Supervised Learning of Perceptron and Output Feedback Dynamic Networks: A Feedback Analysis via the Small Gain Theorem

Markus Rupp and Ali H. Sayed, *Member, IEEE*

***Abstract*—This paper provides a time-domain feedback analysis of the perceptron learning algorithm and of training schemes for dynamic networks with output feedback. It studies the robustness performance of the algorithms in the presence of uncertainties that might be due to noisy perturbations in the reference signals or to modeling mismatch. In particular, bounds are established on the step-size parameters in order to guarantee that the resulting algorithms will behave as robust filters. The paper also establishes that an intrinsic feedback structure can be associated with the training schemes. The feedback configuration is motivated via energy arguments and is shown to consist of two major blocks: a time-variant lossless (i.e., energy preserving) feedforward path and a time-variant feedback path. The stability of the feedback structure is then analyzed via the small gain theorem and choices for the step-size parameter in order to guarantee faster convergence are deduced by appealing to the mean-value theorem. Simulation results are included to demonstrate the findings.**

***Index Terms*—Convergence speed, dynamic networks, feedback structure, l_2–stability, mean-value theorem, perceptron learning, positive realness, robust algorithm, small gain theorem.**

I. Introduction

APPLICATIONS of neural networks span a variety of areas in pattern recognition, filtering, and control. When supervised learning is employed, a training phase is always necessary. During this phase, a recursive update procedure is used to estimate the weight vector of the linear combiner that "best" fits the given data [1]–[3]. The recursive procedure often requires that a suitable adaptation gain (or step-size parameter) be chosen and, in most cases, heuristics and trial-and-error experiences are used to select a suitable step-size value for the training period.

The "common" practice has been to choose small adaptation gains. But the smaller the adaptation gain the slower the convergence speed. In several cases, especially in large-scale applications with many weights and many training patterns, this may require a considerable amount of time and machine power.

Manuscript received July 10, 1995; revised June 19, 1996 and December 20, 1996. This work was supported by the National Science Foundation under Award MIP-9409319 and a scholarship from DAAD (German Academic Exchange Service) as well as the scientifical division of NATO.

M. Rupp is with the Wireless Technology Research Dept., Lucent Technologies, Holmdel, NJ 07733–0400 USA.

A. H. Sayed is with the Department of Electrical Engineering, University of California, Los Angeles, CA 90095 USA.

Publisher Item Identifier S 1045-9227(97)02757-4.

In recent work on the robustness analysis of adaptive schemes [4]–[6], the authors have addressed the following two issues.

1) We have shown how to select the adaptation gain in order to guarantee a robust behavior in the presence of noise and modeling uncertainties (i.e., in order to guarantee a consistent performance in the sense that "small disturbances" would lead to "small estimation errors").
2) We have also shown how to select the adaptation gain in order to guarantee faster convergence speeds.

The formulation in [4]–[6] highlights an intrinsic feedback structure for most adaptive schemes and it relies on tools from system theory, control, and signal processing such as: state-space descriptions, feedback analysis and the small gain theorem, H^∞-design, and transmission lines and lossless systems.

In this paper we address the implications of these results to the training of perceptrons and recurrent neural networks. We start by considering the so-called perceptron learning algorithm (PLA, for short), which involves a nonlinear functional in the update equation due to the presence of an activation function (usually a sigmoid function). We show how to extend the feedback arguments of [4] and [5] in order to handle the presence of the nonlinearity and, as a fallout, we suggest several choices for the step-size parameter in order to guarantee faster convergence and robust performance. We also establish the existence of a feedback structure that can be associated with the PLA.

The feedback configuration is motivated via energy arguments and is shown to consist of two major blocks: a time-variant *lossless* (i.e., energy preserving) feedforward path and a time-variant feedback path. The analysis applies to both cases when the feedback path is static or dynamic (which occurs in the case of recurrent networks), and it provides physical insights into the energy propagation through the feedback system. This enables us to suggest modifications to the training algorithm, in terms of selections of the adaptation gain, in order to accelerate the convergence speed during the training phase.

In the later sections of the paper, we extend the time-domain feedback analysis of the PLA to dynamic neural networks (also recurrent neural networks or RNN's) with output feedback [7], [8], and provide a study of the robustness performance of the

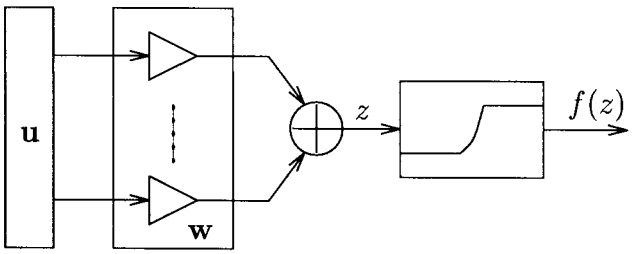

Fig. 1. The perceptron structure.

training phase in the presence of uncertainties. In this context, certain positive-realness conditions arise in much the same way as in the study of IIR adaptive filters and identification schemes [9]–[11]. Here, however, as indicated in the remark after inequality (40), a less restrictive condition is tolerable in view of the presence of the nonlinear activation function.

A. Notation

Small boldface letters are used to denote vectors (e.g., $\mathbf{u}$), the letter "T" to denote transposition, and $\|\mathbf{x}\|$ to denote the Euclidean norm of a vector $\mathbf{x}$. Also, subscripts are used for time-indexing of vector quantities (e.g., $\mathbf{u}_i$) and parenthesis for time-indexing of scalar quantities (e.g., $v(i)$). All vectors are column vectors except for the row vectors $\mathbf{u}_i$.

II. The Perceptron

Consider two sets, $\mathcal{S}_0$ and $\mathcal{S}_1$, of M-dimensional real-valued row vectors $\mathbf{u}$ that are characterized by either property A or property B, say

$$\mathcal{S}_0 = \{\mathbf{u} \in \mathbb{R}^M \,|\, \mathbf{u} \text{ has property } A\}$$
$$\mathcal{S}_1 = \{\mathbf{u} \in \mathbb{R}^M \,|\, \mathbf{u} \text{ has property} B\}.$$

If the two sets are linearly separable, then a classification scheme that can be used to decide whether a given vector $\mathbf{u}$ belongs to one class or the other is to employ a perceptron device [1]–[3].

The perceptron consists of a linear combiner, whose column weight vector we denote by $\mathbf{w}$, followed by a nonlinearity $f[z]$ (also known as an activation function), as depicted in Fig. 1. The value assumed by $y = f[z] = f[\mathbf{uw}]$ can be interpreted as the likelihood that the input vector belongs to one class or another.

A common choice for $f[z]$ is to employ the sigmoid function

$$f_\beta(z) = \frac{1}{1+e^{-\beta z}}, \quad \beta > 0. \tag{1}$$

This is a function that varies monotonically from 0 to 1 for $z \in (-\infty, \infty)$, and its transition region around $z = 0$ is more or less steep depending on whether the parameter β is large or small. In particular, for $\beta \to \infty$, the sigmoid function collapses to the hard-limiting function

$$f_\infty(z) = \frac{1+\operatorname{sgn}(z)}{2} = \begin{cases} 0 & \text{if } z \le 0 \\ 1 & \text{if } z > 0. \end{cases}$$

A. The Perceptron Learning Algorithm (PLA)

Consider a collection of input vectors $\{\mathbf{u}_i\}$ with the corresponding correct (or desired) output values $\{y(i)\}$. The $\{y(i)\}$ are assumed to belong to the range of the activation function $f[\cdot]$, i.e., there exists an unknown column vector $\mathbf{w}$ such that

$$y(i) = f[\mathbf{u}_i\mathbf{w}] \quad \text{for some } \mathbf{w}. \tag{2}$$

This is in agreement with the models used in [13] and [14].

In supervised learning, the perceptron is presented with the given input-output data $\{\mathbf{u}_i, y(i)\}$, and the objective is to estimate the unknown weight vector $\mathbf{w}$. The PLA computes recursive estimates of $\mathbf{w}$ as follows. It starts with an arbitrary initial guess $\mathbf{w}_{-1}$ and applies the update rule

$$\mathbf{w}_i = \mathbf{w}_{i-1} + \mu\mathbf{u}_i^T(y(i) - f[\mathbf{u}_i\mathbf{w}_{i-1}]) \tag{3}$$

where $f[z]$ is the sigmoid activation function or, more generally, any monotonically increasing function.

For generality, we consider in this paper the possibility of noisy perturbations in the reference signal. These can be due to model mismatching or to measurement noise.[1] We denote the perturbed references by $\{d(i)\}$ (which are now the given data instead of $\{y(i)\}$), say

$$d(i) = f[\mathbf{u}_i\mathbf{w}] + v(i) = y(i) + v(i) \tag{4}$$

where $v(i)$ denotes the noise term. Correspondingly, we study the following general form of recursion (3):

$$\mathbf{w}_i = \mathbf{w}_{i-1} + \mu(i)\mathbf{u}_i^T(d(i) - f[\mathbf{u}_i\mathbf{w}_{i-1}]) \tag{5}$$

where $d(i)$ replaces $y(i)$ and where we have allowed for a time-variant step-size parameter $\mu(i)$.

