Zhang Neural Network without Using Time-Derivative Information for Constant and Time-Varying Matrix Inversion

Yunong Zhang, Member, IEEE, Zenghai Chen, Ke Chen, and Binghuang Cai

Abstract—To obtain the inverses of time-varying matrices in real time, a special kind of recurrent neural networks has recently been proposed by Zhang et al. It is proved that such a Zhang neural network (ZNN) could globally exponentially converge to the exact inverse of a given time-varying matrix. To find out the effect of time-derivative term on global convergence as well as for easier hardware-implementation purposes, the ZNN model without exploiting time-derivative information is investigated in this paper for inverting online matrices. Theoretical results of both constant matrix inversion case and time-varying matrix inversion case are presented for comparative and illustrative purposes. In order to substantiate the presented theoretical results, computer-simulation results are shown, which demonstrate the importance of time derivative of given matrices on the exact convergence of ZNN model to time-varying matrix inverses.

I. INTRODUCTION

The problem of obtaining online the inverse of a matrix arises in numerous fields of science, engineering, and business. It is usually a fundamental part of many solutions, e.g., as essential steps for signal-processing [1][2] and robot control [3][4].

The circuit-realizable dynamic-system approach is one of the important parallel-computational methods for solving matrix-inverse problems [1][3][4]. Recently, due to the in-depth research in neural networks, numerous dynamic and analog solvers in the form of recurrent neural networks have been developed and investigated [1][4][6][8][14]-[16]. The neural-dynamic approach is now regarded as a powerful alternative to online computation because of its parallel distributed nature and convenience of hardware implementation [13][14].

A special kind of recurrent neural networks with implicit dynamics has recently been proposed by Zhang et al for time-varying equations solving; e.g., [4][9][10][12]. To solve for the inverse of time-varying matrix \( A(t) \in \mathbb{R}^{n \times n} \), the following ZNN model could be established:

\[
A(t) \dot{X}(t) = -\dot{A}(t)X(t) - \gamma F(A(t)X(t) - I),
\]

where, starting from an initial condition \( X(0) \in \mathbb{R}^{n \times n} \), \( X(t) \) is the activation state matrix corresponding to time-varying inverse \( A^{-1}(t) \). On the other hand, to invert constant matrix \( A \in \mathbb{R}^{n \times n} \), without derivative-circuit \( \dot{A}(t)X(t) \) (in short, D-circuit) due to time-derivative \( \dot{A}(t) \equiv 0 \) in this case, the above ZNN model (1) reduces to

\[
AX(t) = -\gamma F(A(t)X(t) - I),
\]

In (1) and (2), \( I \in \mathbb{R}^{n \times n} \) denotes the identity matrix, design parameter \( \gamma > 0 \) is used to scale the convergence rate of the neural solution, and \( F(\cdot) : \mathbb{R}^{n \times n} \rightarrow \mathbb{R}^{n \times n} \) denotes a matrix-valued activation-function array of neural networks. Processing array \( F(\cdot) \) is made of \( n^2 \) monotonically increasing odd activation-functions \( f(\cdot) \). For example, the first three basic types of the following activation functions are depicted in Fig. 1:

- linear activation function \( f(u) = u \),
- bipolar sigmoid activation function (with \( \xi \geq 2 \)) \( f(u) = (1 - \exp(-\xi u))/(1 + \exp(-\xi u)) \),
- power activation function (with odd integer \( p \geq 3 \)) \( f(u) = u^p \),
- and, power-sigmoid activation function

\[
f(u) = \begin{cases} u^p, & \text{if } |u| > 1 \\ \delta \frac{1 - \exp(-\xi u)}{1 + \exp(-\xi u)}, & \text{otherwise} \end{cases}
\]

with \( \xi \geq 2 \), \( p \geq 3 \), and \( \delta = (1 + \exp(-\xi))/(1 - \exp(-\xi)) > 1 \).

It has been shown that ZNN (1) could globally exponentially converge to the exact inverse of a given time-varying matrix [4][9][10][12]. In order to know the effect of time-derivative information on neural matrix inversion (1) as well as for lower-complexity hardware-implementation, such ZNN models without using time-derivative information are investigated in this paper for both constant-matrix and time-varying-matrix inversion. For comparison between the above presented ZNN model and the conventional gradient-based neural network (GNN) for matrix inversion, please see Appendix A. The remainder of this paper is thus organized in three sections. In Section II, we analyze and simulate the ZNN model (2) for constant matrix inversion. In Section III, we analyze and simulate the ZNN model (1) without D-circuit for time-varying matrix inversion. Section IV concludes this paper with final remarks.

