

Review

Advances in Zeroing Neural Networks: Bio-Inspired Structures, Performance Enhancements, and Applications

Yufei Wang ¹, Cheng Hua ¹ and Ameer Hamza Khan ^{2,3,*}

¹ College of Computer Science and Engineering, Jishou University, Jishou 416000, China; yufeiwang@stu.jsu.edu.cn (Y.W.); chenghua@jsu.edu.cn (C.H.)

² Smart City Research Institute (SCRI), Hong Kong Polytechnic University, Kowloon, Hong Kong

³ Department of Land Surveying and Geo-Informatics (LSGI), Hong Kong Polytechnic University, Kowloon, Hong Kong

* Correspondence: ameer-hamz.khan@polyu.edu.hk

Abstract: Zeroing neural networks (ZNN), as a specialized class of bio-inspired neural networks, emulate the adaptive mechanisms of biological systems, allowing for continuous adjustments in response to external variations. Compared to traditional numerical methods and common neural networks (such as gradient-based and recurrent neural networks), this adaptive capability enables the ZNN to rapidly and accurately solve time-varying problems. By leveraging dynamic zeroing error functions, the ZNN exhibits distinct advantages in addressing complex time-varying challenges, including matrix inversion, nonlinear equation solving, and quadratic optimization. This paper provides a comprehensive review of the evolution of ZNN model formulations, with a particular focus on single-integral and double-integral structures. Additionally, we systematically examine existing nonlinear activation functions, which play a crucial role in determining the convergence speed and noise robustness of ZNN models. Finally, we explore the diverse applications of ZNN models across various domains, including robot path planning, motion control, multi-agent coordination, and chaotic system regulation.

Keywords: zeroing neural network (ZNN); noise-tolerant; time-varying problems; convergence; applications



Academic Editor: Heming Jia

Received: 27 March 2025

Revised: 22 April 2025

Accepted: 27 April 2025

Published: 29 April 2025

Citation: Wang, Y.; Hua, C.; Khan, A.H. Advances in Zeroing Neural Networks: Bio-Inspired Structures, Performance Enhancements, and Applications. *Biomimetics* **2025**, *10*, 279. <https://doi.org/10.3390/biomimetics10050279>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Neural networks, recognized as versatile and highly efficient computational models, have found extensive applications across diverse fields [1–7], especially in the modeling, prediction, and optimization of complex problems [4,8–13]. By mimicking the architecture and operational principles of biological neural systems, these networks are adept at uncovering hidden patterns within large datasets. Consequently, they have become indispensable tools in tasks such as decision support, pattern recognition, and numerous other data-driven applications [14–18].

The development of biomimetic neural networks has been profoundly influenced by the recurrent neural network (RNN) model proposed by Hopfield [19], a distinguished member of the U.S. National Academy of Sciences. His model represents neural networks as graph structures composed of nodes (neurons) and connections (weights), exerting a significant impact on the field of computational neuroscience. Each node corresponds to a neuron, while the connections encode interactions between neurons. Despite its relatively simple architecture, the Hopfield network exhibits remarkable dynamical properties, earning its recognition as one of the foundational models in neural network research.

Unlike traditional optimization algorithms that rely on gradient information for problem-solving, some biomimetic algorithms have been proposed to address non-convex optimization problems [20–27]. These include the Egret Swarm Optimization Algorithm [28–30], the Cuckoo Search Algorithm [31], the Harmony Search Algorithm [32], the Grey Wolf Optimizer and Multi-Strategy Optimization Methods [33], the Whale Optimization Algorithm [34], the Harris Hawks Algorithm (VEH) [35,36], and Ant Colony Optimization [37]. Based on the theoretical framework of the RNN, the zeroing neural network (ZNN) is a biologically inspired subclass of RNN systematically proposed by Zhang et al. in 2002 [38]. It aims to emulate the adaptive behavior of biological systems in response to external changes and is specifically designed for high-precision and robust solutions to optimization and time-varying problems. Unlike traditional RNNs that rely on energy function-based update mechanisms, the ZNN adopts a neurodynamic approach by constructing an error-monitoring function, enabling the system states to dynamically converge toward a zero-error trajectory. This mechanism, known as the “error-zeroing mechanism”, essentially mimics the homeostatic regulation process in biological neurons, where negative feedback continuously corrects deviations from the target, ensuring that the system state approaches a predefined zero-error point.

Compared with conventional methods for solving time-varying problems, such as sliding mode control, adaptive control, or numerical integrators, the ZNN demonstrates superior computational efficiency, while maintaining high precision and strong robustness. It organically integrates the adaptive nature of neural networks with the dynamic regulation strengths of control theory, exhibiting unique advantages in handling time-varying problems. As such, the ZNN occupies an irreplaceable and significant position within the neural network paradigm.

Initially, researchers applied the ZNN in the real-number domain to address the time-varying matrix inversion (TVMI) problem [39–42]. Unlike traditional neural networks (for example, gradient-based optimization techniques in neural networks, including gradient neural networks (GNN) and RNN). The ZNN has also been applied to complex matrix inversion tasks, including time-varying complex matrix inversion (TVCMI) [43–46] and time-varying complex matrix pseudoinversion (TVCMP) [47,48]. In addition, the ZNN has also been cited for solving time-varying equations, such as time-varying overdetermined linear systems (TVOLS) [49], time-varying nonlinear equations (TVNE) [50,51], time-varying Stein matrix equations (TVSME), and the time-varying Sylvester matrix equation (TVSME2) [52,53]. In the field of optimization, the ZNN can also be applied to handle such problems, including time-varying nonlinear minimization (TVNM) [54], nonconvex nonlinear programming (NNP) [55–58], multi-objective optimization (MOO) [59], time-varying quadratic optimization (TVQO) [60–64], and time-varying nonlinear optimization (TVNO) [65,66].

The ZNN enhances its capacity to address time-varying problems primarily by modifying activation functions and integration methods [67]. Notable examples include the Li activation function [68], the FAESAF and FASSAF activation functions [69], the multiplication-based sigmoid activation function [70], the NF1 and NF2 activation functions [71], the symbolic quadratic activation function [72], as well as the hyperbolic sine activation function [73], and the multiply-accumulate activation function [74]. These functions significantly impact the network’s nonlinear characteristics and convergence rate. The ZNN can also be classified into single-integration and double-integration types based on the integration method. The single-integral ZNN [75] corrects errors by incorporating time derivatives, thereby enabling it to tackle relatively simple time-varying problems, such as motion planning and path tracking. In contrast, the double-integration ZNN incorporates the second-order derivative of the error [76], improving the system’s adaptability

to highly dynamic and nonlinear problems. It is particularly effective for more complex time-varying control tasks, such as multi-agent systems and chaotic control [77]. By combining these activation functions and integration methods, the ZNN can also efficiently address dynamic time-varying problems through the flexible setting of both variable and fixed parameters [78]. In many practical applications, variable parameters empower the ZNN to adjust its output in real time, enabling it to adapt to dynamic system changes, while fixed parameters establish a stable framework for addressing static or nearly static problems. By carefully designing these parameters, the ZNN can effectively manage more complex and multidimensional time-varying systems [79].

Researchers have extended the application of the ZNN to several practical fields [80]. In robotics, the ZNN has been employed for robotic arm path tracking [81,82] and motion planning [83–85], enabling high-precision movements in complex environments through the real-time dynamic adjustment of control strategies [86,87]. Additionally, it has been utilized to optimize robotic paths for improved efficiency. In multi-agent systems, the ZNN facilitates the coordination of multiple agents [88,89], achieving group collaboration and synchronization, particularly in addressing global optimization challenges in complex tasks. In the field of chaotic control, the ZNN dynamically adjusts system parameters in real time to mitigate chaotic phenomena and ensure system stability. For signal processing, the ZNN has demonstrated effectiveness in noise suppression and signal recovery, particularly in image and speech processing, where it efficiently removes noise and restores the quality of original signals. Furthermore, the ZNN finds extensive applications in engineering optimization problems, such as in automatic control systems, where it supports the real-time adjustment and optimization of control strategies, ensuring stable and efficient system operation.

Therefore, this paper presents a comprehensive review of the development of ZNN models and their diverse applications. The paper's framework is illustrated in Figure 1, and the remaining sections are organized as follows:

Section 2 offers a detailed review of the structural development of the ZNN model. Section 3 examines nonlinear activation functions and time-varying parameters, with a focus on their roles in enhancing convergence and improving noise tolerance. Section 4 presents a summary of the practical applications of the ZNN in real-world domains. Section 5 concludes the paper, summarizing key findings and suggesting potential directions for future research.

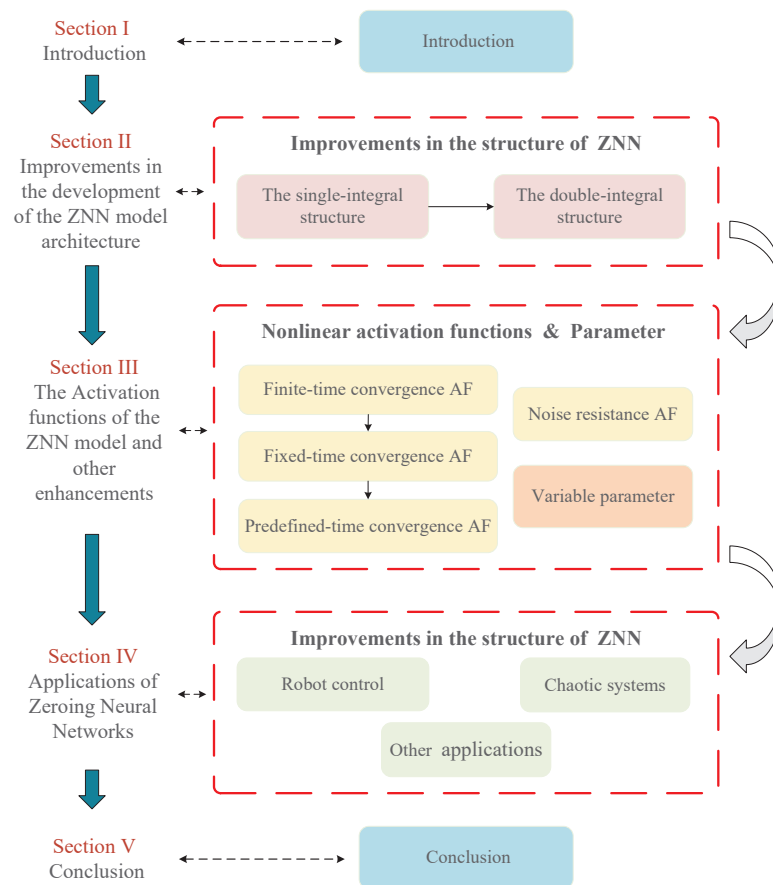


Figure 1. Block diagram of the structure in this paper.

2. Improvement of Zeroing Neural Network Model Structures

This section offers a comprehensive review of the advancements in ZNN models over the past decade, with a primary focus on the design of model structures. It highlights significant research achievements across diverse problem domains and establishes a robust theoretical and practical framework for further analysis.

2.1. Original Zeroing Neural Network Model

The gradient neural network (GNN) method was initially introduced by researchers to solve optimization problems [90]. The GNN is a type of neural network based on gradient optimization principles, specifically designed to address a wide range of optimization challenges and the resolution of dynamic system problems. The central concept behind the GNN involves constructing a performance index and optimizing it through gradient descent, thereby providing a solution to the problem at hand. A key feature of the GNN [91] is the definition of a scalar performance index $J(x)$, which quantifies the deviation of the system state x from the desired target state. A common expression for the performance index is

$$J(x) = \frac{1}{2} \|f(x)\|^2.$$

Here, $f(x)$ denotes the nonlinear function or constraint equation to be solved, typically framed as a root-finding problem where $f(x) = 0$. The fundamental design equation of the GNN is expressed as

$$\dot{v}(t) = -\nabla J(x).$$

The GNN can be expressed as

$$\dot{v}(t) = -f(x)^T \nabla f(x).$$

As the scale of the problems increased, researchers observed that applying the GNN often resulted in significant residual errors. To address this limitation, Zhang et al. proposed the ZNN, specifically designed to accurately solve time-varying scientific computing problems. Since its introduction, the ZNN has been widely applied across various domains [92]. In contrast to the GNN, which relies on optimizing a performance index, the ZNN directly constructs an error function $E(t)$, such as

$$E(t) = B(t)Y(t) - I. \quad (1)$$

The primary objective of the ZNN is to regulate the network dynamics such that the $E(t)$ asymptotically approaches zero over time.

The design equation of the ZNN is given by

$$\dot{E}(t) = -\gamma E(t), \quad (2)$$

where the parameter γ governs the decay rate of the error function. A larger value can accelerate convergence but may increase sensitivity to noise and system stiffness, whereas a smaller value results in slower but smoother convergence. Moderately increasing γ can improve accuracy; however, it often requires a smaller step size, leading to increased computational cost. Therefore, γ should be carefully selected based on simulation requirements and available hardware resources.

In contrast to the traditional GNN, the ZNN offers significant advantages in addressing time-varying parameter problems. The ZNN tracks the solution trajectory of time-varying systems in real time by incorporating the time derivative of residual errors, leading to faster convergence and improved stability. It demonstrates strong robustness against noise and disturbances and eliminates the need for iterative weight updates, thereby reducing computational complexity and enhancing real-time adaptability. Unlike the objective function optimization approach employed by the GNN, the ZNN controls the system by minimizing errors and dynamically adjusting them to accommodate system changes. This enables the ZNN to excel in managing time-varying problems, precisely controlling errors, and effectively mitigating the vanishing or exploding gradient issues commonly encountered in the GNN. As a result, the ZNN is better suited for handling long-term dependencies and dynamic variations.