We shall also, and without loss of generality, assume that the $\{\mathbf{u}_i\}$ are nonzero. For a nonzero step-size $\mu(i)$, a zero $\mathbf{u}_i$ simply corresponds to a nonactive update step since it keeps the weight estimate unaltered, i.e., $\mathbf{w}_i = \mathbf{w}_{i-1}$.

B. Error Measures

The following error quantities are useful for our later analysis: $\tilde{\mathbf{w}}_i$ denotes the difference between the true weight $\mathbf{w}$ and its estimate $\mathbf{w}_i$, $\tilde{\mathbf{w}}_i = \mathbf{w} - \mathbf{w}_i$, $e_a(i)$ denotes the *a priori* estimation error, $e_a(i) = \mathbf{u}_i\tilde{\mathbf{w}}_{i-1} = z(i) - \hat{z}(i)$, and $e_p(i)$ denotes the *a posteriori* estimation error, $e_p(i) = \mathbf{u}_i\tilde{\mathbf{w}}_i$. It follows from (5) that the weight-error vector satisfies the recursion:

$$\tilde{\mathbf{w}}_i = \tilde{\mathbf{w}}_{i-1} - \mu(i)\mathbf{u}_i^T(d(i) - f[\mathbf{u}_i\mathbf{w}_{i-1}]). \tag{6}$$

C. Robustness Issues

In the sequel we focus on model (4) and study the robustness behavior of the update recursion (5). Intuitively, a robust algorithm is one for which the estimation errors are consistent with the disturbances in the sense that "small" disturbances would lead to "small" estimation errors, no matter what the disturbances are! This is not generally true for any adaptive

[1] Assume for example that the reference system employs $f_\infty(z)$ while the trained perceptron employs $f_\beta(z)$ with $\beta = 4$. The differences, which occur mainly around $z = 0$, can be described by an additive noise term.

filter: the estimation errors can still be large even in the presence of small disturbances (see, e.g., [12]). While a more precise mathematical formulation is provided in the sections to follow, we stress here that the motivation for our analysis is twofold.

1) To provide conditions on the adaptation gain (or step-size parameter) in order to guarantee a robust behavior during the training phase. By this we mean a training algorithm that results in "small" errors if the disturbances are "small." It turns out that such a desirable performance is not guaranteed for any choice of the step-size.
2) To suggest choices for the step-size parameter that would result in faster convergence speeds.

The robustness issue is addressed here in a purely deterministic framework and without assuming prior knowledge of noise statistics. This is especially useful in situations where prior statistical information is missing since a robust design would guarantee a desired level of robustness independent of the noise statistics. In loose terms, robustness would imply that the ratio of an estimation error energy to the noise or disturbance energy will be guaranteed to be upper bounded by a positive constant, say the constant one

$$\frac{\text{estimation error energy}}{\text{disturbance energy}} \leq 1. \tag{7}$$

From a practical point of view, a relation of the form (7) is desirable since it guarantees that the resulting estimation error energy will be upper bounded by the disturbance energy, no matter what the nature and the statistics of the disturbances are. One of the contributions of this work is to show how to select the adaptation gains $\mu(i)$ in (5) in order to guarantee 1) a robust behavior and 2) faster convergence. This is addressed in the next sections.

III. A Contractive Mapping

We first establish a passivity relation that shows how the sum of the Euclidean norms of the weight-error vector and the *a priori* estimation error at time i

$$\|\tilde{\mathbf{w}}_i\|^2 + \mu(i)|e_a(i)|^2$$

compares with the sum of the Euclidean norms of the weight-error vector at time $i-1$ and a disturbance term that is defined in (8) further ahead

$$\|\tilde{\mathbf{w}}_{i-1}\|^2 + \mu(i)|\tilde{v}(i)|^2.$$

The significance (and implications) of the relation to be established here will become clear as we progress in our discussions.

We denote the difference $d(i) - f[\mathbf{u}_i\mathbf{w}_{i-1}]$ in (6) by $\tilde{e}_a(i)$ and note that it is equal to $[e_a(i) + \tilde{v}(i)]$, where the modified disturbance $\tilde{v}(i)$ is defined by

$$\tilde{v}(i) = -e_a(i) + f[\mathbf{u}_i\mathbf{w}] - f[\mathbf{u}_i\mathbf{w}_{i-1}] + v(i). \tag{8}$$

This allows us to rewrite (6) as

$$\tilde{\mathbf{w}}_i = \tilde{\mathbf{w}}_{i-1} - \mu(i)\mathbf{u}_i^T\tilde{e}_a(i). \tag{9}$$

If we now compute the squared norm (i.e., energies) of both sides of (9), we conclude that the following equality always holds:

$$\begin{aligned}&\|\tilde{\mathbf{w}}_i\|^2 + \mu(i)|e_a(i)|^2 + \mu(i)\big(1 - \mu(i)\|\mathbf{u}_i\|_2^2\big)|\tilde{e}_a(i)|^2 \\ &\quad = \|\tilde{\mathbf{w}}_{i-1}\|^2 + \mu(i)|\tilde{v}(i)|^2.\end{aligned}$$

This equality allows us to conclude that the following energy bounds are always satisfied, where we have introduced the parameter $\bar{\mu}(i) = 1/\|\mathbf{u}_i\|^2$.

Lemma 1: Consider the perceptron learning recursion (3)–(4). It always holds, at each time instant i, that

$$\frac{\|\tilde{\mathbf{w}}_i\|^2 + \mu(i)|e_a(i)|^2}{\|\tilde{\mathbf{w}}_{i-1}\|^2 + \mu(i)|\tilde{v}(i)|^2} \begin{cases} \leq 1 & \text{for } 0 < \mu(i) < \bar{\mu}(i) \\ = 1 & \text{if } \mu(i) = \bar{\mu}(i) \\ \geq 1 & \text{for } \mu(i) > \bar{\mu}(i) \end{cases}$$

where $\tilde{v}(i)$ is the modified disturbance given by (8).

The first two inequalities in the statement of the lemma establish that if the adaptation gain is chosen such that $\mu(i) \leq \bar{\mu}(i)$, then the mapping from the signals $\{\tilde{\mathbf{w}}_{i-1}, \sqrt{\mu(i)}\tilde{v}(i)\}$ to the signals $\{\tilde{\mathbf{w}}_i, \sqrt{\mu(i)}e_a(i)\}$ is contractive. [A linear map that takes x to y, say $y = T[x]$, is said to be contractive if for all x we have $\|T[x]\|^2 \leq \|x\|^2$. That is, the output energy does not exceed the input energy].

Therefore, we see that the first two cases in the lemma establish a local error-energy bound (or passivity relation) that highlights a robustness property of recursion (5): They state that no matter what the value of the noise component $\tilde{v}(i)$ is, and no matter how far the estimate $\mathbf{w}_{i-1}$ is from the true vector $\mathbf{w}$, the sum of energies $\|\tilde{\mathbf{w}}_i\|_2^2 + \mu(i)|e_a(i)|^2$, will always be smaller than or equal to the sum of energies $\|\tilde{\mathbf{w}}_{i-1}\|_2^2 + \mu(i)|\tilde{v}(i)|^2$.

Moreover, since this contractivity property holds for each time instant i, it should also hold globally over an interval of time. Indeed, assuming $\mu(i) \leq \bar{\mu}(i)$ over $0 \leq i \leq N$, it follows from Lemma 1 that

$$\|\tilde{\mathbf{w}}_N\|^2 + \sum_{i=0}^{N}\mu(i)|e_a(i)|^2 \leq \|\tilde{\mathbf{w}}_{-1}\|^2 + \sum_{i=0}^{N}\mu(i)|\tilde{v}(i)|^2.$$

We may remark that other similar local and global passivity relations can be established by using *a posteriori* (rather than *a priori*) estimation errors [6]. But we shall forgo the details here and focus instead on a time-domain and feedback analysis of the PLA.

