Proc. 13th IFAC World Congress, vol. 1, pp. 243–248, San Francisco, Jun. 1996.

ON THE ROBUSTNESS OF PERCEPTRON LEARNING RECURRENT NETWORKS

M. Rupp
AT&T, Wireless Technology Research Dept.
791 Holmdel-Keyport Rd.
Holmdel, NJ 07733–0400

A. H. Sayed
Dept. Electrical and Computer Eng.
University of California
Santa Barbara, CA 93106

Abstract— **This paper extends a recent time-domain feedback analysis of Perceptron learning networks to recurrent networks and provides a study of the robustness performance of the training phase in the presence of uncertainties. In particular, a bound is established on the step-size parameter in order to guarantee that the training algorithm will behave as a robust filter in the sense of $\mathbf{H}^{\infty}-$theory. The paper also establishes that the training scheme can be interpreted in terms of a feedback interconnection that consists of two major blocks: a time-variant lossless (i.e., energy preserving) feedforward block and a time-variant dynamic feedback block. The l_2-stability of the feedback structure is then analyzed by using the small-gain and the mean-value theorems.**

Keywords—**Perceptron-learning, recurrent networks, feedback structure, convergence speed, robustness, l_2–stability, $\mathbf{H}_{\infty}$-filter, small gain theorem, mean-value 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, The recursive procedure usually requires that a suitable adaptation gain be chosen and, in most cases, heuristics and trial-and-error experiences are used to select a step-size value for the training period. The "common" practice is 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.

In recent work on the robustness analysis of adaptive schemes [4], it has been shown how to select the adaptation gain in order to guarantee both 1) a robust performance in the presence of noise and modeling uncertainties, and 2) faster convergence. This was achieved by exploiting an intrinsic feedback structure and by combining tools from state-space theory, feedback analysis, and small gain analysis.

In subsequent work [5], it was shown how to extend the above results to the case of Perceptron learning, which involves a nonlinear activation function. In particular, modifications to the training algorithm, in terms of selections of the adaptation gain parameter, were suggested in order to accelerate the convergence speed during the training phase. These results are reviewed in the earlier part of this paper, followed by an extension of the analysis to the recurrent network case.

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. 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], [2], [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 Figure 1. A common choice for $f[z]$ is the sigmoid function

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

But, more generally, it can be any monotonically increasing function. The outcome $f[z]$ denotes the likelihood that the input vector belongs to $\mathcal{S}_0$ or $\mathcal{S}_1$.

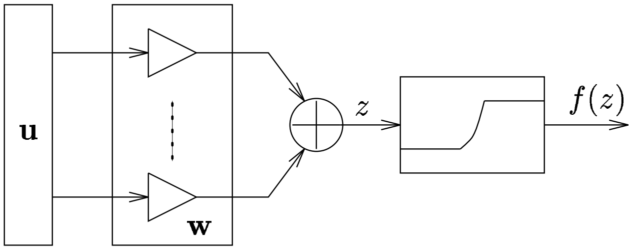

Fig. 1. *The Perceptron structure.*

This material was based on work supported in part by the National Science Foundation under Award No. MIP-9409319. The work of M. Rupp was also supported in part by a scholarship from DAAD (German Academic Exchange Service) as well as the scientific division of NATO (Wissenschaftsausschuß der NATO).

III. THE PERCEPTRON LEARNING ALGORITHM

Consider a collection of input vectors $\{\mathbf{u}_i\}$ with the corresponding (desired) output (or reference) values $\{y(i)\}$. The $\{y(i)\}$ are assumed to belong to the range of the activation function $f[\cdot]$, i.e.,

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

This is in agreement with the models and assumptions used in [7], [6]. In supervised learning, the Perceptron is presented with the given input-output data $\{\mathbf{u}_i, y(i)\}$ and the objective is to estimate $\mathbf{w}$. The PLA computes recursive estimates of $\mathbf{w}$ (with initial guess $\mathbf{w}_{-1}$):

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

For generality, the possibility of noisy perturbations in the reference signal $y(i)$ is included in our analysis. These can be due to model mismatching or to measurement noise. The perturbed references will be denoted by $\{d(i)\}$ (which are now the given data instead of $\{y(i)\}$),

