

SPARSE CONSTRAINT AFFINE PROJECTION ALGORITHM WITH PARALLEL IMPLEMENTATION AND APPLICATION IN COMPRESSIVE SENSING

Dong Yin[†] H. C. So^{*} Yuantao Gu[†]

[†]Tsinghua National Laboratory for Information Science and Technology
Tsinghua University, Beijing 100084, CHINA

^{*} Department of Electronic Engineering, City University of Hong Kong, Hong Kong, CHINA

ABSTRACT

Based on affine projection algorithm (APA) in adaptive filtering and the technique of parallel computing, we propose a novel algorithm called ℓ_0 -APA with its parallel implementation for sparse system identification and sparse signal recovery. For sparse system identification, parallel ℓ_0 -APA can serve as an effective approach for practical hardware implementation, since it lowers the requirement on the processors' clock speed. For sparse signal recovery, it can significantly reduce the convergence time. Prior algorithms such as ℓ_0 -LMS and ℓ_0 -ZAP can be seen as special cases of ℓ_0 -APA. Finally the performance of the proposed algorithm is analyzed and verified by numerical experiments.

Index Terms— Affine projection algorithm, parallel implementation, steady-state behavior, transient behavior, compressive sensing

1. INTRODUCTION

Since the unknown systems in the real world are often sparse [2, 3], there has long been considerable interest in solving sparse system identification problems [4, 5]. Sparsity indicates a small proportion of nonzero coefficients in a long unknown impulse response. Without the exploration of sparsity, traditional adaptive filtering algorithms such as least mean square (LMS), recursive least squares (RLS) and affine projection algorithm (APA) never show further effectiveness in sparse system identification. Consequently, some algorithms utilizing the prior knowledge of sparsity have been proposed in recent years. M-Max Normalized LMS (MMax-NLMS) [6], Gradient Projection (GP) [7], Sequential Partial Update LMS (S-LMS) [8] and Proportionate NLMS (PNLMS) [2, 9] are several examples.

Established in the past few years, a novel branch of sparse signal processing, compressive sensing (CS) [10, 11], gives theoretical guarantee of successful reconstruction from highly incomplete measurements. Borrowing the sparse penalty from some recovery algorithms in CS, several new methods for sparse system identification such as ZA-LMS [12] and ℓ_0 -LMS [13, 14] have been developed recently. Besides the successful application in system identification, ℓ_0 -LMS may also serve as an effective reconstruction algorithm for CS. The relationship between the adaptive filtering framework and

compressed signal recovery is addressed in [14]. By using a zero-point attraction as the sparse penalty, the above adaptive algorithms have behaved pretty well in both sparse system identification and sparse signal recovery. It has been numerically verified [14] and theoretically proved [15] that the overall performance is enhanced in both steady-state mean-square deviation (MSD) and probability of successful reconstruction, especially in the practical noisy scenarios. Very recently, there are several algorithms including TD-ZA-LMS [16] and non-uniform norm constraint LMS [17] proposed based on similar ideas.

However, the drawbacks of ℓ_0 -LMS and other similar algorithms are obvious. They are not practical approaches for hardware implementation due to their sample-by-sample iterative process. The processor has to finish updating the filter tap-weights before next sample comes in. This often results in expensive or impractical high requirements on the processors' clock speed. Designing algorithms with parallel implementation has become an effective way to solve such kind of problems [18, 19]. Based on the advancement of integrated circuits, parallel computing has been popular especially via digital processors such as FPGA and GPU. Parallel algorithms would make it possible to finish a task with lower clock frequency than their serial counterparts, and at the same time no prolonged period of time is needed due to the efficient parallel structures.

Affine projection algorithm (APA) [20] is an effective approach in adaptive filtering particularly for applications such as acoustic echo cancellation [21], active noise control [22] and distributed estimation [23]. Compared to other filtering algorithms such as LMS and normalized LMS, APA has shown better behavior in the scenarios of colored input signal [27]. Several types of fast implementation such as fast affine projection (FAP) [24] and block exact fast affine projection (BEFAP) [25], have been proposed to reduce computational complexity. On the other hand, the steady-state performance and convergence behavior of APA have been extensively studied in [26, 27].

In order to solve the difficulty in hardware implementation caused by the serial structure of available sparsity included adaptive algorithms, we propose a new sparse constraint algorithm based on APA called ℓ_0 -APA with its practical parallel implementation for sparse system identification and sparse signal recovery. We further demonstrate that the parallel implementation is more advantageous in the later than in the former. Because the reconstruction in CS is usually high-dimensional and very expensive in time complexity, parallel computing can significantly reduce the time to attain the desired results, especially in online scenarios. In addition, prior algorithms such as ℓ_0 -LMS and ℓ_0 -ZAP [14] can be regarded as special cases of the proposed algorithm. Performance analysis on ℓ_0 -APA is also conducted and justified by experiments in this work.