2.2. Integration-Enhanced Zeroing Neural Network

To enhance the model's noise robustness, Jin et al. (2015) first proposed the integration enhanced zeroing neural network (IEZNN) [93], building upon the original ZNN. By incorporating a single integral term, the IEZNN improves the network's stability, convergence, and ability to suppress noise while effectively handling time-varying systems. In the original ZNN, the error function is typically used to measure the deviation between the system's output and the desired result. In contrast, the error function in the IEZNN not only depends on the current error but also integrates past errors, enabling smoother dynamic transitions. This approach mitigates the instability caused by instantaneous error fluctuations, particularly when addressing time-varying and uncertain problems. The IEZNN

model controls the evolution of the error by incorporating the single integral term. The design equation can be expressed as

$$\dot{E}(t) = -\gamma E(t) - \lambda_1 \int_0^t E(\tau) d\tau, \quad (3)$$

where $\gamma > 0$ and $\lambda_1 > 0$ are convergence parameters. This equation ensures that the error decreases progressively over time, eventually converging to zero. The IEZNN is an implicit dynamic system that considers not only the current state error but also integrates past error information. This approach enhances the system's stability, especially when operating in time-varying environments. The inclusion of the single-integral term enhances the robustness of the IEZNN, particularly in the presence of noise and disturbances. By mitigating instantaneous error fluctuations, the IEZNN improves its ability to handle uncertainty and external disturbances, making it well-suited for real-time computation in dynamic environments. The network effectively tracks time-varying matrices and computes their values, ensuring smooth convergence based on matrix value errors. This capability is especially critical when noise interference is significant, as the IEZNN maintains high computational accuracy, particularly when solving noisy time-varying Lyapunov equations (TVLE) [94], Liao *et al.* combined nonlinear activation functions with integral terms to propose a unified design formula for the zeroing neural dynamics (ZND). Building on this formula, they introduced the bounded zeroing neural dynamics (BZND) model. First, the error function is defined as

$$E(t) = A(t)^T Z(t) + Z(t) A(t) + B(t).$$

The design formula for ZND is

$$\dot{E}(t) = -\lambda_1 \mathcal{F}_1(E(t)) - \gamma \mathcal{F}_2(E(t)) - \lambda_1 \int_0^t \mathcal{F}_1(E(\iota)) d\iota. \quad (4)$$

Here, $\gamma \in (0, +\infty)$ and $\lambda_1 \in (0, +\infty)$ are scaling factors that adjust the convergence rate. The nonlinear activation function arrays $\mathcal{F}_1(\cdot)$ and $\mathcal{F}_2(\cdot)$ play a pivotal role in the dynamic process of the model. Under noisy conditions, the BZND model, represented by Equations (2) and (4), can be reformulated as

$$\begin{aligned} A(t)^T \dot{Z}(t) + \dot{Z}(t) A(t) &= -\dot{A}(t)^T Z(t) - Z(t) \dot{A}(t) - \dot{B}(t) \\ &- \gamma \mathcal{F}_1(A(t)^T Z(t) + Z(t) A(t) + B(t)) \\ &- \lambda_1 \mathcal{F}_2(A(t)^T Z(t) + Z(t) A(t) + B(t)) \\ &- \gamma \int_0^t \mathcal{F}_1(A(\iota)^T Z(\iota) + Z(\iota) A(\iota) + B(\iota)) d\iota + v(t). \end{aligned}$$

Lei constructed a model based on the IEZNN design formula to address the TVSE problem [95], and the error monitoring function is

$$E(t) = L(t)Z(t) - Z(t)F(t) + G(t).$$

Here, $L(t)$, $F(t)$, and $G(t)$ are given matrices, while $Z(t)$ is the unknown time-varying matrix to be determined. The design process for the noise-resistant integrated enhanced zeroing neural network (NIEZNN) model is outlined below.

The design formula, as presented in Equation (4), is employed to solve the TVSE. To further enhance the model's anti-interference capability, the NIEZNN model is extended

to incorporate additional random noise, resulting in the noise-augmented NIEZNN model. The extended model is expressed as follows:

$$\begin{aligned} L(t)\dot{Z}(t) - \dot{Z}(t)F(t) &= Z(t)\dot{F}(t) - \dot{L}(t)Z(t) - \dot{G}(t) \\ &- \gamma \mathcal{F}_1(L(t)Z(t) - Z(t)F(t) + G(t)) \\ &- \lambda_1 \mathcal{F}_2(L(t)Z(t) - Z(t)F(t) + G(t)) \\ &- \gamma \int_0^t \mathcal{F}_1(L(\iota)Z(\iota) - Z(\iota)F(\iota) + G(\iota))d\iota + v(t). \end{aligned} \quad (5)$$

This model exhibits exceptional performance in solving TVSE, particularly demonstrating significant robustness and noise resilience across a range of noisy environments. Furthermore, when applied to time-varying problems, especially in TVQO [96,97], and other time-varying issues under noisy conditions [93], the IEZNN outperforms the traditional ZNN model, offering superior robustness, noise resistance, and computational accuracy.

2.3. Design of the Double Integral-Enhanced Zeroing Neural Network Model

In scientific computing, tasks such as TVMI, solving linear equations, and other similar problems frequently encounter noise interference, including constant noise, linear noise, and random noise. While traditional ZNN models and the IEZNN can suppress certain types of noise, they remain susceptible to computational inaccuracies when faced with linear or more complex noise forms. To address this limitation, it is essential to incorporate a double integral feedback mechanism, which further mitigates long-term bias and enhances the performance of the model.

The double integral term improves the system's ability to detect and address error accumulation. This feedback mechanism accelerates error correction, thereby enhancing the network's convergence rate. Through double integral control, the system's error function becomes sensitive not only to the current error (instantaneous feedback) and accumulated historical errors (single integral feedback), but also adjusts for more complex error accumulation patterns (double integral feedback). This structure is further elaborated in the article [98]. The multi-level feedback mechanism strengthens the network's dynamic stability, enhancing its robustness in complex environments.

The design formula for the double integral enhanced zeroing neural network (DIEZNN) is given as follows:

$$\begin{aligned} \dot{E}(t) &= -\gamma E(t) - \lambda_1 \int_0^t E(\iota)d\iota - \\ &\lambda_2 \int_0^t \int_0^\iota E(\eta)d\eta d\iota, \end{aligned} \quad (6)$$

where $\gamma > 0$ and $\lambda_1 > 0$, $\lambda_2 > 0$ are design parameters. The first integral term $\int_0^t E(\iota) d\iota$ compensates for global error accumulation. The second integral term $\int_0^t \int_0^\iota E(\eta) d\eta d\iota$ addresses the deeper correction of the error's changing trend.

The introduction of double integrals effectively addresses such problems. Liao constructed a DIEZNN model based on a novel integral design formula, which inherently possesses linear noise tolerance [99]. To monitor the TVMI problem, the error function is designed in the same manner as in Equation (1). In this context, $x(t)$ represents the system state that needs to be solved. Although the IIEZNN model can suppress noise to some extent, it still exhibits limitations when handling linear noise. Therefore, a new model is

required to address the presence of linear noise. Liao *et al.* derived the design formula for the DIEZNN model as follows:

$$\begin{aligned}\dot{E}(t) = & -\gamma E(t) - \lambda_1 \int_0^t E(\iota) d\iota - \\ & \lambda_2 \int_0^t \int_0^\iota E(\eta) d\eta d\iota + v(t).\end{aligned}\quad (7)$$

Further, the DIEZNN model is as follows:

$$\begin{aligned}B(t)\dot{Y}(t) = & -\dot{B}(t)Y(t) - \gamma(B(t)Y(t) - I) \\ & - \lambda_1 \int_0^t (B(t)Y(t) - I) d\iota \\ & - \lambda_2 \int_0^t \int_0^\iota (B(\eta)Y(\eta) - I) d\eta d\iota + v(t).\end{aligned}$$

The article conducted two simulation case studies with varying matrix dimensions and linear noise. Both the theoretical proof and the simulation examples thoroughly demonstrate the inherent linear noise suppression capability of the DIEZNN model.

The double-integral structure possesses a stronger cumulative filtering capability, effectively attenuating both high- and low-frequency noise compared to the single-integral model. In control theory, the integration operation inherently exhibits low-pass filtering characteristics. First-order integration can suppress high-frequency disturbances but is limited in mitigating slowly varying noise. By introducing second-order integration, the system gains enhanced temporal smoothing ability, enabling more accurate extraction of the true error.

This structure delays the impact of instantaneous noise, suppresses error propagation, and significantly enhances the robustness and stability of the system. To validate the design motivation, this paper includes a comparative example of the IEZNN and DIEZNN under linear noise conditions. Figure 2 clearly demonstrates the superiority of the DIEZNN in terms of error convergence and noise resistance.

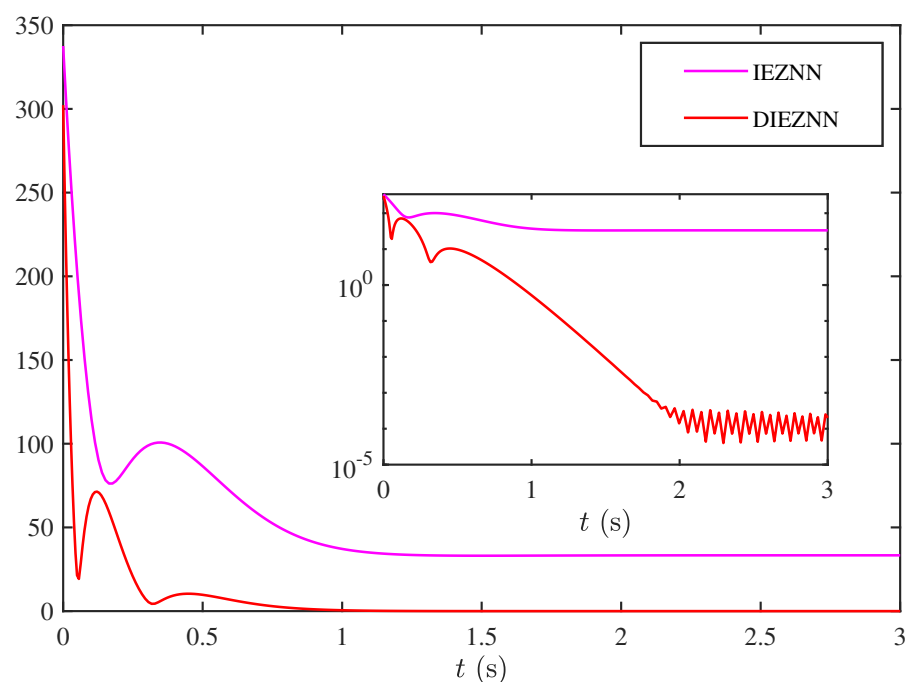


Figure 2. Real-time error plots of IEZNN and DIEZNN under linear noise conditions.

The DIEZNN model, by introducing a proportional-integral-double integral control mechanism, demonstrates significant advantages in solving dynamic computational problems such as TVMI [100], time-varying linear equations, MOO, embedded real-time computation, and control. With its exceptional noise resistance, rapid convergence, and adaptability to various environments, the DIEZNN offers an efficient and reliable solution for dynamic system modeling, control, and optimization. It has contributed to technological advancements and broadened the scope of applications in the field of dynamic computation.

Therefore, the improvements in the ZNN model structure can be summarized as follows: Through the iterative evolution from the traditional ZNN to the enhanced IEZNN, and then to the DIEZNN, these advancements have significantly improved the model's robustness to noise and its interference resistance. This progression has enabled the model to be effectively applied in complex multi-noise scenarios and laid the foundation for the further development of subsequent models.

This paper discusses the single-integral and double-integral ZNN models. The single-integral model improves both stability and convergence. The double-integral model, by incorporating a dual-feedback mechanism, enhances noise resistance and accelerates convergence. Although the t-fold integral model could potentially further improve robustness or trajectory smoothness, its increased complexity introduces a higher computational burden, which may lead to response delays and numerical stability issues in real-time systems. Consequently, this model is not considered in this paper.

3. Activation Functions of Zeroing Neural Network Model and Other Enhancements

Although optimizations based on model architectures have significantly enhanced the robustness of neural networks in noisy environments, the improvement in model convergence speed still faces certain limitations. Consequently, researchers have shifted their focus to optimizing activation functions, with the goal of further enhancing the model's convergence performance and computational efficiency through the design and introduction of more effective activation functions.

3.1. Nonlinear Activation Functions with Enhanced Convergence Properties

From the perspective of convergence speed, three common types of convergence can be distinguished: finite-time convergence, fixed-time convergence, and predefined-time convergence. These types all involve the rate at which system errors converge, but they differ in their specific characteristics. Finite-time convergence refers to a dynamic system's ability to reach its target state or ideal solution within a finite time, typically with the target solution being zero or sufficiently close to zero. Fixed-time convergence refers to a system behavior that ensures the system state converges to the equilibrium point within a finite time, with the convergence time being independent of the initial conditions. However, the actual convergence time is only guaranteed to have an upper bound, and it cannot be explicitly predetermined. In contrast, preset-time convergence describes a framework in which the convergence time is determined during the design phase. Unlike fixed-time convergence, preset-time convergence ensures that the system converges within a user-specified time frame, with the convergence time being adjustable to meet design requirements, thus offering stronger guarantees in time control than fixed-time convergence.

A large number of activation functions have been proposed to accelerate convergence. Finite-time convergence is primarily grounded in Lyapunov stability theory. By constructing an appropriate Lyapunov function, it has been demonstrated that the error or objective function can converge to zero within a finite time. Nonlinear activation functions play

a pivotal role in the design of neural networks that achieve finite-time convergence and are extensively utilized in numerous neural network models endowed with finite-time convergence properties [51]. These activation functions significantly improve convergence by altering both the rate and direction of error reduction.

In the literature [101], Xiao constructed a finite-time convergence model, with the error function defined as

$$E(t) = Y(t)N(t) - I. \quad (8)$$

The expression of the model is as follows:

$$\dot{E}(t) = -\gamma \mathcal{F}(E(t)). \quad (9)$$

Considering Equations (8) and (9), the ZNN-A model is as follows:

$$\dot{Y}(t)N(t) = -Y(t)\dot{N}(t) - \gamma \mathcal{F}(Y(t)N(t) - I).$$

The sign-bi-power (SBP) activation function is defined as follows:

$$\mathcal{F}(y_{ij}) = \frac{1}{2} \left(\text{Lip}^a(y_{ij}) + \text{Lip}^{1/a}(y_{ij}) \right),$$

$$\text{Lip}^a(y_{ij}) = \begin{cases} |y_{ij}|^a, & \text{if } y_{ij} > 0, \\ 0, & \text{if } y_{ij} = 0, \\ -|y_{ij}|^a, & \text{if } y_{ij} < 0. \end{cases}$$

The upper bound of the convergence time for this model is

$$\max \left\{ \frac{2|e^-(0)|^{1-a}}{q(1-a)}, \frac{2|e^+(0)|^{1-a}}{q(1-a)} \right\}.$$

In the formula, $e(0)$ denotes the initial error.

Liao proposed a novel complex-valued zeroing neural network (NCZNN) [53,102], which achieves finite-time convergence in the complex domain through two distinct approaches. The error function is

$$E(t) = B(t)Y(t) - k(t). \quad (10)$$

Considering Equation (9), the NCZNN model is given by

$$B(t)\dot{Y}(t) = -B(t)\dot{Y}(t) - \gamma \mathcal{F}(B(t)Y(t) - k(t)) + \dot{k}(t).$$

In general, there are two approaches for handling complex-valued activation functions, as follows:

$$\mathcal{F}_1(C + iH) = \Lambda(C) + i\Lambda(H).$$

$$\mathcal{F}_2(C + iH) = \Lambda(\tau) \circ \exp(i\Delta).$$

The upper bound of the NCZNN model is given as follows:

$$t_C \leq \frac{|a(0)|^{1-m}}{\gamma(1-m)},$$

where, $a(0) = \max_k \{|e_k(0)|\}$. In comparative experiments, the NCZNN model consistently outperforms the CZNN model [102].

In reference [95], Lei et al. introduced a nonlinear activation integral-enhanced zeroing neural network (NIEZNN) model based on the coalescent activation function (C-AF) activation function, comparing it with existing ZNN models. The experimental results highlighted its superiority.