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

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

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

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

The following error quantities are useful for our later analysis: $\tilde{\mathbf{w}}_i = \mathbf{w} - \mathbf{w}_i$, $e_a(i) = \mathbf{u}_i\tilde{\mathbf{w}}_{i-1} = z(i) - \hat{z}(i)$, and $e_p(i) = \mathbf{u}_i\tilde{\mathbf{w}}_i$. It then follows from (5) that:

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

Moreover, the following relation holds among $\{e_p(i), e_a(i), v(i)\}$,

$$e_p(i) = e_a(i) - \frac{\mu(i)}{\bar{\mu}(i)}\left[f[\mathbf{u}_i\mathbf{w}] - f[\mathbf{u}_i\mathbf{w}_{i-1}] + v(i)\right] , \tag{7}$$

where $\bar{\mu}(i) = 1/\|\mathbf{u}_i\|^2$ (the reciprocal of the input vector energy). Consequently, the update recursion (5) can be rewritten in the equivalent form

$$\mathbf{w}_i \quad = \quad \mathbf{w}_{i-1} + \bar{\mu}(i)\mathbf{u}_i^T[e_a(i) - e_p(i)] , \tag{8}$$

which also implies that the error vector satisfies

$$\tilde{\mathbf{w}}_i = \tilde{\mathbf{w}}_{i-1} - \bar{\mu}(i)\mathbf{u}_i^T[e_a(i) - e_p(i)]. \tag{9}$$

The robustness performance of the update recursion (5) for model (4) was studied in [5] in a purely deterministic framework and without assuming prior knowledge of noise statistics. Choices for the adaptation gains $\mu(i)$ were suggested in order to guarantee i) a robust behavior and ii) faster convergence. The robustness property loosely guarantees that "small" disturbances would lead to "small" estimation errors. That is, it guarantees that the "estimation error energy" does not exceed the "noise or disturbance energy". These facts are reviewed in the next section before considering the case of recurrent networks.

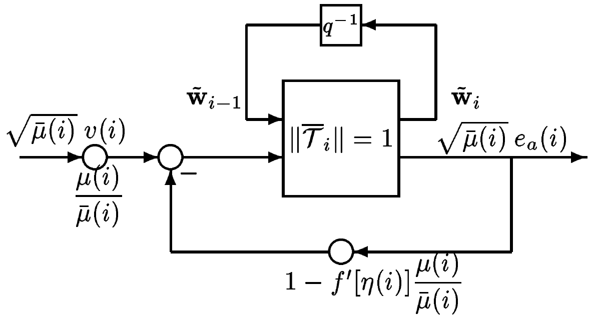
q^{-1}
$\tilde{\mathbf{w}}_{i-1}$
$\tilde{\mathbf{w}}_i$
$\sqrt{\bar{\mu}(i)}\ v(i)$
$\|\overline{\mathcal{T}}_i\| = 1$
$\sqrt{\bar{\mu}(i)}\ e_a(i)$
$\frac{\mu(i)}{\bar{\mu}(i)}$
$-$
$1 - f'[\eta(i)]\frac{\mu(i)}{\bar{\mu}(i)}$

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

IV. A FEEDBACK STRUCTURE

It has been shown in [5] that the following equality holds for all possible choices of $\mu(i)$:

$$\frac{\|\tilde{\mathbf{w}}_i\|^2 + \bar{\mu}(i)e_a^2(i)}{\|\tilde{\mathbf{w}}_{i-1}\|^2 + \bar{\mu}(i)e_p^2(i)} = 1 , \tag{10}$$

which establishes the existence of a lossless mapping $\overline{\mathcal{T}}_i$ 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 $f(z)$, and 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 $\mathbf{u}_i\mathbf{w}_{i-1}$, we obtain from expression (7) that

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

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 the feedback structure shown in Figure 2.

