ON THE ADAPTATION ALGORITHMS FOR GENERALIZED PERCEPTRONS

Stanislaw H. Żak
School of Electrical Engineering
Purdue University
West Lafayette, IN 47907
U.S.A.

Hebertt J. Sira-Ramirez
Departamento Sistemas de Control
Escuela de Ingenieria de Sistemas
Universidad de Los Andes
Merida-VENEZUELA

ABSTRACT

In this paper new adaptation algorithms are proposed for a class of feedforward neural networks with differentiable and nondifferentiable nonlinearities. The class of neural networks considered in this paper can be viewed as generalized single and multi-layered perceptrons. The learning parameters in the proposed algorithms are adjusted to force the error between the actual and desired outputs to satisfy a stable difference equation.

1. INTRODUCTION

An artificial neural network is a large-scale nonlinear circuit of interconnected simple circuits called nodes or neurons. These networks resemble patterns of the biological neural networks hence the term artificial neural networks.

In this paper our interest is in the class of feedforward neural networks which can be viewed as generalized perceptrons.

The development of the perceptron can be traced back to the early days of pattern recognition (See [2-5,7] for more details.) Its application as an adaptive system to the control of many degrees of freedom robotic manipulators was reported by Albus [1] in 1975. More recently, Widrow and Winter [7], discussed numerous applications of perceptrons for adaptive filtering adaptive pattern recognition, and adaptive signal processing.

The central role in advancing the practicality of perceptrons, and neural networks in general, are played by adaptation algorithms. In the case of the single perceptron one of the most well known algorithm that minimizes the mean square error between the desired output and the actual output is due to Widrow and Hoff. For the layered perceptron the central role is played by the back-propagation algorithm (see [5] for the derivation of this algorithm). One of the drawbacks of the back-propagation algorithm is the requirement that the nonlinear activation functions be differentiable.

In this paper we propose a new class of adaptation, or training, algorithms for generalized single and two-layer perceptrons. Our proposed algorithms, unlike

This work was supported by the Consejo de Desarrollo Cientifico, Humanistico y Tecnologico (CDCHT) of the Universidad de Los Andes (Merida-VENEZUELA) under Research Grant I-325-90, by the Consejo Nacional de Investigaciones Cientificas y Tecnologicas (CONICIT) of VENEZUELA and by the School of Electrical Engineering of Purdue University (West Lafayette, IN).

the back-propagation, do not require differentiability along the network's signal paths. On the contrary, we include activation functions which are not only nondifferentiable (saturation functions) but also discontinuous like hard limiters. Another feature of the training algorithms proposed in this paper is that the learning parameters are adjusted in such a way so that the error between the actual and desired outputs satisfies a stable difference equation. This is also characteristic of the celebrated Widrow-Hoff algorithm for single-layer perceptrons.

The paper is organized as follows. In the next Section we briefly review the Widrow-Hoff adaptation rule. This rule constitutes a nice starting point in our development. In Section 3, after introducing some notation and definitions, we propose a new adaptation algorithm for the single layer perceptron. The proposed algorithm is a generalization of the Widrow-Hoff adaptation rule. In Section 4 we present a new training algorithm for generalized two-layered perceptrons. The conclusions of the paper are found in Section 5.

2. BRIEF REVIEW OF THE WIDROW-HOFF ADAPTATION RULE

The single layer perceptron as an adaptive threshold element is shown in Fig. 1.

One can use the Widrow-Hoff delta rule (see [7] for its discussion) to adjust the weights ω_i $(i = 1,2,...,n)$. The algorithm can be written as follows:

