A GRADIENT ADAPTIVE STEP SIZE ALGORITHM FOR IIR FILTERS

Christos G. Boukis, Danilo P. Mandic, Eftychios V. Papoulis, and Anthony G. Constantinides

Communications and Signal Processing Group, Imperial College, UK.
Email:{christos.boukis, d.mandic, eftychios.papoulis, agc}@ic.ac.uk

ABSTRACT

The Output Error method, a fundamental technique for the updating of the coefficients of an adaptive IIR filter, is modified by introducing a time varying step size. The adaptation of this term is based on a gradient descent technique. This scheme can be considered as an extension of the algorithms presented in [1] and [2] to IIR filters. The novel algorithm does not require any *a priori* knowledge of the statistical characteristics of the input signal and the unknown channel, since its step size converges automatically to its optimal value. This algorithm has the ability to converge in time-varying environments, which makes it suitable for processing of nonstationary signals.

1. INTRODUCTION

Adaptive filters attempt to adjust their coefficients, so as to approximate a desired signal with their output, in terms of a predefined distance function. In order to guarantee the existence of extremes, this function, often referred to as cost function, is usually an even function. The most common cost function is the second order norm (Euclidean distance). Minimizing the square of the ℓ_2 norm of the error, defined as the difference between the output of the adaptive filter and the desired response results in the Least Squares (LS) solution. Denoting by $E\{\bullet\}$ the statistical expectation operator, by $d(n)$ the desired response and by $y(n)$ the output of the adaptive filter at time instant n this cost function is usually given by $J(n) = \frac{1}{2}E\{e^2(n)\} = \frac{1}{2}E\{(d(n) - y(n))^2\}$. In practice the computation of the expectation is not feasible. Thus, real time implementations aim to minimize the instantaneous squared error

$$J(n) = \frac{1}{2}e^2(n) = \frac{1}{2}(d(n) - y(n))^2 \tag{1}$$

instead of the ensemble average of the squared error. A representative of this class of algorithms is the Least Mean Square (LMS), a steepest descent iterative algorithm, that asymptotically tends to the Wiener-Hopf solution. Its updating equation is given by

$$\mathbf{\Theta}(n+1) = \mathbf{\Theta}(n) + \mu(n)e(n)\mathbf{\Phi}(n) \tag{2}$$

with $\mathbf{\Theta}(n) = [\Theta_0(n), \Theta_1(n), \ldots, \Theta_{N-1}(n)]^T$, and $\mathbf{\Phi}(n) = [x(n), x(n-1), \ldots, x(n-N+1)]^T$ the parameters and the regressor vector respectively, when adaptive Finite Impulse Response (FIR) filters are used. The output of the adaptive filter at time n is given by $y(n) = \mathbf{\Theta}^T(n)\mathbf{\Phi}(n)$. In its simplest form LMS employs a constant, time invariant step size. This scheme though, requires *a priori* knowledge about the signal and the system parameters, in order to set the step size close to its optimum value. Also, in the case of signals or systems with time varying statistical characteristics, the optimal step size is time varying, and it cannot be approximated by a constant learning rate.

Several approaches that attempt to overcome this problem, have been proposed, especially for FIR filters. The most common is the Normalized LMS (NLMS) algorithm [3], that succeeds the maximum possible minimization for a given gradient direction, by minimizing the *a posteriori* along with the *a priori* error. Alternatively, algorithms that employ variable step size can be applied [1, 2, 4]. These can converge when applied to non-stationary signals, at the expense of increased computational complexity.

The situation is more complicated for adaptive Infinite Impulse Response (IIR) filters. In this case, normalization of the step size improves the performance only when the Equation Error (EE) method is used for the updating of the filter's coefficients. For the Output Error (OE) technique, normalizing the step size does not necessarily improve the convergence behavior, since in this case the adaptation error is not a linear function of the coefficient error [5, 6]. Thus, in this case a way to cope with time varying environments is to apply algorithms with trainable step sizes. In this paper such an algorithm, with time varying step size, is introduced, called Adaptive Step Size Output Error (ASSOE).

