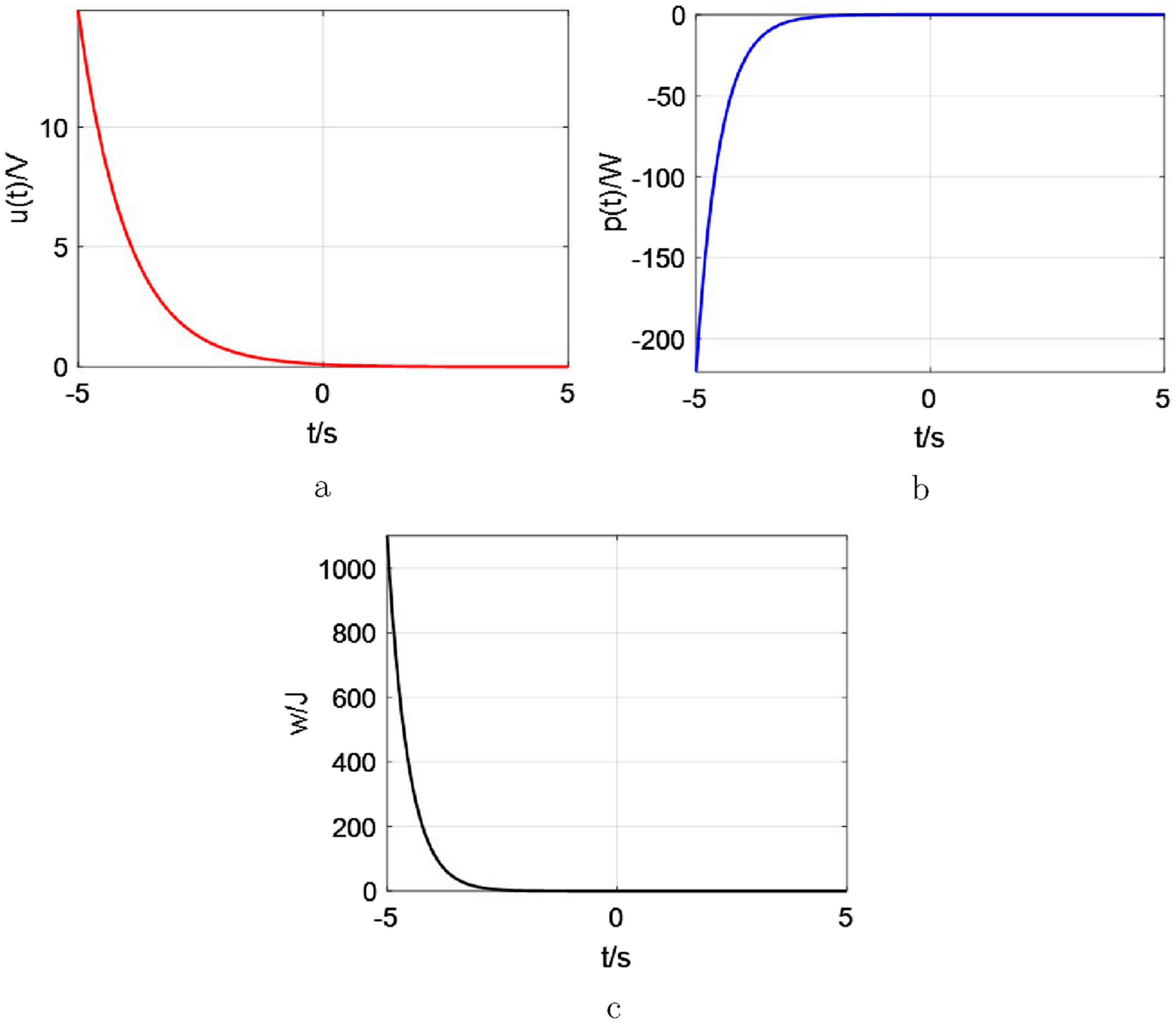


FIGURE 3
The natural response of the *RC* circuits, $u(t_0) = U_0 = 0.1V$, $R = 1$, $C = 1F$. **(a)** the curve of the voltage response. **(b)** $p(t)$ by the resistor. **(c)** $w(t)$ absorbed by the resistor.

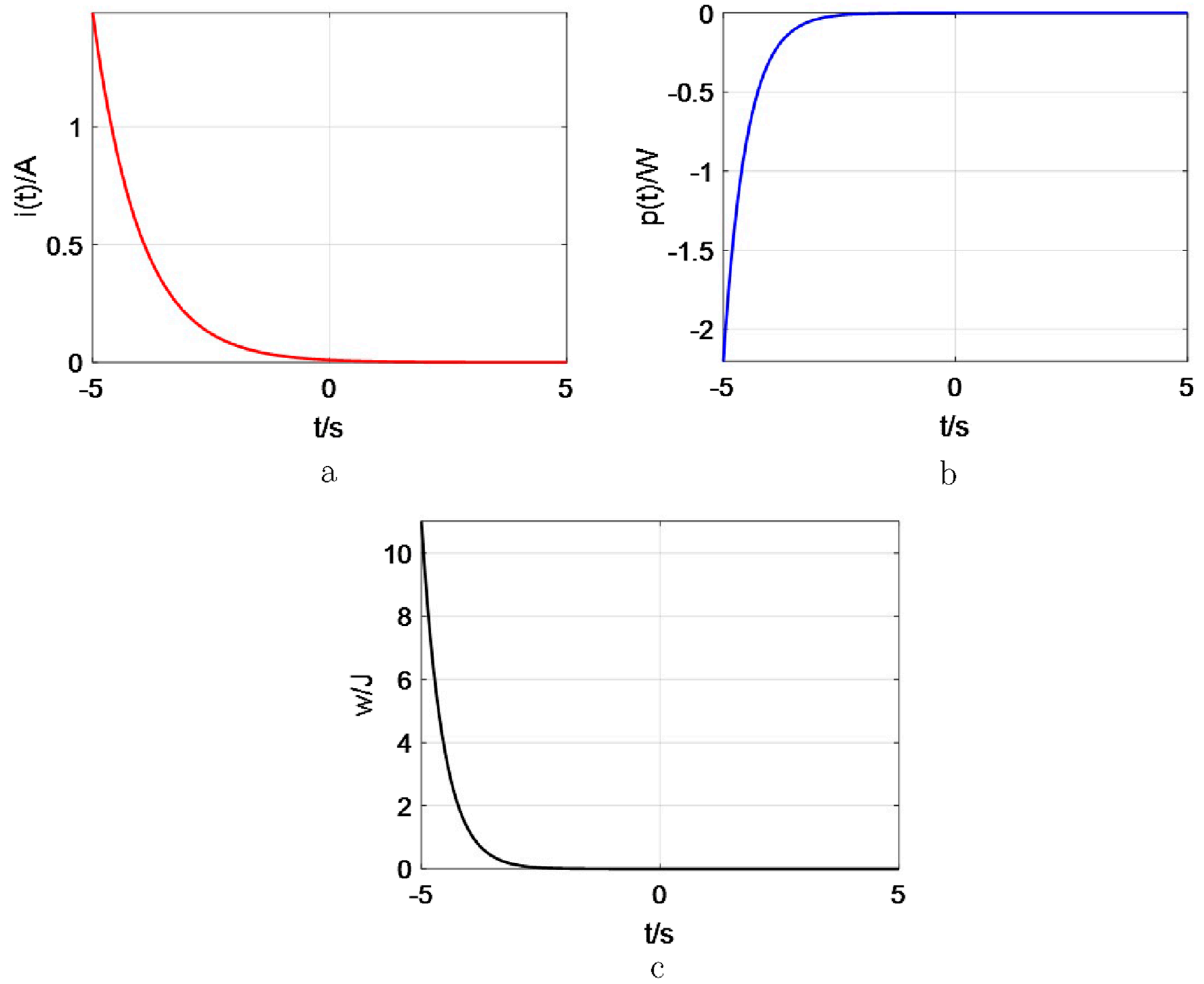


FIGURE 4
The natural response of the *RL* circuits, $i(t_0) = I_0 = 0.01A$, $R = 1$, $L = 1H$. **(a)** the curve of current response. **(b)** $p(t)$ by the resistor. **(c)** $w(t)$ by the resistor.

low concentration of dopant with a considerably higher resistance R_{off}. Also, the Equation 1 was named as the linear drift model due to the velocity of the width being linearly proportional to the current. Then, the variable $q(t)$ has been considered as the charge and means the integral of the current $i(t)$.

In order to study the universality of this class of memristive systems, it can be re-written as

$$\begin{cases} R_M = R_0 + kR_d \int idt = R_0 + kR_d \cdot q \\ R_M = u_M(t)/i_M(t) \end{cases} \tag{2}$$

where the variable $u_M(t)$ is the cross voltage, $i_M(t)$ as one function of current and has been defined as the rate of change of the state variable. Defining the parameters R_0, k and R_d jointly reflect the relationship between $u_M(t)$ and $i_M(t)$. Then, the parameter R_0 stands in the region which has a low concentration of dopant with a considerably higher resistance R_{off}. The parameter R_d is one region with a high dopant concentration and low resistance R_{on}. The parameter $k = -\mu V R_{off}/D^2$ is defined as a coefficient.

The important trajectory curves are depicted in Figure 1.

From Figure 1, it can be seen that these curves are the fingerprints, the dissipated power and energy absorbed in time-domain graphs for the single memristor. They are so complex but cannot be applied directly like the other general discrete elements. Due to the characteristics of the memristive system, its power exhibits a frequency doubling phenomenon. Thus, the related basic fundamentals should be examined as soon as possible via the n-order circuit model with the R_M and an energy storage element.

In circuit theory, for an ordinary circuit, there are two excitation methods. One method involves independent sources. The other utilizes the initial conditions of storage elements within the circuit, which are so-called source-free circuits. When energy is initially stored in capacitive or inductive elements, this stored energy drives current flow, which is gradually dissipated in resistors. The rate of dissipation can be calculated by Kirchhoff's laws. This way has been considered as a sufficient, powerful set of tools to analyze a large variety of electric circuits all the time. Now, this method could be utilized to analyze the following circuits, such as R_MC and R_ML circuits. The classical first-order source-free circuits are shown as Figure 2.

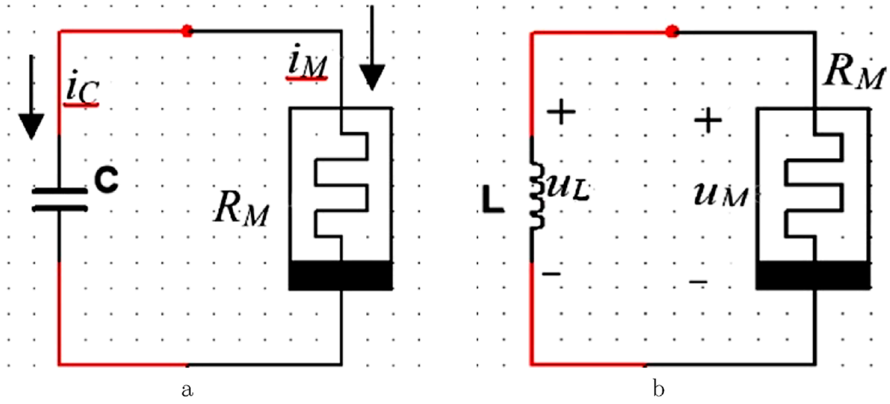


FIGURE 5
The source-free circuits with charge-controlled memristor. **(a)** R_MC circuit. **(b)** R_ML circuit.

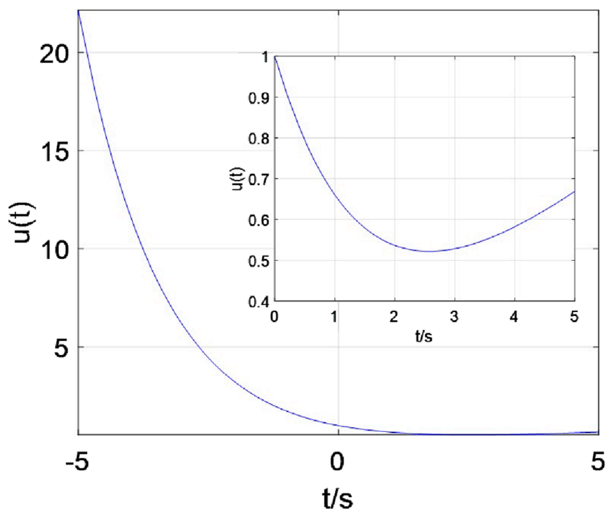


FIGURE 6
The natural response of the R_MC circuits, the initial value of $kR_d = 0.8 * 10^6$, $R_0 = 16k$, $C = 1mF$

Observing from Figure 2a, when the initial condition is $u(0) = U_0$, the voltage response of the *RC* circuit could be expressed by an exponential decay of the initial voltage. Also, this result is attributed to the initially stored energy and the circuit's intrinsic characteristics, rather than its external voltage or current sources. Similar to Figure 2b, it is shown that the natural response of the *RL* circuit is also an exponential decay of the initial current. Furthermore, the time constant for both *RC* and *RL* circuits have been defined as $\tau = RC$ and $\tau = L/R$. Subsequently, the natural response could be illustrated graphically in Figures 3, 4. It has been evidence that an exponential decay of the initial condition, dissipated power and the absorbed energy by the resistor for the *RC* circuit are also given by Equation 3, as the current responses for the *RL* circuit are shown by Equation 4.

For the source-free *RC* circuit, when the initial condition $u(t_0) = u(0) = U_0$, the results could be given as follows

$$\begin{cases} u(t) = U_0 e^{-t/\tau}, \quad \tau = RC \\ p(t) = u(t)\, i_R(t) = -\dfrac{U_0^2}{R} e^{-2t/\tau} \\ w(t) = \displaystyle\int_0^t p(t)\, dt = \frac{1}{2} C U_0^2 \left(1 - e^{-2t/\tau}\right) \end{cases} \tag{3}$$

For the source-free *RL* circuit, when the initial condition $i(t_0) = i(0) = I_0$, the results could be computed as follows

$$\begin{cases} i(t) = I_0 e^{-t/\tau}, \quad \tau = L/R \\ p(t) = u(t)\, i_R(t) = I_0^2 R e^{-2t/\tau} \\ w(t) = \displaystyle\int_0^t p(t)\, dt = \frac{1}{2} L I_0^2 \left(1 - e^{-2t/\tau}\right) \end{cases} \tag{4}$$

Observed from Figures 3, 4, when $t \to \infty$, $w_R(\infty) \to \frac{1}{2} C U_0^2$ for the *RC* circuit and $w_R(\infty) \to \frac{1}{2} L I_0^2$ for the *RL* circuit could be observed. They are the same as the energy initially stored in the capacitor ($w_c(0)$) element in Equation 3 and inductor ($w_L(0)$) element in Equation 4.

The above has already provided a complete description for the properties of classical first-order circuits. Next, we pose a question: when resistors are substituted with memristors (e.g., HP memristors), what kind of conclusions could be obtained? For this purpose, the followingR_MC and R_ML circuits will be configured and analyzed in the next section.

3 The source-free R_MC and R_ML circuits

3.1 The source-free R_MC circuits

The produced source-free R_MC circuit could be drawn as Figure 5a. Applying Kirchhoff's Laws ($i = i_C = i_M$) yields

$$(R_0 + kR_d q)\, C \frac{du_c}{dt} + u_c = 0 \tag{5}$$

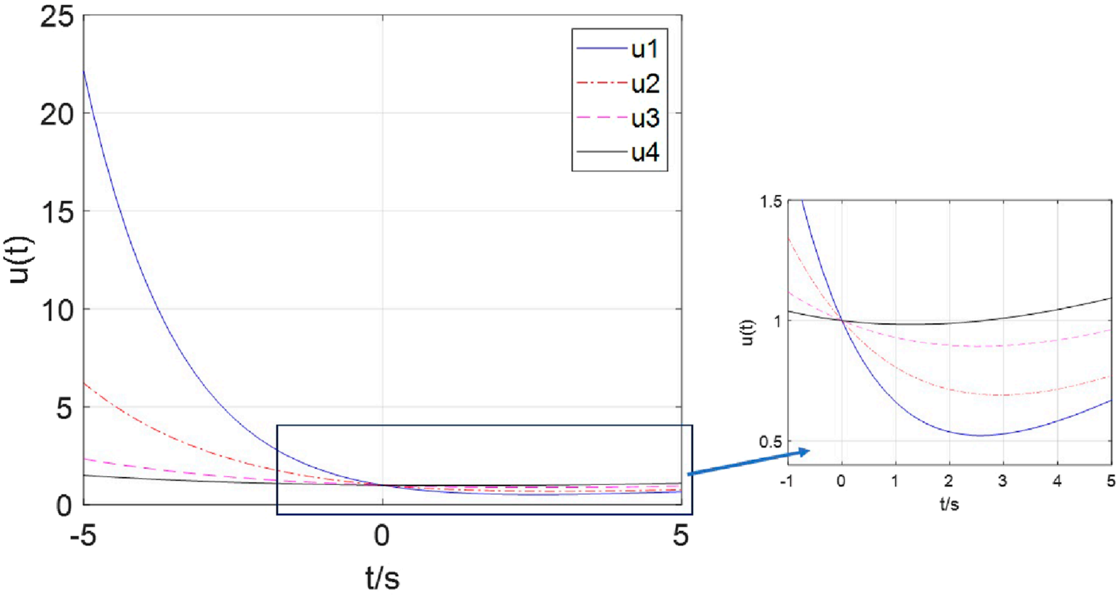


FIGURE 7
The curve of the voltage response with $R_{01} = 16k$ in blue, $R_{02} = 26k$ in red, $R_{03} = 46k$ in pink, $R_{04} = 66k$ in black.

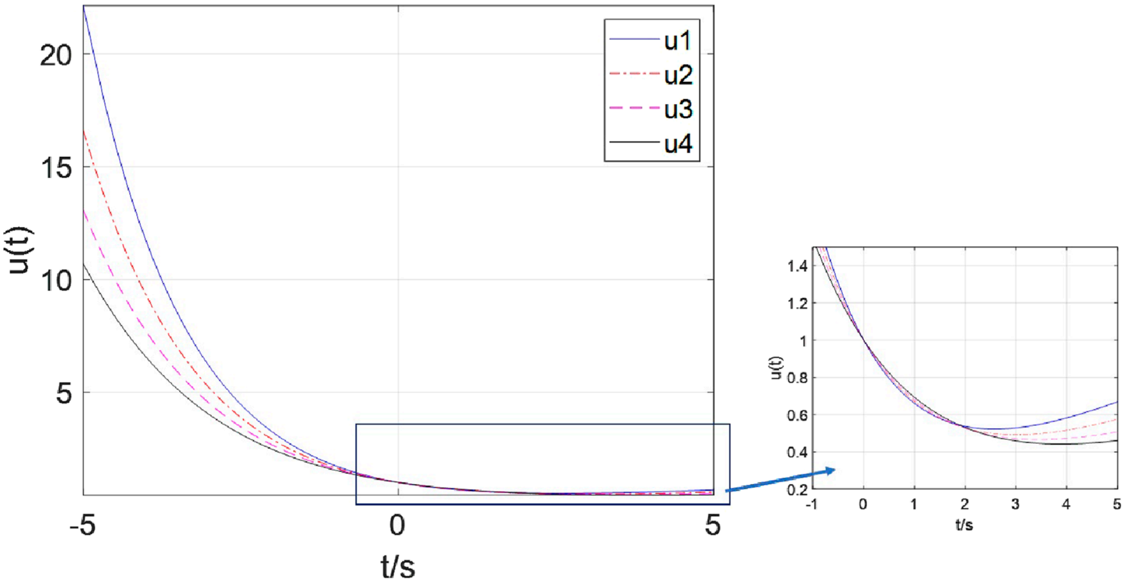


FIGURE 8
The curve of the voltage response with $C = 1mF$ in blue, $C = 1.1mF$ in red, $C = 1.2mF$ in pink, $C = 1.3mF$ in black.

where, the variable $u(t) = u_c(t)$ stands for the voltage of the capacitor. Notably that $dq(t)/dt = i(t) = i_M(t) = i_C(t)$ is the intrinsic variable for Figure 5a. Let $\tau_0 = R_0C$ and $b = kR_dC/R_0$, the terms could be depicted as

$$\ln u + b \cdot u = -\frac{t}{\tau_0} \tag{6}$$

Obviously, Equation 6 a transcendental equation whose solution can only be computed using approximations and cannot be obtained exactly. The natural response curve of the Equation 5 could be illustrated graphically in Figure 6.

From Figure 6, this response curve is fundamentally distinct from that of an *RC* circuit (characterized by a single exponential curve), where the energy stored in the capacitor is entirely dissipated by the resistor. It is more complex and constitutes both exponential and non-exponential functional components. Crucially, although the memristor's unit is also the ohm (Ω), its model reveals that it consists of both linear and nonlinear resistive components [1,2].

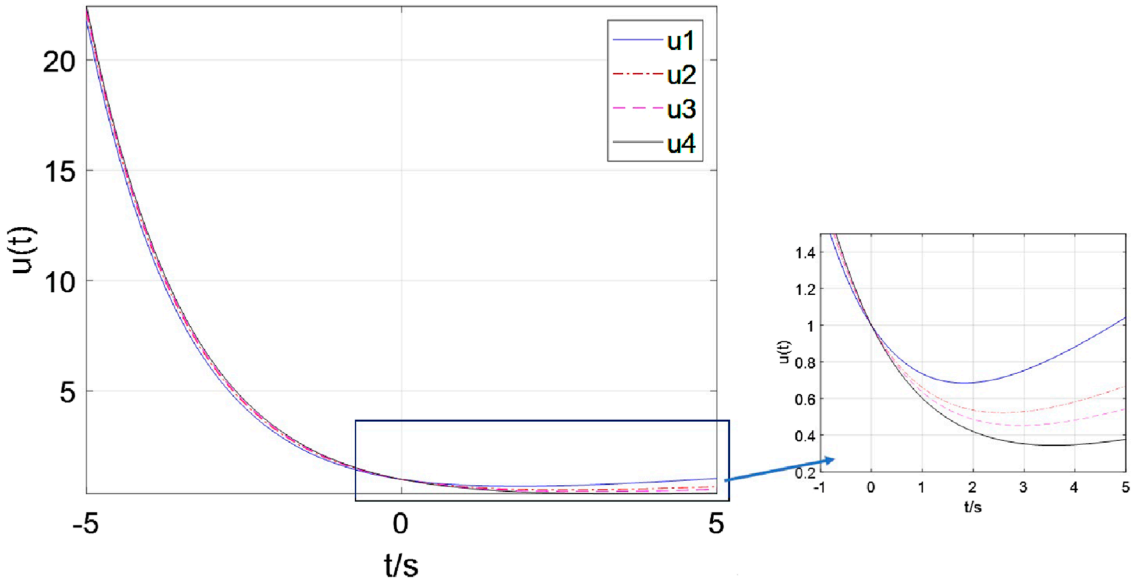


FIGURE 9
The curve of the voltage response with $kR_{d1} = 0.5 * 10^4$ in blue, $kR_{d2} = 0.8 * 10^4$ in red, $kR_{d3} = 10^4$ in pink, $kR_{d4} = 1.5 * 10^4$ in black.