Before ending this introductory section, it is worth mentioning the main contributions of the article as follows.

1) We investigate that ZNN model (2) could globally exponentially converge to the exact inverse of a given constant nonsingular matrix.
2) We show that the time derivative information of given matrices (or say, the D-circuit) plays an important role in ZNN model (1) which inverts time-varying matrices in real time.

3) We substantiate that the ZNN model without D-circuit can work approximately well for time-varying matrix inversion, if we pursue simpler neural-circuit implementation and allow less accurate solution.

II. CONSTANT MATRIX INVERSION

In this section, the ZNN model without time-derivative information [i.e., neural-dynamics (2)] is employed for inverting online constant matrices. Theoretical analysis is presented in detail and verified by computer simulations.

A. Theoretical Results

The following theoretical results are established about the global exponential convergence of ZNN (2) which inverts constant nonsingular matrix $A \in R^{n \times n}$ online.

**Theorem 1:** Consider constant nonsingular matrix $A \in R^{n \times n}$. If a monotonically-increasing odd activation function array $F(\cdot)$ is used, then the state matrix $X(t)$ of ZNN (2) starting from any initial state $X(0) \in R^{n \times n}$ always converges to constant theoretical-inverse $X^* := A^{-1}$ of matrix A. Moreover, for constant A, the ZNN model (2) possesses

1) global exponential convergence with rate $\gamma$ if using linear activation-function array $F(X) = X$;
2) global exponential convergence with rate $\xi \gamma /2$ if using bipolar sigmoid activation-function array;
3) superior convergence to situation 1) for error range $||AX(t) - I||_{ij} > 1$ with $i, j \in \{1, 2, \ldots, n\}$, if using power-activation-function array; and
4) globally superior convergence to situation 1) if using power-sigmoid activation-function array.

**Proof:** Omitted due to space limitation.

B. Simulative Verification

For illustration, let us consider the following constant matrix $A \in R^{3 \times 3}$ with its theoretical inverse given below for comparison:

$$A = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}, \quad X^* := A^{-1} = \begin{bmatrix} 1 & 1 & -1 \\ -1 & 0 & 1 \\ 0 & -1 & 1 \end{bmatrix}.$$

The ZNN model (2) solving for $A^{-1}$ could thus be depicted in the following specific form:

$$\begin{bmatrix} \dot{x}_{11} \\ \dot{x}_{12} \\ \dot{x}_{13} \\ \dot{x}_{21} \\ \dot{x}_{22} \\ \dot{x}_{23} \\ \dot{x}_{31} \\ \dot{x}_{32} \\ \dot{x}_{33} \end{bmatrix} = -\gamma F \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_{11} \\ x_{12} \\ x_{13} \\ x_{21} \\ x_{22} \\ x_{23} \\ x_{31} \\ x_{32} \\ x_{33} \end{bmatrix} - \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

where processing array $F(\cdot)$ could typically be constructed by using $n^2$ power-sigmoid activation functions in the form of (3) with $\xi = 4$ and $p = 3$.

As seen from Fig. 2, starting from any initial states randomly selected in $[-2, 2]^{3 \times 3}$, state matrices of ZNN model (2) all converge to constant theoretical-inverse $A^{-1}$. Evidently, the convergence time could be decreased considerably by increasing design parameter $\gamma$. It follows from this figure and other simulation data that design parameter $\gamma$ plays an important role on the convergence speed of ZNN models. In addition, for the situation of using power-sigmoid activation-function array, Fig. 3 shows that superior convergence could be achieved, as compared to other activation-function situations. Note that $||A||_F := \sqrt{\text{trace}(A^T A)}$ denotes hereafter the Frobenius norm of matrix A. Computer-simulation has now substantiated the theoretical analysis presented in Subsection II-A.

III. TIME-VARYING MATRIX INVERSION

While Section II investigates the performance of ZNN model (2) inverting constant matrices, the time-varying matrices are inverted by the ZNN model in this section.

