

A Family of Selective Partial Update Affine Projection Adaptive Filtering Algorithms

M. Sh. Esfand Abadi*, V. Mehrdad* and M. Noroozi*

Abstract: In this paper we present a general formalism for the establishment of the family of selective partial update affine projection algorithms (SPU-APA). The SPU-APA, the SPU regularized APA (SPU-R-APA), the SPU partial rank algorithm (SPU-PRA), the SPU binormalized data reusing least mean squares (SPU-BNDR-LMS), and the SPU normalized LMS with orthogonal correction factors (SPU-NLMS-OCF) algorithms are established by this general formalism. In these algorithms, the filter coefficients are partially updated rather than the entire filter coefficients at every iteration which is computationally efficient. Following this, the transient and steady-state performance analysis of this family of adaptive filter algorithms are studied. This analysis is based on energy conservation arguments and does not need to assume a Gaussian or white distribution for the regressors. We demonstrate the performance of the presented algorithms through simulations in system identification and acoustic echo cancellation scenarios. The good agreement between theoretically predicted and actually observed performances is also demonstrated.

Keywords: Adaptive filter, affine projection, selective partial update, mean-square performance, energy conservation.

1 Introduction

Adaptive filtering is an important subfield of digital signal processing having been actively researched for more than four decades and having important applications such as noise cancellation, system identification, telecommunications channel equalization, and telephony acoustic and network echo cancellation [1], [2], [3]. In some of these applications, a large number of filter coefficients are needed to achieve an acceptable performance. Therefore the computational complexity is the main problem in these applications. Several adaptive filter algorithms such as the adaptive filter algorithms with selective partial updates have been proposed to solve these problems. These algorithms update only a subset of the filter coefficients in each time iteration. The Max-NLMS [4], the MMax-NLMS [5], [6], the variants of the selective partial update normalized least mean square algorithms (SPU-NLMS) [7], [8], [9], the selective partial update transform domain LMS (SPU-TD-LMS) [10], and SPU affine projection (SPUAP) algorithm [8], are important

examples of this family of adaptive filter algorithms. In contrast to full update adaptive algorithms, the convergence analysis of adaptive filters with selective partial updates has not been widely studied. Many contributions focus on a particular algorithm, making more or less restrictive assumptions on the input signal. For example in [6], the convergence analysis of the MMax-NLMS for zero mean independent Gaussian input signal is presented. Also, the results focus on the steady-state behavior and do not present the theoretical learning curves. In [8], many variants of the SPUNLMS are presented based on the constrained optimization problem. The same assumption as in [6] is used for the input signal and there are no experimental results to justify the theoretical mean square performance of the SPU-NLMS algorithms. Also, the performance analysis of the SPU-AP algorithm was not studied in [8]. The results in [9] present mean square convergence analysis of the SPU-NLMS for the case of white input signals.

In [11], a general formalism for mean-square performance analysis of adaptive filter algorithms with selective partial updates was presented. The analysis was based on energy conservation arguments and did not require a Gaussian or white distribution for the regressors [3], [12], [13], [14]. In this paper we present the general formalism for the family of affine projection algorithm. The regularized APA (R-APA) [15], the binormalized data-reusing LMS (BNDR-LMS) [16], the partial rank algorithm (PRA) [17], the NLMS with

Iranian Journal of Electrical & Electronic Engineering, 2009.

Paper first received 4 Apr. 2009 and in revised form 4 May 2009.

This work was supported by the Iran Telecommunication Research Center (ITRC).

* The Authors are with the Shahid Rajaei Teacher Training University, Department of Electrical Engineering, Tehran, Iran.

E-mail: mshams@srutu.edu, vmehrdad@srutu.edu and mnoroozi@srutu.edu

orthogonal correction factors (NLMS-OCF) [18] are established with this general formalism.

By generalizing the approach in [8], we present SPU-R-APA, SPU-PRA, SPU-BNDR-LMS, and SPU-NLMS-OCF algorithms which are called the family of affine projection algorithms with selective partial updates. In this paper we also present the mean-square performance analysis of the family of SPU-AP algorithms which was not studied in [8]. This analysis is based on energy conservation arguments and does not need to assume the Gaussian or white distribution for the regressors. What we propose in this paper can be summarized as follows:

- Presenting the general formalism to establish of the family of AP algorithms.
- The establishment of the family of SPU-AP algorithms. In these algorithms the filter coefficients are partially updated rather than the entire filter at every adaptation.
- Mean-square performance analysis of the family of SPU-AP algorithms.
- Mean and mean-square stability analysis of the family of SPU-AP algorithms.
- Demonstrating of the presented algorithms in system identification and acoustic echo cancellation scenarios.

This paper is organized as follows. In Section 2, we briefly review the NLMS, and the SPU-NLMS algorithms. Section 3 presents the general update equation for the family of AP algorithms. In the following, the family of SPU-AP algorithms is presented. In the next section the general mean-square performance analysis is developed and the expression for the theoretical learning curves, the mean square coefficient deviation (MSD), and the steady-state mean square error (MSE) are derived. In Section 5, the general expressions for mean and mean-square stability are established. The computational complexity of SPU-APA was presented in Section 7. We conclude the paper by showing a comprehensive set of simulation results.

Throughout the paper, the following notations are used:

$\ \cdot \ $	Euclidean norm of a vector.
$\ \cdot \ _{\Sigma}^2$	Σ Weighted Euclidean norm of a column vector \mathbf{t} defined as $\mathbf{t}^T \Sigma \mathbf{t}$.
$\text{vect}(T)$	Creates an $M^2 \times 1$ column vector \mathbf{t} through stacking the columns of the $M \times M$ column matrix T .
$\text{vect}(\mathbf{t})$	Creates an $M \times M$ matrix T from the $M^2 \times 1$ column vector \mathbf{t} .
$A \otimes B$	Kronecker product of matrices A and B .
$\text{Tr}(\cdot)$	Trace of a matrix.
$(\cdot)^T$	Transpose of a vector or a matrix.

λ_{\max}	The largest eigenvalue of a matrix.
\mathbb{R}^+	The set of positive real numbers.
$E\{\cdot\}$	Expectation operator.

2 Background on NLMS and SPU-NLMS Algorithms

Figure 1 shows a typical adaptive filter setup, where $\mathbf{x}(n)$, $d(n)$ and $e(n)$ are the input, the desired and output error signals, respectively. Here, $\mathbf{h}(n)$ is the $M \times 1$ column vector of filter coefficients at iteration n . The desired signal assumed to conform to the following linear data model

$$d(n) = \mathbf{x}^T(n) \mathbf{h}_t + v(n) \quad (1)$$

where $\mathbf{x}(n) = [x(n), x(n-1), \dots, x(n-M+1)]^T$ is the input signal regressors, $v(n)$ is the measurement noise, assumed to be zero mean, white, Gaussian, and independent of $\mathbf{x}(n)$, and \mathbf{h}_t is the unknown filter vector.

