

An Affine Combination of Adaptive Filters for Channels with Different Sparsity Levels

Maksim Butsenko and Tõnu Trump

Abstract—In this paper we present an affine combination strategy for two adaptive filters. One filter is designed to handle sparse impulse responses and the other one performs better if impulse response is dispersive. Filter outputs are combined using an adaptive mixing parameter and the resulting output shows a better performance than each of the combining filters separately. We also demonstrate that affine combination results in faster convergence than a convex combination of two adaptive filters.

Keywords — Adaptive filters, combination filters, sparse impulse response.

I. INTRODUCTION

ADAPTIVE filtering is a well-known technique in signal processing [1]. A number of well-established algorithms have been introduced and recent advances in this area mostly focus on improving existing algorithms by introducing sophisticated ways of how to deal with circumstances of a particular application [14], [15]. There are two ways to describe how well an adaptive filter performs: steady-state error - how well filter's linear estimation of some impulse response describes the actual system; rate of convergence - how fast the adaptive filter reaches the steady-state. If we have some prior knowledge about the unknown impulse response, we can fine-tune conventional algorithms to perform better in terms of convergence rate [2], [4]. Using system identification task as an example, if we know that the system we attempt to identify is sparse, we can use this fact to design faster converging algorithms [14], [15]. We can consider a system as sparse if the majority of its impulse response coefficients are zero or near-zero.

One example of an application which is likely to exhibit sparse impulse responses is the network echo cancellation. Most part of the public switched telephone network (PSTN) uses four wires to transmit the speech signals to the subscribers. In the local exchange the four wire system is replaced with an analogue two wire system and only two wires connect the subscriber to the rest of the telephony system. The echo occurs due to the impedance mismatch in the hybrid circuits that connect the four wire part with the

two wire part of the system. Because of the impedance mismatch (in the hybrid circuit) a part of the signal energy reflects back and is perceived as echo. An echo with a delay of more than 20 ms can be noticeable and starts to disturb the user [6]. Hence, we would like to cancel that echo out. Adaptive filters are commonly used for this application.

The echo canceller works using an adaptive filter as a means of estimating an echo path that is present in the system [1]. Filtering the input signal $x(k)$ produces an echo estimate $\hat{y}(k)$. The filter itself is described with a vector of filter coefficients $\hat{\mathbf{h}}(k)$ which is the linear estimate of the echo path impulse response. Echo estimate is then subtracted from the signal $d(k)$ which consists of near-end speech $v(k)$ that is corrupted by the echo $y(k)$. In the end of this operation system output $e(k)$ produces an estimate of the near-end speech signal which is less corrupted by echo.

The normalized least mean squares algorithm (NLMS) is a simple but efficient algorithm that is widely used in adaptive filtering. Here the iterative update of filter coefficients (3) is used to compute the estimate of an echo-path. If N is a length of the adaptive filter and μ is a step-size parameter, then the NLMS algorithm is given by following equations [1]:

$$\hat{y}(k) = \hat{\mathbf{h}}(k)\mathbf{x}(k) \quad (1)$$

$$e(k) = d(k) - \hat{y}(k) \quad (2)$$

$$\hat{\mathbf{h}}(k+1) = \hat{\mathbf{h}}(k) + \mu \frac{e(k)\mathbf{x}(k)}{\mathbf{x}^T(k)\mathbf{x}(k)} \quad (3)$$

For stability the step-size must be selected within the interval $0 < \mu < 2$ [1].

In practice, the impulse response of an echo path has a sparse structure in many cases. As the network delay is unknown, the synthesis window used in echo cancellers must be much larger than the echo impulse response itself. Hence, we gain the combined impulse response which most of the time is zero and is thus sparse. ITU-T Rec. G.168 [5] defines several echo path models that are examples of measurements from telephone networks in North America and Europe. These examples typically specify only the dispersive part of the echo path. The actual echo paths usually consist of the delay and the dispersive part and are thus sparse. However, the NLMS algorithm is not able to exploit the sparseness of the system. Therefore, more suitable algorithms have been proposed. Rest of the paper is organized as follows. Section 2 gives a brief overview of PNLMS algorithm. In Section 3 we present the combination

Paper received May 2, 2016; accepted May 15, 2016. Date of publication July 20, 2016. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Branimir Reljin.

