An Accelerated Dual-Integral Structure Zeroing Neural Network Resistant to Linear Noise for Dynamic Complex Matrix Inversion

FeiXiang Yang, TingLei Wang and Yun Huang *

College of Computer Science and Engineering, Jishou University, Jishou 416000, China; 2021401395@stu.jsu.edu.cn (F.Y.); wtl@stu.jsu.edu.cn (T.W.)
* Correspondence: huangyun@jsu.edu.cn; Tel.: +86-139-0744-7310

Abstract: The problem of inverting dynamic complex matrices remains a central and intricate challenge that has garnered significant attention in scientific and mathematical research. The zeroing neural network (ZNN) has been a notable approach, utilizing time derivatives for real-time solutions in noiseless settings. However, real-world disturbances pose a significant challenge to a ZNN’s convergence. We design an accelerated dual-integral structure zeroing neural network (ADISZNN), which can enhance convergence and restrict linear noise, particularly in complex domains. Based on the Lyapunov principle, theoretical analysis proves the convergence and robustness of ADISZNN. We have selectively integrated the SBPAF activation function, and through theoretical dissection and comparative experimental validation we have affirmed the efficacy and accuracy of our activation function selection strategy. After conducting numerous experiments, we discovered oscillations and improved the model accordingly, resulting in the ADISZNN-Stable model. This advanced model surpasses current models in both linear noisy and noise-free environments, delivering a more rapid and stable convergence, marking a significant leap forward in the field.

Keywords: dynamic complex matrix inversion; zeroing neural network; linear noise; activation function; residual fluctuations

MSC: 34A55

1. Introduction

Matrix inversion is a fundamental and crucial problem encountered in various domains [1–5], including mathematics and engineering, chaotic systems [1–3], and robotic dynamics [5]. Numerous methods exist for solving this problem, primarily categorized into two approaches. The first is numerical computation methods, such as Newton’s iterative method [6,7], which, though fundamentally serial in nature, suffer from slow computation speed and high resource consumption, rendering them ineffective for efficiently computing the inverse of high-dimensional matrices. Another approach is neural-network-based methods, inherently parallel in computation, such as gradient neural networks (GNNs) [8–11], renowned for their high computational accuracy and exponential convergence. However, GNNs have their own set of challenges and limitations, particularly when it comes to handling dynamic or time-varying data.

Introduced 20 years ago, the ZNN model proposed by Zhang et al. [12] is a specialized neural network architecture that is more adaptive and efficient for solving real-time matrix inversion problems. However, ZNNs are only applicable in ideal noise-free environments. In reality, various types of noise exist, impairing ZNNs’ convergence to theoretical values. Dynamic matrix inversion encompasses two domains: dynamic real matrix inversion and dynamic complex matrix inversion. According to PID control theory [13], the Integration-Enhanced Zhang Neural Network model (IEZNN) [14], proposed by Jin et al., restricts noise interference and is employed to address dynamic real matrix inversion problems,
demonstrating commendable noise restriction capabilities and convergence performance through theoretical analysis and experimental validation.

The applications of dynamic matrix inversion in the complex domain span various scientific and engineering disciplines [15–19]. Mathematical models in the complex domain are crucial for describing phenomena such as control systems [16], signal processing [17], and optical systems [18]. Therefore, this paper focuses on the problem of dynamic complex matrix inversion.

Expanding upon previous research on zeroing neural network (ZNN) models, Zhang et al. proposed a complex-valued ZNN (CVZNN) to address DCMI problems [20]. Xiao et al. introduced a complex-valued noise-tolerant ZNN (CVNTZNN) model [21] aimed at restricting real-world noise interference, inspired by the noise reduction principle of integral-based zeroing neural networks. However, the CVNTZNN model struggles to effectively restrict linear noise. Recently, Hua et al. introduced the dual-integral structure zeroing neural network (DISZNN) model [22]. Leveraging its inherent dual-integral structure, the DISZNN model demonstrates superior performance in restricting linear noise for DCMI problems, as evidenced by theoretical analysis based on Laplace transforms. Moreover, numerous studies suggest that integrating activation functions (AFs) into ZNN models enhances noise tolerance and convergence performance [23–31]. Therefore, this paper proposes an accelerated dual-integral structure zeroing neural network (ADISZNN) by combining AFs with the DISZNN model to enhance its noise restriction capabilities against linear noise and accelerate convergence. It is noteworthy that the DISZNN model is restructured in this study, and the convergence and robustness of the ADISZNN model are theoretically analyzed and demonstrated in a different manner.

This article delineates the following scholarly contributions: The integration of DISZNN with a novel activation function has culminated in the development of an accelerated dual-integral structure zeroing neural network (ADISZNN). This model utilizes a dual-integral structure and activation function, demonstrating improved convergence speed. This means that the model’s computed results can more quickly approach the theoretical inverse of the target matrix. Oscillatory fluctuations observed in the steady-state residual error of ADISZNN, particularly with the SBPAF function selected for the model’s activation, have been identified and mitigated through targeted enhancements. Theoretical analyses, supported by results from three comparative numerical experiments, confirm the outstanding convergence and robustness of the enhanced stable ADISZNN model. To our knowledge, no prior work has introduced an accelerated dual-integral structure zeroing neural network capable of linear noise cancellation in the context of dynamic complex matrix inversion.

