SPARSITY-AWARE LEARNING IN THE CONTEXT OF ECHO CANCELATION: A SET THEORETIC ESTIMATION APPROACH

Yannis Kopsinis *Symeon Chouvardas* *Sergios Theodoridis*

[1]University of Athens Dept. Informatics & Telecomms., Athens, Greece
kopsinis@ieee.org, schouv@di.uoa.gr, stheodor@di.uoa.gr

ABSTRACT

In this paper, the set-theoretic based adaptive filtering task is studied for the case where the input signal is nonstationary and may assume relatively small values. Such a scenario is often faced in practice, with a notable application that of echo cancellation. It turns out that very small input values can trigger undesirable behaviour of the algorithm leading to severe performance fluctuations. The source of this malfunction is geometrically investigated and a solution complying with the set-theoretic philosophy is proposed. The new algorithm is evaluated in realistic echo-cancellation scenarios and compared with state–of–the–art methods for echo cancellation such as the IPNLMS and IPAPA algorithms.

***Index Terms*—** Adaptive filtering, APSM, Improved proportionate NLMS, echo cancellation

1. INTRODUCTION

Adaptive filtering and learning is among the most widely employed signal processing tasks in numerous applications, such as echo cancellation, channel equalization, prediction, etc., [1]. The Least Means Squares (LMS) algorithm and its normalized version (NLMS) are considered 'classical' in adaptive filtering community, offering robust performance with a very low computational complexity profile [1]. However, the convergence speed of the LMS, for a number of applications, might be considered to be rather slow. As an alternative, the Affine Projection Algorithm (APA), which is essentially a generalization of NLMS, has gained much attention. It deals with multiple data pairs in each adaptation step, achieving faster convergence rates, albeit at the expense of increased computational costs and potential instabilities, due to the involved matrix inversions. For the cases of sparse unknown parameter vectors, the first instance of sparsity promoting online learning algorithm traces back more than a decade ago, to the Proportionate Normalized Least Means Squares (PNLMS) algorithm, applied to the echo cancellation task, *e.g.,* [2]. Since then, this basic scheme was improved, [3], and extended to the Affine Projection Algorithm (APA) rationale [4]. The fundamental idea behind proportionate-based methods is to update each coefficient with a different step size, which is made proportional to the magnitude of the respective estimated coefficient. Recently, motivated by advancements in Compressed Sensing, LMS variants regularized with specially chosen penalties, e.g., [5] have been proposed as promising alternatives to proportionate updating for sparsity promotion.

The goal of the previous classical adaptive filtering approaches, in one way or the other, is to find the minimizer of a properly defined cost function. On the contrary, the set theoretic estimation philosophy [6], as it is adjusted to comply with the adaptive filtering case [7–11], is rather different. Each single piece of information, which reveals information concerning the position of the unknown parameter vector in the Euclidean space, can be used for the design of a closed convex subset of the space, which defines a region where, the unknown parameter vector lies in, with high probability. The goal of the set-theoretic estimation algorithms is to provide an adaptation procedure which finds a point in the common intersection of all the designed convex sets. Note also that, set-theoretic formulations of the Improved PNLMS (IPNLMS) and the improved proportionate APA (IPAPA) are able to cope for sparsity as well [11].

Set-theoretic adaptive algorithms provide large flexibility for the incorporation of a priori knowledge concerning the task at hand. However, it turns out that in some very common in practice situations, such as in echo cancellation, where the norm of the input signal might admit very small values, the performance degrades. To the best of our knowledge, a systematic way of tackling such a limitation has not been proposed so far. This is the focus of our study. The kick off point is the geometric investigation of the source of the problem and then, a solution is proposed exploiting the rich toolbox of projection operators accommodated by the set-theoretic estimation rational and particularly the Adaptive Projected Subgradient Methods (APSM).