Here, the linear resistive component exhibits the conventional ”energy-dissipation” characteristic of resistors, as manifested by the exponential segment of the curve. However, the energy stored in the capacitor is not fully consumed the remaining portion is ”memorized” by the nonlinear component, which corresponds to the non-exponential segment of the curve. Then, according to the definition of the time constant, setting $\tau_0 = R_0C$ and calculators b for this R_MC circuit. When the circuit is excited, C provides the stored energy to the R_M, the R_M works for both memorizing the information and energy dissipation profile, immediately. Memory speed depends on this new time constant (τ_0).

There are three variables related to the decay of the voltage response $u_C(t)$, which are R_d, R_0 and C. Next, the decay behavior would be discussed when only one variable is changed and the other ones are fixed.

(1) when $kR_d = 0.8 * 10^4$ and $C = 1mF$, changing the variable R_0, the response $u_C(t)$ are illustrated in Figure 7.

Observe from Figure 7, When $t < 0$, the blue curve resides innermost while the black curve lies outermost; when $t > 0$, the blue curve shifts to the bottom position and the black curve to the top. This demonstrates that as the value of R_{01} increases, the curves become progressively flatter, indicating slower rates for both energy dissipation (“consumption”) and memory retention (“memorization”). Then, the following conclusion could be drawn:

i. A smaller R_0 results in a larger τ_0 with faster decay dynamics.
ii. A smaller R_0 value leads to a significant increase in the proportion of the exponential segment of the curve. The more pronounced the “energy dissipation” component becomes, the more enhanced the ”memory” effect appears.
iii. A certain energy exists to memorize information for the memristor. Therefore, the voltage $u_C(t)$ cannot decay to 0 at the $t = 0$.

(2) when $kR_d = 0.8 * 10^4$ and $R_0 = 16k$, changing the capacitance C, the response $u_C(t)$ are illustrated in Figure 8.

From Figure 8, when $t < 0$, the blue curve resides innermost while the black curve lies outermost; However, when $t > 0$, the blue curve shifts to the top position and the black curve to the bottom. This demonstrates that as the value of C increases, the curves exhibits a significantly steepened profile, indicating faster rates for both energy dissipation (“consumption”) and memory retention (“memorization”). Then, similar results are still observed.

i. A smaller C results in a larger τ_0 with faster decay dynamics.
ii. A smaller C value leads to a significant increase in the proportion of the exponential segment of the curve. The more pronounced the “energy dissipation” component becomes, the more enhanced the “memory” effect appears.
iii. A certain voltage is required to memorize information for the memristor. Therefore, the voltage $u_C(t)$ cannot decay to 0 at the $t = 0$. However, the different capacitor C could provide the different storage voltage to the memory.

(3) when $C = 1mF$ and $R_0 = 16k$, changing the variable R_d, the response $u_C(t)$ are illustrated in Figure 9.

Observed from Figure 9, when $t < 0$, the black curve resides innermost while the blue curve lies outermost; when $t > 0$, the blue curve shifts to the top position and the black curve to the bottom. This demonstrates that as the value of kR_d increases, the curves exhibits a significantly steepened profile, indicating faster rates for both energy dissipation (“consumption”) and memory retention (“memorization”). Due to the minimal variation in kR_d, the distinction between the curves is not particularly pronounced. Then, different results can be obtained:

i. A larger kR_d results in a larger τ_0 with faster decay dynamics.
ii. A larger kR_d value leads to a significant increase in the proportion of the exponential segment of the curve. The more

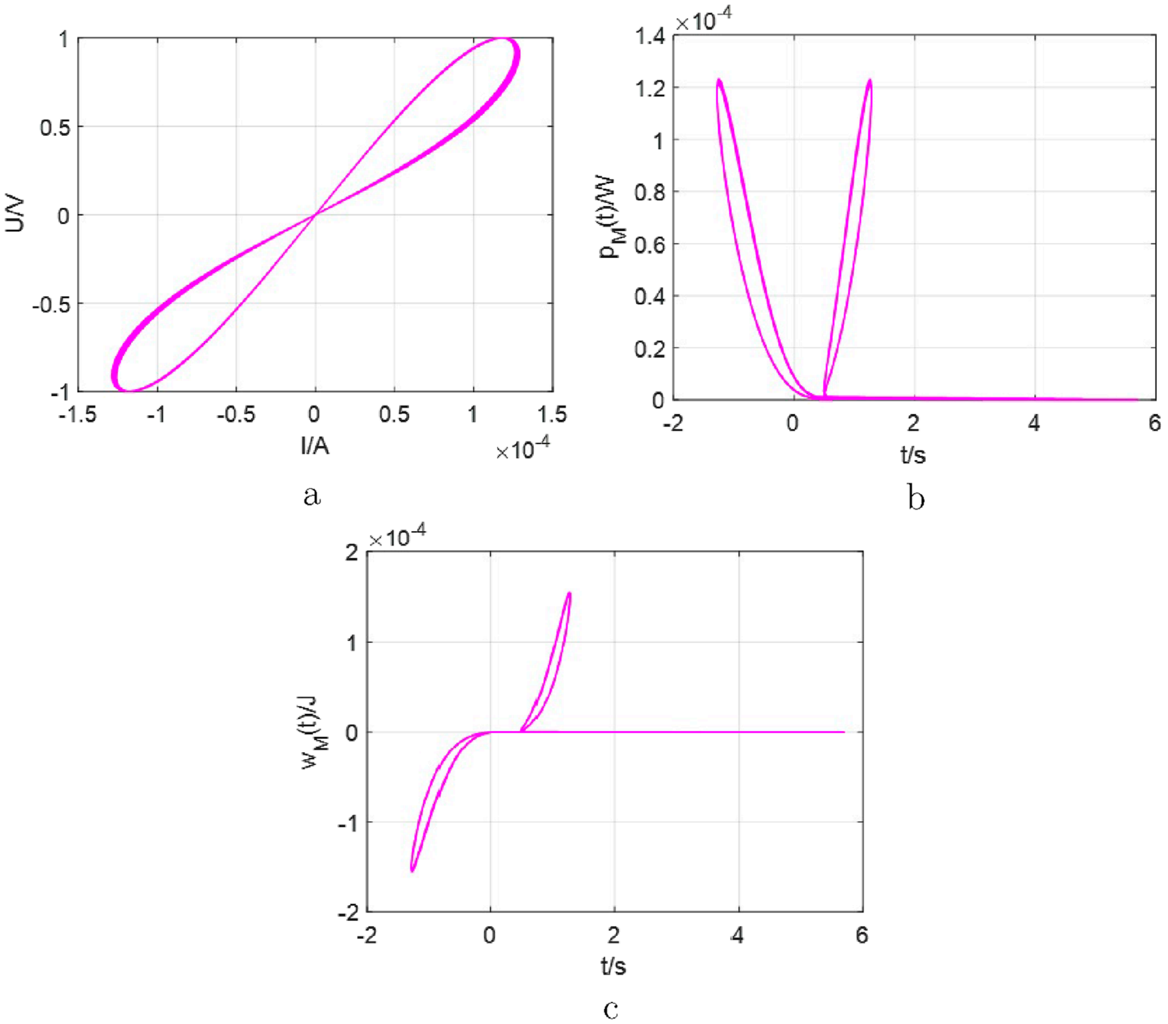


FIGURE 10
The curves for the $R_M C$ circuit with the parameters $kR_d = 0.8 * 10^6$, $R_0 = 16k$, $C = 1mF$. **(a)** the fingerprint characterizes of $v - i$ for R_M). **(b)** the dissipated power by the memristor up to time t/s). **(c)** the energy ($w(t)$)) absorbed by the memristor up to time t/s).

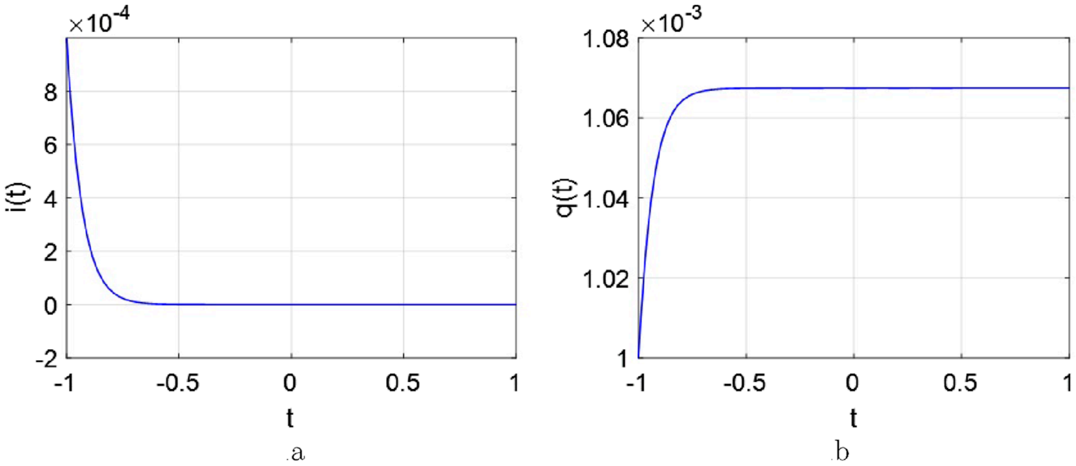
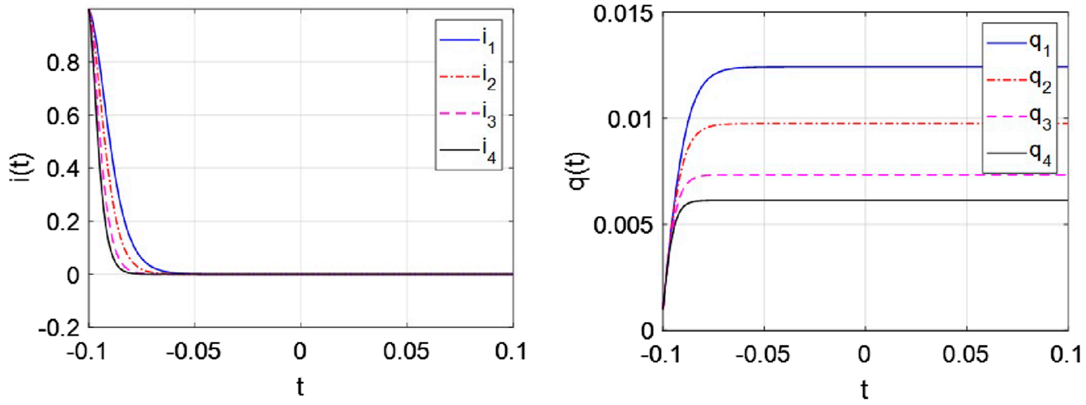
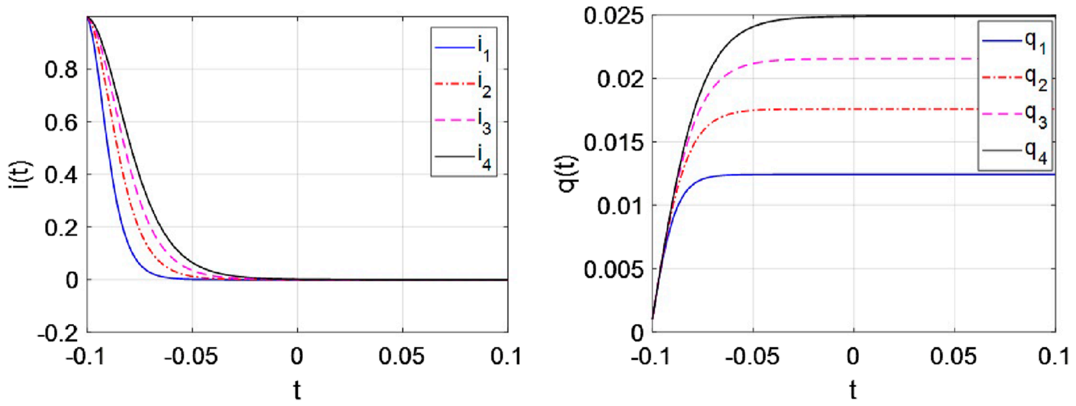


FIGURE 11
The natural response of the $R_M L$ circuits, the initial value of $kR_d = 0.8 * 10^6$, $R_0 = 16k$, $L = 0.1H$. **(a)** $i(t) - t$). **(b)** $q(t) - t$).



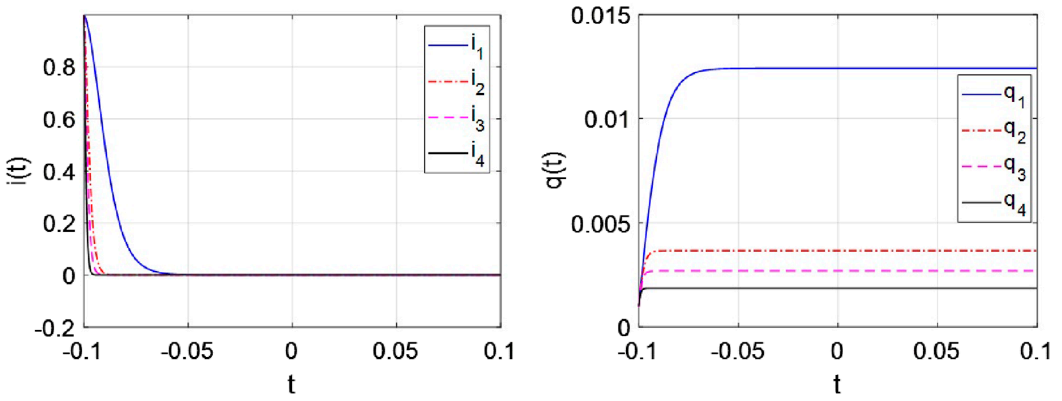
a b

FIGURE 12
The curves of the current and charge response with $R_{01} = 16k$ in blue, $R_{02} = 26k$ in red, $R_{03} = 46k$ in pink, $R_{04} = 66k$ in black. **(a)** $i(t) - t$. **(b)** $q(t) - t$.



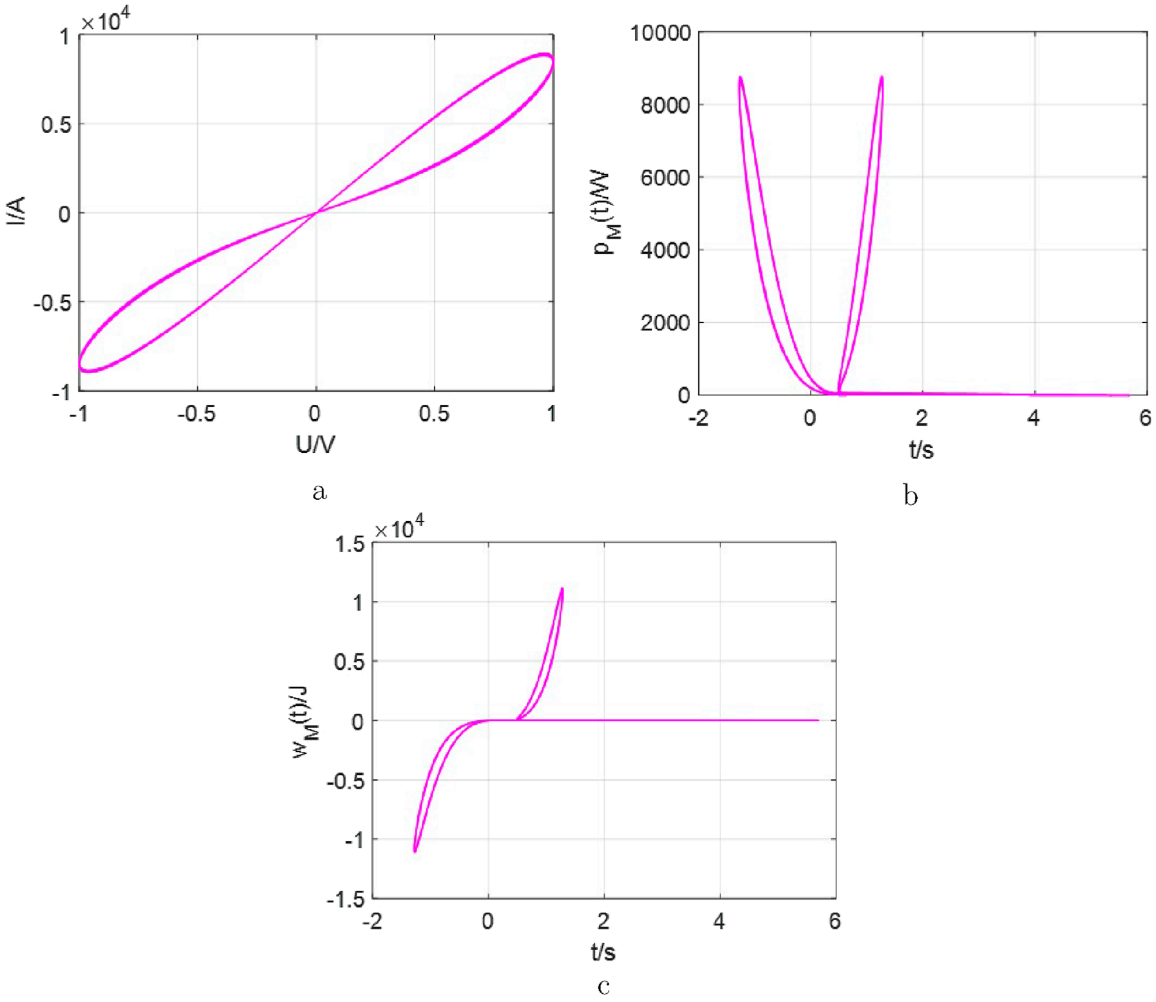
a b

FIGURE 13
The curve of the current and charge response with $L_1 = 0.1H$ in blue, $L_2 = 0.2H$ in red, $L_3 = 0.3H$ in pink, $L_4 = 0.4H$ in black. **(a)** $i(t) - t$). **(b)** $q(t) - t$.



a b

FIGURE 14
The curves for the R_MC circuit with the parameters $kR_d = 0.8 * 10^6$, $R_0 = 16k$, $C = 1mF$. **(a)** $i(t) - t$). **(b)** $q(t) - t$).

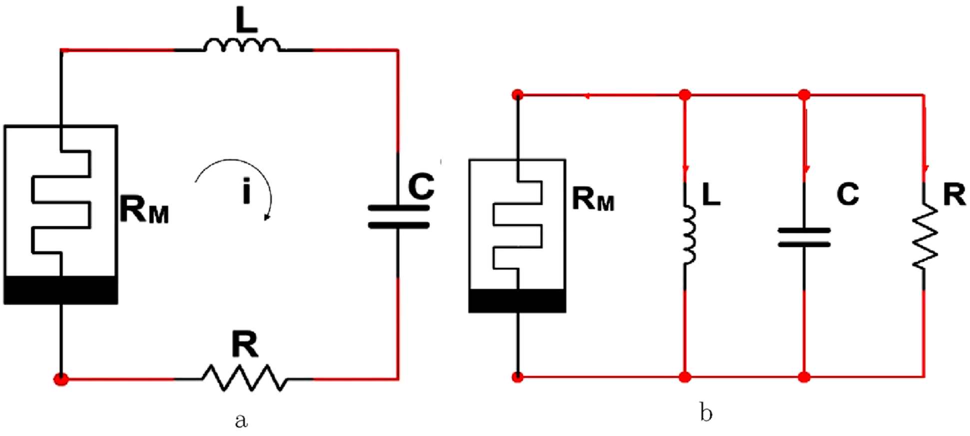


a

b

c

FIGURE 15
The curves for the $R_M L$ circuit. **(a)** the fingerprint characterizes of $v-i$ for R_M). **(b)** the dissipated power by the memristor up to time t/s). **(c)** the energy ($w(t)$) absorbed by the memristor up to time t/s).



a

b

FIGURE 16
The source-free $R_M LC$ circuits with charge-controlled memristors. **(a)** the series circuit. **(b)** the parallel circuit.

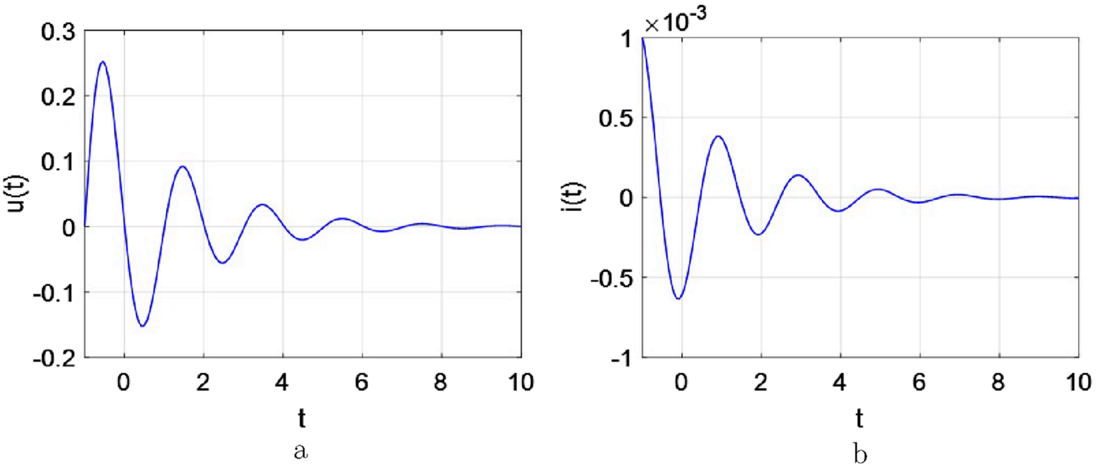


FIGURE 17
The response curves of the source-free R_MLC series circuits, $I_0 = 0.001A$, $U_0 = 0.001V$, $kR_d = 1 * 106$, $R_0 = 16k$, $R = 1$, $L = 0.1H$, $C = 1mF$. **(a)** $u(t) - t$. **(b)** $i(t) - t$.

pronounced the "energy dissipation" component becomes, the more enhanced the "memory" effect appears.

iii. A certain energy exists to memorize information for the memristor. Then, the voltage $u_C(t)$ cannot decay to 0 at the $t = 0$.