$$W(k+1) = W(k) + \frac{\alpha e(k) X}{X^T X}, \quad X \neq 0 \tag{2.1}$$

where

k = the time index or the adaption cycle number,

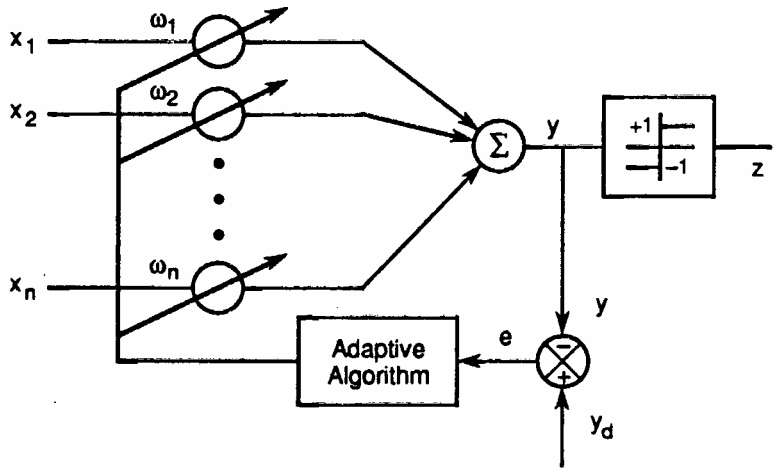

Fig. 1. Single layer perceptron

$W(k) = [\omega_1(k),...,\omega_n(k)]^T$ is the value, at time k, of the weight vector,
$X = [x_1,...,x_n]^T$ is the present input pattern,
$e(k) = y_d - y(k)$ is the present error,

α = the reduction factor whose practical range is (0.1, 1.0).

After some manipulations, one can conclude that the error is reduced by a factor of α at each new learning iteration as the weights are changed while holding the input pattern X fixed. More specifically, the error obeys the following difference equation

$$e(k+1) = (1-\alpha)\, e(k)\,. \tag{2.2}$$

As one can see from the above equation, the choice of α controls the speed of convergence towards zero of the learning error signal e(•).

After discussing the Widrow-Hoff delta rule, we can now propose new adaptation algorithm for the single perceptron shown in Fig. 1.

3. NEW ADAPTATION RULES FOR A SINGLE PERCEPTRON

Before presenting new training algorithms for the single perceptron we shall introduce some notation and definitions.

Let $\Gamma: \mathbb{R}^\ell \to \mathbb{R}^\ell$ denotes a nonlinear operator with the following property

$$\boxed{\Gamma[-X] = -\Gamma[X]} \tag{3.1}$$

The operator Γ can have, in particular, the following form

$$\Gamma = \begin{bmatrix} \text{sgn} & & 0 \\ & \ddots & \\ 0 & & \text{sgn} \end{bmatrix} \quad \text{where} \quad \text{sgn } z_i = \begin{cases} 1 & \text{if} \quad z_i > 0 \\ -1 & \text{if} \quad z_i < 0\,. \end{cases}$$

Another possible form of the operator Γ is

$$\Gamma = \begin{bmatrix} \text{sat} & & 0 \\ & \ddots & \\ 0 & & \text{sat} \end{bmatrix} \quad \text{where} \quad \text{sat } z_i = \begin{cases} +1 & \text{if} \quad z_i > 1 \\ z_i & \text{if} \quad z_i \in [-1,1] \\ -1 & \text{if} \quad z_i < 1\,, \end{cases}$$

or

$$\Gamma = \begin{bmatrix} \text{sig} & & 0 \\ & \ddots & \\ 0 & & \text{sig} \end{bmatrix} \quad \text{where} \quad \text{sig } z_i = \frac{1-e^{-z_i}}{1+e^{-z_i}}\,.$$

Observe that Γ does not have to be a diagonal operator. However, we require that Γ is an odd operator, that is, it satisfies (3.1). Notice that Γ can also be an identity operator. Armed with the above notation we can now present the new algorithm in the following theorem.

Theorem 1

If the weights ω_i of the single perceptron, shown in Fig. 1, are adapted according to the rule

$$W(k+1) = W(k) + \frac{\alpha\, e(k)\, \Gamma[X]}{X^T \Gamma[X]} \qquad (X^T \Gamma[X] \neq 0)\,, \tag{3.2}$$

with $0 < \alpha < 2$, (practical range for α is (0.1, 1.0)) then the error e(k) tends asymptotically to zero with the rate of convergence $(1 - \alpha)$.

Proof: Note that

$$e(k+1) - e(k) = y_d - y(k+1) - [y_d - y(k)] = -\sum_{i=1}^{n} [\omega_i(k+1) - \omega_i(k)]\, x_i$$
$$= -X^T [W(k+1) - W(k)]\,.$$

We can now use the proposed update rule to obtain:

$$e(k+1) - e(k) = -X^T \frac{\alpha\, e(k)\, \Gamma[X]}{X^T\, \Gamma[X]} = -\alpha\, e(k) \quad \text{if } X^T \Gamma[X] \neq 0\,.$$

Hence $e(k+1) = (1 - \alpha)\, e(k)$. Thus, if $0 < \alpha < 2$ then $\lim_{k \to \infty} e(k) = 0$.