2. PROBLEM DESCRIPTION

The adaptive filtering task based on the linear regression model is considered. In particular, a vector $\boldsymbol{a}_* \in \mathbb{R}^L$, which represents any unknown system or signal, is to be estimated using measurements that are sequentially generated as follows:

$$y_n = \boldsymbol{u}_n^T \boldsymbol{a}_* + v_n, \quad \forall n \in \mathbb{N}, \tag{1}$$

where the superscript $(\cdot)^T$ denotes the transpose of the respective vector. The model outputs (observations) $(y_n)_{n\in\mathbb{N}} \in \mathbb{R}$, and the model input vectors $(\boldsymbol{u}_n)_{n\in\mathbb{N}} \in \mathbb{R}^L$ comprise the measurement pairs $(\boldsymbol{u}_n, y_n)_{n\in\mathbb{N}}$, and $(v_n)_{n\in\mathbb{N}}$ is the noise process. The objective of adaptive filtering algorithms is to provide a sequence of estimates $\boldsymbol{a}_n$ aiming at achieving low misadjustment, $\frac{\|\boldsymbol{a}_* - \boldsymbol{a}_n\|^2}{\|\boldsymbol{a}_n\|^2}$, as fast as possible. Moreover, any extra a-priori information regarding the structure of $\boldsymbol{a}_*$, e.g.

This work is partly supported by Marie Curie IEF 302898 and by the NSRF programme, ARISTEIA ASSURANCE, co-funded by the EU and the Greek State.

the fact that it is sparse, should be effectively considered for performance improvements.

Following the set-theoretic estimation rationale, for each pair of data, which is obtained, a closed convex set is constructed, which contains the unknown vector with high probability. A popular choice for such a set takes the form of a *hyperslab* around $(\boldsymbol{u}_n, y_n)$, which is defined as:

$$S_n := \left\{ \boldsymbol{x} \in \mathbb{R}^L : |\boldsymbol{u}_n^T \boldsymbol{x} - y_n| \leq \epsilon \right\}, \quad \forall n \in \mathbb{N}, \quad (2)$$

for some user-defined tolerance $\epsilon \geq 0$, and for $\boldsymbol{u}_n \neq \boldsymbol{0}$. The parameter ϵ determines, essentially, the width of the hyperslabs. Example of two hyperslabs are shown in Fig. 1. As time evolves, new hyperslabs are formed and the true solution, $\boldsymbol{a}_*$, is contained in the common intersection of all of them. Any vector in the common intersection of all the sets, which is refereed to as feasibility set, is considered to be an acceptable solution. Therefore, the objective of the adaptive set-theoretic algorithms is to converge to a vector that belongs to the feasibility set.

3. THE ADAPTIVE PROJECTED SUBGRADIENT METHOD

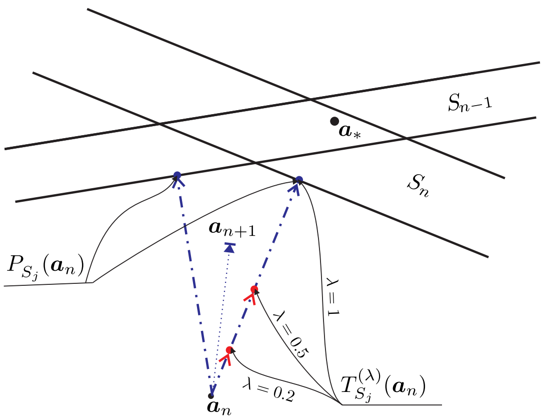

Fig. 1. APSM adaptation step

The goal of reaching a point in the feasibility set is achieved by successively projecting the currently available estimate, $\boldsymbol{a}_n$ onto the q (user-defined) more recently received hyperslabs. An example with metric projections, $P_{S_j}(\boldsymbol{a}_n)$, onto two hyperlsabs, $S_j,\ j = n,\ n-1$, is depicted in Fig. 1.

Analytically, the Euclidean metric projection of $\boldsymbol{a}_n$ onto S_j is given by,

$$P_{S_j}(\boldsymbol{a}_n) = \boldsymbol{a}_n + \begin{cases} \frac{e_j(\boldsymbol{a}_n) - \mathrm{sgn}(e_j(\boldsymbol{a}_n))\epsilon}{\|\boldsymbol{u}_j\|^2} \boldsymbol{u}_j, & \text{if } |e_j(\boldsymbol{a}_n)| > \epsilon, \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

where, the error function $e_j(\boldsymbol{a}_n) = y_j - \boldsymbol{u}_j^T \boldsymbol{a}_n$ and the $\mathrm{sgn}(\cdot)$ stands for the sign function. Starting from an arbitrary $\boldsymbol{a}_1 \in \mathbb{R}^L$, the APSM recursion step is given below:

$$\boldsymbol{a}_{n+1} := T\left(\boldsymbol{a}_n + \mu_n \frac{1}{q}\left(\sum_{j=n-q+1}^{n} (P_{S_j}(\boldsymbol{a}_n) - \boldsymbol{a}_n)\right)\right), \quad (4)$$