This paper is a revised and expanded version of the paper presented at the 23rd Telecommunications Forum TELFOR 2015 [16].

Maksim Butsenko and Tõnu Trump are with the Department of Radio and Communication Engineering, Tallinn University of Technology, Ehitajate tee 5, Tallinn, Estonia (e-mail: maksim.butsenko@ttu.ee, tonu.trump@ttu.ee).

of filters and describe our algorithm. Section 4 presents simulations results for the proposed method.

II. PNLMS

PNLMS stands for proportionate normalized least mean squares algorithm and it was first proposed in [2] and [3]. The basic idea behind the PNLMS is that the algorithm updates each coefficient according to its magnitude, so large coefficients are emphasized. They get larger update coefficients and the algorithm therefore converges faster than NLMS. By update coefficients we mean a factor by which the ordinary increment which updates the filter coefficients is multiplied. If we are dealing with a sparse characteristic then this approach helps the whole algorithm to converge faster than the conventional NLMS. Equations (4-9) describe the PNLMS update mechanism and comparing this to NLMS we can see a new term \mathbf{G} appearing in equation (9). This term is the update coefficient matrix and it is responsible for the proportionate update of the adaptive filter coefficients. \mathbf{G} is a diagonal matrix and its elements are defined as the ratio of an individual $g_n(k)$ and the average of all $g_n(k)$ -s. δ and ρ are regularization parameters which help to avoid the algorithm from stalling in case of a small coefficient and when all $\hat{h}_n(k)$ are zero accordingly.

$$l_\infty(k) = \max\{|\hat{h}_0(k)|, \dots, |\hat{h}_{N-1}(k)|\} \quad (4)$$

$$l'_\infty(k) = \max\{\delta, l_\infty(k)\} \quad (5)$$

$$g_n(k) = \max\{\rho l'_\infty(k), |\hat{h}_n(k)|\} \quad (6)$$

$$\bar{g}(k) = \frac{1}{N} \sum_{n=0}^{N-1} g_n(k) \quad (7)$$

$$\mathbf{G}(k) = \text{diag}\left\{\frac{g_0(k)}{\bar{g}(k)}, \dots, \frac{g_{N-1}(k)}{\bar{g}(k)}\right\} \quad (8)$$

$$\hat{\mathbf{h}}(k+1) = \hat{\mathbf{h}}(k) + \mu \mathbf{G} \frac{e(k)\mathbf{x}(k)}{\mathbf{x}^T(k)\mathbf{x}(k)} \quad (9)$$

The stability limit of a PNLMS algorithm is dependent on the gain distribution within the main diagonal of a matrix \mathbf{G} . According to [2] the stability limit on μ is never lower than 2/3. This occurs when the whole update energy is assigned to only one coefficient. The maximum stability is achieved when the update energy is divided equally between all the update coefficients. In this case the algorithm remains stable if the step size is less than two and the algorithm behaves similarly to the NLMS.

The PNLMS algorithm defines a family of proportionate algorithms. The algorithms are all based on the idea of preferring some coefficients over the others and giving them more weight, which helps to improve algorithm efficiency [2], [3]. The PNLMS algorithm is much better suited for dealing with sparse impulse responses than ordinary NLMS. The PNLMS algorithm has a faster initial convergence but tends to slow down in time and might become slower than the NLMS. Also, if the impulse response is dispersive the PNLMS might perform worse

than the NLMS. As the nature of an echo path is unknown beforehand it is desirable to have an algorithm which can cope well with both sparse and dispersive impulse responses. For this reason we study in this paper a combination of filters, which can provide such an alternative.

III. COMBINATION OF FILTERS

The combination of two (or more) adaptive filters has been studied mostly from the perspective of combining two filters, where one would grant faster convergence and the other a better steady-state performance [7], [8], [11]. Traditionally the outputs of the two adaptive filters are combined according to

$$\hat{y}(k) = \lambda(k)\hat{y}_1(k) + [1 - \lambda(k)]\hat{y}_2(k). \quad (10)$$

The question here is how to choose the λ parameter. A number of authors have opted for the convex combination as in [8], [9], some suggest using an affine combination as in [7], [10].

