WILEY | Hindawi

Research Article

General Recurrent Neural Network for Solving Generalized Linear Matrix Equation

Zhan Li, Hong Cheng, and Hongliang Guo

School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

Correspondence should be addressed to Hong Cheng; hcheng@uestc.edu.cn

This brief proposes a general framework of the nonlinear recurrent neural network for solving online the generalized linear matrix equation (GLME) with global convergence property. If the linear activation function is utilized, the neural state matrix of the nonlinear recurrent neural network can globally and exponentially converge to the unique theoretical solution of GLME. Additionally, as compared with the case of using the linear activation function, two specific types of nonlinear activation functions are proposed for the general nonlinear recurrent neural network model to achieve superior convergence. Illustrative examples are shown to demonstrate the efficacy of the general nonlinear recurrent neural network model and its superior convergence when activated by the aforementioned nonlinear activation functions.

1. Introduction

Solving the generalized linear matrix equation (GLME) and its variants is an important issue which is widely encountered in scientific and engineering areas (e.g., feedback control system design [1], smart antenna array processing [2]). The well-known Lyapunov equation and Sylvester equation can be regarded as main special cases of GLME with reduced numbers of coefficients and variable matrices, which have drawn widespread interest of researchers and engineers in the past decades [3–7]. Without loss of generality, in this brief, the GLME problem is formulated as the following form:

$$\sum_{l=1}^{p} A_l X B_l = C, \tag{1}$$

where $A_l \in R^{m \times m}$, $B_l \in R^{n \times n}$, and $C \in R^{m \times n}$ denote coefficient matrices and $X \in R^{m \times n}$ denotes the unknown matrix to be obtained. Usually, it is complicated to analyze what circumstances the solution of (1) would be under in the traditional numerical way. To guarantee such GLME (1) solvable with the unique theoretical solution, coefficient matrices $A_l \in R^{m \times m}$ and $B_l \in R^{n \times n}$ can be practically configured with their eigenvalues all being positive or negative simultaneously. In many cases, the number of solutions of

(1) can be multiple or even none, depending on what kind of combinations matrices A_l and B_l would make to associate with the unknown matrix X. A lot of conventional serial approaches may be not efficient enough to solve online GLME due to their inherent drawbacks, and parallel computational approaches seem more preferable [8–13].

Regarded as another promising approach for parallel computation, dynamic neural networks based on analog solvers have been exploited comprehensively in computational intelligence fields [12, 14–16]. Different from a number of conventional numerical methods, approaches based on dynamic neural networks can be more realizable on specific parallel and distributed software or/and hardware architectures [17, 18]. This could highly enlarge utility of current neural networks towards various potential application domains towards high-performance computing. One basic type of dynamic neural networks, recurrent neural networks, which is analogous to the natural transient and steady process, has been applied in online parallel computing tasks with large-scale analog/digital circuit prototypes [19].

Our main contribution in this brief is to develop a general framework of recurrent neural network model to solve GLME (1). Since nonlinear phenomena occur frequently in neural network hardware implementation [19], the proposed general nonlinear framework may be more suitable for analog-based

computation. The neural state of the general recurrent neural network can globally converge to the theoretical solutions. If the general recurrent neural network is activated by the linear function, the exponential convergence can be achieved. On the other hand, certain nonlinear forms of such general neural network may be able to obtain more accurate solutions and faster convergence as compared with its linear model, so we propose two specific nonlinear activation functions for the general recurrent neural network model to achieve superior performance to solve GLME (1).

2. General Recurrent Neural Network Solver

In this section, we present and analyze the general model of recurrent neural network to solve GLME (1). If such model is activated by the linear function, state matrix $X(t) \in R^{m \times n}$ of the general recurrent neural network can globally and exponentially converge to the unique theoretical solution $X^* \in R^{m \times n}$. By exploiting specific nonlinear odd monotonically increasing activation functions, superior convergence is expected to be achieved. In the ensuing subsections, we will discuss their convergence properties of the general nonlinear recurrent neural network model together with its linear form.

2.1. General Nonlinear Neural Network Model. In this brief, the general nonlinear recurrent neural network model is proposed to solve GLME (1) as follows:

$$\dot{X}(t) = -\gamma \sum_{r=1}^{p} \left[A_r^T \mathscr{F} \left(\sum_{s=1}^{p} A_s X(t) B_s - C \right) B_r^T \right], \quad (2)$$

where operator $\mathscr{F}(\cdot) : R^{m \times n} \rightarrow R^{m \times n}$ denotes a nonlinear activation function array, with its each scalar-valued mapping unit $f(\cdot) : R \rightarrow R$ being a monotonically increasing odd activation function, and subscript $(\cdot)^T$ denotes transpose of matrix/vector. Such recurrent neural network model (2) can be generalized as the extended nonlinear version of recurrent neural network in [16] and the ensuing linear model. For the general nonlinear recurrent neural network model (2), we would have the following theorem.