It should be noted that when applying both RC and RL circuits, they could be treated as the step functions to configure plenty of circuit-networks. However, both R_MC and R_ML circuits are not the step functions. They do not focus on storing and consuming energy, but on memorizing pieces of information. Secondly, the speed of memorizing information is associated with the determined new time constant (τ_0). The larger τ_0 could lead to the faster the decay as well as speed of memorizing pieces of information. Thirdly, the memristor (R_M) satisfies dual properties: memory (R_0) and energy consumption (kR_d).

Finally, the dissipated power and the absorbed energy by the memristor for the R_MC circuit are depicted in Figure 10.

Between Figure 1 and Figure 10a, the fingerprint characteristics have been presented. Observed from Figure 10b, some information could be memorized by R_MC circuit. Furthermore, dissipative power and absorbed energy are utilized for information storage. This reaffirms the memristor's fundamental divergence from resistors even though they shared dimensional homogeneity and common unit of ohms (Ω). Moreover, their distinct time constants (τ_0) manifests the memory functionality, not the energy dissipation profile.

3.2 The source-free R_ML circuits

The inductor (L) is the other type of energy storage element. In this subsection, the R_ML circuit would be configured and discussed. Similar to analyzing the R_MC circuit, consider one memristor circuit as shown in Figure 5b.

Applying Kirchhoff's Laws $u_L + u_M = 0$, $i = i_L = i_M$ and Figure 4b, yields

$$L\frac{di}{dt} + R_0 i + kR_d q \cdot i = 0 \tag{7}$$

where the variable $i(t)$ stands for the current through the inductor. Hereby, $dq(t)/dt = i(t) = i_M(t) = i_L(t)$ is the determined relationship. Also, let $\tau_0 = L/R_0$. Obviously, this is also a higher-order transcendental equation. Its natural response curve of the Equation 7 could be illustrated graphically in Figure 11.

As shown in Figure 11, the depicted response curve bears similarities to the general RL circuit in Figure 2b but also exhibits significant differences. Model (7) reflects more complex and faster nonlinear behavior of higher-order functions. Then, according to the definition of this new time constant ($\tau_0 = L/R_0$), when this circuit is excited, R_M immediately begins to store information. At the same time, the inductor (L) is busy converting energy to the memristor. Notably, the speed of memorization depends on τ_0.

The decay of the current response $i(t)$ is influenced by three variables: R_d, R_0 and L. Next, the decay behavior would be discussed when only one variable is varied and the other remains fixed.

(1) when $kR_d = 0.8 * 10^4$ and $L = 0.1H$, changing the variable R_0, the response $u_C(t)$ are illustrated in Figure 12.

Observe from Figure 12, the following conclusion could be obtained:

i. A large R_0 brings a small τ_0, and decays the fast.
ii. A certain current is required, when a memristor memorizes information. Therefore, the current $i(t)$ cannot decay to 0 even at $t = 0$.
iii. The memory characteristics could be occurred by R_0.

(2) when $kR_d = 0.8 * 10^4$ and $R_0 = 16k$, changing the inductance L, the response $i(t)$ are illustrated in Figure 13.

From Figure 13, the similar results could be got:

i. A large inductive L brings a large τ_0, and decays the slow.

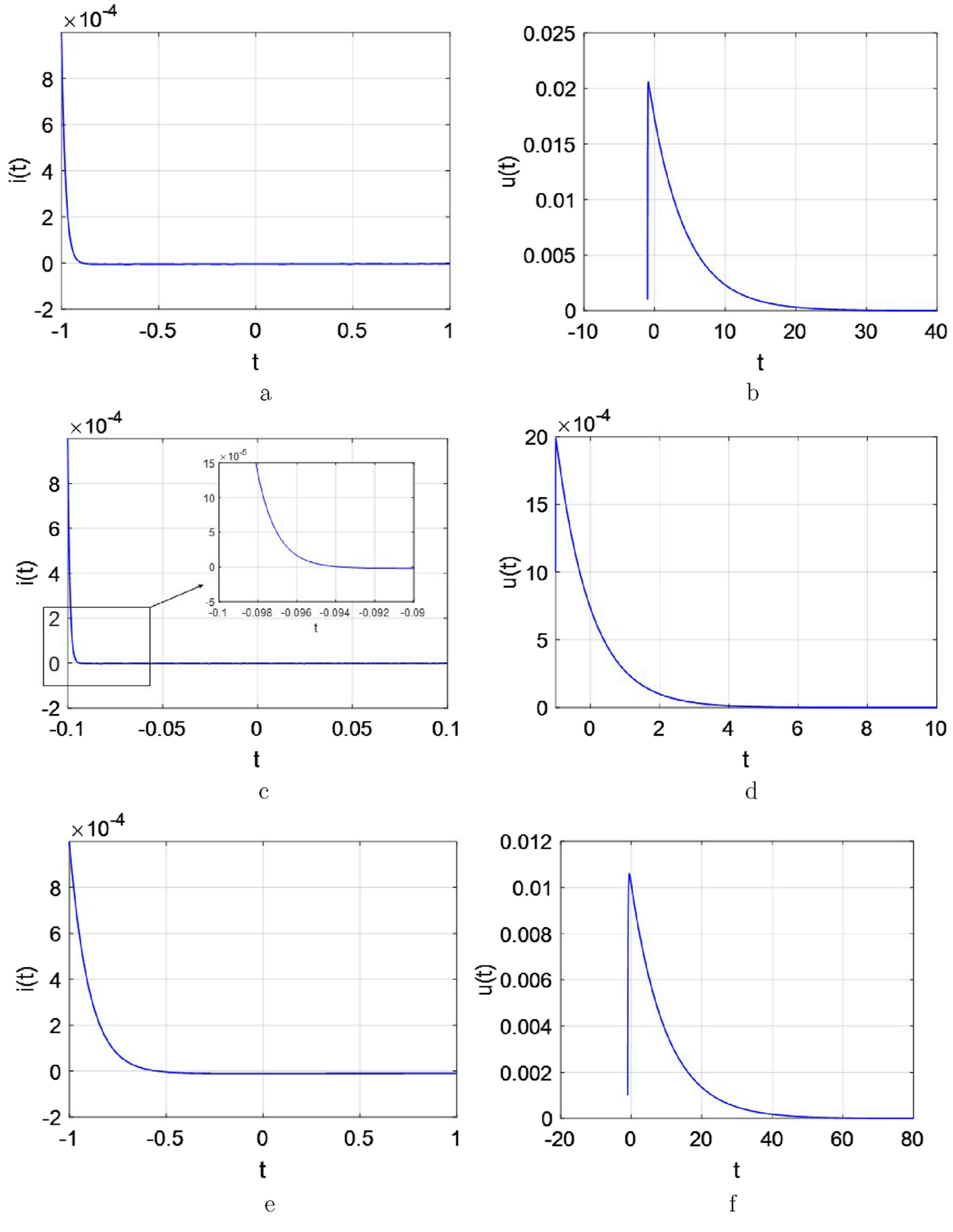


FIGURE 18
The response of current and voltage curves for the $R_M LC$ series circuit. **(a)** $i(t)-t$ with $R=5k$, $L=0.1H$, and $C=1mF$. **(b)** $u(t)-t$ with $R=5k$, $L=0.1H$, and $C=1mF$. **(c)** $i(t)-t$ with $R=1k$, $L=1mH$, and $C=1mF$. **(d)** $u(t)-t$ with $R=1k$, $L=1mH$, and $C=1mF$. **(e)** $i(t)-t$ with $R=1k$, $L=0.1H$, and $C=10mF$. **(f)** $u(t)-t$ with $R=1k$, $L=0.1H$, and $C=10mF$.

ii. When a memristor is utilized for information storage, the current (i) and charge (q) are altered. Furthermore, the role of the energy storage element L can be demonstrated.

(3) when $L=1H$ and $R_0=16k$, changing the variable R_d, the response $u_C(t)$ are illustrated in Figure 14.

Observed from Figure 14, the following results could be given as:

i. A large kR_d also leads to decay the fast similar to Figure 12.
ii. The energy consumption characteristics exist and are presented by kR_d.

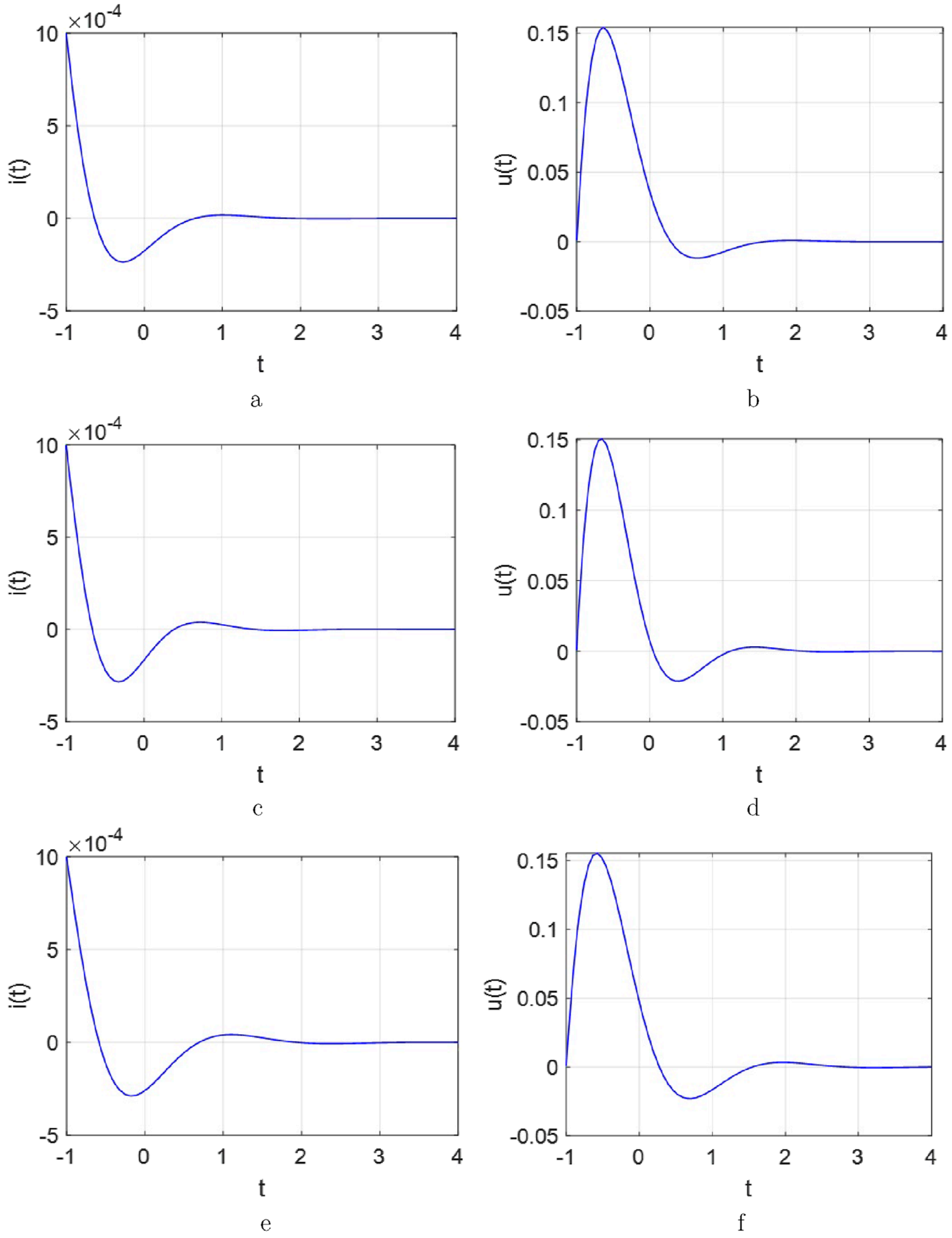


FIGURE 19
The response of current and voltage curves for the $R_M LC$ series circuit. **(a)** $i(t)-t$ with $R = 400$, $L = 0.1H$, and $C = 1mF$. **(b)** $u(t)-t$ with $R = 400$, $L = 0.1H$, and $C = 1mF$. **(c)** $i(t)-t$ with $R = 300$, $L = 0.08H$, and $C = 1mF$. **(d)** $u(t)-t$ with $R = 300$, $L = 0.08H$, and $C = 1mF$. **(e)** $i(t)-t$ with $R = 300$, $L = 0.1H$, and $C = 1.2mF$. **(f)** $u(t)-t$ with $R = 300$, $L = 0.1H$, and $C = 1.2mF$.

Finally, Figure 15 presents the dissipated power of the $R_M L$ circuit and the energy absorbed by the memristor.

Similar to the $R_M C$ circuit, when designing a source-free circuit using an inductor and a memristor, its behavior cannot be treated as a step function, too. Because its primary purpose is to store information. The smaller the new time constant leads to the faster the decay. Furthermore, a higher-order transcendental equation has been obtained and more complex nonlinear behaviors have been captured. There are three crucial points in a source-free $R_M L$ circuit to determine the calculator $i_L(t)$, that is, the initial current I_0, new time constant τ_0, and the integral of the charge $q(t)$ in R_M. Thirdly, the memristor is presented with dual characteristics: the memory behavior (represented by R_0) and the energy consumption characteristics (described by kR_d).

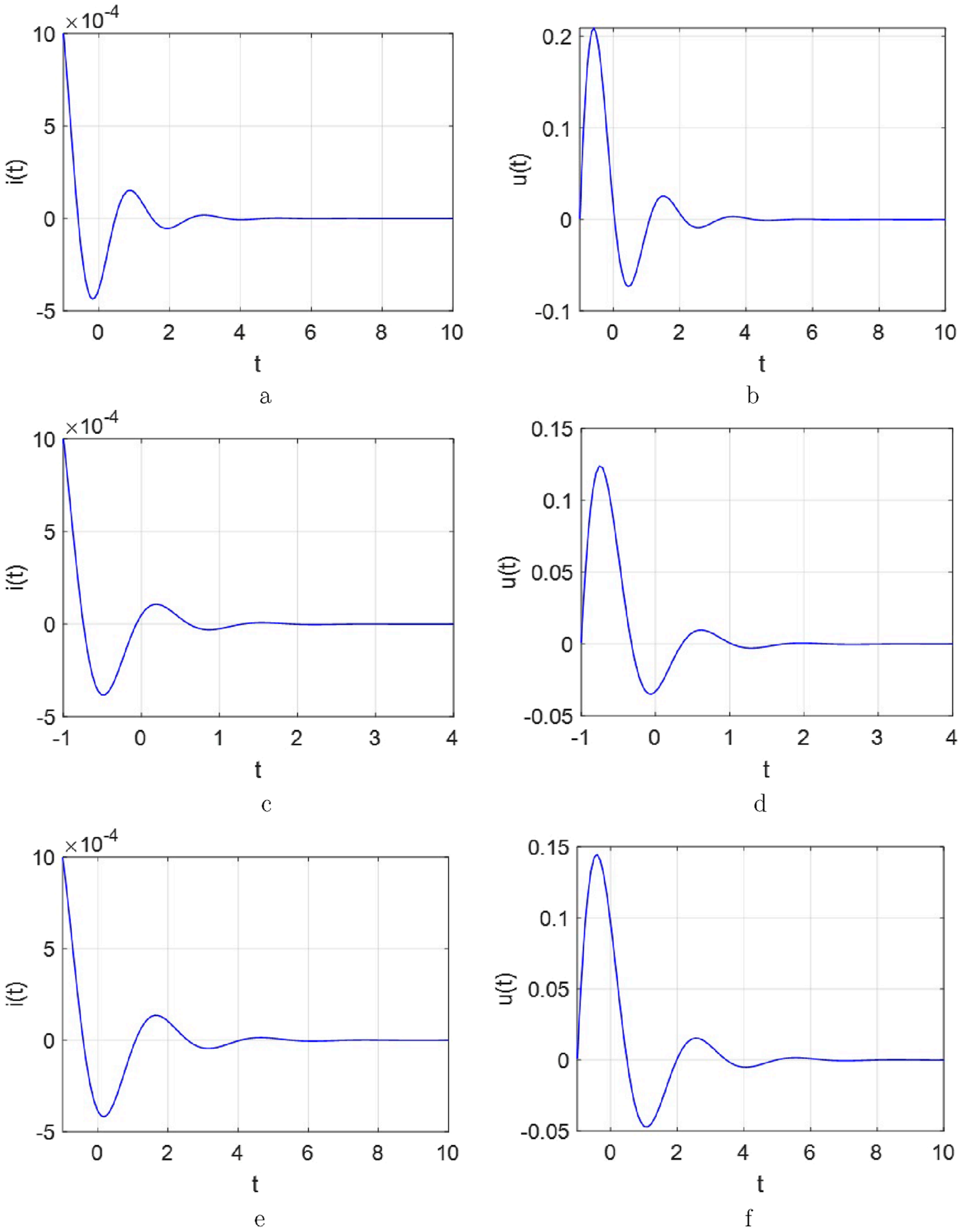
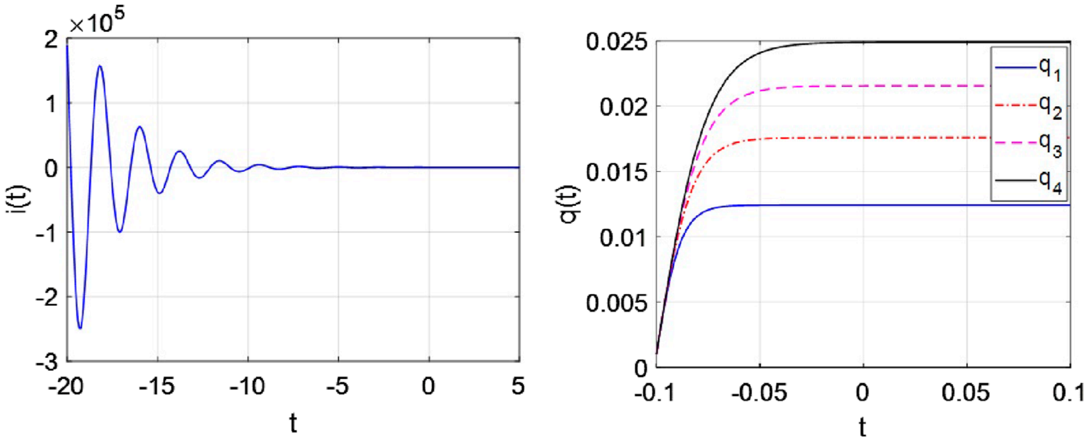


FIGURE 20
The response of current and voltage curves for the $R_M LC$ series circuit. **(a)** $i(t)-t$ with $R = 350$, $L = 0.12H$, and $C = 1mF$. **(b)** $u(t)-t$ with $R = 35$, $L = 0.12H$, and $C = 1mF$. **(c)** $i(t)-t$ with $R = 1k$, $L = 1H$, and $C = 1mF$. **(d)** $u(t)-t$ with $R = 1k$, $L = 1H$, and $C = 1mF$. **(e)** $i(t)-t$ with $R = 1k$, $L = 0.12H$, and $C = 120uF$. **(f)** $u(t)-t$ with $R = 1k$, $L = 0.12H$, and $C = 120uF$.

4 The $R_M LC$ circuits

Importantly, building memristive circuits is inseparable from energy storage components, similarly, the study of source-free circuits cannot proceed without them. In the aforementioned analysis and discussion of the natural response of the source-free circuits, two transcendental equations incorporating memristor models have been established. Additionally, the new time constant for the both circuits has been redefined. In this section, similar to the analysis of RLC circuits, $R_M LC$ circuits could also be connected in two configurations: series and parallel circuits, see Figure 16.



a

b

FIGURE 21
The response curves of the source-free $R_M LC$ parallel circuits, $I_0 = 1A$, $U_0 = 10V$, $kR_d = 1 * 10^6$, $R_0 = 16k$, $R = 1k$, $L = 0.12H$, $C = 1mF$. **(a)** $i(t) - t$. **(b)** $u(t) - t$.

4.1 A.Series circuit

Applying Kirchhoff's Laws ($i = i_R = i_C = i_L = i_M$) and Figure 16a, according to the description of Equation 2, the following Equation 8 could be built as following

$$\begin{cases} Ri + (R_0 + kR_d q)\, i + L\dfrac{di}{dt} + u = 0 \\ i = C\dfrac{du}{dt} \\ q = \displaystyle\int idt \\ i(0) = I_0, \quad u(0) = U_0 \end{cases} \tag{8}$$

where the variables $i(t) = i_L(t)$, $u(t) = u_c(t)$ stand for the current flowing through the inductor and voltage across the capacitor. From the preceding analysis, when energy storage elements are integrated with memristors in a circuit, their response models can be established as transcendental and higher-order equations. In Figure 16a, let $q = Ae^{(st)}$ and $i = Ase^{(st)}$, where s and t critical variables that must be discussed and determined. Additionally, the necessary derivatives can be derived as

$$s^3 + \left(\frac{RC + R_0 C + kR_d q}{L} + \frac{AkR_d}{L}e^{st}\right)s^2 + \frac{1}{LC}s = 0 \tag{9}$$

There is no doubt that Equation 9 is still a high-order transcendental equation. Thus, the natural response curve could be depicted graphically in Figure 17.

Comparison with a conventional *RLC* (second-order) series circuit, the solution of system (9) also could exhibit damping characteristics and generate the type of resonance phenomenon. By varying the values of R_M, L or C, it discusses whether the system could observe the three damping conditions (overdamped, critically damped, and underdamped) analogous to traditional *RLC* circuits. These three cases might be illustrated and analyzed in the following Figures 18–20, respectively.

4.1.1 Overdamped case

When the following conditions are assigned, both response curves of $i(t)$ and $u(t)$ are shown in Figure 18. The decay approaches zero as t increases.

Observed from Figure 17 and Figure 18, the overdamping phenomenon occurs when the memristance increases (i.e., increasing $R_0 + R$), the inductance decreases, or the capacitance increases, while other parameters remain fixed.

4.1.2 Critically damped case

When the following conditions are set, both the current and voltage of the system exhibit maximum and minimum values in Figure 19, respectively. Also, the delays all the way to zero.

Between Figure 17 and Figure 19, the critically damped phenomenon presents immediately when the memristance increases (i.e., increasing $R_0 + R$) but remains much smaller than that in the overdamped case, the inductance decreases, or the capacitance increases, respectively, while other conditions remain fixed.

4.1.3 Underdamped case.

The oscillation period in both $i(t)$ and $u(t)$ curves are depicted in Figure 20. Moreover, the delays all the way to zero.

Compared with Figure 17 and Figure 20, the underdamped phenomenon has been shown as the same situation.