IV. A Time-Domain Feedback-Analysis

The bounds of Lemma 1 can be described in an alternative form that leads to an interesting feedback structure. For this purpose, we first note that it can also be shown that the update equation (5) can be written in the form (cf. the analysis in [4])

$$\mathbf{w}_i = \mathbf{w}_{i-1} + \bar{\mu}(i)\mathbf{u}_i^T[e_a(i) - e_p(i)] \tag{10}$$

where we have used the fact that

$$e_p(i) = e_a(i) - \frac{\mu(i)}{\bar{\mu}(i)}(f[\mathbf{u}_i\mathbf{w}] - f[\mathbf{u}_i\mathbf{w}_{i-1}] + v(i)). \tag{11}$$

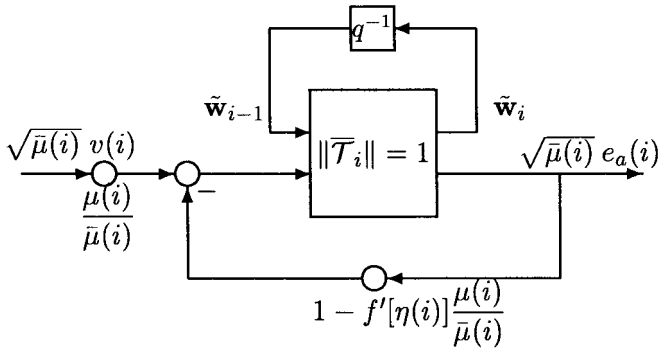

Fig. 2. A time-variant lossless mapping with gain feedback for the perceptron learning algorithm.

Consequently

$$\tilde{\mathbf{w}}_i = \tilde{\mathbf{w}}_{i-1} - \bar{\mu}(i)\mathbf{u}_i^T[e_a(i) - e_p(i)]. \tag{12}$$

Relation (12) has the same form as the update (9), except for a different disturbance ($\tilde{v}(i)$ is now replaced by $-e_p(i)$) and for a step-size that is equal to $\bar{\mu}(i)$ itself. Hence, the same arguments that led to Lemma 1 would imply that the following equality holds for all possible choices of $\mu(i)$:

$$\frac{\|\tilde{\mathbf{w}}_i\|^2 + \bar{\mu}(i)|e_a(i)|^2}{\|\tilde{\mathbf{w}}_{i-1}\|^2 + \bar{\mu}(i)|e_p(i)|^2} = 1. \tag{13}$$

This establishes a lossless mapping $\mathcal{T}_i$ from the signals $\{\tilde{\mathbf{w}}_{i-1}, \sqrt{\bar{\mu}(i)}e_p(i)\}$ to the signals $\{\tilde{\mathbf{w}}_i, \sqrt{\bar{\mu}(i)}e_a(i)\}$.

If we further apply the mean-value theorem to the activation function $f[z]$, we can write

$$f[\mathbf{u}_i\mathbf{w}] - f[\mathbf{u}_i\mathbf{w}_{i-1}] = f'[\eta(i)]e_a(i)$$

for some point $\eta(i)$ along the segment connecting $\mathbf{u}_i\mathbf{w}$ and $\mathbf{u}_i\mathbf{w}_{i-1}$. Therefore, (11) leads to

$$-\bar{\mu}^{\frac{1}{2}}(i)e_p(i) = \frac{\mu(i)}{\bar{\mu}^{\frac{1}{2}}(i)}v(i) - \left[1 - f'[\eta]\frac{\mu(i)}{\bar{\mu}(i)}\right]\bar{\mu}^{\frac{1}{2}}(i)e_a(i).$$

Combining with (13), this relation shows that the overall mapping from the *original* (weighted) disturbances $\sqrt{\bar{\mu}(\cdot)}v(\cdot)$ to the resulting *a priori* (weighted) estimation errors $\sqrt{\bar{\mu}(\cdot)}e_a(\cdot)$ can be expressed in terms of a feedback structure, as shown in Fig. 2.

The stability of such feedback structures can be studied via tools that are by now standard in system theory (e.g., the small gain theorem [15], [16]). This is pursued in the next section where we derive conditions on the step-size parameters $\{\mu(i)\}$ and on the activation function $f[z]$ in order to guarantee a robust training algorithm, as well as faster convergence speeds.

This will be achieved by establishing conditions under which the feedback configuration of Fig. 2 is l_2-stable in the sense that it should map a finite-energy input noise sequence (which includes the noiseless case as a special case) $\{\sqrt{\bar{\mu}(\cdot)}v(\cdot)\}$ to a finite-energy *a priori* error sequence $\{\sqrt{\bar{\mu}(\cdot)}e_a(\cdot)\}$.

V. l_2-Stability and the Small Gain Theorem

Define

$$\gamma(N) = \max_{0\le i\le N} \mu(i)/\bar{\mu}(i)$$
$$\Delta(N) = \max_{0\le i\le N} \left|1 - f'[\eta(i)]\frac{\mu(i)}{\bar{\mu}(i)}\right|.$$

That is, $\Delta(N)$ is the maximum absolute value of the gain of the feedback loop over $0 \le i \le N$.

It can be easily shown that if $\Delta(N) < 1$ (see, e.g., [4]) then the following two relations hold:

$$\sqrt{\sum_{i=0}^{N} \bar{\mu}(i)|e_a(i)|^2} \le \frac{1}{1-\Delta(N)}\left[\|\tilde{\mathbf{w}}_{-1}\| + \gamma(N)\sqrt{\sum_{i=0}^{N}\bar{\mu}(i)|v(i)|^2}\right] \tag{14}$$

and

$$\sqrt{\sum_{i=0}^{N} \mu(i)|e_a(i)|^2} \le \frac{\gamma^{1/2}(N)}{1-\Delta(N)}\left[\|\tilde{\mathbf{w}}_{-1}\| + \gamma^{1/2}(N)\sqrt{\sum_{i=0}^{N}\mu(i)|v(i)|^2}\right]. \tag{15}$$

Expression (14) compares the energies of *a priori* estimation errors and the disturbances (but now normalized by $\bar{\mu}(i)$ rather than $\mu(i)$). In particular, it establishes that the map from $\{\tilde{\mathbf{w}}_{-1}, \sqrt{\bar{\mu}(\cdot)}v(\cdot)\}$ to $\{\sqrt{\bar{\mu}(\cdot)}e_a(\cdot)\}$ is l_2-stable (it maps a finite energy sequence to another finite energy sequence).

The condition $\Delta(N) < 1$ is a manifestation of the so-called small gain theorem in system analysis [15], [16]. In simple terms, the theorem states that the l_2-stability of a feedback configuration (that includes Fig. 2 as a special case) requires that the product of the norms of the feedforward and the feedback maps be strictly bounded by one. Here, the feedforward map has (2-induced) norm equal to one (since it is lossless) while the 2-induced norm of the feedback map is $\Delta(N)$. Hence, the condition $\Delta(N) < 1$ guarantees an overall contractive map.

Note also that for $\Delta(N) < 1$ to hold, we need to choose the step-size $\mu(i)$ such that, for all i

$$0 < \mu(i)f'[\eta(i)] < 2\bar{\mu}(i) = \frac{2}{\|\mathbf{u}_i\|^2}. \tag{16}$$

A. On Convergence and Energy Propagation

The flow of energy through the feedback connection of Fig. 2 provides further insights into the convergence speed of the training algorithm. For this purpose, let us ignore the measurement noise $v(i)$ and assume that we have noiseless measurements $y(i) = f[\mathbf{u}_i\mathbf{w}]$. It is known in the stochastic setting that for Gaussian processes [18], as well as for spherically invariant random processes [19], the maximal speed of convergence for a gradient algorithm (where $f[x] = x$) is obtained for $\mu(i) = \bar{\mu}(i)$, i.e., for the so-called projection LMS

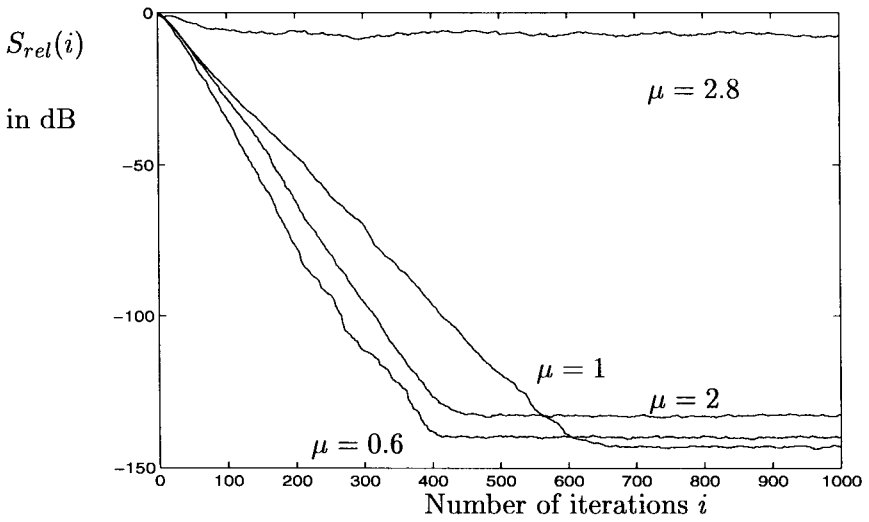

Fig. 3. Learning curves for perceptron learning algorithm with $\beta = 0.4$ and $\mu = 0.6, 1, 2, 2.8$.

algorithm. We now argue that this conclusion is consistent with the feedback configuration of Fig. 2.