For hardware implementation with lower complexity, however, instead of using ZNN (1), we could have the following simplified neural-dynamic model by removing the D-circuit [i.e., the term of time derivative $\dot{A}(t)$] from (1):

$$A(t) \dot{X}(t) = -\gamma F(A(t) X(t) - I),$$

(4)

which could solve approximately for time-varying $A^{-1}(t)$. Note that ZNN model (4) can be viewed as a time-varying version of ZNN model (2) by replacing $A$ therein with $A(t)$.

A. Preliminaries

To lay a basis for further detailed analysis of ZNN model (4), the following invertibility-condition and $||A^{-1}(t)||_F$-lemma are presented [4]. The former guarantees the uniform existence of time-varying matrix inverse $A^{-1}(t)$, whereas the latter gives the uniform upper-bound of $||A^{-1}(t)||_F$. They will be used in the theoretical analysis of ZNN model (4) in the ensuing subsection.
Condition: There exists a real number \( \alpha > 0 \) such that

\[
\min_{i \in \{1, 2, \ldots, n\}} |\lambda_i(A(t))| \geq \alpha, \quad \forall t \geq 0 \tag{5}
\]

where \( \lambda_i(\cdot) \) denotes the \( i \)th eigenvalue of \( A(t) \in R^{n \times n} \).

Lemma: If \( A(t) \) satisfies the invertibility condition (5) with its norm uniformly upper bounded by \( \beta \) (i.e., \( \|A(t)\|_F \leq \beta, \quad \forall t \geq 0 \)), then \( \|A^{-1}(t)\|_F \) is uniformly upper bounded, i.e.,

\[
\|A^{-1}(t)\|_F \leq \varphi := \sum_{i=0}^{n-2} C_n^i \beta^{n-i-1}/\alpha^{n-i} + n^{3/2}/\alpha \tag{6}
\]

for any \( t \geq 0 \), where \( C_n^i := n!/(i!(n-i)!)) \) [4].

B. Theoretical Results

The following theoretical results are established about the solution-error bound of simplified ZNN model (4) inverting time-varying nonsingular matrix \( A(t) \) online.

Theorem 2: Consider time-varying nonsingular matrix \( A(t) \in R^{n \times n} \) which satisfies invertibility condition (5) and norm condition (6). If a monotonically-increasing odd activation function array \( F(\cdot) \) is used, then the computational error \( \|X(t) - A^{-1}(t)\|_F \) of ZNN (4) starting from any initial state \( X(0) \in R^{n \times n} \) is always upper bounded, with its steady-state solution-error no greater than \( nε\varphi^2/(\gamma\rho - ε\varphi) \), provided that \( \|A(t)\|_F \leq ε \) for any \( t \in [0, \infty) \) and design-parameter \( \gamma \) is large enough (\( \gamma > ε\varphi/\rho \)), where coefficient

\[
\rho := \min_{i,j \in \{1, \ldots, n\}} \frac{\max_{i \in \{1, \ldots, n\}} \{f(|x_{ij}(0)|)/|e_{ij}(0)|\}, f'(0)}{ε} \tag{7}
\]

with \( e_{ij}(0) := |A(0)X(0) - I|_{ij}, \quad i, j \in \{1, 2, \ldots, n\} \).

Proof: We can reformulate ZNN (4) as the following [with \( \Delta_B(t) := -A(t) \) and \( \Delta_C(t) := 0 \in R^{n \times n} \)]:

\[
A(t)X(t) = -A(t) + \Delta_B(t)X(t) - \gamma F(A(t)X(t) - I) + \Delta_C \tag{8}
\]

which becomes exactly equation (10) of [4]. In view of \( \|\Delta_B\|_F = \|A\|_F \leq \varepsilon \) and \( \|\Delta_C\|_F = 0 \) for any \( t \in [0, \infty) \), we could now reuse the theoretical results of Theorem 2 in [4]. That is, the computational error \( \|X(t) - A^{-1}(t)\|_F \) of neural-dynamics (8) [equivalently, ZNN (4)] is always upper bounded. In addition, it follows immediately from Theorem 2 and equation (14) of [4] (see Appendix B) that its steady-state computational error

\[
\lim_{t \to \infty} \|X(t) - A^{-1}(t)\|_F \leq nε\varphi^2/(\gamma\rho - ε\varphi) \].