From Figures 18–20, the special characteristics of the $R_M LC$ series circuit could be summarized as follows:

i. In HP-memristor is known as the linear drift model. When current flows through a designed circuit incorporating energy storage elements and a memristor, a higher-order mathematical model can be derived, which surpasses the complexity of conventional *RLC* series circuit models.
ii. Similar to the *RLC* series circuit, its behavior could be characterized by damping phenomena, where the gradual loss of initial stored energy results in a continuous reduction of response amplitude. This explains why such nonlinear circuits

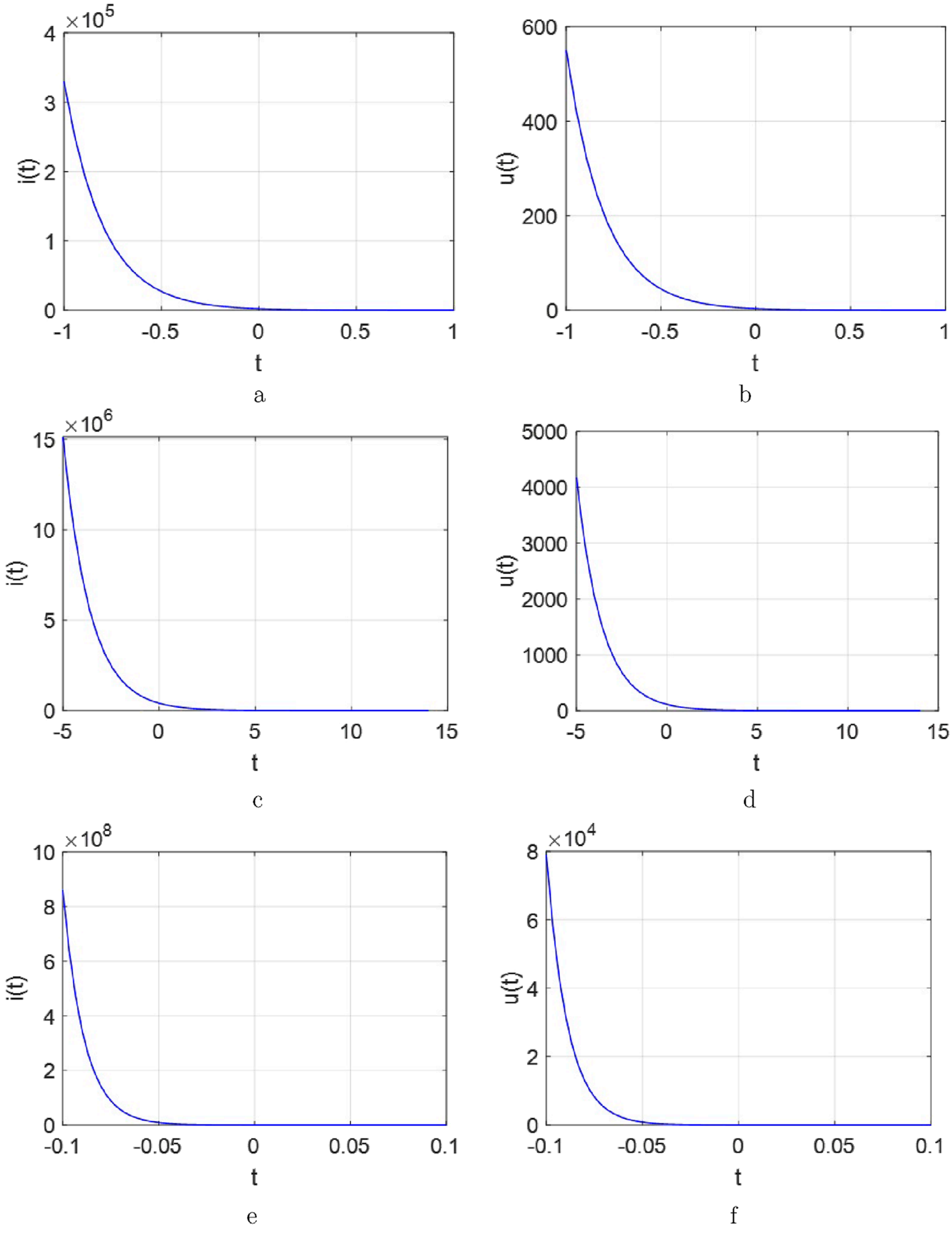


FIGURE 22
The response curves for the $R_M LC$ parallel circuit. **(a)** $i(t) - t$ with $R = 150\Omega$, $L = 0.12H$, and $C = 1mF$. **(b)** $u(t) - t$ with $R = 15\Omega$, $L = 0.12H$, and $C = 1mF$. **(c)** $i(t) - t$ with $R = 1k$, $L = 50H$, and $C = 1mF$. **(d)** $u(t) - t$ with $R = 1k$, $L = 50H$, and $C = 1mF$. **(e)** $i(t) - t$ with $R = 1k$, $L = 0.12H$, and $C = 10uF$. **(f)** $u(t) - t$ with $R = 1k$, $L = 0.12H$, and $C = 10uF$.

with memristors exhibit abundant oscillatory behaviors and strange attractors.

iii. The damping phenomenon arises because a memristor integrates two functional aspects: memory (R_0) and energy dissipation (kR_d). The oscillation period determines the damping rate of the response. To achieve overdamped, critically damped, or underdamped behavior, three discusses can be employed: increasing memristance ($R + R_0$) but remains much smaller than that in the both overdamped and critically damped cases, or capacitance (C) or decreasing the inductance (L), while keeping other parameters fixed, respectively.

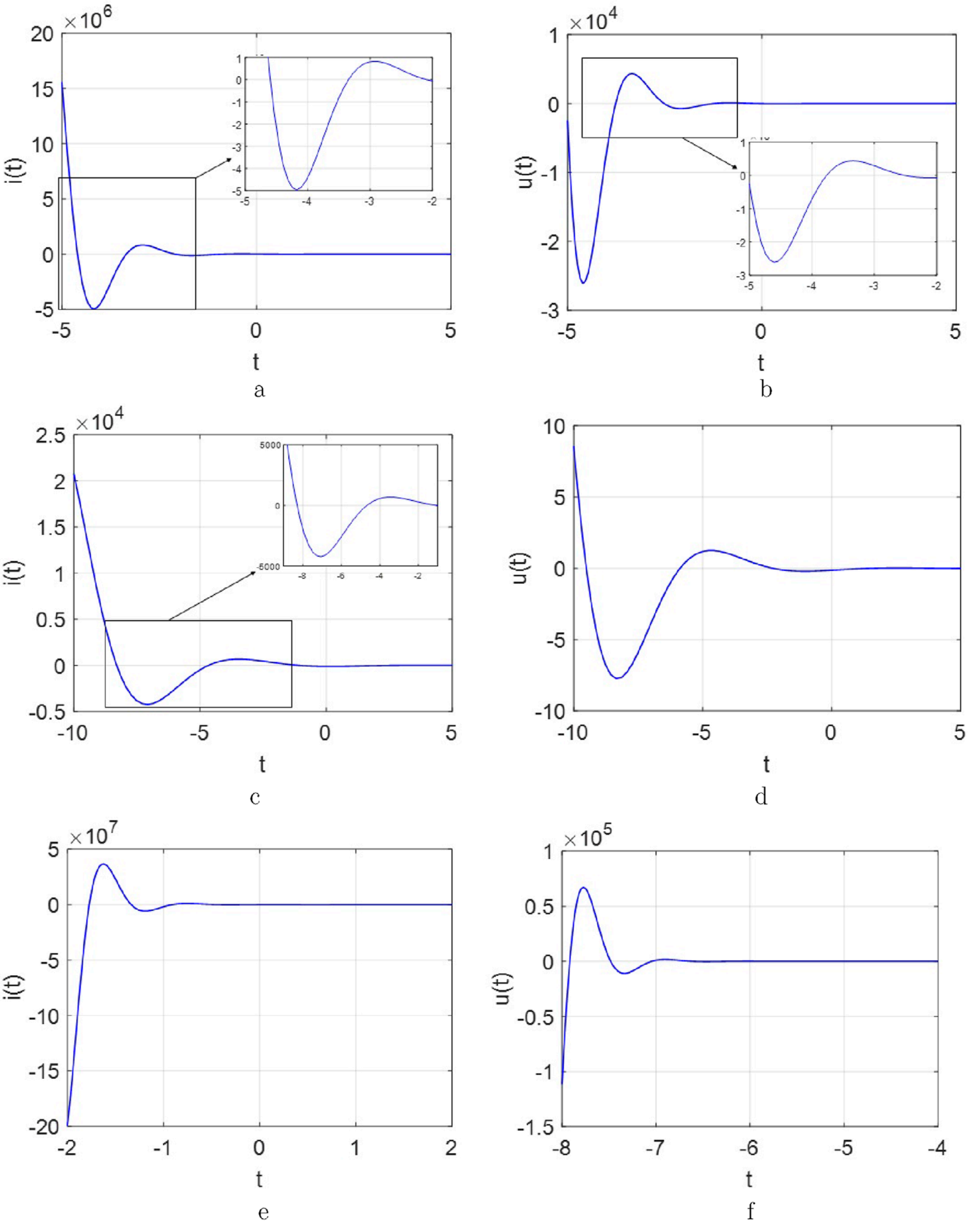


FIGURE 23
The response curves for the $R_M LC$ parallel circuit. **(a)** $i(t) - t$ with $R = 350$, $L = 0.12H$, and $C = 1mF$. **(b)** $u(t) - t$ with $R = 350$, $L = 0.12H$, and $C = 1mF$. **(c)** $i(t) - t$ with $R = 1k$, $L = 1H$, and $C = 1mF$. **(d)** $u(t) - t$ with $R = 1k$, $L = 1H$, and $C = 1mF$. **(e)** $i(t) - t$ with $R = 1k$, $L = 0.12H$, and $C = 120uF$. **(f)** $u(t) - t$ with $R = 1k$, $L = 0.12H$, and $C = 120uF$.

iv. The damped oscillation is possible due to the presence of the nonlinear elements (i.e., R_M, L, and C). Furthermore, the delays all the way to zero, which stems from the ability of the storage elements and memory element to transfer energy back and forth between them.

v. All subplots uniformly validate that the same initial conditions but different component parameters would manifest a similar yet quite different output waveform. This variance could be thought as a kind of catalyst to get various application areas in the future, simultaneously revealing characteristics of chaotic oscillations. It further demonstrates the influence of $(R + R_0)$, L, and C on the decay rate. Therefore, during the design and application of memristive circuits, memristors with appropriate R_0 could be

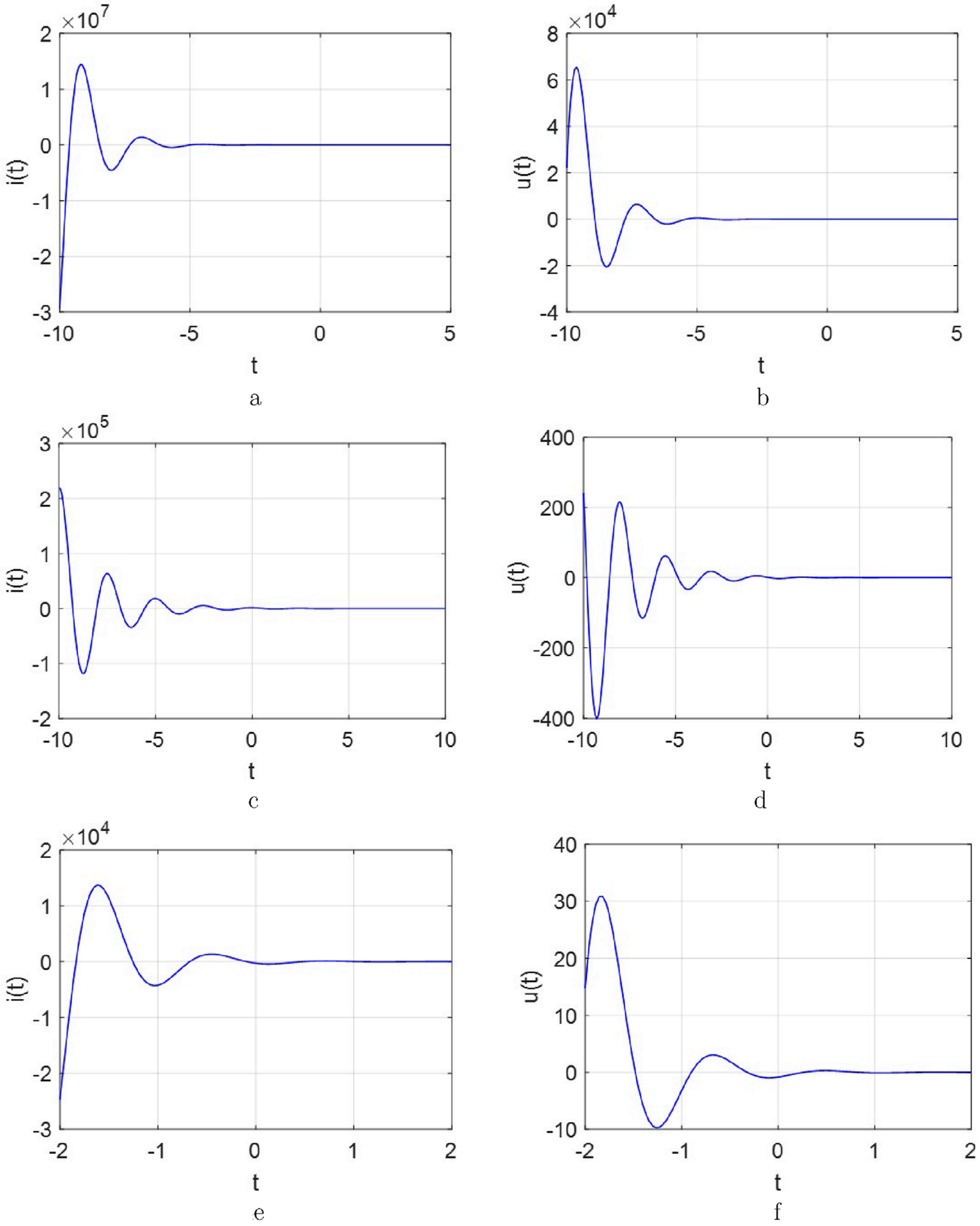


FIGURE 24
The response curves for the $R_M LC$ parallel circuit. **(a)** $i(t) - t$ with $R = 500$, $L = 0.12H$, and $C = 1mF$. **(b)** $u(t) - t$ with $R = 500$, $L = 0.12H$, and $C = 1mF$. **(c)** $i(t) - t$ with $R = 1k$, $L = 0.55H$, and $C = 1mF$. **(d)** $u(t) - t$ with $R = 1k$, $L = 0.55H$, and $C = 1mF$. **(e)** $i(t) - t$ with $R = 1k$, $L = 0.12H$, and $C = 250uF$. **(f)** $u(t) - t$ with $R = 1k$, $L = 0.12H$, and $C = 250uF$.

chosen according to the needs of the actual oscillation and decay rate.

4.2 Parallel circuit

From Figure 16b, when the conditions ($u = u_R = u_C = u_L = u_M$) are satisfied for this parallel circuit, and according to the description of Equation 2, the following Equation 10 have been set as

$$\begin{cases} \dfrac{du}{dt} = -\dfrac{1}{C}\left(\dfrac{u}{R} + \dfrac{u}{R_0 + kR_d q} + i\right) \\ \dfrac{di}{dt} = \dfrac{u}{L} \\ \dfrac{dq}{dt} = i \\ i(0) = I_0, \quad u(0) = U_0 \end{cases} \tag{10}$$

where the variables $i(t) = i_L(t)$, $u(t) = u_c(t)$ stand for the current flowing through the inductor and voltage across the capacitor. System (10) is also a third-order function. Let $u = Ae^{(st)}$ where s and t critical variables. Additionally, the necessary derivatives can be derived as

$$R_0Cs^2 + \left(1 + \frac{R_0}{R}\right)s + \frac{R_0}{L} + \left(kR_dCs^2 + \frac{kR_d}{R}s + \frac{kR_d}{L}\right)Ae^{st} = 0 \quad (11)$$

Equation 11 is still a transcendental equation. Its solution could be obtained through approximately methods. Now, the nature response curve could be drawn in Figure 21.

Next, we investigate the impact of varying parameters (R_M, L or C) and observe whether analogous responses emerge. The current and voltage response curve are presented in Figures 22–24. There are also three cases:

4.2.1 Overdamped case.

When the following conditions are assigned, both response curves of $i(t)$ and $u(t)$ are shown in Figure 22.

Between Figure 21 and Figure 22, the overdamping phenomenon occurs when the memristance decreases (i.e., decreasing $R_0 + R$), the inductance increases, or the capacitance decreases, while other parameters remain fixed.

4.2.2 Critically damped case.

When the following conditions are given, both the current and voltage exhibit maximum and minimum values, respectively (see Figure 23). Also, the delays all the way to zero.

Observed from Figure 21 and Figure 23, the critically damped phenomenon have happened when the memristance decreases (i.e., decreasing $R_0 + R$) but remains much larger than that in the overdamped case, the inductance increases, or the capacitance decreases, while other conditions remain fixed.

4.2.3 Underdamped case.

The oscillation period in both $i(t)$ and $u(t)$ curves are depicted in Figure 24. Moreover, the delays all the way to zero. Compared with Figure 21 and Figure 24, the underdamped phenomenon has been shown under the same conditions.

To summarize the conclusions according to the Figures 22–24 for one R_MLC parallel circuit as follows:

i. Similar to the *RLC* parallel circuit, when energy storage elements and a memristor are integrated into the same parallel system, the energy would be back and forth between them, thereby establishing a damping decay curve.
ii. The coexistence of memory storage and energy dissipation characteristics in this circuit arises from the dual-resistance structure of the memristor, characterized by R_0 and kR_d.
iii. The conditions for achieving overdamped, critically damped, or underdamped phenomena differ from those in R_MLC series circuits. Specifically, these damping regimes can be realized by adjusting the resistance $(R + R_0)$ but remains much larger than that in the both overdamped and critically damped cases or capacitance (C) should be decreasing or increasing the inductance (L), while keeping all other parameters constant under each configuration.
iv. Under identical initial current and voltage conditions but with varying circuit component values, all subplots in the figure were analyzed. These results validate the influence of $(R + R_0)$, L and C on the decay rate. The single regrettable drawback resides in the waveforms lacking sufficient resolution to reveal detailed distinctions between the R_MLC circuit and conventional variable *RLC* systems. However, in the design and application of one memristive circuit, memristors should be selected according to the needs of the actual oscillation and decay rate based on the analysis and discussion in thoery.

5 Application of classic circuits with four fundamental components

A classical four-component application circuit is presented, as shown in Figure 25.

The following analysis would demonstrate how energy storage elements or memristors influence the memory characteristics and oscillatory behavior. The Figure 25a, this circuit shares the same topological structure as the Chua system, but features a different memristor configuration. Consequently, it also produces different phase trajectory curves, the mathematical model has been built and analyzedin the following form:

$$\begin{cases} C_1 \cdot \dfrac{dV_1}{dt} = -\dfrac{1}{R}V_1 + \dfrac{1}{R}V_2 - \dfrac{1}{R_M} \cdot V_1 \\ C_2 \cdot \dfrac{dV_2}{dt} = \dfrac{1}{R}V_1 - \dfrac{1}{R}V_2 + i_L \\ L \cdot \dfrac{di_L}{dt} = -V_2 \\ \dfrac{dq}{dt} = i_L \end{cases} \quad (12)$$

Secondly, when transposing the positions of the HP memristor and resistor in this circuits, the mathematical model is given as follows:

$$\begin{cases} C_1 \cdot \dfrac{dV_1}{dt} = -\dfrac{1}{R_M}V_1 + \dfrac{1}{R_M}V_2 - \dfrac{1}{R} \cdot V_1 \\ C_2 \cdot \dfrac{dV_2}{dt} = \dfrac{1}{R_M}V_1 - \dfrac{1}{R_M}V_2 + i_L \\ L \cdot \dfrac{di_L}{dt} = -V_2 \\ \dfrac{d\varphi}{dt} = \dfrac{d\varphi}{dq} \cdot \dfrac{dq}{dt} = V_1 - V_2 \end{cases} \quad (13)$$

Setting the variable $x = V_1$, $y = V_2$, $z = i_L$ and $\omega = q$; parameters $p_1 = 1/(RC_1)$, $q_1 = 1/(RC_2)$, $a_2 = 1/(RC_1)$, $b = 1/C_2$, $r = 1/L$, $a_1 = p_2 = 1/[(R_0 + kR_dq)C_1]$, $q_2 = 1/[(R_0 + kR_dq)C_2]$, both the built as Equation 12 and Equation 13 can be rewritten in the following dimensionless forms:

$$\begin{cases} \dot{x} = p_1(y - x) - a_1.x \\ \dot{y} = q_1(x - y) + b.z \\ \dot{z} = -r.y \\ \dot{\omega} = z \end{cases} \quad (14)$$

and

$$\begin{cases} \dot{x} = p_2(y - x) - a_2.x \\ \dot{y} = q_2(x - y) + b.z \\ \dot{z} = -r.y \\ \dot{\omega} = x - y \end{cases} \quad (15)$$

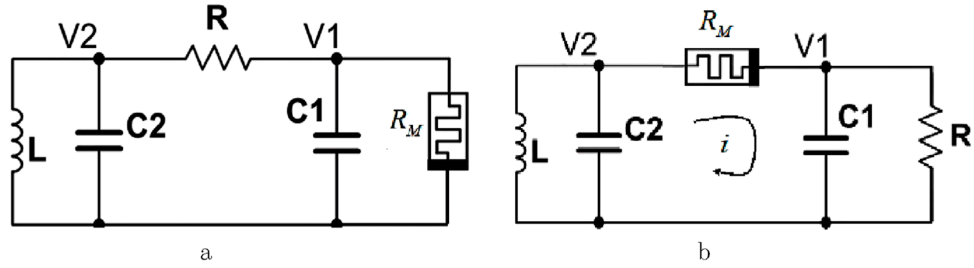


FIGURE 25
A circuit with HP-memristor. **(a)** Replacing the Chua diode with an HP memristor. **(b)** Transposing the positions of HP memristor and resistor.

