Gigabit DSL: a Deep-LMS Approach

Avi Zanko*
Faculty of Engineering
Bar-Ilan University
Ramat-Gan, Israel

Itsik Bergel
Faculty of Engineering
Bar-Ilan University
Ramat-Gan, Israel

Amir Leshem
Faculty of Engineering
Bar-Ilan University
Ramat-Gan, Israel

***Abstract*— In this paper we present the Deep-LMS, a novel algorithm for crosstalk cancellation in DSL. The Deep-LMS crosstalk canceler uses an adaptive non-diagonal preprocessing matrix prior to a conventional LMS crosstalk canceler. The role of the preprocessing matrix is to speed-up the convergence of the conventional LMS crosstalk canceler and hence speed-up the convergence of the overall system. The update of the preprocessing matrix is inspired by deep neural networks. However, since all the operations in the Deep- LMS algorithm are linear, we are capable of providing an exact convergence speed analysis. The Deep-LMS is important for crosstalk cancellation in the novel G.fast standard, where traditional LMS converges very slowly due to the large bandwidth. Simulation results support our analysis and show significant reduction in convergence time compared to existing LMS variants.**

***Index Terms*—Crosstalk canceler, DSL, LMS, G.fast.**

I. Introduction

Digital Subscriber Line (DSL) is a family of technologies that provide digital data transmission over unshielded twisted pairs (UTP) of copper wires that were originally used for telephone services. The performance of a DSL system may be significantly degraded by the effect of crosstalk due to electromagnetic coupling between adjacent pairs. We distinguish between two types of crosstalk: Near end crosstalk (NEXT) and Far end crosstalk (FEXT). In DSL systems, NEXT is avoided by using frequency division duplexing (FDD) [1], or time division duplexing (TDD) [2]. FEXT suppression is performed by spectrum shaping, advanced precoding or crosstalk cancellation (also known as vectoring). A detailed surveys of these techniques can be found in e.g., [3–5].

Gigabit over DSL was presented in a pioneering as early as 2003 paper [6]. Recently, G.fast standard was established [2] aiming to provide aggregate data rates of up to 1Gb/sec over short lines of up to 250m length. To that end, the spectrum bandwidth was increased from 30MHz to 106MHz in G.fast [2] or 212MHz in a more advanced version. However, the increased bandwidth raised new challenges: At high frequencies the channel matrix becomes non-diagonal dominant [7]. For vectored VDSL modems, the channel is weakly row-wise diagonally dominated for the downstream. Hence, approximate matrix inversion is used instead of the zero forcing (ZF) linear precoding [7]. Alternatively adaptive least mean square (LMS) based techniques for computing the precoder have been suggested [8–10]. Therefore, traditional DSL algorithms for crosstalk cancellation/precoding become inefficient or converge very slowly.

This paper focuses on designing a per tone adaptive crosstalk canceler for the upstream transmissions in G.fast systems. Facing the tremendous amount of data to be processed in each second[1], low complexity algorithms are necessary. In the literature, several adaptive algorithms exist that minimize the mean square error (MSE) at the output of the crosstalk canceler. Certainly, the most popular algorithm is the LMS algorithm [11]. LMS is a stochastic gradient descent method that under certain conditions converges to the Minimum MSE (MMSE) solution. The popularity of the LMS algorithm is mostly due to its simplicity. However, it is well known that a bad conditioning of the input correlation matrix may lead to slow convergence of the algorithm.

Being a stochastic gradient descent algorithm, the rate of convergence of the LMS algorithm is controlled by the step-size parameter μ. While small step size leads to a better precision of the steady state solution, large step size is preferable for short transient state (fast convergence). However, if the step size is too large the system becomes unstable, and the algorithm may not converge at all. Thus, even for optimal choice of the step size, the convergence rate strongly depends on the statistics of the input signal [12–14].