If $\mu(i)$ is such that $\mu(i)f'[\eta(i)] = \bar{\mu}(i)$, then the feedback loop is disconnected. This means that there is no energy flowing back into the lower input of the lossless section from its lower output $e_a(\cdot)$. The losslessness of the feedforward path then implies that

$$E_w(i) = E_w(i-1) - E_e(i) \tag{17}$$

where we are denoting by $E_e(i)$ the energy of $\sqrt{\bar{\mu}(i)}\ e_a(i)$ and by $E_w(i)$ the energy of $\tilde{\mathbf{w}}_i$.

But what if $\mu(i)f'[\eta(i)] \neq \bar{\mu}(i)$? In this case the feedback path is active and the convergence speed will be affected. Indeed, we now have

$$E_w(i) = E_w(i-1) - \underbrace{\left(1 - \left|1 - f'[\eta(i)]\frac{\mu(i)}{\bar{\mu}(i)}\right|^2\right)}_{\tau(i)} E_e(i) \tag{18}$$

where we have defined the coefficient $\tau(i)$. It is easy to verify that as long as $\mu(i)$ is chosen to satisfy (16) with $\mu(i)f'[\eta(i)] \neq \bar{\mu}(i)$, we obtain $0 < \tau(i) < 1$. That is, $\tau(i)$ is strictly less than one and the rate of decrease in the energy of $\tilde{\mathbf{w}}_i$ is lowered.

VI. Optimal Choices of Step Sizes

The above energy arguments suggest that faster convergence occurs when $\mu(i)$ is chosen such that $\mu(i) = \bar{\mu}(i)/f'[\eta(i)]$ (which is the middle point of the interval suggested by (16)). But $\eta(i)$ is still unknown and we therefore need to come up with suitable approximations.

The first (but not the most suitable) choice that comes to mind is to assume an upper bound on $f'[\cdot]$, say $f'[\eta] \leq f'_{\max}$ for all η. Then condition (16) can be replaced by the conservative requirement

$$0 < \mu(i) < 2/(f'_{\max}||\mathbf{u}_i||^2). \tag{19}$$

For a large bound $f'_{\max}$, this condition can lead to small step-sizes and, hence, to slow convergence. For the commonly used activation functions, the maximum value of the derivative occurs at the origin. For example, for the sigmoid function we obtain $f'[0] = \beta/4$. We can therefore take $f_{\max} = \beta/4$ and choose the step-size parameter $\mu(i)$ according to $0 < \mu(i) < 8/(\beta||\mathbf{u}_i||^2)$. This is the same bound suggested in [14]. However, the result in [14] was derived under the restrictive assumption that the regression vectors do not change over time, i.e., $\mathbf{u}_i = \mathbf{u} = \text{cte}$ for all i. Our result thus extends the bound to the general scenario of time-variant regression vectors. For improved convergence one might then be tempted to employ $\mu(i) = 4/(\beta||\mathbf{u}_i||^2)$. However, this value is very conservative and usually leads to unsatisfactory results, as the simulations further ahead demonstrate.

For this reason, we take here an alternative route that avoids upper-bounding the derivative of the activation function. Instead, we provide good estimates for the instantaneous derivatives $f'[\eta(i)]$.

To begin with, recall that $f'[\eta(i)]$ is defined by $f'[\eta(i)] = (f[z(i)] - f[\mathbf{u}_i\mathbf{w}_{i-1}])/(z - \mathbf{u}_i\mathbf{w}_{i-1})$, where $z(i) = \mathbf{u}_i\mathbf{w}$. Unfortunately, $z(i)$ and $f[z(i)]$ are not available since $\mathbf{w}$ itself is not known. But one possibility to proceed here is to employ $d(i)$ as an estimate for $f[z(i)]$ since $d(i) = f[z(i)] + v(i)$. This is especially useful if the reference sequence is noise-free or if the noise itself is sufficiently small. Now, with a "known" $f[z(i)]$, it becomes possible to solve for $z(i)$. This motivates us to suggest the following expression for the optimal step-size parameter (we refer to this construction as method A)

$$\mu_{\text{opt}}(i) = \bar{\mu}(i) \min\left(\frac{f^{-1}[d(i)] - \mathbf{u}_i\mathbf{w}_{i-1}}{d(i) - f[\mathbf{u}_i\mathbf{w}_{i-1}]}, T\right) \tag{20}$$

where T is used as a threshold value in order to prevent large step-sizes. For the sigmoid function f_β in (1) we have $f^{-1}[x] = -\ln[1/d(i) - 1]/\beta$ which requires the evaluation of a logarithm at each step. In the case of a symmetric sigmoid function, say $f(x) = [1 + e^{-0.5\beta x}]/[1 - e^{-0.5\beta x}]$, the calculation of an inverse tangent is required since $f^{-1}[x] = 4/\beta \arctan[x]$.

An alternative procedure is to approximate $f'[\eta(i)]$ by the average of $f'[z(i)]$ (or $\approx f'[d(i)]$) and $f'[\mathbf{u}_i\mathbf{w}_{i-1}]$. This is a convenient approximation in light of the "piecewise-linear" form of the activation function. We thus write

$$0 < \mu(i) < 2\bar{\mu}(i)\frac{2}{f'[d(i)] + f'[\mathbf{u}_i\mathbf{w}_{i-1}] + \epsilon} \tag{21}$$

where, for the sigmoid function, $f'[x] = \beta f[x](1 - f[x])$. The positive number ϵ is, similar to T in method A, introduced in order to avoid large step-sizes.

This approximation is however inconvenient in the cases when $\eta(i)$ happens to be close to zero, while $z(i)$ and $\mathbf{u}_i\mathbf{w}_{i-1}$ are reasonably far apart. To avoid a poor approximation in these cases, we may modify the above construction as follows: for improved convergence (i.e., with a disconnected feedback loop) we set

$$\mu_{\text{opt}}(i) = \frac{2\bar{\mu}(i)}{f'[d(i)] + f'[\mathbf{u}_i\mathbf{w}_{i-1}] + \epsilon} \tag{22}$$

if $(d(i) - \frac{1}{2})(f[\mathbf{u}_i\mathbf{w}_{i-1}] - \frac{1}{2}) > 0$ or

$$\mu_{\text{opt}}(i) = \frac{\bar{\mu}(i)}{f'_{\max}} \tag{23}$$

otherwise. We refer to this construction as method B. Condition (23) corresponds to the sigmoid function in (1). For a symmetrical function a similar expression can be obtained.

A third, and perhaps simpler method, is to first estimate $f[\eta(i)]$ by the average of $f[d(i)]$ and $f[\mathbf{u}_i\mathbf{w}_{i-1}]$, i.e., $\hat{f}[\eta(i)] = 0.5(f[d(i)] + f[\mathbf{u}_i\mathbf{w}_{i-1}])$, and then set $f'[\eta(i)] \approx \beta\hat{f}[\eta(i)](1 - \hat{f}[\eta(i)])$. This leads to method C, with the choice

$$\mu_{\text{opt}}(i) = \frac{\bar{\mu}(i)}{\beta\hat{f}[\eta(i)](1 - \hat{f}[\eta(i)]) + \epsilon}. \tag{24}$$

Before extending the earlier results to dynamic networks, we present some simulations that support our conclusions.

VII. Simulation Results for PLA

In all experiments, we have chosen a bipolar white random sequence with variance one as the input signal. The weights to be identified were $\{1, 1, 1, 1, 1, 1, 1, 1, 1\}$. The first coefficient was used for the offset term while the other eight were driven by a bipolar input pattern. A neuron with these weights can be interpreted as one that finds the patterns with more than three $+1$.

The values of the inner signal z are from the set $\{-7, -5, -3, -1, 1, 3, 5, 7, 9\}$. Since the 256 different input patterns consisted of the bipolar values $\{-1, +1\}$, we had $\|\mathbf{u}_i\|_2^2 = M$ and $\bar{\mu}(i) = 0.1111$ at every time instant i. We have chosen the sigmoid function (1) with $\beta = 0.4, 2, 4$. We provide plots of learning curves, given in terms of the relative system mismatch defined as $S_{\text{rel}}(i) = E[\|\tilde{\mathbf{w}}_i\|_2^2]/\|\tilde{\mathbf{w}}_{-1}\|_2^2$. The curves are averaged over 50 Monte Carlo runs in order to approximate $S_{\text{rel}}(i)$. Not depicted here are $E[\tilde{e}_a(i)]$-curves. Their behavior is very much like the system mismatch curves, however, in order to obtain smooth curves they require more averaging.

The first simulation is for $\beta = 0.4$, for which the sigmoid function operates in an almost linear range. The resulting learning curves are depicted in Fig. 3.