Theorem 1. *The neural state matrix $X(t) \in R^{m \times n}$ of general nonlinear recurrent neural network model (2), starting from any initial value $X(0) \in R^{m \times n}$, can globally converge to the theoretical solution(s) $X^* \in R^{m \times n}$ of GLME (1).*

Proof. Firstly, we define the distance between the neural state and the theoretical solution as $\widetilde{X}(t) = X(t) - X^* \in R^{m \times n}$. Accordingly, by substituting $X(t) = \widetilde{X}(t) + X^* \in R^{m \times n}$ into neural network model (2), it can be further equivalently transformed as

$$\dot{\widetilde{X}}(t) = -\gamma \sum_{r=1}^{p} A_r^T \mathscr{F} \left(\sum_{s=1}^{p} A_s \widetilde{X}(t) B_s \right) B_r^T. \quad (3)$$

Next, the corresponding Lyapunov-function candidate is defined as follows:

$$V\left(\widetilde{X}(t), t\right)_{(3)} = \left\| \widetilde{X}(t) \right\|_F^2 = \left\| \text{vec}\left(\widetilde{X}(t)\right) \right\|_2^2, \quad (4)$$

where operators $\|\cdot\|_F$, $\|\cdot\|_2$, and $\otimes$, respectively, denote Frobenius norm of matrix, two norms of vector, and Kronecker product between matrices and $\text{vec}(\widetilde{X}(t)) \in R^{mn}$ generates a new column vector obtained by stacking all column vectors of $\widetilde{X}(t) \in R^{m \times n}$ together.

The time derivative of $V(\widetilde{X}(t), t)_{(3)}$ is

$$\dot{V}\left(\widetilde{X}(t), t\right)_{(3)} = 2 \text{vec}\left(\widetilde{X}(t)\right)^T \text{vec}\left(\dot{\widetilde{X}}(t)\right). \quad (5)$$

Considering the following vectorization equality based on (2),

$$\text{vec}\left(\dot{\widetilde{X}}(t)\right) = -\gamma \left(\sum_{r=1}^{p} B_r \otimes A_r^T \right)$$
$$\cdot \mathscr{F} \left(\left(\sum_{s=1}^{p} B_s^T \otimes A_s \right) \text{vec}\left(\widetilde{X}(t)\right) \right), \quad (6)$$

we can further derive (5) as

$$\dot{V}\left(\widetilde{X}(t), t\right)_{(3)} = -2\gamma \text{vec}\left(\widetilde{X}(t)\right)^T \left(\sum_{r=1}^{p} B_r \otimes A_r^T \right)$$
$$\cdot \mathscr{F} \left(\left(\sum_{s=1}^{p} B_s^T \otimes A_s \right) \text{vec}\left(\widetilde{X}(t)\right) \right)$$
$$= -2\gamma \left[\left(\sum_{l=1}^{p} B_l^T \otimes A_l \right) \text{vec}\left(\widetilde{X}(t)\right) \right]^T$$
$$\cdot \mathscr{F} \left(\left(\sum_{l=1}^{p} B_l^T \otimes A_l \right) \text{vec}\left(\widetilde{X}(t)\right) \right). \quad (7)$$

For nonlinear activation function array $\mathscr{F}(\cdot)$, its individual scalar-valued entry $f(\cdot)$ is odd and monotonically increasing, which can guarantee

$$uf(u) = \begin{cases} > 0, & u \in R, \ u \neq 0; \\ = 0, & u \in R, \ u = 0. \end{cases} \quad (8)$$

Thus, $\dot{V}(\widetilde{X}(t), t)_{(3)} \leq 0$, which implies that $\widetilde{X}(t) \in R^{m \times n}$ could globally converge to zero matrix $\mathbf{0} \in R^{m \times n}$ according to Lyapunov theory [20]; that is, state matrix $X(t) \in R^{m \times n}$ of (2) globally converges to the theoretical solution(s) $X^* \in R^{m \times n}$ of GLME (1). All of these above complete the proof. □

According to Theorem 1, the general nonlinear neural network model (2) can be activated by a number of odd monotonically increasing functions to solve GLME (1) which is with existent theoretical solutions (unique or multiple), which will broadly enlarge the utility domain of (2) towards manifold model generation. As we may know, nonlinear elements are frequently encountered in analog/digital circuit prototypes of neural networks [19, 21]; involving nonlinear activation functions can be beneficial to potential design and implication. On the other hand, faster convergence is indeed required for solving GLME (1) when the linear model

might not satisfy increasing computational requirements. With expectancy, the nonlinear neural network model (2) can attain superior convergence to that of (10) if proper activation functions are exploited. Before inducing the superior nonlinear-function activated models, we herein address the linear model of the general nonlinear recurrent neural network and discuss its convergence property.

