IJICTR International Journal of Information & Communication Technology Research

Volume 8 - Number 3- Summer 2016 (7-14)

A Weighted Soft-Max PNLMS Algorithm for Sparse System Identification

M. Bekrani
Department of electrical and computer engineering
Qom University of Technology
Qom, Iran
bekrani@qut.ac.ir

H. Zayyani
Department of electrical and computer engineering
Qom University of Technology
Qom, Iran
zayyani2009@gmail.com

Received: February 14, 2016- Accepted: June 22, 2016

Abstract—**This paper presents a new Proportionate Normalized Least Mean Square (PNLMS) adaptive algorithm using a soft maximum operator for sparse system identification. To provide a high rate of convergence, soft maximum operator is employed along with a weighting factor, which is proportional to an estimation of output mean square error (MSE). Simulation results show the superiority of the proposed algorithm over its PNLMS-based counterparts.**

Keywords-Sparse adaptive filter, System identification, Echo cancelation, Soft maximum.

I. INTRODUCTION

In the last two decades, adaptive filters have achieved wide applications in many signal processing areas [1]. One important area is sparse system identification, which is employed in many applications including channel estimation [2], [3], echo cancelation [4], [5]. In a sparse system, only a few coefficients of the impulse response are active (non-zero) and many of its coefficients are almost inactive (zero or near zero).

In the identification of sparse systems, conventional adaptive filters could be improved, so that the convergence rate is increased while the computational load even decreases. To this end, sparse adaptive filters have been proposed to exploit the sparsity of the filter coefficients [6], [7], [8]. As a special case, the Normalized Least Mean Square (NLMS) adaptive algorithm which employs a unique step-size for all filter coefficients has a slow convergence rate in identification of sparse impulse responses. To mitigate such deficiency, the Proportionate NLMS (PNLMS) algorithm has been proposed which updates each filter coefficient individually proportional to its magnitude [9]. In addition, some other proportionate adaptive algorithms have been suggested to further improve the performance of NLMS. One of such algorithms is Improved PNLMS (IPNLMS) which employs a combination of proportionate and non-proportionate updating [4], [10]. Moreover, in [11], a μ-law PNLMS (MPNLMS) algorithm has been proposed which calculates an approximation of optimal proportionate step-size.

Another variant of PNLMS algorithm, so-called Individual Activation Factor PNLMS (IAF-PNLMS), employs an individual activation factor for each adaptive filter coefficient, as opposed to a global activation factor in the standard PNLMS algorithm [12]. The IAF-PNLMS algorithm achieves a better distribution of the adaptation energy over the filter coefficients than the standard PNLMS does. Thereby, for systems exhibiting high sparseness, this approach achieves faster convergence, outperforming both the

PNLMS and IPNLMS algorithms. A different class of PNLMS algorithms has also been suggested in [13] which takes into account the sparseness measure of the estimated impulse response via a modified coefficient update function and adapts dynamically to the level of sparseness using a new sparseness-controlled approach.

In the weight update process, on the other hand, one can exploit sparsity with a sparsity promoting regularization term [14-16]. The method in [14] is based on minimizing a regularized MSE criterion. The proposed method in [15] employs a l^0 norm in combination with Least Mean Square (LMS) algorithm which results in an improved version of LMS, so-called l^0 -LMS. Moreover, l^0 norm sparsity promoting regularization term was employed with PNLMS algorithm in [16].

In addition to l^0 norm, l^1 norm penalty has also been utilized in conjunction with PNLMS to improve both convergence speed and excess MSE of adaptive filter [17]. Besides, the relation between basis pursuit (l^1 norm optimization) and NLMS algorithm has been investigated in [18] which lead to a new version of PNLMS. Recently, a non-uniform norm ($p-$ norm like) constraint LMS algorithm has also been proposed [19]. Differently, Maximum a Posteriori (MAP) estimation formulation permits the study of a number of prior distributions which naturally incorporate the sparse property of filter coefficients [7]. Consequently, a MAP-LMS adaptive filter [7] and further, a compressed sensing block based MAP-LMS have been introduced [20].

The convergence speed of the PNLMS algorithm, though very high initially, however, slows down at a later stage, even becoming worse than NLMS. In [25_], this problem is addressed by introducing a penalty of constructed l1 norm of the coefficients in the PNLMS cost function which favors sparsity. This helps in the shrinkage of the coefficients, especially the inactive taps, thereby arresting the slowing down of convergence.

In [21], a family of block-sparse PNLMS adaptive algorithms is proposed that improve the performance of identifying block-sparse systems. They are based on the optimization of a mixed norm of the adaptive filter's coefficients.

In this paper, in order to increase the convergence rate of PNLMS, we modify the standard PNLMS algorithm. To do this, a weighted soft maximum operator is introduced and employed instead of a hard maximum operator. The motivation behind this idea is to consider the effect of coefficients in the whole duration of convergence, especially for iterations in which some coefficients are less than the predefined activation thresholds. This procedure improves the performance of the PNLMS algorithm.

We further weight the terms in the soft maximum so that the weighting factor is decreased throughout the convergence of algorithm. The decreasing of the weighting factor is linearly proportional to the estimated MSE. Finally, we show the superiority of our proposed algorithm using numerical simulations.