Combined filters have been shown to provide improved performance for achieving both faster convergence and good steady-state behaviour. They are also well-suited to handle the situation of the channel with a variable impulse response [8], [9]. In this case two filters are combined where one performs better when the impulse response is sparse and the second filter performs better when the impulse response is dispersive thus improving the robustness of the performance of combined filters for channels with different levels of sparsity. Different authors have approached the study of such systems differently. For example, Garcia et al. [8], Das and Chakraborty [9] have used a convex combination of filters. Bershada et al. [7] describe an optimal affine combiner but are focusing their work on a traditional approach for improving both convergence rate and steady-state error. Gui et al. [10] are also using an affine combination but only for sparse channel identification and are not considering the case with the dispersive impulse response. In this paper we are proposing an affine combination algorithm for channels with different sparsity levels. Results presented in this paper show that our proposed affine combination algorithm outperforms a convex combination algorithm for this particular application.

The algorithm in [8], [9] describes filter output signals that are combined using a convex combination and $\lambda(n)$ is controlled by the sigmoid activation function given by the following equation:

$$\lambda(k) = \text{sgm}[a(k)] = \frac{1}{1 + e^{-a(k)}} \quad (11)$$

where $a(k)$ is defined iteratively as

$$a(k+1) = a(k) - \mu_e e(k)[e_1(k) - e_2(k)]\lambda(k)[1 - \lambda(k)]. \quad (12)$$

The μ_a is a step-size for iteratively adapting the parameter a .

In the case of a convex combination the mixing parameter $\lambda(k)$ is restricted to the range (0,1). However, it has been shown [7] that a sequence of values of $\lambda(k)$ which

minimizes the mean-square deviation of an algorithm does not necessarily lie in the range (0,1) for all k -s. This leads us to the idea of adopting the affine combination even in the problem at hand.

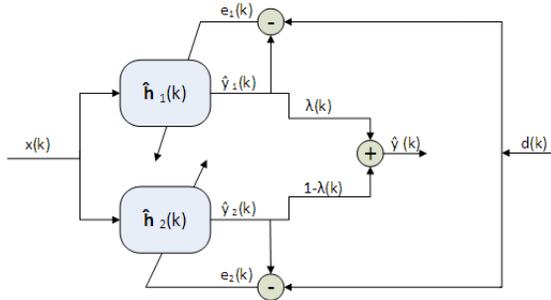


Fig. 1. The combined adaptive filter.

In this paper we thus propose using the affine combination of combinational filters in the context of channels with different levels of sparsity. Let us consider two adaptive filters, as shown in Fig. 1. The first filter $\hat{\mathbf{h}}_1(k)$ is a PNLMS filter for the sparse impulse response and the second filter $\hat{\mathbf{h}}_2(k)$ is a NLMS filter for the dispersive impulse response.

The outputs of the two adaptive filters are combined according to equation (10) where $\hat{y}_i(k) = \hat{\mathbf{h}}_i^T(k-1)\mathbf{x}(k)$ and λ can be any real number. The output combination is thus affine as in [7], not convex as in [8].

We define the *a priori* system error signal as a difference between the output signals of the true system at the time k , given by $y(k) = \mathbf{h}^T \mathbf{x}(k) = d(k) - v(k)$, and the output signal of our adaptive scheme $\hat{y}(k)$

$$e_a(k) = y(k) - \lambda(k)\hat{y}_1(k) - (1 - \lambda(k))\hat{y}_2(k). \quad (13)$$

$\lambda(k)$ can be found by minimizing the mean square of the *a priori* system error. The derivative of $E[e_a^2(k)]$ with respect to $\lambda(k)$ will look as follows

$$\begin{aligned} \frac{\partial E[e_a^2(k)]}{\partial \lambda(k)} &= 2E \left[(y(k) - \lambda(k)\hat{y}_1(k) \right. \\ &\quad \left. - (1 - \lambda(k))\hat{y}_2(k)) (-\hat{y}_1(k) \right. \\ &\quad \left. + \hat{y}_2(k)) \right] \\ &= 2E \left[(y(k) - \hat{y}(k))(\hat{y}_2(k) \right. \\ &\quad \left. - \hat{y}_1(k)) \right. \\ &\quad \left. + \lambda(k)(\hat{y}_2(k) - \hat{y}_1(k))^2 \right]. \end{aligned} \quad (14)$$

Setting the derivative to zero results in

$$\lambda(k) = \frac{E[(d(k) - \hat{y}_2(k))(\hat{y}_1(k) - \hat{y}_2(k))]}{E[(\hat{y}_1(k) - \hat{y}_2(k))^2]}, \quad (15)$$

where the true system output signal $y(k)$ is replaced by its observable noisy version $d(k)$. However, because we can make the standard assumption that the input signal $x(k)$ and measurement noise $v(k)$ are independent random processes, this can be achieved without introducing any error into our calculations.