It is well known that the NLMS algorithm can be derived from the solution of the following optimization problem:

$$\min_{\mathbf{h}(n+1)} \|\mathbf{h}(n+1) - \mathbf{h}(n)\|^2 \quad (2)$$

subject to

$$d(n) = \mathbf{x}^T(n) \mathbf{h}(n+1) \quad (3)$$

Using the method of Lagrange multipliers to solve this optimization problem leads to the following recursion

$$\mathbf{h}(n+1) = \mathbf{h}(n) + \frac{\mu}{\|\mathbf{x}(n)\|^2} \mathbf{x}(n)e(n) \quad (4)$$

where $e(n) = d(n) - \mathbf{x}^T(n) \mathbf{h}(n)$ and μ is the step-size that determines the convergence speed and excess MSE (EMSE). Now partition the input signal vector and the vector of filter coefficients into B blocks each of length L ($B = M/L$ is an integer), which are defined as

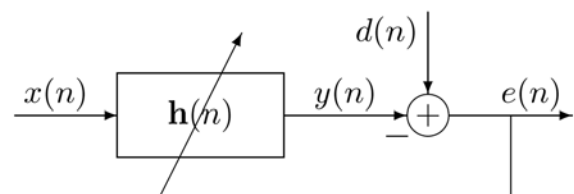


Fig. 1. A typical adaptive filter setup.

$$\mathbf{x}(n) = [\mathbf{x}_1^T(n), \mathbf{x}_2^T(n), \dots, \mathbf{x}_B^T(n)]^T \quad (5)$$

$$\mathbf{h}(n) = [\mathbf{h}_1^T(n), \mathbf{h}_2^T(n), \dots, \mathbf{h}_B^T(n)]^T \quad (6)$$

The SPU-NLMS algorithm for a single block update at every iteration minimizes following optimization problem

$$\min_{\mathbf{h}_j(n+1)} \left\| \mathbf{h}_j(n+1) - \mathbf{h}_j(n) \right\|^2 \quad (7)$$

subject to Eq. (3), where j denotes the index of the block that should be updated [8]. Again by using the method of Lagrange multipliers, the update equation for SPU-NLMS is given by

$$\mathbf{h}_j(n) = \mathbf{h}_j(n) + \frac{\mu}{\left\| \mathbf{x}_j(n) \right\|^2} \mathbf{x}_j(n) \mathbf{e}(n) \quad (8)$$

where $j = \operatorname{argmax}_i \left\| \mathbf{x}_i(n) \right\|^2$ for $1 \leq i \leq B$.

3 Family of Affine Projection Algorithms (APA)

Now define the $M \times K$ matrix of the input signal as

$$\mathbf{X}(n) = [\mathbf{x}(n), \mathbf{x}(n-D), \dots, \mathbf{x}(n-(K-1)D)] \quad (9)$$

and the $K \times 1$ vector of desired signal as

$$\mathbf{d}(n) = [d(n), d(n-D), \dots, d(n-(K-1)D)]^T \quad (10)$$

where K is positive integer (usually, but not necessarily $K \leq M$), and D is the positive integer parameter ($D \geq 1$). that can increase the separation and consequently reduce the correlation among the regressors in $\mathbf{X}(n)$.

The family of APA can be established by minimizing (2) but subject to $\mathbf{d}(n) = \mathbf{X}^T(n)\mathbf{h}(n)$. Again by using the method of Lagrange multipliers, the filter vector update equation for the family of APA is given by

$$\mathbf{h}(n+1) = \mathbf{h}(n) + \mu \mathbf{X}(n) \mathbf{W}(n) \mathbf{e}(n) \quad (11)$$

where $\mathbf{e}(n)$ is the output error vector which is defined as

$$\mathbf{e}(n) = \mathbf{d}(n) - \mathbf{X}^T(n)\mathbf{h}(n) \quad (12)$$

and the matrix $\mathbf{W}(n)$ is obtained from Table 1. In Table 1, ϵ is the regularization parameter, and \mathbf{I} is the identity matrix. The NLMS, ϵ -NLMS, standard version of the APA, the binormalized data-reusing LMS (BNDR-LMS) [16], the regularized APA (RAPA) [15],

the NLMS with orthogonal correction factors (NLMS-OCF) [18] are established from Eq. (11). From Eq. (11), the partial rank algorithm (PRA) [17] can also be established when the adaptation of the filter coefficients is performed only once every K iteration.

4 The Family of Selective Partial Update APA (SPU-APA)

The SPU-APA solves the following optimization problem

$$\min_{\mathbf{h}_F(n+1)} \left\| \mathbf{h}_F(n+1) - \mathbf{h}_F(n) \right\|^2 \quad (13)$$

subject to $\mathbf{d}(n) = \mathbf{X}^T(n)\mathbf{h}(n)$, where $F = \{j_1, j_2, \dots, j_S\}$ denote the indices of the S blocks out of B blocks that should be updated at every adaptation. Again by using the Lagrange multipliers approach, the filter vector update equation is given by

$$\mathbf{h}_F(n+1) = \mathbf{h}_F(n) + \mu \mathbf{X}_F(n) (\mathbf{X}_F^T(n) \mathbf{X}_F(n))^{-1} \mathbf{e}(n) \quad (14)$$

where

$$\mathbf{X}_F(n) = [\mathbf{X}_{j_1}^T(n), \mathbf{X}_{j_2}^T(n), \dots, \mathbf{X}_{j_S}^T(n)]^T \quad (15)$$

is the $SL \times K$ matrix and

$$\mathbf{X}_i(n) = [\mathbf{x}_i(n), \mathbf{x}_i(n-D), \dots, \mathbf{x}_i(n-(K-1)D)]^T \quad (16)$$

is the $L \times K$ matrix. The indices of F are obtained by the following procedure:

1) Compute the following values for $1 \leq i \leq B$.

$$\operatorname{Tr}(\mathbf{X}_i^T(n) \mathbf{X}_i(n)) \quad (17)$$

2) The indices of F are correspond to S largest values of Eq. (17).

By setting $D=1$, the SPU-APA in [8] can be derived from Eq. (14). Furthermore, from Eq. (14), the new SPU adaptive algorithms such as SPU-BNDR-LMS, SPU-NLMS-OCF will be established. Also, the SPU-PRA can be established when the adaptation of the filter coefficients is performed only once every K iterations Equation (14) can be represented in the form of full update equation as

$$\mathbf{h}(n+1) = \mathbf{h}(n) + \mu \mathbf{A}(n) \mathbf{X}(n) (\mathbf{X}^T(n) \mathbf{A}(n) \mathbf{X}(n))^{-1} \mathbf{e}(n) \quad (18)$$

where the $\mathbf{A}(n)$ matrix is the $M \times M$ diagonal matrix with the 1 and 0 blocks each of length L on the diagonal and the positions of 1's on the diagonal determine which coefficients should be updated in each iteration. The positions of 1 blocks (S blocks and $S \leq B$) on the diagonal of $\mathbf{A}(n)$ matrix for each iteration in the family of SPU-APA are determined by the indices of F .

5 Mean-Square Performance Analysis of the Family of SPU-APA

Now we introduce the general filter vector update equation to analysis the performance of the family of SPU affine projection algorithms. The general filter vector update equation is introduced as

$$\mathbf{h}(n+1) = \mathbf{h}(n) + \mu \mathbf{C}(n) \mathbf{X}(n) \mathbf{Z}(n) \mathbf{e}(n) \quad (19)$$

where $\mathbf{C}(n)$ and $\mathbf{Z}(n)$ matrices are obtained from Table 2. In the mean square performance analysis, we need to study the time evolution of the $E\{\|\tilde{\mathbf{h}}(n)\|_{\Sigma}^2\}$, where Σ is any Hermitian and positive-definite matrix, and $\tilde{\mathbf{h}}(n)$ is the weight-error vector which is defined as

$$\tilde{\mathbf{h}}(n) = \mathbf{h}_t - \mathbf{h}(n) \quad (20)$$

For $\Sigma = \mathbf{I}$, the Mean Square Deviation (MSD) and when $\Sigma = \mathbf{R}$ where $\mathbf{R} = E\{\mathbf{x}(n)\mathbf{x}^T(n)\}$ is the autocorrelation matrix of the input signal, the Excess Mean Square (EMSE) expressions are established respectively.

By Generalizing the approaches of [12], [13] or [14] for the above update equation, we obtain after some tedious, but straight forward calculations, the following recursion [19]:

$$E\left\{\left\|\tilde{\mathbf{h}}(n+1)\right\|_{\Sigma}^2\right\} = E\left\{\left\|\tilde{\mathbf{h}}(n)\right\|_{\mathbf{G}\Sigma}^2\right\} + \mu^2 \sigma_v^2 \gamma^T \Sigma \quad (21)$$

where σ_v^2 is the variance of measurement noise, $\sigma = \text{vec}(\Sigma)$ and γ and $M^2 \times M^2$ matrix \mathbf{G} are obtained from the following relations

$$\mathbf{G} = \mathbf{I} - \mu E\{\mathbf{X}(n)\mathbf{D}(n)\} \otimes \mathbf{I} - \mu \mathbf{I} \otimes E\{\mathbf{X}(n)\mathbf{D}(n)\} + \mu^2 E\{(\mathbf{X}(n)\mathbf{D}(n)) \otimes (\mathbf{X}(n)\mathbf{D}(n))\} \quad (22)$$

where $\mathbf{D}(n) = \mathbf{Z}^T(n) \mathbf{X}^T(n) \mathbf{C}^T(n)$ and

$$\gamma = \text{vect}(E\{\mathbf{D}^T(n)\mathbf{D}(n)\}) \quad (23)$$

Derivation of this recursion is based on energy conservation arguments and does not need to assume a Gaussian or white distribution for the regressors [12]. Focusing on the learning curve, we substitute \mathbf{R} for Σ , define $\mathbf{r} = \text{vect}(\mathbf{R})$, and find

$$E\left\{\left\|\tilde{\mathbf{h}}(n)\right\|_{\mathbf{R}}^2\right\} = E\left\{\left\|\tilde{\mathbf{h}}(0)\right\|_{\mathbf{G}^n \mathbf{R}}^2\right\} + \mu^2 \sigma_v^2 \gamma^T \sum_{i=0}^{n-1} \mathbf{G}^i \mathbf{r} \quad (24)$$

This expression is easy to compute recursively once we have estimates for \mathbf{G} and \mathbf{R} . Such estimates are easily obtained from a single realization of the signals involved in the adaptive filter. From Eq. (1) and Eq.

(20), we obtain $\mathbf{e}(n) = \mathbf{x}^T(n) \tilde{\mathbf{h}}(n) + v(n)$. Therefore the time evolution of the mean square error and accordingly the theoretical learning curve is given by [19]

$$\text{MSE} = E\{e^2(n)\} = E\left\{\left\|\tilde{\mathbf{h}}(n)\right\|_{\mathbf{R}}^2\right\} + \sigma_v^2 \quad (25)$$

From Eq. (24), we will be able to evaluate the steady-state meansquare error (MSE), when n goes to infinity,

$$\text{MSE} = \mu^2 \sigma_v^2 \gamma^T (\mathbf{I} - \mathbf{G})^{-1} \mathbf{r} + \sigma_v^2 \quad (26)$$

where $\mu^2 \sigma_v^2 \gamma^T (\mathbf{I} - \mathbf{G})^{-1} \mathbf{r}$ is the steady-state EMSE, and the mean-square deviation (MSD) in the steady-state is given by

$$\text{MSD} = \mu^2 \sigma_v^2 \gamma^T (\mathbf{I} - \mathbf{G})^{-1} \text{vect}(\mathbf{I}) \quad (27)$$

It is important to note that the results in [11], [12], [13], [14], and [19] are special cases of the general expressions (equations (25), (26) and (27)).

6 Mean and Mean-Square Stability of the Family of SPU-APA

From Eq. (19) and Eq. (20), the general weight-error vector update equation is given by

$$E\{\tilde{\mathbf{h}}(n+1)\} = [\mathbf{I} - \mu E\{\mathbf{D}^T(n)\mathbf{X}^T(n)\}] E\{\tilde{\mathbf{h}}(n)\} \quad (28)$$

From Eq. (28), the convergence to the mean of the adaptive algorithm in Eq. (19) is guaranteed for any μ that satisfies

$$\mu \leq \frac{2}{\lambda_{\max}(E\{\mathbf{D}^T(n)\mathbf{X}^T(n)\})} \quad (29)$$

The general recursion (Eq. (21)), is stable if the matrix \mathbf{G} is stable [13]. From Eq. (35), we know that

$\mathbf{G} = \mathbf{I} - \mu \mathbf{M} + \mu^2 \mathbf{N}$, where

$$\mathbf{M} = E\{\mathbf{X}(n)\mathbf{D}(n)\} \otimes \mathbf{I} + \mathbf{I} \otimes E\{\mathbf{X}(n)\mathbf{D}(n)\} \quad \text{and}$$

$\mathbf{N} = E\{(\mathbf{X}(n)\mathbf{D}(n)) \otimes (\mathbf{X}(n)\mathbf{D}(n))\}$. The condition on μ to guarantee the convergence in the mean-square sense of the adaptive algorithms is

$$0 \leq \mu \leq \min\left\{\frac{1}{\lambda_{\max}(\mathbf{M}^{-1}\mathbf{N})}, \frac{1}{\max(\lambda(\mathbf{H}) \in \mathbf{R})}\right\} \quad (30)$$

Table 1. Family of Affine Projection Adaptive filter algorithm

Algorithm	K	D	$\mathbf{W}(n)$
NLMS	$K = 1$	$D = 1$	$\frac{1}{\ \mathbf{x}(n)\ ^2}$
ε -NLMS	$K = 1$	$D = 1$	$\frac{1}{\ \mathbf{x}(n)\ ^2}$
APA	$K \leq M$	$D = 1$	$(\mathbf{X}^T(n)\mathbf{X}(n))^{-1}$
BNDR-LMS	$K = 2$	$D = 1$	$(\mathbf{X}^T(n)\mathbf{X}(n))^{-1}$
R-APA	$K \leq M$	$D = 1$	$(\varepsilon\mathbf{I} + \mathbf{X}^T(n)\mathbf{X}(n))^{-1}$
NLMS-OCF	$K \leq M$	$D \geq 1$	$(\mathbf{X}^T(n)\mathbf{X}(n))^{-1}$

Table 2. Family of SPU Affine Projection Adaptive filter algorithm

Algorithm	K	D	$\mathbf{C}(n) / \mathbf{Z}(n)$
SPU-APA	$K \leq M$	$D = 1$	$\mathbf{C}(n) = \mathbf{A}(n)\mathbf{I}, \mathbf{Z}(n) = (\mathbf{X}^T(n)\mathbf{A}(n)\mathbf{X}(n))^{-1}$
SPU-BND-NLMS	$K = 2$	$D = 1$	$\mathbf{C}(n) = \mathbf{A}(n)\mathbf{I}, \mathbf{Z}(n) = (\mathbf{X}^T(n)\mathbf{A}(n)\mathbf{X}(n))^{-1}$
SPU-R-APA	$K \leq M$	$D = 1$	$\mathbf{C}(n) = \mathbf{A}(n)\mathbf{I}, \mathbf{Z}(n) = (\varepsilon\mathbf{I} + \mathbf{X}^T(n)\mathbf{A}(n)\mathbf{X}(n))^{-1}$
SPU-NLMS-OCF	$K \leq M$	$D \geq 1$	$\mathbf{C}(n) = \mathbf{A}(n)\mathbf{I}, \mathbf{Z}(n) = (\mathbf{X}^T(n)\mathbf{A}(n)\mathbf{X}(n))^{-1}$

where $\mathbf{H} = \begin{bmatrix} \frac{1}{2}\mathbf{M} & -\frac{1}{2}\mathbf{N} \\ \mathbf{I} & 0 \end{bmatrix}$.

7 Computational Complexity

The computational complexity of the APA and SPU-APA has been presented in Table 3. The computational complexity of the APA is from [20]. As we can see, the number of reduction in multiplications for SPU-APA is $(M - SL)(K^2 + 2K)$ which is large in some applications such as network and acoustic echo cancellation. If we save the computations from previous iteration, the number of additional multiplications in SPU-APA will be 1 multiplication. This algorithm needs $\text{Blog}_2S + O(B)$ comparisons when using the heapsort algorithm [21]. Also, the computational complexity of SPU-PRA is reduced by the factor of K, because the adaptation of the filter coefficients is performed only once every K iterations.

8 Simulation Result

We demonstrate the performance of the proposed algorithms by several computer simulations in a system identification and an acoustic echo cancellation scenarios.

8.1 System Identification

The theoretical results presented in this paper are confirmed by several computer simulations in a system identification setup. The unknown system has 16 randomly selected taps. The input signal $x(n)$ is a first order autoregressive (AR) signal generated by

$$x(n) = \rho x(n - 1) + w(n) \tag{31}$$

where $w(n)$ is either a zero mean white Gaussian signal or a zero mean uniformly distributed random sequence between -1 and 1. For the Gaussian case, the value of ρ is set to 0.9, generating a highly colored Gaussian signal. For the uniform distribution case, the value of $\frac{1}{2}$ is set to 0.5. The measurement noise $v(n)$ with $\sigma_v^2 = 10^{-3}$ is added to the noise-free desired signal

$d(n) = \mathbf{h}_t^T \mathbf{x}(n)$. The adaptive filter and the unknown channel are assumed to have the same number of taps. The stability bounds of adaptive filter algorithms were calculated from Eq. (29), and Eq. (30). Tables 4 and 5 show the stability bounds of the SPU-AP algorithms for colored Gaussian and uniform input signals and for different values of the parameters. In all simulations, the simulated learning curves are obtained by ensemble averaging over 200 independent trials. Also, the steady-state MSE is obtained by averaging over 500 steady-state samples from 500 independent realizations for each value of μ for a given algorithm.

8.1.1 Simulation Results for Transient Performance

Figs. 2-7 show the theoretical and simulated learning curves of the family of SPU-AP algorithms for different parameter values. The theoretical learning curves have been obtained from Eq. (25). The number of block (B) was set to 4. Figs. 2 and 3 show the results for different values of K (2 and 3) and for colored Gaussian input signal. In these figures different values for S (1, 2 and 4) were chosen. The step-size of SPU-APA for $B = 4$, and $S = 4$ was set to 0:4. For $S = 1$, and 2, the step-sizes were chosen to get approximately the same steady-state MSE as $S = 4$. As we can see, there is a good agreement between simulated and theoretical learning curves. In Fig. 4, we set K , and S to 4, and 3 respectively and different values for the step-size were selected. Again, there is good agreement between simulated and theoretical learning curves.

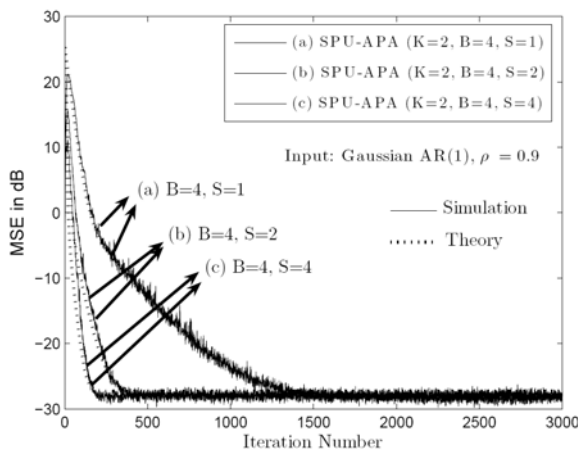


Fig. 2. Simulated and theoretical learning Curves of SPU-APA with $K = 2$, $B = 4$ and $S = 1, 2, 4$ (input: Gaussian AR(1), $\rho = 0.9$)

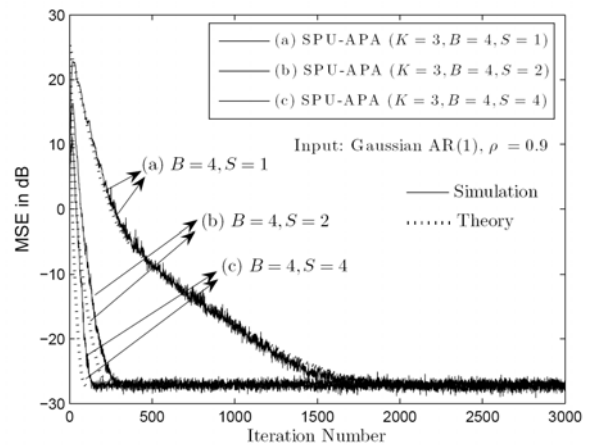


Fig. 3. Simulated and theoretical learning curves of SPU-APA with $K = 3$, $B = 4$ and $S = 1, 2, 4$ (input: Gaussian AR(1), $\rho = 0.9$)

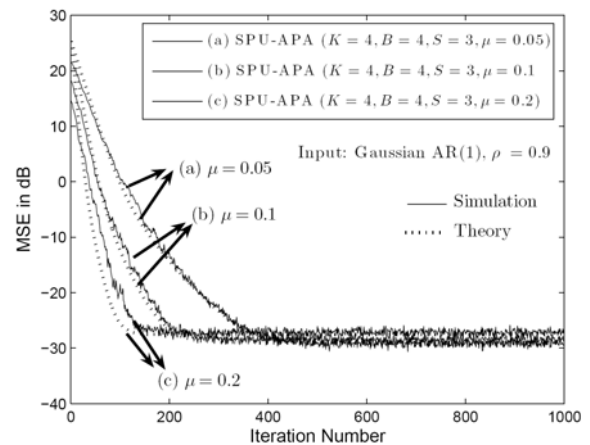


Fig. 4. Simulated and theoretical learning curves of SPU-APA with $K = 4$, $B = 4$, $S = 3$, and different values for μ (input: Gaussian AR(1), $\rho = 0.9$)

Figs. 5 and 6 show the results of SPU-NLMS-OCF algorithm for colored Gaussian input. Different values for K (2 and 4), and D (2 and 4) were used in simulations. Fig. 5 shows the results for $K = 2$, $D = 2$, and different values for S . Again for $B = 4$, and $S = 4$, the step-size was set to 0.4. For $S = 1$, and 2, the step-sizes were chosen to get approximately the same steady-state MSE as $S = 4$. Fig. 6 shows the simulated and theoretical learning curves of SPU-NLMS-OCF for $K = 4$, $D = 4$, and different values for S (2, 3 and 4). The simulation results show good agreement between simulated and theoretical learning curves. Fig. 7 shows the results for colored uniform input signal. The parameter K was set to 2 and different values for S (2, 3 and 4) were chosen. Again, good agreement between simulated and theoretical learning curves can be seen.

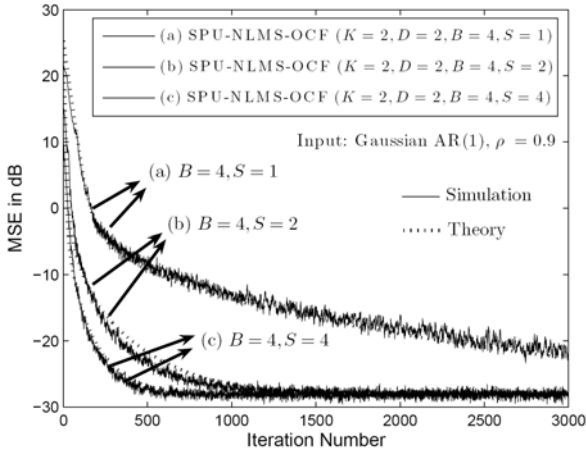


Fig. 5. Simulated and theoretical learning curves of SPU-NLMS-OCF with $K = 2$, $D = 2$, $B = 4$, and $S = 1, 2, 4$ (input: Gaussian AR(1), $\rho = 0.9$)

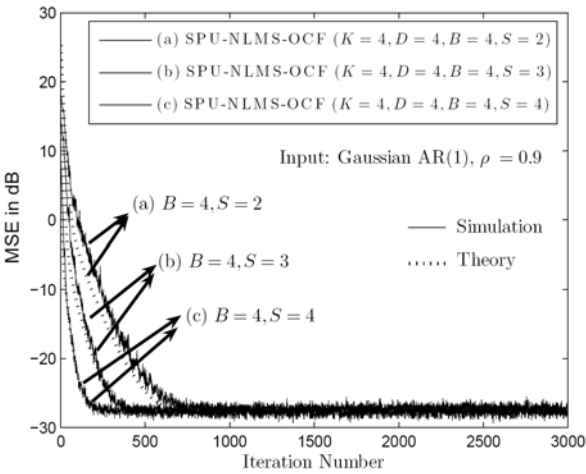


Fig. 6. Simulated and theoretical learning curves of SPU-NLMS-OCF with $K = 4$, $D = 4$, $B = 4$, and $S = 2, 3, 4$ (input: Gaussian AR(1), $\rho = 0.9$)

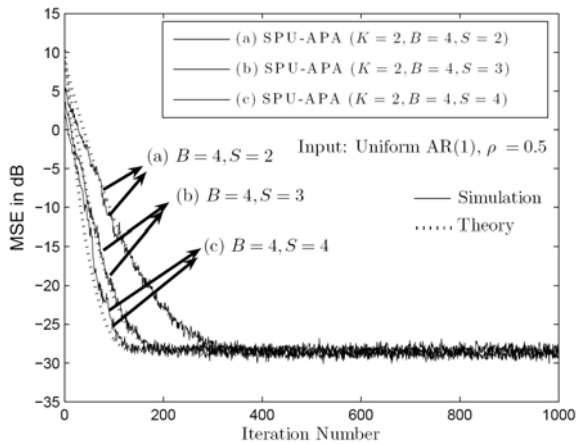


Fig. 7. Simulated and theoretical learning curves of SPU-APA with $K = 2$, $B = 4$ and $S = 2, 3, 4$ (input: uniform AR(1), $\rho = 0.5$)

8.1.2 Simulation Results for Steady-State Performance

Figs. 8-10 show the steady-state MSE of the family of SPU-AP algorithms for colored Gaussian input signal as a function of the step-size. In Fig. 8 and 9, the step-size (μ) changes from 0.04 to 0.2 for $K = 2$, $B = 4$, $S = 2$ and 0.04 to 0.5 for $K = 2$, $B = 4$, $S = 3$. These ranges guarantee the stability of the filter in this algorithm. Theoretical results are calculated from Eq. (26). The degree to which the independence assumption can be assumed valid is dependent on the value of the stepsize. This degree will decrease for the large value of the stepsize. As we can see, for the small values of the stepsize, the agreement between the theoretical and simulated steady state MSE is good. In the larger stepsizes, this agreement is not as good. Also, for the large values of $K = 4$, the agreement decreases compared with $K = 2$. Fig. 10 shows the results of SPU-NLMS-OCF algorithm for different values for D (2 and 4). The step-size (μ) changes from 0.04 to 0.5. In this simulation, the parameters K , and S were set to 4, and 3 respectively. Again good agreement can be seen between simulated and theoretical steady-state MSE.

8.1.3 Simulation Results for Mean-Square Stability

Tables 3 and 4 show the stability bounds of SPU-AP algorithms for different input signals. These values have been obtained from Eq. (29), and Eq. (30). We justified these values by presenting some simulation results. Fig. 11 shows the simulated steady-state MSE curves of SPU-AP algorithm as a function of the stepsize for colored Gaussian input. The parameter B was set to 4 and different values for S (2, 3, and 4) were selected. The step-size changes from 0.04 to μ_{\max} for each parameter adjustment. As we can see, the theoretical values for μ_{\max} show the good estimation of the stability bound of SPU-AP algorithms.

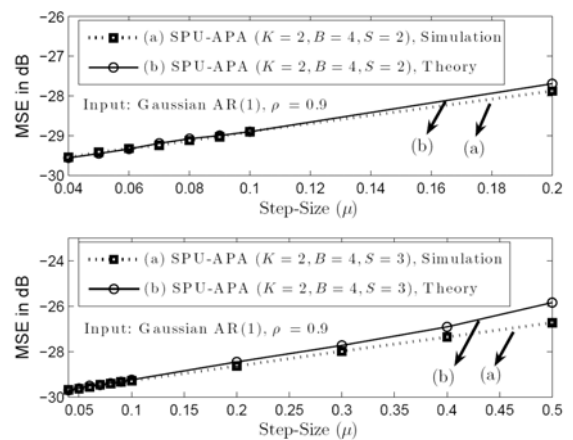


Fig. 8. Steady-state MSE of SPU-APA with $K = 2$, $B = 4$, and $S = 2, 3$ as a function of the step-size for colored Gaussian input signal.

Table 3. The computational complexity of the APA and SPU-APA

Algorithm	Multiplications	Divisions	Additional Multiplications	Comparisons
APA	$(K^2 + 2K)M + K^3 + K^2$	-	-	-
SPU-APA	$(K^2 + 2K)SL + K^3 + K^2$	-	1	$\text{Blog}_2 S + O(B)$

Table 4. Stability bounds of the SPU-APA algorithms with different parameters for colored Gaussian input

Algorithm	$\frac{2}{\lambda_{\max}(E\{D^T(n)X^T(n)\})}$	$\frac{1}{\lambda_{\max}(M^{-1}N)}$	$\frac{1}{\lambda_{\max}(M^{-1}N)}$	μ_{\max}
	SPU-APA (K = 4, B = 4, S = 1)	4.3020	0.0315	
SPU-APA (K = 4, B = 4, S = 2)	3.6542	0.6431	2.7474	0.6431
SPU-APA (K = 4, B = 4, S = 3)	3.4910	1.4723	3.2691	1.4723
SPU-APA (K = 4, B = 4, S = 4)	3.2658	2.0002	3.4391	2.0002

Table 5. Stability bounds of the SPU-APA algorithms with different parameters for colored uniform input.

Algorithm	$\frac{2}{\lambda_{\max}(E\{D^T(n)X^T(n)\})}$	$\frac{1}{\lambda_{\max}(M^{-1}N)}$	$\frac{1}{\lambda_{\max}(M^{-1}N)}$	μ_{\max}
	SPU-APA (K = 4, B = 4, S = 1)	5.9367	0.0479	
SPU-APA (K = 4, B = 4, S = 2)	6.0346	0.7693	3.4874	0.7693
SPU-APA (K = 4, B = 4, S = 3)	6.0572	1.4842	3.6545	1.4842
SPU-APA (K = 4, B = 4, S = 4)	6.0048	2.0006	3.6773	2.0006

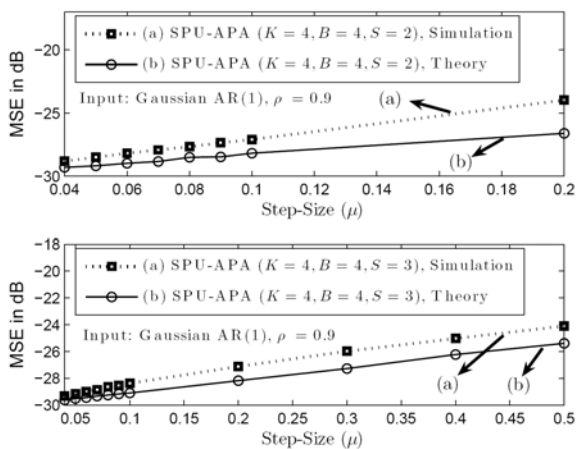


Fig. 9. Steady-state MSE of SPU-APA with $K = 4$, $B = 4$, and $S = 2, 3$ as a function of the step-size for colored Gaussian input signal.

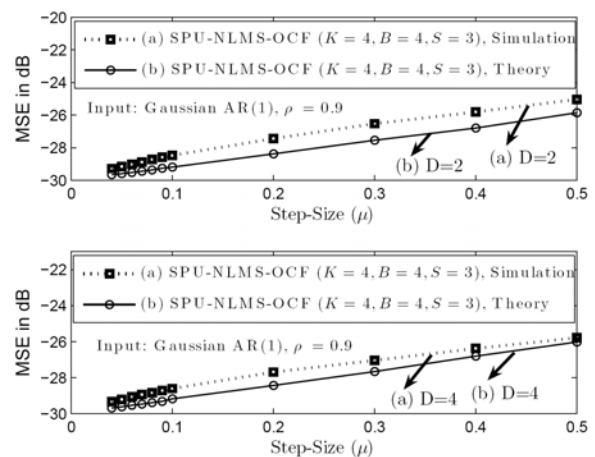


Fig. 10. Steady-state MSE of SPU-NLMS-OCF algorithm with $K = 4$ and $B = 4$, $S = 3$, and different values for $(D = 2, 4)$ as a function of the step-size for colored Gaussian input signal.

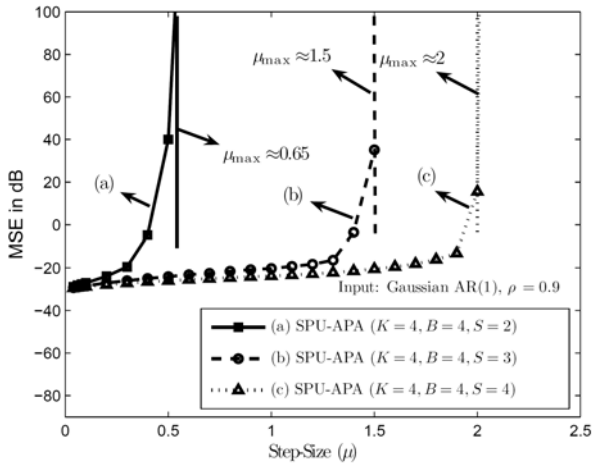


Fig. 11. Simulated steady-state MSE of SPU-APA with $K=4$, $B=4$ and $S=2, 3, 4$ as a function of the step-size for colored Gaussian input signal.

8.2 Acoustic Echo Cancellation

Fig. 12 shows the exact impulse response of the car echo path that should be identified. The number of taps in Fig. 12 is 256. The input signal is the colored Gaussian signal and the order of the filter was 256. Figs. 13-15 show the simulated learning curves of SPU-APA. In Fig. 13, the parameter K and B were set to 4 and different values for S ($S=2, 3, 4$) were used. By increasing the value of S , the convergence speed of SPU-APA will increase. For $S=3$, the convergence speed will be close to the SPU-APA with $S=4$. But the computational complexity of SPU-APA will be less than APA. Fig. 14 shows the results for $K=8$. Again, the same performance as Fig. 14 can be seen. Comparing Fig. 13 and Fig. 14 shows that by increasing the parameter K , the convergence speed increases. In Fig. 15, the performance of the SPU-APA with different values for K , and S has been compared. As we can see, the convergence speed of SPU-APA with $K=8$ will be slightly faster than SPU-APA with $K=4$ for different values of S .