II. FAMILY OF PNLMS ADAPTIVE FILTERS

Suppose the input signal at discrete time index n is $x(n)$ and sparse impulse response of the unknown system is $\mathbf{w}_o = [w_{o,0}, w_{o,1},..., w_{o,N-1}]^T$ where N is the length of the impulse response. As a result, the system output is

$$y(n) = \mathbf{w}_o^T \mathbf{x}(n) \tag{1}$$

where $\mathbf{x}(n) = [x(n), x(n-1),..., x(n-N+1)]^T$ is the tap-input vector. Moreover, the desired signal is $d(n) = y(n) + v(n)$ where $v(n)$ is the measurement noise. The noise $v(n)$ is assumed to be a zero-mean white Gaussian noise with variance σ_v^2 and uncorrelated with the input signal. The error signal is defined as

$$e(n) = d(n) - \mathbf{w}^T(n)\mathbf{x}(n) \tag{2}$$

where $\mathbf{w}(n) = [w_0(n), w_1(n),..., w_{N-1}(n)]^T$ is the estimated impulse response at time index n.

A wide range of adaptive filter algorithms, including PNLMS, employs the following update formula [4]:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \frac{\mu \mathbf{G}(n) e(n) \mathbf{x}(n)}{\mathbf{x}^T(n)\mathbf{G}(n)\mathbf{x}(n) + \zeta} \tag{3}$$

where $\mathbf{G}(n) = \mathrm{diag}\{g_0(n), g_1(n),..., g_{N-1}(n)\}$ is the gain matrix, $\mu > 0$ is the step-size parameter that control the convergence rate of the algorithm, and $\zeta > 0$ is a regularization parameter that prevents the division by zero.

Equation (3) is a recursive equation derived using minimization of a cost function which leads to the minimization of the estimated MSE [27, 28]. To conduct the weight update, an initialize for $\mathbf{w}(n=0)$ is required, e. g. $\mathbf{w}(0) = \mathbf{0}_{N\times 1}$. Then (3) is repeated to some steps n which is typically large, so that the estimated MSE converges to a steady state. The main idea of the proportionate updating is to assign different step sizes to different coefficients based on their optimal magnitudes. The bigger the magnitude, the larger the step-size assigned [26]. The diagonal elements of gain matrix determine the individual step-sizes of each filter coefficient.

For the standard NLMS algorithm, $\mathbf{G}(n) = \mathbf{I}_{N\times N}$, which means all active and inactive coefficients have the same step-size. As a result, a slow convergence rate can be achieved in identification of sparse systems. In contrast, the active coefficients of PNLMS have larger step-sizes than inactive coefficients, to achieve a faster convergence. In the following, we describe a variety of the PNLMS algorithms in detail

and then we will propose a new PNLMS adaptive filter.

A. Standard PNLMS

In the standard PNLMS algorithm, the gain elements are defined as [12]

$$g_i(n) = \frac{\varphi_i(n)}{\sum_{i=1}^{N} \varphi_i(n)} \quad (4)$$

where $i = 0, 1, ..., N-1$ and $\varphi_i(n)$ is the proportionality function which is

$$\varphi_i(n) = \max(f(n), |w_i(n)|) \quad (5)$$

and $f(n)$ is the activation factor defined as

$$f(n) = \tau \max(\delta, \|\mathbf{w}(n)\|_\infty) \quad (6)$$

in which $\|\mathbf{w}(n)\|_\infty$, τ, and δ are infinity norm, activation parameter, and initialization parameter, respectively [12]. Parameter δ permits starting the adaptation at $n = 0$ when all filter coefficients are initialized to zero [12]. Parameter τ prevents an individual coefficient from freezing when its magnitude is much smaller than that of the largest coefficient [10], [12].

B. IPNLMS

In IPNLMS, a combination of proportionate and non-proportionate update is employed. The gain elements, therefore, are [4]

$$g_i(n) = \frac{1-r}{2N} + (1+r)\frac{w_i(n)}{\|\mathbf{w}(n)\|_1 + \varepsilon} \quad (7)$$

where relative weighting between proportionate and non-proportionate is controlled by a parameter $-1 < r < 1$ and ε is a small positive constant that prevents division by zero [4].

C. MPNLMS

The MPNLMS algorithm employs a β -law function in definition of the activation factor and proportionality function which are defined, respectively, as [11]

$$\varphi_i(n) = \max(f(n), F(|w_i(n)|)), \quad (8)$$

$$f(n) = \tau \max(\delta, \|F(\mathbf{w}(n))\|_\infty) \quad (9)$$

where

$F(\mathbf{w}) = [F(|w_0|), F(|w_1|), ... F(|w_{N-1}|)]$ and $F(.)$ is a β -law logarithmic function as [11]

$$F(|w_i(n)|) = \frac{\ln(1+\beta|w_i(n)|)}{\ln(1+\beta)}. \quad (10)$$

where $\beta = 1/\varepsilon$ in which ε is a small positive number and its value is chosen based on the measurement noise level [11].