In our simulations we have slightly altered the equation for finding λ in order to obtain a more practical algorithm.

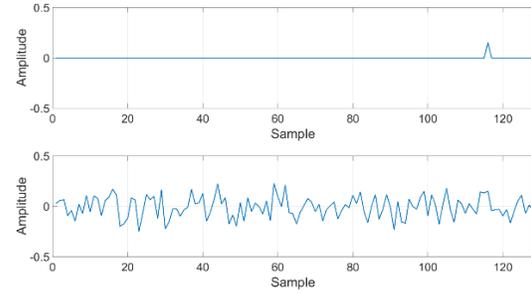


Fig. 2. Sparse impulse response (top) and dispersive impulse response (bottom) describing echo paths for synthetic test.

Namely we have replaced the mathematical expectations in the numerator and the denominator of equation (15) by exponential averaging of the type

$$P_u(k) = (1 - \gamma)P_u(k-1) + \gamma u^2(k) \quad (16)$$

where $u(k)$ is the signal to be averaged, $P_u(k)$ is the averaged quantity, and γ is the averaging parameter which has to take values between zero and one.

IV. SIMULATION AND RESULTS

For testing purposes we have used the following setup. The adaptive filter input is a white Gaussian noise with zero mean. The input signal is filtered through a finite impulse response filter which models the echo path to get the echo signal $y(k)$. The measurement noise $v(k)$ is also a white Gaussian noise, statistically independent of $x(k)$, with zero mean and the resulting SNR of the reference signal $d(k)$ is 20 dB. The algorithm comparison is done by plotting the learning curve based on the normalized weight misalignment which is defined as

$$\eta(k) = \frac{\|\mathbf{h}(k) - \hat{\mathbf{h}}(k)\|_2}{\|\mathbf{h}(k)\|_2}. \quad (17)$$

The filters that are used for combination have the same step size $\mu_{NLMS} = \mu_{PNLMS} = 0.1$.

The first test is a synthetic one. We have used two impulse responses with length $N = 128$ shown in Fig. 2. One impulse response is sparse with only one active coefficient. The second impulse response is dispersive with all coefficients being active and their value pulled randomly from a set of real numbers with zero-mean and $\sigma = 0.1$. In the test we have used an echo path change scenario where initial convergence starts with one echo path model and after algorithms reach their steady-state, the echo path model is changed and adaptation starts over again. We start with the dispersive impulse response and change it to the sparse one. We compare four algorithms - NLMS, PNLMS and two combination filters: an affine combination proposed in this paper and a convex combination filter described in [8]. The results of the test are shown in Fig. 3. We can see that both affine and convex combination perform well and are able to follow the best filter. However, after echo path change, the filter based on the affine combination shows faster initial convergences than the filter based on the convex combination. It is also showing faster convergence than its components (NLMS and PNLMS filters). Fig. 4 shows the

mixing parameter λ for both convex and affine combination corresponding to the learning curves in Fig. 3. We can clearly see that value of λ for the affine combination is above 1 after the echo path change. This is the part where the affine combination outperforms the convex combination. Also, it is worth noting how the PNLMs filter struggles with the dispersive impulse response (the first part of the plot).