The article is structured into five methodical sections. Section 2 delves into the DCMI problem, presenting the design formulation and procedural details of the ADISZNN model. Section 3 offers a theoretical exposition and validation of ADISZNN’s convergence and robustness, utilizing Lyapunov’s theorem and supported by graphical analyses, with the SBPAF function selected for the model’s activation. In Section 4, we conducted three sets of numerical comparison experiments. The article concludes with a summary of the findings in Section 5.

2. Problem Formulation, Design Formula, and ADISZNN Model

2.1. Consideration of the DCMI Problem

The dynamic complex matrix inverse problem can be described as follows:

$$A(t)X(t) = I,$$

or

$$X(t)A(t) = I \in \mathbb{C}^{n \times n},$$

(1)

where $$A(t) \in \mathbb{C}^{n \times n}$$ is a nonsingular and smooth dynamic complex coefficient matrix with rank $$n$$, and $$X(t)$$ represents the real-time solution of Equation (1), obtained through the ADISZNN model, where $$I \in \mathbb{C}^{n \times n}$$ denotes the identity matrix. Our aim is to compute $$X(t)$$ such that Equation (1) holds true at any given time $$t \in [0, +\infty)$$. Hence, we have

$$X^*(t) = A^{-1}(t).$$

As complex numbers consist of real and imaginary parts, Equation (1) can be rewritten as
where \([A_{re}(t) + jA_{im}(t)][A_{re}(t) + jA_{im}(t)]\) is the expansion of the complex matrix \(A(t)\), and \(A_{re}(t)\) and \(A_{im}(t)\) are, respectively, the real and imaginary parts of the given matrix \(A(t)\). Similarly, \(X_{re}(t)\) and \(X_{im}(t)\) are, respectively, the real and imaginary parts of the state solution \(X(t)\), where the imaginary unit is denoted as \(j = \sqrt{-1}\).

### 2.2. Design Formula

To compute the dynamic complex matrix inversion, a function is devised to measure the real-time error in Equation (1), as follows:

\[
E(t) = A(t)X(t) - I. \tag{3}
\]

Its derivative with respect to time \(t\) is given by

\[
\dot{E}(t) = \dot{A}(t)X(t) + A(t)\dot{X}(t). \tag{4}
\]

The design formula of the integration-enhanced zeroing neural network model is as follows [14]:

\[
\dot{E}(t) = -s_0(E(t)) - s_1 \int_0^t E(\tau)d\tau \tag{5}
\]

where design parameters \(s_0 > 0\) and \(s_1 > 0\) are adjusted for the rate. By combining Equations (4) and (5), we can derive the following formula:

\[
A(t)\dot{X}(t) = -\dot{A}(t)X(t) - s_0(\dot{A}(t)X(t) - I) - s_1 \int_0^t (A(\tau)X(\tau) - I)d\tau. \tag{6}
\]

In actuality, a wide array of noise phenomena are consistently present across numerous practical applications. Examples include the superfluous movements observed in robotic arm operations, as discussed in [32], and the chaotic dynamics within permanent magnet synchronous motor (PMSM) systems, as explored in [33], among others. To more accurately reflect real-world conditions, we introduce noise into Equation (6), thereby obtaining the following equation:

\[
A(t)\dot{X}(t) = -\dot{A}(t)X(t) - s_0(\dot{A}(t)X(t) - I) - s_1 \int_0^t (A(\tau)X(\tau) - I)d\tau + N(t). \tag{7}
\]

### 2.3. Dual-Integral Structure ZNN Model Design

The DISZNN model proposed by Hua et al. [22] has demonstrated significant efficacy in the restriction of noise, particularly linear noise. The model for DISZNN is as follows:

\[
\dot{E}(t) = -s_0^3 \int_0^t \int_0^t E(\tau)d\tau d\delta - 3s_0^2 \int_0^t E(\delta)d\delta - 3s_0 E(t) \tag{8}
\]

in which \(s_0 \in \mathbb{R}^+\) is the design parameter, the single-integral term restricts noise, while the double-integral term not only restricts noise but also accelerates convergence speed.

### 2.4. ADISZNN Model Design

It has been mentioned in many papers [23–31] that adding an activation function to some ZNN-like models can accelerate the convergence of the error function and enhance the model’s ability to restrict noise. Therefore, we modified the ZNN model by adjusting its design formula to

\[
E(t) = -a\Phi(E(t)) \tag{9}
\]

in which, \(\Phi() : \mathbb{C}^{n \times n} \rightarrow \mathbb{C}^{n \times n} \) is an activation function.

To provide a more intuitive description of the model’s evolution, we set

\[
\Theta(t) = E(t) + a\Phi(E(t)) \tag{10}
\]
Letting
\[
\Theta(t) = -\lambda \int_0^t \Theta(\tau) d\tau,
\]
where \( \lambda > 0 \).

