# $\mathscr{L}_2$–$\mathscr{L}_\infty$ learning of dynamic neural networks*

Choon Ki Ahn†

*Department of Automotive Engineering, Seoul National University of Technology, 172 Gongneung 2-dong, Nowon-gu, Seoul 139-743, Korea*

(Received 21 November 2009; revised manuscript received 5 April 2010)

This paper proposes an $\mathscr{L}_2$–$\mathscr{L}_\infty$ learning law as a new learning method for dynamic neural networks with external disturbance. Based on linear matrix inequality (LMI) formulation, the $\mathscr{L}_2$–$\mathscr{L}_\infty$ learning law is presented to not only guarantee asymptotical stability of dynamic neural networks but also reduce the effect of external disturbance to an $\mathscr{L}_2$–$\mathscr{L}_\infty$ induced norm constraint. It is shown that the design of the $\mathscr{L}_2$–$\mathscr{L}_\infty$ learning law for such neural networks can be achieved by solving LMIs, which can be easily facilitated by using some standard numerical packages. A numerical example is presented to demonstrate the validity of the proposed learning law.

**Keywords:** $\mathscr{L}_2$–$\mathscr{L}_\infty$ learning law, dynamic neural networks, linear matrix inequality, Lyapunov stability theory

**PACC:** 0200, 0290, 0230

## 1. Introduction

In recent years, dynamic neural networks have attracted considerable attention as they proved to be essential in applications such as pattern recognition, associative memories, signal processing, fixed-point computations, and so on.[1] In such applications, it is of prime importance to know that the designed neural network is stable for each external input vector. Two kinds of stability for dynamic neural networks have been studied. The stability of dynamic neural networks may be found in Refs. [2]–[6]. The stability of learning algorithms was discussed in Refs. [7]–[13]. In this paper, we emphasize deriving a new learning algorithm of dynamic neural networks.

The stability of learning algorithms can be derived by analysing the identification or tracking errors of neural networks. Reference [8] studied the stability conditions of learning laws in a situation where neural networks were used to identify and control a nonlinear system. The notion of persistency of excitation and passivity properties of neural networks were defined and used in the convergence analysis of both the identification error and the weight estimates. In Ref. [9], the dynamic backpropagation was modified with $NL_q$ stability constraints. Sufficient conditions for global asymptotical stability on diagonal scaling and diagonal dominance were proposed for checking stability of the learning algorithm. In Refs. [10] and [12], passivity-based learning laws were proposed for neural networks. It was shown that the passivity-based learning laws can guarantee some stability properties, such as passivity, asymptotical stability, input-to-state stability, and bounded input-bounded output stability of neural networks. Since neural networks cannot match the unknown nonlinear systems exactly, extensive modifications should be made to either the normal gradient algorithm or to the backpropagation algorithm.[7,8,11,13] Despite these advances in neural network learning, most research results were restricted to neural networks without external disturbances. Can we obtain a new learning law for neural networks with external disturbances? This paper gives an answer to this question.

In real physical systems, one is faced with model uncertainties and a lack of statistical information on the signals. This had led in recent years to an interest in $\mathscr{L}_2$–$\mathscr{L}_\infty$ control and filtering, with the belief that an $\mathscr{L}_2$–$\mathscr{L}_\infty$ approach is more robust and less sensitive to disturbance variances and model uncertainties.[14−21] To the best of our knowledge, however, for the $\mathscr{L}_2$–$\mathscr{L}_\infty$ learning law for dynamic neural networks, there is no result in the literature so far, which still remains open and challenging.

In this paper, for the first time, an $\mathscr{L}_2$–$\mathscr{L}_\infty$ approach is used to derive a new learning algorithm for

*Project supported by the Grant of the Korean Ministry of Education, Science and Technology (The Regional Core Research Program/Center for Healthcare Technology Development).