2.2. Linear Neural Network Model. To solve GLME (1), we firstly define a scalar-valued error function $E(t) = \|\sum_{l=1}^{p} A_l X(t) B_l - C\|_F^2 / 2 \in [0, +\infty)$ associated with (1), where operator $\| \cdot \|_F$ denotes the Frobenius norm. In order to eliminate error function $E(t)$ to zero as t increases, gradient-descent manner is adopted:

$$\dot{X}(t) = -\gamma \frac{\partial E(t)}{\partial X(t)}, \tag{9}$$

where design parameter $\gamma > 0$ scales the convergence rate.

According to preliminaries on matrix-differential theory [22], (9) is further expanded to the following dynamic form:

$$\dot{X}(t) = -\gamma \sum_{s=1}^{p} \sum_{r=1}^{p} A_r^T A_s X(t) B_s B_r^T + \gamma \sum_{l=1}^{p} A_l^T C B_l^T. \tag{10}$$

For linear model (10), we would have the following theorem.

Theorem 2. *If the linear neural network model (10) is employed to solve GLME (1), starting from initial condition $X(0) \in R^{m \times n}$, the state matrix $X(t) \in R^{m \times n}$ of (10) can globally exponentially converge to unique theoretical solution $X^* \in R^{m \times n}$.*

Proof. Using the transformation $\widetilde{X}(t) = X(t) - X^* \in R^{m \times n}$ between $X(t)$ and X^* with initial condition $\widetilde{X}(0) = X(0) - X^* \in R^{m \times n}$, dynamic equation (10) is further derived equivalently as the following:

$$
\begin{aligned}
\dot{\widetilde{X}}(t) &= -\gamma \sum_{s=1}^{p} \sum_{r=1}^{p} A_r^T A_s \widetilde{X}(t) B_s B_r^T + \gamma \sum_{l=1}^{p} A_l^T C B_l^T \\
&\quad - \gamma \sum_{s=1}^{p} \sum_{r=1}^{p} A_r^T A_s X^* B_s B_r^T \\
&= -\gamma \sum_{s=1}^{p} \sum_{r=1}^{p} A_r^T A_s \widetilde{X}(t) B_s B_r^T \\
&\quad + \gamma \sum_{r=1}^{p} A_r^T \left(C - \sum_{s=1}^{p} A_s X^* B_s \right) B_r^T.
\end{aligned}
\tag{11}
$$

With $\sum_{l=1}^{p} A_l X^* B_l = C$ considered, (11) can be simplified as

$$\dot{\widetilde{X}}(t) = -\gamma \sum_{s=1}^{p} \sum_{r=1}^{p} A_r^T A_s \widetilde{X}(t) B_s B_r^T. \tag{12}$$

Similarly, we define the following Lyapunov-function candidate:

$$V\left(\widetilde{X}(t), t\right)_{(10)} = \left\| \widetilde{X}(t) \right\|_F^2 \geqslant 0, \tag{13}$$

with its time derivative being

$$
\begin{aligned}
\dot{V}\left(\widetilde{X}(t), t\right)_{(10)} &= \text{trace}\left[\left(\frac{\partial V\left(\widetilde{X}(t), t\right)}{\partial \widetilde{X}(t)} \right)^T \dot{\widetilde{X}}(t) \right] \\
&= \text{trace}\left(-2\gamma \widetilde{X}^T(t) \sum_{s=1}^{p} \sum_{r=1}^{p} A_r^T A_s \widetilde{X}(t) B_s B_r^T \right) \\
&= -2\gamma \left\| \sum_{l=1}^{p} A_l \widetilde{X}(t) B_l \right\|_F^2 \leqslant 0.
\end{aligned}
\tag{14}
$$

There exists a positive scalar $\alpha > 0$ [23] being the minimum eigenvalue of $(\sum_{l=1}^{p} B_l^T \otimes A_l)^T (\sum_{l=1}^{p} B_l^T \otimes A_l)$ satisfying

$$\left\| \sum_{l=1}^{p} A_l \widetilde{X}(t) B_l \right\|_F^2 \geqslant \alpha \left\| \widetilde{X}(t) \right\|_F^2 \tag{15}$$

if the unique solution condition of GLME (1) holds. Thus, we can have

$$\dot{V}\left(\widetilde{X}(t), t\right) \leqslant -2\gamma \alpha V\left(\widetilde{X}(t), t\right); \tag{16}$$

that is,

$$V\left(\widetilde{X}(t), t\right) \leqslant \exp\left(-2\gamma \alpha t\right) V\left(\widetilde{X}(0), 0\right), \tag{17}$$

which could be further equivalently rewritten as

$$\left\| X(t) - X^* \right\|_F \leqslant \exp\left(-\gamma \alpha t\right) \left\| X(0) - X^* \right\|_F. \tag{18}$$

By Lyapunov theory [20], (14) and (18) indicate that state matrix $X(t) \in R^{m \times n}$ of (10) can globally and exponentially converge to the unique theoretical solution $X^* \in R^{m \times n}$ of GLME (1). The proof is thus complete. □

It is worth noting that if GLME (1) is with multiple theoretical solutions $X^* \in R^{m \times n}$, scalar α equals zero. In this situation, the linear model (10) at least could guarantee its global convergence but is not able with explicit exponential convergence rate.