We define
\[
Y(t) = \Theta(t) + \lambda \int_0^t \Theta(\tau) d\tau,
\]
and substituting Equation (10) into Equation (12), we can obtain
\[
Y(t) = \hat{E}(t) + a\Phi(E(t)) + \lambda \int_0^t (\hat{E}(t) + a\Phi(E(\tau))) d\tau
\]
\[
= \hat{E}(t) + a\Phi(E(t)) + \lambda E(t) + \lambda a \int_0^t \Phi(E(\tau)) d\tau.
\]

Similarly, we let
\[
Y(t) = -\lambda \int_0^t Y(\tau) d\tau,
\]
combining Equations (13) and (14), we can obtain the following equation:
\[
\hat{E}(t) + a\Phi(E(t)) + \lambda E(t) + \lambda a \int_0^t \Phi(E(\tau)) d\tau
\]
\[
= -\lambda \int_0^t (\hat{E}(\tau) + a\Phi(E(\tau)) + \lambda E) + \lambda a \int_0^t \Phi(E(\delta)) d\delta d\tau
\]
\[
= -\lambda E(t) - \lambda a \int_0^t \Phi(E(\tau)) d\tau
\]
\[
- \lambda^2 \int_0^t E(\tau) d\tau - \lambda^2 \lambda a \int_0^t \int_0^\tau \Phi(E(\delta)) d\delta d\tau.
\]

Thus, we obtain the ADISZNN model,
\[
\hat{E}(t) = -a\Phi(E(t)) - 2\lambda E(t) - 2\lambda a \int_0^t \Phi(E(\tau)) d\tau
\]
\[
- \lambda^2 \int_0^t E(\tau) d\tau - \lambda^2 \lambda a \int_0^t \int_0^\tau \Phi(E(\delta)) d\delta d\tau.
\]

Therefore, the ADISZNN model form with noise can be reformulated as
\[
\hat{E}(t) = -a\Phi(E(t)) - 2\lambda E(t) - 2\lambda a \int_0^t \Phi(E(\tau)) d\tau
\]
\[
- \lambda^2 \int_0^t E(\tau) d\tau - \lambda^2 \lambda a \int_0^t \int_0^\tau \Phi(E(\delta)) d\delta d\tau
\]
\[
+ N(t).
\]

Furthermore, since we already know that \( E(t) = A(t)X(t) - I \) and \( \hat{E}(t) = \hat{A}(t)X(t) + A(t)\hat{X}(t) \), we can further derive the ADISZNN model incorporating noise:
\[
A(t)\hat{X}(t) = -\hat{A}(t)X(t) - a\Phi(A(t)X(t) - I)
\]
\[
- 2\lambda(A(t)X(t) - I)
\]
\[
- 2\lambda a \int_0^t \Phi(A(\tau)X(\tau) - I) d\tau
\]
\[
- \lambda^2 \int_0^t (A(\tau)X(\tau) - I) d\tau
\]
\[
- \lambda^2 a \int_0^t \int_0^\tau \Phi(A(\delta)X(\delta) - I) d\delta d\tau
\]
\[
+ N(t)
\]
3. Theoretical Analyses

In previous research on DISZNN [22], theoretical analysis of convergence and robustness was demonstrated using Laplace transform methods. However, in this paper, we employ a different approach, based on the Lyapunov principle, for proof. In this section, we primarily discuss and demonstrate the convergence and robustness of the ADISZNN model based on the Lyapunov principle, and analyze and apply lemmas to select the activation function. To better represent the Frobenius norm of $E(t)$, we introduce

$$
\|E(t)\|_F = \|A(t)X(t) - I\|_F.
$$

3.1. Convergence

The convergence of the ADISZNN model in the absence of noise is proven in this subsection.

**Theorem 1.** (Convergence) In the absence of noise, using the ADISZNN model (16) to solve the DCMI problem, as $t$ tends to infinity, the Frobenius norm of the error $E(t)$ approaches zero; that is,

$$
\lim_{t \to \infty} \|E(t)\|_F = 0.
$$

The proof of Theorem 1 is as follows.

**Proof of Theorem 1.** We rewrite Equation (18) in the absence of noise interference as

$$
\dot{E}(t) = -\alpha \Phi(E(t)) - 2\lambda E(t) - 2\lambda \int_0^t \Phi(E(\tau)d\tau

- \lambda^2 \int_0^t E(\tau)d\tau - \lambda^2 \int_0^t \int_0^\tau \Phi(E(\delta))d\delta d\tau.
$$

(19)

To provide a clearer proof, let $a_{xy}(t)$, $x_{xy}(t)$, $e_{xy}(t)$, $\theta_{xy}(t)$, and $\upsilon_{xy}(t)$, respectively, represent the $xy$th subelements of $A(t)$, $X(t)$, $E(t)$, $\Theta(t)$, and $Y(t)$.

Firstly, considering that the equation

$$
Y(t) = \dot{E}(t) + \alpha \Phi(E(t)) + \lambda \int_0^t (E(t) + \alpha \Phi(E(t)))
$$

(20)

under the condition of no noise interference for the ADISZNN model can be transformed into the following form:

$$
Y(t) = -\lambda \int_0^t Y(\tau)d\tau,
$$

(21)

the element-wise item of (21) is

$$
\upsilon_{xy}(t) = -\lambda \int_0^t \upsilon_{xy}(\tau)d\tau,
$$

(22)

its derivative is

$$
\dot{\upsilon}_{xy}(t) = -\lambda \upsilon_{xy}(t).
$$

(23)

Assuming a Lyapunov function $\epsilon(t) = \upsilon_{xy}^2(t)$, its derivative form is as follows:

$$
\dot{\epsilon}(t) = 2\dot{\upsilon}_{xy}(t)\upsilon_{xy}(t).
$$

(24)

Substituting (23) into (24) yields

$$
\dot{\epsilon}(t) = -2\lambda \upsilon_{xy}^2(t).
$$

(25)

Since $\epsilon(t)$ is positive definite, its derivative $\dot{\epsilon}(t)$ is negative definite, thus $\epsilon(t)$ is asymptotically stable. Therefore, we obtain the equation

$$
\lim_{t \to \infty} |\epsilon(t)| = \lim_{t \to \infty} |\upsilon_{xy}^2(t)| = \lim_{t \to \infty} |\upsilon_{xy}(t)| = 0.
$$

(26)
So, based on Equations (20) and (26), we have the following: As \( t \to \infty \), 
\[
|\nu_{xy}| = |e_{xy}(t) + \alpha \Phi(e_{xy}(t)) + \lambda \int_0^t (e_{xy}(\tau) + \alpha \Phi(e_{xy}(\tau))) \, d\tau| = 0.
\]
Considering \( \theta_{xy}(t) = e_{xy}(t) + \alpha \Phi(e_{xy}(t)) \) and we have
\[
\theta_{xy}(t) = -\lambda \int_0^t \theta_{xy}(\tau) \, d\tau, \quad t \to \infty,
\]
thus,
\[
\lim_{t \to \infty} \left| \theta_{xy}(t) + \lambda \int_0^t \theta_{xy}(\tau) \, d\tau \right| = 0,
\]
it is easy to obtain that
\[
\lim_{t \to \infty} |\theta_{xy}(t)| = \lim_{t \to \infty} \left| -\lambda \int_0^t \theta_{xy}(\tau) \, d\tau \right| = 0.
\]
Taking the derivative of the above equation, we obtain
\[
\lim_{t \to \infty} |\dot{\theta}_{xy}(t)| = \lim_{t \to \infty} |-\lambda \theta_{xy}(t)| + \Delta, \quad \Delta \to 0,
\]
in which \( \Delta \) is the small error in the derivative of \( \theta(t) \).

Let us assume another Lyapunov function:
\[
\dot{\rho}(t) = 2\dot{\theta}_{xy}(t) \theta_{xy}(t) = -2\lambda \dot{\theta}_{xy}^2(t).
\]
According to the Lyapunov principle, \( \rho(t) \geq 0, \rho(t) \leq 0 \), we can obtain
\[
\lim_{t \to \infty} |\theta_{xy}(t)| = 0.
\]
Because of \( \dot{\theta}_{xy}(t) = e_{xy}(t) + \alpha \Phi(e_{xy}(t)) \),
\[
\lim_{t \to \infty} |e_{xy}(t) + \alpha \Phi(e_{xy}(t))| = 0,
\]
thus, \( \dot{e}_{xy}(t) = -\alpha \Phi(e_{xy}(t)) \).
Clearly, we obtain
\[
\lim_{t \to \infty} |e_{xy}(t)| = 0.
\]
Therefore, its matrix form is as follows:
\[
\lim_{t \to \infty} \| E(t) \|_F = 0.
\]
Thus, Theorem 1 is proven. \( \Box \)

3.2. Robustness

In the presence of linear noise \( N(t) \) in matrix form, the ADISZNN model can still asymptotically approach the theoretical solution. Its effectiveness and convergence in handling DCMI problems will be analyzed and demonstrated.

**Theorem 2.** (Robustness) In the presence of linear noise, using the ADISZNN model (17) to solve the DCMI problem, as \( t \) tends to infinity, the Frobenius norm of the error \( E(t) \) approaches zero; that is,
\[
\lim_{t \to \infty} \| E(t) \|_F = 0.
\]

**Proof of Theorem 2.** The linear noise is expressed as
\[
N(t) = At + B,
\]
where $A$ and $B$ are constant matrices, and their elements can be written as

$$n_{xy}(t) = a_{xy}t + b_{xy}.$$  

According to Theorem 1, and Equations (20) and (21), the ADISZNN model in the presence of linear noise can be transformed into the following form:

$$Y(t) = -\lambda \int_0^t Y(\tau)d\tau + N(t),$$

with elements as in

$$v_{xy}(t) = -\lambda \int_0^t v_{xy}(\tau)d\tau + n_{xy}(t).$$

Differentiating $v_{xy}$ twice, we can obtain

$$\ddot{v}_{xy}(t) = -\lambda \dot{v}_{xy}(t) + \ddot{n}_{xy}(t),$$ (33)

Taking the first and second derivatives of the noise separately, we obtain $\dot{n}_{xy}(t) = a_{xy}$ and $\ddot{n}_{xy}(t) = 0$. Then,

$$\ddot{v}_{xy}(t) = -\lambda \dot{v}_{xy}(t).$$

Assuming the Lyapunov equation to be

$$\dot{\vartheta}(t) = \dot{v}_{xy}^2(t),$$

therefore,

$$\dot{\vartheta}(t) = 2\dot{v}_{xy}(t)\ddot{v}_{xy}(t) = -\lambda \dot{v}_{xy}^2(t),$$

Since $\dot{\vartheta}(t) \geq 0$ is positive definite and its derivative $\dot{\vartheta}(t) \leq 0$ is negative definite, $\vartheta(t)$ is globally asymptotically stable, and we obtain

$$\lim_{t \to \infty} |\dot{\vartheta}(t)| = \lim_{t \to \infty} |\dot{v}_{xy}^2(t)| = \lim_{t \to \infty} |\ddot{v}_{xy}(t)| = 0.$$ (34)

By combining (32) and (34), we obtain the following equation:

$$\lim_{t \to \infty} |\dot{v}_{xy}(t)| = \lim_{t \to \infty} (-\lambda v_{xy}(t) + \dot{n}_{xy}(t)).$$

Substituting $\dot{n}_{xy}(t) = a$ into it, we obtain

$$\lim_{t \to \infty} (-\lambda v_{xy}(t) + a) = 0,$$

thus, concluding that

$$\lim_{t \to \infty} \lambda v_{xy}(t) = |a|,$$

then, we obtain

$$\lim_{t \to \infty} v_{xy}(t) = \left|\frac{a}{\lambda}\right|.$$  

So, we can derive that as $t \to \infty$,

$$|v_{xy}(t)| = |\dot{e}_{xy}(t) + a\Phi(e_{xy}(t))| = \lambda \int_0^t (e_{xy}(\tau) + a\Phi(e_{xy}(\tau)))d\tau = \left|\frac{a}{\lambda}\right|.$$ (35)

Let

$$\theta_{xy}(t) = \dot{e}_{xy}(t) + a\Phi(e_{xy}(t))$$ (36)

then, we have

$$|v_{xy}(t)| = \left|\theta_{xy}(t) + \lambda \int_0^t \theta_{xy}(\tau)d\tau\right| = \left|\frac{a}{\lambda}\right|. $$
therefore, we can deduce that
\[
\lim_{t \to \infty} \left| \theta_{xy}(t) + \lambda \int_0^t \theta_{xy}(\tau) d\tau \right| = 0,
\]
then, we draw
\[
\lim_{t \to \infty} |\dot{\theta}_{xy}(t)| = \lim_{t \to \infty} \left| -\lambda \theta_{xy}(t) \right|.
\]
Clearly, \( \lambda > 0 \), and \( \dot{\theta} \) and \( \theta \) have opposite signs, thus
\[
\lim_{t \to \infty} |\dot{\theta}_{xy}(t)| = 0.
\]
Furthermore,
\[
\theta_{xy}(t) = \dot{e}_{xy}(t) + a \Phi(e_{xy}(t)),
\]
which means that we can obtain
\[
\lim_{t \to \infty} |\theta_{xy}(t)| = \lim_{t \to \infty} |\dot{e}_{xy}(t) + a \Phi(e_{xy}(t))| = 0,
\]
so, \( \dot{e}_{xy}(t) = -a \Phi(e_{xy}(t)) \).
According to the Lyapunov theorem, we can obtain
\[
\lim_{t \to \infty} |\dot{e}_{xy}(t)| = 0.
\]
The corresponding matrix form is as follows:
\[
\lim_{t \to \infty} \|E(t)\|_F = 0. \tag{37}
\]
Thus, Theorem 2 is proven. \( \square \)

3.3. Selection of Activation Function

For the ADISZNN model, different activation functions will result in different degrees of convergence in the model solution. To maintain generality in our discussion, we consider the three most common types of activation functions: linear-like activation functions [26], sigmoid-like activation functions [27], and sign-like activation functions [34–36]. Here, we take the linear activation function (LAF), smooth bi-polar sigmoid activation function (SBPSAF), and signal bi-power activation function (SBPAF) as examples. They are defined as follows:

- **LAF:**
  \[
  \Phi_1(x) = x \tag{38}
  \]

- **SBPSAF:**
  \[
  \Phi_2(x) = \frac{1}{2} x^{r_1} + \frac{1+e^{-r_2 x}}{1-e^{-r_2 x}} \left( 1 - \frac{e^{-r_2 x}}{1+e^{-r_2 x}} \right), \tag{39}
  \]
  where \( r_1 > 0, r_2 > 0 \).

- **SBPAF:**
  \[
  \Phi_3(x) = (k_1|x|^\eta + k_2|x|^w)\text{sign}(x) + k_3 x, \tag{40}
  \]
  where \( \text{sign}(\cdot) \) is a symbolic function and the design parameters are \( k_1 > 0, k_2 > 0, k_3 > 0, 0 < \eta < 1, \) and \( w > 0 \).