[†]More details of this work can be found in [1]. This work was partially supported by the National Program on Key Basic Research Project (973 Program 2013CB329201) and the National Natural Science Foundation of China (NSFC 61371137). The corresponding author of this paper is Yuantao Gu (guyt@tsinghua.edu.cn).

Table 1. Procedure of ℓ_0 -APA

Input: $\{x_n, d_n\}, M, \mu, \gamma, \alpha;$
Output: filtered result y_n , estimation error e_n , and estimated system coefficients $\tilde{\mathbf{s}}$.

Initialization: $\tilde{\mathbf{s}} = \mathbf{0}, \mathbf{w}_0 = \mathbf{0}, k = 0.$
For $n = 0, 1, 2, \dots$
 Construct \mathbf{x}_n by definition;
 $y_n = \mathbf{x}_n^T \tilde{\mathbf{s}};$
 $e_n = d_n - y_n;$
 If $\text{mod}(n, M) = 0$
 Construct \mathbf{X}_k and \mathbf{d}_k by definition;
 $\hat{\mathbf{w}}_{k+1} = \mathbf{w}_k - \gamma g(\mathbf{w}_k);$
 $\mathbf{w}_{k+1} = \hat{\mathbf{w}}_{k+1} + \mu \mathbf{X}_k^T (\mathbf{X}_k \mathbf{X}_k^T)^{-1} (\mathbf{d}_k - \mathbf{X}_k \hat{\mathbf{w}}_{k+1});$
 $k = k + 1;$
 $\tilde{\mathbf{s}} = \mathbf{w}_k;$
 End if
 Output y_n, e_n , and $\tilde{\mathbf{s}};$
End For

Table 2. Procedure of Parallel ℓ_0 -APA

Input: $\{x_n, d_n\}, M, Q, \mu, \gamma, \alpha;$
Output: filtered result y_n , estimation error e_n , and estimated system coefficients $\tilde{\mathbf{s}}$.

Initialization: $\tilde{\mathbf{s}} = \mathbf{0}, \mathbf{w}_0 = \mathbf{0}, l = 0.$
For $n = 0, 1, 2, \dots$
 Construct \mathbf{x}_n by definition;
 $y_n = \mathbf{x}_n^T \tilde{\mathbf{s}};$
 $e_n = d_n - y_n;$
 If $\text{mod}(n, M \cdot Q) = 0$
 Construct $\{\mathbf{X}_{lQ-i}, \mathbf{d}_{lQ-i}\}_{i=0,1,\dots,Q-1}$ by definition;
 $\hat{\mathbf{w}}_{l+1} = \mathbf{w}_l - \gamma g(\mathbf{w}_l);$
 Calculate in parallel for $i = 0, 1, \dots, Q-1,$
 $\Delta_{l,i} = \mathbf{X}_{lQ-i}^T (\mathbf{X}_{lQ-i} \mathbf{X}_{lQ-i}^T)^{-1} (\mathbf{d}_{lQ-i} - \mathbf{X}_{lQ-i} \hat{\mathbf{w}}_{l+1});$
 $\mathbf{w}_{l+1} = \hat{\mathbf{w}}_{l+1} + \mu \sum_{i=0}^{Q-1} \Delta_{l,i};$
 $l = l + 1;$
 $\tilde{\mathbf{s}} = \mathbf{w}_l;$
 End If
 Output y_n, e_n , and $\tilde{\mathbf{s}};$
End For

2. ℓ_0 -APA AND ITS PARALLEL IMPLEMENTATION

We denote input signal, desired signal, unknown coefficients, and adaptive tap-weights by, respectively, $x_n, d_n, \mathbf{s} \in \mathbb{R}^N$, and $\mathbf{w}_n \in \mathbb{R}^N$, where n and N are the time instant and filter length. One has $d_n = \mathbf{x}_n^T \mathbf{s} + v_n$, where $\mathbf{x}_n = [x_n, x_{n-1}, \dots, x_{n-N+1}]^T$ and v_n denote input vector and measurement noise, respectively. We follow the assumption of Block APA [25] that the adaptive algorithm works once after M new samples are collected. Therefore, the input of ℓ_0 -APA is a matrix of data

$$\mathbf{X}_k = [\mathbf{x}_{kM}, \mathbf{x}_{kM-1}, \dots, \mathbf{x}_{(k-1)M+1}]^T$$

and a vector of $\mathbf{d}_k = [d_{kM}, d_{kM-1}, \dots, d_{(k-1)M+1}]^T$ with relation $\mathbf{d}_k = \mathbf{X}_k \mathbf{s} + \mathbf{v}_k$, where $\mathbf{v}_k = [v_{kM}, v_{kM-1}, \dots, v_{(k-1)M+1}]^T$ denotes the noise vector and k is the iteration instant.

