provide a "linearized" underdetermined BSS problem, which can easily be solved.

The presented method requires sparse sources and invertible nonlinearities that are linear for small input values. Simulation results were included for 2-measurement and 3-measurement cases, and as long as the contributions of the different sources do not overlap in the mixtures, there is no restriction on the number of sources or mixtures.

REFERENCES

[1] J. Cardoso, "Blind signal separation: statistical principles," in *Proc. IEEE. Special Issue on Blind Identification and Estimation*, vol. 9, Oct. 1998, pp. 2009–2025.

[2] P. Comon, C. Jutten, and J. Herault, "Blind separation of sources, part II: problem statement," *Signal Process.*, vol. 24, pp. 11–21, 1991.

[3] P. Comon, "Independent component analysis—a new concept?," *Signal Process.*, vol. 36, pp. 287–314, 1994.

[4] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*. New York: Wiley Interscience, 2001.

[5] M. Joho, H. Mathis, and R. Lambert, "Overdetermined blind source separation: using more sensors than source signals in a noisy mixture," in *Proc. 2nd Int. Conf. on Independent Component Analysis and Blind Signal Separation*, Helsinki, Finland, Jun. 2000, pp. 81–86.

[6] T. W. Lee, M. S. Lewicki, M. Girolami, and T. J. Sejnowski, "Blind source separation of more sources than mixtures using overcomplete representations," *IEEE Signal Process. Lett.*, vol. 6, pp. 87–90, 1999.

[7] P. Bofill and M. Zibulevsky, "Underdetermined blind source separation using sparse representations," *Signal Process.*, vol. 81, no. 11, pp. 2353–2362, 2001.

[8] D. Luengo, I. Santamaría, and L. Vielva, "A general solution to blind inverse problems for sparse input signals: deconvolution, equalization and source separation," *Neurocomputing* , vol. 69, pp. 198–215, 2005.

[9] M. Solazzi, R. Parisi, and A. Uncini, "Blind source separation in nonlinear mixtures by adaptive spline neural networks," in *Proc. 3rd Int. Conf. on Independent Component Analysis and Blind Signal Separation*, San Diego, CA, Dec. 2001, pp. 254–259.

[10] Y. Tan and J. Wang, "Nonlinear blind source separation using higher order statistics and a genetic algorithm," *IEEE Trans. Evol. Comput.*, vol. 5, no. 6, pp. 600–612, Dec. 2001.

[11] M. Babaie-Zadeh, C. Jutten, and K. Nayebi, "A geometric approach for separating post nonlinear mixtures," in *Proc. XI Europ. Signal Processing Conf.*, vol. II, Toulouse, France, Sep. 2002, pp. 11–14.

[12] A. Taleb and C. Jutten, "Source separation in post-nonlinear mixtures," *IEEE Trans. Signal Process.*, vol. 47, no. 10, pp. 2807–2820, Oct. 1999.

[13] F. J. Theis and S. Amari, "Postnonlinear overcomplete blind source separation using sparse sources," in *Proc. 5th Int. Conf. Independent Component Analysis and Blind Signal Separation*, vol. 3195, Lecture Notes in Computer Science, Granada, Spain, Sep. 2004, pp. 718–725.

[14] L. Vielva, I. Santamaría, C. Pantaleón, J. Ibáñez, D. Erdogmus, and J. C. Príncipe, "Estimation of the mixing matrix for underdetermined blind source separation using spectral techniques," in *Proc. XI Europ. Signal Processing Conf.*, vol. 1, Toulouse, France, Sep. 2002, pp. 557–560.

[15] D. Erdogmus, L. Vielva, and J. C. Príncipe, "Nonparametric estimation and tracking of the mixing matrix for underdetermined blind source separation," in *Proc. 3rd Int. Conf. on Independent Component Analysis and Blind Signal Separation*, San Diego, CA, Dec. 2001, pp. 189–194.

[16] F. J. Theis, E. W. Lang, and C. G. Puntonet, "A geometric algorithm for overcomplete linear ICA," *Neurocomputing*, vol. 56, pp. 381–396, 2004.

[17] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: analysis and an algorithm," in *Advances in Neural Information Processing Systems 14*, T. G. Dietterich, S. Becker, and Z. Ghahramani, Eds. Cambridge, MA: MIT Press, 2002, pp. 849–856.

[18] L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering," in *Advances in Neural Information Processing Systems 17*, L. K. Saul, Y. Weiss, and L. Bottou, Eds. Cambridge, MA: MIT Press, 2005, pp. 1601–1608.

Stability Analysis for Stochastic Cohen–Grossberg Neural Networks With Mixed Time Delays

Zidong Wang, Yurong Liu, Maozhen Li, and Xiaohui Liu

***Abstract*—In this letter, the global asymptotic stability analysis problem is considered for a class of stochastic Cohen–Grossberg neural networks with mixed time delays, which consist of both the discrete and distributed time delays. Based on an Lyapunov–Krasovskii functional and the stochastic stability analysis theory, a linear matrix inequality (LMI) approach is developed to derive several sufficient conditions guaranteeing the global asymptotic convergence of the equilibrium point in the mean square. It is shown that the addressed stochastic Cohen–Grossberg neural networks with mixed delays are globally asymptotically stable in the mean square if two LMIs are feasible, where the feasibility of LMIs can be readily checked by the Matlab LMI toolbox. It is also pointed out that the main results comprise some existing results as special cases. A numerical example is given to demonstrate the usefulness of the proposed global stability criteria.**

***Index Terms*—Cohen–Grossberg neural networks, discrete delays, distributed delays, global asymptotic stability, linear matrix inequality (LMI), Lyapunov–Krasovskii functional, stochastic systems.**

I. INTRODUCTION

The past few decades have witnessed tremendous developments in the research field of neural networks. Various neural networks, such as Hopfield neural networks, cellular neural networks, bidirectional associative neural networks and Cohen–Grossberg neural networks, have been widely investigated and successfully applied in many areas. Among them, the renowned Cohen–Grossberg neural network [7] has recently gained particular research attention, since it is quite general to include several well-known neural networks as its special cases, and it has promising application potentials for tasks of classification, associative memory, parallel computation and nonlinear optimization problems; see [16] and [26] for a survey.

On the other hand, time delays are unavoidably encountered in the implementation of neural networks, and may cause undesirable dynamic network behaviors such as oscillation and instability. For example, delay occurs due to the finite speeds of the switching and transmission of signals in a network. This leads to the delayed neural networks that were first explicitly introduced in [17]. Since then, the delayed neural networks have been widely studied. Recently, there has been an increasing research interest on the stability analysis problems for delayed Cohen–Grossberg neural networks, and many results have been reported in the literature. Various sufficient conditions, either delay-dependent or delay-independent, have been proposed to guarantee the asymptotic, exponential, or absolute stability for Cohen–Grossberg neural networks; see [2], [5], [6], [13], [14], and [18] for some recent results concerning *discrete* time-delays.