A big advantage of the APSM framework is that via the operator $T(\cdot)$ in (4), a priori knowledge regarding the unknown vector can be incorporated for performance improvements. Examples of using such an operator when $\boldsymbol{a}_*$ is known to be sparse, i.e., most of its components equal to zero, is given in section 5. The step–size μ_n can take any value in $(0, 2)$ or, when $q > 1$, convergence acceleration can be succeeded paying for some extra computational cost by setting $\mu_n \in (0, 2\mathcal{M}_n)$, with

$$\mathcal{M}_n := \begin{cases} \frac{\sum_{i=n-q+1}^{n} \|P_{S_i}(\boldsymbol{a}_n) - \boldsymbol{a}_n\|^2}{\|\sum_{i=n-q+1}^{n} P_{S_i}(\boldsymbol{a}_n) - \boldsymbol{a}_n\|^2}, & \\ & \sum_{i=n-q+1}^{n} P_{S_i}(\boldsymbol{a}_n) \neq \boldsymbol{a}_n, \\ 1, & \text{otherwise.} \end{cases} \quad (5)$$

In this study, a modified APSM will be adopted, which will be based on the distance between the current estimate and the hyperslabs, $d(\boldsymbol{a}_n, S_j)$. This is a key concept that is exploited later on and the basic APSM formula, (4), is rewritten using $d(\boldsymbol{a}_n, S_j) = \|\boldsymbol{a}_n - P_{S_j}(\boldsymbol{a}_n)\|$. For the case of errors $|e_j(\boldsymbol{a}_n)| > \epsilon$, we get:

$$d(\boldsymbol{a}_n, S_j) = \left\| \frac{e_j(\boldsymbol{a}_n) - \epsilon\, \mathrm{sgn}(e_j(\boldsymbol{a}_n))}{\|\boldsymbol{u}_j\|^2} \boldsymbol{u}_j \right\| \quad (6)$$

$$= \frac{|e_j(\boldsymbol{a}_n) - \epsilon\, \mathrm{sgn}(e_j(\boldsymbol{a}_n))|}{\|\boldsymbol{u}_j\|} \quad (7)$$

and when $|e_j(\boldsymbol{a}_n)| \leq \epsilon$, then $d(\boldsymbol{a}_n, S_j) = 0$. Accordingly, the APSM is rewritten as:

$$\boldsymbol{a}_{n+1} := T\left(\boldsymbol{a}_n + \mu_n \frac{1}{q}\left(\sum_{j=n-q+1}^{n} \mathrm{sgn}(e_j(\boldsymbol{a}_n)) d(\boldsymbol{a}_n, S_j) \frac{\boldsymbol{u}_j}{\|\boldsymbol{u}_j\|}\right)\right). \quad (8)$$

3.1. Case study: Input vectors with time varying statistics

In certain applications, where the statistics of the input vector vary with time, e.g., in echo cancellation, APSM as well as the rest of set-theoretic based algorithms, may exhibit performance fluctuations or even diverge in the case where the noise is not bounded. Let us see this effect with a toy example. The unknown vector is chosen to have $L = 50$ coefficients, all of them nonzero and randomly selected from $\mathcal{N}(0, 1)$. The input vectors have been selected independently at random from a Gaussian distribution and then they have been normalized to have *different* Euclidean norm values. In this way, we simulate an input with time varying statistics and particularly varying norm. The norm of the input vector is constructed such as to have fixed norm for certain intervals, starting with $\|\boldsymbol{u}_1\| = 1$ and then changing to values $0.1, 2, 0.3, 1, 0.5$ at time instants $601, 1401, 2001, 3001, 3501$. Moreover, the unknown vector changes abruptly at time instance 2500 and white Gaussian noise has been added. The results of this example are shown in Fig. 2, where all the participating algorithms have been optimized with respect to their free parameters and APSM uses $q = 1$. Of particular interest is the performance of conventional APSM (dotted curve with open circles) when compared with IPNLMS (curve denote by asterisks). Apparently, APSM diverges from the true solution whenever the *norm of the input vector assumes low values*. This is also true, but to somewhat lesser degree, for other set-theoretic algorithms, [9, 11] (curves denoted by crosses and x-crosses).