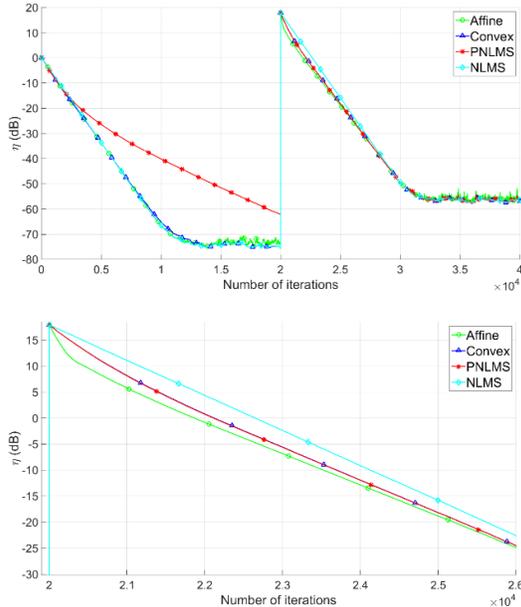


Fig. 3. Normalized misalignment of PNLMs, NLMS, affine and convex combined filters (top). Zoomed in part of the echo path change (bottom).

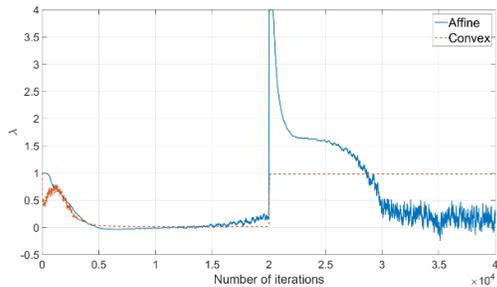


Fig. 4. Mixing parameter λ of convex and affine combination filters.

In the other part of our simulations we wanted to test more realistic scenarios. We used a similar setup with the echo path change but used more realistic impulse responses. Two scenarios were tested. In the first impulse response of an echo path changes from dispersive to sparse. In the second scenario a sparse impulse response is used during the entire run but during echo path change it is shifted in time. The impulse responses that were used in our experiments are shown in Fig. 5. All impulse responses are of length $N=512$. The sparse impulse response is taken from ITU-T Rec. G.168 [5]. The dispersive impulse response is generated using synthetic impulses with random sequences as described in [12]. The sparseness of an impulse response can be evaluated by the following equation which is called the sparseness measure [12], [13]

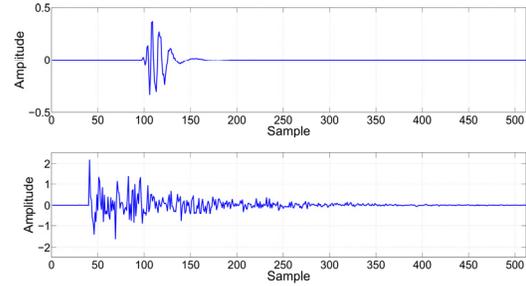


Fig. 5. Sparse (top) and dispersive (bottom) impulse responses.

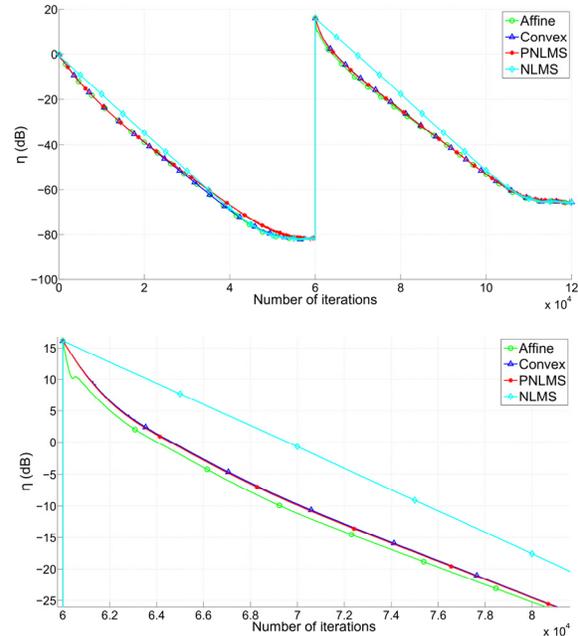


Fig. 6. Normalized misalignment of PNLMs, NLMS, affine and convex combined filters (top). Zoomed in part of the echo path change (bottom).

$$\xi(k) = \frac{N}{N - \sqrt{N}} \left[1 - \frac{\|\mathbf{h}(k)\|_1}{\sqrt{N}\|\mathbf{h}(k)\|_2} \right] \quad (18)$$

