[using (1)] and $\xi(n) = 0.0193$ [using (7)] were in good accordance wth the empirically measured value $\xi^{(5)}(n) = 0.0157$. Again, note that the high number of available patterns privileges the GF-based approximation.

V. Conclusion

This letter has extended Vapnik's theory by exploiting basic properties of KWMs. The validity of the involved research consists in a general procedure allowing one to estimate a classifier's VC-dim in the multiclass case: one need only repeat the procedure exemplified in Section IV-B for the proper value of N_c and then apply expression (1). An additional, peculiar result of the presented approach lies in the opportunity given by (7) to estimate a classifier's GF.

References

[1] V. N. Vapnik, *Estimation of Dependences Based on Empirical Data*. New York: Springer-Verlag, 1982.

[2] ——, *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 1995.

[3] ——, *Statistical Learning Theory*. New York: Wiley, 1998.

[4] V. N. Vapnik, E. Levin, and Y. LeCun, "Measuring the VC-dimension of a learning machine," *Neural Computa.*, vol. 6, pp. 851–876, 1994.

[5] X. Shao, V. Cherkassky, and W. Li, "Measuring the VC-dimension using optimized experimental design," *Neural Comput.*, vol. 12, pp. 1969–1986, 2000.

[6] S. Ridella, S. Rovetta, and R. Zunino, "K-winner machines for pattern classification," *IEEE Trans. Neural Networks*, vol. 12, pp. 371–385, 2001.

[7] T. Kohonen, *Self-Organization and Associative Memory*, 3rd ed. New York: Springer-Verlag, 1989.

[8] A. Ben-Hur, D. Horn, H. T. Siegelmann, and V. Vapnik. A support vector method for clustering. presented at Adv. Neural Inf. Proc. NIPS ∗ 2000. [Online]. Available: http://www.cs.cmu.edu/Web/Groups/NIPS/NIPS2000/00abstracts.html

The Relevance Vector Machine Technique for Channel Equalization Application

S. Chen, S. R. Gunn, and C. J. Harris

***Abstract*—The recently introduced relevance vector machine (RVM) technique is applied to communication channel equalization. It is demonstrated that the RVM equalizer can closely match the optimal performance of the Bayesian equalizer, with a much sparser kernel representation than that is achievable by the state-of-art support vector machine (SVM) technique.**

***Index Terms*—Bayesian classification, equalization, relevance vector machines (RVMs), support vector machines (SVMs).**

I. Introduction

The state-of-art support vector machine (SVM) technique [1] has been gaining popularity in regression and classification, due to its many attractive features and promising empirical performance. It is generally believed that the formulation of SVM embodies the structural risk minimization principle, thus combining excellent generalization properties with a sparse kernel representation. The communication channel equalization is concerned with reliably detecting transmitted data symbols in the presence of distorting channel and noise. Since data symbols are drawn from a finite alphabet set, channel equalization can be viewed as a classification problem [2] and the SVM technique has found its application in equalization problems [3]–[6]. These works have shown that the SVM equalizer can closely match the optimal performance of the Bayesian equalizer. However, the results have also revealed that, for equalization application, the SVM technique does not result in a sufficiently sparse model. Typically, an SVM equalizer will have five to ten times more kernels than the number of the noise-free channel states that is required by the Bayesian equalizer. This "weakness" is due to the nature of the SVM method, as each misclassified training data is necessitated as a support vector (SV).

Modern communication systems have high data rates, and time for an equalizer to make a decision regarding a data symbol is extremely small. Thus decision complexity is a critical factor to consider. The fact that the SVM requires a relatively large number of kernels to approximate the optimal solution will limit its practical application. Recently, Tipping [7] introduced a relevance vector machine (RVM) method, which is based on a Bayesian framework [8], [9] and has an identical functional form to the SVM. The results given in [7] have demonstrated that the RVM has a comparable generalization performance to the SVM but requires dramatically fewer kernel functions than the SVM. In this correspondence, we apply the RVM method to channel equalization. Our results confirm that the RVM equalizer can closely match the Bayesian equalizer performance, and it has a much sparser kernel representation than the SVM equalizer. An RVM equalizer typically has fewer kernel functions than the noise-free channel states. A drawback of the RVM method is that it involves a highly nonlinear iterative optimization process. As in the case of the SVM, however, we only use the RVM method to design an equalizer "off-line" based on a block of training data. This design complexity will not pose too serious a problem.