As expected from (19), and since the sigmoid function operates essentially in the linear region, the fastest convergence speed occurs for $\mu = 10\bar{\mu}$ $(4/\beta = 10)$, while instability occurs for values $\mu > 20\bar{\mu}$.

The next simulation shows learning curves for $\beta = 2$ (see Fig. 4). With fixed step-sizes the fastest convergence was found at $\mu = 0.6$, while for $\mu = 1.2$ the algorithm was already unstable. The bound (19) for which the largest possible step-size is given by $\mu_l = 0.4444$ is now too conservative and the proposed modifications (A), (B) and (C) lead to much faster convergence. For all methods, the step-size was chosen to be optimal (with $T = 100$ and $\epsilon = 0.02$). Since method (C) always showed the same behavior as (B) it is not depicted here. As the figure demonstrates, the first choice leads to excellent convergence, however, at the expense of calculating a logarithm at every time instant. The second choice, although not as perfect as the first one, still shows considerable improvement over the constant step-size choice.

For the third simulation $\beta = 4$. According to (19), convergence is expected for $\mu < 8\bar{\mu}/\beta = 2\bar{\mu} = 0.222\,22$. As Fig. 5 shows, for μ smaller than this bound convergence occurs. However, this bound is rather conservative and fastest convergence occurs for larger step-size values, viz., $\mu \approx 0.4$. A learning curve for $\mu = 0.8$ still shows convergence but with some stopping effect. It seems noteworthy that even very large step-sizes can still lead to convergence, although the parameter estimates seem to diverge. This effect was not observed for small β and seems to arise from the fact that the system behaves highly nonlinearly. This effect could also be observed for $\beta = 8$, where it was even stronger.

Method (B), with the optimal choice for the step-size, was applied again and showed much faster convergence than any other choice of a constant step-size. Instability occurred for approximately $2.2\mu_{\text{opt}}$.

VIII. Dynamic Neural Networks

A. Vector Notation

We show in the following sections how to extend the analysis to the recurrent neural network (RNN) case, for which we have selected Narendra and Parthasarathy's network [7] since it is very suitable for the feedback analysis of the earlier sections.

But first we introduce, for convenience of exposition, the following vector and matrix notation. Define the column vectors:

$$\mathbf{e}_{a,N}^T = [e_a(0), e_a(1), \cdots, e_a(N)] \tag{25}$$

$$\mathbf{v}_N^T = [v(0), v(1), \cdots, v(N)] \tag{26}$$

and the diagonal matrices

$$\mathbf{M}_N \triangleq \text{diag}\,\{\mu(0), \mu(1), \cdots, \mu(N)\} \tag{27}$$

$$\bar{\mathbf{M}}_N \triangleq \text{diag}\,\{\bar{\mu}(0), \bar{\mu}(1), \cdots, \bar{\mu}(N)\} \tag{28}$$

$$\mathbf{F}'_N(\boldsymbol{\eta}) \triangleq \text{diag}\,\{f'[\eta(0)], \cdots, f'[\eta(N)]\}. \tag{29}$$

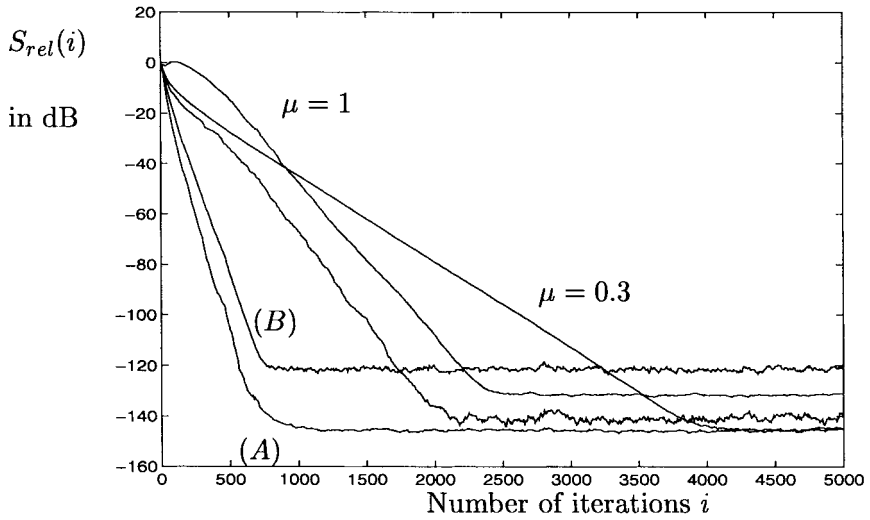

Fig. 4. Learning curves for perceptron learning algorithm with $\beta = 2$ and $\mu = 0.3, 0.6, 1$ and μ_{opt} for methods (A) and (B).

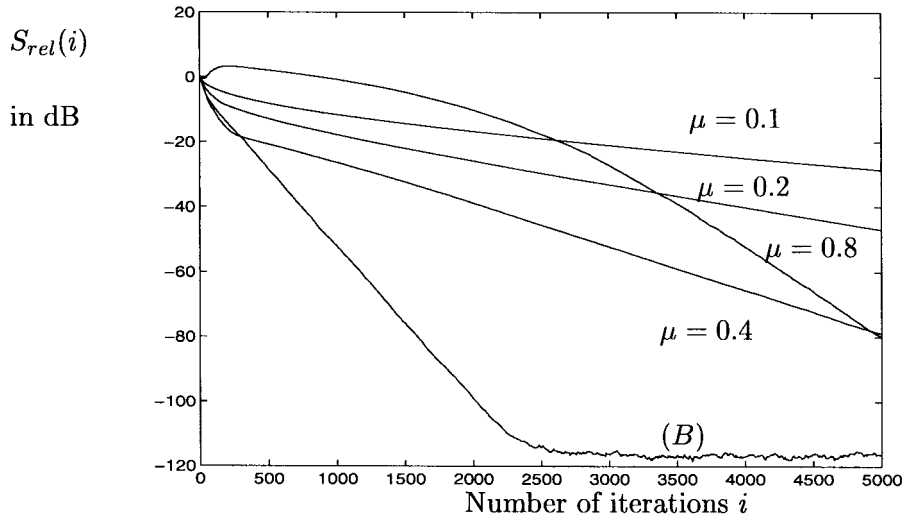

Fig. 5. Learning curves for perceptron learning algorithm with $\beta = 4$ and $\mu = 0.1, 0.2, 0.4, 0.8$ and μ_{opt} from method (B).

We write $\mathbf{F}'_N(\boldsymbol{\eta})$ with a vector argument $\boldsymbol{\eta}$ to indicate the dependence on the set $\{\eta(i)\}_{i=0}^{N}$.

It is easy to see that, due to the diagonal structure of $\mathbf{M}_N, \bar{\mathbf{M}}_N$, and $\mathbf{F}'_N(\boldsymbol{\eta})$, the two-induced norms of the matrices $[\mathbf{I} - \mathbf{M}_N\bar{\mathbf{M}}_N^{-1}\mathbf{F}'_N(\eta)]$ and $\mathbf{M}_N\bar{\mathbf{M}}_N^{-1}$ are equal to $\Delta(N)$ and $\gamma(N)$, respectively.

B. Narendra and Parthasarathy's Recurrent Network

Narendra and Parthasarathy's recurrent network is a dynamic network whose current output is also a function of earlier output values, in much the same way as the output of an IIR filter is dependent on the previous outputs [2]. Fig. 6 depicts a block diagram of a recurrent structure suggested by Narendra and Parthasarathy [2], [7], [8].

The network consists of two linear combiners with weight vectors $\mathbf{a}$ and $\mathbf{b}$. The upper combiner receives an external row input vector $\mathbf{x}_i$ and evaluates the inner product $\mathbf{x}_i\mathbf{b}$. The lower combiner receives the state vector of an FIR filter and computes its inner product with $\mathbf{a}$. The FIR filter is fed with

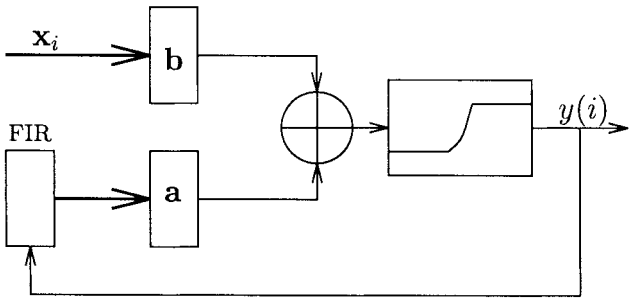

Fig. 6. Narendra and Parthasarathy's dynamic network.

the output $y(i)$ of the network and, hence, its state vector is given by

$$\mathbf{y}_{i-1} \triangleq [y(i-1) \quad y(i-2) \quad \cdots \quad y(i-M)]$$

where M is the order of the filter $\mathbf{a}$.