2. ADAPTIVE STEP SIZE LEAST MEAN SQUARE ALGORITHM

In order to cater for the usual independence assumptions, and to enable the LMS algorithm to adjust its step size according to the input signal's dynamics, a gradient adaptive step size that is updated every time instant n, according a steepest descent technique, can be applied. The update equation for the learning rate $\mu(n)$ is given by

$$\mu(n+1) = \mu(n) - \rho\nabla_\mu J(n)_{|\mu=\mu(n-1)}. \tag{3}$$

The cost function $J(n)$ is given in (1). Hence the gradient of the right hand side of equation (3) is

$$\nabla_\mu J(n) = -e(n)\mathbf{\Phi}^T(n)\frac{\partial\mathbf{\Theta}(n)}{\partial\mu(n-1)}. \tag{4}$$

Utilizing the update equation of the parameters of the adaptive FIR filter (2), and taking into account the dependence of the coefficients vector $\mathbf{\Theta}(n-1)$ and the error $e(n-1)$ on the step size $\mu(n-1)$, the gradient of the right hand side of equation (4) becomes

$$\frac{\partial\mathbf{\Theta}(n)}{\partial\mu(n-1)} = \left(1 - \mu(n-1)\mathbf{\Phi}(n-1)\mathbf{\Phi}^T(n-1)\right)\frac{\partial\mathbf{\Theta}(n-1)}{\partial\mu(n-1)} + e(n-1)\mathbf{\Phi}(n-1). \tag{5}$$

Neglecting the partial derivative $\partial\mathbf{\Theta}(n-1)/\mu(n-1)$ in the right-hand-side of equation (5), results in a biased estimate of

the gradient $\partial\mathbf{\Theta}(n)/\mu(n-1)$, since both variables $\mathbf{\Theta}(n-1)$ and $\mu(n-1)$ depend on previous values of the learning rate ($\mu(n-i)$, $i=2,3,\ldots$), but it relaxes significantly the computational complexity [2]. Since the parameter ρ, that is used for the adaptation of the learning rate $\mu(n)$, is usually very small and $\mu(n-1)\approx\mu(n-2)$, the hypothesis $\partial\mathbf{\Theta}(n-1)/\partial\mu(n-1)\approx\partial\mathbf{\Theta}(n-1)/\partial\mu(n-2)$ holds. Denoting the partial derivative $\partial\mathbf{\Theta}(n)/\partial\mu(n-1)$ by $\mathbf{\Upsilon}(n)$, the following recursive formula can be derived [1]:

$$\begin{aligned}\mathbf{\Upsilon}(n) =&\lambda\left(1-\mu(n-1)\mathbf{\Phi}(n-1)\mathbf{\Phi}^T(n-1)\right)\mathbf{\Upsilon}(n-1)\\ &+e(n-1)\mathbf{\Phi}(n-1),\end{aligned}\tag{6}$$

where λ is a multiplicative constant that compensates for any difference between $\partial\mathbf{\Theta}(n-1)/\partial\mu(n-1)$ and $\partial\mathbf{\Theta}(n-1)/\partial\mu(n-2)$. The use of a constant instead of a time varying term λ is a compromise, but adapting λ as well would result in a very complicated and computationally complex algorithm. Simulations show that computation of the gradient $\mathbf{\Upsilon}(n)=\partial\mathbf{\Theta}(n)/\partial\mu(n-1)$ with the recursive formula presented in (6) improves significantly the performance of the adaptive step size algorithm presented in [2], especially when λ is close to unity.