□

Note that in the new adaptation algorithm, as well as in the Widrow-Hoff algorithm, the error is reduced by a factor of α.

Observe that if Γ is the identity operator then the new algorithm (3.2) is the same as the Widrow-Hoff adaptation rule. If on the other hand,

$$\Gamma[X] = [\text{sgn } x_1,\ \text{sgn } x_2,\ \ldots,\ \text{sgn } x_n]^T \triangleq \text{SGN } X\,,$$

then (3.2) is the same as the algorithm proposed in [6].

We shall next present new adaptation algorithms for the multi-layer perceptrons. The proposed algorithms are an extension of the training algorithms presented in Sira-Ramirez and Żak [6].

4. ADAPTATION ALGORITHMS FOR TWO-LAYER GENERALIZED PERCEPTRONS

In this Section we will be concerned with generalized two-layer perceptron which is a feedforward network with one layer of nodes between the input and output nodes. One of the best known training algorithm for multi-layer perceptrons is the back-propagation algorithm. A disadvantage of the back-propagation algorithm is the inherent requirement of continuous differentiability of the nonlinearities. Our proposed class of algorithms, unlike the back-propagation does not require differentiability along the network's signal paths. On the contrary we include activation functions which are of the hard limiter type, saturation, or any other nonlinearity which satisfies (3.1). In the back-propagation algorithm the learning parameters are adjusted using the gradient descent method. In our new adaptation algorithms the learning parameters are adjusted to force the error between the actual and desired outputs to satisfy a stable difference error equation, rather than to minimize an error function. This approach allows one to

better control the stability and speed of convergence by appropriate choice of parameters of the error difference equation.

We shall start our analysis by considering a two-layer adaptive neural network depicted in Fig. 2. Its schematic representation is shown in Fig. 3. The

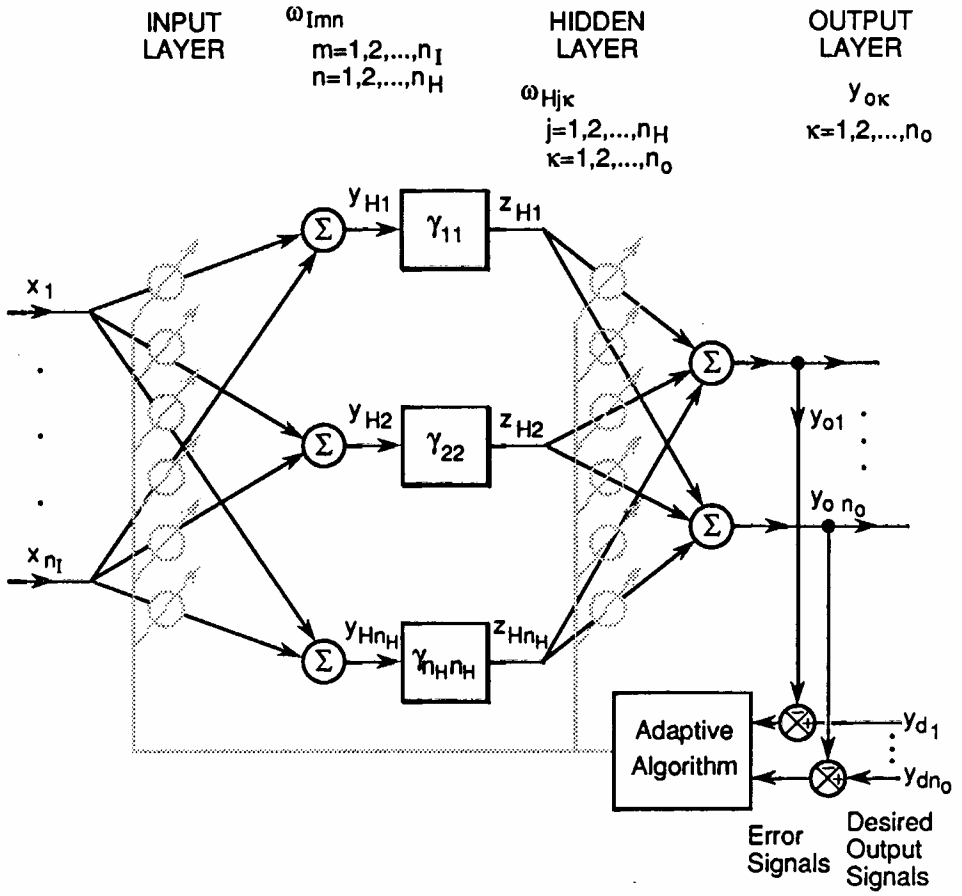

Fig. 2. A two-layer adaptive neural network with a diagonal nonlinear operator Γ in the hidden layer.