Borrowing the idea of zero-point attraction in ℓ_0 -LMS, we may modify Block APA [25] to explore the sparsity of unknown systems. The proposed ℓ_0 -APA may be summarized into a two-step procedure, i.e., a zero-point attraction on the tap-weights is conducted and followed by a projection onto the solution space of $\mathbf{d}_k = \mathbf{X}_k \mathbf{w}$. The procedure of ℓ_0 -APA is given in Table 1. In this algorithm, $g(\mathbf{w}_n) = [g(w_{n,1}), g(w_{n,2}), \dots, g(w_{n,N})]^T$ denotes the zero-point attraction, which is commonly used in [14, 15], where $g(t)$ is defined as follows,

$$g(t) = \begin{cases} -\alpha^2 t + \alpha \text{sgn}(t), & 0 < |t| \leq 1/\alpha; \\ 0, & \text{elsewhere.} \end{cases} \quad (1)$$

One may readily find that ℓ_0 -APA is a generalization of ℓ_0 -LMS. In fact, ℓ_0 -APA degenerates to the latter in the situations with $M = 1$.

Next we consider the parallel implementation of the proposed sparse constraint APA. In the parallel scenario, we assume that there are Q parallel processors that work separately and independently at the same time. Therefore, Q adjacent input matrices

$$\underline{\mathbf{X}}_l = [\mathbf{X}_{lQ}^T, \mathbf{X}_{lQ-1}^T, \dots, \mathbf{X}_{(l-1)Q+1}^T]^T$$

and desired vectors $\underline{\mathbf{d}}_l = [\mathbf{d}_{lQ}^T, \mathbf{d}_{lQ-1}^T, \dots, \mathbf{d}_{(l-1)Q+1}^T]^T$ may be processed simultaneously, where l denotes the parallel iteration number. Therefore, after one step of zero-point attraction, the projections to respective solution spaces of $\{\mathbf{d}_{lQ-i} = \mathbf{X}_{lQ-i} \mathbf{w}\}_{i=0,1,\dots,Q-1}$ are implemented on separated processors, while the increments of all these projections are summed up at the end of the iteration to finally update the tap-weights. The detailed procedures of Parallel ℓ_0 -APA are in Table 2, where $\Delta_{l,i}$ denotes the increment of the projection in $(i+1)$ th processor at l th parallel iteration.

As stated in Section 1, the disadvantages of serial algorithms including ℓ_0 -LMS and ℓ_0 -APA are obvious. The single processor has to finish all task, hence it results in a high requirement on the processor's clock frequency. In the parallel computing scenario, Q processors may work together to update the tap-weights. Therefore, the clock frequency of parallel implementation may reduce to $1/Q$ of the serial one. Actually the parallel algorithm is a trade-off between the area and the clock frequency of the processors. We may notice that it is inevitable for the parallel algorithm to have time delay as it updates the tap-weights less frequently. However, since we can still use the latest tap-weights to filter the inputs, the time delay for filtered output y_n and estimation error e_n is not a significant sacrifice.

3. PERFORMANCE ANALYSIS

Since parallel ℓ_0 -APA is equivalent to ℓ_0 -APA mathematically when we choose $Q = 1$, we only consider the theoretical analysis of parallel ℓ_0 -APA in this section.

3.1. Assumptions

As is stated in [15], the unknown coefficients could be classified into three groups: large, small, and zero. The sets of the indices of these groups are denoted by C_L, C_S , and C_o respectively. We assume that entries of \mathbf{X}_k and \mathbf{v}_k follow i.i.d. Gaussian distribution $\mathcal{N}(0, P_a)$

and $\mathcal{N}(0, P_v)$ respectively. The assumptions in [15] are adopted in this work in order to simplify the analysis. In addition, we add a new assumption that $(\mathbf{X}_k \mathbf{X}_k^T)^{-1} \approx \mathbf{I}/NP_a$.