Several LMS derivatives exhibit a faster convergence: In the normalized LMS (NLMS)[15], a step-size that is normalized with the power of the input is used (for low power inputs a certain regulation parameter is typically used). In the diagonal step-size matrix approach, a diagonal matrix is used instead of the scalar step size μ [16]. Another family of algorithms uses time variant step-size parameter such that $\mu = \mu(n)$ is made inversely proportional to the iteration number n [17] The Leaky LMS algorithm that is developed similarly to the conventional LMS but using a slightly modifying cost function can be recast as a conventional LMS with slightly better eigenvalues spread [18].

Another framework of solutions to accelerate the LMS utilized a variety of averaging methods of the LMS coefficients (see for example [19]). In the accelerated LMS, the LMS component are updated off-line as in the conventional LMS,

* This research was supported by the Israeli Ministry of economy Magneton program.

[1]In G.fast there are 2048/4096 sub-carriers and discrete multitone (DMT) symbol rate of 48kSymbol/sec [2].

978-0-9928-6265-7/16/$31.00 ©2016 IEEE

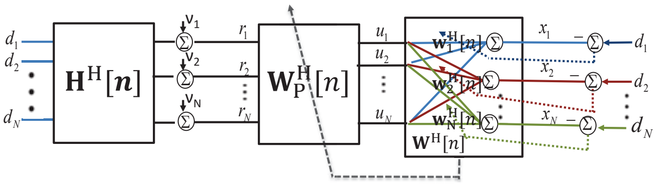

Fig. 1: The Deep-LMS crosstalk canceler.

but the actual filter that is applied to the data at each iteration n is a certain averaged version of previous filters. It should be mentioned that the averaging method show no asymptotic improvement, i.e., when a decreasing step-size is used the optimal rate of convergence of both systems are the same.

As mentioned above, at high frequencies, G.fast channel matrices typically have no diagonal dominant structure. Hence, at these frequencies the received signal's covariance matrix is badly-conditioned even in cases where the direct channel gains of all users are of the same order. Therefore, using the conventional LMS algorithms for FEXT cancellation in G.fast systems is impractical due to the long convergence times.

In this paper, we propose a new algorithm dubbed Deep-LMS, which enjoys almost the same complexity as the conventional LMS algorithm but exhibits a much faster convergence. The key idea behind the proposed algorithm is summarized by: the asymptotic convergence of the LMS algorithm is guaranteed if the step size is properly chosen. Thus, some progress must be achieved by the LMS algorithm, i.e., if we wait a sufficient long time. This progress is harnessed by the Deep-LMS by updating the adaptive preprocessing matrix such that the effective channel matrix becomes more diagonal dominant. Hence, the convergence of the overall system is accelerated.

The update of the preprocessing matrix inspired from deep neural networks. However, since all the operations in the Deep-LMS algorithm are linear, we are able to derive an exact analysis. In particular, we prove that if the initial interference plus noise ratio (SINR) of all users is sufficiently high, the rate of convergence of the LMS crosstalk canceler is accelerated at each update of the preprocessing matrix (assuming that there are enough iterations between the updates).

II. System model

In this section we present the model of the received signal. For simplicity of presentation, the model is described for a single frequency bin. The operation at other frequency bins is similar and independent of the operation at any other bin.

Let $\mathbf{d}[n]$ denote the transmitted data symbols at time n of all N users at a certain frequency bin, where $\mathbf{d}[n]$ is drawn with Gaussian i.i.d components and is assumed to be known at the receiver. More formally, $\mathbf{d}[n]$ is a zero mean Gaussian vector with a correlation matrix of $E\{\mathbf{d}[n]\mathbf{d}^H[n]\} = \mathbf{I}$ and $E\{\mathbf{d}[n]\mathbf{d}^H[m]\} = \mathbf{0}$ for $m \neq n$, where $\mathbf{I}$ is a $N \times N$ identity matrix and $\mathbf{A}^H$ is the conjugate transpose of the matrix $\mathbf{A}$. The received signal at the output of the channel is

$$\check{\mathbf{r}}[n] = \sqrt{p}\mathbf{H}^H[n]\mathbf{d}[n] + \check{\boldsymbol{\nu}}[n], \tag{1}$$

