On the Regularization of the Memory-Improved Proportionate Affine Projection Algorithm

Roxana Mihăescu, Cristian Stanciu, and Constantin Paleologu(✉)

University Politehnica of Bucharest, Bucharest, Romania
{cristian,pale}@comm.pub.ro

Abstract. In order to improve the performance of the conventional algorithms used for network and acoustic echo cancellation, we can exploit the sparseness character of the echo paths (i.e., a small percentage of the impulse response components have a significant magnitude while the rest are zero or small). In this paper, we consider the memory-improved proportionate affine projection algorithm (MIPAPA), which represents an appealing choice for echo cancellation. In this context, we focus on the regularization of this algorithm, relating the regularization parameter to the signal-to-noise ratio. In this way, the algorithm can operate properly in different noisy conditions. Simulation results indicate the good performance of the proposed solution.

Keywords: Adaptive filters · Echo cancellation
Memory-improved affine projection algorithm (MIPAPA)
Regularization

1 Introduction

The main goal in echo cancellation is to model an unknown system, i.e., the echo path [1,2], similar to a system identification problem. Nevertheless, the echo paths (for both network and acoustic echo cancellation scenarios) are sparse in nature, i.e., a small percentage of the impulse response components have a significant magnitude while the rest are zero or small. The sparseness character of the echo paths inspired the idea to "proportionate" the algorithm behavior, i.e., to update each coefficient of the filter independently of the others, by adjusting the adaptation step size in proportion to the magnitude of the estimated filter coefficient [3]. Following this idea, many proportionate-type algorithms were developed for echo cancellation, e.g., see [4] and the references therein.

In this work, we focus on the memory-improved proportionate affine projection algorithm (MIPAPA) [5]. As compared to most of its counterparts, the MIPAPA takes into account the "history" of the proportionate factors. Moreover, this also helps to reduce the computational complexity.

O. Fratu et al. (Eds.): FABULOUS 2017, LNICST 241, pp. 151–157, 2018.
https://doi.org/10.1007/978-3-319-92213-3_22

The MIPAPA requires a matrix inversion within its update. Due to the nature of the input signal (which is mainly speech), this matrix can be very ill-conditioned. Consequently, it needs to be regularized before inversion by adding a positive constant to the elements of its main diagonal. In practice, it was found that the value of this regularization term is highly influenced by the level of the system noise [6].

In this paper, we present a solution for choosing the constant regularization parameter of the MIPAPA, aiming to attenuate the effects of the noise in the adaptive filter estimate. Simulations performed in the context of both network and acoustic echo cancellation indicate the robustness of the algorithm in different noisy conditions.

2 Memory-Improved Proportionate Affine Projection Algorithm

The improved proportionate affine projection algorithm (IPAPA) [7] is one of the most popular algorithms used for echo cancellation. It results as a straightforward combination of the affine projection algorithm (APA) [8] and the improved proportionate normalized least-mean-square (IPNLMS) algorithm [9]. The IPAPA is defined by the following equations:

$$\mathbf{e}(n) = \mathbf{d}(n) - \mathbf{X}^T(n)\widehat{\mathbf{h}}(n-1), \tag{1}$$

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \alpha\mathbf{G}(n-1)\mathbf{X}(n)\left[\delta\mathbf{I}_P + \mathbf{X}^T(n)\mathbf{G}(n-1)\mathbf{X}(n)\right]^{-1}\mathbf{e}(n), \tag{2}$$

where $\mathbf{e}(n)$ is the error signal vector of length P (with P denoting the projection order), $\mathbf{d}(n) = \left[\, d(n)\ d(n-1) \cdots d(n-P+1)\,\right]^T$ is a vector containing the most recent P samples of the desired signal, superscript ^{T} denotes the transpose operator, $\mathbf{X}(n) = \left[\,\mathbf{x}(n)\ \mathbf{x}(n-1) \cdots \mathbf{x}(n-P+1)\,\right]$ is the input data matrix, where $\mathbf{x}(n) = \left[\, x(n)\ x(n-1) \cdots x(n-L+1)\,\right]^T$ is a vector containing the most recent L samples of the input signal $x(n)$,

$$\mathbf{G}(n-1) = \mathrm{diag}\left[\, g_0(n-1)\ g_1(n-1) \cdots g_{L-1}(n-1)\,\right] \tag{3}$$

is a diagonal matrix containing the proportionate (or gain) factors, α represents the step-size of the algorithm, δ is the regularization parameter, and $\mathbf{I}_P$ is the $P \times P$ identity matrix. The proportionate factors are evaluated as [9]

$$g_l(n-1) = \frac{1-\kappa}{2L} + (1+\kappa)\frac{\left|\widehat{h}_l(n-1)\right|}{2\sum_{l=0}^{L-1}\left|\widehat{h}_l(n-1)\right|},\ 0 \le l \le L-1, \tag{4}$$

where κ $(-1 \le \kappa < 1)$ is a parameter that controls the amount of proportionality. Looking of the equations that define the IPAPA, i.e., (1) and (2), it can be noticed that the classical APA [8] is obtained for $\mathbf{G}(n-1) = \mathbf{I}_L$ (where $\mathbf{I}_L$ is the $L \times L$ identity matrix), while the IPNLMS algorithm [9] results when $P = 1$.

In practice, it would be very computationally expensive (and also inefficient) to compute the matrix product $\mathbf{G}(n-1)\mathbf{X}(n)$ in the classical way (i.e., matrices multiplication). Hence, taking into account the diagonal character of the matrix $\mathbf{G}(n-1)$, we can evaluate

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

where $\mathbf{g}(n-1) = \left[\,g_0(n-1)\ g_1(n-1)\cdots g_{L-1}(n-1)\,\right]^T$ is a vector containing the diagonal elements of $\mathbf{G}(n-1)$ and the operator $\odot$ denotes the Hadamard product. Using (5), the IPAPA update (2) can be rewritten as

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \alpha\mathbf{P}(n)\left[\delta\mathbf{I}_P + \mathbf{X}^T(n)\mathbf{P}(n)\right]^{-1}\mathbf{e}(n). \tag{6}$$

However, the IPAPA does not take into account the "proportionate history" of each coefficient $\widehat{h}_l(n-1)$, with $l = 0, 1, \ldots, L-1$, but only its proportionate factor from the current time sample, i.e., $g_l(n-1)$. Therefore, let us consider a modified approach in order to take advantage of the "proportionate memory" of the algorithm, by choosing the matrix [5]

$$\mathbf{G}_l(n-1) = \mathrm{diag}\left[\,g_l(n-1)\ g_l(n-2)\cdots g_l(n-P)\,\right]. \tag{7}$$

In this manner, we take into account the "proportionate history" of the coefficient $\widehat{h}_l(n-1)$, in terms of its proportionate factors from the last P time samples. Thus, the matrix from (5) becomes

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

and consequently, the update (6) is

$$\widehat{\mathbf{h}}(n) = \widehat{\mathbf{h}}(n-1) + \alpha\mathbf{P}'(n)\left[\delta\mathbf{I}_P + \mathbf{X}^T(n)\mathbf{P}'(n)\right]^{-1}\mathbf{e}(n). \tag{9}$$

We refer to this algorithm as the "memory" IPAPA (MIPAPA) [5].

The advantage of this modification is twofold. First, the MIPAPA takes into account the "history" of the proportionate factors from the last P steps. Second, the computational complexity is lower as compared to the IPAPA. This is (8) can be recursively evaluated as

$$\mathbf{P}'(n) = \left[\,\mathbf{g}(n-1)\odot\mathbf{x}(n)\ \mathbf{P}'_{-1}(n-1)\,\right], \tag{10}$$

where the matrix $\mathbf{P}'_{-1}(n-1)$ contains the first $P-1$ columns of $\mathbf{P}'(n-1)$. Thus, the columns from 1 to $P-1$ of the matrix $\mathbf{P}'(n-1)$ can be used directly for computing the matrix $\mathbf{P}'(n)$ [i.e., they become the columns from 2 to P of $\mathbf{P}'(n)$].

Besides, let us examine the matrix to be inverted in the classical IPAPA, as compared to the case of the MIPAPA. In the first case, this matrix is $\mathbf{M}(n) = \delta\mathbf{I}_P + \mathbf{X}^T(n)\mathbf{P}(n)$, which is symmetric but does not have a time-shift character.

On the other hand, the matrix to be inverted in the MIPAPA is not symmetric, but has a time-shift property, which allows us to evaluate

$$\mathbf{M}'(n) = \begin{bmatrix} \delta + \mathbf{x}^T(n)\left[\mathbf{g}(n-1) \odot \mathbf{x}(n)\right] & \mathbf{x}^T(n)\,\mathbf{P}'_{-1}(n-1) \\ \mathbf{X}_{-1}^T(n-1)\left[\mathbf{g}(n-1) \odot \mathbf{x}(n)\right] & \mathbf{M}'_{P-1}(n-1) \end{bmatrix}, \tag{11}$$

where the matrix $\mathbf{M}'_{P-1}(n-1)$ contains the first $P-1$ columns and $P-1$ rows of the matrix $\mathbf{M}'(n-1)$ [i.e., the top-left $(P-1) \times (P-1)$ submatrix of $\mathbf{M}'(n-1)$] and the matrix $\mathbf{X}_{-1}(n-1)$ contains the first $P-1$ columns of the matrix $\mathbf{X}(n-1)$. Consequently, only the first row and the first column of $\mathbf{M}'(n)$ need to be computed. Moreover, using computationally efficient techniques to perform to the matrix inversion operation [10], the overall complexity could be further reduced.

3 Regularization Parameter

Regularization plays a fundamental role in adaptive filtering. An adaptive filter that is not properly regularized will perform very poorly. As shown in Sect. 2, a matrix inversion is required within the MIPAPA. For practical reasons, the matrix needs to be regularized before inversion, i.e., a positive constant is added to the elements of its main diagonal. Usually, this regularization is chosen as $\delta = \beta\sigma_x^2$, where $\sigma_x^2 = E\left[x^2(n)\right]$ is the variance of the zero-mean input signal $x(n)$, with $E[\cdot]$ denoting mathematical expectation, and β is a positive constant (usually referred as the normalized regularization parameter). In practice though, β is more a variable that depends on the level of the additive noise, i.e., the more the noise, the larger is the value of β.

In the case of MIPAPA, we propose to choose the constant regularization parameter based on a condition that intuitively makes sense, i.e., to attenuate the effects of the noise in the adaptive filter estimate. This idea was introduced and explained in detail in [6], in case of the NLMS-based algorithms, including IPNLMS. Moreover, as it was shown in [11,12], the regularization of APA and IPAPA does not depend on the projection order. Thus, following the idea from [6], the regularization parameter of MIPAPA is evaluated as

$$\delta = \frac{1 + \sqrt{1 + \mathrm{SNR}}}{\mathrm{SNR}}\sigma_x^2 = \beta_{\mathrm{SNR}}\sigma_x^2, \tag{12}$$

where $\mathrm{SNR} = \sigma_y^2/\sigma_v^2$, with $\sigma_y^2 = E\left[y^2(n)\right]$ and $\sigma_v^2 = E\left[v^2(n)\right]$ representing the variances of the echo signal $y(n)$ and the near-end background noise $v(n)$, respectively. In (12), the parameter β_{SNR} is the normalized regularization parameter that depends on the SNR (which could be estimated in practice).

4 Simulation Results

Simulations were performed in the context of both network and acoustic echo cancellation. Two echo paths were used, having different sparseness degree, as follows: (i) the first impulse response from G168 Recommendation [13], which can be considered to be very sparse and (ii) a measured acoustic echo path, which is less sparse. Both impulse responses have 512 coefficients, using a sampling rate of 8 kHz. The adaptive filter used in the experiments has the same length ($L = 512$).

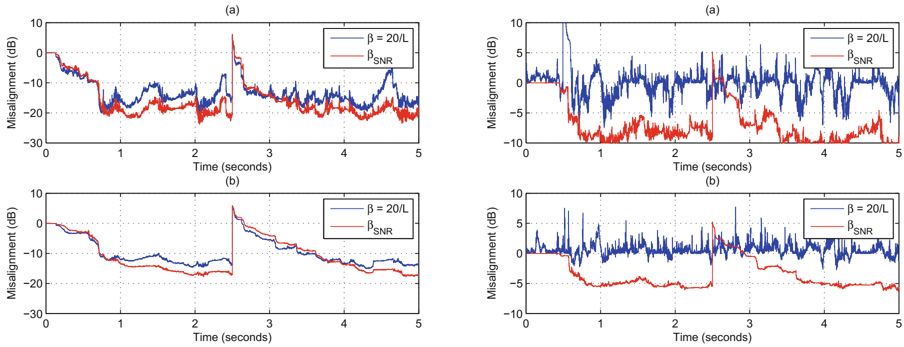

Fig. 1. Misalignment of the MIPAPA using different values of the normalized regularization parameter. (a)–*left*: G168 echo path, SNR = 20 dB; (b)–*left*: acoustic echo path, SNR = 20 dB; (a)–*right*: G168 echo path, SNR = 0 dB; (b)–*right*: acoustic echo path, SNR = 0 dB.

The input signal (i.e., the far-end signal) is a speech sequence. The output of the echo path is corrupted by an independent white Gaussian noise (i.e., the background noise at the near-end) with different SNRs, i.e., 20 dB and 0 dB. In order to evaluate the tracking capabilities of the algorithm, an echo path change scenario is simulated in the experiments, by shifting the impulse response to the right by 12 samples. The performance measure is the normalized misalignment (in dB), which is defined as $20\log_{10}\left[\left\|\mathbf{h} - \widehat{\mathbf{h}}(n)\right\| / \|\mathbf{h}\|\right]$, where $\mathbf{h}$ is the impulse response of the echo path and $\|\cdot\|$ denotes the ℓ_2 norm.

The performance of MIPAPA are compared for two types of regularization. The first one is the "classical" ad-hoc choice $\beta = 20/L$, which was the rule of thumb in many practical scenarios that involved the proportionate-type algorithms [1,2]. The second one is the proposed β_{SNR}. The step-size parameter is set to $\alpha = 0.2$, the projection order is $P = 4$, and the proportionality parameter is chosen as $\kappa = 0$.

The results are presented in Fig. 1. As we can notice from the left side of this figure (where the SNR is set to 20 dB), the performance obtained with the "classical" regularization $\beta = 20/L \approx 0.03$ are very similar to those obtained using β_{SNR}. This is expected, because if we consider $L = 512$ and SNR = 20 dB, we get $\beta_{\mathrm{SNR}} \approx 0.11$, which is quite close to the "classical" value.

The importance of the regularization parameter becomes more apparent in noisy environments. As we can notice from the right side of this figure (where

SNR = 0 dB), the MIPAPA using β_{SNR} outperforms by far the "classical" regularization. In this case, a much higher value of the normalized regularization constant is required; according to (12), for SNR = 0 dB and $L = 512$, we obtain $\beta_{\text{SNR}} \approx 2.41$, which is much higher as compared to the "classical" choice $\beta = 20/L$. Clearly, when using an improper regularization, the misalignment of the adaptive filter fluctuates much and never converges.

5 Conclusions

Adaptive filters with a large number of coefficients are usually involved in echo cancellation. In this context, the MIPAPA represents an appealing choice, since it inherits the good convergence features of the APA and also exploits the sparseness character of the echo paths (specific to the proportionate-type algorithms). However, regularization is an important component of any adaptive filter. It is as much important as the step-size parameter that controls the stability and convergence of the algorithm. In this paper, we have presented a solution to choose the regularization parameter of the MIPAPA as a function of the SNR. The goal was to attenuate the effects of the system noise in the adaptive filter estimate. Simulations performed in the context of both network and acoustic echo cancellation prove the validity of this approach in different noisy environments.

References

1. Benesty, J., Gaensler, T., Morgan, D.R., Sondhi, M.M., Gay, S.L.: Advances in Network and Acoustic Echo Cancellation. Springer, Heidelberg (2001). https://doi.org/10.1007/978-3-662-04437-7
2. Benesty, J., Huang, Y. (eds.): Adaptive Signal Processing-Applications to Real-World Problems. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-662-11028-7
3. Duttweiler, D.L.: Proportionate normalized least-mean-squares adaptation in echo cancelers. IEEE Trans. Speech Audio Process. **8**, 508–518 (2000)
4. Paleologu, C., Benesty, J., Ciochină, S.: Sparse Adaptive Filters for Echo Cancellation. Morgan & Claypool Publishers, USA (2010)
5. Paleologu, C., Ciochină, S., Benesty, J.: An efficient proportionate affine projection algorithm for echo cancellation. IEEE Signal Process. Lett. **17**, 165–168 (2010)
6. Benesty, J., Paleologu, C., Ciochină, S.: On regularization in adaptive filtering. IEEE Trans. Audio Speech Lang. Process. **19**, 1734–1742 (2011)
7. Hoshuyama, O., Goubran, R.A., Sugiyama, A.: A generalized proportionate variable step-size algorithm for fast changing acoustic environments. In: IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pp. IV-161–IV-164 (2004)
8. Ozeki, K., Umeda, T.: An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties. Electron. Commun. Jpn. **67**(A), 19–27 (1984)
9. Benesty, J., Gay, S. L.: An improved PNLMS algorithm. In: IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pp. II-1881–II-1884 (2002)

10. Stanciu, C., Anghel, C., Paleologu, C., Benesty, J., Albu, F., Ciochină, S.: A proportionate affine projection algorithm using dichotomous coordinate descent iterations. In: IEEE International Symposium on Signals, Circuits and Systems (ISSCS), pp. 343–346 (2011)
11. Paleologu, C., Benesty, J., Ciochină, S.: Regularization of the affine projection algorithm. IEEE Trans. Circ. Syst. II Express **58**, 366–370 (2011)
12. Paleologu, C., Benesty, J., Albu, F.: Regularization of the improved proportionate affine projection algorithm. In: IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pp. 169–172 (2012)
13. Digital Network Echo Cancellers. ITU-T Rec. G.168 (2002)