3. THE PROPOSED ADAPTIVE STEP SIZE ALGORITHM FOR IIR FILTERS

Vector Definitions
$\mathbf{\Theta}(n)=[a_1(n),\ldots,a_{N-1}(n),b_0(n),\ldots,b_{M-1}(n)]^T$
$\mathbf{\Phi_o}(n)=[y_o(n-1),\ldots,y_o(n-N+1)$ $,x(n),\ldots,x(n-M+1)]^T$
$\mathbf{\Phi_f}(n)=[y_f(n-1),\ldots,y_f(n-N+1)$ $,x_f(n),\ldots,x_f(n-M+1)]^T$
Output Error Algorithm
$y_o(n)=\mathbf{\Theta}^T(n)\Phi_o(n)$
$e_o(n)=d(n)-y_o(n)$
$x_f(n)=x(n)+\sum_{m=1}^{N-1}a_m(n)x_f(n-m)$
$y_o(n)=\mathbf{\Theta}^T(n)\mathbf{\Phi_o}(n)$
$y_f(n)=y_o(n)+\sum_{m=1}^{N-1}a_m(n)y_f(n-m)$
$\mathbf{\Theta}(n+1)=\mathbf{\Theta}(n)+\mu(n)e_o(n)\mathbf{\Phi_f}(n)$

Table 1. The Output Error method.

Applying the ideas of the previous section to Infinite Impulse Response (IIR) filters, algorithms with trainable step size for recursive adaptive filters can be developed. Adaptive step size algorithms can be applied to both Equation Error (EE) and Output Error (OE) methods [6]. This section deals only with the the Output Error update technique, which can not be straightforwardly normalized like the Equation Error, since in this case the adaptation error is not linearly dependent on the coefficient error [7].

The Output Error method is summarized in Table 1. The filtered regressor vector $\mathbf{\Phi_f}(n)$ is an estimate of the gradient of the output signal with respect to the coefficients of the adaptive IIR filter $\nabla_{\mathbf{\Theta}}y_o(n)$, which is used for their updating [5].

In the standard Output Error method the step size $\mu(n)$ is assumed to be constant. Applying an adaptive step size enables the IIR filter to converge even for nonstationary input signals, since it automatically adjusts to the changes of the environment, provided that the filter operates inside the region of stability.

A steepest descent scheme is adopted for the adaptation of the learning rate [8] (equation (3)), that attempts to minimize the squared error (1). The gradient of the cost function with respect to the step size $\partial J(n)/\partial\mu(n-1)$ is given by (4). Denoting $\partial\mathbf{\Theta}(n)/\partial\mu(n-1)$ by $\mathbf{\Upsilon}(n)$, and expressing $\mathbf{\Theta}(n)$ as a function of $\mathbf{\Theta}(n-1)$ and $\mu(n-1)$ (Table 1) yields

$$\begin{aligned}\mathbf{\Upsilon}(n)\triangleq\frac{\partial\mathbf{\Theta}(n)}{\partial\mu(n-1)}&=\frac{\partial\mathbf{\Theta}(n-1)}{\partial\mu(n-1)}+e(n-1)\mathbf{\Phi_f}(n-1)\\ &+\mu(n-1)\left(\mathbf{\Phi_f}(n-1)\frac{\partial e(n-1)}{\partial\mu(n-1)}+e(n-1)\frac{\partial\mathbf{\Phi_f}(n-1)}{\partial\mu(n-1)}\right)\end{aligned}$$

Following similar considerations with the FIR case, the partial derivative of the error $e(n-1)$ with respect to the step size $\mu(n-1)$ is found to be

$$\frac{\partial e(n-1)}{\partial\mu(n-1)}=-\mu(n-1)\mathbf{\Phi_f}(n-1)\mathbf{\Phi_f}^T(n-1)\frac{\partial\mathbf{\Theta}(n-1)}{\partial\mu(n-1)}.\tag{7}$$

Contrary to the FIR case, the gradient of the regressor vector $\mathbf{\Phi_f}(n-1)$ with respect to the learning rate $\mu(n-1)$ is not zero, and it can be analyzed as follows

$$\frac{\partial\mathbf{\Phi_f}(n-1)}{\partial\mu(n-1)}=\frac{\partial\mathbf{\Phi_f}(n-1)}{\partial\mathbf{\Theta}(n-1)}\frac{\partial\mathbf{\Theta}(n-1)}{\partial\mu(n-1)}\tag{8}$$