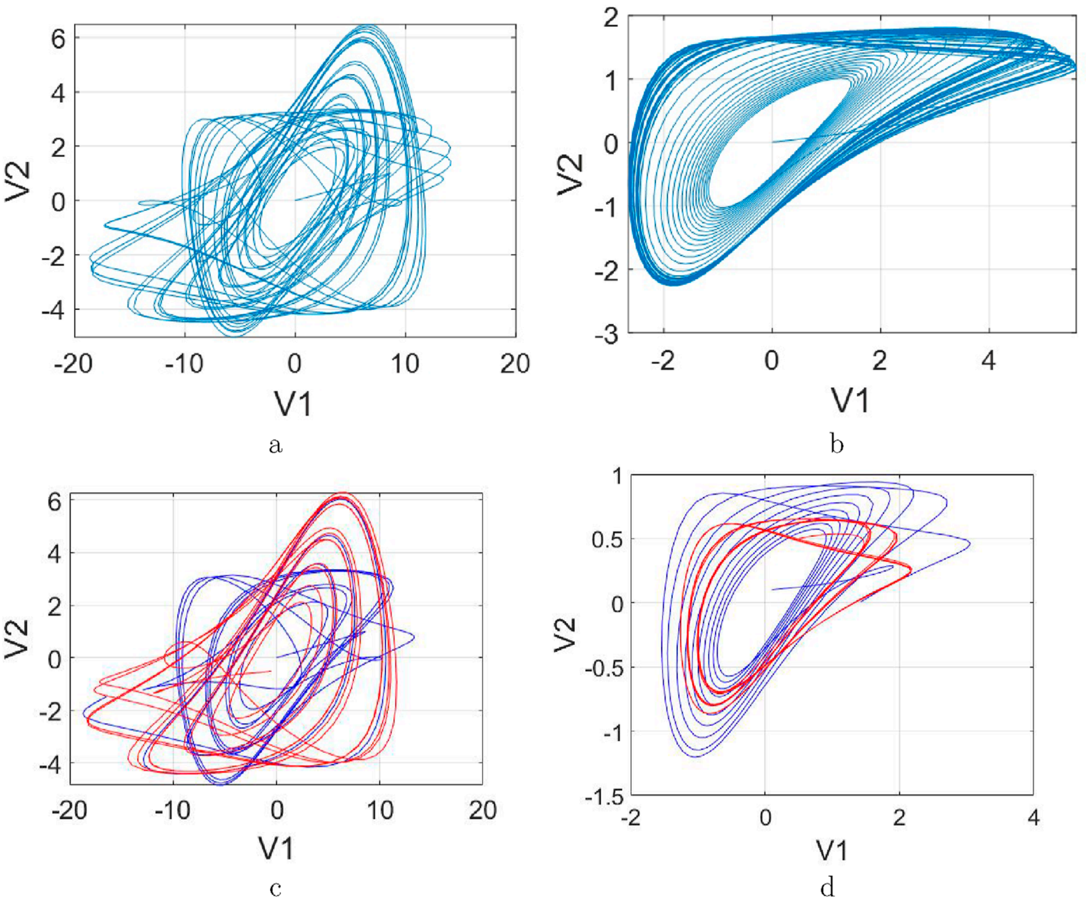


FIGURE 26
Phase portrait in $v_1 - v_2$ and their coexistence attractors. **(a)** replacing Chua Diodes with HP Memristors. **(b)** replace the resistance (R) in original Chua's circuit with an HP memristor. **(c)** Coexistence attractor. **(d)** Hidden attractor.

For Figure 25a, setting the parameters $p_1 = 7.9$, $q_1 = 1$, $b = 1$, $r = 14.5$ are fixed in Equation 14. The phase trajectory curves exhibit the chaotic attractor as shown in Figure 26a. When transposing the positions of the HP memristor and resistor, the phase trajectory becomes a single-scroll attractor as demonstrated in Figure 26b.

These observations demonstrate that as initial values vary, the system not only exhibits irregular oscillations but also manifests chaotic attractors, coexisting attractors in Figure 26c and hidden attractors in Figure 26d. These characteristics serve as critical evidence for the system's capability to facilitate the construction of complex neural networks with memory properties.

Let $p_2 = 1/(6.23 - 0.9q)$, $q_2 = 1/(8.49 - 4.33q)$, $a_2 = 7.9$, $b = 1$, $r = 14.5$, and the initial condition $[x, y, z, w] = [0.01, 0.01, 0.01, 0.01]$, the time domain curves of Equation 15 can be obtained as shown, seeing Figure 27.

Then, the Lyapunov exponent spectrum corresponding to parametrically configured is illustrated in Figure 28.

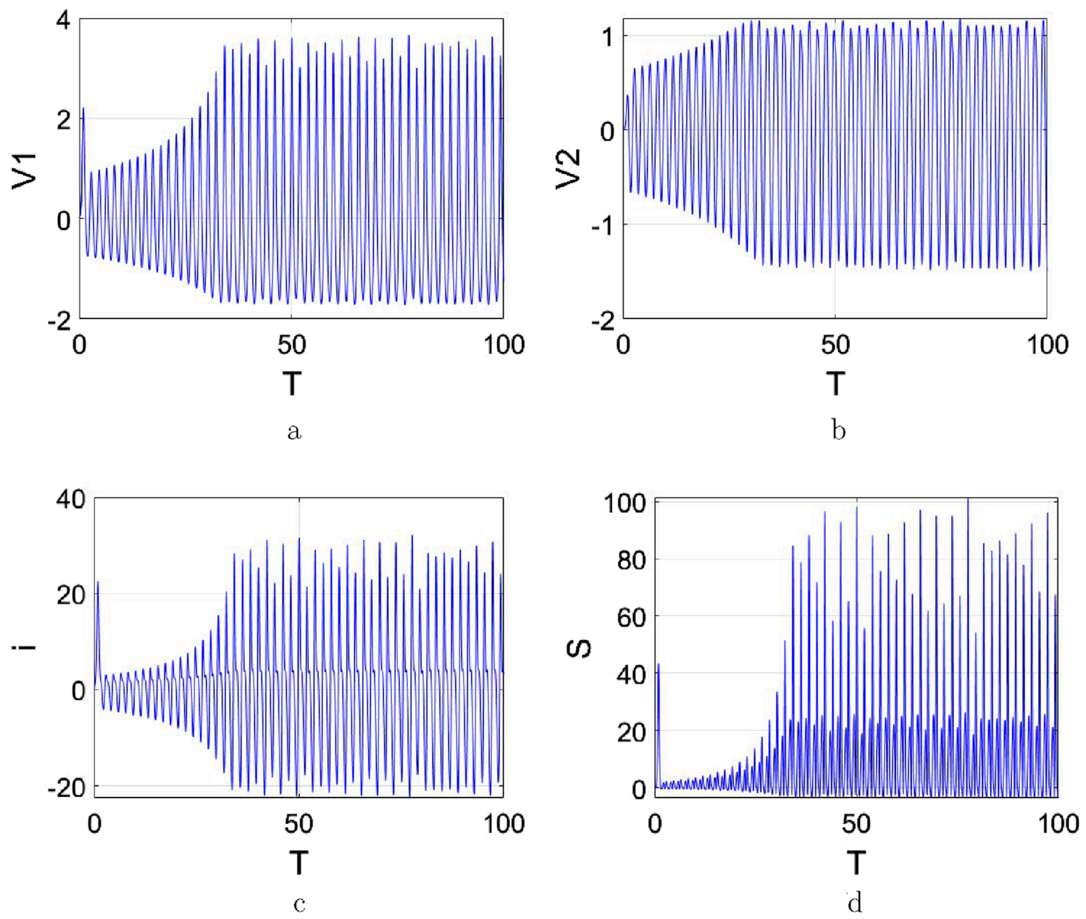


FIGURE 27
Time domain curves. **(a)** $v_1(t) - t$. **(b)** $v_2(t) - t$. **(c)** $i(t) - t$. **(d)** $S(t) - t$.

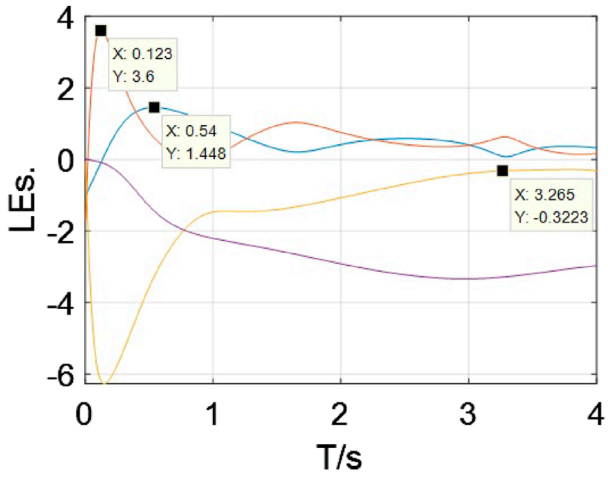


FIGURE 28
The Lyapunov Exponents spectrum.

This provides another perspective to demonstrate that the chaotic oscillation arises from the energy to transfer energy back and forth between the memristor and energy storage elements. From Figure 28, the *LEs* are calculated and illustrated, $LE_1 = 3.6$, $LE_2 = 1.448$, $LE_3 = -0.3223$, $LE_4 = -3.339$. Two positive Lyapunov exponents confirm that the system is a hyperchaotic system. Next, in order to verify the conclusions derived from previous analyses, we systematically modify the values of energy storage elements of Figure 25b to investigate their impact on the memory characteristics and oscillatory behaviors of the HP-memristor from the response of voltage curves.

Observed from Figures 29a–c, they illustrate the effect of varying the inductance L on output voltage (v_2) of the system (15). As inductance L increases, the decay rate diminishes. Conversely, reduction of L induces damped and overdamped dynamical manifestations. When inductance values decrease below critical thresholds, oscillatory phenomena and chaotic attractors undergo complete termination.

Similarly, Figures 29d–f demonstrates the impact of changing the capacitance C_2 on the output voltage (v_2). As capacitance C_2 increases, the decay rate also diminishes. Conversely, reduction of C_2 induces damped and overdamped dynamical manifestations. When capacitance values decrease below critical thresholds, oscillatory phenomena and chaotic attractors undergo complete termination.

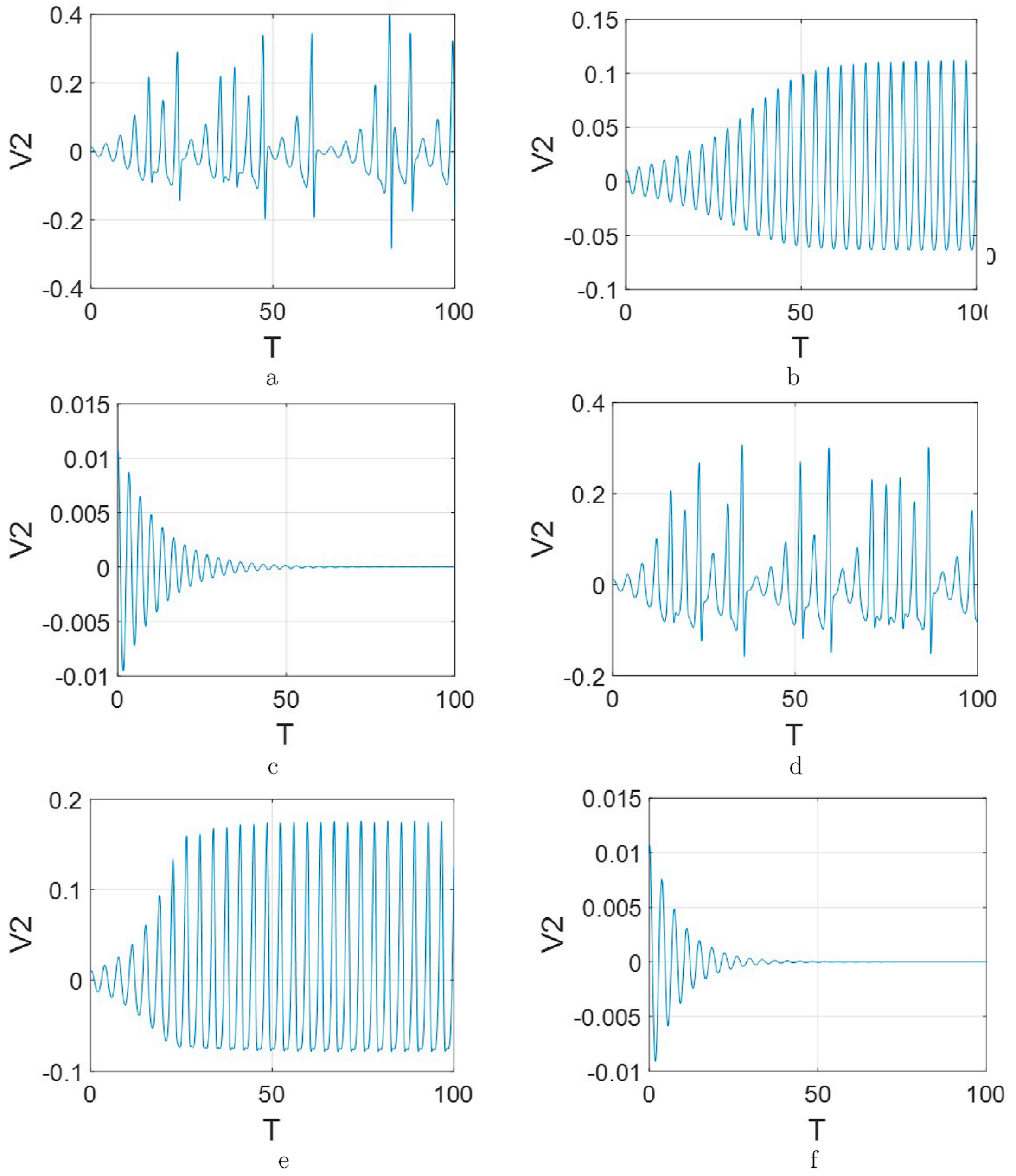


FIGURE 29
The response of voltage curves (v_2) of system (13). **(a)** $a_2 = 7.9$, $b = 1$, $p_2 = 1/(0.9 - 6.23q)$, $q_2 = 1/(8.49 - 4.33q)$, $r = 14.1$. **(b)** $a_2 = 7.9$, $b = 1$, $p_2 = 1/(6.23 - 0.9q)$, $q_2 = 1/(8.49 - 4.33q)$, $r = 16$. **(c)** $a_2 = 7.9$, $b = 1$, $p_2 = 1/(6.23 - 0.9q)$, $q_2 = 1/(8.49 - 4.33q)$, $r = 18$. **(d)** $a_2 = 7.9$, $b = 1$, $p_2 = 1/(6.23 - 0.9q)$, $q_2 = 1/(8.0 - 4.11q)$, $r = 14.5$. **(e)** $a_2 = 7.9$, $b = 1$, $p_2 = 1/(6.23 - 0.9q)$, $q_2 = 1/(8.92 - 4.33q)$, $r = 14.5$. **(f)** $a_2 = 7.9$, $b = 1$, $p_2 = 1/(6.23 - 0.9q)$, $q_2 = 1/(9.2 - 4.92q)$, $r = 14.5$.

6 Conclusion

To advance the fundamental theory of memristive circuits, this study investigates four types of source-free circuits incorporating memristors and energy storage elements following the research methodology of classical source-free circuit analysis.

These circuits are categorized into two groups: one group consists of a memristor combined with a single energy-storing element (denoted as R_MC and R_ML circuits), while the other group includes R_MLC series and parallel circuits. Firstly, their models are built and analyzed, which reveals that they are transcendental equations. Secondly, new time constants are introduced (It pertains exclusively to a specific resistance region in the memristor, such as its low-resistance state R_0, that is, $\tau_0 = R_0C$ and $\tau_0 = L/R_0$), along with key factors influencing the decay rate. Furthermore, this study further verifies that two distinct regions in the memristor manifest two properties: memory characteristics and energy-dissipative behavior. Finally, through a systematic analysis using a classical application circuit with four fundamental circuit elements, we revalidate the critical role of both energy storage components and memristor in modulating oscillatory dynamics and attractor morphologies. More significantly, the characteristics of circuits combining memristors and energy-storage components have been refined, ensuring continuous advancement in memristive circuit principles. This establishes a robust theoretical foundation for innovative applications of memory elements across nonlinear circuits, avionics for UAV systems, and integrated theoretical-design frameworks.

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

XG: Writing – original draft, Writing – review and editing, Supervision. YQ: Formal Analysis, Investigation, Writing – review and editing. SL: Data curation, Methodology, Writing – review and editing. WL: Conceptualization, Data curation, Formal Analysis, Writing – review and editing. YS: Investigation, Supervision, Writing – review and editing. YL: Writing – review and editing, Writing – original draft.

Funding

The author(s) declare that no financial support was received for the research and/or publication of this article.

Conflict of interest

Authors XG, YQ, SL, and WL were employed by Aerospace Times FeiHong Technology Company Limited.

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.

Generative AI statement

The author(s) declare that no Generative AI was used in the creation of this manuscript.

Any alternative text (alt text) provided alongside figures in this article has been generated by Frontiers with the support of artificial intelligence and reasonable efforts have been made to ensure accuracy, including review by the authors wherever possible. If you identify any issues, please contact us.

Publisher's note

References

1. Chua LO. If it's pinched it's a memristor. *Semiconductor Sci Technology* (2014) 49(10):104001. doi:10.1088/0268-1242/29/10/104001

2. Caravelli F, Carbajal JP. Memristors for the curious outsiders. *Technologies* (2018) 6(4):118. doi:10.3390/technologies6040118

3. Zhang Y, Wang X, Li Y, Friedman EG. Memristive model for synaptic circuits. *IEEE Trans Circ Syst.–II: Express Briefs* (2017) 64(7):767–71. doi:10.1109/TCSII.2016.2605069

4. Bao H, Zhang Y, Liu W, Bao B. Memristor synapse-coupled memristive neuron network: synchronization transition and occurrence of chimera. *Nonlinear Dyn* (2020) 100:937–50. doi:10.1007/s11071-020-05529-2

5. Tan Y, Wang C. A simple locally active memristor and its application in HR neurons. *Chaos* (2020) 30(5):053118. doi:10.1063/1.5143071

6. Huang L, Wang S, Lei T, Li C. Coupled HR–HNN neuron with a locally active memristor. *Int J Bifurcation Chaos* (2024) 34(2):2450022. doi:10.1142/S0218127424500226

7. Shama F, Haghiri S, Imani MA. FPGA realization of Hodgkin-Huxley neuronal model. *IEEE Trans Neural Syst Rehabil Eng* (2020) 28(5):1059–68. doi:10.1109/TNSRE.2020.2980475

8. 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) 10:845–58. doi:10.1109/TNSE.2022.3223930

9. Eshraghian JK, Cho K, Zheng C, Nan M, Iu HH, Lei W, et al. Neuromorphic vision hybrid rram-cmos architecture. *IEEE Trans Very Large-Scale Integration (VLSI) Syst* (2018) 26(12):2816–29. doi:10.1109/TVLSI.2018.2829918

10. Yu F, Zhang S, Su D, Wu Y, Gracia YM, Yin H. Dynamic analysis and implementation of FPGA for a new 4D fractional-order memristive hopfield neural network. *Fractal and Fractional* (2025) 9(2):115. doi:10.3390/fractalfract9020115

11. Liu Y, Iu HH, Qian Y. Implementation of Hodgkin-Huxley neuron model with the novel memristive oscillator. *IEEE Trans Circ Syst.–II: Express Briefs* (2021) 68(8):2982–6. doi:10.1109/TCSII.2021.3066471

12. Song Z, Liu Y. Generation of neuromorphic oscillators via second-order memristive circuits with modified Chua corsage memristor. *IEEE Access* (2023) 11:103712–24. doi:10.1109/ACCESS.2023.3318117

13. Yu D, Zhao X, Sun T, Iu HHC, Fernando T. A simple floating mutator for emulating memristor, memcapacitor, and meminductor. *IEEE Trans Circ Syst.–II: Express Briefs* (2020) 67(7):1334–8. doi:10.1109/TCSII.2019.2936453

14. Feali MS, Ahmadi A, Hayati M. Implementation of adaptive neuron based on memristor and memcapacitor emulators. *Neurocomputing* (2018) 309(2):157–67. doi:10.1016/j.neucom.2018.05.006

15. Liu Y, Iu HH, Guo Z, Si G. The simple charge-controlled grounded/floating mem-element emulator. *IEEE Trans Circ Syst.–II: Express Briefs* (2021) 68(6):2177–81. doi:10.1109/TCSII.2020.3041862

16. Yu F, Su D, He S. Resonant tunneling diode cellular neural network with memristor coupling and its application in police forensic digital image protection. *Chin Phys B* (2021) 34(5):050502. doi:10.1088/1674-1056/adb8bb

17. Chai Q, Liu Y. MARR-GAN attention recurrent residual generative Adversarial Network with memristor for Raindrop Removal. *Micromachiness* (2024) 15(2):15020217. doi:10.3390/mi15020217

18. 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 Circ Theor Appl* (2021) 49(2):513–32. doi:10.1002/cta.2915

19. Corinto F, Ascoli A. Memristive diode bridge with LCR filter. *Electronics Lett* (2012) 48(14):824–5. doi:10.1049/el.2012.1480

20. Ntinas V, Vourkas I, Sirakoulis GC. LC filters with enhanced memristive damping. In: *2015 IEEE international symposium on circuits and systems (ISCAS)* (2015). p. 2664–7. doi:10.1109/ISCAS.2015.7169234

21. Torebayev G, Nandakumar A. Implementation of memristor in Bessel filter with RLC components. *Int Conf Comput Netw Commun (Coconet)* (2018) 190–4. doi:10.1109/CoCoNet.2018.8476913

22. Sadecki J, Marszalek W. Analysis of a memristive diode bridge rectifier. *Electronics Lett* (2019) 55(3):120–2. doi:10.1049/el.2018.6921

23. Li C, Zhou Y, Yang Y, Li H, Feng W, Li Z, et al. Complicated dynamics in a memristor-based RLC circuit. *Eur Phys J Spec Top* (2019) 228:1925–41. doi:10.1140/epjst/e2019-800195-8

24. Liu Y, Wang K, Qian Y, Liu X, Wang S. Neuromorphic oscillators and electronic implementation of one novel MFNN model. *IEEE Access* (2023) 11:16076–84. doi:10.1109/ACCESS.2022.3233771

25. Dautovic S, Samardzic N, Juhas A, Ascoli A, Tetzlaff R. Analytical solutions for charge and flux in HP ideal generic memristor model with Joglekar and Prodromakis window functions. *IEEE Access* (2024) 12:94870–84. doi:10.1109/ACCESS.2024.3424568

26. So-Yeon K, Heyi Z, Rivera-Sierra G, Fenollosa R, Rubio-Magnieto J, Bisquert J. Introduction to neuromorphic functions of memristors: the inductive nature of synapse potentiation. *J Appl Phys* (2025) 137:111101. doi:10.1063/5.0257462

27. Liu Y, Liu F, Luo W, Wu A, Li H. AC power analysis for second-order memory elements. *Front Phys* (2023) 11:1135739. doi:10.3389/fphy.2023.1135739

28. Yu F, He S, Yao W, Cai S, Xu Q. Bursting firings in memristive hopffeld neural network with image encryption and hardware implementation. *IEEE Trans Computer-Aided Des Integrated Circuits Syst* (2025) 161–73. doi:10.1109/TCAD.2025.3567878

29. Liu Y, Li H, Guo S, Iu HH. Generation of multi-lobe chua corsage memristor and its neural oscillation. *Micromachines* (2022) 13(8):1330–43. doi:10.3390/mi13081330

30. Liu Y, Iu HH. Novel floating and grounded memory interface circuits for constructing mem-elements and their applications. *IEEE Access* (2020) 8:114761–72. doi:10.1109/ACCESS.2020.3004160

31. Liu Y, Iu HH, Guo S, Li H. Chaotic dynamics in memristive circuits with different -q characteristics. *Jul 2021 Int J Circ Theor Appl* (2022) 49(11):3540–58. doi:10.1002/cta.3112

frontiers | Frontiers in Physics

TYPE Original Research
PUBLISHED 14 October 2025
DOI 10.3389/fphy.2025.1655166

OPEN ACCESS

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

REVIEWED BY
Huihai Wang,
Central South University, China
Baoxiang Du,
Heilongjiang University, China

*CORRESPONDENCE
Li Niu,
8201501009@jiangnan.edu.cn

RECEIVED 27 June 2025
ACCEPTED 17 July 2025
PUBLISHED 14 October 2025

CITATION
Tu C, Niu L and Cui R (2025) A child information protection scheme based on hyperchaotic mapping.
Front. Phys. 13:1655166.
doi: 10.3389/fphy.2025.1655166

A child information protection scheme based on hyperchaotic mapping

Chenchen Tu[1,2], Li Niu[3]* and Rongrong Cui[4]

[1]College of Textile Science and Engineering, Jiangnan University, Wuxi, Jiangsu, China, [2]School of Fashion, Dalian Polytechnic University, Dalian, Liaoning, China, [3]School of Digital Technology & Innovation Design, Jiangnan University, Wuxi, Jiangsu, China, [4]School of Fashion Design and Engineering, Zhejiang Sci-Tech University, Hangzhou, Zhejiang, China

This paper proposes an encryption scheme based on hyperchaotic mapping for child information protection. First, phase diagrams of the hyperchaotic mapping are plotted under different parameter combinations, and the variation in phase trajectories confirms the sensitivity of the hyperchaotic mapping to control parameters. Then, the hyperchaotic mapping is iterated to obtain chaotic sequences, and the chaotic sequences are quantized to obtain pseudo-random sequences. Finally, based on those, a scrambling algorithm and a diffusion algorithm are designed to encrypt and protect the images. The original images are scrambled and diffused to obtain the ciphertext images and used to protect the information of missing children, which can effectively protect the safety of children's information and assist the public security bureaus to quickly contact the parents of missing children.