vector $Y_o(k)$ of the output components $y_{o\kappa}(k)$ can be represented as

$$\boxed{Y_o(k) = [W_H(k)]^T Z_H(k)}$$

where $W_H(k) \in \mathbb{R}^{n_H \times n_o}$, and $Z_H(k) = \Gamma[Y_H(k)]$. Similarly the vector $Y_H(k)$ of the components $y_{Hn}(k)$ is

$$\boxed{Y_H(k) = [W_I(k)]^T X}$$

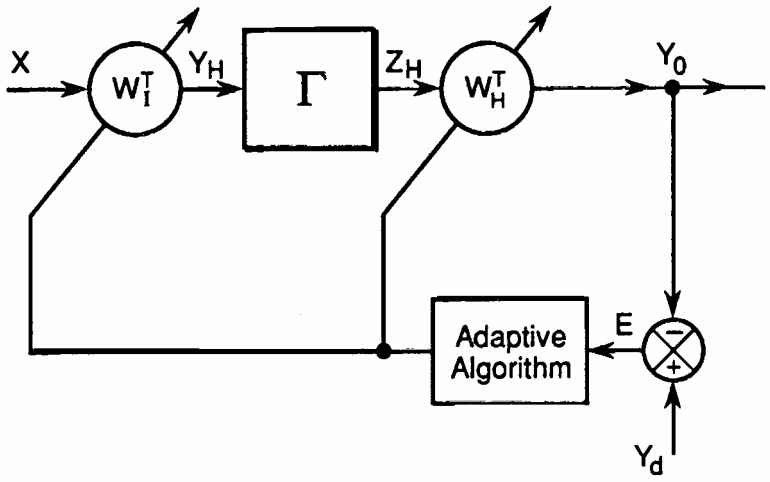

Fig. 3. Schematic representation of the two-layer generalized perceptron.

where $W_I(k) \in \mathbb{R}^{n_I \times n_H}$ and $X = [x_1, x_2, ..., x_{n_I}]^T$. In a schematic representation of the two-layer adaptive generalized perceptron depicted in Fig. 3., the nonlinear operator Γ does not have to be a diagonal one. However, it must satisfy (3.1).

Let Y_d denotes the desired output vector with component. $y_{d\kappa}$, $\kappa = 1, 2, ..., n_o$. The error vector $E(k)$ at time k is

$$E(k) = [e_1(k), \ ..., \ e_{n_o}(k)] = Y_d - Y_o(k) \ .$$

The weights updates are represented by the following equations:

$$W_H(k+1) = W_H(k) + U_H(k) \ , \quad W_I(k+1) = W_I(k) + U_I(k) \ ,$$

where $U_H(k) \in R^{n_H \times n_o}$ and $U_I(k) \in R^{n_I \times n_H}$ are the correction matrices.

The following lemma will be needed in subsequent considerations.

Lemma

The error vector $E(k)$ satisfies the following difference equation as a function of the input layer and hidden layer matrix update weights $U_H(k)$ and $U_I(k)$:

$$\begin{aligned} E(k+1) - E(k) = [W_H(k)]^T \{\Gamma[Y_H(k)] - \Gamma[Y_H(k) + [U_I(k)]^T X]\} \\ - [U_H(k)]^T \Gamma[Y_H(k) + [U_I(k)]^T X] \end{aligned} \tag{4.1}$$

Proof: We have

$$
\begin{aligned}
E(k+1) - E(k) &= Y_o(k) - Y_o(k+1) \\
&= [W_H(k)]^T Z_H(k) - [W_H(k) + U_H(k))]^T Z_H(k+1) \\
&= [W_H(k)]^T [Z_H(k) - Z_H(k+1)] - [U_H(k)]^T Z_H(k+1) \\
&= [W_H(k)]^T \{\Gamma[Y_H(k)] - \Gamma[Y_H(k+1)]\} - [U_H(k)]^T \Gamma[Y_H(k+1)] \, .
\end{aligned}
$$

Since

$$
Y_H(k+1) = [W_I(k+1)]^T X = [W_I(k) + U_I(k)]^T X = Y_H(k) + [U_I(k)]^T X
$$

we have that

$$
\begin{aligned}
E(k+1) - E(k) = [W_H(k)]^T &\{\Gamma[Y_H(k)] - \Gamma[Y_H(k) + [U_I(k)]^T X]\} \\
&- [U_H(k)]^T \Gamma[Y_H(k) + [U_I(k)]^T X]
\end{aligned}
$$

□

The new training algorithm for the generalized two-layer perceptron is presented in the following theorem.