Define $\gamma(N) = \max_{0\le i\le N} \ \mu(i)/\bar{\mu}(i)$ and

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

Define also the column vectors

$$\begin{aligned} \mathbf{e}_{a,N}^T \quad &= \quad [e_a(0), e_a(1), ..., e_a(N)] , \\ \mathbf{v}_N^T \quad &= \quad [v(0), v(1), ..., v(N)] , \end{aligned} \tag{11}$$

and the diagonal matrices

$$\mathbf{M}_N \quad \triangleq \quad \text{diag}\ \{\mu(0), \mu(1), \ldots, \mu(N)\}, \tag{12}$$

$$\overline{\mathbf{M}}_N \quad \triangleq \quad \text{diag}\ \{\bar{\mu}(0), \bar{\mu}(1), \ldots, \bar{\mu}(N)\}, \tag{13}$$

$$\mathbf{F}'_N(\boldsymbol{\eta}) \quad \triangleq \quad \text{diag}\ \{f'[\eta(0)], ..., f'[\eta(N)]\} . \tag{14}$$

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 $\mathbf{M}_N$, $\overline{\mathbf{M}}_N$, and $\mathbf{F}'_N(\boldsymbol{\eta})$, the $2-$induced norms of the matrices $[\mathbf{I}-\mathbf{M}_N\overline{\mathbf{M}}_N^{-1}\mathbf{F}'_N(\boldsymbol{\eta})]$ and $\mathbf{M}_N\overline{\mathbf{M}}_N^{-1}$ are equal to $\Delta(N)$ and $\gamma(N)$, respectively.

Moreover, it can be shown that if $\Delta(N)<1$ then [5]

$$||\overline{\mathbf{M}}_N^{\frac{1}{2}}\mathbf{e}_{a,N}|| \leq \frac{||\tilde{\mathbf{w}}_{-1}||+||\mathbf{M}_N\overline{\mathbf{M}}_N^{-1}||_{2,ind}||\overline{\mathbf{M}}_N^{\frac{1}{2}}\mathbf{v}_N||}{1-||\mathbf{I}-\mathbf{M}_N\overline{\mathbf{M}}_N^{-1}\mathbf{F}'_N(\boldsymbol{\eta})||_{2,ind}}.$$

This expression 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 small gain theorem [8], and can be seen to be equivalent to requiring that $\mu(i)$ be chosen such that

$$0<\mu(i)f'[\eta(i)]<2/||\mathbf{u}_i||^2\ . \tag{15}$$

V. OPTIMAL CHOICES OF STEP-SIZES

It can also be argued, by ignoring the measurement noise $v(i)$, that if $\mu(i)$ is chosen in the middle of the interval specified by (15), say $\mu_{opt}(i)f'[\eta(i)]=\bar{\mu}(i)$, then the feedback loop is disconnected and the convergence speed is faster. In this case, there will be no energy flowing back into the lower input of the lossless section.

But $\eta(i)$ is still unknown and therefore three suitable approximations for $\mu_{opt}(i)$ have been suggested in [5]:

- Choice A: $\mu_{opt}(i)=$

$$\bar{\mu}(i)\min\left(-\frac{1/\beta\ln[1/d(i)-1]+\mathbf{u}_i\mathbf{w}_{i-1}}{d(i)-f[\mathbf{u}_i\mathbf{w}_{i-1}]},T\right),$$

where T is used as a threshold value in order to prevent large step-sizes.

- Choice B: For $\left(d(i)-\frac{1}{2}\right)\left(f(\mathbf{u}_i\mathbf{w}_{i-1})-\frac{1}{2}\right)>0$ we set

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

otherwise $\mu_{opt}(i)=\bar{\mu}(i)/f'_{\max}$.

- Choice C:

$$\mu_{opt}(i)=\frac{\bar{\mu}(i)}{\beta\left[\hat{f}[\eta(i)](1-\hat{f}[\eta(i)])\right]+\epsilon}, \tag{16}$$

where ϵ is a small positive constant.

VI. RECURRENT NETWORKS

The results of the earlier sections can be extended to the case of Recurrent Neural Networks (RNN, for short). An RNN 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. Figure 3 depicts a block diagram of a recurrent structure suggested by Narendra [9], [10].

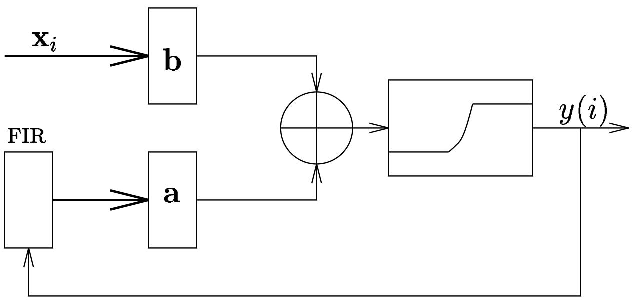

Fig. 3. *Narendra's Dynamic Network.*

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

$$\mathbf{y}_{i-1}\overset{\Delta}{=}\left[\begin{array}{cccc} y(i-1) & y(i-2) & \ldots & y(i-M)\end{array}\right],$$