KEYWORDS

child information protection, hyperchaotic mapping, image encryption, children's clothes, information security

1 Introduction

In the digital era, data are increasingly becoming an important part of personal life and economic development [1, 2]. Among various data formats, images are widely used as information carriers for Internet transmission as they can carry large amounts of information and have high visibility [3, 4]. Due to the dependence of work life on the Internet, the rich information contained in images is at risk of being leaked [5–7]. Among these, the secure transmission and storage of image data face significant challenges as they contain sensitive information such as biometrics and geographic locations [8, 9]. Especially in the field of social welfare, such as missing children tracking, images need to be widely disseminated to expand the search scope, but they also must be prevented from being maliciously utilized to cause secondary damage [10, 11]. Image encryption can be used to encrypt an image into a noise-like ciphertext image by various means [12–15].

As an effective method to protect image information, image encryption techniques, especially those based on chaos theory, have been a hot topic of research in recent years [16]. This is because many inherent properties of chaotic systems, including ergodicity, acyclicity, high sensitivity to initial conditions and control parameters, and pseudo-randomness, meet the needs of cryptography and have an irreplaceable

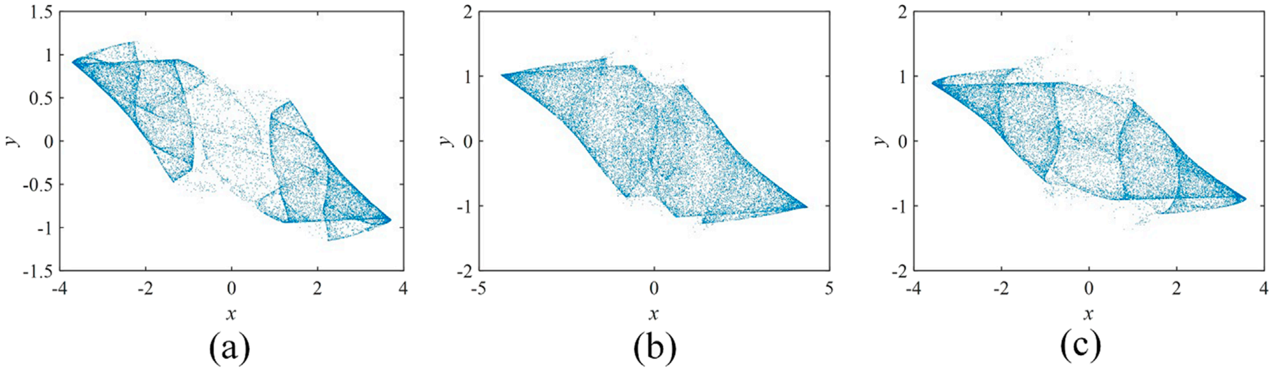


FIGURE 1
Phase diagrams of MCHM, (x_0, y_0) = [0.3, 0.5]: **(a)** (*a*, *b*, *c*, *d*, and *e*) = [-1, 1.5, −1, 1, 0.5]; **(b)** (*a*, *b*, *c*, *d*, and *e*) = [-1, 1.5, −1.1, 1, 0.5]; **(c)** (*a*, *b*, *c*, *d*, and *e*) = [-1, 1.5, −1, 1.2, 0.5].

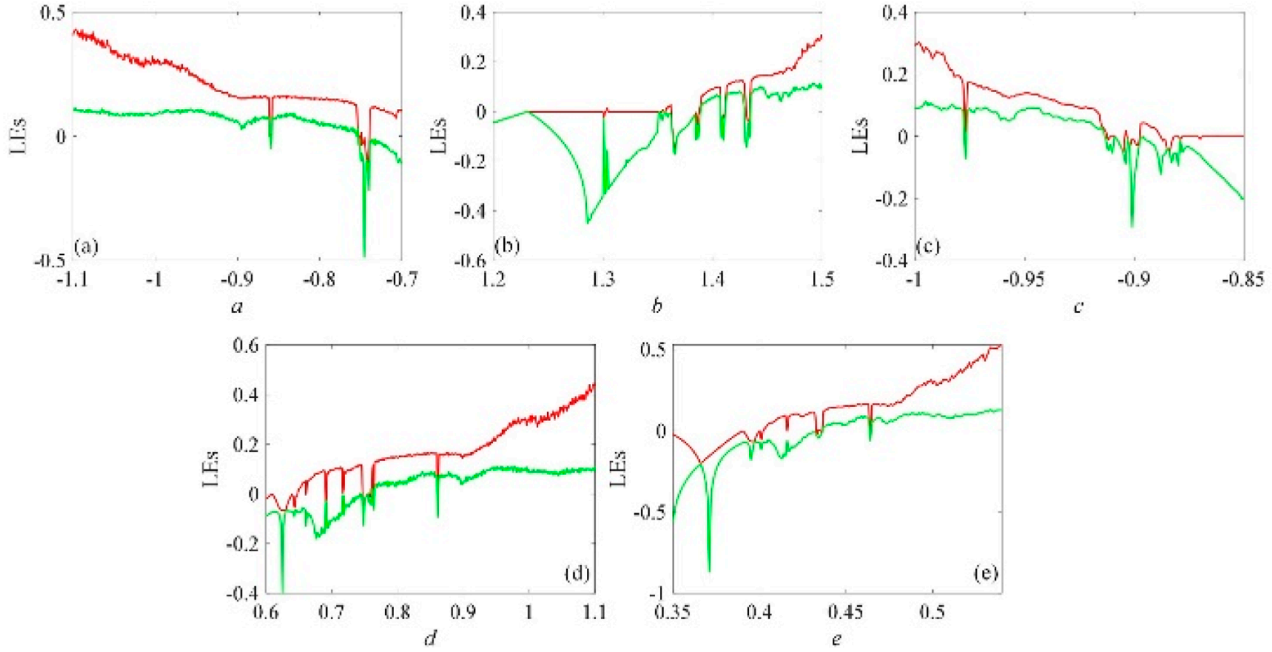


FIGURE 2
Distribution of LEs with different parameters. **(a)** Parameter *a*; **(b)** parameter *b*; **(c)** parameter *c*; **(d)** parameter *d*; **(e)** parameter *e*.

advantage in image encryption [17–21]. Meanwhile, some scholars have pointed out that hyperchaotic systems can provide higher security to encryption algorithms [22, 23]. In the context of information protection and verification of missing children, hyperchaotic mapping is preferred in view of the need for real-time performance. In this study, hyperchaotic mapping [24–30] is used in the design of the missing child information encryption scheme.

In the previous image encryption scheme design and application, usually, the image is compressed and encrypted to realize the fast transmission and protection of the image on the Internet [31–33]; another method includes encoding and encryption of the image to realize the safe storage of the information and prevent leakage or tampering [31, 34–36]; there is also the encryption and steganography of the image to realize the double-layer protection of the image [37–39]. However, information protection and verification of missing children are different from the previous image encryption protection processes, where the main idea is to encrypt children's information to obtain ciphertext images and apply the ciphertext images to children's products as stickers, such as on children's school bags, water cups, and clothes. In the process of children getting lost or being found, the children's information is verified, and it is convenient to get in touch with the children's

TABLE 1 NIST test results for MCHM.

Items	MCHM	
	P-value	PR (%)
Frequency	0.198732	98
Block frequency	0.346291	99
Cumulative sums	0.637462	99
Runs	0.029485	97
Longest run	0.875123	100
Rank	0.372634	99
FFT	0.082712	99
Non-overlapping template	0.028374	100
Overlapping template	0.192734	100
Universal	0.876123	98
Approximate entropy	0.468102	98
Random excursions	0.076390	100
Random excursions variant	0.048716	100
Serial	0.419021	99
Linear complexity	0.289667	100

parents quickly. In view of this application idea, this paper designs a missing child information protection and verification scheme based on hyperchaotic mapping. The information protection scheme is divided into two steps: image scrambling and diffusion, which when combined with chaotic sequences can effectively hide the original information of children, and the reversible encryption scheme ensures that the information of children can be decrypted and verified quickly.

This paper carries out the following tasks:

1. Memristor-coupled hyperchaotic mapping (MCHM) is presented in this paper, and its phase diagram is analyzed.
2. The image or photo containing a child's information is encrypted with a confusion algorithm and a diffusion algorithm.
3. Security analysis of encrypted images to highlight the superiority of the scheme.

2 Chaotic mapping

MCHM is obtained by coupling the memristor and the iterative chaotic map with infinite collapse (ICMIC), and its mathematical model is described as Equation 1:

$$\begin{cases} x_{i+1} = \sin\left(\dfrac{a}{x_i}\right) + b\left(c + dy_i^2\right)x_i \\ y_{i+1} = y_i + ex_i \end{cases}. \tag{1}$$

When the initial value is (x_0, y_0) = [0.3, 0.5] and the system parameters a, b, c, d, and e are [−1, 1.5, −1, 1, 0.5], [−1, 1.5, −1.1, 1, 0.5], and [−1, 1.5, −1, 1.2, 0.5], the phase diagrams of MCHM are as shown in Figure 1. Comparing Figures 1a–c, the trajectory of the MCHM clearly changes when the control parameters are changed. That is, when the key changes slightly during the operation of the encryption scheme, the chaotic sequences generated by the MCHM also change, which changes the cipher images. This means that the mapping can provide great security for the design and operation of the encryption scheme.

The chaotic range of the system parameters is decided by analyzing the Lyapunov exponent (LE) response. If one of the LEs is greater than 0, it is in a chaotic state, and if two LEs are greater than 0, it is in a hyperchaotic state. Encryption is carried out in the chaotic state situation by selecting the parameters. As shown in Figure 2, the range case of each parameter in a chaotic state is $a\in$[-1.1,-0.75], $b\in$[1.35,1.5], $c\in$[-1,-0.9], $d\in$[0.7,1.1], and $e\in$[0.4,0.54]. When parameter a is at [-1.1, -0.875] or [-0.87, -0.747], parameter b is at [1.43, 1.5], parameter c is at [-1, -0.975] or [-0.96, -0.92], parameter d is at [0.85, 1.1], and parameter e is at [0.47, 0.54], the system is in a hyperchaotic state. More complex dynamic characteristics are shown in this state, and the pseudo-random sequence generated by the system through iteration has higher randomness.

To test the randomness of the chaotic sequences, an NIST test (NIST SP800-22) is performed. It includes 15 tests. When the p-value is greater than or equal to 0.01 and the pass rate is greater than 96%, the sequence passes the randomness test. The specific test results are shown in Table 1. It can be seen from the results that this random sequence exhibits good randomness characteristics in statistical tests. It is shown that it is suitable for the proposed encryption scheme.

3 Encryption scheme

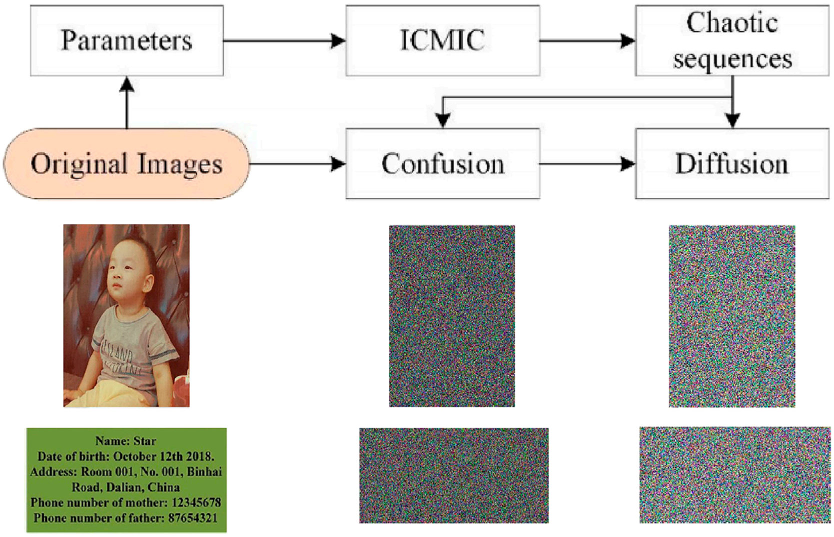
The encryption process includes three stages: parameter setting, image confusion, and diffusion. The encryption schematic is shown in Figure 3. The detailed steps are described as follows:

Step 1: The image containing the child's information and photo is imported, and the size of the ith image is recorded as $m_i \times n_i \times l_i$.

Step 2: All images are converted into column vectors, and all column vectors are stitched into a whole, which is denoted as vector A, with length vl.

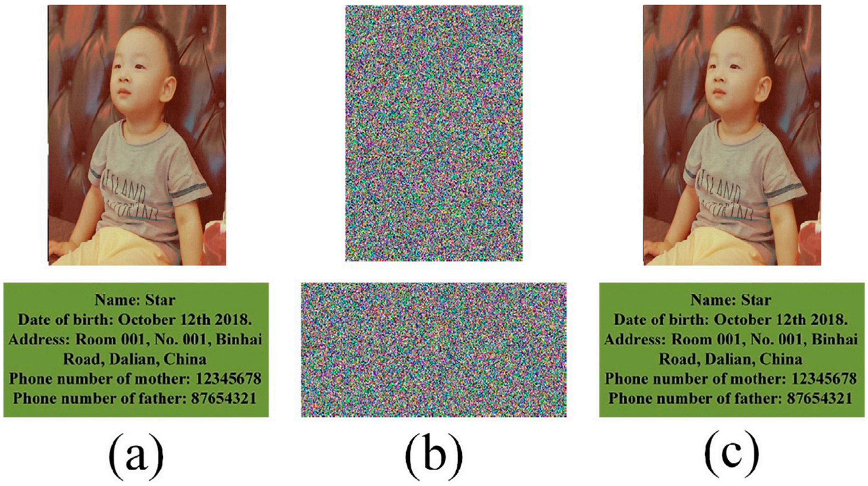
Step 3: Column vector A is converted to cube B with dimensions $M \times N \times L$, where M and N are the height and width of each plane of the cube, respectively, and L is the height of the cube. M and N can be set as desired, and L is obtained by Equation 2.

$$\boldsymbol{L} = \mathrm{ceil}\left(\frac{vl}{MN}\right). \tag{2}$$



Name: Star
Date of birth: October 12th 2018.
Address: Room 001, No. 001, Binhai Road, Dalian, China
Phone number of mother: 12345678
Phone number of father: 87654321

FIGURE 3
Encryption schematics.



Name: Star
Date of birth: October 12th 2018.
Address: Room 001, No. 001, Binhai Road, Dalian, China
Phone number of mother: 12345678
Phone number of father: 87654321

Name: Star
Date of birth: October 12th 2018.
Address: Room 001, No. 001, Binhai Road, Dalian, China
Phone number of mother: 12345678
Phone number of father: 87654321

FIGURE 4
Simulation results: **(a)** original images, "kid" and "information1"; **(b)** encryption images; **(c)** decryption images.

Step 4: Based on the input image, the parameters associated with the plaintext h_i are obtained.

$$\begin{cases} Hm(i) = -\sum_{j=1}^{255} p_j \log_2(p_j), i = 1 \cdots L \\ hm(i) = Hm(i) - \mathrm{floor}(Hm(i)), i = 1 \cdots L \\ h_i = \frac{1}{L} \sum_{j=(i-1)\mathrm{floor}(l)+1}^{i\mathrm{floor}(l)} hm(j), i = 1 \cdots 7 \\ l = \frac{L}{7} \end{cases}, \tag{3}$$

where P_j stands for the pixel value and Hm stands for information entropy.

Step 5: All the keys are inputted, and the MCHM is iterated based on the total image data volume vl to obtain the chaotic sequences of length $2 \times vl$, and they are quantized to finally obtain two pseudo-random sequences x and y. The pseudo-random sequences q_1–q_{12} used in the algorithm are obtained by Equations 4–8.

$$\alpha = \max(M, N, L). \tag{4}$$

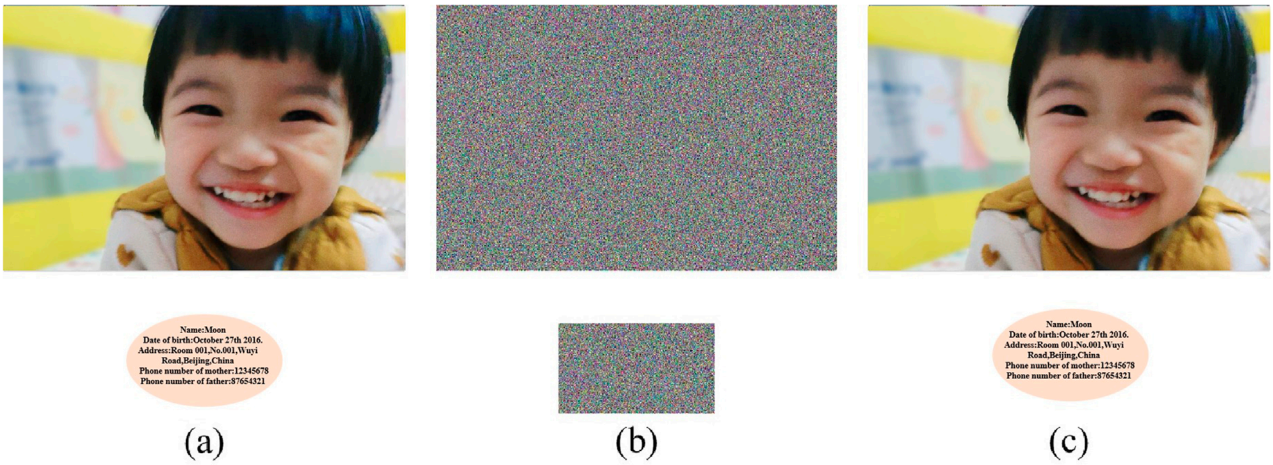


FIGURE 5
Simulation results: **(a)** original images, “girl” and “information2”; **(b)** encryption images; **(c)** decryption images.