Manuscript received September 1, 2005; revised November 20, 2005. This work was supported in part by the Engineering and Physical Sciences Research Council (EPSRC) of the U.K. under Grant GR/S27658/01, by the Nuffield Foundation of the U.K. under Grant NAL/00630/G, and by the Alexander von Humboldt Foundation of Germany.

Z. Wang and X. Liu are with the Department of Information Systems and Computing, Brunel University, Uxbridge, Middlesex, UB8 3PH, U.K. (e-mail: Zidong.Wang@brunel.ac.uk).

Y. Liu is with the Department of Mathematics, Yangzhou University, Yangzhou 225002, P. R. China.

M. Li is with the Department of Electronic and Computer Engineering, Brunel University, Uxbridge, Middlesex, UB8 3PH, U.K.

Digital Object Identifier 10.1109/TNN.2006.872355

Although it has been recognized that discrete time-delays can be introduced into communication channels since they are ubiquitous in both the neural processing and signal transmission, a neural network also has a special nature due to the presence of an amount of parallel pathways with a variety of axon sizes and lengths. Such an inherent nature can be suitably modeled by *distributed* delays [21], because the signal propagation is distributed during a certain time period. For example, in [21], a neural circuit has been designed with distributed delays, which solves a general problem of recognizing patterns in a time-dependent signal. As a matter of fact, a realistic neural network should involve both discrete and distributed delays [19]. Recently, the stability analysis problems for Cohen–Grossberg neural networks with distributed time-delays have begun to receive some attention from some researchers; see, e.g., [20] and [23]. It should be mentioned that, most recently, the global asymptotic stability analysis problem has been investigated in [24], [25] for a general class of neural networks with *both* discrete *and* distributed time-delays, where a linear matrix inequality (LMI) approach has been developed to establish the sufficient stability conditions.

In the past few years, the dynamical behaviors of *stochastic* neural networks have emerged as a new subject of research mainly for two reasons: i) in real nervous systems, the synaptic transmission is a noisy process brought on by random fluctuations from the release of neurotransmitters and other probabilistic causes; and ii) it has been realized that a neural network could be stabilized or destabilized by certain stochastic inputs [3]. In particular, the stability criteria for stochastic neural networks becomes an attractive research problem of prime importance. Some initial results have just appeared, for example, in [12] and [22], for stochastic delayed Hopfield neural networks. However, to the best of the authors' knowledge, the global stability analysis problem for *stochastic Cohen–Grossberg* neural networks with *simultaneous* presence of discrete and distributed delays has not been studied yet, and still remains as a challenging open problem.

In this letter, we deal with the global asymptotic stability analysis problem for a class of stochastic Cohen–Grossberg neural networks with discrete and distributed time-delays. By utilizing a Lyapunov–Krasovskii functional and conducting the stochastic analysis, we recast the addressed stability analysis problem into a numerically solvability problem. Different from the commonly used matrix norm theories (such as the M-matrix method), a unified LMI approach is developed to establish sufficient conditions for the neural networks to be globally asymptotically stable. Note that LMIs can be easily solved by using the Matlab LMI toolbox, and no tuning of parameters is required [4]. A numerical example is provided to show the usefulness of the proposed global stability condition.

Notations: The notations are quite standard. Throughout this letter, $\mathbb{R}^n$ and $\mathbb{R}^{n\times m}$ denote, respectively, the n-dimensional Euclidean space and the set of all $n \times m$ real matrices. The superscript "T" denotes matrix transposition and the notation $X \geq Y$ (respectively, $X > Y$) where X and Y are symmetric matrices, means that $X - Y$ is positive semidefinite (respectively, positive definite). I_n is the $n \times n$ identity matrix. $|\cdot|$ is the Euclidean norm in $\mathbb{R}^n$. If A is a matrix, denote by $\|A\|$ its operator norm, i.e., $\|A\| = \sup\{|Ax| : |x| = 1\} = \sqrt{\lambda_{\max}(A^TA)}$ where $\lambda_{\max}(\cdot)$ (respectively, $\lambda_{\min}(\cdot)$) means the largest (respectively, smallest) eigenvalue of A. $l_2[0,\infty)$ is the space of square integrable vector. Moreover, let $(\Omega, \mathcal{F}, \{\mathcal{F}_t\}_{t\geq 0}, P)$ be a complete probability space with a filtration $\{\mathcal{F}_t\}_{t\geq 0}$ satisfying the usual conditions (i.e., the filtration contains all P-null sets and is right continuous). Denote by $L^p_{\mathcal{F}_0}([-h,0];\mathbb{R}^n)$ the family of all $\mathcal{F}_0$-measurable $C([-h,0];\mathbb{R}^n)$-valued random variables $\xi = \{\xi(\theta) : -h \leq \theta \leq 0\}$ such that $\sup_{-h\leq\theta\leq 0}\mathbb{E}|\xi(\theta)|^p < \infty$ where $\mathbb{E}\{\cdot\}$ stands for the mathematical expectation operator with respect to the given probability measure P. The shorthand $\mathrm{diag}\{M_1, M_2, \cdots, M_N\}$ denotes a block diagonal matrix with diagonal blocks being the matrices $M_1, M_2, \cdots, M_N$. Sometimes, the arguments of a function or a matrix will be omitted in the analysis when no confusion can arise.

II. Problem Formulation

In this letter, the Cohen–Grossberg neural networks with discrete and distributed time delays can be described by the following delay differential equations:

$$\frac{du_i(t)}{dt} = -a_i(u_i(t))\Bigg[b_i(u_i(t)) - \sum_{j=1}^{n} a_{ij}g_{1j}(u_j(t)) - \sum_{j=1}^{n} b_{ij}g_{2j}(u_j(t-h)) - \sum_{j=1}^{n} c_{ij}\int_{t-\tau}^{t} g_{3j}(u_j(s))\,ds + V_i\Bigg] \tag{1}$$

where $u_i(t)$ is the state of the ith unit at time t, $a_i(u_i(t))$ is the amplification function, $b_i(u_i(t))$ denotes the behaved function, and $g_{ki}(u_i(t))$ $(k = 1,2,3)$ are the activation functions. The matrices $A = (a_{ij})_{n\times n}$, $B = (b_{ij})_{n\times n}$ and $C = (c_{ij})_{n\times n}$ are, respectively, the connection weight matrix, the discretely delayed connection weight matrix, and the distributively delayed connection weight matrix. $V = [V_1, V_2, \cdots, V_n]^T$ is a constant external input vector. The scalar $h > 0$, which may be unknown, denotes the discrete time delay, whereas the scalar $\tau > 0$ is the known distributed time-delay.