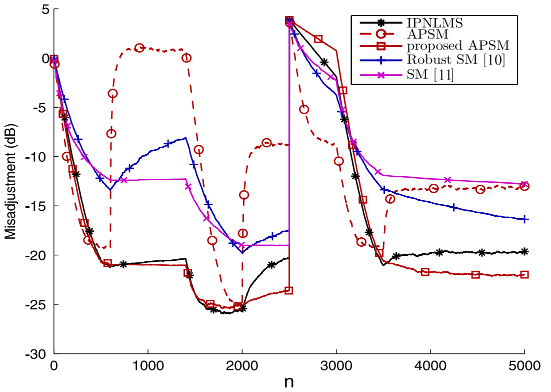

Fig. 2. Toy example with time varying input norms

IPNLMS manages to cope well with the time varying input vector via adequately tuning its regularization parameter, denoted as δ. This parameter is added to the norm of the input, which enters on the denominator of the update formula ([12]). Sometimes in bibliography, the IPNLMS regularization parameter is recommended to be given a very small value in order to avoid division by zero. However, in echo cancellation community, the value $\beta\sigma_u^2$, where β usually takes the value 20 is adopted with σ_u^2 being the variance of the input signal. It is fairly recently that the importance of δ was emphasized and a formula for its proper tuning was proposed both for the case of NLMS and APA and their proportionate counterparts [12, 13]. There it was indicated that without the proper setup of this parameter, the misalignment of the adaptive filter may fluctuate a lot and may even never converge.

In the next section, the source that causes this fluctuations problem is geometrically investigated and a solution is provided.

4. GEOMETRIC INTERPRETATION AND SOLUTION

The theoretical analysis of the APSM has shown that in the case of bounded noise, the algorithm converges *monotonically* arbitrarily close to the unknown vector with probability 1, [14]. A prerequisite for this convergence guarantee is that the hyperlabs are wide enough in order to include the unknown parameter vector. However, whenever in practice the noise is not bounded, e.g., in the additive white Gaussian case, inevitably some of the hyperslabs that will be hit by some large enough noise samples, be moved away from the true parameter vector and and $\boldsymbol{a}_*$ will no more be located inside the formed hyperslab. From (1), (2), it is easy to see that this is happening at all those time instances where $|v_n| > \epsilon$. Such hyperlabs are hereafter referred to as *missing* hyperlabs. It is clear, that the performance fluctuations problem discussed above concerns missing hyperslab only, since in their absence, monotonic convergence is attained. Next, we aim to figure out in which circumstances a missing hyperslab is "problematic" and need to be treated with care in order to avoid divergence from the true solution.

The missing hyperslabs can be characterised by their distance from the true unknown vector i.e.

$$d(\boldsymbol{a}_*, S_j) = \frac{|v_j - \epsilon\,\mathrm{sgn}\,(v_j)|}{\|\boldsymbol{u}_j\|}, \quad |v_n| > \epsilon, \tag{9}$$

where (6) and the fact that $e_j(\boldsymbol{a}_*) = y_j - \boldsymbol{u}_j^T\boldsymbol{a}_* = v_j$ has been taken into account.

From (9), it is clear that for a missing hyperslab, S_j, its distance from the ideal point $\boldsymbol{a}_*$, is getting larger *as* long as the norm of the input vector *decreases*. With respect to its distance from the current estimate $\boldsymbol{a}_n$, i.e., $d(\boldsymbol{a}_n, S_j)$, we cannot say something concrete, however, when the algorithm has converged or it is about to converge, meaning that the current estimate $\boldsymbol{a}_n$ is close to $\boldsymbol{a}_*$, then the distance $d(\boldsymbol{a}_n, S_j)$ is likely to be getting larger with decreasing norms of the input vector as well. Notice also from (8) that the APSM adaptation step is proportional to this latter distance so if the hyperslab is far from the true parameter vector, i.e. $\|\boldsymbol{u}_n\|$ is very small, then the algorithm will make a large step towards a highly misplaced hyperslab! Such hyperslabs are referred to as *unreliable* and they are the cause of the fluctuations observed in the toy example of Fig. 2. On the contrary, when the hyperslab is either not missing or missing but still close to $\boldsymbol{a}_*$, then it is beneficial for the algorithm to make a long adaptation step since it will bring the current estimate closer the the true solution. These latter hyperslabs are called, here, *reliable*.