The weight vector of the network of Fig. 6 is defined by $\mathbf{w}^T = [\mathbf{a}^T \ \mathbf{b}^T]$. The objective of a training phase is to provide

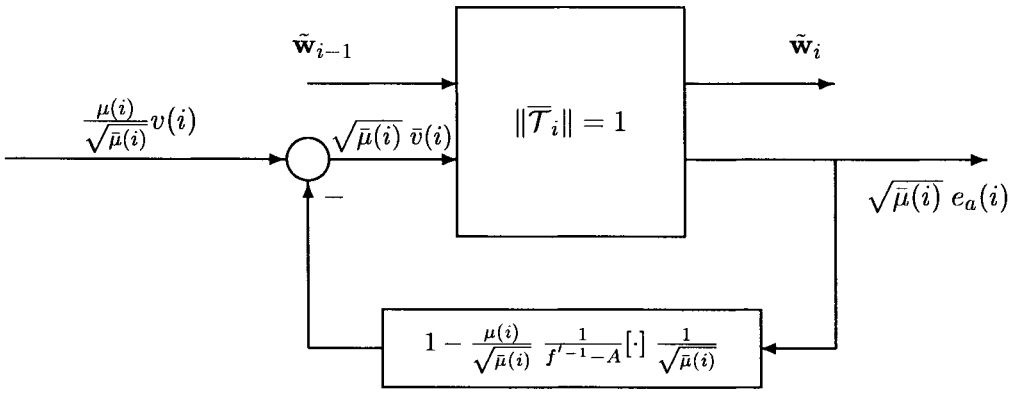

Fig. 7. Narendra and Parthasarathy's algorithm as a time-variant lossless mapping with dynamic feedback.

the network with a collection of input–output data, $\{\mathbf{x}_i, d(i)\}$, in order to estimate the unknown vectors $\mathbf{a}$ and $\mathbf{b}$. Here

$$d(i) = f[\mathbf{x}_i\mathbf{b} + \mathbf{y}_{i-1}\mathbf{a}] + v(i) \triangleq y(i) + v(i).$$

A recursive gradient-type scheme that can be used for the training of such a network is the following. Let $\mathbf{a}_{i-1}$ and $\mathbf{b}_{i-1}$ denote estimates for $\mathbf{a}$ and $\mathbf{b}$ at time $i-1$, respectively. Let also $\hat{y}(i)$ denote the corresponding output, viz.,

$$\hat{y}(i) = f[\mathbf{x}_i\mathbf{a}_{i-1} + \hat{\mathbf{y}}_{i-1}\mathbf{b}_{i-1}] = f[\mathbf{u}_i\mathbf{w}_{i-1}] \tag{30}$$

where

$$\begin{aligned}\hat{\mathbf{y}}_{i-1} &= [\hat{y}(i-1) \quad \hat{y}(i-2) \quad \cdots \quad \hat{y}(i-M)]\\ \mathbf{u}_i &= [\hat{\mathbf{y}}_{i-1} \quad \mathbf{x}_i]\\ \mathbf{w}_{i-1}^T &= [\mathbf{a}_{i-1}^T \quad \mathbf{b}_{i-1}^T].\end{aligned}$$

The estimates for $\mathbf{a}$ and $\mathbf{b}$ are recursively evaluated as follows: start with arbitrary initial conditions for $\mathbf{a}$ and $\mathbf{b}$, say an initial weight vector $\mathbf{w}_{-1}$, and use

$$\mathbf{w}_i = \mathbf{w}_{i-1} + \mu(i)\mathbf{u}_i^T(d(i) - f[\mathbf{u}_i\mathbf{w}_{i-1}]). \tag{31}$$

This can be regarded as an immediate extension of a so-called Feintuch algorithm [17] in IIR modeling (where $f[z] = z$ is linear) to the case of Fig. 6, which now includes a nonlinear activation function $f[\cdot]$. A discussion in the IIR case, with linear $f[\cdot]$, can be found in [20]. Define $e_a(i) = \mathbf{u}_i\tilde{\mathbf{w}}_{i-1}$ and $e_o(i) = y(i) - \hat{y}(i)$. Then

$$\begin{aligned}e(i) &\triangleq z(i) - \hat{z}(i) &(32)\\ &= [\mathbf{x}_i\mathbf{b} + \mathbf{y}_{i-1}\mathbf{a}] - [\mathbf{x}_i\mathbf{b}_{i-1} + \hat{\mathbf{y}}_{i-1}\mathbf{a}_{i-1}]\\ &= \mathbf{u}_i\tilde{\mathbf{w}}_{i-1} + (\mathbf{y}_{i-1} - \hat{\mathbf{y}}_{i-1})\mathbf{a}\\ &= e_a(i) + A(q^{-1})e_o(i) &(33)\end{aligned}$$

where $A(q^{-1})$ stands for the linear operator $A(q^{-1}) = \sum_{k=1}^{M} a_k q^{-k}$, and a_k are the coefficients of the FIR filter $\mathbf{a}$. By invoking the mean-value theorem we can write $e_o(i) = f'[\eta(i)]\, e(i)$, or, equivalently, $e(i) = f'^{-1}[\eta(i)]\, e_o(i)$, for some $\eta(i)$ in the interval connecting $[\mathbf{x}_i\mathbf{b} + \mathbf{y}_{i-1}\mathbf{a}]$ and $[\mathbf{x}_i\mathbf{b}_{i-1} + \hat{\mathbf{y}}_{i-1}\mathbf{a}_{i-1}]$. This allows us to conclude from (33) that

$$e_o(i) = \frac{1}{f'^{-1}[\eta(i)] - A(q^{-1})}[e_a(i)]$$

and, consequently, the update equation (31) leads to

$$\mathbf{w}_i = \mathbf{w}_{i-1} + \mu(i)\mathbf{u}_i^T\left[\frac{1}{f'^{-1}[\eta(i)] - A(q^{-1})}[e_a(i)] + v(i)\right].$$

Following the arguments of [4], we can therefore write:

$$\mathbf{w}_i = \mathbf{w}_{i-1} + \bar{\mu}(i)\mathbf{u}_i^T[e_a(i) + \bar{v}(i)] \tag{34}$$

where the modified noise sequence $\{\bar{v}(\cdot)\}$ is defined by

$$\begin{aligned}\bar{\mu}(i)\bar{v}(i) = \mu(i)v(i) &- \bar{\mu}(i)e_a(i)\\ &+ \mu(i)\frac{1}{f'^{-1}[\eta(i)] - A(q^{-1})}[e_a(i)]\end{aligned}$$

and $\bar{\mu}(i) = 1/\|\mathbf{u}_i\|^2$. This recursion is of the same form as (10). It then follows in a similar way that:

$$\|\tilde{\mathbf{w}}_i\|^2 + \bar{\mu}(i)|e_a(i)|^2 = \|\tilde{\mathbf{w}}_{i-1}\|^2 + \bar{\mu}(i)|\bar{v}(i)|^2 \tag{35}$$

which establishes that the map from $\{\tilde{\mathbf{w}}_{i-1}, \sqrt{\bar{\mu}(i)}\bar{v}(i)\}$ to $\{\tilde{\mathbf{w}}_i, \sqrt{\bar{\mu}(i)}e_a(i)\}$, denoted by $\overline{\mathcal{T}}_i$, is *lossless*, and that the overall mapping from the original disturbance $\sqrt{\bar{\mu}(\cdot)}v(\cdot)$ to the resulting *a priori* estimation error $\sqrt{\bar{\mu}(\cdot)}e_a(\cdot)$ can be expressed in terms of the feedback structure shown in Fig. 7. We remark that the notation

$$1 - \frac{\mu(i)}{\sqrt{\bar{\mu}(i)}}\frac{1}{f'^{-1}[\eta(i)] - A(q^{-1})}[\cdot]\frac{1}{\sqrt{\bar{\mu}(i)}}$$

which appears in the feedback loop, should be interpreted as follows: we first divide $\sqrt{\bar{\mu}(i)}\ e_a(i)$ by $\sqrt{\bar{\mu}(i)}$, followed by the filter $\frac{1}{f'^{-1}[\eta(i)] - A(q^{-1})}$, and then by a subsequent scaling by $\frac{\mu(i)}{\sqrt{\bar{\mu}(i)}}$.