Let $u(t) = (u_1(t), u_2(t), \cdots, u_n(t))^T$, $a(u) = \mathrm{diag}(a_1(u_1), a_2(u_2), \cdots, a_n(u_n))$, $b(u) = (b_1(u_1), b_2(u_2), \cdots, b_n(u_n))^T$, $g_i(u(\cdot)) = (g_{i1}(u_1(\cdot)), g_{i2}(u_2(\cdot)), \cdots, g_{\mathrm{in}}(u_n(\cdot)))^T$ (i = 1, 2, 3). The model (1) can be rewritten as the following compact matrix form:

$$\frac{du(t)}{dt} = -a(u(t))\Bigg[b(u(t)) - Ag_1(u(t)) - Bg_2(u(t-h)) - C\int_{t-\tau}^{t} g_3(u(s))\,ds + V\Bigg]. \tag{2}$$

In this letter, we make the following assumptions on the amplification function, the behaved function, and the neuron activation functions.

Assumption 1: For each $i \in \{1, 2, \cdots, n\}$, the amplification function $a_i(\cdot)$ is positive, bounded, and satisfies

$$0 < \underline{\alpha}_i \leq a_i(\cdot) \leq \bar{\alpha}_i \tag{3}$$

where $\underline{\alpha}_i$ and $\bar{\alpha}_i$ are known positive constants.

Assumption 2: The behaved function $b_i(x) : \mathbb{R} \to \mathbb{R}$ is continuous and differentiable, and

$$b_i'(x) \geq \gamma_i > 0 \qquad \forall x \in \mathbb{R}, \qquad i = 1, 2, \cdots, n. \tag{4}$$

Assumption 3: The neuron activation functions $g_i(\cdot)$ are bounded and satisfy the following Lipschitz conditions:

$$|g_i(x) - g_i(y)| \leq |G_i(x-y)| \qquad \forall x, y \in \mathbb{R}^n \tag{5}$$

where $G_i \in \mathbb{R}^{n\times n}$ (i = 1, 2, 3) are known constant matrices.

In the literature, the activation function is typically assumed to be continuous, differentiable, and monotonically increasing, such as the function of sigmoid type. These restrictions are no longer needed in this letter, and only Lipschitz condition and boundedness condition are imposed in Assumption 3. Note that the type of activation functions in (5) have already been used in numerous papers.

In Assumption 3, it is assumed that the activation functions are bounded, and it is well known that bounded activation functions always guarantee the existence of an equilibrium point for neural networks (2). For notational convenience, we shift the equilibrium point $u^* = (u_1^*, \cdots, u_n^*)^T$ to the origin by translation $x(t) = u(t) - u^*$, which yields the following system:

$$\frac{dx(t)}{dt} = -\alpha(x(t)) \left[\beta(x(t)) - Al_1(x(t)) - Bl_2(x(t-h)) - C \int_{t-\tau}^{t} l_3(u(s))\, ds \right] \tag{6}$$

where $x(t) = [x_1(t), x_2(t), \cdots, x_n(t)]^T \in \mathbb{R}^n$ is the state vector of the transformed system and

$$\begin{aligned} \alpha(x(t)) &= \mathrm{diag}(\alpha_1(x_1(t)), \alpha_2(x_2(t)), \cdots, \alpha_n(x_n(t))) \\ \alpha_i(x_i(t)) &= a_i(x_i(t) + u_i^*) \\ \beta(x(t)) &= (\beta_1(x_1(t)), \beta_2(x_2(t)), \cdots, \beta_n(x_n(t))) \\ \beta_i(x_i(t)) &= b_i(x_i(t) + u_i^*) - b_i(u_i^*) \\ l_i(x(\cdot)) &= (l_{i1}(x_1(\cdot)), l_{i2}(x_2(\cdot)), \cdots, l_{\mathrm{in}}(x_n(\cdot))) \\ l_{ij}(x_j(\cdot)) &= g_{ij}\left(x_j(\cdot) + u_j^*\right) - g_{ij}\left(u_j^*\right). \end{aligned}$$

It follows, respectively, from Assumption 1, Assumption 2, and Assumption 3 that

$$0 < \underline{\alpha}_i \le \alpha_i(\cdot) \le \bar{\alpha}_i, \qquad (i = 1, 2, \cdots, n) \tag{7}$$

$$x_i(t)\beta_i(x_i(t)) \ge \gamma_i x_i^2(t), \qquad (i = 1, 2, \cdots, n) \tag{8}$$

$$|l_i(x)| \le |G_i x|, \qquad (i = 1, 2, 3). \tag{9}$$

As discussed in Section I, in the real world, the neural network is often disturbed by environmental noises that affect the stability of the equilibrium. In this letter, as in [3], [12], and[22], Cohen–Grossberg neural network with stochastic perturbations is introduced as follows:

$$\begin{aligned} dx(t) = \Bigg\{ &-\alpha(x(t)) \\ &\times \left[\beta(x(t)) - Al_1(x(t)) - Bl_2(x(t-h)) - C \int_{t-\tau}^{t} l_3(u(s))\, ds \right] \Bigg\} dt \\ &+ \sigma(t, x(t), x(t-h))\, dw(t) \end{aligned} \tag{10}$$

where $w(t) = [w_1(t), w_2(t), \cdots, w_n(t)]^T \in \mathbb{R}^n$ is a Brownian motion defined on $(\Omega, \mathcal{F}, \{\mathcal{F}_t\}_{t\ge 0}, P)$.

Assume that $\sigma : \mathbb{R}^+ \times \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^{n\times n}$ ($\sigma(t, 0, 0) = 0$) is locally Lipschitz continuous and satisfies the linear growth condition ([11]). Moreover, σ satisfies

$$\mathrm{trace}\left[\sigma^T(t, x(t), x(t-h))\, \sigma(t, x(t), x(t-h))\right] \le |\Sigma_1 x(t)|^2 + |\Sigma_2 x(t-h)|^2 \tag{11}$$

where Σ_1 and Σ_2 are known constant matrices with appropriate dimensions. Let $x(t;\xi)$ denote the state trajectory of the neural network (10) from the initial data $x(\theta) = \xi(\theta)$ on $-h \le \theta \le 0$ in $L^2_{\mathcal{F}0}([-h, 0]; \mathbb{R}^n)$. It can be easily seen that the system (10) admits a trivial solution $x(t; 0) \equiv 0$ corresponding to the initial data $\xi = 0$, see [11].

Remark 1: The assumption (11) on the stochastic disturbance term, $\sigma(t, x(t), x(t-h))$, has been used in recent papers dealing with stochastic neural networks, see [12] and references therein.

Definition 1: For the neural network (10) and every $\xi \in L^2_{\mathcal{F}0}([-h, 0]; \mathbb{R}^n)$, the trivial solution (equilibrium point) is globally asymptotically stable in the mean square if the following holds:

$$\lim_{t\to\infty} \mathbb{E}\, |x(t;\xi)|^2 = 0. \tag{12}$$

The main purpose of the rest of this letter is to establish LMI-based stability criteria, which can then be readily checked by using the Matlab LMI toolbox, such that the global asymptotic stability is guaranteed for the neural network (10) with both discrete and distributed time delays.