Briefly speaking, the solution proposed in this paper is not to allow the unreliable hyperslabs to contribute much and cause large steps in the adaptation formula (8). In order to succeed in this, a proper use of relaxed projections $T_{S_j}^{(\lambda_j)}(\boldsymbol{a}_n)$ in the place of metric projections $P_{S_j}(\boldsymbol{a}_n)$ is suggested. Such a projection choice is realized for the first time in a APSM setting. A relaxed variable metric projection onto a hyperlsab is defined as follows:

$$T_{S_j}^{(\lambda_j)} = I + \lambda_j \left(P_{S_j} - I\right), \tag{10}$$

where I is the identity mapping and $\lambda_j \in (0, 2)$. Examples of relaxed projections onto a hyperlsab are depicted in Fig. 1. For $\lambda_j = 1$ it coincides with the Euclidean projection, whereas for λ_j values up to 1, the relaxed projection covers all points from $\boldsymbol{a}_n$ up to S_j across the direction of the Euclidean projection. When P_{S_j} in (8) is replaced with $T_{S_j}^{(\lambda_j)}$ yields

$$\boldsymbol{a}_{n+1} := T\left(\boldsymbol{a}_n + \mu_n \frac{1}{q}\left(\sum_{j=n-q+1}^{n} \mathrm{sgn}(e_j(\boldsymbol{a}_n)) d_{T_{S_j}^{(\lambda_j)}}(\boldsymbol{a}_n, S_j)\frac{\boldsymbol{u}_j}{\|\boldsymbol{u}_j\|}\right)\right),$$

where, it is easy to show that $d_{T_{S_j}^{(\lambda_j)}}(\boldsymbol{a}_n, S_j) = \lambda_j d(\boldsymbol{a}_n, S_j)$. It will be seen that the adoption of relaxed projections makes it possible to eliminate systematically, via parameter λ_j, the influence of hyperslabs which can be characterized as unreliable. To this end, the notion of *unreliable hyperslab* need to be quantified. It is proposed to treat as unreliable all those hyperslabs S_j, corresponding to $\|\boldsymbol{u}_j\|^2 \leq \rho$, where ρ is a user defined value. Suggestions for the choice of parameter ρ can be found at the end of this section.

Note that the most distant, from $\boldsymbol{a}_*$, hyperlabs, which still remain reliable, are those having $\|\boldsymbol{u}_j\| = \sqrt{\rho}$. The actual distance between $\boldsymbol{a}_*$ and such a "most distant" hyperslab, denoted as $\bar{d}_j$, equals to

$$\bar{d}_j = \frac{|v_j - \epsilon\,\mathrm{sgn}\,v_j|}{\sqrt{\rho}}. \tag{11}$$

In standard APSM, the maximum possible distance involved in (8) is not bounded. Indeed, the unreliable hyperslabs in standard APSM always lead to $d(\boldsymbol{a}_*, S_j) > \bar{d}_j$ and in the extreme case of $\|\boldsymbol{u}_j\| = 0$, then $d(\boldsymbol{a}_*, S_n) = \infty$. This explains the reason why standard APSM leads to performance fluctuations. In order to ameliorate such a problematic behavior, here we propose the *longest possible distance* between the current estimate and the relaxed projection onto any hyperslab $\boldsymbol{u}_j$, i.e., $\|T_{S_j}^{(\lambda_j)} - \boldsymbol{a}_n\|$ to be restricted to be equal to $\bar{d}_j$. Moreover, it is attempted via the relaxed projections, to achieve the following, more robust behaviour; when $\|\boldsymbol{u}_n\| = 0$, then $d_{T_{S_j}^{(\lambda_j)}}(\boldsymbol{a}_*, S_j) = 0$ and for all the $\|\boldsymbol{u}_n\|$ values up to $\sqrt{\rho}$, the distance $d_{T_{S_j}^{(\lambda_j)}}(\boldsymbol{a}_*, S_n)$ should increase linearly and reach the maximum value, $\bar{d}_n$. In order to succeed in the above, the relaxation parameter should be set equal to

$$\lambda_j = \min\left(\frac{\|\boldsymbol{u}_j\|^2}{\rho}, 1\right). \quad (12)$$

Proof is simple but omitted due to lack of space.