†Corresponding author. E-mail: hironaka@hanmail.net

© 2010 Chinese Physical Society and IOP Publishing Ltd

dynamic neural networks with external disturbance. By the proposed $\mathscr{L}_2$–$\mathscr{L}_\infty$ learning law, the neural network is asymptotically stable and the $\mathscr{L}_2$–$\mathscr{L}_\infty$ induced norm from the external disturbance to the state vector is reduced to a disturbance attenuation level. Based on Lyapunov stability theory and linear matrix inequality (LMI) formulation, the design of the $\mathscr{L}_2$–$\mathscr{L}_\infty$ learning law for dynamic neural networks can be realized by solving LMIs, which can be facilitated readily via standard numerical algorithms.[22,23]

This paper is organized as follows. In Section 2, we formulate the problem. In Section 3, an LMI problem for the $\mathscr{L}_2$–$\mathscr{L}_\infty$ learning law of dynamic neural networks is proposed. In Section 4, a numerical example is given, and finally, conclusions are presented in Section 5.

## 2. Problem formulation

Consider the following dynamic neural network:

$$\begin{aligned}\dot{x}(t) &= Ax(t) + W(t)\theta(x(t)) \\ &\quad + \bar{W}(t)\phi(x(t))\sigma(u(t)) + Gd(t), \qquad (1)\\ z(t) &= Hx(t), \qquad (2)\end{aligned}$$

where $x(t) = [x_1(t) \ \dots \ x_n(t)]^{\mathrm{T}} \in R^n$ is the state vector, $z(t) \in R^p$ is a linear combination of the states, $u(t) = [u_1(t) \ \dots \ u_m(t)]^{\mathrm{T}} \in R^m$ is the input vector, $d(t) = [d_1(t) \ \dots \ d_k(t)]^{\mathrm{T}} \in R^k$ is the disturbance vector, $A \in R^{n\times n}$ is the state matrix, $W(t) \in R^{n\times n}$ is the nonlinear feedback weight matrix, $\bar{W}(t) \in R^{n\times n}$ is the input weight matrix, $G \in R^{n\times k}$ and $H \in R^{p\times n}$ are known constant matrices, $\theta(x(t)) = [\theta_1(x(t)) \ \dots \ \theta_n(x(t))]^{\mathrm{T}} : R^n \to R^n$ and $\sigma(u(t)) = [\sigma_1(u(t)) \ \dots \ \sigma_n(u(t))]^{\mathrm{T}} : R^m \to R^n$ are the nonlinear function vectors, and $\phi(x(t)) = [\phi_{ij}(x(t))] : R^n \to R^{n\times n}$ $(i, j = 1, \dots, n)$ is the nonlinear function matrix. The elementary functions $\theta_i(x(t))$ and $\phi_{ij}(x(t))$ $(i, j = 1, \dots, n)$ are usually selected as sigmoid functions. We assume that $A$ is a stable matrix, which is a common assumption for stability analysis of dynamic neural networks. One can see that the Hopfield model[24] is a special case of this kind of neural networks with $A = \{-1/R_iC_i\}$ $(i = 1, \dots, n)$, where $R_i > 0$ and $C_i > 0$ are the resistance and capacitance at the $i$-th node of the neural network, respectively.

**Definition 1** (Asymptotical stability) The neural network (1) is asymptotically stable if the state vector $x(t)$ satisfies

$$\lim_{t\to\infty} x(t) = 0 \qquad (3)$$

under non-zero initial conditions.

**Definition 2** ($\mathscr{L}_2$–$\mathscr{L}_\infty$ stability) The neural network (1) is $\mathscr{L}_2$–$\mathscr{L}_\infty$ stable if the signal $z(t)$ satisfies

$$\sup_{t\geq 0}\{z^{\mathrm{T}}(t)z(t)\} < \gamma^2 \int_0^\infty d^{\mathrm{T}}(t)d(t)\mathrm{d}t \qquad (4)$$

for a given level $\gamma > 0$ under zero initial conditions. The parameter $\gamma$ is called the $\mathscr{L}_2$–$\mathscr{L}_\infty$ induced norm bound or the $\mathscr{L}_2$–$\mathscr{L}_\infty$ disturbance attenuation level.