II. Channel Equalization

Consider the channel which generates the received signal samples of

$$r_k = \sum_{i=0}^{n_c-1} c_i s_{k-i} + n_k \tag{1}$$

where

c_i	channel impulse response (CIR) taps;
n_c	CIR length;
Gaussian white noise n_k	has zero mean and variance σ_n^2;
transmitted symbol s_k	takes the value from the symbol set $\{\pm 1\}$.

An equalizer uses the information contained in the observation vector $\mathbf{r}_k = [r_k r_{k-1} \ldots r_{k-m+1}]^T$ to produce an estimate $\hat{s}_{k-d}$ of s_{k-d}, where m is called the equalizer order and $0 \leq d \leq m + n_c - 2$ the equalizer delay. The received signal vector $\mathbf{r}_k$ can be expressed as

$$\mathbf{r}_k = \bar{\mathbf{r}}_k + \mathbf{n}_k = \mathbf{C}\mathbf{s}_k + \mathbf{n}_k \tag{2}$$

where $\mathbf{C}$ is an $m \times (m + n_c - 1)$ Toeplitz CIR matrix, whose (i, j)th element is c_{j-i} for $0 \leq j - i \leq n_c - 1$ and zero otherwise. Notice that $\mathbf{s}_k$ has $N_s = 2^{m+n_c-1}$ possible combinations, denoted as $\tilde{\mathbf{s}}_j$, $1 \leq j \leq N_s$. Thus, $\bar{\mathbf{r}}_k$ takes values from the channel state set:

Manuscript received April 11, 2001; revised June 27, 2001.

The authors are with Department of Electronics and Computer Science, University of Southampton, Highfield, Southampton SO17 1BJ, U.K. (e-mail: sqc@ecs.soton.ac.uk).

Publisher Item Identifier S 1045-9227(01)08902-0.

$\mathcal{R} \triangleq \{\tilde{\mathbf{r}}_j = \mathbf{C}\tilde{\mathbf{s}}_j, 1 \le j \le N_s\}$. This set can be divided into two subsets conditioned on the value of s_{k-d}: $\mathcal{R}_\pm \triangleq \{\tilde{\mathbf{r}}_j \in \mathcal{R} | s_{k-d} = \pm 1\}$. The optimal maximum *a posteriori* probability (MAP) or Bayesian equalizer is defined as [2]

$$\hat{s}_{k-d} = \text{sgn}\,(y_k) \quad \text{with} \quad y_k = \sum_{\tilde{\mathbf{r}}_j \in \mathcal{R}} \frac{\xi_j \tilde{s}_j^{(d)}}{(2\pi\sigma_n^2)^{m/2}} \cdot \exp\left(-\frac{\|\mathbf{r}_k - \tilde{\mathbf{r}}_j\|^2}{2\sigma_n^2}\right) \quad (3)$$

where $\tilde{s}_j^{(d)} \in \{\pm 1\}$ denotes the dth element of $\tilde{\mathbf{s}}_j$, which serves as the class label, and ξ_j is the *a priori* probability of $\tilde{\mathbf{r}}_j$.

The optimal Bayesian equalizer requires the complete knowledge of all the noise-free channel states, which is generally unknown. Kernel-based models have clear advantages, as they can be trained using noisy data.

