

IMPROVED SET-MEMBERSHIP PARTIAL-UPDATE AFFINE PROJECTION ALGORITHM

Paulo S. R. Diniz, Fellow, IEEE, and Hamed Yazdanpanah

Universidade Federal do Rio de Janeiro
DEL/Poli & PEE/COPPE/UFRJ
P.O. Box 68504, Rio de Janeiro, RJ, 21941-972, Brazil

ABSTRACT

In this paper, we present an improved set-membership partial-update affine projection (I-SM-PUAP) algorithm, aiming at accelerating the convergence, and decreasing the update rates and the computational complexity of the set-membership partial-update affine projection (SM-PUAP) algorithm. To meet these targets, we constrain the weight vector perturbation to be bounded by a hypersphere instead of the threshold hyperplanes as in the standard algorithm. We use the distance between the present weight vector and the expected update in the standard set-membership affine projection (SM-AP) algorithm to construct the hypersphere. With this strategy, the new algorithm shows better behavior in the early iterations. Simulation results verify the excellent performance of the proposed algorithm related to the convergence rate and the required number of updates.

Index Terms— adaptive filtering, set-membership filtering, partial-update, identification problem

1. INTRODUCTION

Adaptive filters have applications in a wide range of areas such as noise cancellation, signal prediction, echo cancellation, communications, radar, and speech processing. In several applications, the large number of coefficients to be updated leads to high computational complexity, turning the adaptation of the filter coefficients prohibitive in terms of hardware requirements. In some cases, like acoustic echo cancellation, the adaptive filter might use a few thousands of coefficients in order to model the underlying physical system with sufficient accuracy. In these applications, the convergence would entail a large number of iterations, calling for more sophisticated updating rule which is inherently more computationally intensive. For a given adaptive filter, the computational complexity can be reduced by updating only part of the filter coefficients at each iteration, forming a family of algorithms called partial-update (PU) algorithms. In the literature, several variants of adaptive filtering algorithms with partial-update have been proposed [1–14].

Another powerful approach to decrease the computational complexity of an adaptive filter is to employ set-membership

filtering (SMF) approach [2, 15]. The SMF allows the reduction in computational complexity, since the filter coefficients are updated only when the estimation error is higher than a pre-determined threshold. Algorithms developed from the SMF framework employ a deterministic objective function related to a bounded error constraint on the filter output, such that the updates belong to a set of feasible solutions. Implementation of SMF algorithms involves two main steps: 1) information evaluation, 2) parameter update. As compared with the standard normalized least mean square (NLMS) and affine projection (AP) algorithms, the set-membership normalized least mean square and affine projection (SM-NLMS and SM-AP) algorithms lead to reduced computational complexity chiefly due to data-selective updates [15–22].

The use of PU strategy decreases the computational complexity while reducing convergence speed. We employ SMF technique to reduce further the computational load due to lower number of updates. However applying the SMF and PU strategies together might result in slow convergence speed. One approach to accelerate the convergence speed is choosing a smaller error estimation bound but it might increase the number of updates. Also, if we adopt a higher error estimation threshold to reduce the number of updates, the convergence rate decreases. Therefore, convergence speed and computational complexity are conflicting requirements.

In this paper, we introduce an interesting algorithm which can accelerate the convergence speed and simultaneously reduce the number of updates (and as a result decrease the computational complexity) in the SM-PUAP algorithm. In the SM-PUAP algorithm, some updates move too far from their SM-AP update; especially when the angle between the updating direction and the threshold hyperplane is small. In this case we might have a large disturbance in the coefficient update while attempting to reach the feasibility set. Therefore, to limit the distance between two consecutive updates, first we will construct a hypersphere centered at the present weight vector whose radius equals the distance between the present weight vector and the weight vector that would be obtained with the SM-AP algorithm. This radius is an upper bound on the Euclidean norm of the coefficient disturbance that is allowed in the proposed I-SM-PUAP algorithm.

The organization of the paper is as follows. In Section 2,

the concept of SMF is briefly described. Section 3 reviews the SM-PUAP algorithm. In Section 4, we derive the M-SM-PUAP algorithm. Section 5 presents simulations of the algorithms. Section 6 contains the conclusions.