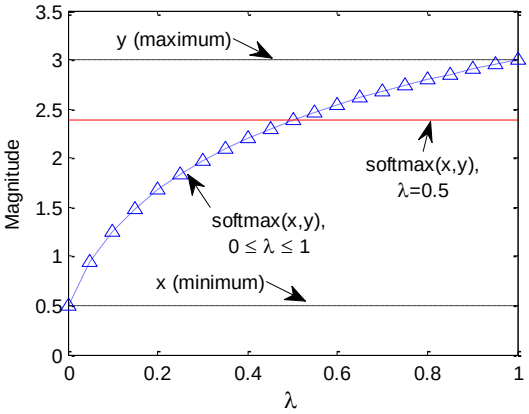

Figure 1: Values of weighted soft-max operator in terms of λ

D. IAF-PNLMS

In IAF-PNLMS, individual coefficients have their own activation factor defined as [12]

$$f_i(n) = \gamma |w_i(n)| + (1-\gamma)\varphi_i(n-1) \quad (11)$$

where $0 < \gamma < 1$ and activation factors are initialized with a small positive constant, typically, $f_i(0) = 0.01/N$ [12].

III. The Proposed Algorithm

The main idea of our proposed algorithm is to employ a soft maximum operator, instead of maximum operator defined in (5), to improve the performance of PNLMS. It is notable from (5) that the result of maximum operator is common and independent from the magnitude of individual coefficients, for those filter coefficients which are less than the predefined activation thresholds. Consequently, in some iterations, the PNLMS performance is similar to the NLMS algorithm which has a common step-size for all coefficients. As a result, a low rate of convergence is derived for a sparse impulse response. In soft maximum, however, the effect of the coefficients is always considered, even in the initial iterations.

A. Soft maximum operator

The soft maximum of two variables is defined

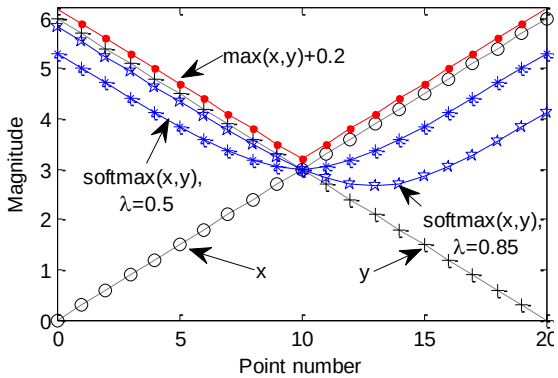

Figure 2: Comparison of max and soft-max operators

as[21]

$$\text{soft}\max(x,y)=\log\{\exp(x)+\exp(y)\}. \quad (12)$$

The soft maximum function has desirable properties such as infinitely differentiability and convexity [21]. It is also notable that, when the difference between two variables is large, the amount of soft maximum approaches that of the maximum operator. We also extend the concept of soft-max operator as follows:

$$\varphi=\log\{\lambda\exp(x)+(1-\lambda)\exp(y)\} \quad (13)$$

where φ is the weighted soft-max operator of x and y, and $0\le\lambda\le1$ is a weighting factor. For $\lambda=0.5$, the weighted soft-max behaves as a scaled soft-max.

As will be shown in the following, the benefit of the weighted soft-max operator is that with selecting $\lambda>0.5$, effect of x on the soft-max result is increased, either x is minimum or maximum. On the other hand, selecting $\lambda<0.5$, result of soft-max is closer to y than those for $\lambda=0.5$. We will later show that this property makes an additional control of coefficient update during the convergence; hence a better convergence performance is achieved.

In order to visualize the effect of weighted soft-max operator, we first consider $x=0.5$ and $y=3$ as minimum and maximum values, respectively. Fig. 1 illustrates the effect of λ on the result of soft-max operator. The result of soft-max for $\lambda=0.5$ is also shown in this figure. As can be seen, the result with $\lambda=0.5$ which is equal to 2.386, tends the maximum. As can be seen, assuming $\lambda=0.5$, the value of weighted soft-max operator is $\varphi=2.386$, which tends the maximum value y.

In addition, the result of weighted soft-max is dependent on the amount of λ such that for $0.5<\lambda\le1$, it is closer to the maximum while for $0\le\lambda<0.5$, it tends more to the minimum. For extremes $\lambda=0$ and $\lambda=1$, the results of soft-max are equal to the minimum and the maximum values, respectively.

We further consider x and y as two scalar variables having 21 values shown in Fig. 2. In this figure, from left to right, x has twenty one increasing values from 0 to 6 while y has twenty one decreasing values from 6 to 0. The results of max operator are shown with an extra value of 0.2 to distinguish from x- and y-points. The results of soft-max operator are also shown for $\lambda=0.5$ and $\lambda=0.85$. As can be seen, the results with $\lambda=0.5$ tend the amounts of maximum of x and y, especially in cases where the maximum is much bigger than the minimum. As opposed to the max operator, the result of soft-max, is affected from both maximum and minimum. The results of soft-max for $\lambda=0.85$ are closer to x than those for $\lambda=0.5$, either x is minimum or maximum. On the other hand, choosing $\lambda<0.5$, the effect of y on the soft-max result is increased. As a result, unlike max operator which its value is equal to the amount of maximum, the value of soft-max is a function of both maximum and minimum values.