Sparseness measure is a bounded function $0 \leq \xi \leq 1$. The closer ξ approaches to 1, the sparser the impulse response and the closer ξ gets to 0, the more dispersive the impulse response is [12]. Sparseness measures used in our simulations are 0.8031 for the sparse impulse response and 0.5390 for the dispersive impulse response. As we can see from Fig. 6, the affine combination algorithm proposed in this paper performs as expected and shows the best performance. At the start, when the impulse response is dispersive, it follows the output of a PNLMs filter as far as it adapts faster than the NLMS (due to preferring the largest coefficients). Then affine and convex combined filters are both switching to follow the NLMS filter. After echo path has changed and impulse response becomes sparse, combination filter follows the output of a PNLMs filter as it tends to show a better performance. The affine combination shows faster convergence than the convex combination with a performance gain up to 2 dB. As the

output of the combination filter is dependent upon mixing parameter λ , then in Fig. 7 we provide a comparison of λ -s produced by convex and affine combination algorithms. It is clear that both algorithms choose which initial filter to follow in a similar way. They start from following PNLMS filter, then slowly drift towards NLMS filter and after echo path change go back to PNLMS filter. The benefit of using the affine combination shows faster convergence for all impulse responses from G.168 Recommendation [5]. Moreover, output for the affine combination filter gives a faster converging curve than any of the filters used in the combinational filter.

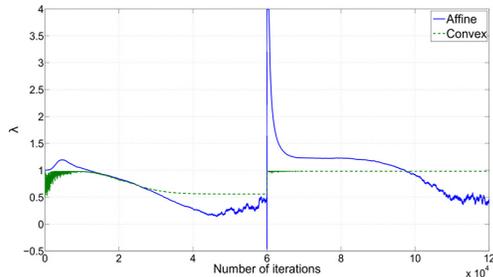


Fig. 7. Mixing parameter λ of convex and affine combination filters.

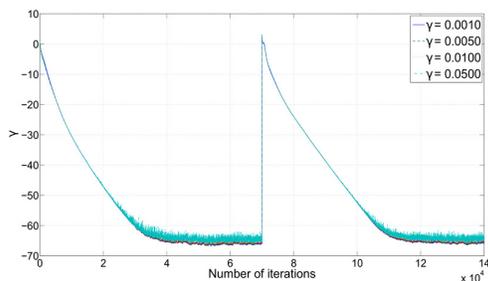


Fig. 8. Learning curves of the affine combination filter with different γ -s.

The second experiment compares the same algorithms but the test setup is changed to the second scenario described above (echo path change from a sparse impulse response to its shifted version). In Fig. 9 we can see that the result is the same as in the previous example. After echo path change the affine combination algorithm shows faster convergence than NLMS, PNLMS and a convex combination. Results were consisted for all impulse responses defined in ITU-T Rec. G.168 [5].

Lastly, we show how parameter γ affects the performance of proposed algorithm. In Fig. 8 we can see that larger gamma values have a negative impact on the performance of the algorithm. $\gamma = 0.05$ produced the curve with a higher steady-state error level, which also exhibits a large variation in steady state. When $0.01 \geq \gamma \geq 0.001$ then the algorithm performs well and there are no significant changes in behaviour of the algorithm. We suggest using $\gamma = 0.01$ for the best results as choosing γ considerably smaller than that slows down the tracking capabilities of the algorithm.

V. CONCLUSION

In this paper we have investigated a combination of two adaptive filters, one using NLMS and the other one PNLMS adaptation. We combine the outputs of two filters with the affine combination of filter outputs where the mixing parameter λ is steered by exponential averaging of input signal power. We have shown that the convergence of the affine combined filter is faster than the convergence of the convex combination of those filters. Directions for future research could involve a transient mean-square analysis for the proposed algorithm and investigation of possible applications of the compressive sensing schemes for the sparse impulse response identification task.