2. SET-MEMBERSHIP FILTERING (SMF)

The goal of the SMF [2] is to find the filter coefficient vector \mathbf{w} such that the magnitude of the estimation error is upper bounded by a prescribed parameter $\bar{\gamma}$. Several valid estimates of \mathbf{w} satisfy the chosen bound $\bar{\gamma}$ for the estimation error at instant k . Let the *constraint set* $\mathcal{H}(k)$ denote the set consisting of all vectors \mathbf{w} such that their estimation errors at time instant k are upper bounded in magnitude by $\bar{\gamma}$, i.e.,

$$\mathcal{H}(k) = \{\mathbf{w} \in \mathbb{R}^N : |d(k) - \mathbf{w}^T \mathbf{x}(k)| \leq \bar{\gamma}\}, \quad (1)$$

where \mathbb{R} and the superscript T denote the real numbers and transpose operator, respectively. The quantities $\mathbf{x}(k)$, \mathbf{w} , and $d(k)$ are input vector, weight vector, and desired signal, respectively. The *membership set* $\psi(k)$ is defined as

$$\psi(k) = \bigcap_{i=0}^k \mathcal{H}(i). \quad (2)$$

The idea of set-membership recursion techniques is to adapt the coefficient vector such that it will always remain within the feasibility set. Due to difficulties to compute $\psi(k)$, we calculate a point estimate using, for example, the information provided by the constraint set $\mathcal{H}(k)$ like in the set-membership NLMS algorithm [15], or several previous constraint sets as is done in the set-membership affine projection algorithm [17].

3. SET-MEMBERSHIP PARTIAL-UPDATE AFFINE PROJECTION ALGORITHM

In this section we introduce the SM-PUAP algorithm [2]. The main objective in the partial-update adaptation is to perform updates in M out of N adaptive filter coefficients, where N is the order of adaptive filter. The M coefficients to be updated at time instant k are specified by an index set $\mathcal{I}_M(k) = \{i_1(k), \dots, i_M(k)\}$ with $\{i_j(k)\}_{j=1}^M$ chosen from the set $\{1, \dots, N\}$. Note that $\mathcal{I}_M(k)$ varies with the time instant k . As a result, the M coefficients to be updated can change according to the time instant. The choice of which M coefficients should be updated is related to the optimization criterion chosen for algorithm derivation. The SM-PUAP algorithm [2] takes the update vector $\mathbf{w}(k+1)$ as the vector minimizing the Euclidean distance $\|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2$ subject to the constraint $\mathbf{w}(k+1) \in \mathcal{H}(k)$ in such a way that only M coefficients are updated.

The optimization criterion in the SM-PUAP algorithm is following described. Let $\psi^L(k)$ indicate the intersection of

the last L constraint sets. A coefficient update is implemented whenever $\mathbf{w}(k) \notin \psi^L(k)$ as follows

$$\begin{aligned} & \min \|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2 \\ & \text{subject to :} \\ & \mathbf{d}_{\text{ap}}(k) - \mathbf{X}_{\text{ap}}^T(k) \mathbf{w}(k+1) = \bar{\gamma}(k) \\ & \tilde{\mathbf{C}}_{\mathcal{I}_M(k)} [\mathbf{w}(k+1) - \mathbf{w}(k)] = 0 \end{aligned} \quad (3)$$

where

$$\begin{aligned} \mathbf{d}_{\text{ap}}(k) \in \mathbb{R}^{L \times 1} & \quad \text{contains the desired output} \\ & \quad \text{from the } L \text{ last time} \\ & \quad \text{instants;} \\ \bar{\gamma}(k) \in \mathbb{R}^{L \times 1} & \quad \text{specifies the point in } \psi^L(k); \\ \mathbf{X}_{\text{ap}}(k) \in \mathbb{R}^{N \times L} & \quad \text{contains the corresponding} \\ & \quad \text{input vectors, i.e.,} \end{aligned}$$

$$\begin{aligned} \mathbf{d}_{\text{ap}}(k) &= [d(k) \ d(k-1) \ \dots \ d(k-L+1)]^T, \\ \bar{\gamma}(k) &= [\bar{\gamma}_0(k) \ \bar{\gamma}_1(k) \ \dots \ \bar{\gamma}_{L-1}(k)]^T, \\ \mathbf{X}_{\text{ap}}(k) &= [\mathbf{x}(k) \ \mathbf{x}(k-1) \ \dots \ \mathbf{x}(k-L+1)], \end{aligned} \quad (4)$$

with $\mathbf{x}(k)$ being the input-signal vector

$$\mathbf{x}(k) = [x(k) \ x(k-1) \ \dots \ x(k-N+1)]^T. \quad (5)$$

Moreover, the matrix $\tilde{\mathbf{C}}_{\mathcal{I}_M(k)} = \mathbf{I} - \mathbf{C}_{\mathcal{I}_M(k)}$ is a complementary matrix that gives $\tilde{\mathbf{C}}_{\mathcal{I}_M(k)} \mathbf{w}(k+1) = \tilde{\mathbf{C}}_{\mathcal{I}_M(k)} \mathbf{w}(k)$, which means that only M coefficients are updated. The threshold vector elements are such that $|\bar{\gamma}_i(k)| \leq \bar{\gamma}$, for $i = 0, \dots, L-1$. The matrix $\mathbf{C}_{\mathcal{I}_M(k)}$ is a diagonal matrix that identifies the coefficients to be updated at instant k , if an update is required. This matrix has M nonzero elements equal to one located at positions declared by $\mathcal{I}_M(k)$.

The updating equation of the SM-PUAP algorithm is given by

$$\mathbf{w}(k+1) = \begin{cases} \mathbf{w}(k) + \mathbf{P}(k) & \text{if } |e_0(k)| > \bar{\gamma} \\ \mathbf{w}(k) & \text{otherwise} \end{cases}, \quad (6)$$

where

$$\mathbf{P}(k) = \mathbf{C}_{\mathcal{I}_M(k)} \mathbf{X}_{\text{ap}}(k) \mathbf{P}'(k) (\mathbf{e}_{\text{ap}}(k) - \bar{\gamma}(k)), \quad (7)$$

$$\mathbf{P}'(k) = (\mathbf{X}_{\text{ap}}^T(k) \mathbf{C}_{\mathcal{I}_M(k)} \mathbf{X}_{\text{ap}}(k) + \delta \mathbf{I})^{-1}, \quad (8)$$

$$\mathbf{e}_{\text{ap}}(k) = [e_0(k) \ \dots \ e_{L-1}(k)]^T, \quad (9)$$

with $e_i(k) = d(k-i) - \mathbf{w}^T(k) \mathbf{x}(k-i)$ for $i = 0, \dots, L-1$. In the equation (8), δ and \mathbf{I} are a small positive constant and an $L \times L$ identity matrix, respectively. The diagonal matrix $\delta \mathbf{I}$ is added to the matrix to be inverted in order to avoid numerical problems in the inversion operation in the cases $\mathbf{X}_{\text{ap}}^T(k) \mathbf{C}_{\mathcal{I}_M(k)} \mathbf{X}_{\text{ap}}(k)$ is ill conditioned.

We can observe that for a fixed value of $\|\mathbf{e}_{\text{ap}}(k) - \bar{\gamma}(k)\|^2$, the value of $\|\mathbf{w}(k+1) - \mathbf{w}(k)\|^2$ is minimized if $\|\mathbf{P}'(k)\|$ is minimized. As a consequence, a natural choice for the M coefficients to be updated are those that will be multiplied by the elements of $\mathbf{X}_{\text{ap}}(k)$ with the largest norm. Figure 1 illustrates the update in SM-PUAP algorithm in \mathbb{R}^2 for $L = 1$.

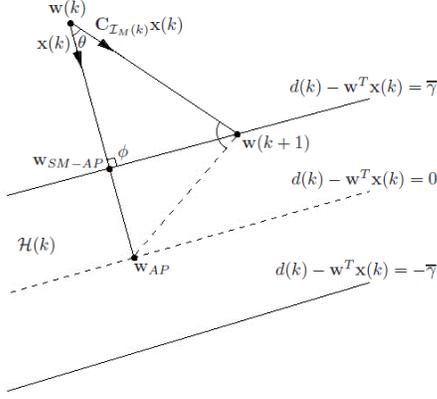


Fig. 1. Update in SM-PUAP algorithm in \mathbb{R}^2 for $L = 1$.

4. IMPROVED SET-MEMBERSHIP PARTIAL-UPDATE AFFINE PROJECTION ALGORITHM

In this section we propose the I-SM-PUAP algorithm aiming at accelerating the convergence speed of SM-PUAP algorithm and decreasing the number of updates.

Since the partial update strategy deviates the updating direction from the one determined by the input signal vector $\mathbf{x}(k)$ utilized by the SM-PUAP algorithm, it is natural that the size of the step for a partial update algorithm should be smaller than the corresponding algorithm that updates all coefficients. A solution to this problem is to constrain the Euclidean norm of the coefficient disturbance of the partial update algorithm to the disturbance implemented by the originating non partial updating algorithm, in our case the SM-AP algorithm. For that we build hypersphere, $S(k)$, whose radius is the distance between the $\mathbf{w}(k)$ and SM-AP update. The SM-AP update takes a step towards the hyperplanes $d(k) - \mathbf{w}^T \mathbf{x}(k) = \pm \bar{\gamma}$ with the minimum disturbance, i.e., when the step in the direction $\mathbf{x}(k)$ touches perpendicularly the hyperplane. Therefore, the radius of the hypersphere $S(k)$ is given by

$$\mu(k) = \min \left(\frac{|\mathbf{w}^T(k) \mathbf{x}(k) - d(k) \pm \bar{\gamma}|}{\|\mathbf{x}(k)\|_2} \right), \quad (10)$$

where $\|\cdot\|_2$ is the Euclidean norm in \mathbb{R}^N . The equation describing the hypersphere $S(k)$ with the radius $\mu(k)$ and centered at $\mathbf{w}(k)$ is as follows

$$(w_1 - w_1(k))^2 + \dots + (w_N - w_N(k))^2 = \mu^2(k). \quad (11)$$

As can be observed in Figure 1, $\mathbf{w}(k+1)$ is the point where, starting from $\mathbf{w}(k)$, the vector representing the $\mathbf{w}(k+1)$ direction touches the hyperplane $d(k) - \mathbf{w}^T \mathbf{x}(k) = \bar{\gamma}$. Unlike the SM-PUAP algorithm, in the I-SM-PUAP algorithm $\mathbf{w}(k+1)$ is the point where, starting from $\mathbf{w}(k)$, the vector representing the partial direction touches the defined N dimensional hypersphere $S(k)$ and points at a sparse version of

$\mathbf{x}(k)$. The process of deriving the I-SM-PUAP algorithm is described in Figure 2.

Define $\hat{\mathbf{w}}(k)$ as the update result of equation (6) with $\bar{\gamma}(k) = [0 \dots 0]^T$. In order to find the update of $\mathbf{w}(k)$ to the boundary of hypersphere $S(k)$ such that $\tilde{\mathbf{C}}_{\mathcal{I}_{\bar{M}}(k)} \mathbf{w}(k+1) = \tilde{\mathbf{C}}_{\mathcal{I}_{\bar{M}}(k)} \mathbf{w}(k)$ we have to find the intersection of hypersphere $S(k)$ with the line $l(k)$ passing through $\mathbf{w}(k)$ and $\hat{\mathbf{w}}(k)$. This line, shown in Figure 2, is parallel to the vector $\mathbf{u}(k) = \frac{\mathbf{a}(k)}{\|\mathbf{a}(k)\|_2}$, where $\mathbf{a}(k) = [\hat{w}_1(k) - w_1(k) \dots \hat{w}_N(k) - w_N(k)]^T$. Hence, the equation of the line $l(k)$ is given as follows

$$\begin{cases} \frac{w_1 - w_1(k)}{u_1(k)} = \dots = \frac{w_i - w_i(k)}{u_i(k)} \\ = \dots = \frac{w_N - w_N(k)}{u_N(k)}, & \text{for } i \in \mathcal{I}_{\bar{M}}(k) \\ w_i = w_i(k) & \text{for } i \notin \mathcal{I}_{\bar{M}}(k) \end{cases} \quad (12)$$

In order to find the intersection of the line $l(k)$ with the hypersphere $S(k)$, we should replace equation (12) in equation (11). Thus, we will attain $w_i = w_i(k)$ for $i \notin \mathcal{I}_{\bar{M}}(k)$, and for $i \in \mathcal{I}_{\bar{M}}(k)$ we have

$$\begin{aligned} & \frac{u_1^2(k)}{u_i^2(k)} (w_i - w_i(k))^2 + \dots + (w_i - w_i(k))^2 + \\ & \dots + \frac{u_N^2(k)}{u_i^2(k)} (w_i - w_i(k))^2 = \mu^2(k). \end{aligned} \quad (13)$$

Then,

$$(w_i - w_i(k))^2 = u_i^2(k) \mu^2(k), \quad (14)$$

where we obtained the last equality owing to $\|\mathbf{u}(k)\|_2 = 1$. Therefore, the intersections of the line $l(k)$ and the hypersphere $S(k)$ are given by

$$w_i = w_i(k) \pm u_i(k) \mu(k). \quad (15)$$

We will choose the positive sign in equation (15) since the direction of the vector $\mathbf{a}(k)$ is from $\mathbf{w}(k)$ to $\hat{\mathbf{w}}(k)$. As a result, the vector $\mathbf{w}(k+1)$ becomes as below

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \mu(k) \mathbf{u}(k). \quad (16)$$

Also, as an alternative method, we can get $\mathbf{w}(k+1)$ through an elegant geometrical view. Denote $\mathbf{w}(k+1)$ in equation (6) as $\hat{\mathbf{w}}(k)$ while taking $\bar{\gamma}(k) = [0 \dots 0]^T$. Define $\mathbf{a}(k)$ as,

$$\mathbf{a}(k) = \hat{\mathbf{w}}(k) - \mathbf{w}(k) = \mathbf{C}_{\mathcal{I}_{\bar{M}}(k)} \mathbf{X}_{\text{ap}}(k) \mathbf{P}'(k) \mathbf{e}_{\text{ap}}(k). \quad (17)$$

Since we want the update up to the boundary of hypersphere $S(k)$ centered at $\mathbf{w}(k)$ with radius $\mu(k)$ in the direction of $\mathbf{a}(k)$, we get the update equation as follows

$$\begin{aligned} \mathbf{w}(k+1) &= \mathbf{w}(k) + \mu(k) \frac{\mathbf{a}(k)}{\|\mathbf{a}(k)\|_2} \\ &= \mathbf{w}(k) + \mu(k) \mathbf{u}(k). \end{aligned} \quad (18)$$

Table 1 summarizes the I-SM-PUAP algorithm.

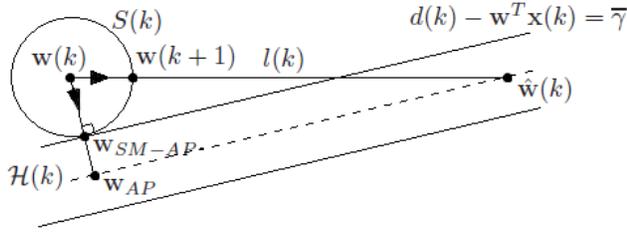


Fig. 2. Update in I-SM-PUAP algorithm in \mathbb{R}^2 for $L = 1$.

Table 1. Improved Set-Membership Partial-Update Affine Projection(I-SM-PUAP) Algorithm.

I-SM-PUAP Algorithm
Initialization $\mathbf{x}(-1) = \mathbf{w}(0) = [0 \ \dots \ 0]^T$ $\delta = \text{small positive constant}$ choose $\bar{\gamma}$ Do for $k \geq 0$ $\mathbf{e}_{\text{ap}}(k) = \mathbf{d}_{\text{ap}}(k) - \mathbf{X}_{\text{ap}}^T(k)\mathbf{w}(k)$ if $ e_0(k) > \bar{\gamma}$ $\mu(k) = \min\left(\frac{ e_0(k) \pm \bar{\gamma} }{\ \mathbf{x}(k)\ _2}\right)$ $\mathbf{a}(k) = \mathbf{C}_{\mathcal{I}_M(k)} \mathbf{X}_{\text{ap}}(k) [\mathbf{X}_{\text{ap}}^T(k) \mathbf{C}_{\mathcal{I}_M(k)} \mathbf{X}_{\text{ap}}(k) + \delta \mathbf{I}]^{-1} \mathbf{e}_{\text{ap}}(k)$ $\mathbf{w}(k+1) = \mathbf{w}(k) + \frac{\mu(k)}{\ \mathbf{a}(k)\ _2} \mathbf{a}(k)$ else $\mathbf{w}(k+1) = \mathbf{w}(k)$ end end

5. SIMULATIONS

In this section, the SM-PUAP algorithm [2] and the proposed I-SM-PUAP algorithm are applied to a system identification problem. The unknown system has order $N = 80$ and its coefficients are random scalars drawn from the standard normal distribution. The input signal is zero-mean Gaussian noise with $\sigma_x^2 = 1$. The signal-to-noise ratio (SNR) is set to 20 dB, i.e., $\sigma_n^2 = 0.01$. The bound on the output estimation error is chosen as $\bar{\gamma} = \sqrt{25\sigma_n^2}$. Also, we adopt the threshold bound vector $\bar{\gamma}(k)$ as $\bar{\gamma}_0(k) = \frac{\bar{\gamma}e_0(k)}{|e_0(k)|}$ and $\bar{\gamma}_i(k) = d(k-i) - \mathbf{w}^T(k)\mathbf{x}(k-i)$, for $i = 1, \dots, L-1$ [2, 23]. The regularization constant, δ , is 10^{-12} and $\mathbf{w}(0) = [1 \ \dots \ 1]^T$ which is not close to the unknown system. All learning curves averaged over 200 trials. We are updating 50 percent of the components randomly chosen of the filter to illustrate the partial updating, i.e., half of the elements of $\mathcal{I}_M(k)$ are nonzero at each time instant k . Figure 3 shows the learning curves for the I-SM-PUAP algorithm with $L = 2, 5$, and 70, and the learning curves for the SM-PUAP algorithm with $L = 65$ and 70. Also, in Figure 3 a blue curve is depicted using correlated inputs and $L = 2$. In fact, for the blue curve all of the specifications of the system are the same as explained above and the only difference is the input signal. The correlated input

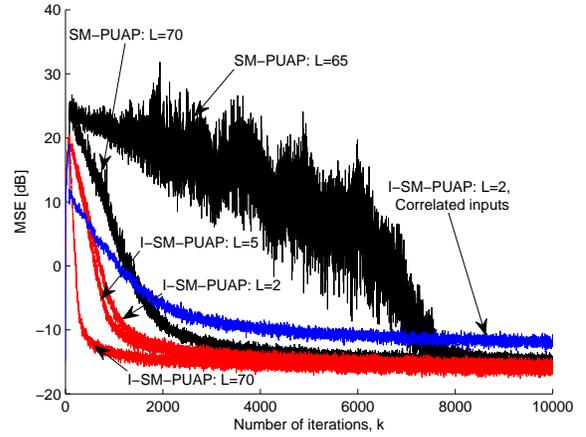


Fig. 3. Learning curves of the I-SM-PUAP algorithm applied on system identification problem.

signal is chosen as $x(k) = 0.95x(k-1) + 0.19x(k-2) + 0.09x(k-3) - 0.5x(k-4) + m(k-4)$, where $m(k)$ is a zero-mean Gaussian noise with unit variance.

The average number of updates performed by the I-SM-PUAP algorithm are 8.3%, 6.5%, and 2% for $L = 2, 5$, and 70, respectively, and 20% in the case of correlated input signal. The average number of updates implemented by the SM-PUAP algorithm are 14% and 25% for $L = 70$ and 65, respectively. Note that in both algorithms we have to find the inverse of an $L \times L$ matrix, thus large L implies high computational complexity. Therefore, the I-SM-PUAP algorithm has lower computational complexity since it presents fast convergence even for small value of L . Also, it is worth mentioning that for $L < 65$ the SM-PUAP algorithm does not reach its steady-state in 10000 iterations. From the results, we can observe that the proposed algorithm, I-SM-PUAP, has faster convergence speed, lower number of updates, and lower computational complexity as compared to the SM-PUAP algorithm.

6. CONCLUSIONS

We have introduced an improved set-membership partial-update affine projection (I-SM-PUAP) algorithm aiming at accelerating the convergence rate of the set-membership partial-update affine projection (SM-PUAP) algorithm, with lower computational complexity and reduced number of updates. In order to achieve this goal, we use the distance between the present weight vector and the one obtained with the SM-AP update, in order to provide a hypersphere that upperbounds the coefficient disturbance. Numerical simulations for system identification problem have confirmed that the I-SM-PUAP algorithm has not only faster convergence rate, but also lower computational complexity and lower number of updates as compared with the previously proposed SM-PUAP algorithm.