III. The Relevance Vector Machine Equalizer

Given a block of N training data $\{\mathbf{r}_k, t_k = s_{k-d}\}_{k=1}^N$, consider the equalizer of the form

$$y(\mathbf{r}) = \sum_{l=1}^{N} w_l K_l(\mathbf{r}) \quad (4)$$

where w_l are the "weights" and $K_l(\mathbf{r}) = K(\mathbf{r}, \mathbf{r}_l)$. For equalization application, the kernel function $K(\cdot, \cdot)$ is naturally chosen to be a Gaussian function with its variance being an estimate of the channel noise variance. The relevance vector (RV) approach for classification [7] can readily be applied to construct the equalizer (4). Denote $\mathbf{t} = [t_1 \ldots t_N]^T$ and $\mathbf{w} = [w_1 \ldots w_N]^T$. The posterior probability of $\mathbf{w}$ is

$$p(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}) = \frac{p(\mathbf{t}|\mathbf{w}, \boldsymbol{\alpha}) p(\mathbf{w}|\boldsymbol{\alpha})}{p(\mathbf{t}|\boldsymbol{\alpha})} \quad (5)$$

where

$p(\mathbf{w}\|\boldsymbol{\alpha})$	prior with $\boldsymbol{\alpha} = [\alpha_1 \cdots \alpha_N]^T$ denoting the vector of hyperparameters;
$p(\mathbf{t}\|\mathbf{w}, \boldsymbol{\alpha})$	likelihood;
$p(\mathbf{t}\|\boldsymbol{\alpha})$	evidence.

Following the Bayesian classification framework [9], the likelihood is expressed as

$$p(\mathbf{t}|\mathbf{w}, \boldsymbol{\alpha}) = \prod_{l=1}^{N} (f(y(\mathbf{r}_l)))^{t_l} (1 - f(y(\mathbf{r}_l)))^{1-t_l} \quad (6)$$

where

$$f(x) = \frac{1}{1 + \exp(-x)} \quad (7)$$

is the logistic sigmoid function. The Gaussian prior is chosen

$$p(\mathbf{w}|\boldsymbol{\alpha}) = \prod_{l=1}^{N} \frac{\sqrt{\alpha_l}}{\sqrt{2\pi}} \exp\left(-\frac{\alpha_l w_l^2}{2}\right). \quad (8)$$

As the marginal likelihood $p(\mathbf{t}|\boldsymbol{\alpha})$ cannot be obtained analytically by integrating out the weights from (6), an iterative procedure is necessitated [9].

With a fixed given $\boldsymbol{\alpha}$, the MAP solution $\mathbf{w}_{\text{MAP}}$ can be obtained by maximizing $\log(p(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}))$ or, equivalently, by minimizing the following cost function:

$$J(\mathbf{w}|\mathbf{t}, \boldsymbol{\alpha}) = \sum_{l=1}^{N} \left(\frac{\alpha_l w_l^2}{2} - t_l \log(f(y(\mathbf{r}_l))) - (1 - t_l)\log(1 - f(y(\mathbf{r}_l)))\right). \quad (9)$$

The gradient of J with respect to $\mathbf{w}$ is

$$\nabla J = \mathbf{A}\mathbf{w} + \boldsymbol{\Phi}^T(\mathbf{f} - \mathbf{t}) \quad (10)$$

where

$\mathbf{A}$	$= \text{diag}\{\alpha_1, \ldots, \alpha_N\}$;
$\mathbf{f}$	$= [f(y(\mathbf{r}_1)) \ldots f(y(\mathbf{r}_N))]^T$;
matrix $\boldsymbol{\Phi}$	has elements $\phi_{i,j} = K(\mathbf{r}_i, \mathbf{r}_j)$.

The Hessian of J is

$$\mathbf{H} = \nabla^2 J = \boldsymbol{\Phi}^T \mathbf{B} \boldsymbol{\Phi} + \mathbf{A} \quad (11)$$

where $\mathbf{B} = \text{diag}\{f(y(\mathbf{r}_1))(1 - f(y(\mathbf{r}_1))), \ldots, f(y(\mathbf{r}_N))(1 - f(y(\mathbf{r}_N)))\}$.

The posterior is approximated around $\mathbf{w}_{\text{MAP}}$ by a Gaussian approximation with the covariance

$$\boldsymbol{\Lambda} = (\mathbf{H}|_{\mathbf{w}_{\text{MAP}}})^{-1} \quad (12)$$

and the mean

$$\boldsymbol{\mu} = [\mu_1 \ldots \mu_N]^T = \boldsymbol{\Lambda}\left(\boldsymbol{\Phi}^T \mathbf{B}\mathbf{t}|_{\mathbf{w}_{\text{MAP}}}\right). \quad (13)$$

The hyperparameters $\boldsymbol{\alpha}$ are updated using [8]

$$\alpha_i^{\text{new}} = \frac{1 - \alpha_i^{\text{old}} \lambda_{i,i}}{\mu_i^2} \quad (14)$$

with $\lambda_{i,i}$ being the diagonal elements of $\boldsymbol{\Lambda}$.

The introduction of an individual hyperparameter for every weight of the model (4) is the key feature of the RVM, and is ultimately responsible for the sparsity properties of the RVM method [7]. During the optimization process, many of the α_i are driven to large values, so that the corresponding model weights w_i are effectively pruned out. Thus the corresponding model terms $K_i(\cdot)$ can be removed from the trained model represented by (4). The simple iterative procedure that we adopt to construct an RVM equalizer is summarized as follows.

Initialization) The $N \times M$ kernel matrix $\boldsymbol{\Phi}$ is initialized with $M = N$, i.e., every training data point is considered as a candidate kernel. Each weight w_i is initially associated with a same value of the hyperparameter α_i.

Step 1) Given current value $\boldsymbol{\alpha}$, find $\mathbf{w}_{\text{MAP}}$ by minimizing the cost function (9). A simplified conjugate gradient algorithm [10] is used in the optimization. Alternatively, the iteratively reweighted least-square algorithm [11] can be used.

Step 2) The hyperparameters are updated using (14). If a $\alpha_i > Lg$, where Lg is a preset large positive value, $M := M - 1$, the corresponding column in $\boldsymbol{\Phi}$ is removed, and thus the corresponding weight w_i and model term $K_i(\cdot)$ is pruned out the model.

Test) If the hyperparameters $\boldsymbol{\alpha}$ remain sufficiently unchanged in two successive iterations (no removal of hyperparameters) or a preset maximum iteration number is reached, stop; otherwise go to Step 1.

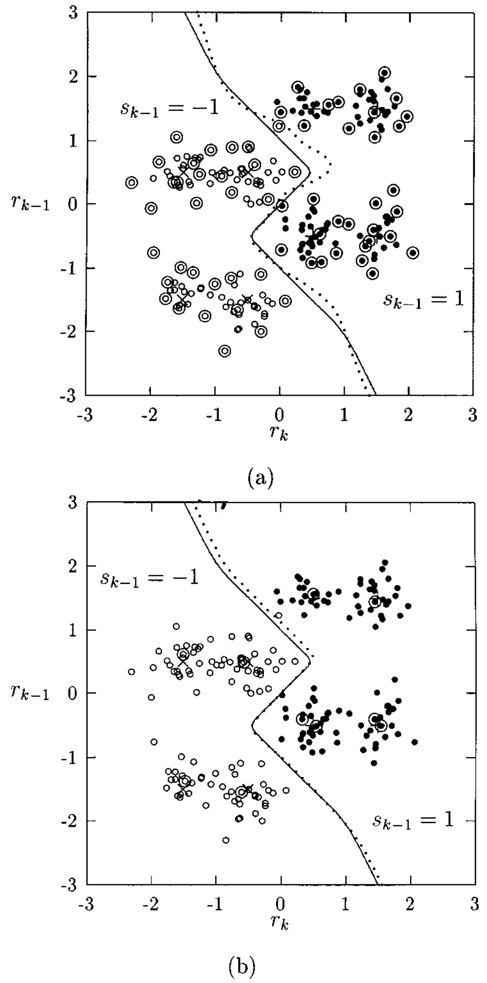
IV. Simulation Results

Two examples were used in simulation to test the RVM algorithm discussed in the previous section and to compare its performance with the SVM.