However, determining whether an activation function is suitable is a challenging task. Ref. [35] elucidates a concept within the Lyapunov stability framework, suggesting that the rate of convergence of a system is positively correlated with the magnitude of its derivative near the origin. Specifically, the larger the derivative, the faster the system converges. To illustrate this, Figure 1 in the paper depicts the derivative curves for three activation functions: \( \Phi_1(x) \) LAF, \( \Phi_2(x) \) SBPSAF, and \( \Phi_3(x) \) SBPAF. It is observed that near the origin,
the derivative of SBPAF exceeds that of LAF, and similarly, the derivative of LAF surpasses that of SBPSAF. Based on this observation, it can be inferred that the ADISZNN model employing the SBPAF activation function may converge in a shorter time compared to the model using the LAF activation function. Likewise, the model with the LAF activation function is likely to converge faster than the one with the SBPSAF activation function.

Figure 1. Three sorts of activation functions presented: linear activation function (LAF) $\Phi_1(x)$ (red solid line), smooth bi-polar sigmoid activation function (SBPSAF) $\Phi_2(x)$ (sky blue solid line), and signal bi-power activation function (SBPAF) $\Phi_3(x)$ (blue solid line).

Therefore, in this article, we will use SBPAF as the activation function adopted by ADISZNN.

4. Simulation and Comparative Numerical Experiments
4.1. Comparison Experiments of Activation Functions

In this section, to further validate the correctness of our activation function selection, we compare the ADISZNN models using three different activation functions.

In this example, a two-dimensional dynamic complex matrix $A$ is presented as follows:

$$A(t) = \begin{bmatrix} j \sin(3t) & j \cos(3t) \\ -j \cos(3t) & j \sin(3t) \end{bmatrix} \in \mathbb{C}^{2 \times 2}. \quad (41)$$

For convenience, this matrix only contains the imaginary part. To verify the correctness of the ADISZNN model, the theoretical inverse of the above dynamic complex matrix is obtained through mathematical calculation:

$$A^{-1}(t) = \begin{bmatrix} -j \sin(3t) & j \cos(3t) \\ -j \cos(3t) & -j \sin(3t) \end{bmatrix} \in \mathbb{C}^{2 \times 2}. \quad (42)$$

Figure 2 delineates the computational and convergence trajectories of the ADISZNN model across various activation functions, all in the absence of noise. A discernible observation from this figure is that models utilizing the LAF and SBPSAF activation functions achieve near-simultaneous steady-state error close to zero at approximately 2.3 s. In stark contrast, the ADISZNN model equipped with the SBPAF activation function demonstrates a markedly swifter convergence, reaching near-zero error within a mere 0.6 s—a rate that is roughly threefold faster than its counterparts.
**Figure 2.** Comparative graph of the computation and convergence processes of ADISZNN with three different activation functions without noise interference; the design parameters are $\lambda = 4$ and $\alpha = 4$: ADISZNN-LAF (green dashed line), ADISZNN-SBPSAF (blue dashed line), ADISZNN-SBPAF (red solid line).

Figure 3, on the other hand, captures the ADISZNN model’s performance under the influence of linear noise, with each subplot showcasing the model’s behavior when driven by the LAF, SBPSAF, and SBPAF activation functions, respectively. For the readers’ ease, a comparative analysis of these models is tabulated in Table 1. The table underscores a significant finding: the proximity of the activation function’s derivative to the origin is positively correlated with the model’s convergence efficiency. Notably, the ADISZNN model harnessing SBPAF exhibits the most rapid convergence. Nonetheless, it is important to note that models incorporating SBPSAF and SBPAF show a relatively diminished robustness when compared to the LAF-equipped model.

**Table 1.** Comparison of ADISZNN model adopting LAF, SBPSAF, and SBPAF.

<table>
<thead>
<tr>
<th>AF</th>
<th>LAF</th>
<th>SBPSAF</th>
<th>SBPAF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Derivative near 0 point</td>
<td>Large</td>
<td>Normal</td>
<td>Larger</td>
</tr>
<tr>
<td>Convergence</td>
<td>Fast</td>
<td>Normal</td>
<td>Faster</td>
</tr>
<tr>
<td>Robustness</td>
<td>Strong</td>
<td>Weak</td>
<td>Normal</td>
</tr>
</tbody>
</table>

These experiments not only confirm the enhanced convergence speed of the ADISZNN model employing the SBPAF activation function proposed in this paper but also validate the appropriateness of the chosen activation function.

Next, we will compare and analyze the convergence performance of the ADISZNN model using the signal bi-power activation function with the DISZNN model without using any activation function under the condition of linear matrix noise interference.
4.2. Comparison Experiments Between DISZNN And ADISZNN

The DISZNN model is rewritten as follows:

\[
\dot{E}(t) = -s_0^3 \int_0^t \int_0^\delta E(\tau)d\tau d\delta - 3s_0^2 \int_0^t E(\delta)d\delta - 3s_0 E(t)
\]  \hspace{1cm} (43)

where \(s_0 \in \mathbb{R}^+\) is a design parameter.