B. *Weighting factors for the proposed algorithm*

To further improve the performance of the adaptive algorithm, time-varying weighting factors $\lambda(n)$ and $1-\lambda(n)$ are assigned to $|w_i(n)|$ and $f(n)$, respectively. Therefore, the weighted soft maximum is proposed instead of a maximum, as

$$\varphi_i(n)=\log\{\lambda(n)\exp(|w_i(n)|)+(1-\lambda(n))\exp(f(n))\} \quad (14)$$

for $i=0,1,...,N-1$.

As will be explained in the next section, to expedite the convergence of algorithm, the weighting factor $\lambda(n)$ could be a decreasing function with time such that when the active coefficients $w_i(n)$ reach to the final values, all proportionality functions $\varphi_i(n)$ approach the amount of $f(n)$.

C. *Experimental validation of the proposed algorithm*

To validate the steps of our proposed soft-max PNLMS (SM-PNLMS) algorithm, we conduct some simulation experiments. In the first experiment, we evaluate the performance of SM-PNLMS employing time-invariant $\lambda(n)=\lambda$. Similar to [12], the input signal is assumed to be correlated unity-variance AR(2) as follows,

$$x(n)=b_1x(n-1)+b_2x(n-2)+u(n) \quad (15)$$

where $b_1=0.4$, $b_2=-0.4$, and $u(n)$ is a white Gaussian noise with variance $\sigma_u^2=0.77$. In

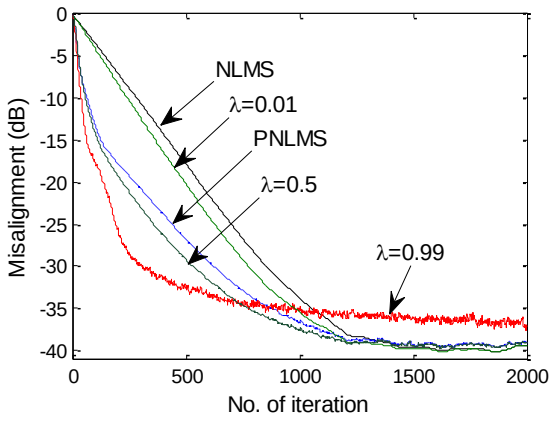

Figure 3: Validation of the proposed algorithm

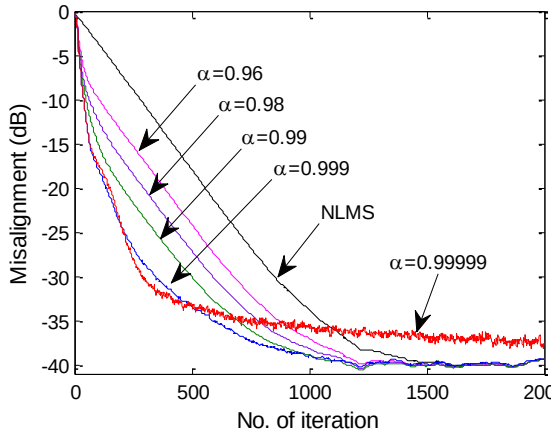

Figure 4: Variation of misalignment with respect to the amount of α

addition, $\mu = 0.5$ and the measurement noise $v(n)$ is white Gaussian with variance $\sigma_v^2 = 10^{-3}$ to achieve SNR=30 dB. The sparse impulse response is assumed with length $N = 100$ and includes only four active coefficients at locations $\{1,30,35,85\}$ with values equal to $\{0.1,1.0,-0.5,0.1\}$.

For evaluation, the normalized misalignment measure (in dB) is employed as [12]

$$\kappa(n) = 20\log_{10}\frac{\|\mathbf{w}_o - \mathbf{w}(n)\|_2}{\|\mathbf{w}_o\|_2}. \quad (16)$$

The resulted normalized misalignment errors are averaged over 100 independent trials. Fig. 3 illustrates five misalignment curves which are derived from NLMS, standard PNLMS, and the proposed algorithm with time-invariant weighting factors $\lambda = 0.01, 0.5, 0.99$. The performance of NLMS and PNLMS are shown for comparison. It is notable from Fig. 3 that employing weighting factor $\lambda = 0.99$ results in the highest convergence rate with the maximum steady-state misalignment error. On the other hand, employing $\lambda = 0.01$ results in a slow convergence, close to that of the NLMS. In addition, the performance of the proposed algorithm with $\lambda = 0.5$ is close to that of PNLMS.