The feedback loop now consists of a dynamic system. But we can still proceed to study the l_2-stability of the overall configuration in much the same way as we did in the former section. For this purpose, we use the vector and matrix quantities introduced in (26)–(28) and define a vector $\bar{\mathbf{v}}_N$, similar to $\mathbf{v}_N$, but with the entries $\bar{v}(\cdot)$ instead of $v(\cdot)$. We also use the diagonal matrix $\mathbf{F}'_N$ from (29), and the lower-triangular matrix $\mathbf{A}_N$ that describes the action of the FIR filter A on a sequence at its input; this is a strictly lower-triangular

Toeplitz matrix with band of width M

$$\mathbf{A}_N = \begin{bmatrix} 0 & & & \\ a_1 & 0 & & \\ a_2 & a_1 & 0 & \\ \vdots & \ddots & \ddots & \ddots \end{bmatrix}.$$

It follows from (35) that we can write:

$$\bar{\mathbf{M}}_N^{\frac{1}{2}}\bar{\mathbf{v}}_N = \mathbf{M}_N\bar{\mathbf{M}}_N^{\frac{1}{2}}\mathbf{v}_N - \left[\mathbf{I} - \bar{\mathbf{M}}_N^{-\frac{1}{2}}\mathbf{M}_N[\mathbf{F}_N'^{-1} - \mathbf{A}_N]^{-1} \times \bar{\mathbf{M}}_N^{-\frac{1}{2}}\right]\bar{\mathbf{M}}_N^{\frac{1}{2}}\mathbf{e}_{a,N}.$$

If we now define

$$\Delta(N) \triangleq \left\|\mathbf{I} - \bar{\mathbf{M}}_N^{-\frac{1}{2}}\mathbf{M}_N\left[\mathbf{F}_N'^{-1} - \mathbf{A}_N\right]^{-1}\bar{\mathbf{M}}_N^{-\frac{1}{2}}\right\|_{2,\text{ind}}$$
$$\gamma(N) \triangleq \left\|\bar{\mathbf{M}}_N^{-1}\mathbf{M}_N\right\|_{2,\text{ind}}$$

and impose the condition $\Delta(N) < 1$, we obtain that a single-neuron Narendra and Parthasarathy's network will be l_2-stable in the sense that the map from $\{\sqrt{\bar{\mu}(\cdot)}v(\cdot), \tilde{\mathbf{w}}_{-1}\}$ to $\{\sqrt{\bar{\mu}(\cdot)}e_a(\cdot)\}$ satisfies

$$\left\|\bar{\mathbf{M}}_N^{\frac{1}{2}}\mathbf{e}_{a,N}\right\| \leq \frac{\|\tilde{\mathbf{w}}_{-1}\| + \gamma(N)\left\|\bar{\mathbf{M}}_N^{\frac{1}{2}}\mathbf{v}_N\right\|}{1 - \Delta(N)}. \tag{36}$$

Moreover, the map from $\{\sqrt{\mu(\cdot)}v(\cdot), \tilde{\mathbf{w}}_{-1}\}$ to $\{\sqrt{\mu(\cdot)}e_a(\cdot)\}$ will also be l_2-stable with

$$\left\|\mathbf{M}_N^{\frac{1}{2}}\mathbf{e}_{a,N}\right\| \leq \frac{\gamma^{\frac{1}{2}}(N)\|\tilde{\mathbf{w}}_{-1}\| + \gamma(N)\left\|\mathbf{M}_N^{\frac{1}{2}}\mathbf{v}_N\right\|}{1 - \Delta(N)}.$$

The robustness (or l_2-stability) condition $\Delta(N) < 1$ corresponds to requiring the feedback matrix to be contractive, i.e.,

$$\left\|\mathbf{I} - \mathbf{M}_N\bar{\mathbf{M}}_N^{-\frac{1}{2}}\left[\mathbf{F}_N'^{-1} - \mathbf{A}_N\right]^{-1}\bar{\mathbf{M}}_N^{-\frac{1}{2}}\right\|_{2,\text{ind}} < 1. \tag{37}$$

If we limit ourselves, for simplicity, to the case of constant step-sizes μ, then a sufficient condition for (37) is to require

$$\frac{2}{\mu}\mathbf{F}_N'^{-1} - \frac{1}{\mu}\left(\mathbf{A}_N + \mathbf{A}_N^T\right) - \bar{\mathbf{M}}_N^{-1} > 0. \tag{38}$$

Let

$$\lambda = \min_i\{f'^{-1}[\eta(i)]\} \quad \zeta^{-1} = \max_i\{\bar{\mu}^{-1}(i)\}.$$

Then a sufficient condition for (38) is to require

$$\mathbf{I} - \frac{\mathbf{A}_N + \mathbf{A}_N^T}{2} > \left[\frac{\mu}{2\zeta} - (\lambda - 1)\right]\mathbf{I}$$

which in turn is satisfied if

$$\text{Re}\{1 - A(e^{j\omega})\} > \frac{\mu}{2\zeta} - (\lambda - 1), \quad \omega \in [0, 2\pi].$$

If we have an *a priori* bound on $\text{Re}\{1 - A\}$, say

$$\text{Re}\{1 - A(e^{j\omega})\} < \delta, \quad \omega \in [0, 2\pi] \tag{39}$$

then a sufficient condition for (37) to be satisfied is to choose μ such that

$$\mu < 2\zeta(\lambda + \delta - 1). \tag{40}$$

This condition has an interesting connection with the linear filtering case. For Feintuch's algorithm [17], the sign of δ is relevant to the stability of the algorithm. Here, the additional term $(\lambda - 1)$ can compensate for this effect and even negative values for δ are allowed (see the simulation examples).

For a sigmoid function $f[z]$, we know that $f'^{-1}[\eta(i)]$ lies in the range $[4/\beta, \infty)$. Therefore, in this case, a sufficient condition for (37) is given by the relation

$$\beta < \frac{8}{\frac{\mu}{\zeta} + 2(1 - \delta)}. \tag{41}$$

IX. Simulation Results for Narendra and Parthasarathy's Network

In the simulation that follows, a bipolar white random sequence with variance one has been used for the entries of the input vector $\mathbf{x}_i$. A plot of the learning curve is provided for the relative system mismatch defined as

$$S_{\text{rel}}(i) = E[\|\tilde{\mathbf{w}}_{i-1}\|^2]/\|\tilde{\mathbf{w}}_{-1}\|^2.$$

The curves are averaged over 50 Monte Carlo runs in order to approximate $S_{\text{rel}}(i)$. Similar curves can be obtained if $E[e_a^2(i)]$ is used instead.

For simulating the behavior of Narendra and Parthasarathy's network, two different sets of values has been chosen with eight input weights, one offset, and two feedback weights

$$\mathbf{w}_A = \{0.6, 0.9;\ 1, 1, 1, 1, 1, 1, 1, 1, 1\}$$

and

$$\mathbf{w}_B = \{0.9, 0.9;\ 1, 1, 1, 1, 1, 1, 1, 1, 1\}.$$

The first weight vector corresponds $\text{Re}\{1 - A\} > 0.05$, while the second weight vector corresponds to $\text{Re}\{1 - A\} > -0.0125$. Fig. 8 shows the learning curves for $\beta = 0.4$ and $\beta = 4$ when the set $\mathbf{w}_A$ was used. In order to provide a symmetrical feedback input the following sigmoid function was applied

$$f_\beta[z] = \frac{1 - \exp(-0.5\beta z)}{1 + \exp(-0.5\beta z)}$$

which has the same maximal derivative as the one defined in (1), i.e., $f'_{\max} = \beta/4$. Since the filter inputs are $\{-1, +1\}$ patterns and the output is also limited to $[-1, 1]$ we can use $\zeta = 1/11$ and according to bound (40) the training phase converges if $\mu < 1.645$ for $\beta = 0.4$ in the case of $\mathbf{w}_A$. Fig. 8 depicts two learning curves for $\mu = 0.5$ and $\mu = 1.1$ for which we found fastest convergence. Instability occurred for $\mu > 2.3$ which is in good agreement with (40). The second (non-SPR) filter showed very similar behavior. Because of the negative real part, the limit step-size is smaller. However, unstable behavior as in the Feintuch algorithm does not occur here. As a general rule of thumb one can say that the larger the β, and thus the larger the influence of the nonlinearity, the smaller the step-size that is is required for stability.

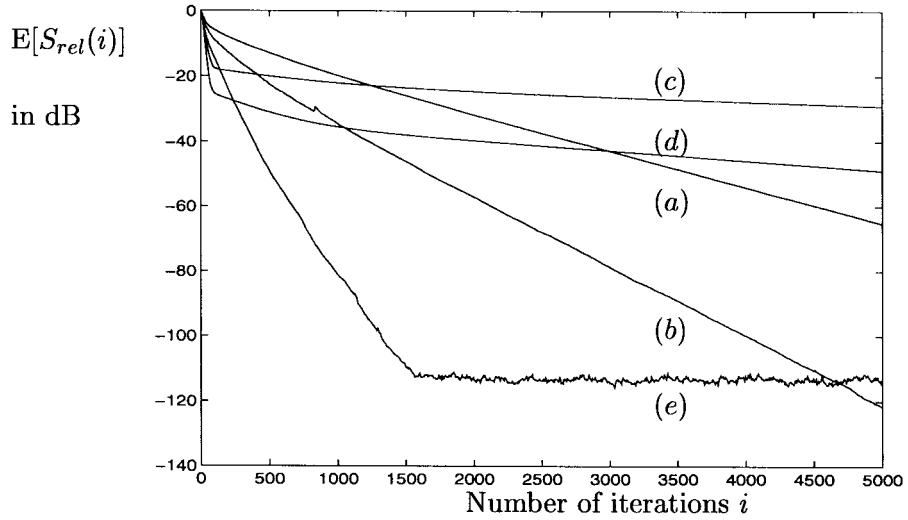

Fig. 8. Learning curves for Narendra and Parthasarathy's network (a) $\mu = 0.5,\ \beta = 0.4$, (b) $\mu = 1.1,\ \beta = 0.4$, (c) $\mu = 0.1,\ \beta = 4$, (d) $\mu = 0.2,\ \beta = 4$, (e) method (C), $\beta = 4$.