Furthermore, design-parameter \( \gamma \) is required therein to be greater than \( ε\varphi/\rho \). In the original proof of Theorem 2 of [4], coefficient \( \rho > 0 \) is defined between \( f(e_{ij}(0))/e_{ij}(0) \) and \( f'(0) \). Following that proof and considering the worst case or such an error bound, we could determine the value of \( \rho \) as in (7). Specifically speaking, if the linear activation-function array \( F(X) = X \) is used, then \( \rho = 1 \); if the bipolar sigmoid activation-function array is used, then \( \rho = \max_{i,j \in \{1, 2, \ldots, n\}} \{f(|e_{ij}(0)|)/|e_{ij}(0)|\} \); and, if the powersigmoid activation-function array (3) is used, then \( \rho \geq 1 \) (where the sign of inequality “>” is taken in most situation). The proof is thus complete. \( \square \)

Fig. 2. Inversion of constant matrix \( A \) by ZNN (2) using power-sigmoid activation-function array with \( \xi = 4 \) and \( p = 3 \)

Fig. 3. Convergence comparison of solution error \( \|X(t) - A^{-1}\|_F \) by ZNN (2) using different activation-function arrays with \( \gamma = 1 \)
ZNN model (4) is thus in this specific form
\[
A(t) \equiv \begin{bmatrix}
\sin t & \cos t \\
-\cos t & \sin t
\end{bmatrix},
A^{-1}(t) = \begin{bmatrix}
\sin t & -\cos t \\
\cos t & \sin t
\end{bmatrix}.
\]
ZNN model (4) is thus in this specific form
\[
\begin{bmatrix}
\sin t & \cos t \\
-\cos t & \sin t
\end{bmatrix}
\begin{bmatrix}
\dot{x}_{11} \\
\dot{x}_{12}
\end{bmatrix}
= -\gamma \mathcal{F}
\begin{bmatrix}
\sin t & \cos t \\
-\cos t & \sin t
\end{bmatrix}
\begin{bmatrix}
x_{11} \\
x_{12}
\end{bmatrix}
- \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix}.
\]
where processing array \( \mathcal{F}(\cdot) \) could typically be constructed by using \( n^2 \) power-sigmoid activation functions in the form of (3) with \( \xi = 4 \) and \( p = 3 \).

Figs. 4 and 5 could thus be generated to show the performance of ZNN (4). According to Figs. 4 and 5, starting from initial states randomly selected in \([-2, 2]^{2 \times 2} \), state matrices of the presented ZNN model (4) could not converge to theoretical inverse exactly. Instead, it could only approach to an approximate solution of \( A^{-1}(t) \). In addition, as shown in Fig. 5, when we increase design-parameter \( \gamma \) from 1 to 10, the steady-state computational error \( \lim_{t \to +\infty} \|X(t) - A^{-1}(t)\|_F \) decreases rapidly. However, there always exists a steady-state solution-error which could not vanish to zero. These computer-simulation results have substantiated the theoretical results presented in Subsection III-B.

For comparison between the simplified ZNN model (4) and the original ZNN model (1) [which has the time-derivative term \( A(t)X(t) \)], we could generate Fig. 6 by applying ZNN (1) to this time-varying inversion. It shows the performance of the original ZNN model (1) for the time-varying matrix inversion under the same design parameters. Seeing and comparing Figs. 4 and 6, we know that the time derivative information \( \dot{A}(t) \) plays an important role on the convergence of ZNN models for time-varying matrix inversion.

C. Simulative Verification

For illustration and comparison, let us consider the time-varying coefficient matrix with its theoretical inverse as the following:
\[
A(t) = \begin{bmatrix}
\sin t & \cos t \\
-\cos t & \sin t
\end{bmatrix},
A^{-1}(t) = \begin{bmatrix}
\sin t & -\cos t \\
\cos t & \sin t
\end{bmatrix}.
\]

IV. Conclusions

An efficient recurrent neural network for online time-varying matrix inversion has been proposed by Zhang et al. [4][9][10][12]. The performance analysis of such a ZNN model without time-derivative term is presented in this paper. For comparative purposes, both constant matrix inversion and time-varying matrix inversion are analyzed. On one hand, as time-derivative \( \dot{A} \equiv 0 \), the global exponential convergence of the ZNN model (2) for such a given constant-matrix inversion could be achieved. On the other hand, without exploiting the \( \dot{A}(t) \) information, the simplified ZNN model only approaches to the approximate inverse (instead of the exact one). Simulation results have demonstrated the importance of the time derivative information \( \dot{A}(t) \), limiting the performance of the simplified ZNN model for online time-varying matrix inversion.
APPENDIX A