III. Main Results and Proofs

The following lemmas will be frequently used in establishing our LMI-based stability criteria.

Lemma 1: Let $x \in \mathbb{R}^n$, $y \in \mathbb{R}^n$ and $\varepsilon > 0$. Then we have $x^T y + y^T x \le \varepsilon x^T x + \varepsilon^{-1} y^T y$.

Lemma 2: [10] For any positive definite matrix $M > 0$, scalar $\gamma > 0$, vector function $\omega : [0, \gamma] \to \mathbb{R}^n$ such that the integrations concerned are well defined, the following inequality holds:

$$\left(\int_0^\gamma \omega(s) ds\right)^T M \left(\int_0^\gamma \omega(s) ds\right) \le \gamma \left(\int_0^\gamma \omega^T(s) M \omega(s) ds\right).$$

Before stating our main results, let us denote

$$\begin{aligned} &\underline{\alpha} := \min_{1\le i\le n} \underline{\alpha}_i \quad \bar{\alpha} := \max_{1\le i\le n} \bar{\alpha}_i \\ &\Gamma := \mathrm{diag}\{\gamma_1, \cdots, \gamma_n\} \quad P := \mathrm{diag}\{p_1, \cdots, p_n\} \end{aligned} \tag{13}$$

$$\Omega_1 := -\underline{\alpha} P\Gamma - \underline{\alpha}\Gamma P + \rho\left(\Sigma_1^T \Sigma_1 + \Sigma_2^T \Sigma_2\right) \tag{14}$$

where $\underline{\alpha}_i$ and $\bar{\alpha}_i$ are defined in (3), γ_i is defined in (4), Σ_1 and Σ_2 are defined in (11), the diagonal positive definite matrix P and the positive scalar $\rho > 0$ are two parameters to be designed.

We are now ready to derive the conditions under which the network dynamics of (10) is globally asymptotically stable in the mean square. The main theorem given below shows that the stability criteria can be expressed in terms of the feasibility of two LMIs.

Theorem 1: If there exist positive scalars $\rho > 0$, $\varepsilon_i > 0$ ($i = 1$, 2, 3) and a diagonal positive definite matrix $P > 0$ such that the two LMIs

$$P < \rho I \tag{15}$$

and (16), shown at the bottom of the next page, hold where Ω_1 is defined in (14), then the dynamics of the neural network (10) is globally asymptotically stable in the mean square.

Proof: Pre- and postmultiplying (16) by the block-diagonal matrix

$$\mathrm{diag}\left\{I, \varepsilon_1^{-\frac{1}{2}} I, \varepsilon_1^{-\frac{1}{2}} I, \varepsilon_2^{-\frac{1}{2}} I, \varepsilon_2^{-\frac{1}{2}} I, \varepsilon_3^{-\frac{1}{2}} I, \varepsilon_3^{-\frac{1}{2}} I\right\}$$

yield (17), as shown at the bottom of the next page, or

$$\begin{bmatrix} \Omega_1 & \Omega_3^T \\ \Omega_3 & -\Omega_2 \end{bmatrix} < 0 \tag{18}$$

where

$$\Omega_1 := -\underline{\alpha}P\Gamma - \underline{\alpha}\Gamma P + \rho\left(\Sigma_1^T\Sigma_1 + \Sigma_2^T\Sigma_2\right)$$
$$\Omega_2 := I$$
$$\Omega_3 := \left[\varepsilon_1^{-\frac{1}{2}}\lambda_{\max}^{\frac{1}{2}}(AA^T)\bar{\alpha}P \quad \varepsilon_1^{\frac{1}{2}}G_1^T \quad \varepsilon_2^{-\frac{1}{2}}\lambda_{\max}^{\frac{1}{2}}(BB^T)\bar{\alpha}P \quad \varepsilon_2^{\frac{1}{2}}G_2^T \quad \varepsilon_3^{-\frac{1}{2}}\lambda_{\max}^{\frac{1}{2}}(CC^T)\bar{\alpha}P \quad \varepsilon_3^{\frac{1}{2}}\tau G_3^T\right]^T.$$

It follows from the Schur Complement Lemma (see [4]) that (18) holds if and only if

$$\Omega_1 + \Omega_3^T\Omega_2^{-1}\Omega_3 < 0$$

or

$$\begin{aligned}-\underline{\alpha}P\Gamma - \underline{\alpha}\Gamma P + \rho\left(\Sigma_1^T\Sigma_1 + \Sigma_2^T\Sigma_2\right) &+\left[\varepsilon_1^{-1}\lambda_{\max}(AA^T) + \varepsilon_2^{-1}\lambda_{\max}(BB^T) + \varepsilon_3^{-1}\lambda_{\max}(CC^T)\right]\\ &\times\bar{\alpha}^2P^2 + \varepsilon_1G_1^TG_1 + \varepsilon_2G_2^TG_2 + \varepsilon_3\tau^2G_3^TG_3 < 0. \qquad (19)\end{aligned}$$

Therefore, we know from the condition of Theorem 1 that, there exist positive scalars $\rho > 0$, $\varepsilon_i > 0$ (i =1, 2, 3) and a diagonal positive definite matrix $P > 0$ such that (19) is true. Also, we know from (9) that

$$l_i^T(x)l_i(x) \le |G_ix|^2 = x^TG_i^TG_ix. \qquad (20)$$

In order to prove the global asymptotic stability in the mean square of the network (10), we define a Lyapunov–Krasovskii functional candidate $V(t, x(t)) \in \mathcal{C}^{2,1}(\mathbb{R}^+ \times \mathbb{R}^n; \mathbb{R}^+)$ by

$$V(t, x(t)) = x^T(t)Px(t) + \int_{t-h}^{t} x^T(s)Q_1x(s)ds + \int_{-\tau}^{0}\int_{t+s}^{t} x^T(\eta)Q_2x(\eta)d\eta ds \qquad (21)$$

where P is the diagonal positive definite solution to (19), and $Q_1 \ge 0$ and $Q_2 \ge 0$ are defined by

$$Q_1 := \varepsilon_2G_2^TG_2 + \rho\Sigma_2^T\Sigma_2, \quad Q_2 := \varepsilon_3\tau G_3^TG_3. \qquad (22)$$

By Itô's differential formula (see, e.g., [8]), the stochastic derivative of $V(t, x(t))$ along (10) can be obtained as follows:

$$\begin{aligned}dV(t, x(t)) = \Bigg\{&-2x^T(t)P\alpha(x(t))\\ &\times\left[\beta(x(t)) - Al_1(x(t)) - Bl_2(x(t-h)) - C\int_{t-\tau}^{t} l_3(u(s))ds\right]\\ &+ \text{trace}\left[\sigma^T(t, x(t), x(t-h))P\sigma(t, x(t), x(t-h))\right]\\ &+ x^T(t)Q_1x(t) - x^T(t-h)Q_1x(t-h)\\ &+ \tau x^T(t)Q_2x(t) - \int_{t-\tau}^{t} x^T(s)Q_2x(s)ds\Bigg\}dt\\ &+ 2x^T(t)P\sigma(t, x(t), x(t-h))dw(t). \qquad (23)\end{aligned}$$