Assuming that the adaptation parameter ρ for the updating of the step sizes is sufficiently small, justifies the assumption $\partial\mathbf{\Theta}(n-1)/\partial\mu(n-1)\approx\partial\mathbf{\Theta}(n-1)/\partial\mu(n-2)$. Hence

$$\begin{aligned}\mathbf{\Upsilon}(n)=&\lambda\mathbf{\Upsilon}(n-1)-\lambda\mu(n-1)\mathbf{\Phi_f}(n-1)\mathbf{\Phi_f}^T(n-1)\mathbf{\Upsilon}(n-1)\\ &+\lambda\mu(n-1)e(n-1)\frac{\partial\mathbf{\Phi_f}(n-1)}{\partial\mathbf{\Theta}(n-1)}\mathbf{\Upsilon}(n-1)\\ &+e(n-1)\mathbf{\Phi_f}(n-1)\end{aligned}\tag{9}$$

The multiplicative term λ is introduced in order to compensate for any difference between $\partial\mathbf{\Theta}(n-1)/\partial\mu(n-1)$ and $\partial\mathbf{\Theta}(n-1)/\partial\mu(n-2)$. The effect of this parameter is crucial since it controls the memory of the algorithm. The gradient $\partial\mathbf{\Phi_f}(n-1)/\partial\mathbf{\Theta}(n-1)$ is a matrix whose (i,j) element is the partial derivative of the i-th element of the regressor vector $\mathbf{\Phi_f}(n-1)$ with respect to the j-th element of the parameters vector $\mathbf{\Theta}(n-1)$, both defined in Table 1. Denoting this $(N+M-1)\times(N+M-1)$ square matrix by $\mathbb{D}(n)$, with $N-1$ the number of poles and $M-1$ the number of zeros of the IIR filter, yields

$$\mathbb{D}(n)=\begin{bmatrix}\mathbf{\Xi}(n) & \mathbf{Z}(n)\\ \mathbf{X}(n) & \mathbf{\Psi}(n)\end{bmatrix},\text{ where}\tag{10}$$

$$\mathbf{\Xi}(n)=\begin{bmatrix}\frac{\partial y_f(n-1)}{\partial a_1(n)} & \cdots & \frac{\partial y_f(n-1)}{\partial a_{N-1}(n)}\\ \vdots & \ddots & \vdots\\ \frac{\partial y_f(n-N+1)}{\partial a_1(n)} & \cdots & \frac{\partial y_f(n-N+1)}{\partial a_{N-1}(n)}\end{bmatrix},\tag{11}$$

$$\mathbf{Z}(n) = \begin{bmatrix} \frac{\partial y_f(n-1)}{\partial b_0(n)} & \cdots & \frac{\partial y_f(n-1)}{\partial b_{M-1}(n)} \\ \vdots & \ddots & \vdots \\ \frac{\partial y_f(n-N+1)}{\partial b_0(n)} & \cdots & \frac{\partial y_f(n-N+1)}{\partial b_{M-1}(n)} \end{bmatrix}, \quad (12)$$

$$\mathbf{X}(n) = \begin{bmatrix} \frac{\partial x_f(n)}{\partial a_1(n)} & \cdots & \frac{\partial x_f(n)}{\partial a_{N-1}(n)} \\ \vdots & \ddots & \vdots \\ \frac{\partial x_f(n-M+1)}{\partial a_1(n)} & \cdots & \frac{\partial x_f(n-M+1)}{\partial a_{N-1}(n)} \end{bmatrix}, \text{and} \quad (13)$$

$$\mathbf{\Psi}(n) = \begin{bmatrix} \frac{\partial x_f(n)}{\partial b_0(n)} & \cdots & \frac{\partial x_f(n)}{\partial b_{M-1}(n)} \\ \vdots & \ddots & \vdots \\ \frac{\partial x_f(n-M+1)}{\partial b_0(n)} & \cdots & \frac{\partial x_f(n-M+1)}{\partial b_{M-1}(n)} \end{bmatrix}. \quad (14)$$