where p is the transmitted power (at that frequency bin), $\mathbf{H}[n] = [\mathbf{h}_1[n], \mathbf{h}_2[n], \cdots, \mathbf{h}_N[n]]$ is the channel gain matrix, $\check{\boldsymbol{\nu}}[n]$ is an additive white Gaussian noise (AWGN), $\check{\boldsymbol{\nu}}[n] \in \mathcal{CN}\left(\mathbf{0}, \check{\sigma}_\nu^2\mathbf{I}\right)$. For convenience, rewrite equation (1) by normalizing it with respect to the transmitting power:

$$\mathbf{r}[n] = \mathbf{H}^H[n]\mathbf{d}[n] + \boldsymbol{\nu}[n], \tag{2}$$

such that $\mathbf{r}[n] = \check{\mathbf{r}}[n]/\sqrt{p}$, $\boldsymbol{\nu}[n] \in \mathcal{CN}\left(\mathbf{0}, \sigma_\nu^2\mathbf{I}\right)$ and $\sigma_\nu^2 = \frac{\check{\sigma}_\nu^2}{p}$.

III. The conventional LMS crosstalk canceler

In this section we describe the model for a multi-user DSL system with a conventional LMS crosstalk canceler. A LMS crosstalk canceler for the upstream transmission of N users can be easily implemented by N parallel LMS blocks. To avoid confusion (for reasons that will become clearer in what follows) we denote the input signal (vector) to the LMS by $\mathbf{u}[n]$. In the conventional approach, the LMS crosstalk canceler is directly applied on the received signal, i.e., $\mathbf{u}[n] = \mathbf{r}[n]$. The LMS crosstalk canceler (in each tone) can be described as a matrix

$$\mathbf{W}[n] \triangleq [\mathbf{w}_1[n], \mathbf{w}_2[n], \cdots, \mathbf{w}_N[n]]. \tag{3}$$

Hence, the output of the LMS crosstalk canceler can be written as $\mathbf{x}[n] = \mathbf{W}^H\mathbf{u}[n]$ and the LMS recursion can be written in a matrix form as

$$\mathbf{W}[n+1] = \mathbf{W}[n] - 2\mu\mathbf{u}[n]\mathbf{e}^H[n], \tag{4}$$

where $\mathbf{e}[n] = \mathbf{d}[n] - \mathbf{x}[n]$.

IV. The Deep-LMS algorithm

In this section, we present the novel Deep-LMS crosstalk canceler. Unlike the conventional approach, in this algorithm the LMS $\mathbf{W}[n]$ is not applied directly on the received signal. Instead, the received signal is preprocessed by a matrix $\mathbf{W}_{\mathrm{P}}$. Hence, the input of the adaptive block can be written as

$$\mathbf{u}[n] = \mathbf{W}_{\mathrm{P}}^H[n]\mathbf{r}[n]. \tag{5}$$

The model is illustrated in Fig .1.

The algorithm is initialized with an identity preprocessing matrix and the preprocessing matrix is updated only at certain time instances. Denote the set of all update time instances by $\mathcal{U}$. The algorithm is summarized with the following set of equations:

$$\mathbf{W}[0] = \mathbf{I} \tag{6}$$

$$\mathbf{W}_{\mathrm{P}}[0] = \tilde{\mathbf{D}}[0] \tag{7}$$

$$\mathbf{u}[n] = \mathbf{W}_{\mathrm{P}}^H[n]\mathbf{r}[n] \tag{8}$$

$$\mathbf{x}[n] = \mathbf{W}^H[n]\mathbf{u}[n] \tag{9}$$

$$\mathbf{e}[n] = \mathbf{d}[n] - \mathbf{x}[n] \tag{10}$$

$$\check{\mathbf{W}}[n+1] = \mathbf{W}[n] + 2\mu\mathbf{u}[n]\mathbf{e}^H[n] \tag{11}$$

$$\mathbf{W}[n+1] = \begin{cases} \check{\mathbf{W}}[n+1] & n \notin \mathcal{U} \\ \mathbf{I} & n \in \mathcal{U} \end{cases} \tag{12}$$