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

The weight vector of the network of Figure 3 is defined by $\mathbf{w}^T=\left[\begin{array}{cc}\mathbf{a}^T & \mathbf{b}^T\end{array}\right]$. The objective of a training phase is to provide 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,

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

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., $\hat{y}(i)=f[\mathbf{x}_i\mathbf{a}_{i-1}+\hat{\mathbf{y}}_{i-1}\mathbf{b}_{i-1}]=f[\mathbf{u}_i\mathbf{w}_{i-1}]$, where $\hat{\mathbf{y}}_{i-1}=\left[\begin{array}{cccc}\hat{y}(i-1) & \hat{y}(i-2) & \ldots & \hat{y}(i-M)\end{array}\right]$, $\mathbf{u}_i=\left[\begin{array}{cc}\hat{\mathbf{y}}_{i-1} & \mathbf{x}_i\end{array}\right]$, and $\mathbf{w}_{i-1}^T=\left[\begin{array}{cc}\mathbf{a}_{i-1}^T & \mathbf{b}_{i-1}^T\end{array}\right]$. 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

$$\mathbf{w}_i=\mathbf{w}_{i-1}+\mu(i)\mathbf{u}_i^T\left[d(i)-f[\mathbf{u}_i\mathbf{w}_{i-1}]\right]. \tag{17}$$

This can be regarded as an immediate extension of a so-called Feintuch algorithm[11] in IIR modeling (where $f(z)=z$ is linear) to the case of Figure 3, which now includes a nonlinear activation function $f[\cdot]$. A discussion in the IIR case, with linear $f[\cdot]$, can be found in [12]. Define $e_a(i)=\mathbf{u}_i\tilde{\mathbf{w}}_{i-1}$, and $e_o(i)=y(i)-\hat{y}(i)$. Then

$$\begin{aligned} e(i) &\overset{\Delta}{=} z(i)-\hat{z}(i) && (18)\\ &= [\mathbf{x}_i\mathbf{b}+\mathbf{y}_{i-1}\mathbf{a}]-[\mathbf{x}_i\mathbf{b}_{i-1}+\hat{\mathbf{y}}_{i-1}\mathbf{a}_{i-1}], \\ &= \mathbf{u}_i\tilde{\mathbf{w}}_{i-1}+(\mathbf{y}_{i-1}-\hat{\mathbf{y}}_{i-1})\mathbf{a}, \\ &= e_a(i)+A(q^{-1})e_o(i)\ , && (19)\end{aligned}$$

where $A(q^{-1})$ stands for the linear operator $A(q^{-1})=\sum_{k=1}^{M}a_kq^{-k}$, 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 (19) that

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

and, consequently, the update equation (17) leads to

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

Following the arguments of [5], we can therefore write

$$\mathbf{w}_i \;=\; \mathbf{w}_{i-1}+\bar{\mu}(i)\mathbf{u}_i^T[e_a(i)+\bar{v}(i)]\ , \qquad (20)$$

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

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

and $\bar{\mu}(i)=1/\|\mathbf{u}_i\|^2$. This recursion is of the same form as (8). It then follows in a similar way that

$$\|\tilde{\mathbf{w}}_i\|^2+\bar{\mu}(i)\,|e_a(i)|^2=\|\tilde{\mathbf{w}}_{i-1}\|^2+\bar{\mu}(i)\,|\bar{v}(i)|^2\ , \qquad (21)$$

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 Figure 4. 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 (11)–(13) 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 (14), 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 (21) that we can write

$$\begin{aligned}\overline{\mathbf{M}}_N^{\frac{1}{2}}\bar{\mathbf{v}}_N &= \mathbf{M}_N\overline{\mathbf{M}}_N^{\frac{1}{2}}\mathbf{v}_N\\ &\quad -\Big[\mathbf{I}-\overline{\mathbf{M}}_N^{-\frac{1}{2}}\mathbf{M}_N[\mathbf{F}_N'^{-1}-\mathbf{A}_N]^{-1}\overline{\mathbf{M}}_N^{-\frac{1}{2}}\Big]\,\overline{\mathbf{M}}_N^{\frac{1}{2}}\mathbf{e}_{a,N}.\end{aligned}$$