Example 1: The transfer function of the CIR was $C(z) = 0.5 + 1.0z^{-1}$, and the equalizer structure was given by $m = 2$ and $d = 1$. The number of the noise-free channel states is $N_s = 8$. The training data set contained 200 points. For a channel signal to noise ratio (SNR) of 12 dB, when the error/margin tradeoff parameter C for the SVM was chosen to be $C \ge 2.0$, the size of the SVM was not reduced any further and remained typically 64 SVs in various runs. The RVM on the other hand found the number of RVs in the range of six to ten in various runs. Fig. 1 shows typical results of the SVM and RVM, respectively. It is clear that the RVM method results in a much sparser model and the

TABLE I
BER PERFORMANCE AND NUMBER OF KERNELS USED BY VARIOUS EQUALIZERS FOR EXAMPLE 2, GIVEN THREE SNRS. THE SVM HAD A $C \geq 6.0$

SNR	5 dB			11 dB			17 dB		
model	SVM	RVM	Bayesian	SVM	RVM	Bayesian	SVM	RVM	Bayesian
$\log_{10}$(BER)	-0.93	-0.93	-0.98	-1.63	-1.71	-1.82	-3.45	-3.46	-3.69
kernels	275	12	32	277	20	32	280	30	32



(a)

(b)

Fig. 1. Comparison of the optimal Bayesian decision boundary (solid) with those (dotted) of the SVM equalizer (a) and the RVM equalizer (b) for Example 1, given SNR $= 12$ dB. The × and + are the two classes of the noise-free channel states, small circles and dishes are the two classes of the training data, respectively, and big circles denote SVs (a) and RVs (b).

RVs bear close resemblance to the prototypes, the noise-free channel states. Fig. 2 compares the bit error rate (BER) of the optimal Bayesian equalizer as a function of SNR with that of the RVM equalizer. Again, for a given SNR value, a training set of 200 points was used to construct the RVM equalizer, and the number of RVs was found to in the range of five to ten. The BERs of the SVM equalizer constructed under the same training conditions are not shown in Fig. 2, as they are similar to those of the RVM equalizer. However, the SVM equalizer has a much larger number of kernels, typically around 64.

Example 2: The transfer function of the CIR was $C(z) = 0.3 + 0.8z^{-1} + 0.3z^{-2}$, and the equalizer was defined by $m = 3$ and $d = 1$. The number of the noise-free channel states is $N_s = 32$. Given three SNR conditions with each having a training data set of 500 points, Table I summarized the results obtained using the SVM and RVM methods, respectively, in comparison with the optimal Bayesian equalizer. Fig. 3 compares the BER performance of the Bayesian equalizer with that of the RVM equalizer, given a range of SNR values. For a given SNR value, the number of training data for constructing the RVM was 500. The numbers of RVs found were typically in the range of 12 (for lower SNRs) to 30 (for high SNRs). The BERs of the SVM equalizer, not shown in Fig. 3, are similar to those of the RVM equalizer. The numbers of SVs found however were typically more than 270.

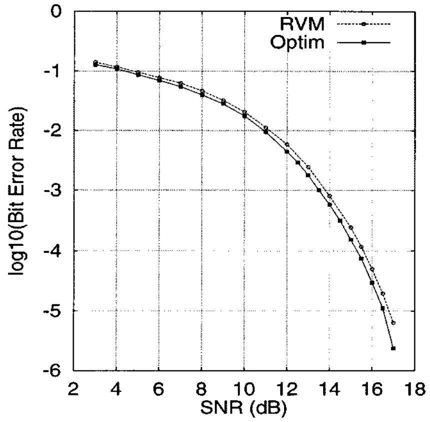


Fig. 2. Performance comparison for Example 1. The number of training data for each given SNR was 200 and average numbers of RVs were six (low SNRs) and eight (high SNRs).