The design of the gradient-based neural network (GNN) and the ZNN models (1)-(2)-(4) could be viewed from the online solution of the following defining equation:

\[ AX(t) - I = 0, \quad t \in [0, +\infty), \]  

(9)

where coefficient \( A \) is an \( n \)-dimensional square-matrix (being constant for GNN design while able to be time-varying for ZNN design), and \( X(t) \in R^{n \times n} \) is to be solved.

GNN Design

The GNN model for online constant-matrix inversion could be developed by the following procedure. Note that the gradient-descent design method could only be employed here for constant-matrix inversion problem.

• Firstly, to solve (9) for \( X(t) \) via a neural-dynamic approach, we can define a scalar-valued norm-based error function, \( E(t) = \|AX(t) - I\|_F^2/2 \). It is worth pointing out that a minimum point of residual-error function \( E(t) = \|AX(t) - I\|_F^2/2 \) could be achieved with \( E(t) = 0 \), if and only if \( X(t) \) is the exact solution of equation (9) (in other words, \( X(t) = X^* := A^{-1} \)).

• Secondly, a computational scheme could be designed to evolve along a descent direction of this error function \( E(t) \), until the minimum point \( X^* \) is reached. Note that a typical descent direction is the negative gradient of \( E(t) \), i.e., \(-\nabla E/\nabla X \in R^{n \times n} \).

• Thirdly, in view of \( \nabla E/\nabla X = AT(AX(t) - I) \in R^{n \times n} \), it follows from the gradient-descent design formula:

\[ \dot{X}(t) = -\gamma \frac{\nabla E}{\nabla X} = -\gamma A^T(AX(t) - I), \quad t \in [0, +\infty), \]

where design parameter \( \gamma > 0 \) is defined same as in the aforementioned ZNN models.

ZNN Design

The original ZNN model (1) for online constant and/or time-varying matrix inversion could be developed by the following procedure by Zhang et al [4][9][10][12]. Afterwards, it could be simplified to be (1) and (2). Note that this ZNN design method could be employed for both constant and time-varying problems solving.

• Firstly, we could construct a matrix-valued error function \( E(X(t), t) = AX(t) - I \). Note that \( E(X(t), t) \) equals zero if and only if \( X(t) \) is the solution of (9).

• Secondly, the error-function time-derivative \( \dot{E}(X(t), t) \) is chosen to guarantee that every entry \( e_{ij}(t), \ i,j = 1,2, \cdots, n \), of \( E(X(t), t) \) converges to zero. Its general form could be given as the following:

\[ \frac{dE(X(t), t)}{dt} = -\gamma F(E(X(t), t)), \]

(10)

where design parameter \( \gamma > 0 \) and activation function array \( F(\cdot) \) have been defined as in the previous sections.

• Finally, according to ZNN design formula (10), we could thus have the ZNN model (1) for time-varying matrix inversion (effective as well for constant matrix inversion). Other ZNN variants [such as (1) and (2)] could then be derived readily from this original ZNN model (1).

APPENDIX B

Theorem 2 of [4]: Consider the general RNN model with implementation errors \( \Delta_p \) and \( \Delta_c \) in (8). If \( \|\Delta_p(t)\|_F \leq \varepsilon_2 \) and \( \|\Delta_c(t)\|_F \leq \varepsilon_1 \) for any \( t \in [0, \infty) \), then the computation error \( \|X-X^*\|_F \) is bounded by steady-state residual error

\[ \lim_{t \to \infty} \|X(t) - X^*(t)\|_F \leq n\varepsilon_1/(\gamma \rho - \varepsilon_2 \varphi) \]

(14)

under the design-parameter requirement \( \gamma > \varepsilon_2 \varphi/\rho \), where the parameter \( \rho > 0 \) is defined between \( f(e_{ij}(0))/e_{ij}(0) \) and \( f'(0) \). Furthermore, as \( \gamma \) tends to positive infinity, the steady-state residual error can be diminished to zero.

REFERENCES