D. Modified version of the proposed algorithm

As shown in Fig. 3, values of $\lambda(n)$ close to one, results in a higher convergence rate while lower values of $\lambda(n)$ cause the algorithm to achieve a less steady-state error. As a result, to expedite the convergence, one can reduce the weighting factor $\lambda(n)$ gradually such that all the proportionality functions $\varphi_i(n)$ approach the scalar $f(n)$. As $f(n)$ is the same for all coefficients, in such a case, the SM-PNLMS behave similar to the NLMS algorithm. As can be seen from Fig. 3, the NLMS algorithm achieves the minimum steady-state misalignment error. Therefore, we expect the SM-PNLMS algorithm achieves a low steady-state error, once the amount of $\lambda(n)$ becomes low (near zero). In order to decrease the weighting

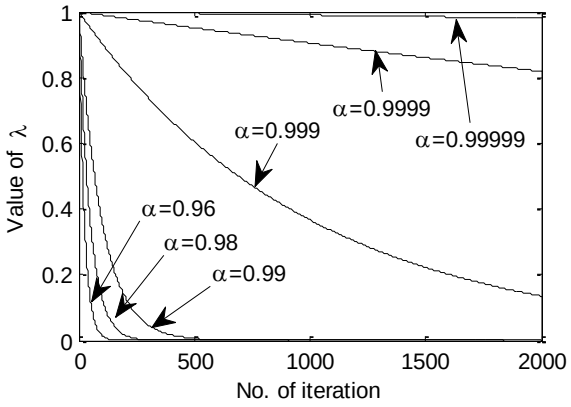

Figure 5: Variation of $\lambda(n)$ with respect to the value of α

factor, we initially employ the following linear function:

$$\lambda(n) = \alpha\lambda(n-1) \quad (17)$$

where $0 << \alpha < 1$ is a constant which should be close to one such as to decrease the weighting factor slowly in accordance with the convergence time.

Fig. 4 illustrates the variation of misalignment for various amounts of α. A small α (still near one) means a fast transition of $\lambda(n)$ from one to zero, while a bigger α very close to one means slow variation of $\lambda(n)$. The variations of $\lambda(n)$ with respect to α are shown in Fig. 5.

As can be seen from Fig. 4, for $\alpha = 0.99999$ the algorithm acts similar to the case of time-invariant $\lambda(n)$. The reason is because the variation (reduction) of $\lambda(n)$ -as shown in Fig. 5- is so slow that we could assume that variation is negligible. On the other hand, for smaller α, for example $\alpha = 0.96$, $\lambda(n)$ dramatically drop to zero and hence, convergence behavior of the algorithms are close to that of NLMS, as shown in Fig. 4.

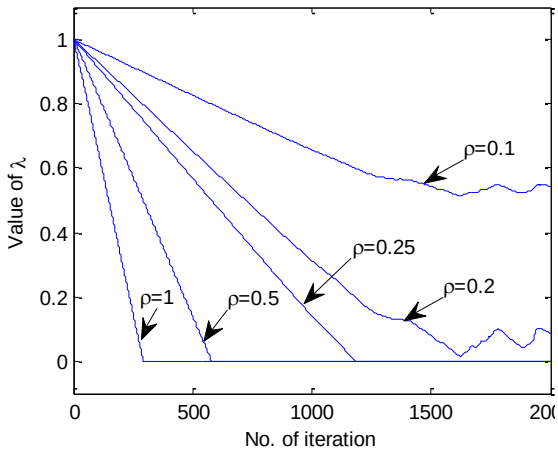

Figure 6: Variation of $\lambda(n)$ with respect to the value of ρ

E. Further modification: A weighting factor based on an estimation of MSE

As mentioned in (17), the value of $\lambda(n)$ and hence, the performance of algorithm strongly depends on the parameter α and the starting point $\lambda(0)$ which is not desirable. In addition, as shown in Fig. 4, for α's close to one, typically $\alpha = 0.99999$, the steady-state misalignment of the proposed algorithm is relatively high in comparison to that of the NLMS and PNLMS algorithms. On the other hand, for a smaller α , typically 0.96 in our simulation $\lambda(n)$ drop to zero very fast, before finalizing the convergence. As a result, the algorithm behaves similar to the NLMS algorithm, during the convergence and hence, its convergence rate is reduced. To mitigate this issue, $\lambda(n)$ could be reduced gradually with convergence of the algorithm.

To this end, we may correlate the weighting factor with the MSE of the adaptive filter. When the algorithm is in initial stages of convergence, the MSE is high and thus the value of $\lambda(n)$ should be near one. After decreasing the MSE error floor, the amount of weighting factor should be decreased.
We estimate the MSE by averaging the square error $e^2(n)$ by a one pole low-pass filter as

$$\psi(n) = \begin{cases} e^2(n), & n = 0 \\ \eta\psi(n-1) + (1-\eta)e^2(n), & n > 0 \end{cases} \quad (18)$$

where $0 << \eta < 1$ is the forgetting factor and typically is assumed to be close to one.

We then make $\lambda(n)$ to be a function of the estimated MSE, so that it could reduce gradually during the convergence of algorithm. To this end, we define

$$\varepsilon(n) = \rho\left(\xi(n) - \xi_{\max}(n)\right) \quad (19)$$

as scaled and normalized output MSE, where $\xi(n) = \log_{10}\psi(n)$ and $\xi_{\max}(n) = \max\{\xi(0), \cdots, \xi(n)\}$. The parameter ρ is a constant which can be chosen such that to be inversely proportional to the amount of the estimated steady-state MSE. According to the definition of (19), $\varepsilon(n)$ starts from zero and approaches -1 on the mean. Accordingly, we propose to assign $\lambda(n)$ as

$$\lambda(n) = \max\{0, 1 + \varepsilon(n)\}. \quad (20)$$

The summation $1+\varepsilon(n)$ in (20) is to set the initial amount of $\lambda(n)$ to be one. In addition, the max operation in (20) acts as a hard limitation to restrict the minimum amount of $\lambda(n)$ to be zero. Therefore during convergence, $\lambda(n)$ starts from one and gradually reduces to finally reach to a value near zero at the end of convergence.