If we now define

$$\begin{aligned}\Delta(N) &\triangleq \|\mathbf{I}-\overline{\mathbf{M}}_N^{-\frac{1}{2}}\mathbf{M}_N[\mathbf{F}_N'^{-1}-\mathbf{A}_N]^{-1}\overline{\mathbf{M}}_N^{-\frac{1}{2}}\|_{2,ind}\\ \gamma(N) &\triangleq \|\overline{\mathbf{M}}_N^{-1}\mathbf{M}_N\|_{2,ind}\ ,\end{aligned}$$

and impose the condition $\Delta(N)\;<\;1$, we obtain that a single-neuron Narendra 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:

$$\|\overline{\mathbf{M}}_N^{\frac{1}{2}}\mathbf{e}_{a,N}\|\;\leq\;\frac{\|\tilde{\mathbf{w}}_{-1}\|+\gamma(N)\|\overline{\mathbf{M}}_N^{\frac{1}{2}}\mathbf{v}_N\|}{1-\Delta(N)}. \qquad (22)$$

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

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

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

$$\left\|\mathbf{I}-\mathbf{M}_N\overline{\mathbf{M}}_N^{-\frac{1}{2}}[\mathbf{F}_N'^{-1}-\mathbf{A}_N]^{-1}\overline{\mathbf{M}}_N^{-\frac{1}{2}}\right\|_{2,ind}<1. \qquad (23)$$

If we limit ourselves, for simplicity, to the case of constant step-sizes μ, then a sufficient condition for (23) is to require:

$$\frac{2}{\mu}\mathbf{F}_N'^{-1}-\frac{1}{\mu}(\mathbf{A}_N+\mathbf{A}_N^T)-\overline{\mathbf{M}}_N^{-1}\;>\;0\,. \qquad (24)$$

Let

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

Then a sufficient condition for (24) 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

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

If we have an a-priori bound on $\mathrm{Re}(1-A)$, say

$$\mathrm{Re}\left[1-A(e^{j\omega})\right]\;<\delta,\;\;\omega\in[0,2\pi] \qquad (25)$$

then a sufficient condition for (23) to be satisfied is to choose μ such that

$$\mu\;<\;2\zeta\,(\lambda+\delta-1)\,. \qquad (26)$$

This condition has an interesting connection with the linear filtering case. For Feintuch's algorithm [11], 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 also 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 (23) is given by the relation

$$\beta<\frac{8}{\frac{\mu}{\zeta}+2(1-\delta)}. \qquad (27)$$

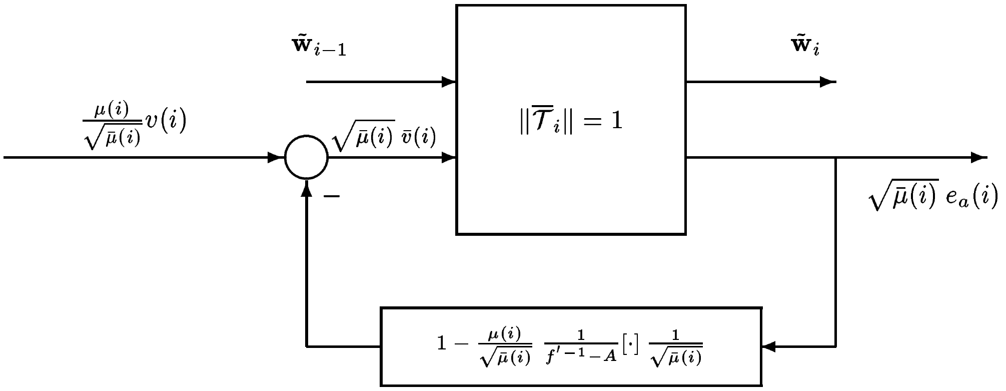
$\tilde{\mathbf{w}}_{i-1}$
$\tilde{\mathbf{w}}_i$
$\frac{\mu(i)}{\sqrt{\bar{\mu}(i)}}v(i)$
$\sqrt{\bar{\mu}(i)}\ \bar{v}(i)$
$\|\overline{\mathcal{T}}_i\| = 1$
$-$
$\sqrt{\bar{\mu}(i)}\ e_a(i)$
$1 - \frac{\mu(i)}{\sqrt{\bar{\mu}(i)}}\ \frac{1}{f'^{-1} - A}[\cdot]\ \frac{1}{\sqrt{\bar{\mu}(i)}}$