**Remark 1** The $\mathscr{L}_2$–$\mathscr{L}_\infty$ induced norm[25,26] is defined as

$$\|T_{zd}\|_{\mathscr{L}_2-\mathscr{L}_\infty} = \frac{\sqrt{\sup_{t\geq 0}\{z^{\mathrm{T}}(t)z(t)\}}}{\sqrt{\int_0^\infty d^{\mathrm{T}}(t)d(t)\mathrm{d}t}},$$

where $T_{zd}$ is a transfer function matrix from $d(t)$ to $z(t)$. For a given level $\gamma > 0$, $\|T_{zd}\|_{\mathscr{L}_2-\mathscr{L}_\infty} < \gamma$ can be restated in the equivalent form (4). If we define

$$L(t) = \frac{\sup_{0\leq\sigma<t}\{z^{\mathrm{T}}(\sigma)z(\sigma)\}}{\int_0^t d^{\mathrm{T}}(\sigma)d(\sigma)\mathrm{d}\sigma}, \qquad (5)$$

the relation (4) can be represented by

$$L(\infty) < \gamma^2. \qquad (6)$$

In Section 4, through the plot of $L(t)$ versus time, the relation (6) is verified.

The objective of this study is to design a learning algorithm for the neural network (1) with a guaranteed performance in the $\mathscr{L}_2$–$\mathscr{L}_\infty$ sense. That is, given a prescribed level of disturbance attenuation $\gamma$, find a learning algorithm such that the neural network (1) with $d(t) = 0$ is asymptotically stable and $\mathscr{L}_2$–$\mathscr{L}_\infty$ stable under zero-initial conditions for all nonzero $d(t)$.

## 3. Main results

In this section, we will derive the $\mathscr{L}_2$–$\mathscr{L}_\infty$ learning law for the neural network (1). The following theorem presents an LMI-based criterion to check the $\mathscr{L}_2$–$\mathscr{L}_\infty$ performance of the neural network (1).

**Theorem 1** Let $\epsilon_1$ and $\epsilon_2$ be positive scalars. For a given level $\gamma > 0$, assume that there exists $P = P^{\mathrm{T}} > 0$ such that

$$\begin{bmatrix} A^{\mathrm{T}}P + PA & PG \\ G^{\mathrm{T}}P & -I \end{bmatrix} < 0, \qquad (7)$$

$$\begin{bmatrix} P & H^{\mathrm{T}} \\ H & \gamma^2 I \end{bmatrix} > 0. \qquad (8)$$

If the weights $W(t)$ and $\bar{W}(t)$ are updated as

$$\dot{W}(t) = -\Phi\left[Px(t)\theta^{\mathrm{T}}(x(t)) + \epsilon_1 W(t)\right], \tag{9}$$

$$\dot{\bar{W}}(t) = -\Psi\left[Px(t)\sigma^{\mathrm{T}}(u(t))\phi^{\mathrm{T}}(x(t)) + \epsilon_2 \bar{W}(t)\right], \tag{10}$$

where $\Phi$ and $\Psi$ are symmetric positive-definite matrices, then the $\mathscr{L}_2$–$\mathscr{L}_\infty$ stability with the disturbance attenuation level $\gamma$ is achieved.

**Proof** Consider the following Lyapunov function:

$$V(t) = x^{\mathrm{T}}(t)Px(t) + \mathrm{trace}\Big\{W^{\mathrm{T}}(t)\,\Phi^{-1}W(t)\Big\} + \mathrm{trace}\Big\{\bar{W}^{\mathrm{T}}(t)\,\Psi^{-1}\bar{W}(t)\Big\}, \tag{11}$$

where trace$\{\cdot\}$ stands for the "trace" and is defined as the sum of all the diagonal elements of a matrix. The time derivative of $V(t)$ along the trajectory of Eq. (1) is