3. Superior Convergence with Specific Nonlinear Activation Functions

According to Theorem 1, The odd monotonically increasing activation function is able to guarantee global convergence of the general recurrent neural network (2). If the linear activation function is adopted, the general recurrent neural network model reduces to the linear model (10). Such linear model (10) possesses global exponential convergence property. In order to achieve superior convergence to global exponential convergence of the linear model (10), specific types of nonlinear activation functions should be chosen
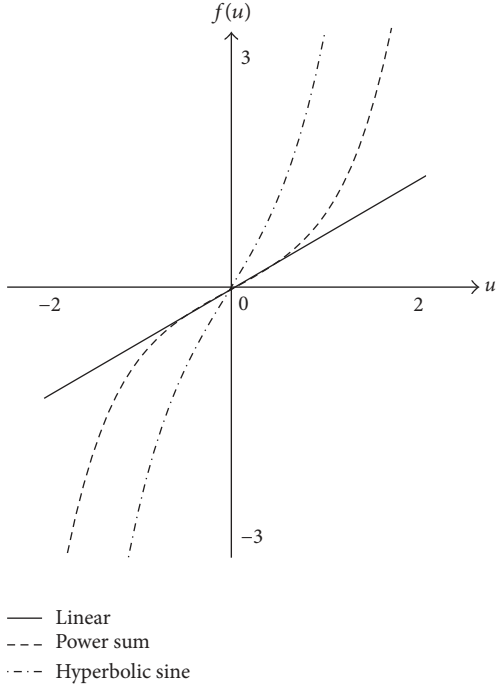
FIGURE 1: The general recurrent neural network model (2) utilizes the three types of activation functions: linear, power sum ($N = 3$), and hyperbolic sine ($\xi = 3$).

properly. Owing to the aforementioned considerations, two types of nonlinear activation functions, power sum and hyperbolic sine functions, are proposed to activate the general recurrent neural network model (2). Figure 1 shows the curve plotting of the three aforementioned activation functions used in (2). Correspondingly, we will have the following theorems on the two neural network models' convergence properties.

Theorem 3. *If the general recurrent neural network (2) is activated by the power sum function $f(u) = \sum_{k=1}^{N} u^{2k-1}$, the state matrix $X(t) \in R^{m \times n}$ of (2) can globally and superiorly converge to the unique theoretical solution $X^* \in R^{m \times n}$, as compared with linear model (10).*

Proof. In order to prove the convergence property of (2) activated by the power sum function $f(u) = \sum_{k=1}^{N} u^{2k-1}$ under this situation, we define the following Lyapunov-function candidate:

$$V_{\mathrm{ps}}\left(\widetilde{X}(t), t\right) := V\left(\widetilde{X}(t), t\right)_{(3)} = V\left(\widetilde{X}(t), t\right)_{(10)}, \quad (19)$$

with its time derivative being

$$\dot{V}_{\mathrm{ps}}\left(\widetilde{X}(t), t\right) = -2\gamma \Delta^{T} \mathscr{F}(\Delta) = -2\gamma \sum_{i=1}^{mn} \Delta_{i} f(\Delta_{i})$$

$$= -2\gamma \sum_{i=1}^{mn} \Delta_{i} \sum_{k=1}^{N} (\Delta_{i})^{2k-1}$$

$$= -2\gamma \sum_{i=1}^{mn} \sum_{k=1}^{N} (\Delta_{i})^{2k} \leqslant -2\gamma \sum_{i=1}^{mn} (\Delta_{i})^{2}$$

$$= \dot{V}\left(\widetilde{X}(t), t\right)_{(10)},$$

$$(20)$$

where

$$\Delta = \left[\Delta_{1}, \Delta_{2}, \ldots, \Delta_{i}, \ldots, \Delta_{mn}\right]^{T} \in R^{mn} \quad (21)$$

and $\Delta_{i} \in R$ denotes the ith element of vector $(\sum_{l=1}^{p} B_{l}^{T} \otimes A_{l})\mathrm{vec}(\widetilde{X}(t)) \in R^{mn}$. This implies that when power sum functions are used, (2) possesses global convergence to zero matrix, with larger Lyapunov-function vanishing rate (i.e., faster convergence), as compared with the situation of (10). The proof is thus complete. $\square$