We now turn our attention to the choice of ρ. First a hyperlsab (and the corresponding input vector) can be characterized as reliable or not only with respect to the rest of the hyperslabs. This means that in a situation where most of the input vectors admit very small values or, equivalently, the input signal have small variance, then ρ should be relatively small and vice versa. As a result, ρ should be proportionate to the input signal variance. Moreover, ρ, should also depend on the SNR. When SNR is low, then, due to the increased noise variance, assuming, for example Gaussian distribution, the missing hyperlsabs are likely to appear more far from $\boldsymbol{a}_*$ compared to a high SNR scenario. As a result, parameter ρ should get smaller as long as SNR is getting larger. We have observed, that the formula proposed in [13] for the NLMS regularization parameter δ which, as it was discussed before, plays a similar role with that of ρ and complies with the characteristics discussed above. Accordingly, as a first attempt, we adopted this formula adjusted for the APSM case by multiplying it with parameter q, i.e.

$$\rho = \frac{qL(1+\sqrt{1+SNR})}{SNR}\sigma_u^2, \quad (13)$$

σ_u^2 is the variance of the input signal. Such a choice is further encouraged by the tight relation underlying APSM and NLMS. Indeed, it has been shown that when $q = 1$, $\epsilon = 0$, then NLMS coincides algorithmically with APSM [7]. As it will be seen in the simulations section the adoption of (13) leads to improved performance compared to IPNLMS (and IPAPA for the $q > 1$ case). Although this ρ option is a safe choice to use in practice, it does not mean that it is the optimum one. Perhaps an other formula, which will incorporate the peculiarities of APSM regarding the nonlinear nature of hyperlabs, would perform even better. This is let for future work.

In the toy example of Fig. 2, the performance of the proposed algorithm is shown with the curve marked with squares.

4.1. Extension to variable metric relaxed projections

Next, the performance of the proposed method will be evaluated in echo cancellation. It should be stressed, that in order to keep discussion simple and put emphasis on the novelties of this work, the relaxed projections onto hyperslabs were incorporated in the basic form of APSM involving Euclidean projections. A more suitable version of APSM, which fit well for the case of echo cancellation with sparse impulse response, is the one that uses variable metric projections [15, 16]. In order to incorporate the proposed approach of relaxed projections onto hyperslabs in the variable metric case, the metric projection operator $P_{S_j}(\cdot)$ of (4) should be replaced by the variable metric projection operator $P_{S_j}^{(\boldsymbol{G}_n)}(\cdot)$ of (3) in [15] and multiplied by λ_j derived exactly as in (12). In the simulation section that follows, matrix $\boldsymbol{G}_n$ of the variable projections is diagonal with diagonal entries $g_n(i) = \left(\frac{1-k}{2L} + (1+k)\frac{|a_n(i)|}{2\|\boldsymbol{a}_n\|_1}\right)^{-1}$. This is equal to the weighting used in IPNLMS. In this way, the proportionate philosophy is incorporated in APSM framework. Parameter, $k \in [-1, 1]$, determines to which extend the sparsity will be taken into account [3]. Setting $k = 0$ is a commonly chosen option which is also adopted here.

5. APPLICATION TO THE ECHO CANCELLATION

In the first example (Fig. 3), the low complexity implementations, i.e. the IPNLMS and the APSM with $q = 1$ are examined whereas in the second and more advanced one (Fig. 4), the focus is on APSM implementations with $q > 1$, which are evaluated against the improved proportionate APA algorithm. In both examples, the input is real voice signals sampled at 8 KHz. Moreover, the SNR is fixed to 20 dB.

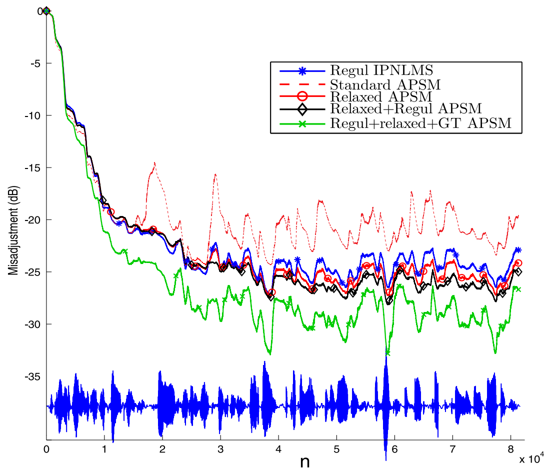

Fig. 3. Echo Cancellation example 1.