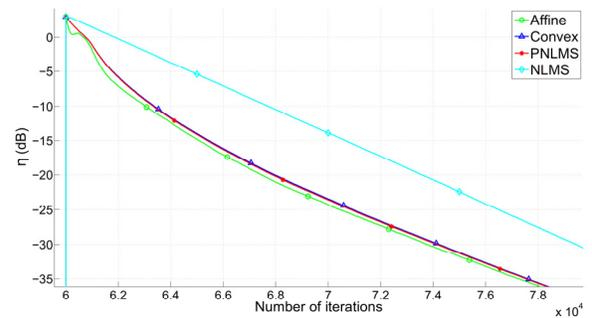
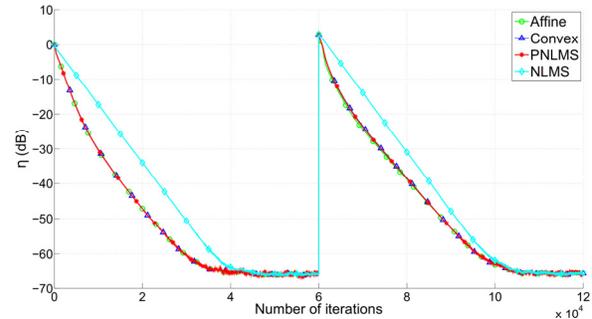


Fig. 9. Normalized misalignment of PNLMS, NLMS, affine and convex combined filters (top). Zoomed in part of the echo path change (bottom).

REFERENCES

- [1] S. Haykin, *Adaptive Filter Theory*, Fourth Edition, Prentice Hall, 2002.
- [2] D. L. Duttweiler, "Proportionate normalized least-mean-squares adaptation in echo cancellers," *IEEE Transactions on Speech and Audio Processing*, vol. 8, pp. 508-515, September 2000.
- [3] S. L. Gay, "An efficient, fast converging adaptive filter for network echo cancellation," *Conference Record of the Thirty-Second Asilomar Conference on Signals, Systems and Computers*, vol. 1, pp. 394-398, November 1998.
- [4] J. Benesty and S. L. Gay, "An improved PNLMS algorithm," *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 2, May 2002.
- [5] ITU-T, *Digital Network Echo Cancellers*, Rec. G.168, 2012.
- [6] S. V. Vaseghi, *Advanced Digital Signal Processing and Noise Reduction*, Fourth Edition, John Wiley and Sons, 2008.
- [7] N. J. Bershad, J. C. M. Bermudez, J.-Y. Tourneret, "An affine combination of two LMS adaptive filters — transient mean-square analysis," *IEEE Transactions on Signal Processing*, vol. 56, no. 5, pp. 1853-1864, May 2008.

- [8] J. Arenas-Garcia, A. R. Figueiras-Vidal, "Adaptive combination of IPNLMS filters for robust sparse echo cancellation," *IEEE Workshop on Machine Learning for Signal Processing*, pp. 221-226, October 2008.
- [9] B. K. Das, and M. Chakraborty, "Sparse adaptive filtering by an adaptive convex combination of the LMS and the ZA-LMS algorithms," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, issue. 5, pp. 1499-1507, January 2014.
- [10] G. Gui, S. Kumagai, A. Mehdodniya, F. Adachi, "Two are better than one: adaptive sparse system identification using affine combination of two sparse adaptive filters," *2014 IEEE 79th Vehicular Technology Conference*, pp. 1-5, May 2014.
- [11] T. Trump, "A combination of two NLMS filters in an adaptive line enhancer," *2011 17th International Conference on Digital Signal Processing (DSP)*, pp. 1-6, July 2011.
- [12] A. W. H. Khong, P. A. Naylor, "Efficient use of sparse adaptive filters," *Fortieth Asilomar Conference on Signals, Systems and Computers*, pp. 1375-1379, October-November 2006.
- [13] P. O. Hoyer, "Non-negative matrix factorization with sparseness constraints," *Journal of Machine Learning Research*, vol. 5, pp. 1457-1469, November 2004.
- [14] Y. Chen, Y. Gu, A.O. Hero, "Sparse LMS for system identification," *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 3125-3128, April 2009.
- [15] Y. Gu, J. Jin, S. Mei, "l0 norm constraint LMS algorithm for sparse system identification," *IEEE Signal Processing Letters*, vol.16, no. 9, pp. 774-777, September 2009.
- [16] M. Butsenko and T. Trump, "An affine combination of adaptive filters for sparse impulse response identification," *Telecommunications Forum Telfor (TELFOR), 2015 23rd*, Belgrade, 2015, pp. 396-399.