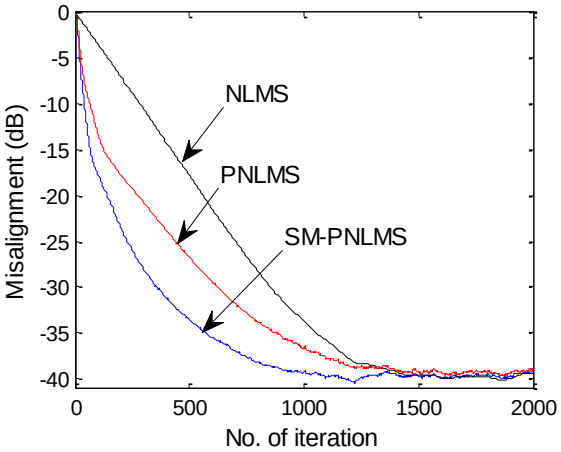

Figure 7: Comparison of our proposed algorithm with other PNLMS algorithms

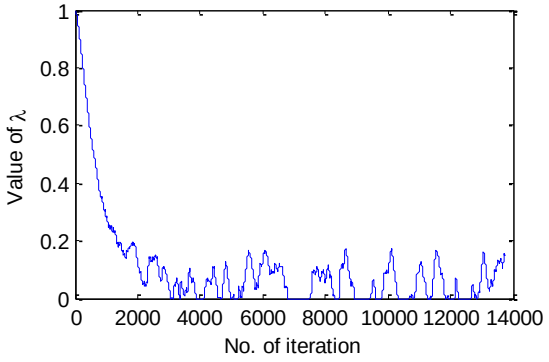

Figure 8: Variation of $\lambda(n)$ in terms of n

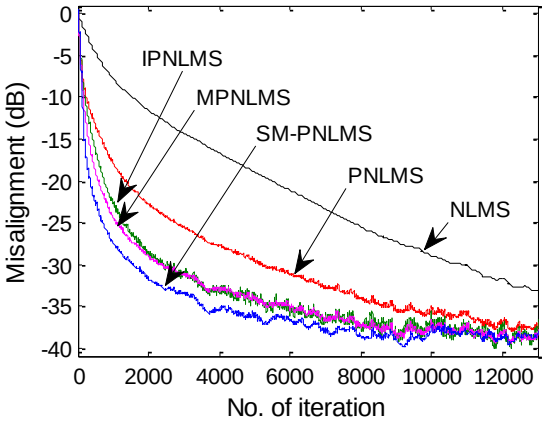

Figure 9: Comparison of our proposed algorithm with other PNLMS algorithms

Fig. 6 illustrates the variation of $\lambda(n)$ in terms of n for different values of ρ for the abovementioned simulation. The curves are obtained employing $\eta = 0.99$.

Our extensive simulations illustrate that the performance of the proposed algorithm is not highly sensitive to the amount of ρ , however, a suitable choice to achieve a high performance is obtained when the amount of ρ is inversely proportional to the estimated steady-state MSE.

Table 1: The proposed algorithm

Initialize:	$\mathbf{w}(0) = \mathbf{0}_{N\times 1}$
Step 1:	$e(n) = d(n) - \mathbf{w}^T(n)\mathbf{x}(n)$
Step 2:	$\psi(n) = \begin{cases} e^2(n), & n = 0 \\ \eta\psi(n-1) + (1-\eta)e^2(n), & n > 0 \end{cases}$
Step 3:	$\xi(n) = \log_{10}\psi(n)$
Step 4:	$\xi_{\max}(n) = \max\{\xi(0), \cdots, \xi(n)\}$
Step 5:	$\varepsilon(n) = \rho(\xi(n) - \xi_{\max}(n))$
Step 6:	$\lambda(n) = \max\{0, 1+\varepsilon(n)\}$
Step 7:	$\varphi_i(n) = \log\{\lambda(n)\exp(\lvert w_i(n)\rvert) + (1-\lambda(n))\exp(f(n))\}$
Step 8:	$g_i(n) = \dfrac{\varphi_i(n)}{\sum_{i=1}^{N}\varphi_i(n)}$
Step 9:	$\mathbf{G}(n) = \mathrm{diag}\{g_0(n), g_1(n), \ldots, g_{N-1}(n)\}$
Step 10:	$\mathbf{w}(n+1) = \mathbf{w}(n) + \dfrac{\mu\mathbf{G}(n)e(n)\mathbf{x}(n)}{\mathbf{x}^T(n)\mathbf{G}(n)\mathbf{x}(n) + \zeta}$
Step 11:	Go to step 1

Now, if we replace the weighting factor (17) with the function of (20) and substitute (20) in (14), an improved performance for SM-PNLMS is obtained. A typical result using AR(2) input signal is shown in Fig. 7.