$$\mathbf{W}_{\mathrm{P}}[n+1] = \begin{cases} \mathbf{W}_{\mathrm{P}}[n] & n \notin \mathcal{U} \\ \mathbf{W}_{\mathrm{P}}[n]\breve{\mathbf{W}}[n+1]\tilde{\mathbf{D}}[n+1] & n \in \mathcal{U} \end{cases} \quad (13)$$

where $\tilde{\mathbf{D}}[n]$ is the diagonal normalization matrix that satisfies $(\mathbf{H}\mathbf{W}_{\mathrm{P}}[n])_{i,i} = 1$ for all i. In other words, the values of the diagonal of $\tilde{\mathbf{D}}[n]$ are the inverse of the direct effective channel at the input of the LMS block[2]. The step size μ was selected as in [14] (an explicitly expresion will be given in Theorem 1 after establishing the required notations). As can be seen, for $n \notin \mathcal{U}$ the preprocessing matrix remains unchanged and the LMS block $\mathbf{W}[n]$ is updated exactly as in the conventional LMS crosstalk canceler. For $n \in \mathcal{U}$ the preprocessing matrix $\mathbf{W}_{\mathrm{P}}[n]$ is updated to include the combined effect of the preprocessing and the current LMS, and the LMS $\mathbf{W}[n]$ is initiated back to $\mathbf{I}$.

The role of the preprocessing matrix is to speed-up the convergence of the LMS crosstalk canceler $\mathbf{W}[n]$ between the updates, i.e., $n \notin \mathcal{U}$ and hence speed-up the convergence of the overall system.

We study the convergence of the Deep-LMS algorithm through the characterization of the minimal SINR at the input and the output of the LMS crosstalk canceler between two consecutive updates of the preprocessing matrix. The main result of this paper (Theorem 1) states that if the update time was properly chosen, the rate of the improvement in the SINR at the output of the Deep-LMS will be higher after the update of the preprocessing matrix.

Formally, let $\tilde{\mathbf{H}}[n] = \mathbf{W}_{\mathrm{P}}[n]\mathbf{H}[n]$ denote the effective channel matrix for the LMS crosstalk canceler. The SINR at the i-th input of the LMS crosstalk canceler, i.e., the SINR that is measured in $u_i[n]$, see Fig. 1 is given by:

$$\Phi_i[n] = \frac{|\tilde{h}_{i,i}[n]|^2}{\sum_{j\neq i} |\tilde{h}_{j,i}[n]|^2 + \tilde{\sigma}_i[n]}, \quad (14)$$

where $\tilde{\sigma}_i[n]$ is the variance of the i-th entry of the colored noise $\tilde{\boldsymbol{\nu}}[n] = \mathbf{W}_{\mathrm{P}}^H[n]\boldsymbol{\nu}[n]$. The minimal input SINR to the LMS block is defined as $\Phi[n] = \min_i \Phi_i[n]$.

In the following analysis, we focus on the set of times between two updates of the preprocessing matrix, i.e., the set of times $\mathcal{F}_\ell \triangleq \{n_\ell, n_\ell + 1, \ldots, n_{\ell+1} - 1\}$, where $n_\ell \in \mathcal{U}$ and $n_{\ell+1} = \min_{n\in\mathcal{U}}\{n > n_\ell\}$, i.e., $n_{\ell+1}$ is the first update of the preprocessing matrix $\mathbf{W}_{\mathrm{P}}[n]$ after time n_ℓ. Note that $\mathbf{W}_{\mathrm{P}}[n]$ and $\tilde{\mathbf{H}}[n]$ do not change during the analyzed interval, and hence $\Phi[k] = \Phi[m]$ for any $k, m \in \mathcal{F}_\ell$. Therefore, it will be convenient to denote the minimal input SINR at these times by Φ_ℓ. The same argument applies to the covariance matrix $\mathbf{R}_\ell = E\{\mathbf{u}[n]\mathbf{u}^H[n]\}$. Also note that at time $n_\ell \in \mathcal{U}$, the LMS crosstalk canceler is initiated and therefore $\mathbf{W}[n_\ell] = \mathbf{I}$. Hence, at the update time instance, the input SINR to the LMS block is also the output SINR of the the Deep-LMS, i.e., the SINR that is measured in $x_i[n]$.