3.2. Steady-state Performance

For simplicity we take $\mu = 1$ in the following sections. Denote the deviation of \mathbf{w}_n by \mathbf{h}_n , i.e., $\mathbf{h}_n = \mathbf{w}_n - \mathbf{s}$. And we use $\Sigma_{L,n} = \sum_{i \in C_L} E\{h_{n,i}^2\}$ and $\Sigma_{S,n} = \sum_{i \in C_S} E\{h_{n,i}^2\}$ to denote expectation of the sum of the squared deviations of the first two groups. Here E denotes the expectation operator. For the zero coefficient group, $w = (E\{h_{\infty,i}^2\})^{1/2}$ ($i \in C_o$) is adopted to represent the deviation of the tap-weights. Another three notations we will utilize are $D_n = E\{\|\mathbf{h}_n\|_2^2\}$, $P_n = E\{\mathbf{h}_n^T g(\mathbf{w}_n)\}$, and $G_n = E\{\|g(\mathbf{w}_n)\|_2^2\}$. Armed with these assumptions and notations, the following theorem on the steady-state behavior of parallel ℓ_0 -APA is derived.

Theorem 1 *Let*

$$\mathbf{S}_\infty = [\Sigma_{L,\infty} \quad \Sigma_{S,\infty} \quad D_\infty \quad P_\infty \quad G_\infty]^T.$$

Then \mathbf{S}_∞ and w are approximately determined by the following linear and quadratic equations

$$\mathbf{D}(w) = \mathbf{B}\mathbf{S}_\infty, \quad (2)$$

$$d_2 w^2 + d_1 w + e(D_\infty - 2\gamma P_\infty + \gamma^2 G_\infty) + f = 0, \quad (3)$$

where the matrix \mathbf{B} , vector $\mathbf{D}(w)$, scalars d_1 , d_2 , e , and f are given in subsection 7.1.

The proof of this theorem is in section 7. Note that D_n is the mean-square deviation (MSD) of parallel ℓ_0 -APA after the n th update and consequently D_∞ is the steady-state MSD. Since the highest degree of the polynomials of w appeared in the equations is 2, these equations can be solved analytically. Then we can get the steady-state MSD.

Due to the limited space this paper, the analysis of transient behavior is omitted. More details can be found in [1].

4. APPLICATION IN COMPRESSIVE SENSING

In CS, the problem of reconstructing a sparse signal $\mathbf{z} \in \mathbb{R}^N$ from the highly incomplete measurements $\mathbf{y} = \mathbf{A}\mathbf{z} + \mathbf{v}$, where $\mathbf{A} \in \mathbb{R}^{M \times N}$ and $M \ll N$ is considered. The relationship between adaptive filtering framework and compressive sensing has been addressed by [14].

To make reasonable comparison, we divide \mathbf{A} into Q sub-matrices $\mathbf{A}_0, \mathbf{A}_1, \dots, \mathbf{A}_{Q-1}$ of size $M/Q \times N$ and divide \mathbf{y} and \mathbf{v} into \mathbf{y}_i and \mathbf{v}_i ($0 \leq i \leq Q-1$) correspondingly. Then there is $\mathbf{y}_i = \mathbf{A}_i \mathbf{z} + \mathbf{v}_i$. For ℓ_0 -APA, we choose $\mathbf{X}_k = \mathbf{A}_{\text{mod}(k,Q)}$, which means the Q sub-matrices are put into the adaptive filtering framework circularly. In the scenarios of $Q = M$ and $Q = 1$, ℓ_0 -APA becomes the ℓ_0 -LMS and ℓ_0 -ZAP respectively. Therefore, ℓ_0 -APA is also a general framework for prior algorithms for CS. For parallel ℓ_0 -APA, we choose $\mathbf{X}_l = \mathbf{A}$ for all $l \geq 0$, which means the Q sub-matrices are processed in parallel in each iteration. In this way, the serial processor and each independent processor of the parallel algorithm share the same computational complexity since they all deal with an $M/Q \times N$ sub-matrix. When the two processors have the same clock frequency, the parallel algorithm will converge much faster because Q projections are conducted at the same time. This will be illustrated by experiments in section 5.

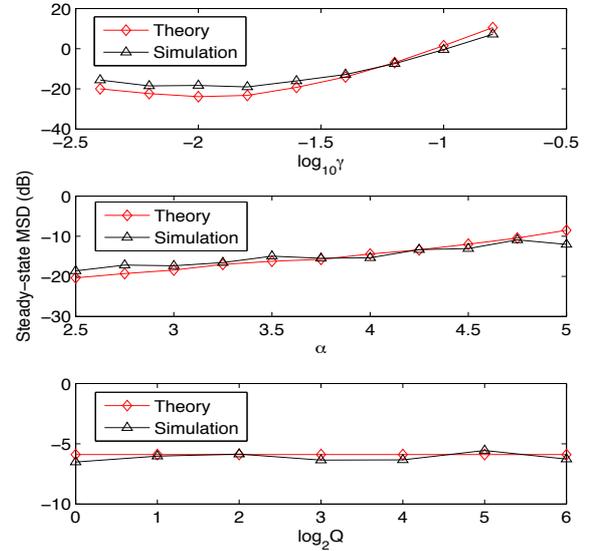


Fig. 1. Steady-state MSD versus γ , α , and Q .

5. NUMERICAL EXPERIMENTS

Experiments are conducted to justify the presented results.¹

Figure 1 illustrates the accuracy of the steady-state performance analysis in Theorem 1. In this experiment, we investigate how the steady-state MSD changes with parameters γ , α , and Q and compare the results by the simulations and Theorem 1. Note that in the experiment with Q , we keep the product of Q and M the same. This can illustrate the relationship between the steady-state MSD and the number of parallel processors when the update frequency is kept the same. From the notations in section 7.1, we know that the steady state MSD is a function of the product of Q and M , and the experiment verifies this since the steady-state MSD does not change significantly with Q .

Figure 2 shows that for CS, the probability of exact reconstruction of parallel ℓ_0 -APA is approximately the same as prior algorithms such as ℓ_0 -LMS and ℓ_0 -ZAP. This figure implies that we may need less measurements to recover a sparse signal using the adaptive filtering framework than referred methods.

Figure 3 demonstrates that for CS, to achieve the same steady-state MSD, the parallel ℓ_0 -APA has the highest convergence speed among ℓ_0 -LMS, ℓ_0 -ZAP, and ℓ_0 -APA. This advantage naturally comes from the parallel structure.

6. CONCLUSIONS

The proposed algorithm ℓ_0 -APA is a general framework for prior algorithms and its parallel implementation is an effective approach to sparse system identification and CS. For sparse system identification, parallel ℓ_0 -APA lowers the requirements on the clock speed of

¹The code for these experiments is available at <http://gu.ee.tsinghua.edu.cn/publications#ydl>

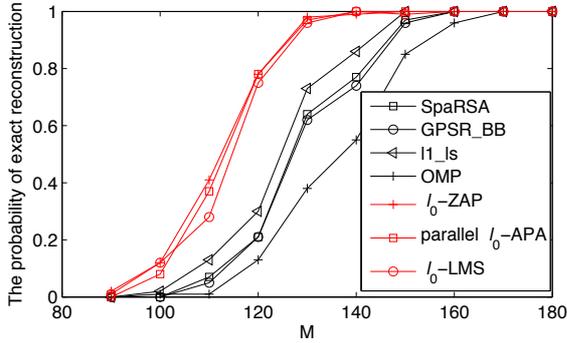


Fig. 2. Probability of exact reconstruction versus M .

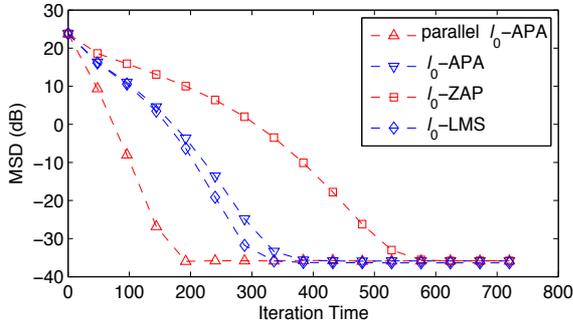


Fig. 3. Comparison of CS performance.

the processor, and for CS, the parallel algorithm enhances the convergence speed. The performance analysis in this work is also in good accordance with the simulation results.

7. APPENDIX

7.1. Notations

In this section we give the detailed expressions of \mathbf{B} , $\mathbf{D}(w)$, d_1 , d_2 , e , and f . First denote the cardinality of \mathcal{C}_L , \mathcal{C}_S , and \mathcal{C}_o by N_L , N_S , and N_o . Then we have

$$d_1 = -2\sqrt{2/\pi}\gamma\alpha C_1(1 + \gamma\alpha^2), \quad d_2 = C_1 C_3 - 1,$$

$$e = MQ/N^2, \quad f = eP_v/P_a + C_1\gamma^2\alpha^2,$$

$$\mathbf{B} = \begin{bmatrix} C_1 - 1 & 0 & eN_L & -2\gamma eN_L & \gamma^2 eN_L \\ 0 & C_1 C_3 - 1 & eN_S & -2\gamma eN_S & \gamma^2 eN_S \\ 1 & 1 & -1 & 0 & 0 \\ 0 & \alpha^2 & 0 & 1 & 0 \\ 0 & -\alpha^4 & 0 & 0 & 1 \end{bmatrix},$$

$$\mathbf{D}(w) = \begin{bmatrix} -eP_v N_L / P_a \\ C_1(C_4 C_6 - \gamma^2 C_5) - eP_v N_S / P_a \\ -N_o w^2 \\ C_6 + N_o \alpha (\sqrt{2/\pi} w - \alpha w^2) \\ C_5 - 2\alpha^2 C_6 + N_o \alpha^2 (\alpha^2 w^2 - 2\sqrt{2/\pi} \alpha w + 1) \end{bmatrix},$$

where $C_1 = 1 - 2MQ/N + MQ(MQ + 1)/N^2$, $C_2 = (1 + \gamma\alpha^2)(1 - MQ/N)$, $C_3 = 1 + 2\gamma\alpha^2 + \gamma^2\alpha^4$, $C_4 = 2\gamma(1 + \gamma\alpha^2)$,

$$C_5 = \|g(\mathbf{s})\|_2^2, \text{ and } C_6 = \gamma(1 - MQ/N)C_5/(C_2 - 1).$$

7.2. Proof of Theorem 1

Proof: The update principle given by parallel ℓ_0 -APA is

$$\mathbf{w}_{l+1} = \hat{\mathbf{w}}_l + \sum_{i=0}^{Q-1} \mathbf{X}_{lQ-i}^T (\mathbf{X}_{lQ-i} \mathbf{X}_{lQ-i}^T)^{-1} (\mathbf{d}_{lQ-i} - \mathbf{X}_{lQ-i} \hat{\mathbf{w}}_l).$$

Considering $\mathbf{w}_{l+1} = \mathbf{h}_{l+1} + \mathbf{s}$, $\hat{\mathbf{w}}_{l+1} = \mathbf{w}_l - \gamma g(\mathbf{w}_l)$, $\mathbf{d}_{lQ-i} = \mathbf{X}_{lQ-i} \mathbf{s} + \mathbf{v}_{lQ-i}$, and using the assumption that $(\mathbf{X}_{lQ-i} \mathbf{X}_{lQ-i}^T)^{-1} \approx \mathbf{I}/N P_a$, we know that the following equation holds approximately

$$\begin{aligned} \mathbf{h}_{l+1} &= \mathbf{h}_l - \gamma g(\mathbf{w}_l) \\ &+ \frac{1}{N P_a} \sum_{i=0}^{Q-1} \mathbf{X}_{lQ-i}^T (-\mathbf{X}_{lQ-i} (\mathbf{h}_l - \gamma g(\mathbf{w}_l)) + \mathbf{v}_{lQ-i}). \end{aligned}$$

By taking expectation on $\mathbf{h}_{l+1} \mathbf{h}_{l+1}^T$ and after a series of derivation, we get

$$\begin{aligned} \mathbb{E}\{\mathbf{h}_{l+1} \mathbf{h}_{l+1}^T\} &= \left(1 - \frac{2MQ}{N} + \frac{(Q-1)QM^2}{N^2}\right) E_{l,0} \\ &+ \frac{Q}{N^2 P_a^2} (E_{l,1} - \gamma E_{l,2} + \gamma^2 E_{l,3}) + \frac{MQ P_v}{N^2 P_a}, \end{aligned}$$

where

$$\begin{aligned} E_{l,0} &= \mathbb{E}\{\mathbf{h}_l \mathbf{h}_l^T\} - \gamma \mathbb{E}\{g(\mathbf{w}_l) \mathbf{h}_l^T, \\ &\quad + \mathbf{h}_l g(\mathbf{w}_l)^T\} + \gamma^2 \mathbb{E}\{g(\mathbf{w}_l) g(\mathbf{w}_l)^T\}, \\ E_{l,1} &= \mathbb{E}\{\mathbf{X}_{lQ}^T \mathbf{X}_{lQ} \mathbf{h}_l \mathbf{h}_l^T \mathbf{X}_{lQ}^T \mathbf{X}_{lQ}\}, \\ E_{l,2} &= \mathbb{E}\{\mathbf{X}_{lQ}^T \mathbf{X}_{lQ} (\mathbf{h}_l g(\mathbf{w}_l)^T + g(\mathbf{w}_l) \mathbf{h}_l^T) \mathbf{X}_{lQ}^T \mathbf{X}_{lQ}\}, \\ E_{l,3} &= \mathbb{E}\{\mathbf{X}_{lQ}^T \mathbf{X}_{lQ} g(\mathbf{w}_l) g(\mathbf{w}_l)^T \mathbf{X}_{lQ}^T \mathbf{X}_{lQ}\}. \end{aligned}$$

Since the columns of \mathbf{X}_{lQ}^T are \mathbf{x}_{lQM-i} , $0 \leq i \leq M-1$, there is

$$\begin{aligned} E_{l,1} &= \mathbb{E}\left\{\left(\sum_{i=0}^{M-1} \mathbf{x}_{lQ-i} \mathbf{x}_{lQ-i}^T\right) \mathbf{h}_l \mathbf{h}_l^T \left(\sum_{i=0}^{M-1} \mathbf{x}_{lQ-i} \mathbf{x}_{lQ-i}^T\right)\right\} \\ &= M \mathbb{E}\left\{\mathbf{x}_{lQ} \mathbf{x}_{lQ}^T \mathbf{h}_l \mathbf{h}_l^T \mathbf{x}_{lQ} \mathbf{x}_{lQ}^T\right\} + M(M-1) P_a^2 \mathbb{E}\{\mathbf{h}_l \mathbf{h}_l^T\} \\ &= M \left[P_a \text{tr}(P_a \mathbb{E}\{\mathbf{h}_l \mathbf{h}_l^T\}) \mathbf{I} + 2P_a^2 \mathbb{E}\{\mathbf{h}_l \mathbf{h}_l^T\} \right] \\ &\quad + M(M-1) P_a^2 \mathbb{E}\{\mathbf{h}_l \mathbf{h}_l^T\} \\ &= M P_a^2 \left[D_{l-1} \mathbf{I} + (M+1) \mathbb{E}\{\mathbf{h}_l \mathbf{h}_l^T\} \right], \end{aligned}$$

where $\text{tr}(\cdot)$ denotes the trace of matrices. Similar results can be derived for $E_{l,2}$ and $E_{l,3}$. By substituting the expressions of $E_{l,1}$, $E_{l,2}$ and $E_{l,3}$ into $\mathbb{E}\{\mathbf{h}_{l+1} \mathbf{h}_{l+1}^T\}$ and taking out the i th element on the diagonal of $\mathbb{E}\{\mathbf{h}_{l+1} \mathbf{h}_{l+1}^T\}$, we get

$$\begin{aligned} \mathbb{E}\{h_{l+1,i}^2\} &= C_1 (\mathbb{E}\{h_{l,i}^2\} - 2\gamma \mathbb{E}\{h_{l,i} g(w_{l,i})\} + \gamma^2 \mathbb{E}\{g(w_{l,i})^2\}) \\ &\quad + \frac{MQ}{N^2} (D_{l-1} - 2\gamma P_{l-1} + \gamma^2 G_{l-1}) + \frac{MQ P_v}{N^2 P_a}, \end{aligned}$$

where C_1 is defined in subsection 7.1. By analyzing the situations when $i \in \mathcal{C}_L$, $i \in \mathcal{C}_S$, and $i \in \mathcal{C}_o$ respectively with the method in [15], we arrive at the conclusion. \square

8. REFERENCES

- [1] D. Yin, H. C. So, and Y. Gu, "Parallel affine projection algorithm for sparse system identification and compressed signal recovery," submitted to IEEE for possible publication, available at <http://gu.ee.tsinghua.edu.cn/publications#yd2>.
- [2] D. L. Duttweiler, "Proportionate normalized least-mean-square adaptation in echo cancellers," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 5, pp. 508-518, Sept. 2000.
- [3] W. F. Schreiber, "Advanced television systems for terrestrial broadcasting: some problems and some proposed solutions," *Proc. IEEE*, vol. 83, no. 6, pp. 958-981, Jun. 1995.
- [4] R. K. Martin, W. A. Sethares, R. C. Williamson and C. R. J. Jr., "Exploiting sparsity in adaptive filters," *IEEE Trans. Signal Process.*, vol. 50, no. 8, pp. 1883-1894, Aug. 2002.
- [5] B. D. Rao and B. Song, "Adaptive filtering algorithms for promoting sparsity," in *Proc. ICASSP*, pp. 361-364, Jun. 2003.
- [6] M. Abadi, and J. Husoy, "Mean-square performance of the family of adaptive filters with selective partial updates," *Signal Processing*, vol. 88, no. 8, pp. 2008-2018, Aug. 2008.
- [7] M. A. T. Figueiredo, R. D. Nowak, and S. J. Wright, "Gradient projection for sparse reconstruction: application to compressed sensing and other inverse problems," *IEEE J. Sel. Topics Signal Process.*, vol. 1, no. 4, Dec. 2007.
- [8] M. Godavarti, and A. O. Hero, "Partial update LMS algorithms," *IEEE Trans. Signal Process.*, vol. 53, no. 7, pp. 2382-2399, Jul. 2005.
- [9] H. Deng, and R. A. Dyba, "Partial update PNLMS algorithm for network echo cancellation," in *Proc. ICASSP*, pp. 1329-1332, Taiwan, Apr. 2009.
- [10] E. J. Candes, J. K. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2 pp. 489-509, Feb. 2006.
- [11] E. J. Candes, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Commun. Pure Appl. Math.*, vol. 59, no. 8, pp. 1207-1223, Aug. 2006.
- [12] Y. Chen, Y. Gu, and A. O. Hero, "Sparse LMS for system identification," in *Proc. ICASSP*, pp. 3125-3128, Taiwan, Apr. 2009.
- [13] Y. Gu, J. Jin, and S. Mei, " ℓ_0 Norm constraint LMS algorithm for sparse system identification," *IEEE Signal Process. Lett.*, vol. 16, no. 9, pp. 774-777, Sep. 2009.
- [14] J. Jin, Y. Gu, and S. Mei, "A Stochastic gradient approach on compressive sensing signal reconstruction based on adaptive filtering framework," *IEEE J. Sel. Topics Signal Process.*, vol. 4, no. 2, pp. 409-420, Apr. 2010.
- [15] G. Su, J. Jin, Y. Gu and J. Wang, "Performance analysis of ℓ_0 norm constraint least mean square algorithm," *IEEE Trans. Signal Process.*, vol. 60, no. 5, May. 2012.
- [16] K. Shi, and X. Ma, "Transform domain LMS algorithms for sparse system identification," in *Proc. ICASSP*, pp. 3714-3717, Mar. 2010.
- [17] F. Y. Wu, and F. Tong, "Non-uniform norm constraint LMS algorithm for sparse system identification," in *IEEE Commun. Lett.*, vol. 17, no. 2, pp. 385-388, Feb. 2013.
- [18] W. T. Padgett, "Efficient parallel implementation of the LMS algorithm on a multi-ALU architecture," in *Proc. ICASSP*, vol. 4, pp. 3928-3931, May. 2002.
- [19] R. A. Iltis, J. J. Shynk, and K. Giridhar, "Bayesian algorithms for blind equalization using parallel adaptive filtering," *IEEE Trans. Commun.*, vol. 42, no. 2/3/4, pp. 1017-1032, Feb/Mar/Apr. 1994.
- [20] K. Ozeki and T. Umeda, "An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties," *Electron. Commun. Jpn.*, vol. 67-A, pp. 19C27, 1984.
- [21] J. Benesty, P. Duhamel and Y. Grenier, "A multichannel affine projection algorithm with applications to multichannel acoustic echo cancellation," *IEEE Signal Process. Lett.*, vol. 3, no. 2, pp. 35-37, 1996.
- [22] M. Bouchard, "Multichannel affine and fast affine projection algorithms for active noise control and acoustic equalization systems," *IEEE Trans. Speech and Audio Process.*, vol. 11, no. 1, pp. 54-60, Jan. 2003.
- [23] L. Li, J. A. Chambers, C. G. Lopes and A. H. Sayed, "Distributed estimation over an adaptive incremental network based on the affine projection algorithm," *IEEE Trans. Signal Process.*, vol. 58, no. 1, pp. 151-164, Jan. 2010.
- [24] S. L. Gay and S. Tavathia, "The fast affine projection algorithm," in *Proc. ICASSP*, vol. 5, 1995.
- [25] M. Tanaka, S. Makino and J. Kojima, "A block exact fast affine projection algorithm," *IEEE Trans. Speech and Audio Process.*, vol. 7, no. 1, pp. 79-86, Jan. 1999.
- [26] S. G. Sankaran, and A. A. Beex, "Convergence behavior of affine projection algorithms," *IEEE Trans. Signal Process.*, vol. 48, no. 4, pp. 1086-1096, Apr. 2000.
- [27] H. C. Shin and A. H. Sayed, "Mean-square performance of a family of affine projection algorithms," *IEEE Trans. Signal Process.*, vol. 52, no. 1, pp. 90-102, Jan. 2004.