Figure 16 shows the results for SPU-PRA. The parameter K , and B were set to 4. For $S=3$, the convergence speed of SPU-PRA will be close to PRA. Also, the computational complexity of SPU-PRA is reduced by the factor of K , because, the adaptation of the filter coefficients is performed only once every K iterations. Fig. 17 shows the learning curved of SPU-NLMS-OCF algorithm for different values of $D=2, 3, 4$. As we can see, by increasing the parameter D , the convergence speed of SPU-NLMS-OCF will increase due to increasing the separation and consequently reducing the correlation among the regressors in $\mathbf{X}(n)$. But for $D=3$, and 4, the convergence speed will be close together. In Fig. 18, the learning curves of SPU-APA and SPU-PRA were compared. For $S=2$, the performance of SPU-APA and SPU-PRA will be close. For $S=3$, the convergence speed of SPU-APA is faster than SPU-PRA. But the computational complexity of

SPU-PRA is less than SPU-APA. Because in SPU-PRA, the filter coefficients are partially updated only once every K iterations.

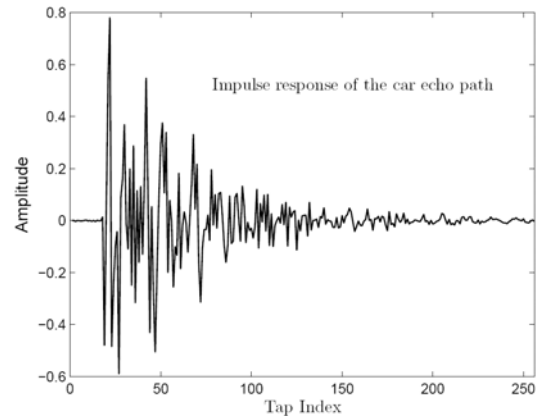


Fig. 12. Impulse response of the car echo path

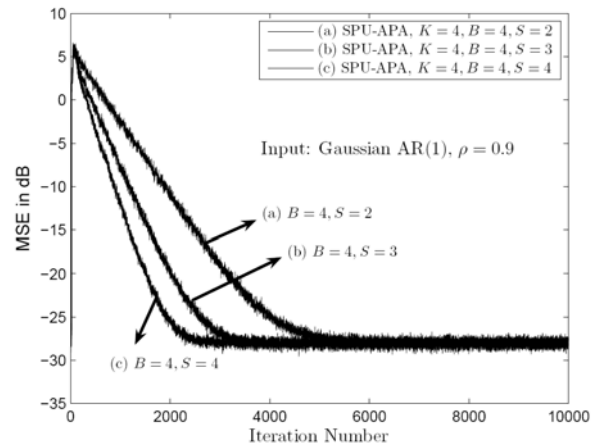


Fig. 13. Simulated learning Curves of SPU-APA with $K=4$, $B=4$ and $S=2, 3, 4$ (input: Gaussian AR(1), $\rho=0.9$)

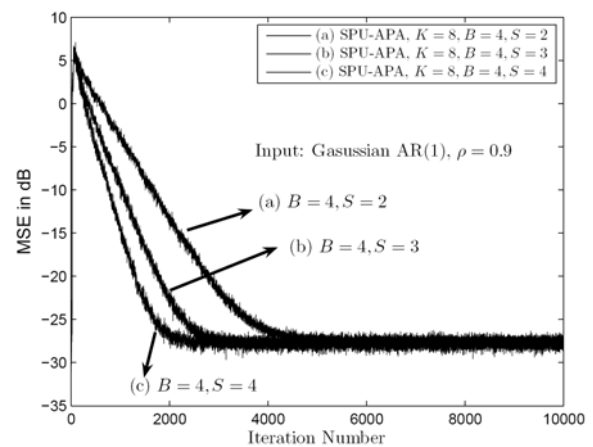


Fig. 14. Simulated learning Curves of SPU-APA with $K=8$, $B=4$ and $S=2, 3, 4$ (input: Gaussian AR(1), $\rho=0.9$)

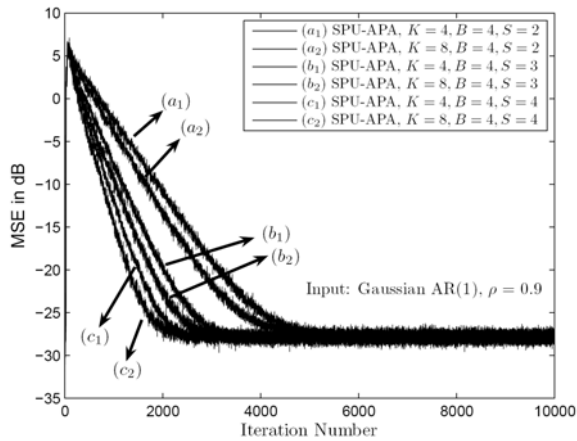


Fig. 15. Simulated learning Curves of SPU-APA with $K = 4$, $B = 4$ and $S = 2, 3, 4$ (input: Gaussian AR(1), $\rho = 0.9$)

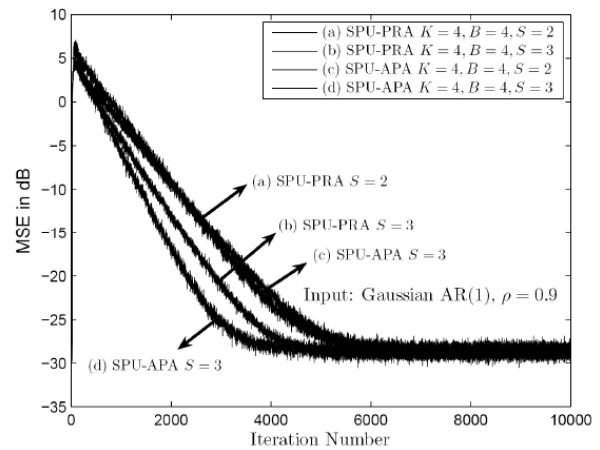


Fig. 18. Simulated learning Curves of SPU-APA and SPU-PRA with $K = 4$, $B = 4$ and $S = 2, 3$ (input: Gaussian AR(1), $\rho = 0.9$)

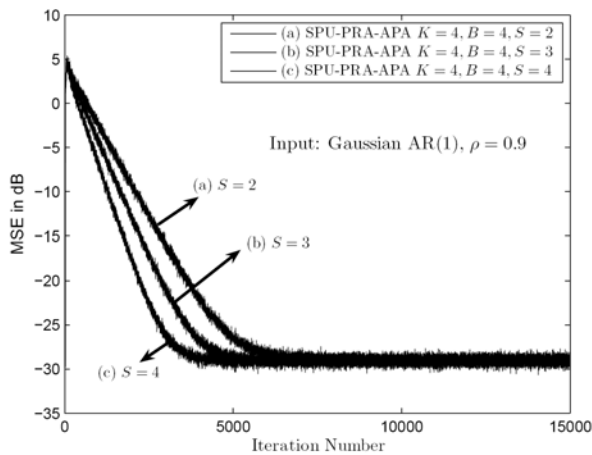


Fig. 16. Simulated Learning Curves of SPU-PRA with $K = 4$, $B = 4$, and $S = 2, 3, 4$ (input: Gaussian AR(1), $\rho = 0.9$)