Theorem 1. *If $\mu = \frac{1}{3\mathrm{Tr}(\mathbf{R}_\ell)}$, the minimal SINR at time $n_{\ell+1}$ is lower bounded by*

$$\Phi_{\ell+1} \geq \left(\left(c \cdot a^{n_{\ell+1}-n_\ell}\Phi_\ell\right)^{-1} + \eta_\infty\right)^{-1} - 1, \quad (15)$$

where $c = \left(1 + \delta(\Phi)\right)^{-1}$, $a = \left(1 - \beta(\Phi)\right)^{-1}$, $\delta(\Phi) = \frac{1+2\alpha(\Phi)}{\Phi-\alpha(\Phi)}$, $\beta(\Phi) = \frac{8}{9}g\left(\frac{1}{N}(1 - \frac{\alpha(\Phi)+1}{\Phi+1})\right)$, $\alpha(\Phi) = \left(N - 1 + \sqrt{N-1}\right)\sqrt{\Phi} + 2(N-1)$, $g(x) = x - \frac{1}{2}x^2$ *for complex LMS and* $g(x) = x - x^2$ *for LMS over the reals and* η_∞ *is the maximal MSE of the optimal solution at steady state.*

Proof. As was mentioned above, the convergence rate of the LMS depends on the correlation matrix of the input signal, and more specifically, on the spread of the eigenvalues of this matrix. The proof uses the knowledge of the minimal input SINR to bound the spread of these eigenvalues. Then, we use a novel bounding analysis that gives the desired bound on the output SINR. However, due to space constraints, the proof is omitted from this paper, and will be given in [20]. □

Note that η_∞ approaches zero when the variance of the noise approaches zero. This fact is highly relevant in DSL systems, where the crosstalk interference is much stronger than the background noise. Thus, η_∞ is typically negligible and we can write:

$$\Phi_{\ell+1} \gtrsim c \cdot a^{n_{\ell+1}-n_\ell}\Phi_\ell. \quad (16)$$

Furthermore, if $\Phi_\ell > 1.5N^2 + 3N$, both c and $a > 1$ are monotonically increasing functions of Φ_ℓ. Hence, updating the preprocessing matrix $\mathbf{W}_{\mathrm{P}}$ at (sufficiently large) time $n_{\ell+1}$, i.e., when $c \cdot a^{n_{\ell+1}-n_\ell}$ is large enough, will improve the input SINR of the LMS component at times $n > n_{\ell+1}$. Once the preprocessing matrix is updated, the (new) minimal SINR at the input of the LMS is improved accordingly, leading to an increase in c and a, and hence to a faster convergence rate of the LMS. Hence, each update of the preprocessing matrix speeds up the convergence of the entire system.

V. Numerical Results

In this section we demonstrate the performance of the Deep-LMS algorithm and compare it to the performance of the conventional LMS. We also study the performance of the accelerated versions where the LMS component is replaced by its averaged version.

All channel matrices in this section were drawn randomly according to the model in [21]. In particular, at each frequency f in a cable of length ℓ, the IL can be modeled as

$$|H_{ii}(f, \ell)|^2 = e^{-2\alpha\ell\sqrt{f}} \quad (17)$$

and the FEXT can be modeled as

$$|H_{ij}(f, \ell)|^2 = |H_{ii}(f, \ell)|^2 K(\ell) f^2 \qquad i \neq j, \quad (18)$$

[2]The estimation of the direct channel is an essential part of the demodulation process in DSL. For the analysis of the algorithm, we assumed that this estimation is accurate.