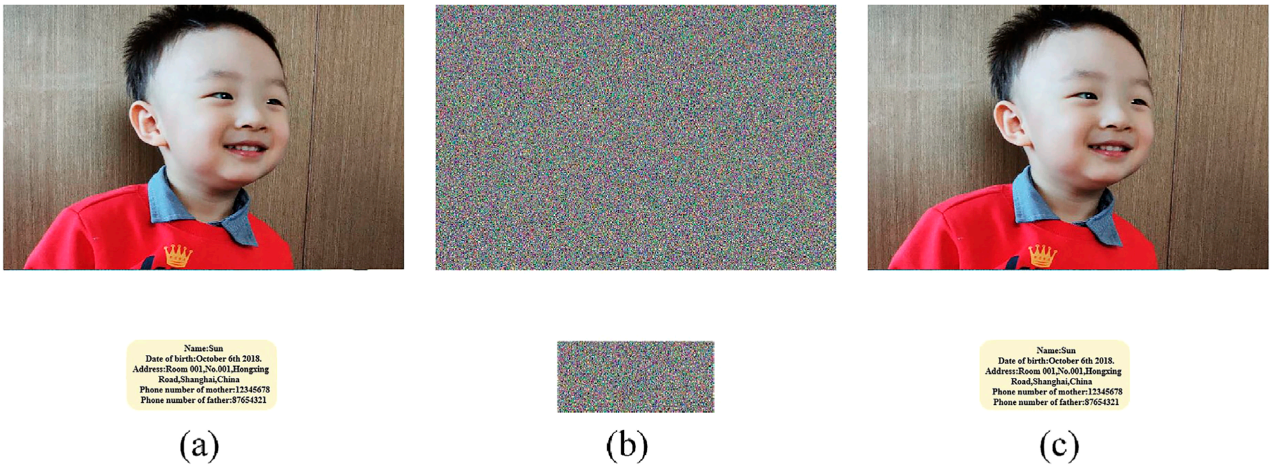


FIGURE 6
Simulation results: **(a)** original images, “boy” and “information3”; **(b)** encryption images; **(c)** decryption images.

$$\begin{cases} q_1 = ((x(1{:}\alpha) + y(1{:}\alpha)) \bmod N) + 1 \\ q_2 = ((x(\alpha+1{:}2\alpha) + y(\alpha+1{:}2\alpha)) \bmod N) + 1 \end{cases}. \tag{5}$$

$$\begin{cases} q_3 = (x(2\alpha+1{:}3\alpha) \bmod M) + 1 \\ q_4 = (y(2\alpha+1{:}3\alpha) \bmod L) + 1 \\ q_5 = (x(3\alpha+1{:}4\alpha) \bmod M) + 1 \\ q_6 = (y(3\alpha+1{:}4\alpha) \bmod L) + 1 \end{cases}. \tag{6}$$

$$\begin{cases} q_7 = (x(4\alpha+1{:}5\alpha) \bmod N) + 1 \\ q_8 = (y(4\alpha+1{:}5\alpha) \bmod L) + 1 \\ q_9 = (x(5\alpha+1{:}6\alpha) \bmod N) + 1 \\ q_{10} = (y(5\alpha+1{:}6\alpha) \bmod L) + 1 \end{cases}. \tag{7}$$

$$\begin{cases} q_{11} = x(end - MNL + 1{:}end) \\ q_{12} = y(end - MNL + 1{:}end) \end{cases}. \tag{8}$$

Step 6: The sequences q_1 and q_2 are processed and used to control the length of the permutation sequence, and q_1 and q_2 can be obtained by Equations 9, 10.

$$\boldsymbol{q}1(i) = \begin{cases} q1(i) + \mathrm{ceil}\left(\frac{N}{4}\right), q1(i) < \frac{N}{4} \\ q1(i) - floor\left(\frac{N}{4}\right), q1(i) > \frac{3N}{4} \\ q1(i), other \end{cases}. \tag{9}$$

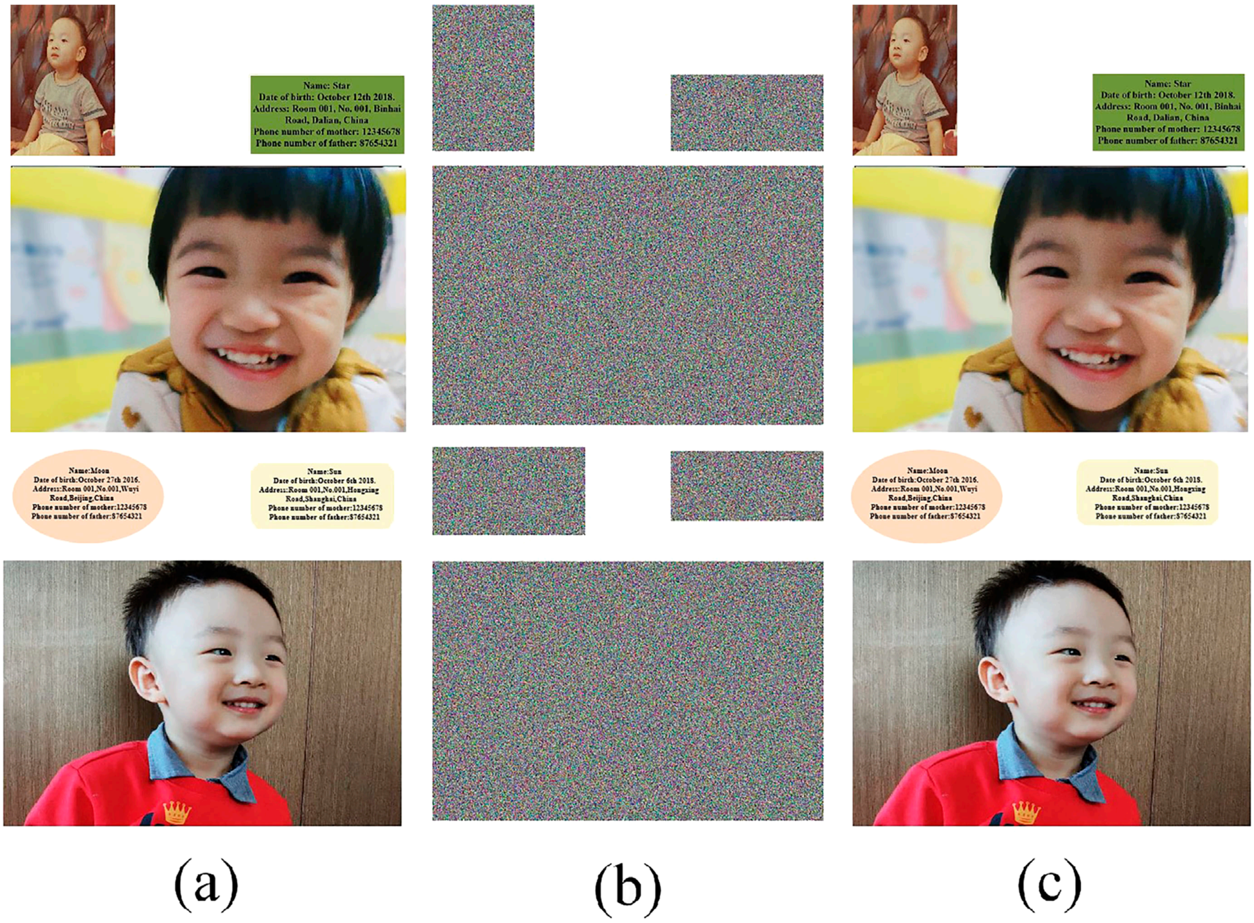


FIGURE 7
Simulation results: **(a)** original images; **(b)** encryption images; **(c)** decryption images.

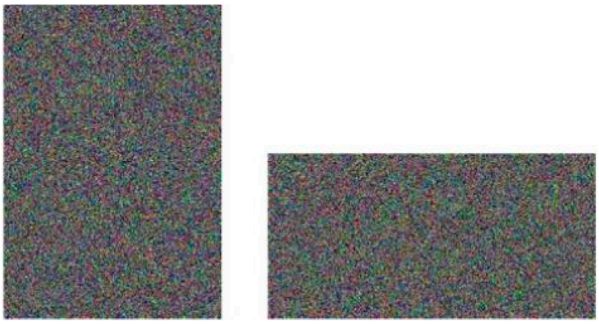


FIGURE 8
Key sensitivity test results, $a = a{+}10^{-16}$.

$$\boldsymbol{q}2(i) = \begin{cases} q2(i) + \mathrm{ceil}\left(\frac{M}{4}\right), q1(i) < \frac{M}{4} \\ q2(i) - floor\left(\frac{M}{4}\right), q1(i) > \frac{3M}{4} \\ q2(i), other \end{cases}. \tag{10}$$

Step 7: Each row vector of cube B is split into two parts of random length, and the positions are swapped with the row vectors at random locations.

$$\begin{cases} t1 = B(i, 1{:}q1(i), k) \\ t2 = B(i, q1(i) + 1{:}end, k) \\ t3 = B(q3((ik \bmod M) + 1), 1{:}q1(i), q4((ik \bmod L) + 1)) \\ t4 = B(q5((ik \bmod M) + 1), q1(i) + 1{:}end, q6((ik \bmod L) + 1)) \end{cases}, \begin{matrix} i = 1 \ldots M \\ k = 1 \ldots L \end{matrix}. \tag{11}$$

$$\begin{cases} B(i, 1{:}q1(i), k) = t3 \\ B(q3((ik \bmod M) + 1), 1{:}q1(i), q4((ik \bmod L) + 1)) = t1 \\ B(i, q1(i) + 1{:}end, k) = t4 \\ B(q5((ik \bmod M) + 1), q1(i) + 1{:}end, q6((ik \bmod L) + 1)) = t2 \end{cases}, \begin{matrix} i = 1 \ldots M \\ k = 1 \ldots L \end{matrix}. \tag{12}$$

Step 8: Each column vector of cube B is split into two parts of random length, and the positions are swapped with the column vectors at random locations. The cube with the completed column swap is noted as C. It can be obtained by Equations 13, 14.

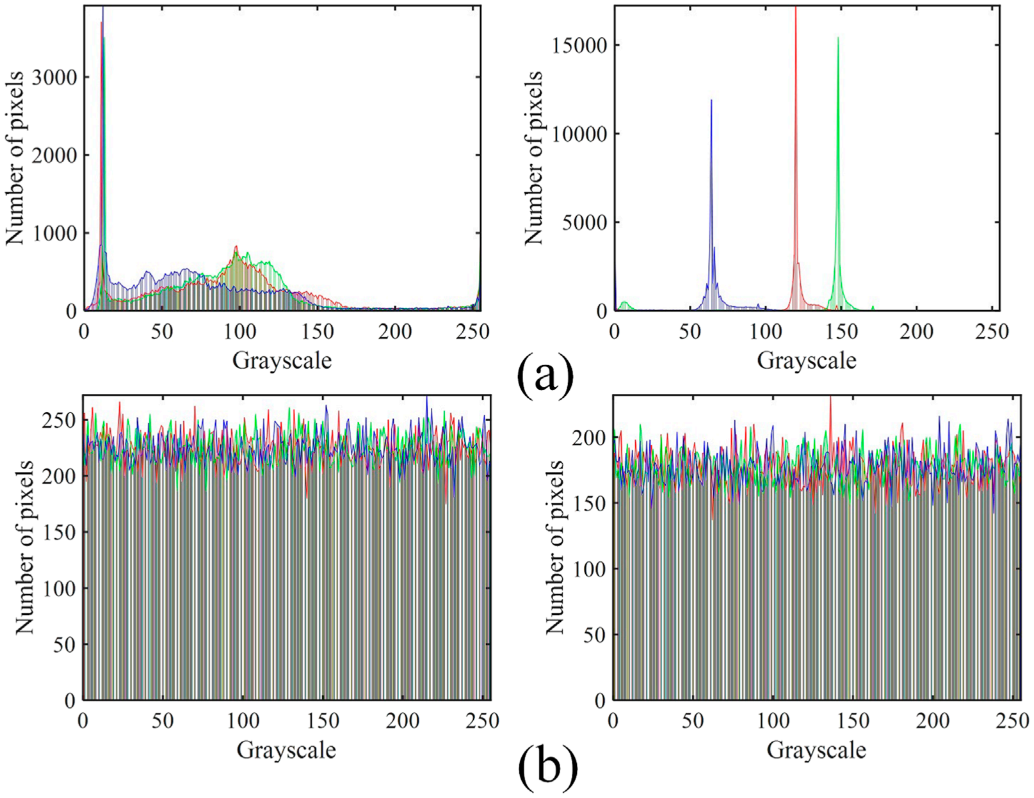


FIGURE 9
Histogram of "kid" and "information1": **(a)** original images and **(b)** encryption images.

TABLE 2 Comparison of key spaces

Algorithms	Reference [40]	Reference [16]	Reference [41]	Reference [42]	Proposed
Key space	2^{249}	2^{266}	2^{352}	2^{398}	2^{471}

$$\begin{cases} t1 = B(1{:}q2(j),j,k) \\ t2 = B(q2(i)+1{:}end,j,k) \\ t3 = B(1{:}q2(j),q7((jk \bmod N)+1),q8((jk \bmod L)+1)) \\ t4 = B(q2(i)+1{:}end,q9((jk \bmod N)+1),q10((jk \bmod L)+1)) \end{cases}, \begin{matrix} j = 1 \dots N \\ k = 1 \dots L \end{matrix}. \tag{13}$$

$$\begin{cases} B(1{:}q2(j),j,k) = t3 \\ B(1{:}q2(j),q7((jk \bmod N)+1),q8((jk \bmod L)+1)) = t1 \\ B(q2(i)+1{:}end,j,k) = t4 \\ B(q2(i)+1{:}end,q9((jk \bmod N)+1),q10((jk \bmod L)+1)) = t2 \end{cases}, \begin{matrix} j = 1 \dots N \\ k = 1 \dots L \end{matrix}. \tag{14}$$

Step 9: Cube C is converted into column vector D, and the first pixel value is combined with the pseudo-random sequence to get the new pixel value. It can be obtained by Equations 15, 16.

$$\boldsymbol{E}(1) = D(1) \oplus q11(1). \tag{15}$$

$$\begin{cases} E(i) = D(i) \oplus q11(i) \oplus E(i-1), i \bmod 2 = 1 \\ E(i) = D(i) \oplus q12(i) \oplus E(i-1), i \bmod 2 = 0 \end{cases}, i = 2 \dots MNL. \tag{16}$$

Step 10: The vector E is segmented and shaped according to the dimensions of the original images to obtain the corresponding ciphertext images.

4 Simulation result

Being able to completely encrypt and decrypt children's information and photos is the first requirement for practical applications. In the simulation experiment, three sets of images are used ("kid" with size 200 × 289 × 3, "information1" with size 300 × 152 × 3; "girl" with size 768 × 512 × 3, "information2" with size 300 × 174 × 3; and "boy" with size 768 × 512 × 3, "information3" with size 300 × 138 × 3), and they are encrypted and decrypted separately and in a hybrid manner. The simulation results are shown in Figures 4–7. From Figures 4–6, it can be seen that the scheme can successfully encrypt and decrypt children's information and photographs. As shown in Figure 7, it is also possible to securely encrypt and decrypt a large number of children's information and photographs if necessary. In other words, the proposed encryption and decryption scheme can perform both individual

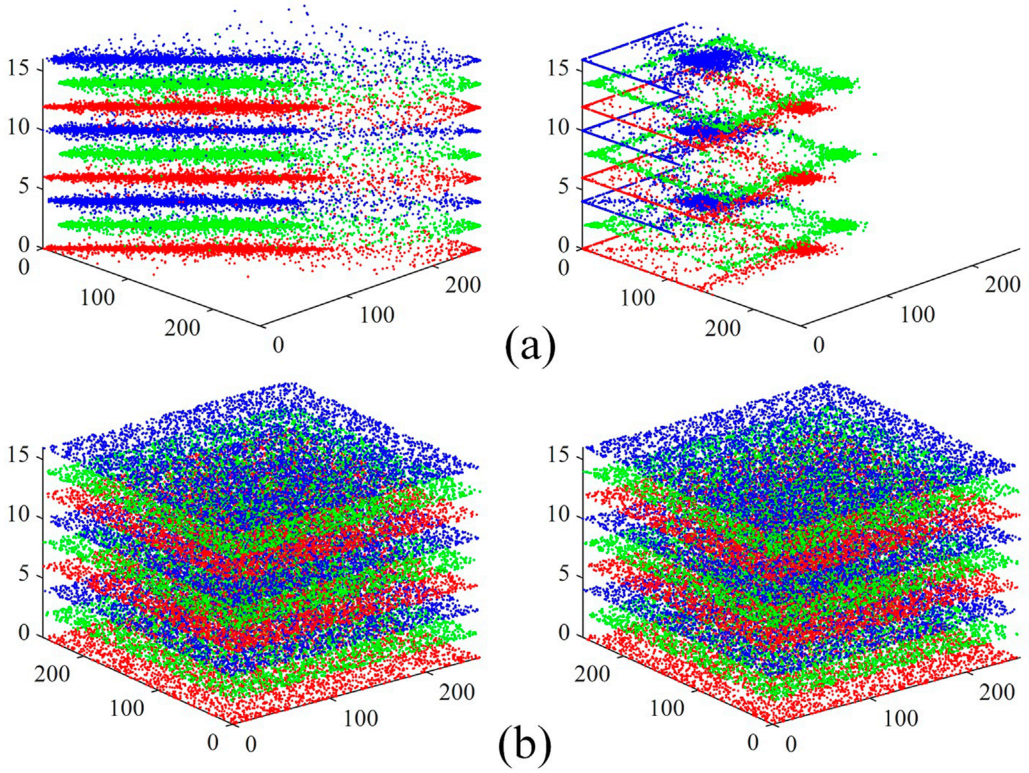


FIGURE 10
Correlation coordinate diagram of "kid" and "information1": **(a)** original images and **(b)** encryption images.

TABLE 3 Correlation coefficients of different images.

Images	Original images			Encryption images		
	Horizontal	Vertical	Diagonal	Horizontal	Vertical	Diagonal
Kid	0.9739	0.9776	0.9551	0.0015	0.0009	−0.0023
Information-1	0.8458	0.7476	0.6864	−0.0016	0.0018	−0.0008
Girl	0.9924	0.9879	0.9813	0.0019	0.0022	−0.0009
Information-2	0.7887	0.6612	0.6005	−0.0008	0.0016	−0.0012
Boy	0.9718	0.9638	0.9534	−0.0012	−0.0016	−0.0036
Information-3	0.7692	0.6208	0.5325	−0.0021	0.0013	−0.0024

processing and batch protection of children's information and photographs.

5 Performance tests

5.1 Key security

5.1.1 Key space

The size of the key space determines whether the encryption scheme can resist exhaustive attacks. Generally, when the key space reaches 2^{100}, it is considered to be capable of resisting exhaustive attacks, and the more the key space is, the better the scheme. In this encryption scheme, the key comprises two components: parameters related to the original images and those associated with hyperchaotic mapping. All the keys are tested one by one; the key space of parameters b and d is 10^{15}, and the key space of the remaining parameters is 10^{16}, so the total key space is $10^{(15\times 2+16\times 7)} = 10^{142} \approx 2^{471}$. The key space of different algorithms is shown in Table 2 [16, 40–42]. The key space test and comparison results indicate that the proposed encryption scheme has adequate capability to resist brute-force attacks.

TABLE 4 Information entropy of different images.

Images	Original images			Encryption images		
	R	G	B	R	G	B
Kid	7.3133	7.1356	7.2659	7.9972	7.9973	7.9972
Information1	4.2216	4.4069	4.6804	7.9962	7.9966	7.9963
Girl	7.8485	7.0813	7.2107	7.9996	7.9995	7.9996
Information2	4.0338	4.3841	4.7260	7.9973	7.9972	7.9972
Boy	7.5723	7.5670	7.5405	7.9995	7.9995	7.9995
Information3	3.1523	4.6179	5.2750	7.9959	7.9958	7.9959

5.1.2 Key sensitivity

The encryption scheme can be considered key-sensitive when a small error in the key can cause decryption failure on the decryption side. In the key sensitivity test, “kid” and “information1” are used as the test images. During the test, each key on the encryption side is kept constant, and the key $a = a + 10^{-16}$ on the decryption side. The decryption results are shown in Figure 8. The ciphertext image cannot be decrypted successfully with smaller parameter variations. As shown in Figure 8, a small error in the key causes the decryption to fail, verifying the key sensitivity of this scheme.

5.2 Statistical characterization

5.2.1 Histogram

A histogram can visually depict the strength of the pixels in the image. By comparing the histograms of the original image and the encrypted image, the ability of the encryption scheme to change the pixel values of the image can be verified. The histograms of “kid” and “information1” are shown in Figure 9. The histograms of the original images have distinct crests and varying distributions at each pixel level. The histograms of the encrypted images show an undifferentiated uniform distribution, which means that the pixel-level distribution of the original images is effectively changed and hidden by the encryption scheme.

5.2.2 Correlation

The property of local smoothing of the image determines a strong correlation between the adjacent pixels of the image, and the intensity of the correlation is measured by both the coordinate plot and the coefficient, which are shown in Figure 9 and Table 2, respectively. As shown in Figure 10, neighboring pixels of “kid” and “information1” are compactly distributed on a straight line with slope 1, which means that the neighboring pixels have the same or similar values. The adjacent pixels of the corresponding encrypted images are distributed throughout the coordinate space, and the values of the adjacent pixels are not correlated. As shown in Table 3, the correlation coefficients of the original images are large, while the correlation coefficients of the encrypted images are close to 0. The change in the correlation between the adjacent pixels of the image indicates that the encryption scheme effectively swaps the location and changes the values of the pixels, thus hiding the correlation characteristics of the original images.

5.2.3 Information entropy

Information entropy is used to test the statistical characteristics of an image. For an image, the higher the information entropy is, the more information it contains, and the more confusing the image is. The original images contain a certain amount of visual information, and their information entropy is a constant value. The information of the encrypted images is confusing, and the information entropy increases with a theoretical maximum value of 8 [43]. The information entropy test results for different images are shown in Table 4, and the information entropy test results for different algorithms are shown in Table 5 [35, 42, 44–46]. As shown in Table 4, compared to the original images, the information entropy of the encrypted images increases significantly and is close to the theoretical maximum. As shown in Table 5, the designed encryption scheme has some advantages in hiding the statistical features of the image data.

5.3 Anti-rolling edge test

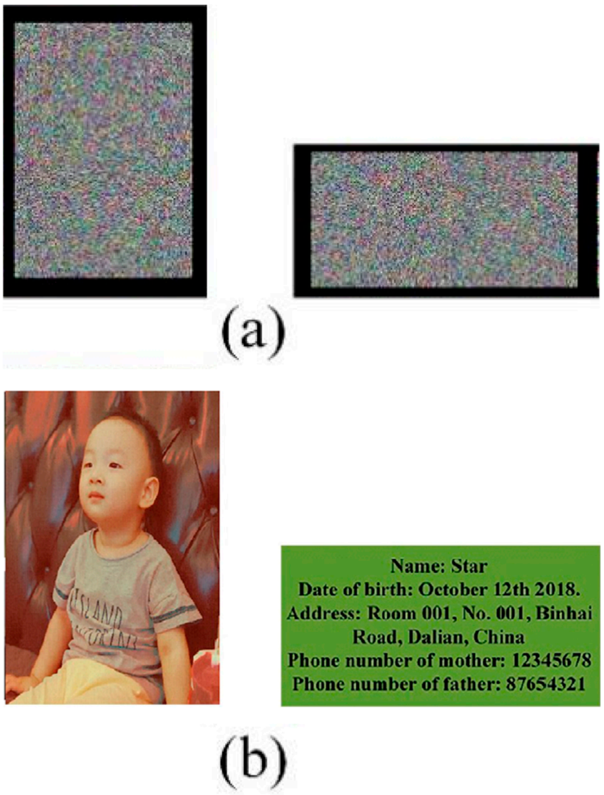
The encrypted images containing children’s information are printed on the clothes, and if the edge of the image rolls up as the clothes are used, then the edge information may be invalidated. When the edge information of the images is invalidated, the decryption effects of the encrypted images are shown in Figure 11. The edge of the original images is cropped by one circle, and the invalidated information accounts for 23.44%; the cropping effects are shown in Figure 10a. The visual effects of the damaged ciphertext images after being decrypted on the decryption side are shown in Figure 10b. As shown in Figure 11, the original information can still be recovered even if the child’s information and photo edges are rolled up to some extent.