Theorem 4. *If the general recurrent neural network (2) is activated by hyperbolic sine function $f(u) = \exp(\xi u)/2 - \exp(-\xi u)/2$ with coefficient $\xi \geqslant 1$, the state matrix $X(t) \in R^{m \times n}$ of (2) can globally and superiorly converge to the unique theoretical solution $X^* \in R^{m \times n}$, as compared with the linear model (10).*

Proof. Similarly, the following Lyapunov function is defined to investigate convergence:

$$V_{\mathrm{hs}}\left(\widetilde{X}(t), t\right) := V_{\mathrm{ps}}\left(\widetilde{X}(t), t\right) = V\left(\widetilde{X}(t), t\right)_{(10)}, \quad (22)$$

and its time derivative is

$$\dot{V}_{\mathrm{hs}}\left(\widetilde{X}(t), t\right) = -2\gamma \sum_{i=1}^{mn} \Delta_{i} f(\Delta_{i})$$

$$= -\gamma \sum_{i=1}^{mn} \Delta_{i} \left(\exp(\xi \Delta_{i}) - \exp(-\xi \Delta_{i})\right)$$

$$= -2\gamma \sum_{i=1}^{mn} \sum_{j=1}^{+\infty} \frac{(\xi \Delta_{i})^{2j}}{(2j-1)!} \leqslant -2\gamma \sum_{i=1}^{mn} (\Delta_{i})^{2}$$

$$= \dot{V}\left(\widetilde{X}(t), t\right)_{(10)},$$

$$(23)$$

which indicates that when the hyperbolic sine activation function is employed, the nonlinear recurrent neural network model (2) possesses global convergence as its state matrix is approaching zero, with larger Lyapunov-function vanishing rate (i.e., faster convergence) as compared with the situation of linear model (10). These complete the proof. $\square$

4. Illustrative Examples

In this section, three examples are presented to illustrate the efficiency of the general nonlinear recurrent neural network (2) with its specific models under different types of activation functions (linear, power sum, and hyperbolic sine activation functions) for online solving GLME (1).
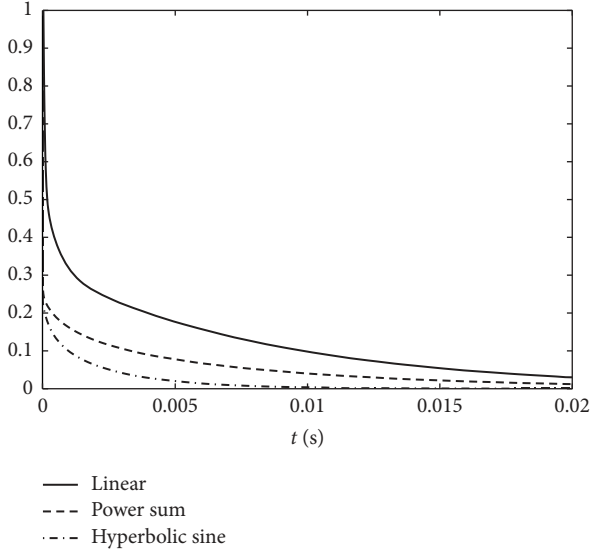
FIGURE 2: Solution error $\|X(t) - X^*\|_F$ synthesized by (2) for GLME (24) with the unique theoretical solution X^*, starting from the same randomly generated initial state $X(0) \in R^{2\times2}$.

Example 1. Let us consider the following GLME with $l = 2$:

$$A_1 X B_1 + A_2 X B_2 = C, \qquad (24)$$

where

$$A_1 = \begin{bmatrix} 2 & 1 \\ 1 & 3 \end{bmatrix},$$

$$A_2 = \begin{bmatrix} 6 & 2 \\ 3 & 4 \end{bmatrix},$$

$$B_1 = \begin{bmatrix} 5 & 3 \\ 3 & 4 \end{bmatrix}, \qquad (25)$$

$$B_2 = \begin{bmatrix} 6 & 2 \\ 5 & 8 \end{bmatrix},$$

$$C = \begin{bmatrix} 3 & 4 \\ 5 & 6 \end{bmatrix}.$$

GLME (24) has unique theoretical solution

$$X^* = \begin{bmatrix} -0.0023 & 0.0308 \\ 0.0395 & 0.1019 \end{bmatrix}, \qquad (26)$$

since the eigenvalues of coefficient matrices A_1, A_2, B_1, and B_2 are all positive values. We employ the general recurrent neural network model (2) with $\gamma = 1$ activated by linear function, power sum function with $N = 4$, and hyperbolic sine function with $\xi = 3$.