The values of the quantities $x_f(n)$ and $y_f(n)$ were defined in Table 1. Proceeding like the output error method, where it is assumed that $\partial y(n-i)/\partial a_l(n) \approx \partial y(n-i)/\partial a_l(n-i)$, for $l, i = 1, 2, \ldots, N-1$ and $\partial y(n-i)/\partial b_m(n) \approx \partial y(n-i)/\partial b_m(n-i)$, for $i = 1, 2, \ldots, N-1$ and $m = 1, 2, \ldots, M-1$, it can assumed that $\partial y_f(n-i)/\partial a_l(n) \approx \partial y_f(n-i)/\partial a_l(n-i)$, $\partial x_f(n-j)/\partial a_l(n) \approx \partial x_f(n-j)/\partial a_l(n-j)$, $\partial y_f(n-i)/\partial b_m(n) \approx \partial y_f(n-i)/\partial b_m(n-i)$ and $\partial x_f(n-j)/\partial b_m(n) \approx \partial x_f(n-j)/\partial b_m(n-j)$, for $i, l = 1, 2, \ldots, N-1$ and $j, m = 1, 2, \ldots, M-1$. Denoting by $\xi_i(n)$ the term $\partial y_f(n)/\partial a_i(n)$, by $\zeta_i(n)$ the quantity $\partial y_f(n)/\partial b_i(n)$, $\partial x_f(n)/\partial a_i(n)$ by $\chi_i(n)$ and $\partial y_f(n)/\partial b_i(n)$ by $\psi_i(n)$, and observing from Table 1 that $y_f(n) = y(n) + \sum_{i=1}^{N-1} a_i(n) y_f(n-i)$ and $x_f(n) = x(n) + \sum_{i=1}^{N-1} a_i(n) x_f(n-i)$, yields

$$\xi_l(n) \triangleq \frac{\partial y_f(n)}{\partial a_l(n)} = y_f(n-l) + \sum_{i=1}^{N-1} a_i(n)\xi_l(n-i) \quad (15)$$

$$\zeta_l(n) \triangleq \frac{\partial y_f(n)}{\partial b_l(n)} = x_f(n-l) + \sum_{i=1}^{N-1} a_i(n)\zeta_l(n-i) \quad (16)$$

$$\chi_l(n) \triangleq \frac{\partial x_f(n)}{\partial a_l(n)} = x_f(n-l) + \sum_{i=1}^{N-1} a_i(n)\chi_l(n-i) \quad (17)$$

$$\psi_l(n) \triangleq \frac{\partial x_f(n)}{\partial b_l(n)} = \sum_{i=1}^{N-1} a_i(n)\psi_l(n-i) \quad (18)$$

Without loss of generality it can be assumed that $\psi_l(n) = 0$ for $n < 0$. Then from (18) $\psi_l(n) = 0$ for every n. Furthermore, from equations (17) and (16), and assuming that $\chi_l(n) = \zeta_l(n) = 0$ for $n < 0$ it is concluded that $\chi_l(n) = \zeta_l(n)$ for every n, thus $\mathbf{X}(n) = \mathbf{Z}^T(n)$. Finally it can be assumed that $\xi_l(n) = \xi_0(n-l) \triangleq \xi(n-l)$ and $\zeta_l(n) = \zeta_0(n-l) \triangleq \zeta(n-l)$. Applying those assumptions to the matrices defined in equations (11)–(14) gives

$$\mathbf{\Xi}(n) = \begin{bmatrix} \xi(n-2) & \cdots & \xi(n-N) \\ \vdots & \ddots & \vdots \\ \xi(n-N) & \cdots & \xi(n-2N+2) \end{bmatrix}, \text{and} \quad (19)$$

$$\mathbf{Z}(n) = \begin{bmatrix} \zeta(n-1) & \cdots & \zeta(n-M) \\ \vdots & \ddots & \vdots \\ \zeta(n-N+1) & \cdots & \zeta(n-N-M+1) \end{bmatrix}, \quad (20)$$

where

$$\xi(n) = y_f(n) + \sum_{i=1}^{N-1} a_i(n)\xi(n-i) \quad (21)$$

$$\zeta(n) = x_f(n) + \sum_{i=1}^{N-1} a_i(n)\zeta(n-i) \quad (22)$$

Thus the matrix $\mathbb{D}(n)$ finally becomes

$$\mathbb{D}(n) = \frac{\partial \mathbf{\Phi}(n)}{\partial \mathbf{\Theta}(n)} = \begin{bmatrix} \mathbf{\Xi}(n) & \mathbf{Z}(n) \\ \mathbf{Z}^T(n) & \mathbf{0}_{\mathbf{M}\times\mathbf{M}} \end{bmatrix}. \quad (23)$$