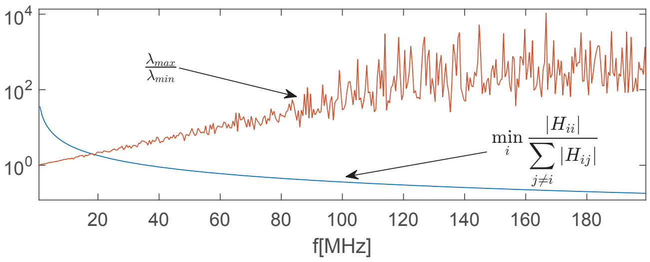

Fig. 2: An illustration of the eigenvalues spread of the correlation matrix of the received signal and the non-diagonal dominant structure of the channel matrix at high frequencies.

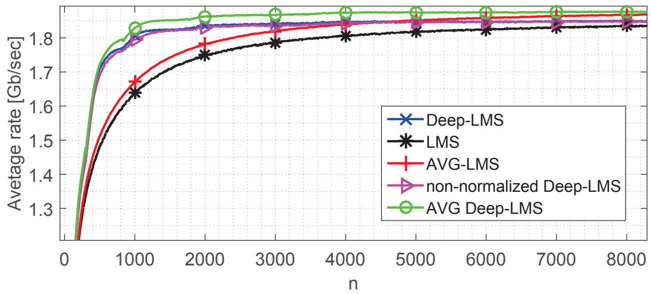

Fig. 3: Average sum rate comparison (G.fast frequency bins 17MHz-200MHz).

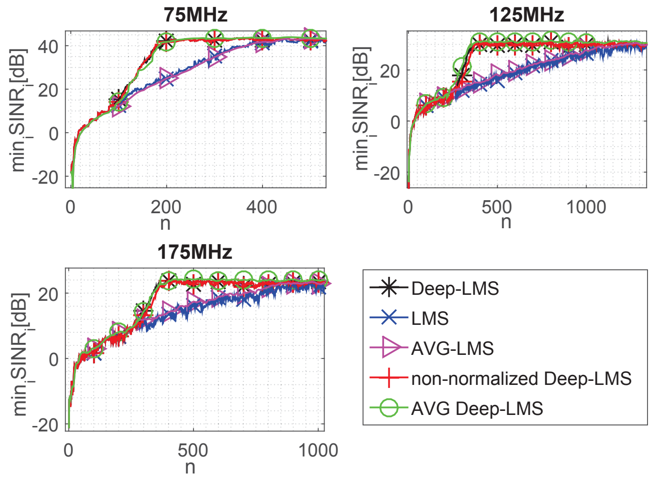

Fig. 4: Minimal output SINR (over all users) comparison at certain frequency bins.

where $\alpha = 3.72 \cdot 10^{-6}$ and $K(\ell)$ is a log-normal random variable and we set $\ell = 100$m. The phase of all lines is uniformly distributed over $[0, 2\pi)$. We simulate a typical upstream scenario with a carrier spacing of $W = 497.5$KHz a transmit PSD mask as in [22] and a colored background noise of -140dBm/Hz below 30MHz and -150dBm/Hz above 30MHz [23].

An illustration of the eigenvalues spread of the received signal correlation matrix is shown in Fig 2. The figure also illustrates the non-diagonal dominant structure of the channel matrix at high frequencies. The diagonal dominance is measured by the ratio between the diagonal term and the sum of the absolute value of all non diagonal terms [24]. Most traditional DSL algorithms require this ratio to be larger than 1. Thus Fig. 2 demonstrates the need for novel vectoring algorithms that can operate in the G.fast bandwidth.

Before we present the performance of the Deep-LMS algorithm, we need to characterize the choice of update times. Moreover, as an inherent part of the bit-loading component of the DMT system in DSL systems, an estimation of the SINR at the output of the crosstalk canceler is available to the modem. Hence, the set of update times $\mathcal{U}$ can be easily determined in real time using the measured SINR.

We update the preprocessing matrix of the Deep-LMS whenever the SINR is improved by 5dB or when more than $\tilde{n}$ iterations have passed since the last non-SINR based update of the preprocessing matrix.