From Figure 2, we could observe that the solution errors $\|X(t) - X^*\|_F$ decline to almost zero at around 0.02 s and faster convergence to the solution can be achieved with power sum

and hyperbolic sine activation functions used in (2). These can demonstrate the effectiveness of the general recurrent neural network model (2) for solving GLME (24).

Example 2. Let us consider the following GLME with multiple theoretical solutions $X^* \in R^{2\times2}$:

$$\widetilde{A}_1 X \widetilde{B}_1 + \widetilde{A}_2 X \widetilde{B}_2 = \widetilde{C}, \qquad (27)$$

where

$$\widetilde{A}_1 = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix},$$

$$\widetilde{A}_2 = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix},$$

$$\widetilde{B}_1 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, \qquad (28)$$

$$\widetilde{B}_2 = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix},$$

$$\widetilde{C} = \begin{bmatrix} 0 & 3 \\ 0 & 5 \end{bmatrix}.$$

We use linear model (10) with design parameter $\gamma = 1$ to solve GLME (27). The trajectories of entries of state matrix $X(t) \in R^{2\times2}$ are shown in Figure 3. From Figure 3, we could see that, starting from two different initial matrices $X(0) \in R^{2\times2}$, the state matrices $X(t) \in R^{2\times2}$ of linear model (10), respectively, converge to two different trajectories (or say two different theoretical solutions $X^* \in R^{2\times2}$). This indicates that the choices of the initial value impact greatly the steady-state results of the recurrent neural network (2) and determine the starting points of convergence for solution of GLME (27), if multiple theoretical solutions exist for GLME (27). Correspondingly, the residual errors $\|\widetilde{A}_1 X(t) \widetilde{B}_1 + \widetilde{A}_2 X(t) \widetilde{B}_2 - \widetilde{C}\|_F$ synthesized by (10) can always diminish to zero within finite time from twenty different initial values, as illustrated by Figure 4.

Example 3. Let us consider the following GLME in a larger dimension with $l = 10$:

$$\sum_{l=1}^{10} A_l X B_l = C, \qquad (29)$$

where coefficient matrices $A_l \in R^{10\times10}$, $B_l \in R^{10\times10}$, and $C \in R^{10\times10}$ are all positive-definite randomly generated and fall within interval $[-2, 2] \in R^{10\times10}$. We exploit nonlinear neural network models (2) activated by power sum and hyperbolic sine functions and the linear model (10) to solve GLME (29) with design parameter $\gamma = 1$. From Table 1, we could observe that general recurrent neural network models (2) activated by power sum and hyperbolic sine activation functions exhibit faster error diminishing speed than that of the linear model (10), with all of their residual errors reaching the level of
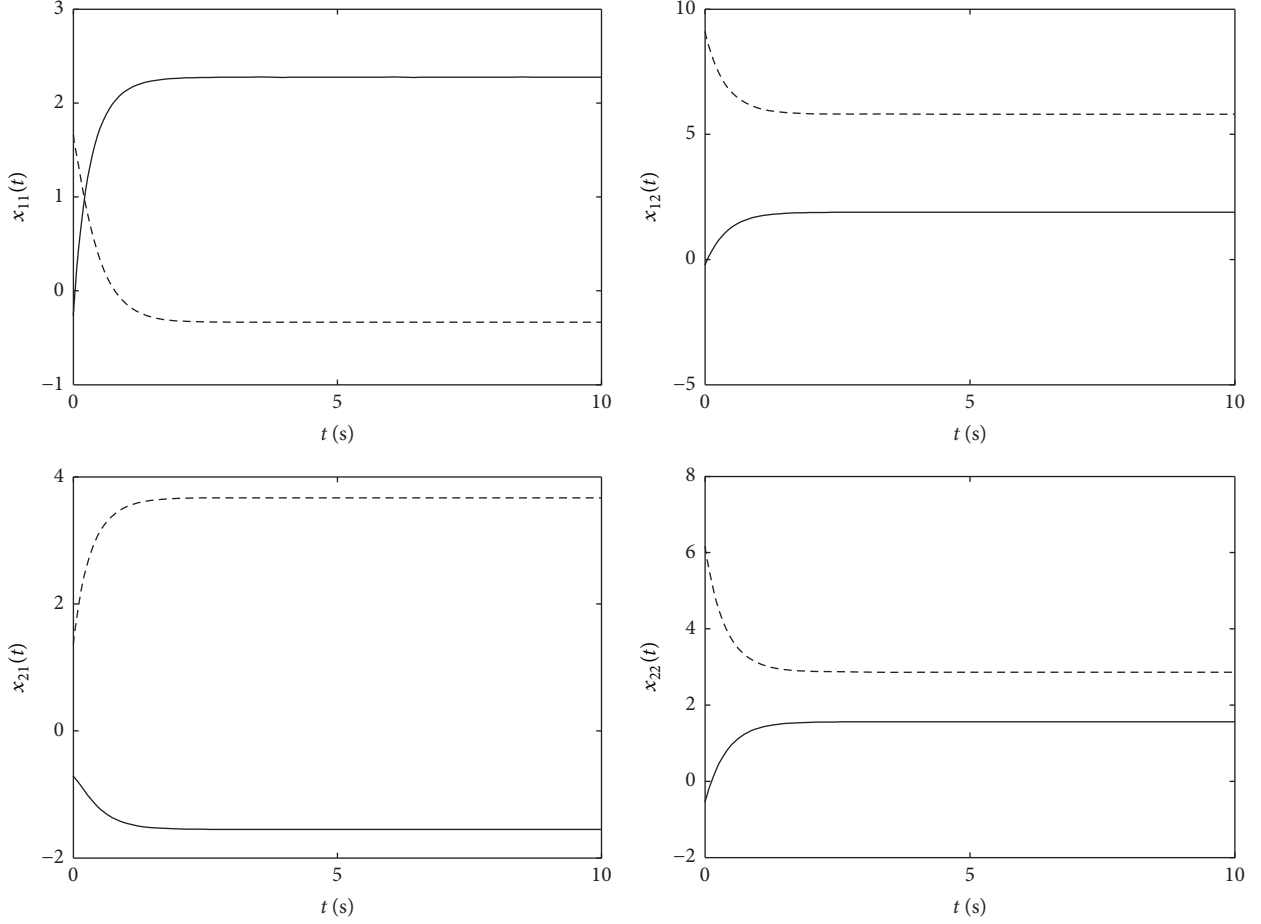
FIGURE 3: Trajectories of state matrix $X(t) \in R^{2\times2}$ for solving GLME (27) with the multiple theoretical solutions X^*, starting from two different initial states $X(0) \in R^{2\times2}$.
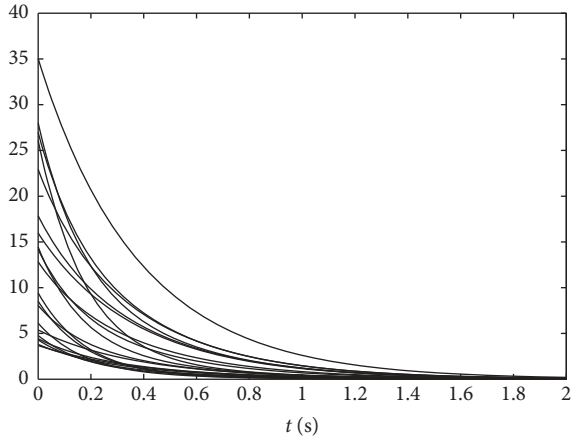