Updating the step size of the OE method according to equations (19)– (23) and (9), results in the Adaptive Step Size Output Error (ASSOE) algorithm for adaptive IIR filters. This algorithm can be seen as an extension of the Variable Step Size algorithms from FIR to IIR filters. More specifically, for $\lambda = 0$ this algorithm is an application of the algorithm presented in [2] to IIR filters, while for $\lambda = 1$ ASSOE is the extension of the variable step size scheme of [1] to IIR filters.

4. SIMULATIONS

The proposed algorithm was evaluated in a system identification context. The unknown channel, depicted in Figure 1, was an IIR filter with four poles and three zeros. The input signal was white noise of zero mean and unit variance. The desired response was contaminated with 60dB of measurement noise. To perform this identification task, an IIR filter was employed, with numerator and denominator order both set to ten. Notice that in adaptive IIR filters, the order of the numerator must be at least equal to that of the denominator [5]. The weights of the adaptive filter were updated with the Adaptive Step Size Output Error (ASSOE) method. The initial value of the adaptive step size was zero.The results were averaged over an ensemble of 50 simulation runs. The performance of the algorithm was evaluated for $\lambda = 1$ and $\lambda = 0$.

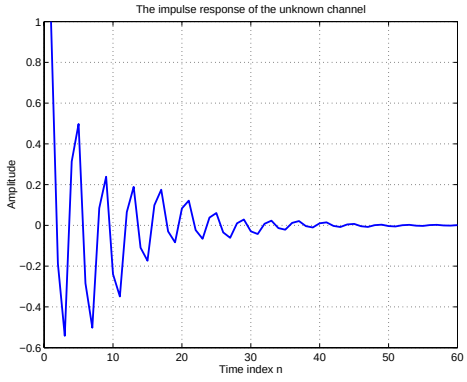

Fig. 1. The impulse response of the unknown channel.

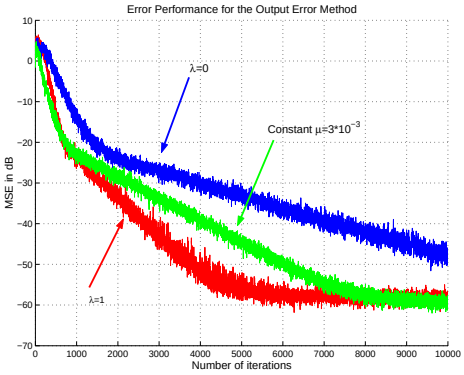

Fig. 2. The dependence of the performance of the Adaptive Step Size Output Error (ASSOE) technique on the multiplicative term λ.

From Figure 2 it is observed that the larger the value of λ the faster the convergence of the algorithm ($\lambda \in [0, 1]$), which was expected, since for values of λ close to unity, the adaptive step size is closer to the exact solution. This behavior can be also justified from Figure 3, where it is depicted that a larger value for λ results in faster convergence, and in larger steady state value $\mu_o = \lim_{n\to\infty} \mu(n)$ for the adaptive step size.

The parameter ρ for $\lambda = 1$ is set to 10^{-7}, while for $\lambda = 0$, $\rho = 10^{-5}$. These values were empirically chosen to be those that provide the faster convergence under the constraint that the system remains stable. When ρ is very small λ should be chosen close to unity, because in this case the difference between $\partial\mathbf{\Theta}(n-1)/\partial\mu(n-1)$ and $\partial\mathbf{\Theta}(n-1)/\partial\mu(n-2)$ is negligible. But while ρ increases λ decreases, since the distance between these two gradients increases. Contrary to what was the case for FIR filters, the range of allowable values for the step size $\mu(n)$ in the IIR case depends on the unknown channel, a property that is inherited to the parameter ρ as well. The reason for that behavior is that the oscillation of the poles of the adaptive filter around their optimum values, which depends on the value of the step size, should be smaller than the distance of the unknown channel's poles from the unit circle in order for this filter to remain stable.