$$\begin{aligned}
\dot{V}(t) &= \dot{x}(t)^{\mathrm{T}}Px(t) + x^{\mathrm{T}}(t)P\dot{x}(t) + 2\ \mathrm{trace}\left\{\dot{W}^{\mathrm{T}}(t)\,\Phi^{-1}W(t)\right\} + 2\ \mathrm{trace}\left\{\dot{\bar{W}}^{\mathrm{T}}(t)\,\Psi^{-1}\bar{W}(t)\right\} \\
&= x^{\mathrm{T}}(t)\big[A^{\mathrm{T}}P + PA\big]x(t) + x^{\mathrm{T}}(t)PGd(t) + d^{\mathrm{T}}(t)G^{\mathrm{T}}Px(t) + 2x^{\mathrm{T}}(t)PW(t)\theta(x(t)) \\
&\quad + 2x(t)^{\mathrm{T}}P\bar{W}(t)\phi(x(t))\sigma(u(t)) + 2\ \mathrm{trace}\Big\{\dot{W}^{\mathrm{T}}(t)\,\Phi^{-1}W(t)\Big\} + 2\ \mathrm{trace}\Big\{\dot{\bar{W}}^{\mathrm{T}}(t)\,\Psi^{-1}\bar{W}(t)\Big\} \\
&= x^{\mathrm{T}}(t)\big[A^{\mathrm{T}}P + PA\big]x(t) + x^{\mathrm{T}}(t)PGd(t) + d^{\mathrm{T}}(t)G^{\mathrm{T}}Px(t) + 2\ \mathrm{trace}\Big\{\theta(x(t))x^{\mathrm{T}}(t)PW(t)\Big\} \\
&\quad + 2\ \mathrm{trace}\Big\{\phi(x(t))\sigma(u(t))x(t)^{\mathrm{T}}P\bar{W}(t)\Big\} + 2\ \mathrm{trace}\Big\{\dot{W}^{\mathrm{T}}(t)\,\Phi^{-1}W(t)\Big\} + 2\ \mathrm{trace}\Big\{\dot{\bar{W}}^{\mathrm{T}}(t)\,\Psi^{-1}\bar{W}(t)\Big\} \\
&= \begin{bmatrix} x(t) \\ d(t) \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} A^{\mathrm{T}}P + PA & PG \\ G^{\mathrm{T}}P & -I \end{bmatrix} \begin{bmatrix} x(t) \\ d(t) \end{bmatrix} + d^{\mathrm{T}}(t)d(t) + 2\ \mathrm{trace}\Big\{\Big[\theta(x(t))x^{\mathrm{T}}(t)P + \dot{W}^{\mathrm{T}}(t)\,\Phi^{-1}\Big]W(t)\Big\} \\
&\quad + 2\ \mathrm{trace}\Big\{\Big[\phi(x(t))\sigma(u(t))x(t)^{\mathrm{T}}P + \dot{\bar{W}}^{\mathrm{T}}(t)\,\Psi^{-1}\Big]\bar{W}(t)\Big\}.
\end{aligned}$$

If we use the updating laws (9)–(10) and the LMI (7) is satisfied, we have

$$\dot{V}(t) < d^{\mathrm{T}}(t)d(t) - 2\epsilon_1\ \mathrm{trace}\Big\{W^{\mathrm{T}}(t)W(t)\Big\} - 2\epsilon_2\ \mathrm{trace}\Big\{\bar{W}^{\mathrm{T}}(t)\bar{W}(t)\Big\} \tag{12}$$

$$\leq d^{\mathrm{T}}(t)d(t). \tag{13}$$

Under the zero-initial condition, one has $V(t)|_{t=0} = 0$ and $V(t) \geq 0$. Define

$$J(t) = V(t) - \int_0^t d^{\mathrm{T}}(\sigma)d(\sigma)\mathrm{d}\sigma. \tag{14}$$

Then, for any nonzero $d(t)$, we obtain

$$\begin{aligned}
J(t) &= V(t) - V(t)|_{t=0} - \int_0^t d^{\mathrm{T}}(\sigma)d(\sigma)\mathrm{d}\sigma \\
&= \int_0^t \big[\dot{V}(\sigma) - d^{\mathrm{T}}(\sigma)d(\sigma)\big]\mathrm{d}\sigma.
\end{aligned}$$

From inequality (13), we have $J(t) < 0$. It means that

$$V(t) < \int_0^t d^{\mathrm{T}}(\sigma)d(\sigma)\mathrm{d}\sigma.$$

On the other hand, by the Schur complement, the LMI (8) is equivalent to

$$P - \frac{1}{\gamma^2}H^{\mathrm{T}}H > 0. \tag{15}$$

Noting expressions (2) and (11), we find that inequality (15) implies

$$\begin{aligned}
z^{\mathrm{T}}(t)z(t) &= x^{\mathrm{T}}(t)H^{\mathrm{T}}Hx(t) \leq \gamma^2 x^{\mathrm{T}}(t)Px(t) \\
&\leq \gamma^2 V(t) < \gamma^2 \int_0^t d^{\mathrm{T}}(\sigma)d(\sigma)\mathrm{d}\sigma \\
&\leq \gamma^2 \int_0^\infty d^{\mathrm{T}}(\sigma)d(\sigma)\mathrm{d}\sigma.
\end{aligned} \tag{16}$$

Taking the supremum over $t > 0$ we obtain inequality (4). This completes the proof.