FIGURE 4: Residual error $\|\widetilde{A}_1 X(t)\widetilde{B}_1 + \widetilde{A}_2 X(t)\widetilde{B}_2 - \widetilde{C}\|_F$ synthesized by (2) for solution of GLME (27), starting from twenty different initial states $X(0) \in R^{2\times2}$.

10^{-7} within 1 s. From computational results of these three examples above, we could see that the proposed general nonlinear recurrent neural network can solve the GLME (1) problem well.

TABLE 1: Performance of the general recurrent neural network model (2) with three different activation functions (linear, power sum, and hyperbolic sine) for solving GLME (29).

	Residual error $\left\|\sum_{l=1}^{10} A_l X(t)B_l - C\right\|_F$		
Time t (s)	Linear	Power sum	Hyperbolic sine
$t = 0$ s	37.4543	37.4543	37.4543
$t = 0.1$ s	0.9886	0.3678	0.5242
$t = 1$ s	4.6368×10^{-7}	9.7531×10^{-8}	3.0543×10^{-7}

5. Conclusion

In this brief, we present a general recurrent neural network model for solving GLME. The general nonlinear model of recurrent neural network possesses global convergence property in finding solutions of GLME. By using specifically proposed nonlinear activation functions, superior convergence can be achieved, as compared with the linear model which is with exponential convergence rate. Illustrative results are shown to demonstrate the effectiveness and superiority of nonlinear recurrent neural network models for solution of GLME.
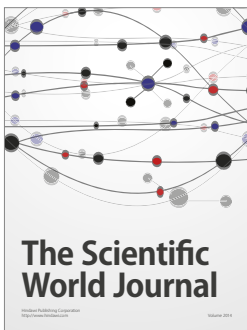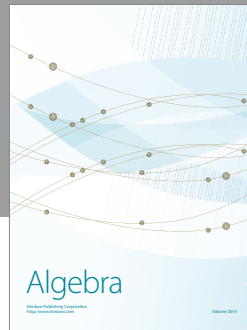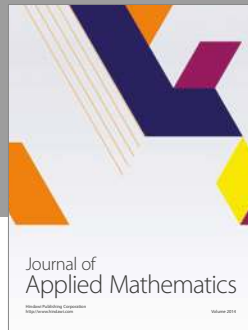
Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

References

[1] V. Hernandez, E. S. Quintana, and M. Marques, "Solving linear matrix equations in control problems on distributed memory multiprocessors," in *Proceedings of the 33rd IEEE Conference on Decision and Control. Part 1 (of 4)*, pp. 448–454, December 1994.