As can be seen, the convergence rate of SM-PNLMS is better than PNLMS and NLMS so that misalignment of SM-PNLMS reaches -35 dB after 570 iterations while PNLMS and NLMS reach that level after 881and 1054 iterations, respectively. Table 1 illustrates the summary of the proposed algorithm.

numerical simulations

In this section, we employ Monte Carlo simulations to evaluate the performance of the proposed algorithm and compare it with other lgorithms. The results obtained by averaging over 10 independent trials. For all simulations, a colored speech-like signal is used as input signal. This signal is obtained by passing a white Gaussian noise through a low-pass filter which has coefficients $[0.3574,\ 0.9,\ 0.3574]$ [22]. Variance of input is assumed to be equal to $\sigma_x^2 = 1$. We evaluate our proposed algorithm using $\eta = 0.99$ and $\rho = 0.25$ and compare it with NLMS, PNLMS, IPNLMS and MPNLMS.

In the first simulation experiment, the sparse impulse response with length $N = 256$ is considered. The active coefficients are located at $\{1,50,100,12\ 0,200,220\}$ with the values $\{0.1,1.0,-0.5,0.4,-0.2,1.0\}$, respectively.

The step-size is equal to 0.95 for all algorithms. The variation of $\lambda(n)$ in terms of n is shown in Fig. 8. As can be seen, it gradually reduces from its initial value and after around 2000 iterations; it reaches below 0.2 which means the algorithm approaches NLMS after initial convergence. The variations of misalignment for algorithms are shown in Fig. 9. As we can see, to achieve -35 dB, the proposed SM-PNLMS algorithm requires 4000 iterations, while the MPNLMS, IPNLMS, PNLMS, and NLMS require about 5400, 5900, 8800, 14000 iterations, respectively. As a result, SM-PNLMS, achieves the fastest convergence rate among the mentioned algorithms.

In the next simulation, we evaluate identification of a simulated acoustic room impulse response (RIR) which is derived using image methods [23]. For this simulation, the dimension of the room is $4^{\mathrm{m}} \times 4^{\mathrm{m}} \times 3^{\mathrm{m}}$ and the locations of the source and receiver are $(1, 0.95, 1.5)$ meters and $(1.1, 1.05, 1.53)$ meters, respectively. Fig. 10 shows the simulated RIR which is an example of sparse impulse responses.

For simulation, the step-size is considered 1.2 for all algorithms and other specifications are same as the previous experiments. Fig. 11 compares the misalignment of the proposed SM-PNLMS algorithm ith its counterparts for the above RIR with length

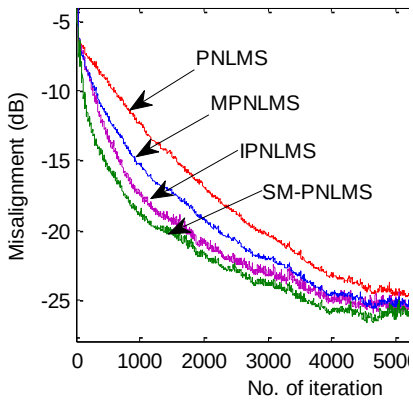

Figure 11: Comparison of the pro with other PNLMS based algorith impulse response.

$N = 333$. As can be seen, the proposed algorithm achieves a higher rate of convergence, so that its misalignment error reach -20 dB after 1400 iterations while it takes 1700, 2200, and 2800 iterations for IPNLMS, MPNLMS, and PNLMS, respectively.

IV. CONCLUSIONS

We proposed a new Proportionate Normalized Least Mean Square (PNLMS) adaptive algorithm for sparse system identification. This algorithm employs a weighted soft maximum operator along with a variable weighting factor to achieve a high convergence rate. We experimentally found a formula for the weighting factor in terms of the estimated mean square error (MSE). Finally, we showed the superiority of our proposed algorithm over its counterparts using numerical Monte Carlo simulations.

ACKNOWLEDGMENT

H. Zayyani thanks Iran's National Elites Foundation. This work was supported in part by a grant from Iran's National Elites Foundation.

REFERENCES

[1] S. Haykin, Adaptive filter theory, 4rd ed.. Englewood Cliffs, NJ: Prentice Hall, 2002.

[2] S. F. Cotter and B.D. Rao, "Sparse channel estimation via matching pursuit with application to equalization," IEEE. Trans. Communication, vol. 50, pp. 374–377, 2002.

[3] S. Vedantam, C. Carbonelli, and U. Mitra, "Sparse channel estimation with zero tap detection," IEEE. Trans. Wireless. Communication, vol. 6, pp. 1743–1763, 2007.

[4] P. A. Naylor, J. Cui, and M. Brookes, "Adaptive algorithms for sparse echo cancelation," Signal Processing, vol.86, pp. 1182–1192, 2006.

[5] J. Benesty, T. Gansler, D. R. Morgan, M. M. Sondhi, and S. L. Gay, Advances in network and acoustic echo cancelation, Berlin: Springer, 2001.