Fig. 3 shows the average sum rate per user in each iteration using the Deep-LMS and the traditional LMS. The sum rate of each user was computed by:

$$R_t = W \sum_{k=1}^{K} \left[\log_2 \left(1 + \mathrm{SINR_{i,k}[n]}\right) \right]_{b_{\max}} \tag{19}$$

where $\mathrm{SINR_{i,k}[n]}$ is the SINR at the output of each algorithm at the i-th user at the k-th frequency, $b_{\max} = 12$ is the maximal number of bits per DMT frequency bin as defined in G.fast, [2]) and $[x]_b = \min\{x, b\}$. As can be seen the Deep-LMS algorithm converges much faster than the traditional LMS, for example, it reaches 1.7Gbps in one third of the time.

Fig. 3 also depicts the performance of the accelerated LMS that uses filter averaging of [19]. In this algorithm (marked as AVG-LMS) the LMS component was updated as in the conventional LMS (4) but the actual filter that was applied to the data at each iteration n was an averaged weight matrix $\sum_{i=0}^{n} \theta^{n-i} \mathbf{W}[i]$ with a forgetting factor of $\theta = 0.95$. An analysis of a stochastic approximation with such an averaging method can be found for example in [25]. This averaging indeed accelerate the convergence, but is still much slower than the Deep-LMS, Furthermore, the same averaging technique can also be applied to the Deep-LMS (marked as AVG Deep-LMS), resulting in an even faster convergence.

To further simplify the algorithm, we also tested the performance when $\tilde{\mathbf{D}}$ is set to $\mathbf{I}$ in (13), i.e., when we violated the assumption that the preprocessing matrix is normalized such that the direct effective channel gain of all users is 1. As can be seen, the loss due to this simplification is negligible. Hence, we conjecture that this normalization is required mostly for the analysis, but has no significant effect on the actual performance.

To better illustrate the behavior of the different algorithms, Fig. 4 shows the minimal SINR at the output of the adaptive crosstalk canceler for each of the algorithms in specific frequency bins. This figure gives a better understanding of the nature of the proposed Deep-LMS algorithm. It shows that the

Deep-LMS starts exactly the same as the traditional LMS. But, when the traditional LMS manages to improve the SINR above certain point, the Deep-LMS takes advantage of this improved SINR to gain a significant increase in the convergence rate.

VI. Conclusions

In this paper we presented a new LMS-based crosstalk canceler for the upstream transmission in G.fast systems. The new crosstalk canceler preprocesses the received signal using an adaptive matrix prior to a conventional LMS crosstalk canceler. This preprocessing matrix is initiated by the identity matrix and at any update of the preprocessing matrix it is set into the product of the current LMS crosstalk canceler and the current preprocessing matrix. The main goal of the preprocessing matrix is to alter the effective channel matrix into a diagonal dominant structure. We showed that the method can be used to speed up the convergence of the entire system given that the SINR is sufficiently high. Since, the preprocessing matrix is not frequently updated, the complexity of the algorithm is approximately twice the complexity of the conventional LMS.

References

[1] J. Bingham, *ADSL, VDSL, and multicarrier modulation*. Wiley New York, 2000.

[2] ITU-T G.9700, "Fast access to subscriber terminals (G.fast) Power spectral density specification," 2014.

[3] J. Cioffi, M. Brady, V. Pourahmad, S. Jagannathan, W. Lee, Y. Kim, C. Chen, K. Seong, D. Yu, and M. Ouzzif, "Vectored DSLs with DSM: the road to ubiquitous gigabit DSLs," 2006.

[4] S. Huberman, C. Leung, and T. Le-Ngoc, "Dynamic spectrum management (DSM) algorithms for multi-user xDSL," *IEEE Communications Surveys & Tutorials*, vol. 14, no. 1, pp. 109–130, 2012.

[5] I. Bergel and A. Leshem, "Signal Processing for Vectored Multichannel VDSL," *Academic Press Library in Signal Processing: Communications and Radar Signal Processing*, vol. 2, p. 295, 2013.