In reference [103], Xiao et al. investigated the time-varying inequality constrained quaternion matrix least squares (TVIQLS) problem and proposed a fixed-time noise-tolerant zeroing neural network (FTNTZNN) model to solve it in complex environments. The TVIQLS problem is reformulated into matrix form, that is, the error function is analogous to Equation (10). By combining the error equation and the design Equation (4), the FTNTZNN model is formulated as follows:

$$\dot{E}(t) = -\gamma \mathcal{F}(E(t)) - \lambda_1 \left(\gamma \int_0^t \mathcal{F}(E(\iota)) d\iota + \mathcal{F}(E(t)) \right).$$

When solving the TVIQLS problem, only finite-time convergence can be achieved, and not fixed-time convergence. To address both challenges simultaneously, an improved activation function $\mathcal{F}(\cdot)$ is integrated into the ZNN model, defined as follows:

$$\mathcal{F}_l(z) = \begin{cases} \xi_1 |z|^{\mu} \text{sign}(z_l) + \xi_2 z_l, & \text{if } |z_l| \leq 1 \\ \xi_2 |z|^{\mu} \text{sign}(z_l) + \xi_3 z_l, & \text{if } |z_l| > 1 \end{cases}$$

Here, $0 < \mu_1 < 1$ and $\mu_2 > 1$, ξ_1, ξ_2, ξ_3 are positive parameters, and the upper bound of the model's convergence is given by

$$T = T_1 + T_2 \leq \frac{\rho + \lambda}{\rho \lambda \xi_1 (1 - \mu_1)} + \frac{\rho + \lambda}{\rho \lambda \xi_2 (\mu_2 - 1)}.$$

The FTNTZNN model is robust to initial values and external noise, offering a significant advantage over traditional zeroing neural network (CZNN) models. When compared to other ZNN models employing conventional activation functions, the FTNTZNN model exhibits faster convergence and enhanced robustness.

Xia et al. incorporated the activation function [36] into the ZNN model, achieving fixed-time convergence. Its form is as follows:

$$\mathcal{F}(x) = \frac{1}{2} h_1 \text{wsbp}^b(x) + \frac{1}{2} h_2 \text{wsbp}^{1/b}(x) + \frac{1}{2} h_3 x.$$

where $h_1, h_2, h_3 > 0$, $b \in (0, 1)$, and the function $\text{wsbp}(\cdot)$ is defined as

$$\text{wsbp}^b(y) = \begin{cases} |y|^b, & \text{if } y > 0 \\ 0, & \text{if } y = 0 \\ -|y|^b, & \text{if } y < 0 \end{cases}$$

Define the error function as

$$E(t) = A(t)A(t)Z(t) - L(t)Y(t).$$

The design formula is identical to that in Equation (9), i.e., the corresponding first-order fixed-time ZNN model (FOZNN-1) is

$$\begin{aligned} A(t)A(t)\dot{Z}(t) &= \dot{A}(t)Y(t) + A(t)\dot{Y}(t) \\ &\quad - \dot{A}(t)A(t)Z(t) - A(t)\dot{A}(t)Z(t) \\ &\quad - \gamma\mathcal{F}(A(t)A(t)Z(t) - A(t)Y(t)). \end{aligned}$$

It can be concluded that the upper bound of its convergence time is

$$T_1 \leq \begin{cases} \frac{\mu}{n_a(b-g)} \log\left(\frac{|b|}{|g|}\right), & n_c > 2\sqrt{n_a n_b} \\ \frac{\mu}{\sqrt{n_a n_b}} \frac{k}{1-k}, & n_c = 2\sqrt{n_a n_b} \\ \frac{\mu}{n_a k_1} \left(\frac{\pi}{2} - \tan^{-1} k_2\right), & 0 < n_c < 2\sqrt{n_a n_b} \\ \frac{\mu\pi}{2\sqrt{n_a n_b}}, & n_c \leq 0 \end{cases}$$

where n_a, n_b, n_c, b , and g are parameters, and $0 < k < 1$. The values $-b$ and $-g$ are the solutions of

$$r(s) = n_a s^2 - n_c s + n_b,$$

with

$$k_1 = \frac{\sqrt{4n_a n_b - n_c^2}}{4n_a^2}, \quad k_2 = \frac{n_c}{\sqrt{4n_a n_b - n_c^2}}.$$

The experiment shows that, compared to other models, this model achieves stronger convergence performance and realizes fixed-time convergence.

In the literature [86], Xiao introduced a versatile activation function (VAF) to address the TVMI problem. Considering Equations (1) and (9), the model can be expressed as follows:

$$B(t)\dot{Y}(t) = -B(t)\dot{Y}(t) - \gamma\mathcal{F}(B(t)Y(t) - I) + S(t), \quad (11)$$

where $S(t)$ represents general noise, and the design formula for the activation function is as follows:

$$\mathcal{F}(x) = (r_1|x|^\epsilon + r_2|x|^\zeta)\text{sgn}(x) + r_3x + r_4\text{sgn}(x).$$

The upper bound is given by

$$t_r = \frac{1}{\zeta(1-\epsilon)} + \frac{1}{v(\zeta-1)},$$

where $\zeta > 0$ and $v > 0$.

In the literature [104], Li et al. were the first to achieve predefined-time convergence for the ZNN model by introducing two novel activation functions. The error function is defined as follows:

$$E(t) := Y^2(t) - N(t).$$

Given the dynamic matrix $N(t)$ and the system dynamics $Y(t)$ to be solved, the perturbation time-varying ZNN (PTZNN) model is expressed as follows:

$$\begin{aligned} Y(t)\dot{Y}(t) + \dot{Y}(t)Y(t) &= -\lambda\mathcal{F}(Y^2(t) \\ &\quad - N(t)) - \dot{N}(t) + G(t). \end{aligned}$$

To achieve predefined-time convergence, two activation functions are proposed, which are defined as follows:

$$\mathcal{F}_1(x) = (\kappa_1|x|^p + \kappa_2|x|^q)\text{sign}(x) + \kappa_3x + \kappa_4\text{sign}(x).$$

The design formula for the second activation function is as follows:

$$\mathcal{F}_2(x) = \gamma_1 \exp(|x|^p)|x|^{1-p}\text{sign}(x)/p + \gamma_2x + \gamma_3\text{sign}(x).$$

The upper bound of the convergence time derived from the first activation function is as follows:

$$t_r \leq \frac{1}{\lambda\kappa_1(1-p)} + \frac{1}{\lambda\kappa_2(q-1)}.$$

When utilizing the second activation function, the upper bound of the convergence time is

$$t_c \leq \frac{1}{\lambda\gamma_1}.$$

In addressing the dynamic matrix square root (DMSR) problem, the PTZNN model outperforms existing models in both convergence and robustness.

In reference [105], Li et al. proposed a strict predefined-time convergence and noise-tolerant ZNN (SPTC-NT-ZNN) for solving time-varying linear systems. The model is consistent with Equation (11), where the activation function is defined as

$$h(\delta) := \begin{cases} \delta/(t_c - t), & t \in [0, t_c) \\ \delta + |\delta|^p\text{sign}(\delta) + \xi\text{sign}(\delta), & t \in [t_c, +\infty) \end{cases}$$

where the parameters $0 < p < 1$ and $\xi \geq 0$ are given, and the parameter t_c is related to the convergence time. Additionally, the function $\text{sign}(\delta) = \frac{\delta}{|\delta|}$ is defined for $\delta \neq 0$.

This ensures the required timely convergence and robustness for time-critical applications. The strict predefined-time convergence and noise tolerance of the SPTC-NT-ZNN have been theoretically proven and further validated through comparative experiments to demonstrate its superiority. The comparison focused on two illustrative problems: TVOLE and TVULE. The numerical results demonstrate that, in both convergence and robustness, the SPTC-NT-ZNN outperforms other existing ZNN models in solving these problems.

Remark 1. In ZNN, nonlinear activation functions play a crucial role in shaping the neural dynamics, facilitating the achievement of the desired convergence behavior. These functions typically operate on error terms and, through careful design, can guide the system to systematically reduce the error to zero. The convergence behaviors influenced by well-designed nonlinear activation functions include finite-time convergence, fixed-time convergence, and preset-time convergence.

Remark 2. This section presents the mathematical definitions and fundamental properties of three distinct types of convergence: finite-time convergence, fixed-time convergence, and predefined-time convergence.

Finite-time convergence refers to the system's convergence to the equilibrium point $x = 0$ from an initial state x_0 within a finite time. If the system is Lyapunov stable, and there

exists a finite convergence time function $T(x_0)$ depending on the initial state x_0 , then the system will converge within that time. Specifically, given the system's dynamics,

$$\dot{x}(t) = f(x(t), t), \quad x(0) = x_0,$$

the finite-time stability satisfies the following condition:

$$\dot{V}(x) \leq -kV(x)^q, \quad 0 < q < 1$$

where $V(x)$ is a positive definite Lyapunov function, and k is a constant. The convergence time is

$$T(x_0) = \frac{1}{k(1-q)} V(x_0)^{1-q}$$

This time explicitly depends on the initial condition x_0 and indicates that the system will converge to the equilibrium point within a finite time.

Fixed-time convergence refers to the system's convergence to the equilibrium point in a fixed, initial-condition-independent time. For all initial conditions x_0 , there exists a fixed maximum convergence time $T_{\max}$ such that

$$x(t) = 0, \quad \forall t \geq T_{\max}$$

A typical Lyapunov condition is

$$\dot{V}(x) \leq -(\alpha V(x)^p + \beta V(x)^q)^k$$

where p, q are constants satisfying $pk < 1$ and $qk > 1$, and constants $\alpha, \beta > 0$. The upper bound for the fixed-time convergence is

$$T_{\max} = \frac{1}{\alpha^k(1-pk)} + \frac{1}{\beta^k(qk-1)}$$

This upper bound is independent of the initial condition x_0 and ensures that the system converges in fixed time.

Predefined-time convergence requires that the system fully converges to the equilibrium point within a user-specified fixed time T_p , regardless of the initial condition. The system's Lyapunov condition is

$$\dot{V}(x) \leq -\frac{1}{pT_p} e^{V(x)^p} V(x)^{1-p}$$

where p is a constant, satisfying $0 < p \leq 1$. If this condition holds, the system exhibits strong predefined-time stability, and the convergence time is strictly T_p .

The Figure 3 illustrates the convergence behavior of the ZNN error $\|E(t)\|_F$ over time under various activation functions. Among them, the orange curve—corresponding to the predefined-time activation function—achieves the fastest convergence, followed by the yellow curve representing the fixed-time activation function. In contrast, the purple curve, associated with the finite-time activation function, exhibits the slowest convergence. These results clearly indicate that the predefined-time activation function facilitates the most rapid error reduction in the ZNN, outperforming the fixed-time and finite-time counterparts. This observation underscores the significant influence of activation function design on the convergence performance of the ZNN.

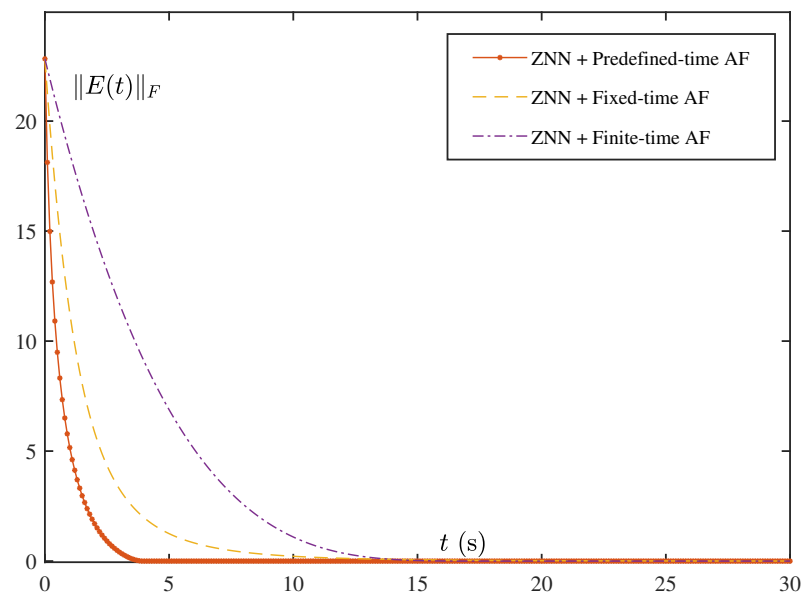


Figure 3. Real-time errors of the three different models.

3.2. Nonlinear Activation Functions with Noise-Tolerant Capabilities

With ongoing advancements in neural network models, nonlinear activation functions have become crucial not only for accelerating convergence but also for significantly enhancing noise robustness. Specifically, noise robustness refers to the model's ability to maintain stability and perform inference and prediction effectively in the presence of input noise or system disturbances.

In summary, nonlinear activation functions enhance the model's noise robustness, enabling the network to effectively handle complex and uncertain environments in real-world applications. Simultaneously, they accelerate convergence while preserving the model's efficiency and robustness. This combination lays a strong foundation for the widespread adoption of neural networks in fields such as real-time control, intelligent decision making, and dynamic optimization.

In reference [106], Xiao et al. constructed and analyzed a novel recursive neural network (NRNN) that exhibits finite-time convergence and exceptional robustness, specifically for solving the TVSE with additive noise. In contrast to the design methodology of the ZNN, the proposed NRNN utilizes a sophisticated integral design formula in conjunction with a nonlinear activation function. This integration not only accelerates the convergence rate but also effectively mitigates the impact of unknown additive noise in the process of solving dynamic Sylvester equations. The integral design formula is analogous to Equation (4). The design of the activation function is as follows:

$$\mathcal{F}_1(e_{ij}) = \mathcal{F}_2(e_{ij}) = \varphi^u(e_{ij}) + \varphi^{1/v}(e_{ij}),$$

where the design parameters $0 < v < 1$, The definition of $\varphi^v(\cdot)$ is given by

$$\varphi^v(e_{ij}) = \begin{cases} |e_{ij}|^v, & \text{if } e_{ij} > 0 \\ 0, & \text{if } e_{ij} = 0 \\ -|e_{ij}|^v, & \text{if } e_{ij} < 0 \end{cases}$$

The ij -th subsystem of the integral design formula can be expressed as follows:

$$\dot{e}_i(t) = -\gamma f_1(e_{ij}(t)) - \lambda f_2(e_{ij}(t)) + \gamma \int_0^t f_1(e_{ij}(\tau)) d\tau.$$

By combining the error function with the design formula, the NRNN model can be constructed to solve the dynamic Sylvester equation, which has a similar form to Equation (5).