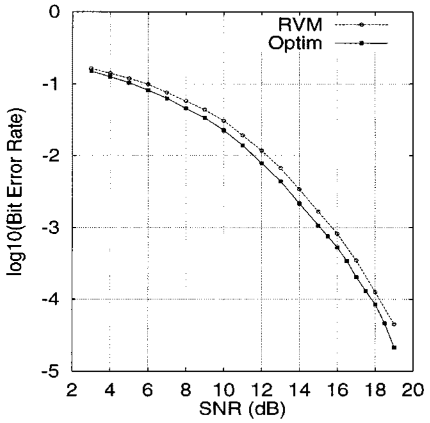


Fig. 3. Performance comparison for Example 2. The number of training data for each given SNR was 500 and average numbers of RVs were in the range of 12 to 30 (for low SNRs to high SNRs).

V. CONCLUDING REMARKS

For high-speed data communications, to meet real-time computational requirements, an equalizer should have a size as small as possible without sacrificing too much performance. In this correspondence, the RVM has been shown to provide an effective method for constructing such an equalizer. The results obtained have demonstrated that the RVM equalizer can closely match the optimal performance of the Bayesian equalizer with fewer kernels.

For time-varying channels, it is necessary to adapt an equalizer in a sample-by-sample manner. Like the SVM, the RVM is a block-data-based method and cannot offer this desired sample-by-sample adaptation. A possible solution is to use the RVM in link initialization to construct an initial equalizer, and then switch to using a stochastic gradient minimum BER adaptive algorithm [12] for tracking the time-varying channel.

References

[1] V. Vapnik, *The Nature of Statistical Learning Theory*. New York: Springer-Verlag, 1995.

[2] S. Chen, B. Mulgrew, and P. M. Grant, “A clustering technique for digital communications channel equalization using radial basis function networks,” *IEEE Trans. Neural Networks*, vol. 4, pp. 570–579, July 1993.

[3] S. Chen, S. Gunn, and C. J. Harris, “Decision feedback equalizer design using support vector machines,” *Proc. Inst. Elect. Eng. Vision, Image, Signal Processing*, vol. 147, no. 3, pp. 213–219, 2000.

[4] D. J. Sebald and J. A. Bucklew, “Support vector machine techniques for nonlinear equalization,” *IEEE Trans. Signal Processing*, vol. 48, pp. 3217–3226, Nov. 2000.

[5] F. Albu and D. Martinez, “The application of support vector machines with Gaussian kernels for overcoming cochannel interference,” in *Proc. 9th IEEE Int. Workshop Neural Networks Signal Processing*, Madison, WI, Aug. 23–25, 1999, pp. 49–57.

[6] S. Chen, A. K. Samingan, and L. Hanzo, “Support vector machine multiuser receiver for DS-CDMA signals in multipath channels,” *IEEE Trans. Neural Networks*, vol. 12, pp. 604–611, May 2001.

[7] M. E. Tipping, “The relevance vector machine,” in *Advances in Neural Information Processing Systems 12*, S. A. Solla, T. K. Leen, and K.-R. Müller, Eds. Cambridge, MA: MIT Press, 2000.

[8] D. J. C. MacKay, “Bayesian interpolation,” *Neural Comput.*, vol. 4, no. 3, pp. 415–447, 1992.

[9] ——, “The evidence framework applied to classification networks,” *Neural Comput.*, vol. 4, pp. 720–736, 1992.

[10] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty, *Nonlinear Programming: Theory and Algorithms*. New York: Wiley, 1993.

[11] I. T. Nabney, “Efficient training of RBF networks for classification,” in *Proc. 9th ICANN*, vol. 1, 1999, pp. 210–215.

[12] S. Chen, A. K. Samingan, and L. Hanzo, “Adaptive minimum error rate training for neural networks with application to multiuser detection in CDMA communication systems,” IEEE Trans. Neural Networks, 2001, submitted for publication.

Global Convergence of Delayed Dynamical Systems

Tianping Chen

***Abstract*—In this paper, we discuss some delayed dynamical systems, investigating their stability and convergence in critical case. To ensure the stability, the coefficients of the dynamical system must satisfy some inequalities. In most existing literatures, the restrictions on the coefficients are strict inequalities. The tough question is what will happen in the case (critical case) the strict inequalities are replaced by nonstrict inequalities (i.e., “$<$” is replaced by “$\leq$”). The purpose of this paper is to discuss this critical case and give an affirmative answer in the case that the activation functions are hyperbolic tangent.**

***Index Terms*—Delayed neural networks, global convergence, stability.**

I. Introduction

Recently, delayed Hopfield neural networks and cellular neural networks attracted attentions of researchers. A single time delay $\tau > 0$ was introduced in [13] by Marcus and Westervelt, where they considered following differential equations with delay:

$$C_i \frac{du_i(t)}{dt} = -\frac{u_i(t)}{R_i} + \sum_{j=1}^{N} T_{ij} v_j(t-\tau) + I_i$$
$$i = 1, \ldots, N \qquad u(t_0) = u^0 \in R^N \tag{1}$$

where

$v_j \quad = g_j(u_j);$
$j \quad = 1, \cdots, N;$
all g_j activation functions, for example, sigmoidal functions, etc.

System (1) has much more complicated dynamics than original Hopfield networks due to the delay. For results on this system, refer, for example, to [1].

In [9], Gopalsamy and He considered a modification of (1) by incorporating different delays $\tau_{ij} \geq 0$, i.e.,

$$\frac{du_i(t)}{dt} = -b_i u_i(t) + \sum_{j=1}^{N} a_{ij} g_j(u_j(t-\tau_{ij}) + I_i$$
$$i = 1, \ldots, N \tag{2}$$

where g_j are sigmoidal functions.

After then, there are several papers (see [3], [4], [7], [12], [14], etc.) discussing the following delayed neural networks:

$$\frac{du_i(t)}{dt} = -\gamma_i u_i(t) + \sum_{j=1}^{N} a_{ij} g_j(u_j(t))$$
$$+ \sum_{j=1}^{N} b_{ij} f_j(u_j(t-\tau_j)) + I_i \qquad i = 1, \ldots, N \tag{3}$$

where all g_i and f_i satisfy Lipschitz condition with Lipschitz constants μ_i and ν_i, $i = 1, \ldots, N$, respectively. And the following global convergence theorem has been proved.

Theorem A: If the activation functions g_i $i = 1, \ldots, N$ satisfy $(H1)$, $(H2)$, and the coefficients in the dynamical system (3) satisfy the following inequalities:

$$-\gamma_i + \sum_{i=1}^{N} |a_{ij}|\mu_j + \sum_{i=1}^{N} |b_{ij}|\nu_j < 0 \qquad i = 1, \ldots, N \tag{4}$$

where $\gamma_i > 0$ are certain constants. Then the dynamical system (3) has a unique equilibrium u^*, which is globally and asymptotically stable (exponentially).

If the inequality sign “$<$” is reversed, the previous theorem fails to be true. One can give an example of a delayed dynamical system, which has several equilibrium points and some are unstable or an example of a system, of which the solutions diverge.

It is natural to ask: What will happen if the strict inequalities (4) are replaced by the following nonstrict inequalities?

$$-\gamma_i + \sum_{i=1}^{N} |a_{ij}|\mu_j + \sum_{i=1}^{N} |b_{ij}|\nu_j \leq 0 \qquad i = 1, \ldots, N. \tag{5}$$

The purpose of this paper is to give an affirmative answer when the activation functions are hyperbolic tangent. (For the systems without delays, see [8]).

Manuscript received May 11, 2001; revised July 16, 2001. This work was supported by the National Natural Science Foundations of China under Grants 69982003 and 60074005. This work was completed when the author visited Riken Brain Science Institute, Japan.

The author is with the Lab of Nonlinear Mathematics Science, Institute of Mathematics, Fudan University, Shanghai, P. R. China.

Publisher Item Identifier S 1045-9227(01)08903-2.