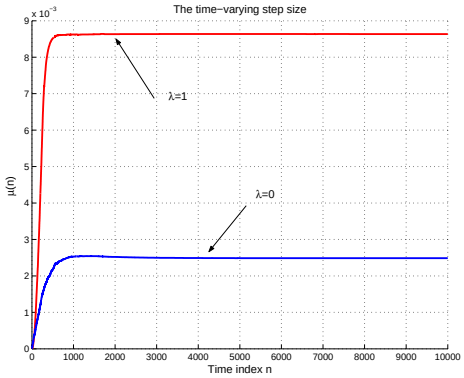

Fig. 3. The effect of the multiplicative term λ on the time evolution of the varying step size, for the Adaptive Step Size Output Error (ASSOE) algorithm.

From Figure 2 observe that the convergence speed of the algorithm changes suddenly after crossing the -20dB threshold. This happens because the poles of the adaptive filter that are not used in the identification of the poles of the unknown channel are cancelled from its own zeros, a procedure that is usually very fast. Thus when the zeros of the adaptive filter "lock" on its extra poles, the convergence rate instantly changes. The performance of the Output Error method, when constant step size is employed is also shown in this figure. Applying a constant learning rate, under the assumption that its optimum value is known performs slightly better that the ASSOE algorithm, since the transition period required for the adaptive step size to settle to its optimal value is avoided. This is not a realistic situation though, since in practise usually there is no *a priori* information concerning the statistical characteristics of the input signal and the unknown system, and thus the optimum step size can be only empirically found. In the case of signals whose statistical properties vary with time (i.e. speech) and there is no time-invariant optimum value for the step size, the ASSOE algorithm performs significantly better, since the step size is adjusted according to the signal dynamics.

5. CONCLUSIONS

A gradient adaptive step size algorithm for IIR adaptive filters has been derived, by applying similar considerations to those that led to the development of the Variable Step Size (VSS) algorithms for FIR filters. The problems that were encountered in the computation of the adaptive step size due to the existence of the recursion have been overcome by introducing assumptions similar to those used in the derivation of the Output Error method. The novel Adaptive Step Size Output Error (ASSOE) algorithm, does not require any *a priori* knowledge, since its step size automatically converges to its optimum value. This adaptive nature of the step size enables this algorithm to converge in time-varying environments.

6. REFERENCES

[1] A. Benveniste, M. Metivier, and P. Priouret, *Adaptive Algorithms and Stochastic Approximations*, New York:Springer-Verlang, 1990.

[2] V.J. Mathews and Z. Xie, "Stochastic Gradient Adaptive Filters eith Gradient Adaptive Step Size," *IEEE Transactions on Signal Processing*, vol. 41, pp. 2075–2087, June 1993.

[3] B. Widrow and S.D. Stearns, *Adaptive Signal Processing*, Prentice Hall, 1985.

[4] W.-P. Ang and B. Farhang Boroujeny, "A New Class of Gradient Adaptive Step-Size LMS Algorithms," *IEEE Transactions on Signal Processing*, vol. 49, no. 4, pp. 805–810, April 2001.

[5] J.J. Shynk, "Adaptive IIR Filtering," *IEEE ASSP Magazine*, pp. 4–21, April 1989.

[6] C.R. Johnson, "Adaptive IIR Filtering: Current Results and Open Issues," *IEEE Transactions on Information Theory*, vol. IT-30, no. 2, pp. 237–250, March 1984.

[7] P.A. Regalia, *Adaptive IIR Filtering in Signal processing and Control*, New York: Marcel Decker, 1995.

[8] D.P. Mandic and J.A. Chambers, "A Normalized Real Time Recurrent Learning Algorithm," *Signal Processing*, vol. 80, no. 9, pp. 1900–1916, September 2000.