[2] H. Lev-Ari, "Efficient solution of linear matrix equations with application to multistatic antenna array processing," *Communications in Information & Systems*, vol. 5, no. 1, pp. 123–130, 2005.

[3] Y. Fang, K. A. Loparo, and X. Feng, "New estimates for solutions of Lyapunov equations," *Institute of Electrical and Electronics Engineers. Transactions on Automatic Control*, vol. 42, no. 3, pp. 408–411, 1997.

[4] W. H. Kwon, Y. S. Moon, and S. C. Ahn, "Bounds in algebraic Riccati and Lyapunov equations: a survey and some new results," *International Journal of Control*, vol. 64, no. 3, pp. 377–389, 1996.

[5] B. Zhou and G.-R. Duan, "A new solution to the generalized sylvester matrix equation av-evf= bw," *Systems & Control Letters*, vol. 2, no. 3, pp. 193–198, 2006.

[6] B. Zhou and G.-R. Duan, "On the generalized sylvester mapping and matrix equations," *Systems & Control Letters*, vol. 57, no. 3, pp. 200–208, 2008.

[7] S. Li and Y. Li, "Nonlinearly activated neural network for solving time-varying complex sylvester equation," *IEEE Transactions on Cybernetics*, vol. 44, no. 8, pp. 1397–1407, 2014.

[8] Y. Zhang, D. Jiang, and J. Wang, "A recurrent neural network for solving sylvester equation with time-varying coefficients," *IEEE Transactions on Neural Networks*, vol. 13, no. 5, pp. 1053–1063, 2002.

[9] F. Ding and T. Chen, "Gradient based iterative algorithms for solving a class of matrix equations," *IEEE Transactions on Automatic Control*, vol. 50, no. 8, pp. 1216–1221, 2005.

[10] L. Xie, J. Ding, and F. Ding, "Gradient based iterative solutions for general linear matrix equations," *Computers & Mathematics with Applications*, vol. 58, no. 7, pp. 1441–1448, 2009.

[11] F. Ding and T. Chen, "Iterative least-squares solutions of coupled Sylvester matrix equations," *Systems & Control Letters*, vol. 54, no. 2, pp. 95–107, 2005.

[12] Y. Chen, C. Yi, and D. Qiao, "Improved neural solution for the Lyapunov matrix equation based on gradient search," *Information Processing Letters*, vol. 113, no. 22-24, pp. 876–881, 2013.

[13] K. Chen, "Improved neural dynamics for online Sylvester equations solving," *Information Processing Letters*, vol. 116, no. 7, pp. 455–459, 2016.

[14] S. Zhang and A. G. Constantinides, "Lagrange programming neural networks," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 39, no. 7, pp. 441–452, 1992.

[15] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 79, no. 8, pp. 2554–2558, 1982.

[16] J. Wang, "Recurrent neural networks for solving linear matrix equations," *Computers & Mathematics with Applications*, vol. 26, no. 9, pp. 23–34, 1993.

[17] Z. Zhang and Y. Zhang, "Design and experimentation of acceleration-level drift-free scheme aided by two recurrent neural networks," *IET Control Theory Applications*, vol. 7, no. 1, pp. 25–42, 2013.

[18] Z. Zhang, Z. Li, Y. Zhang, Y. Luo, and Y. Li, "Neural-dynamic-method-based dual-arm CMG scheme with time-varying constraints applied to humanoid robots," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 12, pp. 3251–3262, 2015.

[19] C. Mead and M. Ismail, *Analog VLSI Implementation of Neural Systems*, vol. 80, Springer Science & Business Media, 2012.

[20] S. Sastry and M. Bodson, *Adaptive Control: Stability, Convergence and Robustness*, Courier Corporation, 2011.

[21] J. Liu, M. A. Brooke, and K. Hirotsu, "A CMOS feedforward neural-network chip with on-chip parallel learning for oscillation cancellation," *IEEE Transactions on Neural Networks*, vol. 13, no. 5, pp. 1178–1186, 2002.

[22] A. H. Roger and R. J. Charles, *Matrix Analysis*, Cambridge University Press, New York, NY, USA, 2nd edition, 2012.

[23] Y. Zhang and K. Chen, "Global exponential convergence and stability of Wang neural network for solving online linear equations," *Electronics Letters*, vol. 44, no. 2, pp. 145-146, 2008.