Fig. 4. *Narendra's algorithm as a time-variant lossless mapping with dynamic feedback.*

VII. Simulation Results

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

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

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

For simulating the behaviour of Narendra'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 Real$(1 - A) > 0.05$, while the second weight vector corresponds to Real$(1 - A) > -0.0125$. Figure 5 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 (26) the training phase converges if $\mu < 1.645$ for $\beta = 0.4$ in the case of $\mathbf{w}_A$. Figure 5 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 (26). The second (non-SPR) filter showed very similar behavior. Because of the negative real part, the limit step-size is smaller. However, instable behaviour as in the Feintuch algorithm does not occur here.

Also, for both filters, modifications as described before in Sec. V 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 behaviour (see (d) and (e)). Simulations were performed with the recurrent network by employing the optimal choices of Sec. V. Curve (e) shows that method (C) (also (A) and (B)) can be used to accelerate the training phase considerably (for large β).

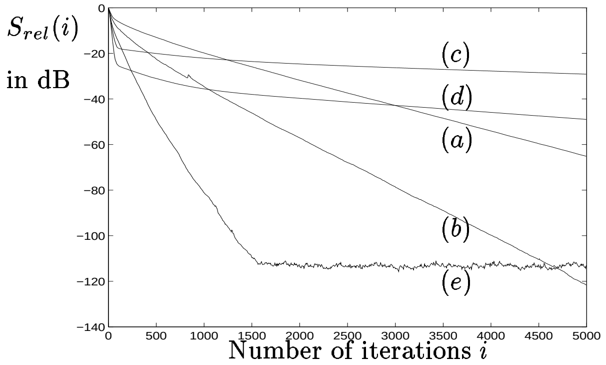

Fig. 5. *Learning curves for Narendra's network (a)$\mu = 0.5, \beta = 0.4$,(b)$\mu = 1.1, \beta = 0.4$,(c)$\mu = 0.1, \beta = 4$,(d) $\mu = 0.2, \beta = 4$, (e) method (C), $\beta = 4$.*

References

[1] Lippmann, R.P. (1987). An introduction to computing with neural nets. *IEEE Acoustics, Speech and Signal Processing Mag.* **4**(2), 4-22.

[2] Hush,D.R. and B.G. Horne. (1993). Progress in supervised neural networks. *IEEE Signal Processing Magazine* **10**(1), 8-93.

[3] Haykin, S. (1994). *Neural Networks: A Comprehensive Foundation.* MacMillan.

[4] Sayed, A.H. and M. Rupp. (1995*a*). A time-domain feedback analysis of adaptive gradient algorithms via the Small Gain Theorem. *Proc. SPIE Conference on Advanced Signal Processing: Algorithms, Architectures, and Implementations* , vol. 2563, pp. 458–469, San Diego, CA, July 1995.

[5] Sayed, A.H. and M. Rupp. (1995*b*). A feedback analysis of Perceptron learning for neural networks. *Proc. 29th Asilomar Conference on Signals, Systems, and Computers*, Pacific Grove, CA, Oct. 1995.

[6] Shynk, J.J. and N.J. Bershad. (1994). On the system identification convergence model for perceptron learning algorithms. *Proc. of Asilomar Conference on Signals, Systems, and Computers,* 879-886.

[7] Hui, S. and S.H. Zak. (1994). The Widrow-Hoff algorithm for McCulloch-Pitts type neurons. *IEEE Trans. on Neural Networks.* **5**(6), 924-929.

[8] Khalil, H. K. (1992). *Nonlinear Systems.* MacMillan.

[9] Narendra, K.S. and K. Parthasarathy. (1990). Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks.* **1**(1), 4-27.

[10] Narendra, K.S. and K. Parthasarathy. (1991). Gradient methods for the optimization of dynamical systems containing neural networks. *IEEE Transactions on Neural Networks.* **2**, 252-262.

[11] Feintuch, P. L. (1976). An adaptive recursive LMS filter. *Proc. IEEE* **64**(11), 1622–1624.

[12] Rupp, M. and A. H. Sayed (1995*c*). On the stability and convergence of Feintuch's algorithm for adaptive IIR filtering. *Proc. IEEE ICASSP.* Detroit, MI. 1388-1391.