Noticing that P and $\alpha(x(t))$ are diagonal positive–definite matrices, we obtain from (7) and (8) that

$$\begin{aligned}&-2x^T(t)P\alpha(x(t))\beta(x(t))\\ &= -2\sum_{i=1}^{n} x_i(t)p_i\alpha_i(x_i(t))\beta_i(x_i(t))\\ &= -2\sum_{i=1}^{n} p_i\alpha_i(x_i(t))[x_i(t)\beta_i(x_i(t))]\\ &\le -2\sum_{i=1}^{n} p_i\alpha_i(x_i(t))\gamma_ix_i^2(t) \le -2\sum_{i=1}^{n} p_i\underline{\alpha}_i\gamma_ix_i^2(t)\\ &\le -2\underline{\alpha}\sum_{i=1}^{n} p_i\gamma_ix_i^2(t) = -2\underline{\alpha}x^T(t)P\Gamma x(t). \qquad (24)\end{aligned}$$

Next, it follows from the conditions (11) and (15) that

$$\begin{aligned}&\text{trace}\left[\sigma^T(t, x(t), x(t-h))P\sigma(t, x(t), x(t-h))\right] \qquad (25)\\ &\le \lambda_{\max}(P)\text{trace}\left[\sigma^T(t, x(t), x(t-h))\sigma(t, x(t), x(t-h))\right]\\ &\le \rho\left[x^T(t)\Sigma_1^T\Sigma_1x(t) + x^T(t-h)\Sigma_2^T\Sigma_2x(t-h)\right]. \qquad (26)\end{aligned}$$

$$\begin{bmatrix} \Omega_1 & \lambda_{\max}^{\frac{1}{2}}(AA^T)\bar{\alpha}P & \varepsilon_1G_1^T & \lambda_{\max}^{\frac{1}{2}}(BB^T)\bar{\alpha}P & \varepsilon_2G_2^T & \lambda_{\max}^{\frac{1}{2}}(CC^T)\bar{\alpha}P & \varepsilon_3\tau G_3^T\\ \lambda_{\max}^{\frac{1}{2}}(AA^T)\bar{\alpha}P & -\varepsilon_1I & 0 & 0 & 0 & 0 & 0\\ \varepsilon_1G_1 & 0 & -\varepsilon_1I & 0 & 0 & 0 & 0\\ \lambda_{\max}^{\frac{1}{2}}(BB^T)\bar{\alpha}P & 0 & 0 & -\varepsilon_2I & 0 & 0 & 0\\ \varepsilon_2G_2 & 0 & 0 & 0 & -\varepsilon_2I & 0 & 0\\ \lambda_{\max}^{\frac{1}{2}}(CC^T)\bar{\alpha}P & 0 & 0 & 0 & 0 & -\varepsilon_3I & 0\\ \varepsilon_3\tau G_3 & 0 & 0 & 0 & 0 & 0 & -\varepsilon_3I \end{bmatrix} < 0 \qquad (16)$$

$$\begin{bmatrix} \Omega_1 & \varepsilon_1^{-\frac{1}{2}}\lambda_{\max}^{\frac{1}{2}}(AA^T)\bar{\alpha}P & \varepsilon_1^{\frac{1}{2}}G_1^T & \varepsilon_2^{-\frac{1}{2}}\lambda_{\max}^{\frac{1}{2}}(BB^T)\bar{\alpha}P & \varepsilon_2^{\frac{1}{2}}G_2^T & \varepsilon_3^{-\frac{1}{2}}\lambda_{\max}^{\frac{1}{2}}(CC^T)\bar{\alpha}P & \varepsilon_3^{\frac{1}{2}}\tau G_3^T\\ \varepsilon_1^{-\frac{1}{2}}\lambda_{\max}^{\frac{1}{2}}(AA^T)\bar{\alpha}P & -I & 0 & 0 & 0 & 0 & 0\\ \varepsilon_1^{\frac{1}{2}}G_1 & 0 & -I & 0 & 0 & 0 & 0\\ \varepsilon_2^{-\frac{1}{2}}\lambda_{\max}^{\frac{1}{2}}(BB^T)\bar{\alpha}P & 0 & 0 & -I & 0 & 0 & 0\\ \varepsilon_2^{\frac{1}{2}}G_2 & 0 & 0 & 0 & -I & 0 & 0\\ \varepsilon_3^{-\frac{1}{2}}\lambda_{\max}^{\frac{1}{2}}(CC^T)\bar{\alpha}P & 0 & 0 & 0 & 0 & -I & 0\\ \varepsilon_3^{\frac{1}{2}}\tau G_3 & 0 & 0 & 0 & 0 & 0 & -I \end{bmatrix} < 0 \qquad (17)$$

For the positive scalars $\varepsilon_1 > 0$, $\varepsilon_2 > 0$, $\varepsilon_3 > 0$, it follows from Lemma 1 and (20) that

$$\begin{aligned}
&2x^T(t)P\alpha(x(t))Al_1(x(t))\\
&\quad\le \varepsilon_1^{-1}x^T(t)P\alpha(x(t))AA^T\alpha(x(t))Px(t)\\
&\qquad+\varepsilon_1 l_1^T(x(t))l_1(x(t))\\
&\quad\le \varepsilon_1^{-1}\lambda_{\max}(AA^T)\bar{\alpha}^2x^T(t)P^2x(t)\\
&\qquad+\varepsilon_1 x^T(t)G_1^TG_1x(t) \qquad (27)\\
&2x^T(t)P\alpha(x(t))Bl_2(x(t-h))\\
&\quad\le \varepsilon_2^{-1}x^T(t)P\alpha(x(t))BB^T\alpha(x(t))Px(t)\\
&\qquad+\varepsilon_2 l_2^T(x(t-h))l_2(x(t-h))\\
&\quad\le \varepsilon_2^{-1}\lambda_{\max}(BB^T)\bar{\alpha}^2x^T(t)P^2x(t)\\
&\qquad+\varepsilon_2 x^T(t-h)G_2^TG_2x(t-h). \qquad (28)
\end{aligned}$$