In Fig. 3, the unknown vector, $\boldsymbol{a}_*$, has $L = 1024$ coefficients and among them only $S = 100$ are non-zeros. All the curves are the result of the ensemble average of 20 realizations and in each one of them the support of $\boldsymbol{a}_*$ and the values of the nonzero components are randomly generated. The dashed curve exhibiting the worse performance corresponds to standard APSM, which takes no care for low input norms, and the solid curve marked with open circles is the proposed APSM employing relaxed instead of Euclidean projections onto hyperslabs. Asterisks indicate the IPNLMS curve, which performs slightly worse than the proposed one. Interestingly, (see curve with diamonts), a further improvement against IPNLMS is succeeded when APSM not only uses relaxed projections based on ρ, but also when it simultaneously adopts the same regularization as that employed by IPNLMS. In the case of APSM corresponds to adding the IPNLMS δ value in the denominator of (3). Finally, a significant boost in performance is achieved when the operator $T(\cdot)$ in APSM adaptation rule is replaced with a Generalized Thresholding (GT), [17] and particularly the adaptive $T_{S_{l_0}}$ as it is defined in [18]. This latter method need a rough estimate of the sparsity level, e.g. in the specific example a value twice as large as the true one, i.e. $\hat{S} = 200$, was given. Moreover, the parameter β (see [18]), was set equal to 10, however, any value between 5 to 20 would roughly lead to the same results.

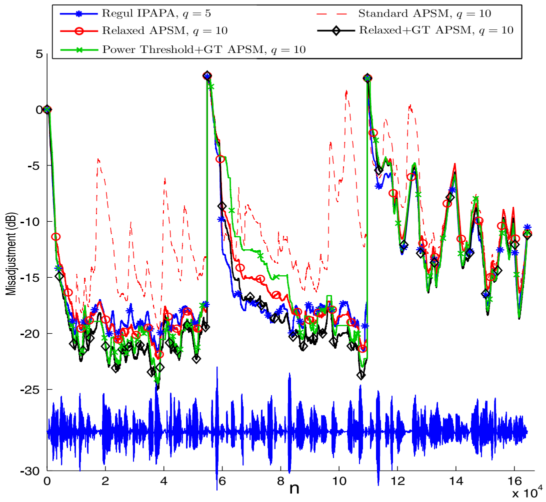

Fig. 4. Echo Cancellation example 2.

In Fig. 4, two abrupt changes are taking place splitting the example to 3 distinct parts. In the first part, the unknown vector is as those used in Fig. 3, in the second part, is the 4th echo path of the G168 Recommendation and in the last part the components are continuously varying with time. The dashed curve corresponds to the conventional APSM exhibiting even larger fluctuations compared to those observed in Fig. 3. The IPAPA performance curve is solid, marked with asterisks and the curve of the APSM with relaxed projections is marked with circles. Once again, as it is shown with the curve marked with diamonds, performance improvement is achieved when GT is employed. Finally, the curve with x-crosses shows the result when GT is used but instead of the method proposed here, the tactic to seize adaptation whenever $\|\boldsymbol{u}\|^2$ fall bellow a certain value is followed. The latter approach has been adopted in [19]. A big disadvantage is that there is not any formula or rule of thumb for the tuning of this parameter, therefor here, it was hand-tuned in order the achieve good performance with limited fluctuations. It is observed that even though it is not performing as well as the proposed one, in the first part it achieves a fairly close performance. However, in the second part a much slower convergence speed is observed.

In all the cases above, APSM processes 10 data pairs concurrently, i.e. $q = 10$, whereas IPAPA uses $q = 5$. Note however that when comes to APSM, its complexity scales linearly with q, specifically $\mathcal{O}(qL)$, in contrast to IPAPA which has complexity $\mathcal{O}(q^2 L)$. As a result in this example, the proposed APSM (without GT) is roughly half as complex as IPAPA. On top of that, the APSM framework allows the employment of subdimentional projections which are able to further decrease complexity without affecting performance [20].

REFERENCES

[1] A. H. Sayed, *Fundamentals of Adaptive Filtering*, John Wiley & Sons, New Jersey, 2003.

