Intermittently Updated Simplified Proportionate Affine Projection Algorithm

Felix Albu, Henri Coanda, Dinu Coltuc, Marius Rotaru
Dept. of Electronics
Valahia University of Targoviste
Targoviste, Romania
E-mails: {felix.albu, coanda, coltuc}@valahia.ro; marius.rotaru@gmail.com

***Abstract*—In this paper, an intermittent update interval for filter coefficients and a simplified output error vector computation is proposed for a proportionate affine projection algorithm. It is shown that the proposed algorithm has good convergence performance and much smaller computation complexity than other proportionate-type APAs. Also, the accuracy of its implementation using the logarithmic number system was investigated. We demonstrated the performance of the proposed algorithm for echo cancellation and adaptive feedback cancellation applications.**

Keywords-Proportionate-type algorithms; adaptive filters; affine projection algorithm; logarithmic number system.

I. INTRODUCTION

There are many adaptive algorithms proposed for adaptive systems [1][2]. The most used algorithms are: the Normalized Least Mean Square (NLMS) algorithm, the Affine Projection Algorithm (APA) [3], and fast versions of APA for various applications like echo cancellation, hearing aids and active noise control (e.g., [4]–[9]). It is known that in echo cancellation systems, the echo paths are often sparse [1]. An intuitive idea for this case is to exploit the sparseness of the echo path by updating filter coefficients independently and proportionally to their estimated magnitude. One of the first such algorithms was proposed by Duttweiler [10], and it was called the Proportionate Normalized Least-Mean-Square (PNLMS) algorithm. Several proportionate algorithms were designed (e.g., [11], μ-law PAPA [12], Improved PAPA (IPAPA) [13], Memory IPAPA (MIPAPA) [14], μ-law MIPAPA (MMIPAPA) [15], and Approximated MIPAPA (AMIPAPA) [16]). The latter algorithm is still too complex, and an approximation for the output error computation of AMIPAPA was proposed in [17]. It was termed Simplified AMIPAPA (SAMIPAPA) and the complexity reduction come at a price of a reduced performance by several dB, especially when using speech signals and sparse echo paths. In [18], an algorithm that uses a combination of recursive filtering, dichotomous coordinate descent iterations and an approximation of a matrix in order to further reduce its numerical complexity in terms of multiplications was also proposed.

Therefore, a new proportionate algorithm with little performance degradation that incorporates an approximation of the output error and an intermittent update of filter coefficients depending on a computed threshold [19][20] is proposed in this paper. The algorithm proposed by Albu et al. in [20] used an intermittent update on an affine projection algorithm. It is shown that the threshold derived for the affine projection algorithm by Shin, Sayed & Song in [21] it is good enough for the proposed proportionate APA. The new algorithm is termed Intermittently Updated SAMIPAPA (IUSAMIPAPA). IUSAMIPAPA distinguishes from the algorithm proposed by Albu et al. in [20], called Intermittently Updated APA (IU-APA), because it is a proportionate-type algorithm and uses other steady-state MSE estimation formula. Also, the update formula of [20] is related linearly to the logarithm of the estimated variance of the filter output error. IUSAMIPAPA is different from the algorithm proposed by Albu in [18] because it does not include DCD iterations and uses other approximation. The algorithm proposed in Albu and Kwan [22] is a sign algorithm without an intermittent weights update unlike the proportionate algorithm presented in this paper.

The paper is organized as follows. Section 2 presents a short overview of the proportionate-type algorithms for echo cancellation. In Section 3, SAMIPAPA is derived and the proposed intermittently updated SAMIPAPA is investigated. In Section 4, the proposed algorithm is compared with AMIPAPA and SAMIPAPA in the context of echo cancellation and adaptive feedback cancellation. Also, the accuracy of its simulation using the logarithmic number system is verified. Finally, the conclusions are given in Section 5.

II. PROPORTIONATE-TYPE ALGORITHMS

In an echo cancellation system, we consider the far-end signal $x(n)$, and the reference signal $d(n)$, where n is the time index. The adaptive FIR filter is given by the coefficients vector $\hat{\mathbf{h}}(n)=\left[\hat{h}_0(n),\hat{h}_1(n),...,\hat{h}_{L-1}(n)\right]^T$, where L is the length of the adaptive filter and superscript T denotes transposition. The error signal is given by [1]

$$e(n)=d(n)-\hat{\mathbf{h}}^T(n-1)\mathbf{x}(n) \tag{1}$$

where $\mathbf{x}(n) = [x(n), x(n-1),\ldots, x(n-L+1)]^T$ is a vector containing the L most recent samples of the input signal. If p is the projection order, the error signal vector is given by

$$\mathbf{e}(n)=\mathbf{d}(n)-\mathbf{X}^T(n)\hat{\mathbf{h}}(n-1), \quad (2)$$

where $\mathbf{X}(n)=\left[\mathbf{x}(n),\mathbf{x}(n-1),\ldots,\mathbf{x}(n-p+1)\right]$ is the input signal matrix, $\mathbf{d}(n)=\left[d(n),d(n-1),\ldots,d(n-p+1)\right]^T$ is the reference signal vector, and $\mathbf{e}(n)=\left[e(n),e(n-1),\ldots,e(n-p+1)\right]^T$ is the error vector.

The coefficients of the proportionate-type affine projection algorithms (PAPA) are updated as follows [18]

$$\begin{aligned}\hat{\mathbf{h}}(n)&=\hat{\mathbf{h}}(n-1)+\mu\mathbf{G}(n-1)\mathbf{X}(n)\times\\&\left[\delta\mathbf{I}_p+\mathbf{X}^T(n)\mathbf{G}(n-1)\mathbf{X}(n)\right]^{-1}\mathbf{e}(n),\end{aligned} \quad (3)$$

where $\mathbf{G}(n-1)$ is an L x L diagonal matrix, δ is a regularization constant, μ is the normalized step-size parameter, and $\mathbf{I}_p$ is the p x p identity matrix. In the case of the improved PAPA (IPAPA) [13], the diagonal elements of $\mathbf{G}(n-1)$, denoted by $g_l(n-1)$, are evaluated as

$$g_l(n-1)=\frac{1-\alpha}{2L}+(1+\alpha)\frac{\left|\hat{h}_l(n-1)\right|}{2\sum_{i=0}^{L-1}\left|\hat{h}_i(n-1)\right|+\xi}, \quad (4)$$

where $-1\le\alpha<1$, $0\le l<L-1$ and ξ is a small positive constant. Let us denote [14]

$$\begin{aligned}\mathbf{P}(n)&=\mathbf{G}(n-1)\mathbf{X}(n)\\&=\left[\mathbf{g}(n-1)\odot\mathbf{x}(n)\ldots\mathbf{g}(n-1)\odot\mathbf{x}(n-p+1)\right],\end{aligned} \quad (5)$$

where $\mathbf{g}(n-1)$ is a vector containing the diagonal elements of $\mathbf{G}(n-1)$ and the operator $\odot$ denotes the Hadamard product [14]. $\mathbf{P}(n)$ is approximated with

$$\mathbf{P}'(n)=\left[\mathbf{g}(n-1)\odot\mathbf{x}(n)\ldots\mathbf{g}(n-p)\odot\mathbf{x}(n-p+1)\right], \quad (6)$$

where $\mathbf{g}(n-k)$ are the vectors containing the diagonal elements of the matrixes $\mathbf{G}(n-k)$, with $k = 1, 2, \ldots, p$ [14]. We have

$$\mathbf{P}'(n)=\left[\mathbf{g}(n-1)\odot\mathbf{x}(n) \quad \mathbf{P}'_{-1}(n-1)\right], \quad (7)$$

where the matrix

$$\begin{aligned}&\mathbf{P}'_{-1}(n-1)=\\&\left[\mathbf{g}(n-2)\odot\mathbf{x}(n-1)\ldots\mathbf{g}(n-p)\odot\mathbf{x}(n-p+1)\right],\end{aligned} \quad (8)$$

contains the first $p-1$ columns of $\mathbf{P}'(n-1)$. The MIPAPA equations are written as in [16]:

$$\mathbf{S}_1(n)=\delta\mathbf{I}_p+\mathbf{X}^T(n)\mathbf{P}'(n) \quad (9)$$

$$\hat{\mathbf{h}}(n)=\hat{\mathbf{h}}(n-1)+\mu\mathbf{P}'(n)\mathbf{S}_1^{-1}(n)\mathbf{e}(n) \quad (10)$$

The coefficients of the approximated MIPAPA (AMIPAPA) are given by [16]

$$\hat{\mathbf{h}}(n)=\hat{\mathbf{h}}(n-1)+\mu\mathbf{P}'(n)\mathbf{S}_2^{-1}(n)\mathbf{e}(n) \quad (11)$$

where, $\mathbf{S}_2(n)$, is updated by changing both its first row and column with $\mathbf{X}^T(n)\mathbf{P}'_{:,1}(n)$ and adding δ to the first element. $\mathbf{P}'_{:,1}(n)$ denotes the first column of $\mathbf{P}'(n)$ and is given by $\mathbf{g}(n-1)\odot\mathbf{x}(n)$. The bottom-right $(p-1)\times(p-1)$ submatrix of $\mathbf{S}_2(n)$ is replaced with the top-left $(p-1)\times(p-1)$ submatrix of $\mathbf{S}_2(n-1)$ [16].

III. INTERMITTENTLY UPDATED SIMPLIFIED AMIPAPA

Firstly, an important numerical complexity reduction is obtained if $\mathbf{e}(n)=\mathbf{d}(n)-\mathbf{X}^T(n)\hat{\mathbf{h}}(n-1)$ is approximated as in the original fast affine projection algorithm [4]

$$\mathbf{e}(n)=\left[e(n);(1-\mu)\overline{\mathbf{e}}^T(n-1)\right]^T, \quad (12)$$

where $\overline{\mathbf{e}}(n-1)$ represents the first $p-1$ elements of $\mathbf{e}(n-1)$. The algorithm proposed in [17] used (12) instead of (2) and was called simplified AMIPAPA (SAMIPAPA).

The numerical complexity of the following algorithms in terms of multiplications is presented in equations (13)-(15) ($P_m = O(p^3)$ [23] indicates the numerical complexity in terms of multiplications):

$$C_{\text{MIPAPA}}=L(4p+1)+p+P_m \quad (13)$$

$$C_{\text{AMIPAPA}}=L(3p+2)+p+P_m \quad (14)$$

$$C_{\text{SAMIPAPA}}=L(2p+3)+2p+P_m. \quad (15)$$

It can be noticed that the complexity of SAMIPAPA is roughly half of that of MIPAPA for typical echo cancellation systems where $L\gg p$. However, the complexity can be further reduced using the intermittently updated procedure proposed in [19]. Thus, the update equation of (11) can be replaced by

$$\hat{\mathbf{h}}(n)=\begin{cases}\hat{\mathbf{h}}(n-1)+\mu\mathbf{P}'(n)\mathbf{S}_2^{-1}(n)\mathbf{e}(n), & \text{if } n \bmod i_n=0\\ \hat{\mathbf{h}}(n-1) & \text{otherwise}\end{cases} \tag{16}$$

where i_n is the computed update interval at time n. Starting with an initial update interval of 1, i_n is given by

$$i_n=\begin{cases}\max\left[1, i_{n-1}-1\right], & \text{if } e^2(n)\geq\gamma\\ \min\left[i_{n-1}+1, i_M\right] & \text{otherwise}\end{cases} \tag{17}$$

where i_M is the maximum update interval and γ is the threshold [19] computed as in (18)

$$\gamma=\frac{\mu\sigma_v^2 p}{2-\mu}+\sigma_v^2, \tag{18}$$

where σ_v^2 is estimated during silences [24]. The numerical savings are important because (11) requires $Lp+P_m$ multiplications and the filter can have hundreds of coefficients in echo cancellation systems. The update of the filter coefficients from (16) is performed only when $n \bmod i_n=0$ and not at every iteration like in (11). The new algorithm is termed Intermittently Updated SAMIPAPA (IUSAMIPAPA). The algorithm can have a periodic update if the update interval is fixed to $i_n>1$.

IV. Simulation Results

Most of the simulations were performed in the context of echo cancellation, where the input signal is either white Gaussian noise or speech. The first impulse response from ITU-T G168 Recommendation [25] is padded with zeros in order to have 512 coefficients. A white Gaussian noise with a SNR = 30 dB is added at the output of the echo path. The performance measure used is the normalized misalignment (in dB), defined as $20\log_{10}(\|\mathbf{h}-\hat{\mathbf{h}}(n)\|_2/\|\mathbf{h}\|_2)$, where $\mathbf{h}$ denotes the true impulse response of the echo path. In the simulations with white noise, the performance curves are averaged over 10 independent trials. The regularization constant is δ = 0.01, p = 8 and α = 0. In all the simulations where the input signal is a white signal, the step size of all algorithms is 0.11.

Figure 1 shows the misalignment performance of the periodic SAMIPAPA with fixed periodically updated filter coefficients. It can be noticed that the larger the update interval, the lower steady-state error and the slower the convergence speed. Therefore, similar conclusions as those of [18] and [19] are obtained and this indicates that a variable updating interval for SAMIPAPA could lead to a good compromise between fast convergence and low steady-state error.

Figure 2 shows the misalignment curves for the proposed IUSAMIPAPA ($i_M=8$), SAMIPAPA, and the periodic SAMIPAPA with $i=8$. An abrupt change of the echo path after 25000 iterations by shifting the impulse response to the right by 12 samples was introduced in order to verify the tracking ability of the algorithms. It can be seen that IUSAMIPAPA has roughly the same initial convergence as SAMIPAPA and steady-state error of the periodic SAMIPAPA. The update of the filter weights is made on average only on a fifth of the number of iterations. Overall, for the investigated case, IUSAMIPAPA obtains an impressive 35% complexity reduction over SAMIPAPA in terms of multiplications (SAMIPAPA has 9884 multiplications, while IUSAMIPAPA has 6495 multiplications).

Figure 3 shows the misalignment curves for IUSAMIPAPA for different update intervals.

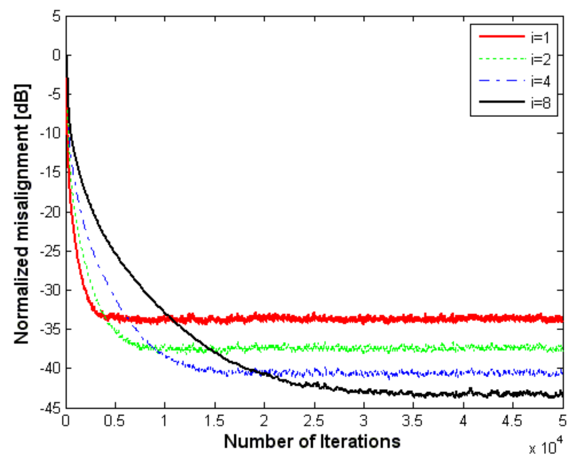

Figure 1. Misalignment of periodic SAMIPAPA for different update intervals, white noise, $p=8$, $L=512$, SNR = 30 dB.

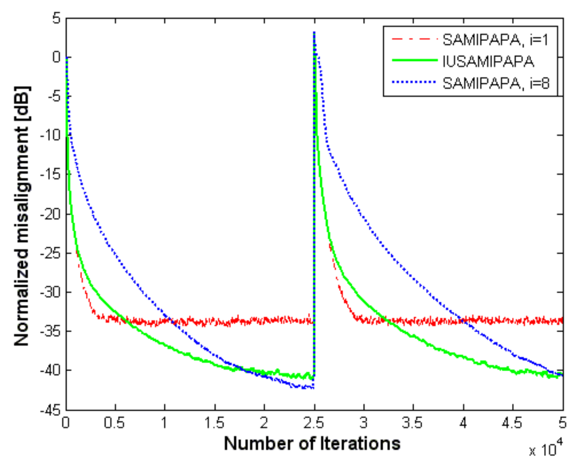

Figure 2. Misalignment of SAMIPAPA, periodic SAMIPAPA, i = 8, and IUSAMIPAPA $i_M=8$. Other conditions are the same as in Figure 1.

Similar conclusions with those obtained in [18] and [19] are obtained regarding the influence of i_M. It can be seen that the time to reach steady-state increases with i_M value.

For the considered case, the percentage of updates is about 15% for $i_M = 8$, 9% for $i_M = 16$, and 6% for $i_M = 32$. The overall number of updates is reduced by increasing i_M. The maximum update interval is set to the projection order in the following simulations. An example of computed i_n values and their histogram for the case $i_M = 8$ (Figure 3) is shown in Figure 4. It can be seen that during the initial convergence, the updating intervals are closer to 1, while they are closer to 8 in the steady-state region.

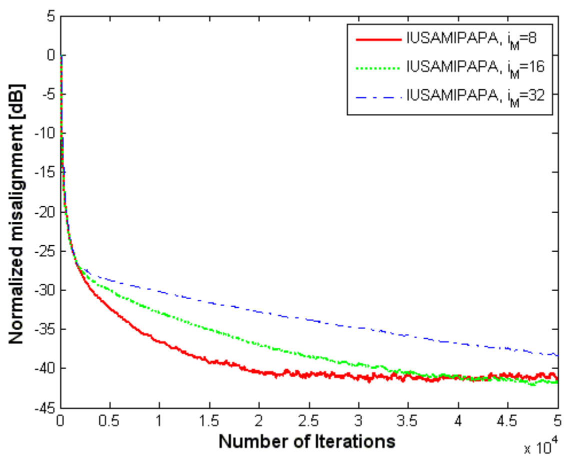

Figure 3. Misalignment of IUSAMIPAPA with $i_M = 8$, $i_M = 16$ and $i_M = 32$ respectively. Other conditions are the same as in Figure 1.

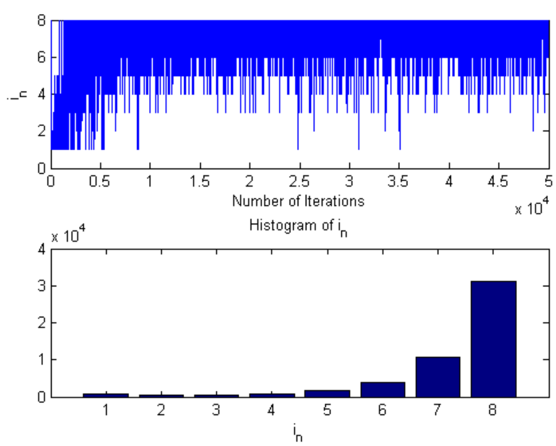

Figure 4. Computed update interval values (upper); and histogram of computed i_n values (lower)

In Figure 5, the input signal is speech, with p = 8, the output of the echo path is corrupted by independent white Gaussian noise SNR = 30 dB and the echo path changes after 0.5 seconds. The step-size for all algorithms is 0.2 for the following simulation. It was shown in [16] that MIPAPA has virtually identical performance with AMIPAPA at a higher computational cost. Therefore, for the following simulations, there is no need to plot the misalignment curves of MIPAPA. Also, the superiority of MIPAPA to APA for echo cancellation applications has been proved in previous publications [14]-[16]. Figure 5 shows that the approximation used by SAMIPAPA and the intermittent update of filter weights lead to slightly reduced performance (1 to 3 dB for this example) in comparison with AMIPAPA in case of a speech signal input. However, IUSAMIPAPA offers a better performance/complexity tradeoff than AMIPAPA, due to its reduced numerical complexity by about 42% (7766 multiplications vs. 13460 multiplications).

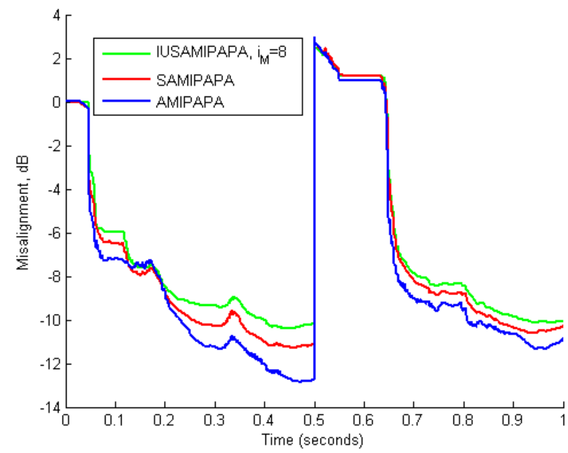

Figure 5. Misalignment of the AMIPAPA, SAMIPAPA and IUSAMIPAPA. Speech sequence, $p = 8$, $L = 512$, SNR = 30 dB, and echo path changes at time 0.5s.

The same conclusions can be drawn for results using colored noise as input signal, different filter lengths or maximum projection orders.

In the next simulation, the performance of MMIPAPA [15], AMIPAPA [16], SAMIPAPA [17] and IUSAMIPAPA is investigated in the acoustic feedback context [26]. The feedback path and the adaptive filter have 64 coefficients. A delay of 60 samples and a constant gain of 30 dB in the forward path were assumed. The sampling frequency was 16 kHz, $M = 8$, $\mu = 0.1$, and $\delta = 0.001$. The logarithmic factor of MMIPAPA [15] was 100. It can be seen from Figure 6, that most of the time, the performance of IUSAMIPAPA is superior to that of MMIPAPA, SAMIPAPA and AMIPAPA in case of a coloured input signal. IUSAMIPAPA obtains a smaller misalignment than the other algorithms, although has a slower convergence speed at some moments in time.

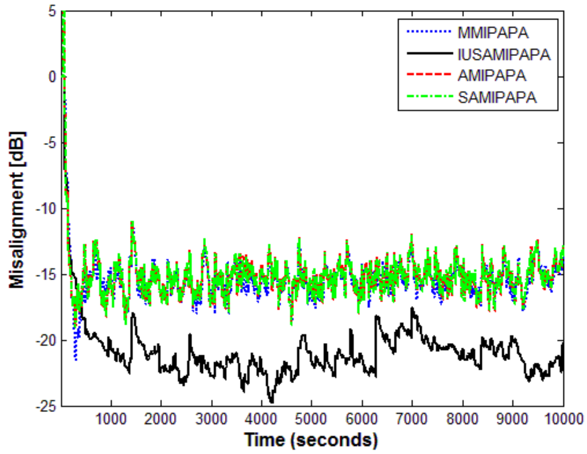

Figure 6. Misalignment of MMIPAPA, AMIPAPA, SAMIPAPA, and IUSAMIPAPA for an AFC application with coloured input signal, $M = 8$ and $\mu = 0.1$.

Figure 7 shows the same behaviour for a speech input signal. The parameters of the algorithms are the same as above example. It can be noticed that the performance of MMIPAPA, AMIPAPA and SAMIPAPA is most of the time similar. However, MMIPAPA has the highest numerical complexity from all the investigated algorithms. MMIP-APSA requires additional L logarithmic functions and L additions per iteration in comparison with MIPAPA.

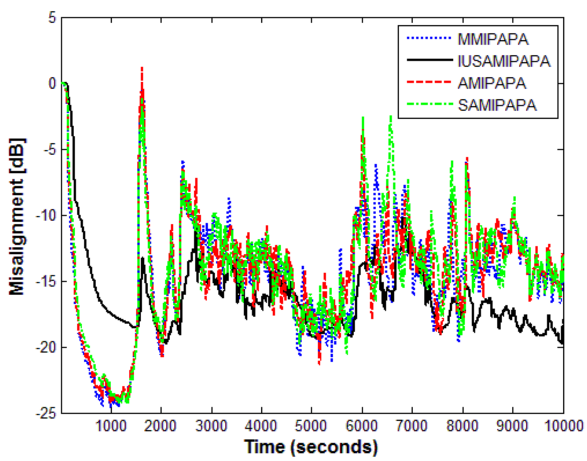

Figure 7. Misalignment of MMIPAPA, AMIPAPA, SAMIPAPA, and IUSAMIPAPA for an AFC application with speech input signal, $M = 8$ and $\mu = 0.1$.

We've also investigated the performance of the algorithm using 32-bit simulation using the logarithmic number system (LNS) and compared with 32-bit floating point results for the AFC example. The logarithmic number system is an alternative to floating-point that offers the potential to perform real multiplication, division and square-root at fixed-point speed and, in the case of multiply and divide, with no rounding error at all [27]. The logarithmic addition and subtraction are performed with the speed and accuracy equivalent to that of floating-point. The LNS format compares favorably against its floating-point counterpart, having greater range and slightly smaller representation error [27]. Impressive speed-ups were obtained over conventional floating point implementations for a wide range of algorithms [28][29]. More details about the logarithmic number system are available at http://www.ncl.ac.uk/eece/elm.

We considered the AFC experiment results for both 32-bit LNS and 32-bit floating point simulations. An accurate standard for comparison of the outputs was obtained by considering the corresponding double precision version results. The corresponding sum of absolute errors was computed for IUSAMIPAPA. The 32-bit LNS and 32-bit floating-point simulations have almost identical results. This confirmed similar conclusions obtained in the past for a wide range of algorithms. However, the sum of absolute errors of the 32 bit LNS implementation of IUSAMIPAPA was about 10% smaller than that of the 32-bit floating point implementation. Therefore, an LNS implementation could benefit from an increased accuracy.

V. CONCLUSION AND FUTURE WORK

In this paper, a low complexity proportionate-type AP algorithm was proposed. IUSAMIPAPA offers an excellent convergence performance/numerical complexity compromise in comparison with other proportionate AP algorithms. The performance was verified on an echo cancellation and adaptive feedback cancellation applications. Also, an accuracy investigation of an LNS implementation was performed. Future work will be focused on investigating the performance of the proposed algorithm on AFC application using two microphones in hearing devices [30] and compare it variable projection order versions [31].

ACKNOWLEDGMENT

This work was supported by a grant of the Romanian National Authority for Scientific Research, CNCS-UEFISCDI project number PN-II-ID-PCE-2011-3-0097.

REFERENCES

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

[2] E. Haensler and G. Schmidt, Eds., Topics in Acoustic Echo and Noise Control, Berlin, Germany: Springer-Verlag, 2006.

[3] 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, no. 5, May 1984, pp. 19–27.

[4] S. L. Gay and S. Tavathia, “The fast affine projection algorithm,” Proc. of IEEE ICASSP, May 1995, pp. 3023-3026, doi: 10.1109/ICASSP.1995.479482.

[5] A. Gonzales, F. Albu, M. Ferrer, and M. Diego, “Evolutionary and variable step size affine projection algorithms for active noise control ”, IET Signal Processing, vol. 7, (6), Aug. 2013, pp. 471-476, doi: 10.1049/iet-spr.2012.0213.

[6] S. Lee, I. Kim, and Y. Park, “Approximated affine projection algorithm for feedback cancellation in hearing aids,” Computer Methods and Programs in Biomedicine, Vol. 87, (3), Sept. 2007, pp. 254-261, doi: 10.1016/j.cmpb.2007.05.014

[7] A. Gonzalez, M. Ferrer, F. Albu, and M. de Diego, “Affine projection algorithms: evolution to smart and fast multichannel algorithms and applications,” Proc. of Eusipco 2012, Bucharest, Romania, Aug. 2012, pp. 1965-1969.

[8] F. Albu and A. Fagan, “The Gauss-Seidel pseudo affine projection algorithm and its application for echo cancellation,” Conference Record of the Thirty-Seventh Asilomar Conference on Signals, Systems and Computers, 2003, Vol. 2, 9-12 Nov. 2003, pp. 1303 – 1306, doi: 10.1109/ACSSC.2003.1292199

[9] F. Albu and H.K. Kwan, “Combined echo and noise cancellation based on Gauss-Seidel pseudo affine projection algorithm,” Proc. of IEEE ISCAS 2004, May 2004, pp. 505-508, doi:10.1109/ISCAS.2004.1328794.

[10] D. L. Duttweiler, “Proportionate normalized least-mean-squares adaptation in echo cancellers,” IEEE Transactions on Speech and Audio Processing, vol. 8, no. 5, Sept. 2000, pp. 508–518.

[11] J. Benesty and S. L. Gay, “An improved PNLMS algorithm,” in Proc. of IEEE ICASSP, 2002, vol. II, May 2002, pp. 1881–1884, doi: 10.1109/ICASSP.2002.5744994.

[12] H. Deng and M. Doroslovački, “Proportionate adaptive algorithms for network echo cancellation,” IEEE Transactions on Signal Processing, vol. 54, no. 5, May 2006, pp. 1794–1803, doi: 10.1109/TSP.2006.872533.

[13] O. Hoshuyama, R. A. Goubran, and A. Sugiyama, “A generalized proportionate variable step-size algorithm for fast changing acoustic environments,” Proc. of IEEE ICASSP, 2004, vol. IV, May 2004, pp. 161-164, doi: 10.1109/ICASSP.2004.1326788.

[14] C. Paleologu, S. Ciochina, and J. Benesty, “An efficient proportionate affine projection algorithm for echo cancellation,” IEEE Signal Processing Letters, vol. 17, no. 2, Feb. 2010, pp. 165–168, doi: 10.1109/LSP.2009.2035665.

[15] J. Yang and G.E. Sobelman, “Efficient μ -law improved proportionate affine projection algorithm for echo cancellation”, Electronics Letters, vol. 47, Issue 2, Jan. 2010, pp. 73 – 74, doi: 10.1049/el.2010.7937.

[16] F. Albu, C. Paleologu, J. Benesty, and S. Ciochina, “A low complexity proportionate affine projection algorithm for echo cancellation,” Proc. of EUSIPCO 2010, August 2010, pp. 6-10.

[17] F. Albu, “Simplified proportionate affine projection algorithm,” Proc. of IWSSIP 2012, April 2012, pp. 382-385.

[18] F. Albu, “New proportionate affine projection algorithm,” 41st International Congress and Exposition on Noise Control Engineering 2012, INTER-NOISE 2012, (9), Aug. 2012, pp. 7726 – 7733.

[19] K.-H. Kim, Y.-S. Choi, S.-E. Kim, and W. -J. Song, “An Affine Projection Algorithm with Periodically Evolved Update Interval,” IEEE Trans on Circuits and Systems-II, vol. 58, no.11, Nov. 2011, pp. 763-767, doi: 10.1109/TCSII.2011.2168023.

[20] F. Albu, M. Rotaru, R. Arablouei, and K. Dogancay,”Intermittently-updated affine projection algorithm,” Proc. of ICASSP 2013, May 2013, pp. 585 – 589, doi: 10.1109/ICASSP.2013.6637715.

[21] H.-C. Shin, A. H. Sayed and W.-J. Song, “Variable step-size NLMS and affine projection algorithms,” IEEE Signal Processing Leters, vol. 11, no. 2, Feb. 2004, pp. 132–135, doi: 10.1109/LSP.2003.821722.

[22] F. Albu and H.K. Kwan, “New proportionate affine projection sign algorithms”, in Proc. of ISCAS 2013, pp. 521 – 524, doi: 10.1109/ISCAS.2013.6571895.

[23] G. H. Golub and C. F. Van Loan, Matrix computation, 3rd edition. Baltimore, MD: The John Hopkins Univ. Press, 1996.

[24] J. Benesty, H. Rey, L. Rey Vega, and S. Tressens, “A nonparametric VSS NLMS algorithm,” IEEE Signal Processing Letters, vol. 13, no. 10, Oct. 2006, pp. 581–584, doi: 10.1109/LSP.2006.876323.

[25] Digital Network Echo Cancellers, ITU-T Rec. G.168, 2002.

[26] M. Rotaru, C. Stanciu, S. Ciochina, F. Albu, and H. Coanda, “A FPGA Implementation of Prediction Error Method for Active Feedback Cancellation using Xilinx System Generator,” Proc. of ADAPTIVE 2013, May 2013, pp. 26-29.

[27] J.N.Coleman, E.Chester, C.Softley and J.Kadlec "Arithmetic on the European Logarithmic Microprocessor," IEEE Trans. Comput. Special Edition on Computer Arithmetic, Vol. 49, No. 7, Jul. 2000, pp. 702-715, doi: 10.1109/12.863040.

[28] F. Albu, J. Kadlec, N. Coleman, and A. Fagan, “Pipelined Implementations of the A Priori Error-Feedback LSL Algorithm Using Logarithmic Arithmetic”, Proceedings of ICASSP 2002, May 2002, pp. 2681-2684, doi: 10.1109/ICASSP.2002.5745200.

[29] F. Albu, J. Kadlec, C. Softley, and R. Matousek,, A. Hermanek, A. Fagan, N. Coleman, "Implementation of (Normalized) RLS Lattice on VIRTEX", Field Programmable Logic and Applications, Gordon Brebner and Roger Woods Editors, 2001, Aug. 2001, pp. 91-100, doi: 10.1007/3-540-44687-7_10.

[30] C. R. C. Nakagawa, S. Nordholm, F. Albu, W.-Y. Yan, "Closed-loop feedback cancellation utilizing two microphones and transform domain processing", Proceedings of ICASSP 2014, pp. 3673-3677.

[31] F. Albu, C. Paleologu, and J. Benesty, “A Variable Step Size Evolutionary Affine Projection Algorithm,” Proc. of ICASSP 2011, May 2011, pp. 429-432, doi: 10.1109/ICASSP.2011.5946432.

The matlab code for the proposed algorithm can be obtained by filling the request form from
http://falbu.50webs.com/List_of_publications_aec.htm

The reference for the paper is:

F. Albu, H. Coanda, D. Coltuc, & M. Rotaru, “Intermittently Updated Simplified Proportionate Affine Projection Algorithm”, in *Proc. of ADAPTIVE 2014*, Venice, Italy, pp. 42-47, ISBN: 978-1-61208-341-4