7. REFERENCES

- [1] K. Doğançay, *Partial-Update Adaptive Signal Processing: Design, Analysis and Implementation*, Academic Press, Elsevier, 2008.
- [2] P. S. R. Diniz, *Adaptive Filtering: Algorithms and Practical Implementation*, Springer, New York, USA, 4th edition, 2013.
- [3] S. C. Douglas, "Adaptive filters employing partial updates," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 44, no. 3, pp. 209–216, March 1997.
- [4] T. Aboulnasr and K. Mayyas, "Complexity reduction of the nlms algorithm via selective coefficient updating," *IEEE Transactions on Signal Processing*, vol. 47, no. 5, pp. 1421–1427, May 1999.
- [5] K. Doğançay and O. Tanrikulu, "Adaptive filtering algorithms with selective partial updates," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 48, no. 8, pp. 762–769, August 2001.
- [6] S. Werner, M. L. R. de Campos, and P. S. R. Diniz, "Mean-squared analysis of the partial-update nlms algorithm," *Brazilian Telecommunications Journal - SBrT*, vol. 18, pp. 77–85, June 2003.
- [7] S. Werner, M. L. R. de Campos, and P. S. R. Diniz, "Partial-update nlms algorithms with data-selective updating," *IEEE Transactions on Signal Processing*, vol. 52, no. 4, pp. 938–949, April 2004.
- [8] M. Godavarti and A. O. Hero III, "Partial update lms algorithms," *IEEE Transactions on Signal Processing*, vol. 53, pp. 2384–2399, 2005.
- [9] M. Gira and J. A. Chambers, "Adaptive partial update channel shortening in impulsive noise environments," in *15th International Conference on Digital Signal Processing*, 2007, pp. 555–558.
- [10] R. Arablouei, K. Doğançay, and S. Perreau, "Partial-update adaptive decision-feedback equalization," in *19th European Signal Processing Conference*, 2011, pp. 2205–2209.
- [11] P. S. R. Diniz, G. O. Pinto, and A. Hjørungnes, "Data selective partial-update affine projection algorithm," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2008, pp. 3833–3836.
- [12] A. Tandon, M. N. S. Swamy, and M. O. Ahmad, "Partial-update l^∞ -norm based algorithms," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 54, no. 2, pp. 411–419, 2007.
- [13] M. Z. A. Bhotto and A. Antoniou, "A new partial-update nlms adaptive-filtering algorithm," in *IEEE 27th Canadian Conference on Electrical and Computer Engineering (CCECE)*, 2014, pp. 1–5.
- [14] G. Deng, "Partial update and sparse adaptive filters," *IET Signal Processing*, vol. 1, no. 1, pp. 9–17, 2007.
- [15] S. Gollamudi, S. Nagaraj, S. Kapoor, and Y. F. Huang, "Set-membership filtering and a set-membership normalized lms algorithm with an adaptive step size," *IEEE Signal Processing Letters*, vol. 5, pp. 111–114, May 1998.
- [16] P. S. R. Diniz and S. Werner, "Set-membership binormalized data-reusing lms algorithms," *IEEE Transactions on Signal Processing*, vol. 51, no. 1, pp. 124–134, January 2003.
- [17] S. Werner and P. S. R. Diniz, "Set-membership affine projection algorithm," *IEEE Signal Processing Letters*, vol. 8, no. 8, pp. 231–235, August 2001.
- [18] R. Arablouei and K. Doğançay, "Set-membership recursive least-squares adaptive filtering algorithm," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Kyoto, Japan, March 2012, pp. 3765–3768.
- [19] M. Z. A. Bhotto and A. Antoniou, "A robust constrained set-membership affine-projection adaptive-filtering algorithm," in *5th International Symposium on Communications, Control and Signal Processing (ISCCSP)*, Rome, Italy, May 2012, pp. 1–4.
- [20] N. Takahashi and I. Yamada, "Steady-state mean-square performance analysis of a relaxed set-membership nlms algorithm by the energy conservation argument," *IEEE Transactions on Signal Processing*, vol. 57, no. 9, pp. 3361–3372, Sept. 2009.
- [21] M. Z. A. Bhotto and A. Antoniou, "Robust set-membership affine-projection adaptive-filtering algorithm," *IEEE Transactions on Signal Processing*, vol. 60, no. 1, pp. 73–81, Jan. 2012.
- [22] M. S. E. Abadi and J. H. Husoy, "Set-membership sub-band adaptive filters," in *3rd International Symposium on Communications, Control and Signal Processing (ISCCSP)*, St. Julian's, Malta, March 2008, pp. 193–196.
- [23] M. V. S. Lima and P. S. R. Diniz, "Fast learning set theoretic estimation," in *21st European Signal Processing Conference (EUSIPCO 2013)*, Marrakech, Morocco, September 2013, pp. 1–5.