Furthermore, it can be seen from (22) and Lemma 2 that

$$\begin{aligned}
&\varepsilon_3\left(\int_{t-\tau}^{t} l_2(x(s))ds\right)^T\int_{t-\tau}^{t} l_2(x(s))ds\\
&\quad\le \varepsilon_3\tau\int_{t-\tau}^{t} l_2^T(x(s))l_2(x(s))ds\\
&\quad\le \varepsilon_3\tau\int_{t-\tau}^{t} x^T(s)G_3^TG_3x(s)ds\\
&\quad= \int_{t-\tau}^{t} x^T(s)Q_2x(s)ds \qquad (29)
\end{aligned}$$

and, hence

$$\begin{aligned}
&2x^T(t)P\alpha(x(t))C\int_{t-\tau}^{t} l_3(u(s))ds\\
&\quad\le \varepsilon_3^{-1}x^T(t)P\alpha(x(t))CC^T\alpha(x(t))Px(t)\\
&\qquad+\varepsilon_3\left(\int_{t-\tau}^{t} l_3(u(s))ds\right)^T\left(\int_{t-\tau}^{t} l_3(u(s))ds\right)\\
&\quad\le \varepsilon_3^{-1}\lambda_{\max}(CC^T)\bar{\alpha}^2x^T(t)P^2x(t)\\
&\qquad+\int_{t-\tau}^{t} x^T(s)Q_2x(s)ds. \qquad (30)
\end{aligned}$$

Using (22) and (24)–(30), we obtain from (23) that

$$\begin{aligned}
&dV(t,x(t))\\
&\le \Big\{x^T(t)\Big[-\underline{\alpha}P\Gamma-\underline{\alpha}\Gamma P+\rho\left(\Sigma_1^T\Sigma_1+\Sigma_2^T\Sigma_2\right)\\
&\qquad+\Big[\varepsilon_1^{-1}\lambda_{\max}(AA^T)+\varepsilon_2^{-1}\lambda_{\max}(BB^T)\\
&\qquad\quad+\varepsilon_3^{-1}\lambda_{\max}(CC^T)\Big]\bar{\alpha}^2P^2\\
&\qquad+\varepsilon_1G_1^TG_1+\varepsilon_2G_2^TG_2+\varepsilon_3\tau^2G_3^TG_3\Big]x(t)\Big\}dt\\
&\quad+2x^T(t)P\sigma(t,x(t),x(t-h))dw(t)\\
&=x^T(t)\Pi x(t)dt+2x^T(t)P\sigma\\
&\quad\times(t,x(t),x(t-h))dw(t). \qquad (31)
\end{aligned}$$

where Π is defined as

$$\begin{aligned}
\Pi := &-\underline{\alpha}P\Gamma-\underline{\alpha}\Gamma P+\rho\left(\Sigma_1^T\Sigma_1+\Sigma_2^T\Sigma_2\right)\\
&+\Big[\varepsilon_1^{-1}\lambda_{\max}(AA^T)+\varepsilon_2^{-1}\lambda_{\max}(BB^T)\\
&\quad+\varepsilon_3^{-1}\lambda_{\max}(CC^T)\Big]\bar{\alpha}^2P^2\\
&+\varepsilon_1G_1^TG_1+\varepsilon_2G_2^TG_2+\varepsilon_3\tau^2G_3^TG_3. \qquad (32)
\end{aligned}$$

From (19), we know that $\Pi < 0$. Taking the mathematical expectation of both sides of (31), we have

$$\frac{d\mathbb{E}V(t,x(t))}{dt} \le \mathbb{E}\left(x^T(t)\Pi x(t)\right) \le -\lambda_{\min}(-\Pi)\mathbb{E}|x(t)|^2. \quad (33)$$

It can now be concluded from Lyapunov stability theory that the dynamics of the neural network (10) is robustly, globally, asymptotically stable in the mean square. This completes the proof of Theorem 1. ■

Remark 2: By employing the Matlab LMI toolbox, it would be very convenient to verify the feasibility of (15) and (16) without tuning any parameters, and determine the global asymptotic stability of the neural network (10) directly. Compared with the existing results relying on matrix norm computation, such as those given in [3] and [6], the LMI approach developed in this letter is numerically more efficient [4] and less conservative. It is worth pointing out that, following the similar line of [15], it is not difficult to prove the exponential stability (in the mean square) of the neural network (10) under same conditions in Theorem 1.

In what follows, we will show that our results can be specialized to several cases including those have been studied extensively in the literature. All the corollaries given below are easy consequences of Theorem 1, hence the proofs are omitted.

We first consider the following Cohen–Grossberg neural network *without* stochastic perturbations:

$$\frac{dx(t)}{dt} = -\alpha(x(t))\left[\beta(x(t))-Al_1(x(t))-Bl_2(x(t-h))\right.$$
$$\left.-C\int_{t-\tau}^{t} l_3(u(s))ds\right]. \quad (34)$$

Corollary 1: If there exist positive scalars $\varepsilon_i > 0$ $(i = 1, 2, 3)$ and a diagonal positive definite matrix $P > 0$ such that the following LMI, as shown in (35) at the top of the next page, holds, then the dynamics of the neural network (34) is globally asymptotically stable.

Remark 3: Although there have been some papers published on the stability analysis problems for Cohen–Grossberg neural networks with discrete or distributed time-delays [2], [18], [20], [23] to the best of the authors' knowledge, there are few results concerning the simultaneous presence of discrete or distributed time-delays. Hence, the results in Corollary are still new.

If we are only interested in discrete-time delays, the Cohen–Grossberg neural network (34) can be further reduced to

$$\frac{dx(t)}{dt} = -\alpha(x(t))[\beta(x(t))-Al_1(x(t))-Bl_2(x(t-h))]. \quad (36)$$

Corollary 2: If there exist positive scalars $\varepsilon_i > 0$ $(i = 1, 2)$ and a diagonal positive definite matrix $P > 0$ such that the following LMI, as shown in (37) at the top of the next page, holds, then the dynamics of the neural network (36) is globally asymptotically stable.

Remark 4: The Cohen–Grossberg neural networks (36) with discrete time-delays have been well investigated in the literature, see, e.g., [2] and [18]. The result in Corollary provides alternative criteria based on an LMI approach, which is numerically traceable.

$$\begin{bmatrix} -\underline{\alpha}P\Gamma - \underline{\alpha}\Gamma P & \lambda_{\max}^{\frac{1}{2}}(AA^T)\bar{\alpha}P & \varepsilon_1 G_1^T & \lambda_{\max}^{\frac{1}{2}}(BB^T)\bar{\alpha}P & \varepsilon_2 G_2^T & \lambda_{\max}^{\frac{1}{2}}(CC^T)\bar{\alpha}P & \varepsilon_3 \tau G_3^T \\ \lambda_{\max}^{\frac{1}{2}}(AA^T)\bar{\alpha}P & -\varepsilon_1 I & 0 & 0 & 0 & 0 & 0 \\ \varepsilon_1 G_1 & 0 & -\varepsilon_1 I & 0 & 0 & 0 & 0 \\ \lambda_{\max}^{\frac{1}{2}}(BB^T)\bar{\alpha}P & 0 & 0 & -\varepsilon_2 I & 0 & 0 & 0 \\ \varepsilon_2 G_2 & 0 & 0 & 0 & -\varepsilon_2 I & 0 & 0 \\ \lambda_{\max}^{\frac{1}{2}}(CC^T)\bar{\alpha}P & 0 & 0 & 0 & 0 & -\varepsilon_3 I & 0 \\ \varepsilon_3 \tau G_3 & 0 & 0 & 0 & 0 & 0 & -\varepsilon_3 I \end{bmatrix} < 0 \tag{35}$$

$$\begin{bmatrix} -\underline{\alpha}P\Gamma - \underline{\alpha}\Gamma P & \lambda_{\max}^{\frac{1}{2}}(AA^T)\bar{\alpha}P & \varepsilon_1 G_1^T & \lambda_{\max}^{\frac{1}{2}}(BB^T)\bar{\alpha}P & \varepsilon_2 G_2^T \\ \lambda_{\max}^{\frac{1}{2}}(AA^T)\bar{\alpha}P & -\varepsilon_1 I & 0 & 0 & 0 \\ \varepsilon_1 G_1 & 0 & -\varepsilon_1 I & 0 & 0 \\ \lambda_{\max}^{\frac{1}{2}}(BB^T)\bar{\alpha}P & 0 & 0 & -\varepsilon_2 I & 0 \\ \varepsilon_2 G_2 & 0 & 0 & 0 & -\varepsilon_2 I \end{bmatrix} < 0 \tag{37}$$

$$\begin{bmatrix} -\underline{\alpha}P\Gamma - \underline{\alpha}\Gamma P & \lambda_{\max}^{\frac{1}{2}}(AA^T)\bar{\alpha}P & \varepsilon_1 G_1^T & \lambda_{\max}^{\frac{1}{2}}(CC^T)\bar{\alpha}P & \varepsilon_3 \tau G_3^T \\ \lambda_{\max}^{\frac{1}{2}}(AA^T)\bar{\alpha}P & -\varepsilon_1 I & 0 & 0 & 0 \\ \varepsilon_1 G_1 & 0 & -\varepsilon_1 I & 0 & 0 \\ \lambda_{\max}^{\frac{1}{2}}(CC^T)\bar{\alpha}P & 0 & 0 & -\varepsilon_3 I & 0 \\ \varepsilon_3 \tau G_3 & 0 & 0 & 0 & -\varepsilon_3 I \end{bmatrix} < 0 \tag{39}$$

If there appears only distributed time-delay in the Cohen–Grossberg neural network (34), as in [20] and [23] the model can now be simplified to

$$\frac{dx(t)}{dt} = -\alpha(x(t)) \left[\beta(x(t)) - Al_1(x(t)) - C \int_{t-\tau}^{t} l_3(u(s))\, ds \right]. \tag{38}$$

Corollary 3: If there exist positive scalars $\varepsilon_i > 0$ ($i = 1, 3$) and a diagonal positive definite matrix $P > 0$ such that the following LMI, as shown in (39) at the top of the page, holds, then the dynamics of the neural network (38) is globally asymptotically stable.

Remark 5: In [20] and [23], the stability criteria of the Cohen–Grossberg neural networks (38) with distributed time-delays have been established in terms of some nonlinear inequalities, which involve the tuning of some scalar parameters. However, there lacks a systematic tuning law. Also, in [20] and [23], in order to verify the stability of the neural network, we have to compute the norms of the parameters A and C separately, while in Corollary 3, we just need to check the feasibility of one integrated LMI, which can be done more conveniently by the Matlab LMI toolbox.

IV. Numerical Example

Let us consider a third-order delayed stochastic Cohen–Grossberg neural network (10) with both discrete and distributed delays. The network data are given as follows:

$$\Gamma = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 2 \end{bmatrix} \quad A = \begin{bmatrix} 0.3 & -1.8 & 0.5 \\ -1.1 & 1.6 & 1.1 \\ 0.6 & 0.4 & -0.3 \end{bmatrix}$$
$$B = \begin{bmatrix} 0.8 & 0.2 & 0.1 \\ 0.2 & 0.6 & 0.6 \\ -0.8 & 1.1 & -1.2 \end{bmatrix} \quad C = \begin{bmatrix} 0.5 & 0.2 & 0.1 \\ 0.3 & 0.7 & -0.3 \\ 1.2 & -1.1 & -0.5 \end{bmatrix}$$
$$G_1 = G_2 = G_3 = 0.2I_3 \quad \Sigma_1 = \Sigma_2 = 0.08I_3$$
$$\underline{\alpha} = 0.7, \quad \bar{\alpha} = 0.8, \quad \tau = 0.5, \quad h = 0.12.$$

By solving the LMIs (15), (16) for $\rho > 0$, $\varepsilon_i > 0$ ($i = 1, 2, 3$), and $P > 0$, we obtain

$$\rho = 1.7128 \quad \varepsilon_1 = 1.5038 \quad \varepsilon_2 = 1.5038$$
$$\varepsilon_3 = 1.5422 \quad P = \mathrm{diag}\{0.2289, 0.2289, 0.2291\}$$

which implies from Theorem 1 that the delayed stochastic Cohen–Grossberg neural network (10) is globally asymptotically stable in the mean square.

V. Conclusion

In this letter, we have dealt with the problem of global asymptotic stability analysis for a class of stochastic Cohen–Grossberg neural networks, which involve both discrete and distributed time delays. We have removed the traditional monotonicity and smoothness assumptions on the activation function. A LMI approach has been developed to solve the problem addressed. The stability criteria have been derived in terms of the positive definite solution to two LMIs involving several scalar parameters, which can be easily solved by using the Matlab toolbox. A simple example has been used to demonstrate the usefulness of the main results.

References

[1] S. Arik, "Global robust stability analysis of neural networks with discrete time delays," *Chaos Solitons Fractals*, vol. 26, no. 5, pp. 1407–1414, 2005.

[2] S. Arik and Z. Orman, "Global stability analysis of Cohen–Grossberg neural networks with time varying delays," *Phys. Lett. A*, vol. 341, no. 5–6, pp. 410–421, 2005.

[3] S. Blythe, X. Mao, and X. Liao, "Stability of stochastic delay neural networks," *J. Franklin Inst.*, vol. 338, pp. 481–495, 2001.

[4] S. Boyd, L. EI Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities In System and Control Theory*. Philadelphia, PA: SIAM, 1994.