In [81], Xiao et al. applied the ZNN model to solve TVSME. The use of a noise-resistant activation function allows the ZNN model to effectively solve the Stein equation in noisy environments. Therefore, the ZNN model not only exhibits enhanced convergence performance but also improves noise immunity. To address this issue, a complex-valued error function is defined:

$$E(t) = D(t)Y(t)Z(t) + Y(t) - F(t),$$

By utilizing the Kronecker product, the error function $E(t)$ can be reformulated as

$$\vec{E}(t) = V(t)\vec{Y}(t) - \vec{F}(t).$$

Since a complex number can be expressed as the sum of its real and imaginary parts, $\vec{E}(t)$ is represented as $\vec{E}_r(t) + i\vec{E}_i(t)$, where i is the imaginary unit. Furthermore, we have

$$\vec{E}(t) = -\varrho(t)(\mathcal{F}(\vec{E}_r(t)) + i\mathcal{F}(\vec{E}_i(t))).$$

To ensure noise robustness, the following activation function is adopted:

$$\mathcal{F}(x) = \frac{r_1}{d} \exp(|x|^d) |x|^{1-d} \text{sign}(x).$$

The PTAN-VP ZNN model presented below can be derived using the design formula outlined above, as detailed in [107].

$$\begin{cases} V(t)\dot{\vec{Y}}(t) = -\dot{V}(t)\vec{Y}(t) + \dot{\vec{F}}(t) \\ \quad -\varrho(t)(\mathcal{F}(\vec{E}_r(t)) + i\mathcal{F}(\vec{E}_i(t))), \\ \dot{\gamma}(t) = \exp(\alpha \text{sign}(|\vec{E}(t)|)) - 1. \end{cases}$$

Compared to other ZNN models, such as the LZNN [108], NLZNN [109], FTCZNN [72] and PTCZNN [104], the PTAN-VP ZNN exhibits superior interference rejection performance. This paper presents a theoretical analysis of the stability and robustness of the PTAN-VP ZNN. The validity of the theoretical results has been confirmed through numerical simulations, which also highlight the advantages of the PTAN-VP ZNN. Moreover, the PTAN-VP ZNN has been successfully applied to mobile robotic arms, demonstrating its potential for use in robotic control.

In [97], a nonlinear activation-based integral design formula was proposed to address the effects of additive noise. Building upon this design formula, a NRNN was developed to solve dynamic quadratic optimization problems. Compared to the ZNN applied to this problem, the proposed RNN model demonstrates significant finite-time convergence and inherent noise-resistance capabilities.

The activation functions and their convergence types are shown in Table 1. In addition to using activation functions to improve the robustness of the model, adaptive compensation terms can be introduced to mitigate the impact of noise. For example, Liao et al. [47] proposed a harmonic noise-tolerant zeroing neural network (HNTZNN) model to efficiently solve matrix pseudoinversion problems.

Table 1. Activation functions and their convergence.

Activation Function	Parameter	Convergence
$\frac{1}{2} \left(e ^\kappa + e ^{\frac{1}{\kappa}} \right) \text{sgn}(e)$	$\kappa > 0$	Finite-time [109]
$\begin{cases} \frac{b}{a} \sqrt{2a e_{ij} - e_{ij} ^2}, & e_{ij} < a \\ b, & e_{ij} \geq a \end{cases}$	$a, b > 0$	Finite-time [110]
$\begin{cases} \frac{1 - \exp(-\varepsilon x)}{1 + \exp(-\varepsilon x)}, & \frac{1 + \exp(-\varepsilon)}{1 - \exp(-\varepsilon)} \end{cases}$	$\varepsilon > 0$	Finite-time [111]
$(\rho_1 x ^a + \rho_2 x ^{1/a} + \rho_3 x) \text{sgn}(x)$	$\rho_1, \rho_2, \rho_3, \rho_4, a > 0$	Finite-time [75]
$(b_1 e ^\kappa + b_2 e ^\eta) \text{sgn}(e) + b_3 e$	$b_1, b_2, b_3, \kappa, \eta > 0$	Fixed-time [112,113]
$(a_1 x ^\nu + a_2 x ^{\nu+1} + a_3 x - a_4) \text{sgn}(x)$	$a_1, a_2, a_3, a_4, \nu > 0$	Fixed-time [80]
$\frac{1}{p} (\kappa_1 x ^{1-p} + \kappa_2 x ^q) \text{sgn}(x) + \kappa_3 \text{sgn}(x)$	$q, \kappa_1, \kappa_2, \kappa_3 > 0, p \in (0, 1)$	Fixed-time [114]
$h \sin(x ^n) \text{sgn}(x)$	$h, n > 0$	Fixed-time [115]
$\begin{cases} \frac{e}{t_m - t}, & t \in [0, t_m) \\ e + e ^q \text{sgn}(e) + p \text{sgn}(e), & t \in [t_m, +\infty) \end{cases}$	$t_m, q, p > 0$	Predefined-time [105]
$\begin{cases} \frac{\exp(e) - 1}{\exp(e)(t_m - t)}, & t \in [0, t_m) \\ e, & t \in [t_m, +\infty) \end{cases}$	$t_m > 0$	Predefined-time [116]
$(d_1 e ^\nu + d_2 e ^\epsilon) \text{sgn}(e) + d_3 \text{sgn}(e) + d_4 e$	$d_1, d_2, d_3, d_4, \nu, \epsilon > 0$	Predefined-time [86,117]
$(c_1 \sum_{i=1}^m e ^{v_i} + c_2 \sum_{i=1}^m e ^{\kappa_i}) \text{sgn}(e) + c_3 e + c_4 \sinh(e)$	$c_1, c_2, c_3, c_4, v_i, \kappa_i > 0$	Predefined-time [118]
$\frac{\eta}{c} \exp(x ^c) x ^{1-c} \text{sgn}(x) + \nu \text{sgn}(x)$	$\eta, \nu > 0, c \in (0, 1)$	Predefined-time [119]

Algorithm 1 presents the pseudocode in a unified format, illustrating the discrete-time implementation process of four ZNN models: the original ZNN model, the ZNN model enhanced with nonlinear activation functions, the IEZNN model, and the DIEZNN model. This algorithmic framework is suitable for typical application scenarios such as real-time control systems, trajectory tracking, robotic control, and the solution of dynamic matrix equations.

The pseudocode mainly consists of the following components:

- **Parameter initialization:** including the initial state $Y(0)$;
- **Time-step iteration:** iterating from $n = 0$ to $t_{\max}/\tau$ with a fixed step size τ ;
- **Model-specific control law and state update:** updating the state variable $Y(t_n)$ based on the corresponding control law of each ZNN model;
- **Introduction and update of auxiliary variables:** where $Z(t_n)$ denotes the single-integral term and $N(t_n)$ denotes the double-integral term.

3.3. The Variable Parameter Improves the Convergence Performance of Zeroing Neural Network Models

To enhance the convergence rate, the use of variable parameters (VPs) presents another effective strategy. These parameters are dynamically adjusted over time, typically following a time-dependent function (e.g., exponential or power functions) that governs their evolution. The dynamic adjustment capability of VPs offers significant benefits, including improved system convergence, enhanced robustness, and better alignment with practical hardware constraints. Although the design and implementation may be more complex, these advantages make variable parameters the preferred approach for addressing complex dynamic problems, particularly in scenarios characterized by time-varying properties or external disturbances.

In the literature [52], a variable-parameter recurrent neural network (VP-CDNN) is proposed, and Equation (9) is reformulated as follows:

$$\dot{E}(t) = -\chi(t)\mathcal{F}(E(t)) = -(t^\kappa + \kappa)\mathcal{F}(E(t)).$$

Algorithm 1 Pseudocode of Discrete Controllers Based on Different ZNN Models**Parameters initialization:** e.g., $Y(0)$ **for** $n = 0$ to $t_{\max}/\tau$ **do** Compute relevant coefficients with $t_n = n\tau$, e.g., $Q(t_n)$, γ , λ_1 , λ_2 . **if** (ZNN controller I) **then** Update $Y(t_{n+1})$ by following the control law:

$$\dot{Y}(t_n) = B^\dagger(-\gamma(BY(t_n) - K(t_n)) + \dot{K}(t_n))$$

$$Y(t_{n+1}) = Y(t_n) + \tau\dot{Y}(t_n)$$

else if (ZNN controller II) **then** Update $Y(t_{n+1})$ by following the control law:

$$\dot{Y}(t_n) = B^\dagger(-\gamma\mathcal{F}(BY(t_n) - K(t_n)) + \dot{K}(t_n))$$

$$Y(t_{n+1}) = Y(t_n) + \tau\dot{Y}(t_n)$$

else if (IEZNN controller under noise-free conditions) **then** Update $Y(t_{n+1})$ by following the control law:

$$\dot{Y}(t_n) = B^\dagger(-\gamma\mathcal{F}(BY(t_n) - K(t_n)) + Z(t_n)) + \dot{K}(t_n)$$

$$\dot{Z}(t_n) = -\lambda_1\mathcal{F}(BY(t_n) - K(t_n))$$

$$Z(t_{n+1}) = Z(t_n) + \tau\dot{Z}(t_n)$$

$$Y(t_{n+1}) = Y(t_n) + \tau\dot{Y}(t_n)$$

else if (DIEZNN controller under noisy conditions) **then** Update $Y(t_{n+1})$ by following the control law:

$$\dot{Y}(t_n) = B^\dagger(-\gamma\mathcal{F}(BY(t_n) - K(t_n)) + Z(t_n) + N(t_n)) + \dot{K}(t_n)$$

$$\dot{N}(t_n) = \lambda_2/\lambda_1 * Z(t_n)$$

$$\dot{Z}(t_n) = -\lambda_1\mathcal{F}(BY(t_n) - K(t_n))$$

$$N(t_{n+1}) = N(t_n) + \tau\dot{N}(t_n)$$

$$Z(t_{n+1}) = Z(t_n) + \tau\dot{Z}(t_n)$$

$$Y(t_{n+1}) = Y(t_n) + \tau\dot{Y}(t_n)$$

end if**end for**

Further, the CDNN model is obtained as follows:

$$\begin{aligned} L(t)\dot{Z}(t) - \dot{Z}(t)F(t) &= Z(t)\dot{F}(t) - \dot{L}(t)Z(t) - \dot{G}(t) \\ &- (t^\kappa + \kappa)\mathcal{F}(L(t)Z(t) - Z(t)F(t) + G(t)). \end{aligned}$$

This paper presents a novel VP-CDNN model, which innovatively incorporates a time-varying parameter function $\chi(t)$, significantly enhancing the solving performance of time-varying Sylvester equations. Through rigorous mathematical proofs, the study confirms the superior performance of this model in terms of convergence and robustness. Specifically, the VP-CDNN not only achieves super-exponential convergence performance but also demonstrates strong robustness characteristics, all of which have been thoroughly validated through multiple theorems. In the comparative simulation experiments, the VP-CDNN exhibited significant advantages in convergence speed.

In the literature [56], Xiao constructed an innovative finite-time varying-parameter convergent differential neural network (FT-VP-CDNN) aimed at solving nonlinear and non-convex optimization problems. The study not only provides a detailed analysis of the network's performance but also presents its design formula, which is specifically expressed as follows:

$$\dot{E}(t) = -\vartheta(t)\mathcal{F}(E(t) - E^+(t) + \tilde{E}(t)),$$

where $\vartheta(t) = \varepsilon \exp(t) = \varepsilon e^t > 0$ represents a time-varying parameter function. Research indicates that the proposed finite-time varying-parameter convergent differential neural network (FT-VP-CDNN) demonstrates significant performance advantages over the finite-time fixed-parameter convergent differential neural network (FT-FP-CDNN).

In [120], an IEZNN model was proposed to address the TVMI under noise interference. The IEZNN model performs well in handling relatively small time-varying noise; however, its performance is significantly affected by noise interference. As the noise level increases, the convergence accuracy of the model may degrade, and it may even fail to accurately approximate the theoretical solution. To address this limitation and further enhance performance, Xiao constructed a novel variable-parameter noise-tolerant zeroing neural network (VPNTZNN) model in this study. The mathematical formulation of the model is presented as follows:

$$B(t)\dot{Y}(t) = -\dot{B}(t)Y(t) - \mu_1(t)(B(t)Y(t) - I) - \mu_2(t) \int_0^t (B(\tau)Y(\tau) - I) d\tau + D(t).$$

The design formula is derived from the error function in Equation (1) and the design principles outlined in Equation (3), where $\mu_1(t)$ and $\mu_2(t)$ are defined as follows:

$$\begin{cases} \mu_1(t) = 3 \exp(\frac{\alpha t}{2}) - \frac{\alpha}{2}, \\ \mu_2(t) = \exp(\alpha t). \end{cases}$$

Here, $t \in [0, +\infty)$ and $\alpha \in (0, +\infty)$. Notably, $\mu_1(t)$ and $\mu_2(t)$ are time-varying parameters that remain strictly positive. Additionally, $D(t)$ denotes the external noise.

Through rigorous theoretical analysis and proof, the superior performance of the VPNTZNN model in terms of convergence and robustness has been fully validated. For further developments on variable parameters, refer to Table 2. The taxonomy of ZNN architectures discussed in this section is illustrated in Figure 4.

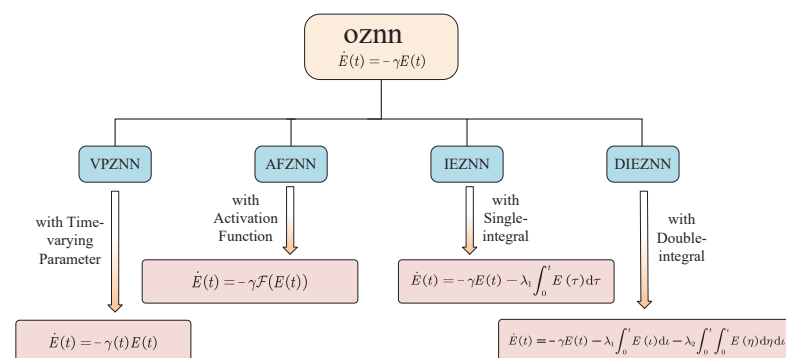


Figure 4. Taxonomy of ZNN Architectures.

Table 2. Development of varying parameters.

Variable Parameter	Parameter	Year	Literature
$\theta(t) = \zeta + t^\gamma$	$\zeta > 0, \gamma > 0$	2018	[121]
$\theta(t) = p^t + p$	$p > 0$	2019	[122]
$\theta(t) = t^p + p$	$p > 0$	2019	[123]
$\dot{\theta}(t, x) = c \cdot \text{sign}(x)$	$c > 0$	2020	[124]
$\theta(t) = \begin{cases} t^q + q, & \text{if } 0 < q \leq 1 \\ q^t + 2qt + p, & \text{if } q > 1 \end{cases}$	$q > 0$	2021	[125]
$\theta(t) = \beta \exp(\lambda_1 \operatorname{arccot}(t) + \lambda_2 t)$	$\beta, \lambda_1, \lambda_2 > 0$	2021	[126]
$\begin{cases} \theta_1(t) = 3 \exp(\frac{\alpha t}{2}) - \frac{\alpha}{2} \\ \theta_2(t) = \exp(\alpha t) \end{cases}$	$\alpha > 0$	2022	[120]
$\theta(t) = \begin{cases} \gamma_1 k_1^{\alpha_1 t}, & \text{if } t < \delta_0 \\ \gamma_1 k_1^{\alpha_1 \delta_0}, & \text{if } t \geq \delta_0 \end{cases}$	$\gamma_1, k_1, \alpha_1, \delta_0 > 0$	2023	[96]
$\theta(t) = \gamma \exp(\ E(t)\)$	$\gamma > 0$	2023	[127]
$\theta(t) = \varrho \exp((t^\kappa + \kappa)\theta(t))$	$\varrho, \kappa > 0$	2024	[128]
$\theta(t) = \varrho \exp((\beta^t + \beta)\ E(t)\)$	$\varrho, \beta > 0$	2024	[129]

4. Applications of Zeroing Neural Networks

This chapter will comprehensively discuss the research progress and practical applications of ZNN in various fields.