[6] R. K. Martin, W. A. Sethares, R. C. Williamson, and C.R. Johnson, "Exploiting sparsity in adaptive filters," IEEE. Trans. Signal Processing, vol. 50, pp. 1883–1894, 2002.

[7] G. Deng, "Partial update and sparse adaptive filters," IET

[8] Signal Processing, vol.1, pp. 9–17, 2007.

[9] R. L. Das and M. Chakraborty, "Sparse adaptive filters- an overview and some new results," in: Proceeding ISCAS-2012, pp. 2745–2748, 2012.

[10] D. L. Duttweiler, "Proportionate normalized least-mean-squares adaptation in echo cancelers," IEEE. Trans. Speech Audio Processing, vol. 8, pp. 508–518, 2000.

[11] J. Benesty and S. L. Gay, "An improved PNLMS algorithm," in: Proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing, pp. 1881–1884, 2002.

[12] H. Deng and M. Doroslovacki, "Improving convergence of the PNLMS algorithm for sparse impulse response identification," IEEE. Signal Processing Letters, vol. 12, pp. 181–184, 2005.

[13] F. de Souza, O. J. Tobias, R. Seara, and D. R. Morgan, "A PNLMS algorithm with individual activation factors," IEEE. Trans. Signal Processing, vol. 58, pp. 2036–2047, 2010.

[14] P. Loganathan, A. W. H. Khong, and P. A. Naylor, "A class of sparsness-controlled algorithms for echo cancellation," IEEE. Trans. Speech Audio Processing, vol. 17, pp. 1591–1601, 2009.

[15] B. D.Rao and S. Bongyong, "Adaptive filtering algorithms for promoting sparsity," in: Proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing, pp. 361–364, 2003.

[16] Y. Ju, J. Jin, and S. Mei "l_0 norm constraint LMS algorithm for sparse system identification," IEEE. Signal Processing Letters, vol. 16, pp. 774–777, 2009.

[17] C. Paleologu, J. Benesty, and S. Siochina, "An improved proportionate NLMS algorithm based on l_0 norm," in: Proceedings of the IEEE International Conference on Acoustic, Speech and Signal Processing , pp. 309–312, 2010.

[18] R. L. Das and M. Chakraborty, "A zero attracting proportionate normalized least mean square algorithm," in: Signal Signal and Information Processing Association Annual Summit and Conference, pp. 1–4, 2012.

[19] J. Benesty, C. Paleologu, and S. Siochina, "Proportionate adaptive filters from a basis pursuit prespective," IEEE. Signal Processing Letters, vol. 17, pp. 985–988, 2010.

[20] F. Y. Wu and F. Tong, "Non-uniform norm constraint LMS algorithm for sparse system identification," IEEE. Communication Letters, vol. 17, pp. 385–388, 2013.

[21] H. Zayyani, M. Babaie-zadeh, and C. Jutten, "Compressed sensing block MAP-LMS adaptive filter for sparse channel estimation and a Bayesian Cramer Rao bound," in: Proceeding MLSP2009, 2009.

[22] S. Boyd and L. Vandenbergh, Convex optimization, Camcridge University, 2004.

[23] M. Bekrani, A. W. H. Khong, and M. Lotfizad, "`A Linear Neural Network based Approach to Stereophonic Acoustic Echo Cancellation," IEEE Trans. Speech Audio Processing, vol. 19, pp. 1743–1753, 2011.

[24] J. B. Allen and D. A. Berkley, "Image method for efficiently simulating small-room acoustics," *J. Acoust. Soc. Amer*., vol. 65, no. 4, pp.943–950, 1979.

[25] J. Liu and S. L. Grant, "Proportionate Adaptive Filtering for Block-Sparse System Identification," IEEE/ACM Trans. Audio, Speech, Language Process., vol. 24, no. 4, pp. 623-630, 2016.

[26] R. L. Das and M. Chakraborty, "Improving the Performance of the PNLMS Algorithm Using l1 Norm Regularization," IEEE/ACM Trans. Audio, Speech, Language Process., vol. 24, no. 7, pp. 1280-1290, 2016.

[27] H. Deng and M. Doroslovacki, "Improving Convergence of the PNLMS Algorithm for Sparse Impulse Response Identification," IEEE. Signal Processing Letters, vol. 12, no. 3 pp. 181–184, 2005.

[28] S. Haykin, Adaptive Filter Theory, 5th Ed., Prentice Hall, 2013.

[29] B. Farhang-Boroujeny, Adaptive Filters, Theory and Applications, Wiley 1999.

Mehdi Bekrani received the B.Sc. degree from Ferdowsi University of Mashhad in 2002, and the M.Sc. and Ph.D. degrees from Tarbiat Modares University in 2004 and 2010, respectively, all in electrical engineering. From 2011 to 2013, he was a Research Fellow at Nanyang Technological University, Singapore. He is currently an assistant professor of electrical engineering at Qom University of Technology (QUT).

Hadi Zayyani received his B.Sc., M.Sc. and P.hD. degrees all in Electrical Engineering and all in the communications field from Sharif University of Technology in 2000, 2003 and 2010, respectively. From 2012 to now, he is assistant professor of Qom University of Technology (QUT).