**Corollary 1** Without the external disturbance, if we use the learning laws (9) and (10), the asymptotical stability is obtained.

**Proof** When $d(t) = 0$, we obtain

$$\begin{aligned}
\dot{V}(t) &< -2\epsilon_1\ \mathrm{trace}\Big\{W^{\mathrm{T}}(t)W(t)\Big\} \\
&\quad - 2\epsilon_2\ \mathrm{trace}\Big\{\bar{W}^{\mathrm{T}}(t)\bar{W}(t)\Big\} \\
&\leq 0
\end{aligned}$$

from inequality (12). This guarantees

$$\lim_{t\to\infty} x(t) = 0 \tag{17}$$

from Lyapunov stability theory. This completes the proof.

Based on Theorem 1, the optimal $\mathscr{L}_2$–$\mathscr{L}_\infty$ induced norm bound for the $\mathscr{L}_2$–$\mathscr{L}_\infty$ learning law is obtained.

**Corollary 2** The optimal $\mathscr{L}_2$–$\mathscr{L}_\infty$ induced norm bound $\gamma$ is obtained by solving the following semi-definite programming problem:

$$\min_{\gamma>0} \gamma^2 \tag{18}$$

subject to inequalities (7), (8), and $P > 0$.

**Remark 2** The LMI problem given in Theorem 1 is to determine whether the solution exists or not. It is called the feasibility problem. Also, the solution of the problem in Corollary 2 can be found by solving a convex optimization problem in terms of the LMI. The LMI problems can be solved efficiently by using recently developed convex optimization algorithms.[22] In this paper, in order to solve the LMI problems, we utilize MATLAB LMI Control Toolbox,[23] which implements state-of-the-art interior-point algorithms.

# 4. Numerical example

Consider the following dynamic neural network:

$$\begin{aligned}\dot{x}(t) &= Ax(t) + W(t)\theta(x(t)) \\ &\quad + \bar{W}(t)\phi(x(t))\sigma(u(t)) + Gd(t), \qquad (19)\\ z(t) &= Hx(t), \qquad (20)\end{aligned}$$

where

$$x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix},$$

$$u(t) = \begin{bmatrix} u_1(t) \\ u_2(t) \end{bmatrix} = \begin{bmatrix} 2t \\ 5t \end{bmatrix},$$

$$d(t) = \begin{bmatrix} d_1(t) \\ d_2(t) \end{bmatrix}, \quad A = \begin{bmatrix} -1 & 0 \\ 0 & -0.6 \end{bmatrix},$$

$$G = \begin{bmatrix} 0.3 & 0.2 \\ 0 & 0.4 \end{bmatrix}, \quad H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

$$W(t) = \begin{bmatrix} W_{11}(t) & W_{12}(t) \\ W_{21}(t) & W_{22}(t) \end{bmatrix},$$

$$\bar{W}(t) = \begin{bmatrix} \bar{W}_{11}(t) & \bar{W}_{12}(t) \\ \bar{W}_{21}(t) & \bar{W}_{22}(t) \end{bmatrix},$$

$$\theta(x(t)) = \begin{bmatrix} \dfrac{1}{1+\mathrm{e}^{-x_1(t)}} \\ \dfrac{1}{1+\mathrm{e}^{-x_2(t)}} \end{bmatrix},$$

$$\phi(x(t)) = \begin{bmatrix} \dfrac{1}{1+\mathrm{e}^{-x_1(t)}} & \dfrac{1}{1+\mathrm{e}^{-x_1(t)}} \\ \dfrac{1}{1+\mathrm{e}^{-x_2(t)}} & \dfrac{1}{1+\mathrm{e}^{-x_2(t)}} \end{bmatrix},$$

$$\sigma(u(t)) = \begin{bmatrix} \cos(u_1(t)) \\ \sin(u_2(t)) \end{bmatrix}.$$

For the numerical simulation, we use the following parameters:

$$\epsilon_1 = \epsilon_2 = 0.5, \qquad \Phi = \Psi = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \tag{21}$$