In 2023, Liao et al. proposed a dynamic robot position tracking method based on complex number representation (see reference [43]) and designed an optimization strategy for the real-time measurement and minimization of robot spacing. They further developed a CZND model for dynamic solving, with the formulation expressed as follows:

$$E\Phi(t) + Y(t) = 0.$$

In the CTVME, $\Phi(t)$, representing the instantaneous position of the following robot, needs to be solved in real time. Based on the ZNN design framework, by solving the CTVME problem online, the zeroing neural dynamics method demonstrates its efficiency and feasibility in robot coordination. The error function is

$$P(t) = E\Phi(t) + Y(t).$$

This is used to quantify the error in the CTVME problem. The time derivative of $P(t)$ is defined as follows:

$$\dot{P}(t) = -\gamma(t)P(t).$$

Further, we can obtain

$$\dot{E}\Phi(t) + E\dot{\Phi}(t) + \dot{Y}(t) = -\gamma(EY(t) + Y(t)).$$

The CZND model is

$$E\dot{\Phi}(t) = -\gamma E\Phi(t) - \dot{Y}(t) - \gamma Y(t).$$

In 2014, Xiao et al. proposed a method based on the ZNN model to solve problems related to robotic arms [130]. The kinematic equation of the robotic arm is typically expressed as

$$\begin{aligned} g(t) &= f(\theta(t)), \\ \dot{g}(t) &= G(\theta(t))\dot{\theta}(t). \end{aligned}$$

The error function is defined as follows:

$$E(t) = r_w - r(t),$$

where $r_w(t)$ represents the desired path to be tracked. By integrating the previously proposed formulas with the original ZNN model design equations outlined in the article, the wheeled mobile manipulator's dynamics are derived as follows:

$$G(\theta(t))\dot{\theta} = \dot{r}_w(t) + \gamma(r_w - f(\theta(t))).$$

A series of comprehensive experiments were carried out using the formulas outlined earlier. The results indicate that the ZNN method outperforms the traditional GNN approach in terms of accuracy.

With the development of ZNN, its application in robotic arm control has become increasingly widespread. Notable examples include minimum motion planning and control for redundant robotic arms [131,132], cooperative motion planning for robotic manipulator arms [83,84,93,106,133], multi-robot systems [134], path tracking for mobile robots [81,135], redundant robotic manipulators [86,87,136,137], four-joint planar robotic arms [47,94], motion tracking for mobile manipulators [102], coordinated path tracking for dual robotic manipulators [88], solving multi-robot tracking and formation problems [89], vehicular edge computing [138], and mobile object localization [49], among others.

Chaotic systems, first discovered by Edward Lorenz half a century ago, are a class of typical nonlinear systems [139]. Since their discovery, chaotic systems have become a focal point of research due to their wide range of practical applications, including in fields such as power systems [140], financial systems [141], ecological systems [142], and secure communication [143]. However, their inherent uncertainty, non-repeatability, and unpredictability make solving chaotic system problems highly challenging. The introduction of the ZNN model offers a reliable solution to effectively address issues in chaotic systems, particularly in environments with noise and uncertainties. The basic approach involves constructing models for the master and response systems:

$$\dot{x}_m(t) = f_m(x_m(t)) + \omega(t), \quad \dot{x}_r(t) = f_r(x_r(t)) + \mu(t),$$

where $x_m(t)$ and $x_r(t)$ represent the states of the master and response systems, $f_m(\cdot)$ and $f_r(\cdot)$ are the nonlinear dynamics of the systems, $\omega(t)$ denotes external disturbances, and $\mu(t)$ is the control input. The synchronization error is defined as $e(t) = x_m(t) - x_r(t)$, and the ZNN control law is given by $\dot{e}(t) = -\gamma e(t)$, ensuring exponential convergence of the error. To enhance noise immunity, the ZNN can incorporate integral and double-integral structures to mitigate low-frequency and high-frequency disturbances.

Studies have shown that ZNN-based models perform excellently in chaotic control. For example, Aoun et al. [144] proposed the NZNN, which successfully achieved three-

dimensional synchronization in the SFM system. Xiao et al. [145] combined the ZNN with sliding mode control to develop the FXTRC strategy, achieving nearly 10 times faster convergence in various chaotic systems.

In 2023, Jin et al. introduced a time-varying fuzzy parameter zeroing neural network (TVFPZNN) model designed to achieve the synchronization of chaotic systems in the presence of external noise interference [146].

To demonstrate the superiority of the TVFPZNN, Jin conducted two synchronization experiments using the Chen chaotic system and an autonomous chaotic system, employing different fuzzy membership functions. During these experiments, three types of irregular noise were introduced to rigorously evaluate the model's robustness. As documented in [146], the mathematical formulation of the Chen chaotic system is given by

$$\begin{cases} \dot{y}_1(t) = a(y_2(t) - y_1(t)), \\ \dot{y}_2(t) = dy_1(t) - y_1(t)y_3(t) + cy_2(t), \\ \dot{y}_3(t) = y_1(t)y_2(t) - by_3(t). \end{cases}$$

In the presence of external noise interference, the master chaotic system can be represented as

$$\begin{cases} \dot{y}_{m1}(t) = a(y_{m2}(t) - y_{m1}(t)) + \omega_1(t), \\ \dot{y}_{m2}(t) = dy_{m1}(t) - y_{m1}(t)y_{m3}(t) + cy_{m2}(t) + \omega_2(t), \\ \dot{y}_{m3}(t) = y_{m1}(t)y_{m2}(t) - by_{m3}(t) + \omega_3(t). \end{cases}$$

The response chaotic system, incorporating the controller, can be represented as

$$\begin{cases} \dot{y}_{r1}(t) = a(y_{r2}(t) - y_{r1}(t)) + \mu_1(t), \\ \dot{y}_{r2}(t) = dy_{r1}(t) - y_{r1}(t)y_{r3}(t) + cy_{r2}(t) + \mu_2(t), \\ \dot{y}_{r3}(t) = y_{r1}(t)y_{r2}(t) - by_{r3}(t) + \mu_3(t). \end{cases}$$

As presented in reference [146], the mathematical formulation of the TVFPZNN model is given by

$$\begin{aligned} & f_m(z_m(t)) + \eta(t) - f_r(z_r(t)) - \mu(t) \\ & = -(a^{t+2k} + \lambda pt + p^2)\mathcal{F}(z_m(t) - z_r(t)) + \eta(t). \end{aligned}$$

The expression $a^{t+2k} + \lambda pt + p^2$ denotes the fuzzy time-varying parameters.

In Experiment B [146], the researchers performed a comparative analysis of the PTVR-ZNN [147], AFT-ZNN [148], FPZNN [149], and TVFPZNN models for controlling the Chen chaotic system in both noise-free and noisy environments. The results demonstrated that, in the noise-free environment, all four models successfully achieved synchronization. However, in the presence of noise, only the TVFPZNN model was able to stably synchronize the Chen chaotic system. Moreover, under noise-free conditions, the Chen chaotic system controlled by the TVFPZNN exhibited the fastest convergence speed and the lowest error, further validating the superior performance of this model.

In Experiment C [146], the researchers examined the synchronization problem of the autonomous chaotic system. The mathematical formulation of the autonomous chaotic system is given by

$$\begin{cases} \dot{z}_1(t) = p(z_2(t) - z_1(t)) + z_2(t)z_3(t), \\ \dot{z}_2(t) = (r - p)z_1(t) - z_1(t)z_3(t) + rz_2(t), \\ \dot{z}_3(t) = sz_2(t)z_2(t) - qz_3(t), \end{cases}$$

Similarly, in the presence of external noise interference, the master chaotic system can be represented as

$$\begin{cases} \dot{z}_{m1}(t) = p(z_{m2}(t) - z_{m1}(t)) + z_{m2}(t)z_{m3}(t) + \omega_1(t), \\ \dot{z}_{m2}(t) = (r - p)z_{m1}(t) - z_{m1}(t)z_{m3}(t) + rz_{m2}(t) + \omega_2(t), \\ \dot{z}_{m3}(t) = sz_{m2}(t)z_{m2}(t) - qz_{m3}(t) + \omega_3(t). \end{cases}$$

The response chaotic system, incorporating the controller, can be represented as

$$\begin{cases} \dot{z}_{r1}(t) = p(z_{r2}(t) - z_{r1}(t)) + z_{r2}(t)z_{r3}(t) + \mu_1(t), \\ \dot{z}_{r2}(t) = (r - p)z_{r1}(t) - z_{r1}(t)z_{r3}(t) + rz_{r2}(t) + \mu_2(t), \\ \dot{z}_{r3}(t) = sz_{r2}(t)z_{r2}(t) - qz_{r3}(t) + \mu_3(t). \end{cases}$$

In this context, $\pi(t)$ and $\mu(t)$ represent the controllers.

The experiment evaluated the performance of the aforementioned models in controlling the autonomous chaotic system under noisy conditions. The results showed that the TVFP-ZNN model outperformed the other models by a significant margin.

Over the past decade, the rapid development of the ZNN has made a significant impact across various fields. It has demonstrated significant effectiveness, particularly in robotic control. In tasks such as trajectory tracking, motion planning, and formation control, the ZNN is especially suitable for real-time control applications due to its rapid convergence and ability to handle disturbances, providing precise responses. Table 3 presents a performance comparison of the ZNN across various applications. Additionally, the ZNN has been successfully applied in chaos system control, drone coordination, and chaos circuit synchronization, highlighting its versatility and strong performance in dynamic control tasks. In addition to its applications in robot control and chaotic systems, as mentioned earlier, the ZNN has also been widely used in image information processing [150,151], multidimensional spectral estimation [152], mathematical ecology [153], IPC system pendulum tracking [154], and mobile target localization [155–157], among other areas.

Table 3. Performance comparison table of ZNN in various applications.

Model	Application Scenarios	Position Error	Integral Structure	Reference
HADTZTM	Manipulator motion planning	10^{-5}	No	[158]
FTZNN	Manipulator motion planning	10^{-5}	No	[159]
ITFCZNN	Manipulator motion planning	10^{-4}	No	[160]
RZND	Manipulator motion planning	10^{-4}	Single	[161]
FTCND	Manipulator motion planning	10^{-4}	No	[162]
STZNN	Manipulator motion planning	10^{-5}	Single	[163]
VP-CDNN	Manipulator motion planning	10^{-7}	No	[121]
DZNN	Manipulator motion planning	10^{-8}	No	[164]
CNDSM	Manipulator motion planning	10^{-4}	Single	[165]
FER-DZNN	Manipulator motion planning	10^{-5}	Single	[166]
CZND	multi-agent system control	10^{-4}	No	[43]
AP-FTZND	multi-agent system control	10^{-7}	No	[167]
TVFP-ZNN	Chaotic system	10^{-6}	No	[146]
NZNN	Chaotic system	10^{-4}	Single	[144]

5. Conclusions

This paper presents a comprehensive review on the application of the ZNN model in addressing time-varying problems, focusing on its model structure. The models discussed include single-integral and double-integral structures with noise immunity, general nonlinear function structures, finite-time convergence structures, fixed-time convergence structures, predefined-time convergence structures, and variable-parameter structures. Furthermore, the paper also explores the robustness of the ZNN in addressing noise, external disturbances, and system uncertainties, demonstrating its engineering practicality in tasks such as trajectory tracking and chaos control. The successful application of the ZNN in complex systems, such as multi-arm collaborative control, multi-agent formation, and nonholonomic robot path planning, highlights its powerful capability in handling high-dimensional, dynamic, and coupled problems.

As the ZNN model continues to evolve, its applications have become widespread across various practical domains. However, several key challenges still persist in this field. (1) Higher-order dynamics or multiple-integral structures improve performance but increase computational complexity. In real-time or resource-constrained environments, balancing performance and computational cost is a key challenge. (2) The ZNN model relies on gradient information, making it suitable for convex optimization problems, but it still faces challenges in non-convex or multi-modal optimization problems. In the future, combining the ZNN with swarm intelligence or evolutionary algorithms could enhance its global search capability. (3) The application of zeroing neural networks could be expanded to more fields. In conclusion, this review provides a reference for beginners who wish to gain a deeper understanding of how zeroing neural networks efficiently solve time-varying problems.