[5] J. Cao and J. Liang, "Boundedness and stability for Cohen–Grossberg neural network with time-varying delays," *J. Math. Anal. Appl.*, vol. 296, no. 2, pp. 665–685, 2004.
[6] T. Chen and L. Rong, "Delay-independent stability analysis of Cohen–Grossberg neural networks," *Phys. Lett. A*, vol. 317, no. 5–6, pp. 436–449, 2003.
[7] M. A. Cohen and S. Grossberg, "Absolute stability and global pattern formation and parallel memory storage by competitive neural networks," *IEEE Trans. Syst., Man, Cybern.*, vol. 13, no. 5, pp. 815–821, 1983.
[8] A. Friedman, *Stochastic Differential Equations and Their Applications*. New York: Academic, 1976.
[9] P. Gahinet, A. Nemirovsky, A. J. Laub, and M. Chilali, *LMI Control Toolbox: For Use With Matlab*. Natick, MA: The Math Works, Inc., 1995.
[10] F. Gu, "An integral inequality in the stability problem of time-delay systems," in *Proc. IEEE 39th Conf. Decision Contr.*, Sydney, Australia, Dec. 2000, pp. 2805–2810.
[11] J. K. Hale, *Theory of Functional Differential Equations*. New York: Springer-Verlag, 1977.
[12] H. Huang, D. W. C. Ho, and J. Lam, "Stochastic stability analysis of fuzzy Hopfield neural networks with time-varying delays," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 52, no. 5, pp. 251–255, May 2005.
[13] C.-C. Hwang, C.-J. Cheng, and T.-L. Liao, "Globally exponential stability of generalized Cohen–Grossberg neural networks with delays," *Phys. Lett. A*, vol. 319, no. 1–2, pp. 157–166, 2003.
[14] Y. Li, "Existence and stability of periodic solutions for Cohen–Grossberg neural networks with multiple delays," *Chaos Sol. Fract.*, vol. 20, no. 3, pp. 459–466, 2004.
[15] Y. Liu, Z. Wang, and X. Liu, "Global exponential stability of generalized recurrent neural networks with discrete and distributed delays," *Neural Netw.*, 2006, to be published.
[16] W. Lu and T. Chen, "Dynamical behaviors of Cohen–Grossberg neural networks with discontinuous activation functions," *Neural Netw.*, vol. 18, no. 3, pp. 231–242, 2005.
[17] C. M. Marcus and R. M. Westervelt, "Stability of analog neural networks with delay," *Phys. Rev. A*, vol. 39, pp. 347–359, 1989.
[18] L. Rong, "LMI-based criteria for robust stability of Cohen–Grossberg neural networks with delay," *Phys. Lett. A*, vol. 339, no. 1–2, pp. 63–73, 2005.
[19] S. Ruan and R. S. Filfil, "Dynamics of a two-neuron system with discrete and distributed delays," *Physica D*, vol. 191, pp. 323–342, 2004.
[20] J. Sun and L. Wan, "Global exponential stability and periodic solutions of Cohen–Grossberg neural networks with continuously distributed delays," *Physica D: Nonlin. Phenom.*, vol. 208, no. 1–2, pp. 1–20, 2005.
[21] D. W. Tank and J. J. Hopfield, "Neural computation by concentrating information in time," *Proc. Nat. Acad. Sci.*, vol. 84, pp. 1896–1991, 1987.
[22] L. Wan and J. Sun, "Mean square exponential stability of stochastic delayed Hopfield neural networks," *Phys. Lett. A*, vol. 343, no. 4, pp. 306–318, 2005.
[23] L. Wang, "Stability of Cohen–Grossberg neural networks with distributed delays," *Appl. Math. Comput.*, vol. 160, no. 1, pp. 93–110.
[24] Z. Wang, Y. Liu, and X. Liu, "On global asymptotic stability of neural networks with discrete and distributed delays," *Phys. Lett. A*, vol. 345, no. 4–6, pp. 299–308, 2005.
[25] Z. Wang, H. Shu, Y. Liu, D. W. C. Ho, and X. Liu, "Robust stability analysis of generalized neural networks with discrete and distributed time delays," *Chaos Solitons Fractals*, 2006, to be published.
[26] W. Xiong and J. Cao, "Global exponential stability of discrete-time Cohen–Grossberg neural networks," *Neurocomputing*, vol. 64, pp. 433–446, 2005.

A Tighter Bound for the Echo State Property

Michael Buehner and Peter Young

***Abstract*—This letter provides a brief explanation of echo state networks (ESNs) and provides a rigorous bound for guaranteeing asymptotic stability of these networks. The stability bounds presented here could aid in the design of echo state networks that would be applicable to control applications where stability is required.**

***Index Terms*—Echo state networks (ESNs), Lyapunov stability, nonlinear systems, recurrent neural networks (RNN), robust controls, weighted operator norms.**

I. Introduction

Artificial neural networks may be used in control system applications such as modeling nonlinear system dynamics and control of nonlinear systems. Two types of networks are used for these applications, namely feed-forward neural networks (FFNN) and recurrent neural networks (RNN). FFNN are attractive since they are easy to train in a stable manner (e.g., using back propagation), but are limited in the sense that they are only capable of providing a static map between inputs and outputs (i.e., they have no way of internally representing the dynamics of a nonlinear system). In contrast, RNN may be very difficult (and take a long time) to train stably; however, the recurrent connections of a RNN form a dynamical system. It is this dynamical nature of RNN that allows them to capture the dynamics of a nonlinear system, which makes them more applicable to nonlinear system modeling and control. Examples of RNN are Hopfield networks, Elman networks, liquid state machines, and echo state networks (ESNs), the last-mentioned being the focus of this letter. For a review of these RNN and the problems associated with training them, see either [1], [2], or [3].

The recent development of echo state networks [4] ESNs provides a class of RNN that alleviates the problem of training, but the design methodology of ESNs is still not fully understood. ESNs are characterized by their ability to uniquely map a temporal input history to an "echo state." An ESN that has this characteristic is said to have the *echo state property*. Currently, the echo state property may be verified from two sufficient conditions, namely one for the existence of echo states for all inputs and one for the nonexistence of echo states for certain inputs. This letter reformulates and further develops these conditions into separate necessary and sufficient condition for the existence of echo states for all inputs. As mentioned in [4], the current sufficient condition for the existence of echo states appears, in practice, to be rather restrictive. This problem is addressed by deriving a new sufficient condition that is less conservative. Specifically, a result that is well known in the *Robust Controls* community is used to reduce the conservatism and in some cases make the bounds tight (i.e., provide a single bound that is both necessary and sufficient). This letter concludes with some simple simulations to demonstrate the improvement that can be achieved by using the new sufficient condition.

II. Notation

Let $\mathbb{F} = \mathbb{C}$ or $\mathbb{R}$ be either the field of complex or real numbers, respectively. For any square matrix $W \in \mathbb{F}^{n\times n}$, let $\overline{\sigma}(W)$ denote

Manuscript received August 2, 2005; revised November 6, 2005. This work was supported by the National Science Foundation under Award 0 245 291.

The authors are with the Electrical and Computer Engineering Department, Colorado State University (CSU), Fort Collins, CO 80523 USA (e-mail: mbuehner@engr.colostate.edu; pmy@engr.colostate.edu).

Digital Object Identifier 10.1109/TNN.2006.872357