Let the $\mathscr{L}_2$–$\mathscr{L}_\infty$ performance be specified by $\gamma = 0.4$. Applying Theorem 1 to the neural networks (19) and (20) yields

$$P = \begin{bmatrix} 18.0181 & -6.2332 \\ -6.2332 & 9.5543 \end{bmatrix}.$$

Figure 1 shows the plot of $L(t)$ versus time when $d_1(t) = \cos(20t)$ and $d_2(t) = \sin(10t)$. Figure 1 verifies $L(\infty) < \gamma^2 = 0.16$. This means that the $\mathscr{L}_2$–$\mathscr{L}_\infty$ induced norm from the external disturbance $d(t)$ to the state vector $x(t)$ is reduced within the $\mathscr{L}_2$–$\mathscr{L}_\infty$ induced norm bound $\gamma$. Figure 2 shows state trajectories when the initial conditions are given by

$$x(0) = \begin{bmatrix} 1.5 \\ -1 \end{bmatrix},$$

$$W(0) = \begin{bmatrix} 0.1 & -0.25 \\ 0.25 & -0.1 \end{bmatrix},$$

$$\bar{W}(0) = \begin{bmatrix} 0.2 & -0.1 \\ 0.1 & -0.2 \end{bmatrix}, \tag{22}$$

and the external disturbance $d_i(t)$ $(i = 1, 2)$ is given by a Gaussian noise with mean 0 and variance 1. Figure 2 shows, by the proposed $\mathscr{L}_2$–$\mathscr{L}_\infty$ learning law, that the state vector $x(t)$ is bounded on the interval where the external disturbance $d(t)$ exists. Evolution of the weights $W(t)$ and $\bar{W}(t)$ is illustrated in Figs. 3 and 4, respectively. Although the weights are not convergent, they are bounded. These results appear to be of little use for the storage of patterns, but they are very important for identification and control using neural networks.[27]