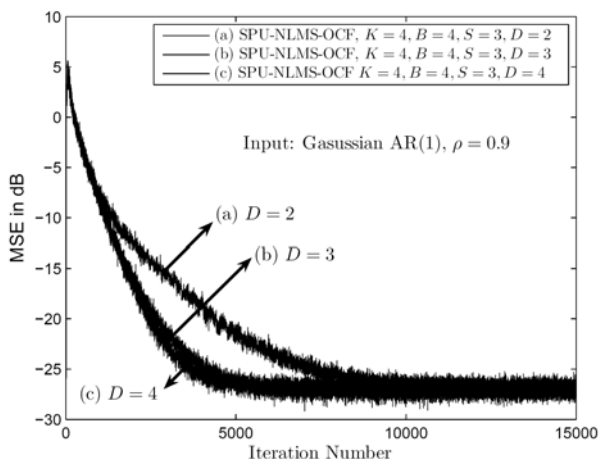


Fig. 17. Simulated learning Curves of SPU-NLMS-OCF with $K = 4$, $B = 4$, $S = 3$ and $D = 2, 3, 4$ (input: Gaussian AR(1), $\rho = 0.9$)

9 Summary and Conclusions

In this paper we presented the family of SPU affine projection algorithms. Accordingly, the mean-square performance analysis of the family of SPU-AP algorithms was presented and the general expressions for the learning curves, steady-state MSE, MSD, and mean-square stability of these algorithms were established. This analysis was based on energy conservation arguments and did not need to assume a Gaussian or white distribution for the regressors. We demonstrated the simulated and theoretical performance of the presented algorithms through several simulations in system identification and acoustic echo cancellation applications.

Acknowledgment

We would like to express our thanks to ITRC for their financial support of this project.

References

- [1] Haykin S., "Adaptive Filter Theory", NJ: Prentice-Hall, 4th edition, 2002.
- [2] Diniz P. S. R., "Adaptive Filtering: Algorithms and practical implementation", 2nd ed. Kluwer, 2002.
- [3] Sayed H., "Fundamentals of Adaptive Filtering", Wiley, 2003.
- [4] Douglas S. C., "Analysis and implementation of the max-NLMS adaptive filter," in *Proc. 29th Asilomar Conf. on Signals, Systems, and Computers, Pacific Grove, CA*, pp. 659–663 Oct. 1995.
- [5] Aboulnasr T. and Mayyas K., "Selective coefficient update of gradientbased adaptive algorithms," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing, Munich, Germany*, pp. 1929–1932, Apr. 1997.

- [6] Aboulnasr T. and Mayyas K., "Complexity reduction of the NLMS algorithm via selective coefficient update", *IEEE Trans. Signal Processing*, Vol. 47, pp. 1421–1424, May 1999.
- [7] Schertler T., "Selective block update NLMS type algorithms," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Seattle, WA, pp. 1717–1720, May 1998.
- [8] Dogancay K. and Tanrikulu O., "Adaptive filtering algorithms with selective partial updates," *IEEE Trans. Circuits, Syst. II: Analog and Digital Signal Processing*, Vol. 48, pp. 762–769, Aug. 2001.
- [9] Werner S., de Campos M. L. R., and Diniz P. S. R., "Partial-update NLMS algorithms with data-selective updating," *IEEE Trans. Signal Processing*, Vol. 52, No. 4, pp. 938–948, Apr. 2004.
- [10] Dogancay K., "Complexity considerations for transform-domain adaptive filters", *Signal Processing*, Vol. 83, pp. 1177–1192, 2003.
- [11] Abadi M. S. E. and Husøy J. H., "Mean-square performance of the family of adaptive filters with selective partial updates", *Signal Processing*, Vol. 88, pp. 2008–2018, 2008.
- [12] Al-Naffouri T. Y. and Sayed A. H., "Transient analysis of data-normalized adaptive filters", *IEEE Trans. Signal Processing*, Vol. 51, pp. 639–652, Mar. 2003.
- [13] Shin H.-C. and Sayed A. H., "Mean-square performance of a family of affine projection algorithms", *IEEE Trans. Signal Processing*, Vol. 52, pp. 90–102, Jan. 2004.
- [14] Shin H. -C., Song W. J. and Sayed A. H., "Mean-square performance of data-reusing adaptive algorithms", *IEEE Signal Processing Letters*, Vol. 12, pp. 851–854, Dec. 2005.
- [15] Gay S. L. and Benesty J., "Acoustic Signal Processing for Telecommunication", Boston, MA: Kluwer, 2000.
- [16] Apolinario J., Campos M. L. and Diniz P. S. R., "Convergence analysis of the binormalized data-reusing LMS algorithm", *IEEE Trans. Signal Processing*, Vol. 48, pp. 3235–3242, Nov. 2000.
- [17] Kratzer S. G. and Morgan D. R., "The partial-rank algorithm for adaptive beamforming", in *Proc. SPIE Int. Soc. Opt. Eng.*, pp. 9–14, 1985.
- [18] Sankaran S. G. and Beex A. A. L., "Normalized LMS algorithm with orthogonal correction factors", *Proc. Asilomar Conf. on Signals, Systems, and Computers*, pp. 1670–1673, 1997.
- [19] Husøy J. H. and Abadi M. S. E., "Unified approach to adaptive filters and their performance", *IET Signal Processing*, Vol. 2, pp. 97–109, 2008.
- [20] Sayed A. H., "Adaptive Filters". Wiley, 2008.
- [21] Knuth D. E., "Sorting and Searching", Vol. 3 of The Art of Computer Programming, 2nd ed. Reading, MA: Addison-Wesley, 1973.



Mohammad Shams Esfand-Abadi was born in Tehran, Iran, on September 18, 1978. He received the B.S. degree in Electrical Engineering from Mazandaran University, Mazandaran, Iran and the M.S. degree in Electrical Engineering from Tarbiat Modarres University, Tehran, Iran in 2000 and 2002, respectively, and the Ph.D. degree in Biomedical

Engineering from Tarbiat Modarres University, Tehran, Iran in 2007. Since 2004 he has been with the Department of Electrical Engineering, Shahid Rajaei University, Tehran, Iran. During the fall of 2003, spring 2005, and again in the spring of 2007, he was a visiting scholar with the Signal Processing Group at the University of Stavanger, Norway. His research interests include digital filter theory and adaptive signal processing algorithms.



Vahid Mehrdad was born in Khoramabad, Iran, in 1985. He received the B.S. degree in electronic engineering from Shahid Rajaei Teacher Training university, Tehran, Iran, in 2007. He is the master student in the Shahid Rajaei Teacher Training University, Department of Electrical Engineering Tehran, Iran. His research

activities include adaptive signal processing, and adaptive filtering.



Majid Noroozi was born in Rey, Iran, on February 02, 1988. He is the B.S. student of electronic engineering at Shahid Rajaei Teacher Training University. His research interests include adaptive signal processing.