[2] D. L. Duttweiler, “Proportionate NLMS adaptation in echo cancelers,” *IEEE Trans. Speech Audio Processing*, vol. 8, pp. 508–518, 2000.

[3] Jacob Benesty and Steven L. Gay, “An improved PNLMS algorithm,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, May 2002, vol. 2.

[4] Constantin Paleologu, Silviu Ciochina, and Jacob Benesty, “An efficient proportionate affine projection algorithm for echo cancellation,” *Signal Processing Letters, IEEE*, vol. 17, no. 2, pp. 165–168, 2010.

[5] Y. Chen, Y. Gu, and A. O. Hero, “Sparse LMS for system identification,” in *Proceedings of the IEEE ICASSP*, 2009, pp. 3125–3128.

[6] P. L. Combettes, “The foundations of set theoretic estimation,” *Proc. IEEE*, vol. 81, no. 2, pp. 182–208, 1993.

[7] S. Theodoridis, K. Slavakis, and I. Yamada, “Adaptive learning in a world of projections,” *Signal Processing Magazine, IEEE*, vol. 28, no. 1, pp. 97–123, Jan. 2011.

[8] Y. Kopsinis, K. Slavakis, and S. Theodoridis, “Online sparse system identification and signal reconstruction using projections onto weighted ℓ_1 balls,” *IEEE Trans. Signal Proc.*, vol. 59, no. 3, pp. 905–930, Mar. 2011.

[9] M.Z.A. Bhotto and A. Antoniou, “Robust set-membership affine-projection adaptive-filtering algorithm,” *Signal Processing, IEEE Transactions on*, vol. 60, no. 1, pp. 73 –81, Jan. 2012.

[10] P. S R Diniz, R.P. Braga, and S. Werner, “Set-membership affine projection algorithm for echo cancellation,” in *Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on*, May 2006, pp. 4 pp.–.

[11] Stefan Werner, Jos A. Apolinrio,Jr., and Paulo S. R. Diniz, “Set-membership proportionate affine projection algorithms,” *EURASIP J. Audio Speech Music Process.*, vol. 2007, no. 1, pp. 1010, Jan. 2007.

[12] Constantin Paleologu, Jacob Benesty, and Silviu Ciochin, “Regularization of the affine projection algorithm,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 58, no. 6, pp. 366–370, June 2011.

[13] Jacob Benesty, Constantin Paleologu, and Silviu Ciochina, “On regularization in adaptive filtering,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 6, pp. 1734–1742, Aug. 2011.

[14] Symeon Chouvardas, Konstantinos Slavakis, Sergios Theodoridis, and Isao Yamada, “Stochastic analysis of hyperslab-based adaptive projected subgradient method under bounded noise,” *IEEE Signal Processing Letters*, vol. 20, pp. 729–732, 2013.

[15] S. Chouvardas, K. Slavakis, Y. Kopsinis, and S. Theodoridis, “A sparsity promoting adaptive algorithm for distributed learning,” *Signal Processing, IEEE Transactions on*, vol. 60, no. 10, pp. 5412–5425, oct. 2012.

[16] Masahiro Yukawa and Isao Yamada, “A unified view of adaptive variable-metric projection algorithms,” *EURASIP Journal on Advances in Signal Processing*, vol. 2009, pp. 34, 2009.

[17] K. Slavakis, Y. Kopsinis, S. Theodoridis, and S. McLaughlin, “Generalized thresholding and online sparsity-aware learning in a union of subspaces,” *Signal Processing, IEEE Transactions on*, vol. 61, no. 15, pp. 3760–3773, Aug 2013.

[18] Y. Kopsinis, K. Slavakis, S. Theodoridis, and S. McLaughlin, “Thresholding-based online algorithms of complexity comparable to sparse lms methods,” in *Circuits and Systems (ISCAS), 2013 IEEE International Symposium on*, May 2013, pp. 513–516.

[19] M. Yukawa, K. Slavakis, and I. Yamada, “Adaptive parallel quadratic-metric projection algorithms,” *IEEE Trans. Audio, Speech, and Signal Processing*, vol. 15, no. 5, pp. 1665–1680, July 2007.

[20] Y. Kopsinis, K. Slavakis, S. Theodoridis, and S. McLaughlin, “Reduced complexity online sparse signal reconstruction using projections onto weighted l1 balls,” in *2011 17th International Conference on Digital Signal Processing (DSP)*, July 2011, pp. 1 –8.