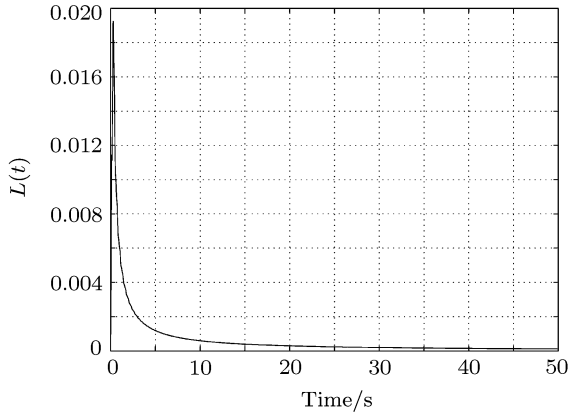



**Fig. 1.** The plot of $L(t)$ versus time.

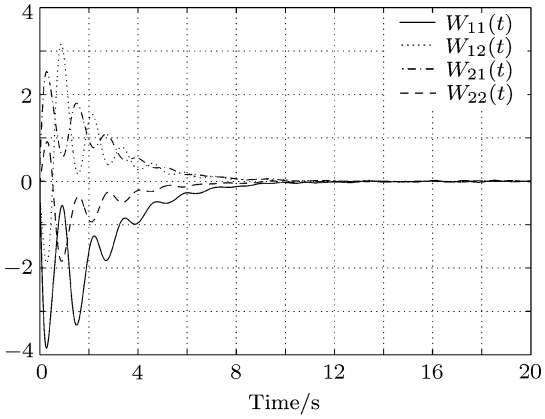



**Fig. 3.** The evolution of the weight $W(t)$.

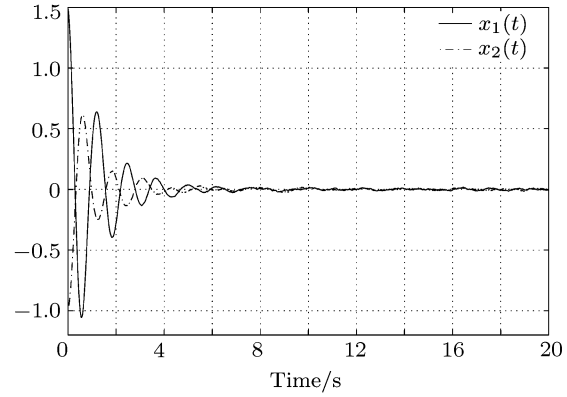



**Fig. 2.** State trajectories.

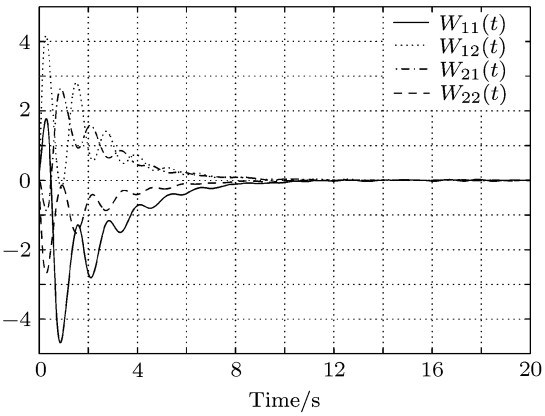



**Fig. 4.** The evolution of the weight $\bar{W}(t)$.

## 5. Conclusion

In this paper, we proposed a new $\mathscr{L}_2$–$\mathscr{L}_\infty$ learning law for dynamic neural networks with external disturbance. A gradient-learning law based on LMIs guarantees asymptotical stability and reduces the $\mathscr{L}_2$–$\mathscr{L}_\infty$ induced norm from the external disturbance to the state vector within a disturbance attenuation level. A simulation example is given to show the effectiveness of the proposed learning law. It is expected that the results proposed in this paper can be extended to dynamic neural networks with time-varying and distributed delays.

## References

[1] Gupta M M, Jin L and Homma N 2003 *Static and Dynamic Neural Networks* (New York: Wiley-Interscience)

[2] Liang X B and Wu L D 1998 *IEEE Trans. Circuits Syst.* I **45** 1010

[3] Sanchez E and Perez J P 1999 *IEEE Trans. Circuits Syst.* I **46** 1395

[4] Hu S and Wang J 2003 *IEEE Trans. Neural Networks* **14** 35

[5] Chu T, Zhang C and Zhang Z 2003 *Neural Networks* **16** 1223

[6] Chu T and Zhang C 2007 *Neural Networks* **20** 94

[7] Rovithakis G and Christodoulou M 1994 *IEEE Trans. Syst., Man, Cybern.* **24** 400

[8] Jagannathan S and Lewis F 1996 *Automatica* **32** 1707

[9] Suykens J A K, Vandewalle J and Moor B D 1997 *IEEE Trans. Signal Process.* **45** 2682

[10] Yu W and Li X 2001 *IEEE Trans. Circuits Syst.* I **48** 256

[11] Chairez I, Poznyak A and Poznyak T 2006 *IEEE Trans. Circuits Syst.* II **53** 1338

[12] Yu W and Li X 2007 *Neural Processing Letters* **25** 143

[13] Rubio and Yu W 2007 *Neurocomputing* **70** 2460

[14] Grigoriadis K and Watson J 1997 *IEEE Trans. Aerosp. Electron. Syst.* **33** 1326

[15] Watson J and Grigoriadis K 1998 *Systems and Control Letters* **35** 111
[16] Palhares R and Peres P 2000 *Automatica* **36** 851
[17] Gao H and Wang C 2003 *IEEE Trans. Trans. Circ. Syst.* I **50** 594
[18] Gao H and Wang C 2003 *IEEE Trans. Automat. Control* **48** 1661
[19] Mahmoud M 2007 *IET Control Theory and Applications* **1** 141
[20] Qiu J, Feng G and Yang J 2008 *IET Control Theory and Applications* **2** 795
[21] Zhou Y and Li J 2008 *IET Control Theory and Applications* **2** 773
[22] Boyd S, Ghaoui L E, Feron E and Balakrishinan V 1994 *Linear Matrix Inequalities in Systems and Control Theory* (Philadelphia: SIAM)
[23] Gahinet P, Nemirovski A, Laub A J and Chilali M 1995 *LMI Control Toolbox* (Natick: The Mathworks Inc.)
[24] Hopfield J 1984 *Proc. Nat. Acad. Sci.* **81** 3088
[25] Wilson D 1989 *IEEE Trans. Automat. Control* **34** 94
[26] Skelton R, Iwasaki T and Grigoriadis K 1997 *A Unified Algebraic Approach to Linear Control Design* (London: Taylor & Francis)
[27] Hunt K, Sbarbaro D, Zbikowski R and Gawthrop P 1992 *Automatica* **28** 1083