Theorem 2

If the weight correction matrices $U_I(k)$ and $U_H(k)$ are respectively chosen as

$$
U_I(k) = \frac{-2\Gamma_1[X][Y_H(k)]^T}{X^T \Gamma_1[X]} \, , \quad (X^T \Gamma_1[X] \neq 0) \, , \tag{4.2}
$$

and

$$
U_H(k) = -2 \, W_H(k) - \frac{\Gamma_2[Z_H(k)] \, \{AE(k)\}^T}{[Z_H(k)]^T \Gamma_2[Z_H(k)]} \, , \quad (Z_H^T \Gamma_2[Z_H] \neq 0) \, , \tag{4.3}
$$

where Γ_1 and Γ_2 are operators, then the learning error vector $E(k)$ satisfies the following asymptotically stable difference equation:

$$
E(k+1) = (I - A) \, E(k) \, , \tag{4.4}
$$

where A is an $n_o \times n_o$ diagonal matrix given by

$$
A = \text{diag}\{\alpha_1, ..., \alpha_{n_o}\} \tag{4.5}
$$

such that $|1 - \alpha_\kappa| < 1$, $\kappa = 1,2,...n_o$. (Practical range for α_k is (0.1, 1.0))

Proof: Note that the transpose of the weight correction matrix $U_I(k)$ is given by

$$
[U_I(k)]^T = \frac{-2Y_H(k)\{\Gamma_1[X]\}^T}{X^T \Gamma_1[X]} \, .
$$

Substituting this last expression into the error difference equation (4.1), of the Lemma, and taking into account (3.1) one obtains

$$E(k+1) - E(k) = 2[W_H(k)]^T \Gamma[Y_H(k)] + [U_H(k)]^T \Gamma[Y_H(k)]$$
$$= 2[W_H(k)]^T Z_H(k) + [U_H(k)]^T Z_H(k)$$
$$= [2\, W_H(k) + U_H(k)]^T Z_H(k)\,.$$

Substituting (4.3) into the above error vector difference equation yields the following asymptotically stable error dynamics

$$E(k+1) = (I - A)\, E(k)\,.$$

□

Notice that A may also be chosen as an arbitrary nondiagonal matrix such that the matrix (I-A) has its eigenvalues in the open unit circle of the complex plane. A particulary simple form of (4.2) and (4.3) is obtained when Γ_1 and Γ_2 are chosen to be the identity operators.

The above approach can be used to formulate adaptation algorithms for three-layered feedforward networks.

5. CONCLUSIONS

In this paper training procedures have been presented for a class of feedforward neural networks. The class of neural networks we have considered can be viewed as generalized perceptrons. The learning parameters are adjusted in such a way so that the error between the desired and actual outputs satisfies a stable difference equation. Research is now underway to apply the proposed adaptation algorithms to the control of nonlinear dynamic processes.

REFERENCES

1. Albus, J. S. (1975). A new approach to manipulator control: The Cerebellar Model Articulation Controller (CMAC), *Transactions of the ASME, J. Dynamic Systems, Measurement, and Control,* **97**, Series G, No. 3, pp. 220-227.
2. Haykin, S. (1984). *Introduction to adaptive filters,* Macmillan Publishing Co., New York.
3. Lippmann, R. P. (1987). An introduction to computing with neural nets, *IEEE ASSP Magazine,* **4**, No. 2, pp. 4-22.
4. Pao, Y. H. (1989). *Adaptive pattern recognition and neural networks,* Addison-Wesley, Reading, Mass.
5. Rumelhart, D. E. McClelland, J. L. and the PDP Research Group (1986). *Parallel distributed processing: Explorations in the microstructure of cognition, Vol. 1: Foundations.* "A Bradford book", The MIT Press, Cambridge, Mass.
6. Sira-Ramirez, H. J. and Żak, S. H. On the adaptation of single and multilayer perceptrons, submitted for publication.
7. Widrow, B. and Winter, R. (1988). Neural nets for adaptive filtering and adaptive pattern recognition, *Computer,* **21**, No. 3, pp. 25-39.