[6] J. Cioffi, M. Mosheni, A. Leshem, and Y. Li, "GDSL (Gigabit DSL)," *T1E1.4 contribution T1E1.4/2003-487R1*, 2003.

[7] A. Leshem and L. Youming, "A low complexity linear precoding technique for next generation VDSL downstream transmission over copper," *IEEE Transactions on Signal Processing*, vol. 55, no. 11, pp. 5527–5534, 2007.

[8] I. Bergel and A. Leshem, "Convergence analysis of downstream VDSL adaptive multichannel partial FEXT cancellation," *IEEE Transactions on Communications*, vol. 58, no. 10, pp. 3021–3027, 2010.

[9] I. Binyamini, I. Bergel, and A. Leshem, "Arbitrary partial FEXT cancellation in adaptive precoding for multichannel downstream VDSL," *IEEE Transactions on Signal Processing*, vol. 60, no. 11, pp. 5754–5763, 2012.

[10] R. Cendrillon, G. Ginis, and M. Moonen, "A near-optimal linear crosstalk precoder for downstream VDSL," *IEEE Transactions on Communications*, vol. 55, no. 5, pp. 860–863, 2007.

[11] B. Widrow and M. Hoff, "Adaptive switching circuits," 1960.

[12] B. Widrow, "Adaptive filters," *Aspects of network and system theory*, pp. 563–587, 1971.

[13] L. Horowitz and K. Senne, "Performance advantage of complex LMS for controlling narrow-band adaptive arrays," *IEEE Transactions on Circuits and Systems*, vol. 28, no. 6, pp. 562–576, 1981.

[14] A. Feuer and E. Weinstein, "Convergence analysis of LMS filters with uncorrelated Gaussian data," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 33, no. 1, pp. 222–230, 1985.

[15] J. Nagumo and A. Noda, "A learning method for system identification," *IEEE Transactions on Automatic Control*, vol. 12, no. 3, pp. 282–287, 1967.

[16] R. Harris, D. Chabries, and F. Bishop, "A variable step (VS) adaptive filter algorithm," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 34, no. 2, pp. 309–316, Apr 1986.

[17] H. Kushner and G. Yin, *Stochastic approximation and recursive algorithms and applications*. Springer Science & Business Media, 2003, vol. 35.

[18] K. Mayyas and T. Aboulnasr, "Leaky LMS algorithm: MSE analysis for Gaussian data," *IEEE Transactions on Signal Processing*, vol. 45, no. 4, pp. 927–934, Apr 1997.

[19] H. J. Kushner and J. Yang, "Stochastic approximation with averaging of the iterates: Optimal asymptotic rate of convergence for general processes," *SIAM Journal on Control and Optimization*, vol. 31, no. 4, pp. 1045–1062, 1993.

[20] A. Zanko, I. Bergel, and A. Leshem, "Deep-LMS for Gigabit transmission over unshielded twisted pair cables," submitted to IEEE Transactions on Signal Processing, 2016.

[21] E. Karipidis, N. Sidiropoulos, A. Leshem, L. Youming, R. Tarafi, and M. Ouzzif, "Crosstalk models for short VDSL2 lines from measured 30MHz data," *EURASIP Journal on applied signal processing*, vol. 2006, pp. 90–90, 2006.

[22] ITU-T G.9701, "Fast Access to Subscriber Terminals (FAST) Physical layer specification," 2014.

[23] *Coexistence of G. fast and VDSL in FTTdp and FTTC Deploy-ments*. Proceedings of the 23rd European Signal Processing Conference (EUSIPCO-2015), 2015.

[24] I. Bergel and A. Leshem, "The performance of zero forcing DSL systems," *IEEE Signal Processing Letters*, vol. 20, no. 5, pp. 527–530, 2013.

[25] I. Wang, E. Chong, and S. Kulkarni, "Weighted averaging and stochastic approximation," *Mathematics of Control, Signals and Systems*, vol. 10, no. 1, pp. 41–60, 1997.