Author Contributions: Conceptualization, C.H. and Y.W.; validation, A.H.K.; investigation, A.H.K. and C.H.; writing—original draft preparation, Y.W.; visualization, Y.W.; supervision, C.H.; project administration, A.H.K.; funding acquisition, C.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China grant number 62466019.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ZNN	zeroing neural network
GNN	gradient neural network
TVCMI	time-varying complex matrix inversion
TVCMP	time-varying complex matrix pseudoinversion
TVNE	time-varying nonlinear equation
TVOLS	time-varying overdetermined linear system
TVSME	time-varying Stein matrix equation
TVNM	time-varying nonlinear minimization
NNP	nonconvex nonlinear programming
MOO	multi-objective optimization
TVQO	time-varying quadratic optimization

TVP	time-varying problems
NT	noise-tolerant
RNN	recurrent neural network
VEH	harris hawks algorithm
ZND	zeroing neural dynamics
BZND	bounded zeroing neural dynamics
NCZNN	novel complex-valued zeroing neural network
NIEZNN	nonlinear activation integral-enhanced zeroing neural network
C-AF	coalescent activation function
FTNTZNN	fixed-time noise-tolerant zeroing neural network
VAF	versatile activation function
NRNN	novel recursive neural network
TVFPZNN	time-varying fuzzy parameter zeroing neural network
VPNTZNN	variable-parameter noise-tolerant zeroing neural network
FT-VP-CDNN	finite-time varying-parameter convergent differential neural network
VP-CDNN	variable-parameter recurrent neural network

References

1. Zhong, J.; Zhao, H.; Zhao, Q.; Zhou, R.; Zhang, L.; Guo, F.; Wang, J. RGCNPPIS: A Residual Graph Convolutional Network for Protein-Protein Interaction Site Prediction. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2024**, *21*, 1676–1684. [\[CrossRef\]](#) [\[PubMed\]](#)
2. Ding, Y.; Mai, W.; Zhang, Z. A novel swarm budorcas taxicolor optimization-based multi-support vector method for transformer fault diagnosis. *Neural Netw.* **2025**, *184*, 107120. [\[CrossRef\]](#)
3. Zhang, Z.; Zhang, J.; Mai, W. VPT: Video portraits transformer for realistic talking face generation. *Neural Netw.* **2025**, *184*, 107122. [\[CrossRef\]](#)
4. Xiang, Z.; Guo, Y. Controlling Melody Structures in Automatic Game Soundtrack Compositions with Adversarial Learning Guided Gaussian Mixture Models. *IEEE Trans. Games* **2021**, *13*, 193–204. [\[CrossRef\]](#)
5. Long, C.; Zhang, G.; Zeng, Z.; Hu, J. Finite-time stabilization of complex-valued neural networks with proportional delays and inertial terms: A non-separation approach. *Neural Netw.* **2022**, *148*, 86–95. [\[CrossRef\]](#)
6. Zhang, Z.; Ding, C.; Zhang, M.; Luo, Y.; Mai, J. DCDLN: A densely connected convolutional dynamic learning network for malaria disease diagnosis. *Neural Netw.* **2024**, *176*, 106339. [\[CrossRef\]](#) [\[PubMed\]](#)
7. Xiang, Z.; Xiang, C.; Li, T.; Guo, Y. A self-adapting hierarchical actions and structures joint optimization framework for automatic design of robotic and animation skeletons. *Soft Comput.* **2021**, *25*, 263–276. [\[CrossRef\]](#)
8. Chen, L.; Jin, L.; Shang, M. Efficient Loss Landscape Reshaping for Convolutional Neural Networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, 1–15. [\[CrossRef\]](#) [\[PubMed\]](#)
9. Cao, X.; Lou, J.; Liao, B.; Peng, C.; Pu, X.; Khan, A.T.; Pham, D.T.; Li, S. Decomposition based neural dynamics for portfolio management with tradeoffs of risks and profits under transaction costs. *Neural Netw.* **2025**, *184*, 107090. [\[CrossRef\]](#)
10. 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 Netw.* **2025**, *185*, 107213. [\[CrossRef\]](#)
11. Liu, Y.; Li, S.; Lin, X.; Chen, X.; Li, G.; Liu, Y.; Liao, B.; Li, J. QoS-Aware Multi-AIGC Service Orchestration at Edges: An Attention-Diffusion-Aided DRL Method. *IEEE Trans. Cogn. Commun. Netw.* **2025**, *11*, 1078–1090. [\[CrossRef\]](#)
12. Xiang, Y.; Zhou, K.; Sarkheyli-Hägele, A.; Yusoff, Y.; Kang, D.; Zain, A.M. Parallel fault diagnosis using hierarchical fuzzy Petri net by reversible and dynamic decomposition mechanism. *Front. Inf. Technol. Electron. Eng.* **2025**, *26*, 93–108. [\[CrossRef\]](#)
13. Zhong, J.; Zhao, H.; Zhao, Q.; Wang, J. A Knowledge Graph-Based Method for Drug-Drug Interaction Prediction With Contrastive Learning. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2024**, *21*, 2485–2495. [\[CrossRef\]](#)
14. Zhang, Z.; He, Y.; Mai, W.; Luo, Y.; Li, X.; Cheng, Y.; Huang, X.; Lin, R. Convolutional Dynamically Convergent Differential Neural Network for Brain Signal Classification. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, 1–12. [\[CrossRef\]](#)
15. Sun, Q.; Wu, X. A deep learning-based approach for emotional analysis of sports dance. *PeerJ Comput. Sci.* **2023**, *9*, e1441. [\[CrossRef\]](#)
16. Sun, L.; Mo, Z.; Yan, F.; Xia, L.; Shan, F.; Ding, Z.; Song, B.; Gao, W.; Shao, W.; Shi, F.; et al. Adaptive Feature Selection Guided Deep Forest for COVID-19 Classification With Chest CT. *IEEE J. Biomed. Health Inform.* **2020**, *24*, 2798–2805. [\[CrossRef\]](#) [\[PubMed\]](#)
17. Liu, Z.; Wu, X. Structural analysis of the evolution mechanism of online public opinion and its development stages based on machine learning and social network analysis. *Int. J. Comput. Intell. Syst.* **2023**, *16*, 99. [\[CrossRef\]](#)
18. Chu, H.M.; Kong, X.Z.; Liu, J.X.; Zheng, C.H.; Zhang, H. A New Binary Biclustering Algorithm Based on Weight Adjacency Difference Matrix for Analyzing Gene Expression Data. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2023**, *20*, 2802–2809. [\[CrossRef\]](#)

19. Hopfield, J.J.; Tank, D.W. “Neural” computation of decisions in optimization problems. *Biol. Cybern.* **1985**, *52*, 141–152. [[CrossRef](#)]
20. Khan, A.T.; Li, S.; Pham, D.T.; Cao, X. Beetle antennae search reimaged: Leveraging ChatGPT’s AI to forge new frontiers in optimization algorithms. *Cogent Eng.* **2024**, *11*, 2432548. [[CrossRef](#)]
21. Khan, A.T.; Cao, X.; Li, S. Using quadratic interpolated beetle antennae search for higher dimensional portfolio selection under cardinality constraints. *Comput. Econ.* **2023**, *62*, 1413–1435. [[CrossRef](#)]
22. Khan, A.T.; Cao, X.; Liao, B.; Francis, A. Bio-inspired Machine Learning for Distributed Confidential Multi-Portfolio Selection Problem. *Biomimetics* **2022**, *7*, 124. [[CrossRef](#)] [[PubMed](#)]
23. Khan, A.T.; Cao, X.; Brajevic, I.; Stanimirovic, P.S.; Katsikis, V.N.; Li, S. Non-linear Activated Beetle Antennae Search: A novel technique for non-convex tax-aware portfolio optimization problem. *Expert Syst. Appl.* **2022**, *197*, 116631. [[CrossRef](#)]
24. Ijaz, M.U.; Khan, A.T.; Li, S. Bio-Inspired BAS: Run-Time Path-Planning and the Control of Differential Mobile Robot. *EAI Endorsed Trans. AI Robot.* **2022**, *1*, 1–10. [[CrossRef](#)]
25. Khan, A.T.; Cao, X.; Li, S. Dual Beetle Antennae Search system for optimal planning and robust control of 5-link biped robots. *J. Comput. Sci.* **2022**, *60*, 101556. [[CrossRef](#)]
26. Khan, A.T.; Cao, X.; Li, S.; Katsikis, V.N.; Brajevic, I.; Stanimirovic, P.S. Fraud detection in publicly traded U.S firms using Beetle Antennae Search: A machine learning approach. *Expert Syst. Appl.* **2022**, *191*, 116148. [[CrossRef](#)]
27. Khan, A.; Cao, X.; Li, Z.; Li, S. Evolutionary computation based real-time robot arm path-planning using beetle antennae search. *EAI Endorsed Trans. AI Robot.* **2022**, *1*, e3–e3. [[CrossRef](#)]
28. Huang, Z.; Zhang, Z.; Hua, C.; Liao, B.; Li, S. Leveraging enhanced egret swarm optimization algorithm and artificial intelligence-driven prompt strategies for portfolio selection. *Sci. Rep.* **2024**, *14*, 26681. [[CrossRef](#)]
29. Chen, Z.; Li, S.; Khan, A.T.; Mirjalili, S. Competition of tribes and cooperation of members algorithm: An evolutionary computation approach for model free optimization. *Expert Syst. Appl.* **2025**, *265*, 125908. [[CrossRef](#)]
30. Yi, Z.; Cao, X.; Pu, X.; Wu, Y.; Chen, Z.; Khan, A.T.; Francis, A.; Li, S. Fraud detection in capital markets: A novel machine learning approach. *Expert Syst. Appl.* **2023**, *231*, 120760. [[CrossRef](#)]
31. Ye, S.; Zhou, K.; Zain, A.M.; Wang, F.; Yusoff, Y. A modified harmony search algorithm and its applications in weighted fuzzy production rule extraction. *Front. Inf. Technol. Electron. Eng.* **2023**, *24*, 1574–1590. [[CrossRef](#)]
32. Qin, F.; Zain, A.M.; Zhou, K.Q. Harmony search algorithm and related variants: A systematic review. *Swarm Evol. Comput.* **2022**, *74*, 101126. [[CrossRef](#)]
33. Ou, Y.; Qin, F.; Zhou, K.Q.; Yin, P.F.; Mo, L.P.; Mohd Zain, A. An Improved Grey Wolf Optimizer with Multi-Strategies Coverage in Wireless Sensor Networks. *Symmetry* **2024**, *16*, 286. [[CrossRef](#)]
34. Liu, J.; Qu, C.; Zhang, L.; Tang, Y.; Li, J.; Feng, H.; Zeng, X.; Peng, X. A new hybrid algorithm for three-stage gene selection based on whale optimization. *Sci. Rep.* **2023**, *13*, 3783. [[CrossRef](#)] [[PubMed](#)]
35. Liu, J.; Feng, H.; Tang, Y.; Zhang, L.; Qu, C.; Zeng, X.; Peng, X. A novel hybrid algorithm based on Harris Hawks for tumor feature gene selection. *PeerJ Comput. Sci.* **2023**, *9*, e1229. [[CrossRef](#)]
36. Qu, C.; Zhang, L.; Li, J.; Deng, F.; Tang, Y.; Zeng, X.; Peng, X. Improving feature selection performance for classification of gene expression data using Harris Hawks optimizer with variable neighborhood learning. *Brief. Bioinform.* **2021**, *22*, bbab097. [[CrossRef](#)]
37. Wu, W.; Tian, Y.; Jin, T. A label based ant colony algorithm for heterogeneous vehicle routing with mixed backhaul. *Appl. Soft Comput.* **2016**, *47*, 224–234. [[CrossRef](#)]
38. Zhang, Y.; Jiang, D.; Wang, J. A recurrent neural network for solving Sylvester equation with time-varying coefficients. *IEEE Trans. Neural Netw.* **2002**, *13*, 1053–1063. [[CrossRef](#)]
39. Hua, C.; Cao, X.; Liao, B. Real-Time Solutions for Dynamic Complex Matrix Inversion and Chaotic Control Using ODE-Based Neural Computing Methods. *Comput. Intell.* **2025**, *41*, e70042, [<https://onlinelibrary.wiley.com/doi/pdf/10.1111/coin.70042>]. [[CrossRef](#)]
40. Tamoor Khan, A.; Wang, Y.; Wang, T.; Hua, C. Neural Dynamics for Computing and Automation: A Survey. *IEEE Access* **2025**, *13*, 27214–27227. [[CrossRef](#)]
41. Cao, X.; Li, P.; Khan, A.T. A Novel Zeroing Neural Network for the Effective Solution of Supply Chain Inventory Balance Problems. *Computation* **2025**, *13*, 32. [[CrossRef](#)]
42. Xiao, L. A new design formula exploited for accelerating Zhang neural network and its application to time-varying matrix inversion. *Theor. Comput. Sci.* **2016**, *647*, 50–58. [[CrossRef](#)]
43. Liao, B.; Hua, C.; Xu, Q.; Cao, X.; Li, S. Inter-robot management via neighboring robot sensing and measurement using a zeroing neural dynamics approach. *Expert Syst. Appl.* **2024**, *244*, 122938. [[CrossRef](#)]
44. Xiao, L.; Liao, B.; Li, S.; Chen, K. Nonlinear recurrent neural networks for finite-time solution of general time-varying linear matrix equations. *Neural Netw.* **2018**, *98*, 102–113. [[CrossRef](#)] [[PubMed](#)]
45. Ding, L.; Xiao, L.; Liao, B.; Lu, R.; Peng, H. An Improved Recurrent Neural Network for Complex-Valued Systems of Linear Equation and Its Application to Robotic Motion Tracking. *Front. Neurobot.* **2017**, *11*, 45. [[CrossRef](#)]