Also, for both filters, modifications as described before in Section VI can be suggested but they may or may not bring advantages in general. This is not surprising since we need to compensate the filtering effect of $A(q^{-1})$ rather than only the effect of the derivative $f'[\eta]$. However, for larger β the effect of the derivative $f'[\eta]$ becomes stronger and may exceed the filter effect. This situation is of particular interest since the network with constant step-size has very poor convergence behavior [see Fig. 8 curves (c) and (d)]. Simulations were performed with the recurrent network by employing the optimal choices of Section VI. Fig. 8 curve (e) shows that method (C) [also (A) and (B)] can be used to accelerate the training phase considerably (for large β).

X. Concluding Remarks

We have provided a time-domain feedback analysis of training algorithms for perceptron and dynamic networks with output feedback. The derivation highlights a feedback structure in terms of a lossless feedforward path and either a memoryless or a dynamic feedback loop. The interconnection is amenable to analysis to standard tools in system theory, such as the small gain theorem, and indicates choices for the step-size parameters in order to guarantee both robustness and faster convergence speed. Several simulation examples are provided to support the theoretical findings.

In the dynamic network case, certain positive-realness conditions arise in much the same way as in the study of IIR adaptive filters and identification schemes. Here, however, as indicated in the remark after (40), a less restrictive condition is tolerable in view of the presence of the nonlinear activation function.

We may also mention that the analysis in this paper has focused on the case of single neurons (perceptrons and RNN's) and has addressed questions related to the three issues of robustness, optimal step-sizes, and convergence. A local robustness analysis of the backpropagation algorithm that is employed in the training of multilayer perceptrons has been provided in [21] by invoking results from H^∞ filtering. This analysis ties up nicely with the result in [12] that establishes that the instantaneous gradient-based (LMS) adaptive filter is a minimax filter; thus highlighting its robustness properties. A related discussion on these issues can also be found in the companion paper [5].

References

[1] S. Haykin, *Neural Networks: A Comprehensive Foundation*. New York: MacMillan, 1994.
[2] D. R. Hush and B. G. Horne, "Progress in supervised neural networks," *IEEE Signal Processing Mag.*, vol. 10, no. 1, pp. 8–39, Jan. 1993.
[3] R. P. Lippmann, "An introduction to computing with neural nets," *IEEE Acoust., Speech, Signal Processing Magazine*, vol. 4, no. 2, pp. 4–22, Apr. 1987.
[4] M. Rupp and A. H. Sayed, "A time-domain feedback analysis of filtered-error adaptive gradient algorithms," *IEEE Trans. Signal Processing*, vol. 44, pp. 1428–1439, June 1996.
[5] A. H. Sayed and M. Rupp, "An l_2-stable feedback structure for nonlinear adaptive filtering and identification," to appear in *Automatica*, 1997.
[6] ——, "Error-energy bounds for adaptive gradient algorithms," *IEEE Trans. Signal Processing*, vol. 44, pp. 1982–1989, Aug. 1996.
[7] K. S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Trans. Neural Networks*, vol. 1, pp. 4–27, Mar. 1990.
[8] ——, "Gradient methods for the optimization of dynamical systems containing neural networks," *IEEE Trans. Neural Networks*, vol. 2, pp. 252–262, Mar. 1991.
[9] L. Ljung and T. Söderström, *Theory and Practice of Recursive Identification*. Cambridge, MA: MIT Press, 1983.
[10] I. D. Landau, "A feedback system approach to adaptive filtering," *IEEE Trans. Inform. Theory*, vol. IT-30, Mar. 1984.
[11] V. Solo and X. Kong. *Adaptive Signal Processing Algorithms: Stability and Performance*. Englewood Cliffs, NJ: Prentice-Hall, 1995.
[12] B. Hassibi, A. H. Sayed, and T. Kailath, "H^∞ optimality of the LMS algorithm," *IEEE Trans. Signal Processing*, vol. 44, pp. 267–280, Feb. 1996.
[13] J. J. Shynk and N. J. Bershad, "On the system identification convergence model for perceptron learning algorithms," in *Proc. Asilomar Conf. Signals, Syst., Computers*, Oct. 1994, pp. 879–886.
[14] S. Hui and S.H. Zak, "The Widrow–Hoff algorithm for McCulloch–Pitts type neurons," *IEEE Trans. Neural Networks*, vol. 5, pp. 924–929, Nov. 1994.
[15] H. K. Khalil, *Nonlinear Systems*. New York: MacMillan, 1992.
[16] M. Vidyasagar, *Nonlinear Systems Analysis*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1993.
[17] P. L. Feintuch, "An adaptive recursive LMS filter," *Proc. IEEE*, vol. 64, no. 11, pp. 1622–1624, Nov. 1976.

[18] N. J. Bershad, "Analysis of the normalized LMS algorithm with Gaussian inputs," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 793–806, Aug. 1986.
[19] M. Rupp, "The behavior of LMS and NLMS algorithms in the presence of spherically invariant processes," *IEEE Trans. Signal Processing*, vol. 41, pp. 1149–1160, Mar. 1993.
[20] M. Rupp and A. H. Sayed, "On the stability and convergence of Feintuch's algorithm for adaptive IIR filtering," in *Proc. IEEE Conf. ASSP*, Detroit, MI, May 1995.
[21] B. Hassibi, A. H. Sayed, and T. Kailath, "LMS and backpropagation are minimax filters," in *Neural Computation and Learning*, V. Roychowdhury, K. Y. Siu, and A. Orlitsky, Eds. Boston, MA: Kluwer, 1994, ch. 12, pp. 425–447.

Markus Rupp received the Diploma in electrical engineering from FHS Saarbruecken, Germany, and Universität Saarbruecken in 1984 and 1988, respectively. He received the Doctoral degree in 1993 summa cum laude at the TH Darmstadt in the field of acoustical echo cancellation. He received the DAAD Postdoctoral Fellowship and spent from November 1993 to September 1995 in the Department of Electrical and Computer Engineering of the University of California, Santa Barbara.

From 1984 to 1988 he was a Lecturer in Digital Signal Processing and High Frequency Techniques at the FHS. Since October 1995 he has been with the Lucent Technologies, Wireless Technology Research Department, Homdel, NJ, where he is currently working on new adaptive equalization schemes. His interests include algorithms for speech applications, speech enhancement and recognition, echo compensation, adaptive filter theory, H_∞-filtering, neural nets, classification algorithms, and signal detection. He has over 40 publications in the field of adaptive filters.

Ali H. Sayed (S'90–M'92) was born in São Paulo, Brazil. In 1981 he graduated in first place in the National Lebanese Baccalaureat, with the highest score in the history of the examination. In 1987 he received the bachelor's degree in electrical engineering from the University of São Paulo, and was the first-place graduate of the School of Engineering. In 1989 he received the M.S. degree with distinction in electrical engineering from the University of São Paulo. In 1992 he received the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA.

From September 1992 to August 1993 he was a Research Associate with the Information Systems Laboratory at Stanford University, after which he joined, as an Assistant Professor, the Department of Electrical and Computer Engineering at the University of California, Santa Barbara. Since July 1996, he has been an Associate Professor in the Department of Electrical Engineering at UCLA. He has more than 70 publications. His research interests are in the areas of adaptive and statistical signal processing, linear and nonlinear filtering and estimation, interplays between signal processing and control methodologies, interpolation theory, and structured computations in systems and mathematics.

Dr. Sayed is a member of SIAM and ILAS. He was awarded the Institute of Engineering Prize and the Conde Armando Alvares Penteado Prize, both in 1987 in Brazil. He was awarded an FAPESP fellowship for overseas studies in 1991 and his Ph.D. dissertation on structured algorithms in signal processing and mathematics received a special mention, among the top three Ph.D. dissertations written during the period from 1990 to 1993, for the Householder Prize in numerical algebra. He is a recipient of a 1994 NSF Research Initiation Award, and of the 1996 IEEE Donald G. Fink Prize Award. He is an Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING.