An Efficient Iterative Method for Basis Pursuit Adaptive Filters for Sparse Systems[1]

Steven L. Grant*, Pratik Shah* and Jacob Benesty†
*Missouri University of Science and Technology, Rolla, Missouri
E-mail: sgrant@mst.edu, pratik.shah@mail.mst.edu
+INRS-EMT, University of Quebec, Montreal, Canada
E-mail: benesty@emt.inrs.ca

Abstract— **The "proportionate" family of adaptive filters has been in use over the past decade. Their fast convergence for sparse systems makes them particularly useful in the network echo canceller application. Recently, an iterative form of the proportionate affine projection algorithm (PAPA), derived from the basic principles of basis pursuit, has been shown to have remarkably fast convergence for such sparse systems. The number of samples for convergence is proportional to the sparseness of the system which means that often full convergence occurs in fewer samples than the length of the system's impulse response. Here, we introduce a lower complexity implementation with the same performance that is an iterative version of proportionate normalized least mean squares (PNLMS).**

I. INTRODUCTION

Basis pursuit (BP) [1] has been shown to be an effective method of solving inverse problems with a small amount of data when the system to be determined has a sparse representation. This is especially attractive to adaptive filtering problems in that it promises the possibility of fast convergence when the solution the adaptive filter seeks is sparse. One such application is network echo cancellation where a number of approaches exploiting its sparse nature have been discussed extensively in the literature, in particular the proportionate normalized least-mean-square (PNLMS) and the proportionate affine projection algorithm (PAPA) [2] ,[3]. One feature of BP is that it requires the minimization of an L_1 norm with equality constraints, i.e.,

$$\min \|\mathbf{h}\|_1 \text{ subject to } \mathbf{d} = \mathbf{X}^T\mathbf{h} . \quad (1)$$

Typically this is accomplished using the simplex method [4]. Recently however, an iterative method based on PAPA [5] has been introduced that is more amenable to the adaptive filtering context.

This paper introduces a technique that offers a reduction in complexity over the iterative PAPA approach. Where PAPA requires a matrix inverse or the solution to a system of equations, in the approach taken here, that system of equations is solved by the method of row action projections (RAP) [6].

The signal model is described in section 2. Section 3 reviews the iterated PAPA method, section 4 introduces a more efficient solution using RAP, and section 5 shows the simulation results and a discussion based on these results. The paper is concluded in section 6.

II. SIGNAL MODEL

The observed or desired signal is given by

$$d(n) = \mathbf{x}^T(n)\mathbf{h} + v(n), \quad (2)$$

where n is the discrete time index

$$\mathbf{h} = \left[h_0 \ h_1 \ \ldots h_{L-1}\right]^T \quad (3)$$

is the L-tap impulse response of the system to be identified, the superscript ^{T} denotes the transpose of a vector or a matrix,

$$\mathbf{x}(n) = \left[x(n) \ x(n-1) \ \ldots \ x(n-L+1)\right]^T \quad (4)$$

is a vector containing the L most recent samples of the zero mean input at time n, and $v(n)$ is the zero mean additive white Gaussian noise which is independent of the input. In the affine projection algorithm (APA) and PAPA [3] it is typical to expand (2) to consider, say, M samples at a time where $1 \le M \le L$, thus,

$$\mathbf{d}(n) = \mathbf{X}^T(n)\mathbf{h} + \mathbf{v}(n) \quad (5)$$

where

$$\mathbf{X}(n) = \left[\mathbf{x}(n) \ \mathbf{x}(n-1) \ \ldots \mathbf{x}(n-M+1)\right], \quad (6)$$

$$\mathbf{d}(n) = \left[d(n) \ d(n-1) \ \ldots \ d(n-M+1)\right]^T, \quad (7)$$

and

$$\mathbf{v}(n) = \left[v(n) \ v(n-1) \ \ldots \ v(n-M+1)\right]^T. \quad (8)$$

The aim is to find an estimate of $\mathbf{h}$ with an adaptive filter

$$\hat{\mathbf{h}}(n) = \left[\hat{h}_0(n) \ \hat{h}_1(n) \ \ldots \hat{h}_{L-1}(n)\right]^T \quad (9)$$

such that the estimation error given by $\left\|\hat{\mathbf{h}}(n) - \mathbf{h}\right\|_2^2$ is bounded by an upper bound of ε, which is a small positive number and $\|\bullet\|_2^2$ is the square of the L_2-norm.

The BP approach to this problem (in the noise-free case) is thus,

[1] This work was partly performed under the Wilkens Missouri Endowment

$$\min\left\|\hat{\mathbf{h}}(n)\right\|_1 \text{ subject to } \mathbf{d}(n)=\mathbf{X}^T(n)\hat{\mathbf{h}}(n). \quad (10)$$

It has been shown that if $\hat{\mathbf{h}}(n)$ and $\mathbf{X}(n)$ meet certain conditions given in [1], BP provides the sparsest solution for $\hat{\mathbf{h}}(n)$. Typically, the simplex method is used to solve (10).

III. ITERATED PAPA

The steps of the PAPA algorithm are,

$$\mathbf{e}(n)=\mathbf{d}(n)-\mathbf{X}^T(n)\hat{\mathbf{h}}(n-1), \quad (11)$$

$$\mathbf{G}(n-1)=diag\left\{f\left(\left|\hat{\mathbf{h}}(n-1)\right|\right)\right\}, \quad (12)$$

and

$$\hat{\mathbf{h}}(n)=\hat{\mathbf{h}}(n-1) +\mathbf{G}(n-1)\mathbf{X}(n)\left[\mathbf{X}^T(n)\mathbf{G}(n-1)\mathbf{X}(n)\right]^{-1}\mathbf{e}(n). \quad (13)$$

Typically, the function $f\left(\left|\hat{\mathbf{h}}(n-1)\right|\right)$ in (12) is designed to prevent the diagonal elements of the scaling matrix $\mathbf{G}(n)$ from taking on zero values which would stall the adaptation of the elements of $\hat{\mathbf{h}}(n)$. In addition, various definitions of $f\left(\left|\hat{\mathbf{h}}(n-1)\right|\right)$ have appeared in the literature to optimize the performance of the proportionate algorithms for impulse responses of varying sparseness. Details are given in [7] and [8].

The link between PAPA, and basis pursuit was shown in [9]. There, the coefficient update derived from BP was shown to be

$$\hat{\mathbf{h}}(n)=\hat{\mathbf{h}}(n-1)+ +\mathbf{G}(n)\mathbf{X}(n)\left[\mathbf{X}^T(n)\mathbf{G}(n)\mathbf{X}(n)\right]^{-1}\mathbf{e}(n) \quad (14)$$

where the scaling matrix of sample period n was required for the update of the coefficients of the same sample period . To avoid this difficulty, the authors assumed that

$$\mathbf{G}(n)\approx\mathbf{G}(n-1), \quad (15)$$

arriving at the PAPA update of (13).

In [5] an iterative method was proposed to avoid the expedient of (15). The idea was to initialize $\hat{\mathbf{h}}^{[0]}$ to a small value and then apply PAPA repeatedly *on the same data*. This resulted in the following algorithm:

Algorithm One: Iterated PAPA

1) *Initialization*: $\hat{\mathbf{h}}^{[0]}=[\iota \ \cdots \ \iota]^T$, $i=0$
2) *Loop*: while $i< I$
   a. $i=i+1$
   b. $\mathbf{e}^{[i]}=\mathbf{d}(n)-\mathbf{X}^T(n)\hat{\mathbf{h}}^{[i-1]}$
   c. $\mathbf{G}^{[i-1]}=diag\left\{\left|\hat{\mathbf{h}}^{[i-1]}\right|\right\}$
   d. $\hat{\mathbf{h}}^{[i]}=\hat{\mathbf{h}}^{[i-1]} +\mu\mathbf{G}^{[i-1]}\mathbf{X}(n)\left[\mathbf{X}^T(n)\mathbf{G}^{[i-1]}\mathbf{X}(n)+\delta\mathbf{I}\right]^{-1}\mathbf{e}^{[i]}.$

The algorithm is initialized with $\hat{\mathbf{h}}^{[0]}=[\iota \ \cdots \ \iota]^T$, where $\iota\approx 10^{-20}$, with $\mu\approx 0.01$, and $\delta=\iota$. Using this method one obtains virtually the same result as when the simplex method is used to solve the optimization problem of problem (10). Again, the impressive aspect of this algorithm is that convergence is observed in only M sample periods.

An advantage of iterative PAPA over the simplex method is that the iterations applied to data from different sample periods and the associated computational complexity can be spread out over those samples. In addition, we may add "gear shifting" to the stepsize parameter, μ, to accelerate convergence and vary the iteration parameter, I, to lower the computational complexity once the algorithm has converged. These considerations suggested the following approach:

Algorithm Two: Iterated PAPA with Gear Shifting

1) *Initialization*: $\mu=0.5$, $I=I_{init}$, $\hat{\mathbf{h}}(n)=[\iota \ \cdots \ \iota]^T$
2) $\hat{\mathbf{h}}^{[0]}=\hat{\mathbf{h}}(n-1)$ and $i=0$
3) *Loop*: while $i< I$
   A. $i=i+1$
   B. $\mathbf{e}^{[i]}=\mathbf{d}(n)-\mathbf{X}^T(n)\hat{\mathbf{h}}^{[i-1]}$
   C. $\mathbf{G}^{[i-1]}=diag\left\{\left|\hat{\mathbf{h}}^{[i-1]}\right|\right\}$
   D. $\mathbf{S}=\mathbf{X}^T(n)\mathbf{G}^{[i-1]}\mathbf{X}(n)+\delta\mathbf{I}$
   E. $\boldsymbol{\varepsilon}=\mathbf{S}^{-1}\mathbf{e}^{[i]}$
   F. $\hat{\mathbf{h}}^{[i]}=\hat{\mathbf{h}}^{[i-1]}+\mu\mathbf{G}^{[i-1]}\mathbf{X}(n)\boldsymbol{\varepsilon}$
4) If $\left\|\mathbf{e}^{[I]}\right\|_2^2<\tau$, let $\mu=0.005$ and $I=1$, otherwise $\mu=0.5$ and $I=I_{init}$
5) *Update:* $\hat{\mathbf{h}}(n)=\hat{\mathbf{h}}^{[I]}$, $n=n+1$, Goto step 2.

Step (5) is the gear shifting step. Here, if $\left\|\mathbf{e}^{[I]}\right\|_2^2<\tau$, where τ is a threshold typically set in the neighborhood of 0.005, we lower the stepsize and decrease the number of iterations, I. Otherwise we use the initial stepsize (set for fast convergence) and the initial number of iterations I_{init} which is typically set in the neighborhood of 10 to 20.

The main computational complexity of the algorithm is in steps 3B through 3F when the algorithm has $I=I_{init}$. The break-down of the complexity per sample period is given in the Table 1. Here, we assign the complexity of KM^2 for the system-of-equations solution of step 3E. The constant K is a constant which is assumed to be approximately 3. By far, the

TABLE I
ITERATED PAPA COMPLEXITY

Step	Complexity per Sample Period
3B	IML
3C	IL
3D	$IML + IM^2L$
3E	IKM^2
3F	$IL + IML$
Total	$2IL + 3IML + IKM^2 + IM^2L$

most complex step is step 3D. If the matrix $\mathbf{G}^{[i-1]}$ were the identity matrix, as in the standard APA algorithm, this could be calculated with $2M^2$ operations using a sliding windowed approach. Here, though $\mathbf{G}^{[i-1]}$ prevents this simplification with the result that about IM^2L computations are required. With typical values of $I = 12$, $M = 60$, and $L = 512$, the number of computations is over 22 million operations per sample. Obviously, a technique that avoids this step would be highly desirable.

IV. ITERATED PAPA WITH ROW ACTION PROJECTIONS

One such approach is the method of row action projections (RAP) [6]. RAP is also known in the literature as a *data reuse* algorithm and has been extensively studied, see [10] and [11] for example. It was shown in [12] that RAP is effectively the same as APA, except that the system of equations problem that is solved with a matrix inversion in APA is solved iteratively in RAP. Here we will develop a version of RAP that corresponds to PAPA by taking into account the scaling matrix, $\mathbf{G}$.

Consider the system of equations in (5) when there is no noise,

$$\mathbf{d}(n) = \mathbf{X}^T(n)\mathbf{h} \tag{16}$$

An equivalent system is

$$\mathbf{d}(n) = \mathbf{X}^T(n)\mathbf{G}^{1/2}\mathbf{G}^{-1/2}\mathbf{h} \tag{17}$$

Or,

$$\mathbf{d}(n) = \mathbf{Y}^T(n)\mathbf{w} \tag{18}$$

where

$$\mathbf{Y}(n) = \mathbf{G}^{1/2}\mathbf{X}(n) \tag{19}$$

and

$$\mathbf{w} = \mathbf{G}^{-1/2}\mathbf{h} \tag{20}$$

The RAP procedure for solving (18) is

Algorithm Three: RAP

A. *Initialization*: $m = 0$
B. *Loop*: while $m < M$
   a) $\alpha(m) = \mu / \left(\mathbf{y}^T(n-m)\mathbf{y}(n-m) + \delta\right)$
   b) $m = m + 1$
C. *Initialization*: $m = 0$
D. *Loop*: while $m < JM - 1$
   a) for $j = m_{\bmod M}$
   b) $e^{[j]} = d(n-j) - \mathbf{y}^T(n-j)\mathbf{w}^{[m]}$
   c) $\hat{\mathbf{w}}^{[m+1]} = \hat{\mathbf{w}}^{[m]} + \alpha(j)\mathbf{y}(n-j)e^{[j]}$
   d) $m = m + 1$

Effectively, the algorithm cycles J times through the M rows of the system of equations and executes a relaxed, regularized NLMS update on the solution vector estimate, $\hat{\mathbf{w}}^{[m]}$ for each iteration.

Using (19) and (20) the algorithm becomes

Algorithm Four: Proportionate RAP

A. *Initialization*: $m = 0$
B. *Loop*: while $m < M$
   a) $\alpha(m) = \mu / \left(\mathbf{x}^T(n-m)\mathbf{G}\mathbf{x}(n-m) + \delta\right)$
   b) $m = m + 1$
C. *Initialization*: $m = 0$
D. *Loop*: while $m < JM - 1$
   a) $j = m_{\bmod M}$
   b) $e^{[j]} = d(n-j) - \mathbf{x}^T(n-j)\hat{\mathbf{h}}^{[m]}$
   c) $\hat{\mathbf{h}}^{[m+1]} = \hat{\mathbf{h}}^{[m]} + \alpha(j)\mathbf{G}\mathbf{x}(n-j)e^{[j]}$
   d) $m = m + 1$

Where in step Dc we have multiplied through by $\mathbf{G}^{1/2}$.

Using RAP in the Iterated PAPA algorithm of Algorithm 2 we have:

Algorithm Five: Iterated PAPA with RAP and Gear Shifting

1) *Initialization*: $\mu = 0.5$, $I = I_{init}$, $J = J_{init}$, $\hat{\mathbf{h}}(n) = [\iota \ \cdots \ \iota]^T$ where $\iota = 10^{-20}$
2) $\hat{\mathbf{h}}^{[0]} = \hat{\mathbf{h}}(n-1)$, $i = 0$
3) $e(n) = d(n) - \mathbf{x}^T(n)\hat{\mathbf{h}}(n-1)$
4) $\hat{\sigma}_e^2(n) = \lambda\hat{\sigma}_e^2(n-1) - (1-\lambda)e^2(n)$
5) *Loop*: while $i < I$
   A. $i = i + 1$
   B. $\mathbf{G}^{[i-1]} = diag\left\{\left|\hat{\mathbf{h}}^{[i-1]}\right|\right\}$
   C. *Initialization*: $m = 0$
   D. *Loop*: *While* $m \le M - 1$
      a) $\alpha(m) = \mu / \left(\mathbf{x}^T(n-m)\mathbf{G}^{[i-1]}\mathbf{x}(n-m) + \delta\right)$
      b) $m = m + 1$
   E. *Initialization*: $m = 0$

TABLE II
COMPLEXITY OF ITERATED PAPA WITH RAP

Step	Complexity per Sample Period
5B	IML
5Da	$2IML + MD$
5Fb	$IJML$
5Fc	$2IJML$
Total	$3IJML + 3IML + D$

F. *Loop*: *While* $m \le JM - 1$

a) $j = m_{\mathrm{mod}\,M}$

b) $e^{[j]} = d(n-j) - \mathbf{x}^T(n-j)\hat{\mathbf{h}}^{[m]}$

c) $\hat{\mathbf{h}}^{[m+1]} = \hat{\mathbf{h}}^{[m]} + \alpha(j)\mathbf{G}^{[i-1]}\mathbf{x}(n-j)e^{[j]}$

d) $m = m + 1$

G. *Update*: $\hat{\mathbf{h}}(n) = \hat{\mathbf{h}}^{[JM]}$

6) If $\hat{\sigma}_e^2(n) < \tau$, let $\mu = 0.005$, $I = 1$, and $J = 1$, else $\mu = 0.5$, $I = I_{init}$, and $J = J_{init}$

7) $n = n + 1$, goto step 2.

The complexity analysis of algorithm five is shown in table 2. The dominant term in the complexity is $3IJML$. For computational savings we require

$$J < M/3 \quad (21)$$

and the computational gain of Iterated PAPA with RAP compared to Iterated PAPA (using matrix inversion) is

$$G_{comp} \approx M/3J \,. \quad (22)$$

V. SIMULATIONS

In order to verify the performance of these algorithms, a simulation was performed based on the signal model described in section 2. The sample rate was 8000 samples per second. In the simulation, algorithm five, Iterated PAPA-RAP, is compared to PAPA. The echo path impulse response is comparable to a realistic network echo path. The signal to noise ratio (SNR) is set to 40 dB, L is set to 1000, and M is set to 200. For Iterated PAPA-RAP, the maximum iterations per sample, I_{init} is set to 75 and J_{init} is set to 3. Fig. 1 shows the coefficient error of the two algorithms. Iterated PAPA-RAP, converges considerably faster than PAPA.

The computational gain of Iterated PAPA-RAP over Iterated PAPA (using matrix inversion) during convergence is about 22 for the values used in this simulation.

VI. CONCLUSIONS

Previously, it has been shown that the network echo cancellation problem may be addressed using an iterative method based on PAPA to implement a version of BP. Here, we introduced a more efficient version of that algorithm using RAP, Iterated PAPA-RAP. Simulations show that the new algorithm converges very fast even when compared to PAPA, an algorithm especially fast at converging for sparse impulse responses such as the one used in the simulation. Moreover, Iterated PAPA-RAP is about 22 times as computationally efficient as Iterated PAPA using matrix inversion.

REFERENCES

[1] S. Chen, D. Donoho, and M. Saunders, “Atomic decomposition by basis pursuit,” *SIAM J. Sci. Comput.*, vol 20, no. 1, pp. 33—62, 1998.

[2] D. L. Duttweiler, “Proportionate normalized least-mean-squares adaptation in echo cancelers,” *IEEE Trans. Speech, Audio Process.*, vol. 8, pp. 508—518, Sept. 2000.

[3] J. Benesty, T. Gaensler, D. R. Morgan, M. M. Sondhi, and S. L. Gay, *Advances in Network and Acoustic Echo Cancellation*. Berlin: Springer-Verlag, 2001.

[4] G. Strang, *Linear Algebra and its Applications. Second edition*. New York: Academic Press, 1980.

[5] P. Shah, S. Grant, and J. Benesty, “On an iterative method for basis pursuit with application to echo cancellation with sparse impulse responses,” in *Proc. IEEE ICASSP,* 2012, pp. 177—180.

[6] S. Kaczmarz, “Angenaherte auflosung von systemen linearer gleichungen,” *Bull. Acad. Polon. Sci. Lett*. A. 35, 1937, pp. 355—357.

[7] S. Gay, “An efficient, fast converging adaptive filter for network echo cancellation,” *Proc. IEEE Signals, Systems & Computers*, 1998, pp. 394 – 398 vol. 1.

[8] J. Benesty, S. Grant, “An improved PNLMS algorithm,” in *Proc. IEEE ICASSP,* 2002, pp. II-1881 - II-1884.

[9] J. Benesty, C. Paleologu, and S. Ciochină, “Proportionate adaptive filters from a basis pursuit perspective,” *IEEE Signal Process. Lett.*, vol. 17, no. 12, pp. 985—988, Dec. 2010.

[10] J. Doherty and R. Mammone, “A fast method for regularized adaptive filtering,” *Digital Signal Processing Journal, Academic Press*, vol. 2, No. 1, pp. 14 – 26, Jan. 1992.

[11] J. Benesty, T. Gaensler, “On data-reuse adaptive algorithms,” *Proc. Intl. Workshop on Acoustic Echo and Noise Control (IWAENC2003)*, 2003, pp. 31 – 34.

[12] S. Gay, “Fast projection algorithms with application to voice echo cancellation.” Ph.D. dissertation, Rutgers, The State University of New Jersey, USA, 1994.

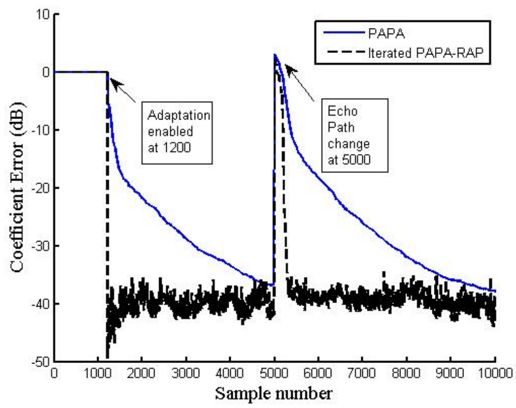

Fig. 1 A comparison of Iterated PAPA with RAP and PAPA alone. Here the echo path length is $L = 1000$, and the projection size is $M = 200$.