46. Xiang, Q.; Gong, H.; Hua, C. A new discrete-time denoising complex neurodynamics applied to dynamic complex generalized inverse matrices. *J. Supercomput.* **2025**, *81*, 1–25. [\[CrossRef\]](#)
47. Liao, B.; Wang, Y.; Li, J.; Guo, D.; He, Y. Harmonic Noise-Tolerant ZNN for Dynamic Matrix Pseudoinversion and Its Application to Robot Manipulator. *Front. Neurobot.* **2022**, *16*, 928636. [\[CrossRef\]](#)
48. Xiang, Q.; Liao, B.; Xiao, L.; Lin, L.; Li, S. Discrete-time noise-tolerant Zhang neural network for dynamic matrix pseudoinversion. *Soft Comput.* **2019**, *23*, 755–766. [\[CrossRef\]](#)
49. 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. [\[CrossRef\]](#)
50. Dai, L.; Xu, H.; Zhang, Y.; Liao, B. Norm-based zeroing neural dynamics for time-variant non-linear equations. *Caa Trans. Intell. Technol.* **2024**, *9*, 1561–1571. [\[CrossRef\]](#)
51. Xiao, L.; Lu, R. Finite-time solution to nonlinear equation using recurrent neural dynamics with a specially-constructed activation function. *Neurocomputing* **2015**, *151*, 246–251. [\[CrossRef\]](#)
52. Zhang, Z.; Zheng, L.; Weng, J.; Mao, Y.; Lu, W.; Xiao, L. A New Varying-Parameter Recurrent Neural-Network for Online Solution of Time-Varying Sylvester Equation. *IEEE Trans. Cybern.* **2018**, *48*, 3135–3148. [\[CrossRef\]](#) [\[PubMed\]](#)
53. Xiao, L.; Liao, B. A convergence-accelerated Zhang neural network and its solution application to Lyapunov equation. *Neurocomputing* **2016**, *193*, 213–218. [\[CrossRef\]](#)
54. Xiao, L.; Dai, J.; Lu, R.; Li, S.; Li, J.; Wang, S. Design and Comprehensive Analysis of a Noise-Tolerant ZNN Model With Limited-Time Convergence for Time-Dependent Nonlinear Minimization. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *31*, 5339–5348. [\[CrossRef\]](#)
55. Luo, Y.; Li, X.; Li, Z.; Xie, J.; Zhang, Z.; Li, X. A Novel Swarm-Exploring Neurodynamic Network for Obtaining Global Optimal Solutions to Nonconvex Nonlinear Programming Problems. *IEEE Trans. Cybern.* **2024**, *54*, 5866–5876. [\[CrossRef\]](#)
56. Zhang, Z.; Zheng, L.; Li, L.; Deng, X.; Xiao, L.; Huang, G. A new finite-time varying-parameter convergent-differential neural-network for solving nonlinear and nonconvex optimization problems. *Neurocomputing* **2018**, *319*, 74–83. [\[CrossRef\]](#)
57. Wei, L.; Jin, L. Collaborative Neural Solution for Time-Varying Nonconvex Optimization With Noise Rejection. *IEEE Trans. Emerg. Top. Comput. Intell.* **2024**, *8*, 2935–2948. [\[CrossRef\]](#)
58. Zhang, Z.; Sun, X.; Li, X.; Liu, Y. An adaptive variable-parameter dynamic learning network for solving constrained time-varying QP problem. *Neural Netw.* **2025**, *184*, 106968. [\[CrossRef\]](#)
59. Zhang, Z.; Yu, H.; Ren, X.; Luo, Y. A swarm exploring neural dynamics method for solving convex multi-objective optimization problem. *Neurocomputing* **2024**, *601*, 128203. [\[CrossRef\]](#)
60. Xiao, L.; Li, K.; Duan, M. Computing Time-Varying Quadratic Optimization with Finite-Time Convergence and Noise Tolerance: A Unified Framework for Zeroing Neural Network. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *30*, 3360–3369. [\[CrossRef\]](#)
61. Jin, L.; Liao, B.; Liu, M.; Xiao, L.; Guo, D.; Yan, X. Different-Level Simultaneous Minimization Scheme for Fault Tolerance of Redundant Manipulator Aided with Discrete-Time Recurrent Neural Network. *Front. Neurobot.* **2017**, *11*, 50. [\[CrossRef\]](#)
62. Liao, B.; Zhang, Y.; Jin, L. Taylor $O(h^3)$ Discretization of ZNN Models for Dynamic Equality-Constrained Quadratic Programming With Application to Manipulators. *IEEE Trans. Neural Netw. Learn. Syst.* **2016**, *27*, 225–237. [\[CrossRef\]](#) [\[PubMed\]](#)
63. Xiao, L. A nonlinearly-activated neurodynamic model and its finite-time solution to equality-constrained quadratic optimization with nonstationary coefficients. *Appl. Soft Comput.* **2016**, *40*, 252–259. [\[CrossRef\]](#)
64. Liu, M.; Jiang, Q.; Li, H.; Cao, X.; Lv, X. Finite-time-convergent support vector neural dynamics for classification. *Neurocomputing* **2025**, *617*, 128810. [\[CrossRef\]](#)
65. Liu, M.; Liao, B.; Ding, L.; Xiao, L. Performance analyses of recurrent neural network models exploited for online time-varying nonlinear optimization. *Comput. Sci. Inf. Syst.* **2016**, *13*, 691–705. [\[CrossRef\]](#)
66. Zhang, Z.; Zhu, M.; Ren, X. Double center swarm exploring varying parameter neurodynamic network for non-convex nonlinear programming. *Neurocomputing* **2025**, *619*, 129156. [\[CrossRef\]](#)
67. Lv, X.; Xiao, L.; Tan, Z.; Yang, Z. Wsbp function activated Zhang dynamic with finite-time convergence applied to Lyapunov equation. *Neurocomputing* **2018**, *314*, 310–315. [\[CrossRef\]](#)
68. Liao, B.; Zhang, Y. From different ZFs to different ZNN models accelerated via Li activation functions to finite-time convergence for time-varying matrix pseudoinversion. *Neurocomputing* **2014**, *133*, 512–522. [\[CrossRef\]](#)
69. Dai, J.; Yang, X.; Xiao, L.; Jia, L.; Li, Y. ZNN with Fuzzy Adaptive Activation Functions and Its Application to Time-Varying Linear Matrix Equation. *IEEE Trans. Ind. Inform.* **2022**, *18*, 2560–2570. [\[CrossRef\]](#)
70. Zhang, Y.N.; Li, Z.; Guo, D.S.; Chen, K.; Chen, P. Superior robustness of using power-sigmoid activation functions in Z-type models for time-varying problems solving. In Proceedings of the 2013 International Conference on Machine Learning and Cybernetics, Tianjin, China, 14–17 July 2013; Volume 2, pp. 759–764. [\[CrossRef\]](#)
71. Jin, J.; Zhu, J.; Gong, J.; Chen, W. Novel activation functions-based ZNN models for fixed-time solving dynamirc Sylvester equation. *Neural Comput. Appl.* **2022**, *34*, 14297–14315. [\[CrossRef\]](#)

72. Li, S.; Chen, S.; Liu, B. Accelerating a recurrent neural network to finite-time convergence for solving time-varying Sylvester equation by using a sign-bi-power activation function. *Neural Process. Lett.* **2013**, *37*, 189–205. [\[CrossRef\]](#)
73. Zhang, Y.; Jin, L.; Ke, Z. Superior performance of using hyperbolic sine activation functions in ZNN illustrated via time-varying matrix square roots finding. *Comput. Sci. Inf. Syst.* **2012**, *9*, 1603–1625. [\[CrossRef\]](#)
74. Yang, Y.; Zhang, Y. Superior robustness of power-sum activation functions in Zhang neural networks for time-varying quadratic programs perturbed with large implementation errors. *Neural Comput. Appl.* **2013**, *22*, 175–185. [\[CrossRef\]](#)
75. Li, H.; Liao, B.; Li, J.; Li, S. A Survey on Biomimetic and Intelligent Algorithms with Applications. *Biomimetics* **2024**, *9*, 453. [\[CrossRef\]](#)
76. Li, H.; Zhang, Z.; Liao, B.; Hua, C. An improving integration-enhanced ZNN for solving time-varying polytope distance problems with inequality constraint. *Neural Comput. Appl.* **2024**, *36*, 18237–18250. [\[CrossRef\]](#)
77. Wang, T.; Zhang, Z.; Huang, Y.; Liao, B.; Li, S. Applications of Zeroing Neural Networks: A Survey. *IEEE Access* **2024**, *12*, 51346–51363. [\[CrossRef\]](#)
78. Hua, C.; Cao, X.; Liao, B.; Li, S. Advances on intelligent algorithms for scientific computing: An overview. *Front. Neurobot.* **2023**, *17*, 1190977. [\[CrossRef\]](#) [\[PubMed\]](#)
79. Jin, L.; Li, S.; Liao, B.; Zhang, Z. Zeroing neural networks: A survey. *Neurocomputing* **2017**, *267*, 597–604. [\[CrossRef\]](#)
80. 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**, *1*. [\[CrossRef\]](#)
81. Xiao, L.; Li, L.; Tao, J.; Li, W. A predefined-time and anti-noise varying-parameter ZNN model for solving time-varying complex Stein equations. *Neurocomputing* **2023**, *526*, 158–168. [\[CrossRef\]](#)
82. Yan, J.; Jin, L.; Hu, B. Data-Driven Model Predictive Control for Redundant Manipulators With Unknown Model. *IEEE Trans. Cybern.* **2024**, *54*, 5901–5911. [\[CrossRef\]](#)
83. Zhang, Z.; Cao, Z.; Li, X. Neural Dynamic Fault-Tolerant Scheme for Collaborative Motion Planning of Dual-Redundant Robot Manipulators. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, 1–13. [\[CrossRef\]](#) [\[PubMed\]](#)
84. Xu, H.; Zhang, B. A Two-Phase Algorithm for Reliable and Energy-Efficient Heterogeneous Embedded Systems. *IEICE Trans. Inf. Syst.* **2024**, *107*, 1285–1296. [\[CrossRef\]](#)
85. Jin, L.; Huang, R.; Liu, M.; Ma, X. Cerebellum-Inspired Learning and Control Scheme for Redundant Manipulators at Joint Velocity Level. *IEEE Trans. Cybern.* **2024**, *54*, 6297–6306. [\[CrossRef\]](#) [\[PubMed\]](#)
86. Xiao, L.; Zhang, Y.; Dai, J.; Chen, K.; Yang, S.; Li, W.; Liao, B.; Ding, L.; Li, J. A new noise-tolerant and predefined-time ZNN model for time-dependent matrix inversion. *Neural Netw.* **2019**, *117*, 124–134. [\[CrossRef\]](#)
87. Zhang, Y.; Li, S.; Kadry, S.; Liao, B. Recurrent Neural Network for Kinematic Control of Redundant Manipulators With Periodic Input Disturbance and Physical Constraints. *IEEE Trans. Cybern.* **2019**, *49*, 4194–4205. [\[CrossRef\]](#)
88. Xiao, L.; Zhang, Y.; Liao, B.; Zhang, Z.; Ding, L.; Jin, L. A Velocity-Level Bi-Criteria Optimization Scheme for Coordinated Path Tracking of Dual Robot Manipulators Using Recurrent Neural Network. *Front. Neurobot.* **2017**, *11*, 47. [\[CrossRef\]](#)
89. Li, X.; Ren, X.; Zhang, Z.; Guo, J.; Luo, Y.; Mai, J.; Liao, B. A varying-parameter complementary neural network for multi-robot tracking and formation via model predictive control. *Neurocomputing* **2024**, *609*, 128384. [\[CrossRef\]](#)
90. Zhang, Y.; Li, S.; Weng, J.; Liao, B. GNN Model for Time-Varying Matrix Inversion With Robust Finite-Time Convergence. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, *35*, 559–569. [\[CrossRef\]](#)
91. Xiao, L.; Li, K.; Tan, Z.; Zhang, Z.; Liao, B.; Chen, K.; Jin, L.; Li, S. Nonlinear gradient neural network for solving system of linear equations. *Inf. Process. Lett.* **2019**, *142*, 35–40. [\[CrossRef\]](#)
92. Xiao, L. A finite-time convergent neural dynamics for online solution of time-varying linear complex matrix equation. *Neurocomputing* **2015**, *167*, 254–259. [\[CrossRef\]](#)
93. Jin, L.; Zhang, Y.; Li, S. Integration-Enhanced Zhang Neural Network for Real-Time-Varying Matrix Inversion in the Presence of Various Kinds of Noises. *IEEE Trans. Neural Netw. Learn. Syst.* **2016**, *27*, 2615–2627. [\[CrossRef\]](#)
94. Liao, B.; Xiang, Q.; Li, S. Bounded Z-type neurodynamics with limited-time convergence and noise tolerance for calculating time-dependent Lyapunov equation. *Neurocomputing* **2019**, *325*, 234–241. [\[CrossRef\]](#)
95. Lei, Y.; Luo, J.; Chen, T.; Ding, L.; Liao, B.; Xia, G.; Dai, Z. Nonlinearly Activated IEZNN Model for Solving Time-Varying Sylvester Equation. *IEEE Access* **2022**, *10*, 121520–121530. [\[CrossRef\]](#)
96. Xiao, L.; He, Y.; Wang, Y.; Dai, J.; Wang, R.; Tang, W. A Segmented Variable-Parameter ZNN for Dynamic Quadratic Minimization with Improved Convergence and Robustness. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *34*, 2413–2424. [\[CrossRef\]](#) [\[PubMed\]](#)
97. Xiao, L.; Li, S.; Yang, J.; Zhang, Z. A new recurrent neural network with noise-tolerance and finite-time convergence for dynamic quadratic minimization. *Neurocomputing* **2018**, *285*, 125–132. [\[CrossRef\]](#)
98. Liao, B.; Hua, C.; Cao, X.; Katsikis, V.N.; Li, S. Complex Noise-Resistant Zeroing Neural Network for Computing Complex Time-Dependent Lyapunov Equation. *Mathematics* **2022**, *10*, 2817. [\[CrossRef\]](#)
99. Liao, B.; Han, L.; Cao, X.; Li, S.; Li, J. Double integral-enhanced Zeroing neural network with linear noise rejection for time-varying matrix inverse. *Caai Trans. Intell. Technol.* **2024**, *9*, 197–210. [\[CrossRef\]](#)