5.4 Noise test

The cipher image is usually acquired using the photographing method, which produces noise on the cipher image. Salt and pepper noise and Gaussian noise are chosen to model the effect of noise on image restoration. Figure 12 shows the cipher image subjected to salt and pepper (S&P) noise with 0.1, 0.01, and 0.001 intensity and Gaussian noise with 0.001 intensity. The content of the image can be clearly seen at the reduction end, which in turn illustrates the feasibility of the scheme.

TABLE 5 Information entropy of different algorithms.

Algorithms	Image size	Encryption images (average)		
		R Channel	G channel	B Channel
Reference [44]	256 × 256	7.99720		
Reference [35]	256 × 256	7.99698		
Reference [42]	256 × 256	7.99705		
Reference [45]	256 × 256 × 3	7.9958	7.9950	7.9949
Reference [46]	256 × 256 × 3	7.9837	7.9916	7.9950
Proposed	200 × 289 × 3	7.9972	7.9973	7.9972



Name: Star
Date of birth: October 12th 2018.
Address: Room 001, No. 001, Binhai Road, Dalian, China
Phone number of mother: 12345678
Phone number of father: 87654321

FIGURE 11
Anti-rolling edge test results: **(a)** cipher image and **(b)** restored image.

5.5 Differential attack

Differential attack is a common method used by attackers to crack algorithms. The attacker randomly changes one pixel point of the plaintext image to get the cipher image and analyzes the difference between the two cipher images to crack the scheme.

In the differential attack test, the plaintext image is encrypted twice; the first time is normal encryption, and the cipher image is T_1; the second time, the attacker randomly changes one pixel point of the plaintext image to get the cipher image, and the cipher image is T_2. Since the scheme plaintext information is associated with the initial value of the chaotic system, randomly changing one pixel value of the plaintext image will again result in a different initial value of the chaotic system, and its chaotic sequence also changes. Therefore, the encrypted structure and content are changed, and the resulting encrypted image is also changed.

The difference between T_1 and T_2 is evaluated by the number of pixels change rate (NPCR) and the unified average changing intensity (UACI). The test results are shown in Table 6.

$$\begin{cases} \mathrm{NPCR}(T_1,T_2) = \dfrac{1}{MN}\displaystyle\sum_{i=1}^{M}\sum_{j=1}^{N}|\mathrm{Sign}(T_1(i,j) - T_2(i,j))| \times 100\% \\ \mathrm{UACI}(T_1,T_2) = \dfrac{1}{MN}\displaystyle\sum_{i=0}^{M}\sum_{j=0}^{N}\dfrac{|T_1(i,j) - T_2(i,j)|}{255-0} \times 100\% \end{cases},$$

where Sign (•) is a symbolic function.

5.6 Comparison with other state-of-the-art encryption schemes

In conclusion, the various performance metrics mentioned above are discussed to compare the proposed encryption algorithm with other state-of-the-art chaotic and non-chaotic encryption algorithms. Reference [46] and Reference [47] proposed chaotic encryption schemes. Reference [48] used the advanced encryption standard (AES) scheme. The comparison results are shown in Table 7.

6 Discussion and conclusion

6.1 Discussion

In recent years, significant progress has been made in visual information mapping techniques based on deep learning [49, 50]. The visual consistency of feature embedding has been optimized

Name: Star
Date of birth: October 12th 2018.
Address: Room 001, No. 001, Binhai Road, Dalian, China
Phone number of mother: 12345678
Phone number of father: 87654321

Name: Star
Date of birth: October 12th 2018.
Address: Room 001, No. 001, Binhai Road, Dalian, China
Phone number of mother: 12345678
Phone number of father: 87654321

Name: Star
Date of birth: October 12th 2018.
Address: Room 001, No. 001, Binhai Road, Dalian, China
Phone number of mother: 12345678
Phone number of father: 87654321

Name:Star
Date of birth:October 12th 2018.
Address: Room 001, No.001, Binhai Road, Dalian, China
Phone number of mother: 12345678
Phone number of father: 87654321

FIGURE 12
Noise test results: **(a)** S&P 0.1; **(b)** S&P 0.01; **(c)** S&P 0.001; **(d)** Gaussian noise 0.001.

TABLE 6 Test results of different images.

Images	NPCR (%)	UACI (%)
Kid	99.6012	33.4311
Girl	99.6114	33.4821
Boy	99.6241	33.4636
Average	99.6122	33.4589

TABLE 7 Comparison with other encryption schemes.

Process	Proposed	[46]	[47]	[48]
Key space	2^{471}	2^{326}	2^{448}	2^{128}
NPCR (%)	99.6122	99.6025	99.60	99.5650
UACI (%)	33.4589	33.4612	33.42	33.4675
Entropy	7.9972	7.9993	7.9993	7.9971

Name： Moon
Gender： Male
Date of birth： May 12th 2020
Address : Room 101 ,No.5 Zhongshan Street, Beijing,China
Cell-phone number of mother： 87873489
Cell-phone number of father： 85456932

FIGURE 13
Lost children information.

through the cascading attention mechanism, and the robustness of cross-modal information has been improved using adversarial generative networks [51, 52]. These techniques are better able to print child-protective cipher information on clothing in the future.

The existing solutions mainly conduct anti-edge curling and noise tests for image encryption on carriers such as clothing and schoolbags. However, in practical applications, children's information may be printed on more complex carriers (such as clothes with rough fabric textures and easily worn plastic labels). In the future, the decryption effect of encrypted images under extreme physical conditions (such as high temperature, water stains, and tensile deformation) can be further tested, and combined with image restoration algorithms (such as damaged area completion based on deep learning), the adaptability of the scheme to diverse carriers and environments can be enhanced.

6.2 Conclusion

An image encryption scheme is proposed in this paper for the protection and verification of missing children's information. First, the dynamical behavior of the hyperchaotic mapping used in the design of the encryption scheme is analyzed, and the analysis results prove that the hyperchaotic mapping is suitable for image-encryption design. Then, the pseudo-random sequences are used to swap the missing child image information with random length random positions, divided into row swap and column swap. Next, a selective XOR is used between the image sequence and the pseudo-random sequences. Finally, the effectiveness of the encryption scheme is verified by simulation. Considering that the missing child's information should be decrypted by a specific person, the security of the encryption scheme should also be guaranteed. The sensitivity to the key and the large key space guarantee the resistance of the encryption scheme to exhaustive attacks. Comparing the statistical characteristics of the data between the cipher images and the original images, the pixel-level distribution status of the original images, the correlation between the adjacent pixels, and the amount of information contained in the image are hidden or broken. Considering that children's clothes will have curled edges in the process of use, the image encryption scheme is tested against curled edges. The test results show that even if the missing child information has a certain degree of curled edges,

FIGURE 14
Encrypted image (represented by the blue labels) integrated into different areas of the garment.

it can be recovered. In summary, this scheme provides technical support for the protection and verification of missing children's information.

Moreover, this technique can be applied to prevent the missing of children. For example, the detailed information of children, along with those of their parents (Figure 13), can be encoded. The encrypted image will be attached to the clothes (Figure 14). In instances where children go missing, law enforcement agencies can employ specialized readers to decrypt the encrypted information, thereby facilitating accurate and expeditious contact with designated guardians. Moreover, the amalgamation of image encryption significantly amplifies the computational complexity faced by malicious entities attempting to breach the encryption, effectively impeding easy access to children's information and mitigating concerns regarding privacy breaches. It can enhance the probability of successfully locating missing children.

Data availability statement

The data used in this study is available from the corresponding author upon reasonable request.

Ethics statement

Written informed consent was obtained from the minor(s)' legal guardian/next of kin for the publication of any potentially identifiable images or data included in this article.

Author contributions

CT: Conceptualization, Methodology, Software, Writing – original draft. LN: Funding acquisition, Supervision, Writing – review and editing. RC: Validation, Formal analysis, Writing – review and editing.

Funding

The author(s) declare that financial support was received for the research and/or publication of this article. The research was supported by the Scientific Research Project of Dalian Polytechnic University (KJ20250095) and the Liaoning Provincial Department of Education Scientific Research Project (JYTMS20230410).

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.

Generative AI statement

The author(s) declare that no Generative AI was used in the creation of this manuscript.

Publisher's note

References

1. Cao Y, Tan L, Xu X, Li B. A universal image compression sensing–encryption algorithm based on DNA-triploid mutation. *Mathematics* (2024) 12(13):1990. doi:10.3390/math12131990

2. Tan L, Cao Y, Banerjee S, Mou J. Multi-medical image protection: compression-encryption scheme based on TLNN and mask cubes. *J Supercomputing* (2025) 81(1):96. doi:10.1007/s11227-024-06624-6

3. Liu Z, Li P, Cao Y, Mou J. A novel multimodal joint information encryption scheme based on multi-level confusion and hyperchaotic map. *Int J Mod Phys C* (2025). doi:10.1142/S012918312550038X

4. Zhang Z, Cao Y, Zhou N, Xu X, Mou J. Novel discrete initial-boosted tabu learning neuron: dynamical analysis, DSP implementation, and batch medical image encryption. *Appl Intelligence* (2025) 55(1):61. doi:10.1007/s10489-024-05918-9

5. Chu R, Zhang S, Gao X. A novel 3D image encryption based on the chaotic system and RNA crossover and mutation. *Front Phys* (2022) 10:1–14. doi:10.3389/fphy.2022.844966

6. Bi X, Shuai C, Liu B, Xiao B, Li W, Gao X. Privacy-preserving color image feature extraction by quaternion discrete orthogonal moments. *IEEE Trans Inf Forensics Security* (2022) 17:1655–68. doi:10.1109/tifs.2022.3170268

7. Mou J, Zhang Z, Banerjee S, Zhang Y. Combining Semi-tensor product compressed sensing and session keys for low-cost encryption of batch information in WBANs. *IEEE Internet Things J* (2024) 11(20):33565–76. doi:10.1109/jiot.2024.3429349

8. Yuan S, Chen D, Liu X, Zhou X. Optical encryption based on biometrics and single-pixel imaging with random orthogonal modulation. *Opt Commun* (2022) 522:128643. doi:10.1016/j.optcom.2022.128643

9. Mou J, Tan L, Cao Y, Zhou N, Zhang Y. Multi-face image compression encryption scheme combining extraction with STP-CS for face database. *IEEE Internet Things J* (2025) 12:19522–31. doi:10.1109/JIOT.2025.3541228

10. Liang W, Yang Y, Yang C, Hu Y, Xie S, Li KC, et al. PDPChain: a consortium blockchain-based privacy protection scheme for personal data. *IEEE Trans Reliability* (2022) 72(2):586–98. doi:10.1109/tr.2022.3190932

11. Liu Y, Hao X, Ren W, Xiong R, Zhu T, Choo KKR, et al. A blockchain-based decentralized, fair and authenticated information sharing scheme in zero trust internet-of-things. *IEEE Trans Comput* (2022) 72(2):501–12. doi:10.1109/tc.2022.3157996

12. Cai C, Cao Y, Mou J, Banerjee S, Sun B. A versatile image encryption scheme based on optical hologram technology and chess rules. *Int J Bifurcation Chaos* (2024) 34(15):2450185. doi:10.1142/s0218127424501852

13. Zhang Z, Mou J, Zhou N, Banerjee S, Cao Y. Multi-cube encryption scheme for multi-type images based on modified klotski game and hyperchaotic map. *Nonlinear Dyn* (2024) 112(7):5727–47. doi:10.1007/s11071-024-09292-6

14. Han Z, Cao Y, Banerjee S, Mou J. Hybrid image encryption scheme based on hyperchaotic map with spherical attractors. 34(3), 030503 (2025).

15. Yu F, He S, Yao W, Cai S, Xu Q. Bursting firings in memristive hopffeld neural network with image encryption and hardware implementation. *IEEE Trans Computer-Aided Des Integrated Circuits Syst* (2025) 1. doi:10.1109/tcad.2025.3567878

16. Yang F, An X, xiong L. A new discrete chaotic map application in image encryption algorithm. *Physica Scripta* (2022) 97(3):035202. doi:10.1088/1402-4896/ac4fd0

17. Ma T, Mou J, Yan H, Cao Y. A new class of hopfield neural network with double memristive synapses and its DSP implementation. *The Eur Phys J Plus* (2022) 137(10):1135. doi:10.1140/epjp/s13360-022-03353-8

18. Yu F, Su D, He S, Wu 吴 Y亦, Zhang 张 S善, Yin 尹 H挥. Resonant tunneling diode cellular neural network with memristor coupling and its application in police forensic digital image protection. *Chin Phys B* (2025) 34(5):050502. doi:10.1088/1674-1056/adb8bb

19. Ma Y, Mou J, Jahanshahi H, Abdulhameed A, Bi X. Design and DSP implementation of a hyperchaotic map with infinite coexisting attractors and intermittent chaos based on a novel locally active memcapacitor. *Chaos, Solitons and Fractals* (2023) 173:113708. doi:10.1016/j.chaos.2023.113708

20. Ma T, Mou J, Banerjee S, Cao H. Analysis of the functional behavior of fractional-order discrete neuron under electromagnetic radiation. *Chaos, Solitons and Fractals* (2023) 176:114113. doi:10.1016/j.chaos.2023.114113

21. Chen Y, Cao Y, Mou J, Sun B, Banerjee S. A simple photosensitive circuit based on a mutator for emulating memristor, memcapacitor, and meminductor: light illumination effects on dynamical behaviors. *Int J Bifurcation Chaos* (2024) 34(6):2450069. doi:10.1142/s021812742450069x

22. Wang X, Xu X, Sun K, Jiang Z, Li M, Wen J. A color image encryption and hiding algorithm based on hyperchaotic system and discrete cosine transform. *Nonlinear Dyn* (2023) 111(15):14513–36. doi:10.1007/s11071-023-08538-z

23. Toktas F, Erkan U, Yetgin Z. Cross-channel color image encryption through 2D hyperchaotic hybrid map of optimization test functions. *Expert Syst Appl* (2024) 249:123583. doi:10.1016/j.eswa.2024.123583

24. Gao X, Mou J, Banerjee S, Cao Y, Xiong L, Chen X. An effective multiple-image encryption algorithm based on 3D cube and hyperchaotic map. *J King Saud Univ - Computer Inf Sci* (2022) 34(4):1535–51. doi:10.1016/j.jksuci.2022.01.017

25. Wang X, Mou J, Cao Y, Jahanshahi H. Modeling and analysis of cellular neural networks based on memcapacitor. *Int J Bifurcation Chaos* (2025) 35. doi:10.1142/S0218127425300101

26. Jun M, Cao H, Zhou N, Cao Y. An FHN-HR neuron network coupled with a novel locally active memristor and its DSP implementation. *IEEE Trans Cybernetics* (2024) 54(12):7333–42. doi:10.1109/TCYB.2024.3471644

27. Jun M, Zhang Z, Zhou N, Zhang Y, Cao Y. Mosaic tracking: lightweight batch video frame awareness multi-target encryption scheme based on a novel discrete tabu learning neuron and YoloV5. *IEEE Internet Things J* (2024). doi:10.1109/JIOT.2024.3482289

28. Cao H, Cao Y, Lei Q, Mou J (2025). Dynamical analysis, multi-cavity control and DSP implementation of a novel memristive autapse neuron model emulating brain behaviors. *Chaos, Solitons and Fractals*, 191, 115857. doi:10.1016/j.chaos.2024.115857

29. Shi F, Cao Y, Xu X, Mou J. A novel memristor-coupled discrete neural network with multi-stability and multiple state transitions. *The Eur Phys J Spec Top* (2025). doi:10.1140/epjs/s11734-024-01440-8

30. Yu F, Wu C, Xu S, Yao W, Xu C, Cai S, et al. Color video encryption transmission in IoT based on memristive hopfield neural network. *Signal Image Video Process.* (2025) 19(77):77. doi:10.1007/s11760-024-03697-x

31. Gan Z, Chai X, Bi J, Chen X. Content-adaptive image compression and encryption *via* optimized compressive sensing with double random phase encoding driven by chaos. *Complex & Intell Syst* (2022) 8:2291–309. doi:10.1007/s40747-022-00644-6

32. Aburturab MR, Alfalou A. Multiple color image fusion, compression, and encryption using compressive sensing, chaotic-biometric keys, and optical fractional fourier transform. *Opt & Laser Technology* (2022) 151:108071. doi:10.1016/j.optlastec.2022.108071

33. Wang X, Liu C, Jiang D. A novel triple-image encryption and hiding algorithm based on chaos, compressive sensing and 3D DCT. *Inf Sci* (2021) 574:505–27. doi:10.1016/j.ins.2021.06.032

34. Zhang J, Yang D, Ma R, Shi Y. Multi-image and color image encryption *via* multi-slice ptychographic encoding. *Opt Commun* (2021) 485:126762. doi:10.1016/j.optcom.2021.126762

35. Yan X, Wang X, Xian Y. Chaotic image encryption algorithm based on arithmetic sequence scrambling model and DNA encoding operation. *Multimedia Tools Appl* (2021) 80(7):10949–83. doi:10.1007/s11042-020-10218-8

36. Wang X, Su Y. Image encryption based on compressed sensing and DNA encoding. *Signal Processing: Image Commun* (2021) 95:116246. doi:10.1016/j.image.2021.116246

37. Ghanbari-Ghalehjoughi H, Eslami M, Ahmadi-Kandjani S, Ghanbari-Ghalehjoughi M, Yu Z. Multiple layer encryption and steganography *via* multi-channel ghost imaging. *Opt Lasers Eng* (2020) 134:106227. doi:10.1016/j.optlaseng.2020.106227

38. Yang Y-G, Wang B-P, Yang Y-L, Zhou Y-H, Shi W-M, Liao X. Visually meaningful image encryption based on universal embedding model. *Inf Sci* (2021) 562:304–24. doi:10.1016/j.ins.2021.01.041

39. Hua Z, Zhang K, Li Y, Zhou Y. Visually secure image encryption using adaptive-thresholding sparsification and parallel compressive sensing. *Signal Process.* (2021) 183:107998. doi:10.1016/j.sigpro.2021.107998

40. Zhu S, Deng X, Zhang W, Zhu C. Image encryption scheme based on newly designed chaotic map and parallel DNA coding. *Mathematics* (2023) 11(1):231. doi:10.3390/math11010231

41. Zhu S, Deng X, Zhang W, Zhu C. Secure image encryption scheme based on a new robust chaotic map and strong S-box. *Mathematics Comput Simulation* (2023) 207:322–46. doi:10.1016/j.matcom.2022.12.025

42. Wu Y, Zhou Y, Saveriades G, Agaian S, Noonan JP, Natarajan P. Local shannon entropy measure with statistical tests for image randomness. *Inf Sci* (2013) 222:323–42. doi:10.1016/j.ins.2012.07.049

43. Kaur G, Agarwal R, Patidar V. Color image encryption system using combination of robust chaos and chaotic order fractional hartley transformation. *J King Saud Univ - Computer Inf Sci* (2021) 34:5883–97. doi:10.1016/j.jksuci.2021.03.007

44. Lu Q, Yu L, Zhu C. Symmetric image encryption algorithm based on a new product trigonometric chaotic map. *Symmetry* (2022) 14(2):373. doi:10.3390/sym14020373

45. Kaur G, Agarwal R, Patidar V. Color image encryption scheme based on fractional hartley transform and chaotic substitution–permutation. *The Vis Computer* (2021) 38(3):1027–50. doi:10.1007/s00371-021-02066-w

46. Li S-Y, Gai Y, Shih K-C, Chen C-S. An efficient image encryption algorithm based on innovative DES structure and hyperchaotic keys. *IEEE Trans Circuits Syst Regular Pap* (2023) 70(10):4103–11. doi:10.1109/tcsi.2023.3296693

47. Zhou S, Wang X, Zhang Y. Novel image encryption scheme based on chaotic signals with finite-precision error. *Inf Sci* (2023) 621:782–98. doi:10.1016/j.ins.2022.11.104

48. Yi G, Cao Z. An algorithm of image encryption based on AES & rossler hyperchaotic modeling. *Mobile Networks Appl* (2023) 29:1451–9. doi:10.1007/s11036-023-02216-5

49. Wu Z, Sun C, Xuan H, Yan Y. Deep stereo video inpainting. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2023). p. 5693–702.

50. Wu Z, Sun C, Xuan H, Liu G, Yan Y. WaveFormer: wavelet transformer for noise-robust video inpainting. *Proc AAAI Conf Artif Intelligence* (2024) 38(6):6180–8. doi:10.1609/aaai.v38i6.28435

51. Zhang W, Li Z, Li G, Zhuang P, Hou G, Zhang Q, et al. GACNet: generate adversarial-driven cross-aware network for hyperspectral wheat variety identification. *IEEE Trans Geosci Remote Sensing* (2023) 62:1–14. doi:10.1109/tgrs.2023.3347745

52. Gao X, Sun B, Cao Y, Banerjee S, Mou J. A color image encryption algorithm based on hyperchaotic map and DNA mutation. *Chin Phys B* (2023) 32(3):030501. doi:10.1088/1674-1056/ac8cdf

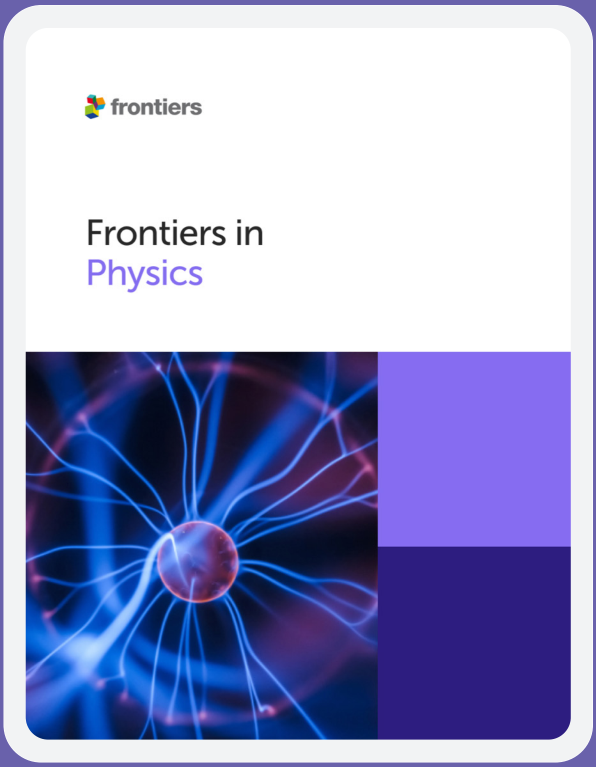
Frontiers in
Physics

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

Frontiers in
Physics

Frontiers

Avenue du Tribunal-Fédéral 34
1005 Lausanne, Switzerland
frontiersin.org

Contact us

+41 (0)21 510 17 00
frontiersin.org/about/contact