The error results of the DISZNN model and the ADISZNN model using SBPAF are shown in Figure 4. Under the condition without noise interference, for any value of the dynamic complex matrix \(X(0) \in [-(2 + 2j), 2 + 2j]^{2 \times 2}\) the error of the DISZNN model converges almost completely to 0 around 2.8 s. When the SBPAF activation function is introduced, the error of the ADISZNN model converges almost completely to 0 within 0.6 s. Therefore, the convergence speed of the ADISZNN model is significantly faster than that of DISZNN.

To compare the tolerance of ADISZNN and DISZNN to noise, a common linear noise \(N = [0.8 + 4t]^{2 \times 2}\) is introduced. Their numerical experimental comparison is shown in Figure 5, where the design parameters are set as \(s_0 = 4, \lambda = 4, \) and \(\alpha = 4\). Under the interference of linear noise, both DISZNN and ADISZNN can still make the residual \(||E(t)||_F\) close to 0 within approximately 2.8 s and 0.6 s, respectively, which is nearly the same as the case without noise interference. This demonstrates that ADISZNN and DISZNN possess inherent tolerance to linear noise.

However, during the experimental process, we observed that the residual plot of the ADISZNN model with the signal bi-power activation function exhibits oscillatory fluctuations after reaching the magnitude of \(10^{-3}\) at 0.6 s. This indicates a decrease in the precision of the model’s computations, as it fails to maintain stable convergence at the \(10^{-3}\) magnitude level. This implies a reduction in the robustness of the ADISZNN model. The residual plots of the ADISZNN model with noise interference and the comparison of residuals between the DISZNN and ADISZNN models without noise interference are
shown in Figures 6 and 7, respectively. In the next subsection, we will discuss the stable version (high-precision version) of the ADISZNN model.

Figure 4. Convergence comparison of ADISZNN and DISZNN without noise interference; the design parameters are $s_0 = 4\lambda = 4$ and $\alpha = 4$: ADISZNN-SBPAF (red solid line) and DISZNN (blue solid line).

Figure 5. Convergence comparison of ADISZNN and DISZNN, under linear noise $N = (0.8 + 4t)^2 + 2$: ADISZNN-SBPAF (pink dotted line) and DISZNN (blue dashed line).
Figure 6. The amplified residual errors of ADISZNN under linear noise $N = [(0.8 + 4t)^2]^2$, with design parameters $\alpha = 4$, $\lambda = 4$, $k_1 = 1$, $k_2 = 1$, $k_3 = 1$, $\eta = 1/3$, and $\omega = 3$.

Figure 7. The amplified residual errors of DISZNN and ADISZNN under no noise, with design parameters $\alpha = 4$, $\lambda = 4$, $k_1 = 1$, $k_2 = 1$, $k_3 = 1$, $\eta = 1/3$, $\omega = 3$, and $s_0 = 4$.

4.3. The Stable ADISZNN Model

In this subsection, we propose an improved version of the ADISZNN-SBPAF model to address the oscillation phenomenon (or precision degradation phenomenon). According to the table in experiment 1, we observe that not only can ADISZNN with the signal bi-power activation function accelerate the convergence compared to the DISZNN model, but also the ADISZNN model with the linear activation function (LAF) can similarly accelerate convergence and exhibit stronger robustness.

Based on this observation, we innovatively propose a stable version of the ADISZNN model: When the error of the ADISZNN model using SBPAF approaches zero (i.e., reaches the order of $10^{-3}$), we transition the model to use LAF. This transition alters the calculation and convergence of $||A(t)X(t) - I||_F$, transforming the ADISZNN-SBPAF model into the ADISZNN-LAF model. The convergence performance of this approach is illustrated in the Figures 8–10.

The residual plots in Figures 9 and 10, respectively, depict the effects of our improvement on the stable version of the ADISZNN model compared to the unstable version (For the reader’s enhanced comprehension, Figure 8 illustrates the comparison of amplified residual errors between the stable and unstable variants of ADISZNN in the absence of noise). While this enhancement results in a slight increase in the convergence time, it strengthens the model’s resistance to noise and improves its robustness. Additionally, the computational accuracy is elevated from the order of $10^{-3}$ to $10^{-4}$, thereby enhancing the convergence performance of the model.
Figure 8. Comparison of the amplified residual errors of stable ADISZNN and unstable ADISZNN under no noise, with design parameters $\alpha = 4$, $\lambda = 4$, $k_1 = 1$, $k_2 = 1$, $k_3 = 1$, $\eta = 1/3$, $\omega = 3$, and $s_0 = 4$.

Figure 9. The detailed residual errors of ADISZNN-Stable and ADISZNN-SBPAF, with noise of $[0.8 + 4t]^2 \times 2$.

Figure 10. Residual errors of ADISZNN-Stable and ADISZNN-SBPAF with noise of $[0.8 + 4t]^2 \times 2$.

In Figure 11, a comparison between the stable version of the ADISZNN model and the original DISZNN model is presented. Compared to the original DISZNN model, the stable version of the ADISZNN model exhibits significant improvements. When the computed solution $X(t)$ of the model converges to a theoretical state approximation $A(t)^{-1}$,
the convergence time of the stable ADISZNN model is reduced from 2.8 s to 1.9 s. Moreover, the convergence curve of the stable ADISZNN model appears smoother and more refined. Both models achieve a computational accuracy of $10^{-4}$ when fully converged. These results indicate that the improved stable version of the ADISZNN model not only enhances the convergence speed but also maintains robustness comparable to that of the DISZNN model.

To underscore the merits of the ADISZNN-Stable model, Figures 12 and 13 depict comparative trajectory plots of the DISZNN alongside the ADISZNN-Stable model under conditions of linear noise. Additionally, Figure 14 presents an analysis of residual errors, contrasting the performance of the DISZNN with that of the ADISZNN-Stable model in an environment devoid of noise.

Figure 11. The comparison of the residual errors of stable ADISZNN and DISZNN under linear noise of $[0.8 + 4t]^{2 \times 2}$, with design parameters $\alpha = 4$, $\lambda = 4$, $k_1 = 1$, $k_2 = 1$, $k_3 = 1$, $\eta = 1/3$, $\omega = 3$, and $s_0 = 4$.

Figure 12. Trajectory analysis for problem (41) under linear noise of $[0.8 + 4t]^{2 \times 2}$; the red line points represent the theoretical solution, while the blue line points show the DISZNN model’s solutions.
Figure 13. Trajectory analysis for problem (41) under linear noise of $[0.8 + 4t]^2 \times 2$; the red line points represent the theoretical solution, while the blue line points show the ADISZNN-Stable model’s solutions.

Figure 14. Comparison of the amplified residual errors of stable ADISZNN and DISZNN under no noise, with design parameters $\alpha = 4$, $\lambda = 4$, $k_1 = 1$, $k_2 = 1$, $k_3 = 1$, $\eta = 1/3$, $\omega = 3$, and $s_0 = 4$.

5. Conclusions

This article introduces a novel enhancement to the DISZNN model through the integration of an activation function, culminating in an accelerated dual-integral structure ZNN model. This model exhibits enhanced resilience against linear noise interference, particularly pertinent for dynamic complex matrix inversion challenges. The paper unfolds with the following key contributions: initially, the design formula for a single-integral structure and the DISZNN model are presented and analyzed; subsequently, the architecture of the ADISZNN model is designed, with a theoretical examination of its convergence and robustness; thirdly, both experimental and theoretical analyses are employed to assess the influence of various activation functions on the ADISZNN’s convergence efficacy, thereby substantiating the efficacy of our selected activation function; fourthly, comparative tests under linear noise conditions between the ADISZNN and DISZNN models underscore the ADISZNN’s superior convergence capabilities, albeit with the caveat that the ADISZNN model utilizing the SBPAF activation function exhibits oscillatory behavior, potentially compromising its robustness. In light of these findings, we propose refinements to the ADISZNN-SBPAF model, yielding a more stable iteration of the ADISZNN. Comparative experimentation facilitates the identification of the optimal ZNN configuration. Future inquiries are suggested to investigate the potential applications of the ADISZNN model within the engineering sector. This paper presents the ADISZNN model, which has certain limitations, specifically detailed in Appendix A.
**Author Contributions:** Conceptualization, Y.H. and F.Y.; methodology, T.W. and Y.H.; software, F.Y. and Y.H.; validation, F.Y. and Y.H.; formal analysis, F.Y.; investigation, T.W.; resources, Y.H.; data curation, F.Y.; writing—original draft preparation, F.Y.; writing—review and editing, Y.H.; visualization, F.Y.; supervision, Y.H.; project administration, F.Y.; funding acquisition, Y.H. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was funded by the National Natural Science Foundation of China under Grant No. 62062036, Grant No. 62066015 and Grant No. 62006095, and the College Students’ Innovation Training Center Project at Jishou University under Grant No. JDCX20231012.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Data are contained within the article.

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

**Abbreviations**

The following abbreviations are used in this article:

- ZNN: Zeroing neural network
- GNN: Gradient neural network
- DNSZNN: Dual noise-suppressed ZNN
- DISZNN: Dual-integral structure zeroing neural network
- CVZNN: Complex-valued ZNN
- IEZNN: Integral-enhanced ZNN
- CVNNTZNN: Complex-valued noise-tolerant ZNN
- ADISZNN: Accelerated dual-integral structure zeroing neural network
- DRMI: Dynamic real matrix inversion
- DCMI: Dynamic complex matrix inversion
- AF: Activation function
- LAF: Linear activation function
- SBPAF: Signal bi-power activation function
- SBPSAF: Smooth bi-polar sigmoid activation function

**Appendix A. Limitation**

1. The ADISZNN model and the ADISZNN-Stable model proposed in this paper currently do not handle discontinuous noise.

2. This paper restricts the inversion of matrices to be non-singular, smooth, dynamic, and complex. The problem of inverting singular or non-smooth matrices is not addressed in this paper.

**References**


**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.