100. Li, J.; Qu, L.; Zhu, Y.; Li, Z.; Liao, B. A Novel Zeroing Neural Network for Time-varying Matrix Pseudoinversion in the Presence of Linear Noises. *Tsinghua Sci. Technol.* **2024**. [\[CrossRef\]](#)
101. Xiao, L.; Tan, H.; Jia, L.; Dai, J.; Zhang, Y. New error function designs for finite-time ZNN models with application to dynamic matrix inversion. *Neurocomputing* **2020**, *402*, 395–408. [\[CrossRef\]](#)
102. Xiao, L.; Yi, Q.; Dai, J.; Li, K.; Hu, Z. Design and analysis of new complex zeroing neural network for a set of dynamic complex linear equations. *Neurocomputing* **2019**, *363*, 171–181. [\[CrossRef\]](#)
103. Xiao, L.; Cao, P.; Song, W.; Luo, L.; Tang, W. A Fixed-Time Noise-Tolerance ZNN Model for Time-Variant Inequality-Constrained Quaternion Matrix Least-Squares Problem. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, *35*, 10503–10512. [\[CrossRef\]](#) [\[PubMed\]](#)
104. Li, W.; Liao, B.; Xiao, L.; Lu, R. A recurrent neural network with predefined-time convergence and improved noise tolerance for dynamic matrix square root finding. *Neurocomputing* **2019**, *337*, 262–273. [\[CrossRef\]](#)
105. Li, W.; Guo, C.; Ma, X.; Pan, Y. A Strictly Predefined-Time Convergent and Noise-Tolerant Neural Model for Solving Linear Equations With Robotic Applications. *IEEE Trans. Ind. Electron.* **2024**, *71*, 798–809. [\[CrossRef\]](#)
106. Xiao, L.; Zhang, Z.; Zhang, Z.; Li, W.; Li, S. Design, verification and robotic application of a novel recurrent neural network for computing dynamic Sylvester equation. *Neural Netw.* **2018**, *105*, 185–196. [\[CrossRef\]](#) [\[PubMed\]](#)
107. Zhang, Y.; Ge, S. Design and analysis of a general recurrent neural network model for time-varying matrix inversion. *IEEE Trans. Neural Netw.* **2005**, *16*, 1477–1490. [\[CrossRef\]](#)
108. Zhang, Y.; Yi, C.; Guo, D.; Zheng, J. Comparison on Zhang neural dynamics and gradient-based neural dynamics for online solution of nonlinear time-varying equation. *Neural Comput. Appl.* **2011**, *20*, 1–7. [\[CrossRef\]](#)
109. Li, S.; Li, Y. Nonlinearly Activated Neural Network for Solving Time-Varying Complex Sylvester Equation. *IEEE Trans. Cybern.* **2014**, *44*, 1397–1407. [\[CrossRef\]](#)
110. Sun, M.; Zhang, Y.; Wang, L.; Wu, Y.; Zhong, G. Time-variant quadratic programming solving by using finitely-activated RNN models with exact settling time. *Neural Comput. Appl.* **2025**, *37*, 6067–6084. [\[CrossRef\]](#)
111. Zhang, Y.; Liao, B.; Geng, G. GNN Model With Robust Finite-Time Convergence for Time-Varying Systems of Linear Equations. *IEEE Trans. Syst. Man Cybern. Syst.* **2024**, *54*, 4786–4797. [\[CrossRef\]](#)
112. Jin, J.; Zhu, J.; Zhao, L.; Chen, L. A fixed-time convergent and noise-tolerant zeroing neural network for online solution of time-varying matrix inversion. *Appl. Soft Comput.* **2022**, *130*, 109691. [\[CrossRef\]](#)
113. Sun, Z.; Tang, S.; Zhang, J.; Yu, J. Nonconvex Noise-Tolerant Neural Model for Repetitive Motion of Omnidirectional Mobile Manipulators. *IEEE/CAA J. Autom. Sin.* **2023**, *10*, 1766–1768. [\[CrossRef\]](#)
114. Cao, M.; Xiao, L.; Zuo, Q.; Tan, P.; He, Y.; Gao, X. A Fixed-Time Robust ZNN Model with Adaptive Parameters for Redundancy Resolution of Manipulators. *IEEE Trans. Emerg. Top. Comput. Intell.* **2024**, *8*, 3886–3898. [\[CrossRef\]](#)
115. Xiao, L.; Luo, J.; Li, J.; Jia, L.; Li, J. Fixed-Time Consensus for Multiagent Systems Under Switching Topology: A Distributed Zeroing Neural Network-Based Method. *IEEE Syst. Man, Cybern. Mag.* **2024**, *10*, 44–55. [\[CrossRef\]](#)
116. Luo, J.; Xiao, L.; Cao, P.; Li, X. A new class of robust and predefined-time consensus protocol based on noise-tolerant ZNN models. *Appl. Soft Comput.* **2023**, *145*, 110550. [\[CrossRef\]](#)
117. Jia, L.; Xiao, L.; Dai, J.; Wang, Y. Intensive Noise-Tolerant Zeroing Neural Network Based on a Novel Fuzzy Control Approach. *IEEE Trans. Fuzzy Syst.* **2023**, *31*, 4350–4360. [\[CrossRef\]](#)
118. Li, S.; Ma, C. A novel predefined-time noise-tolerant zeroing neural network for solving time-varying generalized linear matrix equations. *J. Frankl. Inst.* **2023**, *360*, 11788–11808. [\[CrossRef\]](#)
119. Liu, X.; Zhao, L.; Jin, J. A noise-tolerant fuzzy-type zeroing neural network for robust synchronization of chaotic systems. *Concurr. Comput. Pract. Exp.* **2024**, *36*, e8218. [\[CrossRef\]](#)
120. Xiao, L.; He, Y.; Dai, J.; Liu, X.; Liao, B.; Tan, H. A Variable-Parameter Noise-Tolerant Zeroing Neural Network for Time-Variant Matrix Inversion With Guaranteed Robustness. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *33*, 1535–1545. [\[CrossRef\]](#)
121. Zhang, Z.; Fu, T.; Yan, Z.; Jin, L.; Xiao, L.; Sun, Y.; Yu, Z.; Li, Y. A Varying-Parameter Convergent-Differential Neural Network for Solving Joint-Angular-Drift Problems of Redundant Robot Manipulators. *IEEE/ASME Trans. Mechatron.* **2018**, *23*, 679–689. [\[CrossRef\]](#)
122. Zhang, Z.; Zheng, L.; Wang, M. An exponential-enhanced-type varying-parameter RNN for solving time-varying matrix inversion. *Neurocomputing* **2019**, *338*, 126–138. [\[CrossRef\]](#)
123. Zhang, Z.; Zheng, L. A Complex Varying-Parameter Convergent-Differential Neural-Network for Solving Online Time-Varying Complex Sylvester Equation. *IEEE Trans. Cybern.* **2019**, *49*, 3627–3639. [\[CrossRef\]](#) [\[PubMed\]](#)
124. Tan, Z.; Li, W.; Xiao, L.; Hu, Y. New Varying-Parameter ZNN Models With Finite-Time Convergence and Noise Suppression for Time-Varying Matrix Moore–Penrose Inversion. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *31*, 2980–2992. [\[CrossRef\]](#)
125. Xiao, L.; Zhang, Y.; Dai, J.; Zuo, Q.; Wang, S. Comprehensive Analysis of a New Varying Parameter Zeroing Neural Network for Time Varying Matrix Inversion. *IEEE Trans. Ind. Inform.* **2021**, *17*, 1604–1613. [\[CrossRef\]](#)
126. Xiao, L.; He, Y. A Noise-Suppression ZNN Model with New Variable Parameter for Dynamic Sylvester Equation. *IEEE Trans. Ind. Inform.* **2021**, *17*, 7513–7522. [\[CrossRef\]](#)

127. Xiao, L.; Huang, W.; Li, X.; Sun, F.; Liao, Q.; Jia, L.; Li, J.; Liu, S. ZNNs With a Varying-Parameter Design Formula for Dynamic Sylvester Quaternion Matrix Equation. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *34*, 9981–9991. [\[CrossRef\]](#)
128. Xiao, L.; Zhang, Y.; Huang, W.; Jia, L.; Gao, X. A Dynamic Parameter Noise-Tolerant Zeroing Neural Network for Time-Varying Quaternion Matrix Equation With Applications. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, *35*, 8205–8214. [\[CrossRef\]](#)
129. Xiao, L.; Li, X.; Cao, P.; He, Y.; Tang, W.; Li, J.; Wang, Y. A Dynamic-Varying Parameter Enhanced ZNN Model for Solving Time-Varying Complex-Valued Tensor Inversion With Its Application to Image Encryption. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, *35*, 13681–13690. [\[CrossRef\]](#)
130. Xiao, L.; Zhang, Y.; Solving time-varying inverse kinematics problem of wheeled mobile manipulators using Zhang neural network with exponential convergence. *Nonlinear Dynamics*. **2014**, *76*, 1543–1559. [\[CrossRef\]](#)
131. Tang, Z.; Zhang, Y.; Ming, L. Novel Snap-Layer MMPC Scheme via Neural Dynamics Equivalency and Solver for Redundant Robot Arms With Five-Layer Physical Limits. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, *36*, 3534–3546. [\[CrossRef\]](#)
132. Khan, A.T.; Li, S. Smart surgical control under RCM constraint using bio-inspired network. *Neurocomputing* **2022**, *470*, 121–129. [\[CrossRef\]](#)
133. Khan, A.T.; Li, S.; Cao, X. Human guided cooperative robotic agents in smart home using beetle antennae search. *Sci. China Inf. Sci.* **2022**, *65*, 122204. [\[CrossRef\]](#)
134. Liu, M.; Li, Y.; Chen, Y.; Qi, Y.; Jin, L. A Distributed Competitive and Collaborative Coordination for Multirobot Systems. *IEEE Trans. Mob. Comput.* **2024**, *23*, 11436–11448. [\[CrossRef\]](#)
135. Khan, A.T.; Li, S.; Li, Z. Obstacle avoidance and model-free tracking control for home automation using bio-inspired approach. *Adv. Control Appl.* **2022**, *4*, e63. [\[CrossRef\]](#)
136. Tang, Z.; Zhang, Y. Refined Self-Motion Scheme With Zero Initial Velocities and Time-Varying Physical Limits via Zhang Neurodynamics Equivalency. *Front. Neurobot.* **2022**, *16*, 945346. [\[CrossRef\]](#)
137. 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. Cybern.* **2020**, *50*, 3195–3207. [\[CrossRef\]](#)
138. Wang, C.; Wang, Y.; Yuan, Y.; Peng, S.; Li, G.; Yin, P. Joint computation offloading and resource allocation for end-edge collaboration in internet of vehicles via multi-agent reinforcement learning. *Neural Netw.* **2024**, *179*, 106621. [\[CrossRef\]](#)
139. Lorenz, E.N. Deterministic nonperiodic flow. *J. Atmos. Sci.* **1963**, *20*, 130–141. [\[CrossRef\]](#)
140. Tuna, M.; Fidan, C.B. Electronic circuit design, implementation and FPGA-based realization of a new 3D chaotic system with single equilibrium point. *Optik* **2016**, *127*, 11786–11799. [\[CrossRef\]](#)
141. Yu, H.; Cai, G.; Li, Y. Dynamic analysis and control of a new hyperchaotic finance system. *Nonlinear Dyn.* **2012**, *67*, 2171–2182. [\[CrossRef\]](#)
142. Brindley, J.; Kapitaniak, T.; Kocarev, L. Controlling chaos by chaos in geophysical systems. *Geophys. Res. Lett.* **1995**, *22*, 1257–1260. [\[CrossRef\]](#)
143. Naderi, B.; Kheiri, H. Exponential synchronization of chaotic system and application in secure communication. *Optik* **2016**, *127*, 2407–2412. [\[CrossRef\]](#)
144. Aoun, S.B.; Derbel, N.; Jerbi, H.; Simos, T.E.; Mourtas, S.D.; Katsikis, V.N. A quaternion Sylvester equation solver through noise-resilient Zeroing neural networks with application to control the SFM chaotic system. *AIMS Math.* **2023**, *8*, 27376–27395. [\[CrossRef\]](#)
145. Xiao, L.; Li, L.; Cao, P.; He, Y. A fixed-time robust controller based on Zeroing neural network for generalized projective synchronization of chaotic systems. *Chaos Solitons Fractals* **2023**, *169*, 113279. [\[CrossRef\]](#)
146. 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. Emerg. Top. Comput. Intell.* **2024**, *8*, 364–376. [\[CrossRef\]](#)
147. Zadeh, L. Fuzzy sets. *Inf. Control* **1965**, *8*, 338–353. [\[CrossRef\]](#)
148. Jia, L.; Xiao, L.; Dai, J.; Cao, Y. A Novel Fuzzy-Power Zeroing Neural Network Model for Time-Variant Matrix Moore–Penrose Inversion With Guaranteed Performance. *IEEE Trans. Fuzzy Syst.* **2021**, *29*, 2603–2611. [\[CrossRef\]](#)
149. Yu, F.; Zhang, Z.; Shen, H.; Huang, Y.; Cai, S.; Du, S. FPGA implementation and image encryption application of a new PRNG based on a memristive Hopfield neural network with a special activation gradient. *Chin. Phys. B* **2022**, *31*, 020505. [\[CrossRef\]](#)
150. Charif, F.; Benchabane, A.; Djedi, N.; Taleb-Ahmed, A. Horn & Schunck meets a discrete Zhang neural networks for computing 2D optical flow. In Proceedings of the International Conference on Electronics & Oil: From Theory to Applications, Ouargla, Algeria, 5–6 March 2013.
151. Zlateski, A.; Lee, K.; Seung, H.S. ZNN—A Fast and Scalable Algorithm for Training 3D Convolutional Networks on Multi-core and Many-Core Shared Memory Machines. In Proceedings of the 2016 IEEE International Parallel and Distributed Processing Symposium (IPDPS), Chicago, IL, USA, 23–27 May 2016; pp. 801–811.
152. Benchabane, A.; Bennia, A.; Charif, F.; Taleb-Ahmed, A. Multi-dimensional Capon spectral estimation using discrete Zhang neural networks. *Multidimens. Syst. Signal Process.* **2013**, *24*, 583–598. [\[CrossRef\]](#)

153. Zhang, Y.; Yan, X.; Liao, B.; Zhang, Y.; Ding, Y. Z-type control of populations for Lotka–Volterra model with exponential convergence. *Math. Biosci.* **2016**, *272*, 15–23. [\[CrossRef\]](#)
154. Zhang, Y.; Qiu, B.; Liao, B.; Yang, Z. Control of pendulum tracking (including swinging up) of IPC system using zeroing-gradient method. *Nonlinear Dyn.* **2017**, *89*, 1–25. [\[CrossRef\]](#)
155. Guo, J.; Qiu, B.; Yang, M.; Zhang, Y. Zhang Neural Network Model for Solving LQ Decomposition Problem of Dynamic Matrix With Application to Mobile Object Localization. In Proceedings of the 2021 International Joint Conference on Neural Networks (IJCNN), Shenzhen, China, 18–22 July 2021; pp. 1–6. [\[CrossRef\]](#)
156. Xiao, L.; He, Y.; Li, Y.; Dai, J. Design and Analysis of Two Nonlinear ZNN Models for Matrix LR and QR Factorization with Application to 3-D Moving Target Location. *IEEE Trans. Ind. Inform.* **2023**, *19*, 7424–7434. [\[CrossRef\]](#)
157. He, Y.; Xiao, L.; Sun, F.; Wang, Y. A variable-parameter ZNN with predefined-time convergence for dynamic complex-valued Lyapunov equation and its application to AOA positioning. *Appl. Soft Comput.* **2022**, *130*, 109703. [\[CrossRef\]](#)
158. Sun, Z.; Liu, Y.; Wei, L.; Liu, K.; Jin, L.; Ren, L. Two DTZNN Models of $O(\tau^4)$ Pattern for Online Solving Dynamic System of Linear Equations: Application to Manipulator Motion Generation. *IEEE Access* **2020**, *8*, 36624–36638. [\[CrossRef\]](#)
159. Xiao, L.; Liao, B.; Li, S.; Zhang, Z.; Ding, L.; Jin, L. Design and Analysis of FTZNN Applied to the Real-Time Solution of a Nonstationary Lyapunov Equation and Tracking Control of a Wheeled Mobile Manipulator. *IEEE Trans. Ind. Inform.* **2018**, *14*, 98–105. [\[CrossRef\]](#)
160. Jin, J.; Gong, J. An interference-tolerant fast convergence Zeroing neural network for dynamic matrix inversion and its application to mobile manipulator path tracking. *Alex. Eng. J.* **2021**, *60*, 659–669. [\[CrossRef\]](#)
161. Chen, D.; Zhang, Y. Robust Zeroing Neural-Dynamics and Its Time-Varying Disturbances Suppression Model Applied to Mobile Robot Manipulators. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *29*, 4385–4397. [\[CrossRef\]](#)
162. Jin, L.; Zhang, F.; Liu, M.; Xu, S.S.D. Finite-Time Model Predictive Tracking Control of Position and Orientation for Redundant Manipulators. *IEEE Trans. Ind. Electron.* **2023**, *70*, 6017–6026. [\[CrossRef\]](#)
163. Chen, D.; Li, S.; Wu, Q. A Novel Supertwisting Zeroing Neural Network With Application to Mobile Robot Manipulators. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *32*, 1776–1787. [\[CrossRef\]](#)
164. Yang, M.; Zhang, Y.; Tan, N.; Hu, H. Concise Discrete ZNN Controllers for End-Effector Tracking and Obstacle Avoidance of Redundant Manipulators. *IEEE Trans. Ind. Inform.* **2022**, *18*, 3193–3202. [\[CrossRef\]](#)
165. Jin, L.; Zhao, J.; Chen, L.; Li, S. Collective Neural Dynamics for Sparse Motion Planning of Redundant Manipulators Without Hessian Matrix Inversion. *IEEE Trans. Neural Netw. Learn. Syst.* **2025**, *36*, 4326–4335. [\[CrossRef\]](#) [\[PubMed\]](#)
166. Qiu, B.; Guo, J.; Mao, M.; Tan, N. A Fuzzy-Enhanced Robust DZNN Model for Future Multiconstrained Nonlinear Optimization With Robotic Manipulator Control. *IEEE Trans. Fuzzy Syst.* **2024**, *32*, 160–173. [\[CrossRef\]](#)
167. Liao, B.; Wang, T.; Cao, X.; Hua, C.; Li, S. A Novel Zeroing Neural Dynamics for Real-Time Management of Multi-vehicle Cooperation. *IEEE Trans. Intell. Veh.* **2024**, 1–16. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.