A TENSOR LMS ALGORITHM

Markus Rupp, Stefan Schwarz[*]

Technical University of Vienna, Austria
Institute of Telecommunications
Email: {mrupp,sschwarz}@nt.tuwien.ac.at

ABSTRACT

Although the LMS algorithm is often preferred in practice due to its numerous positive implementation properties, once the parameter space to estimate becomes large, the algorithm suffers of slow learning. Many ideas have been proposed to introduce some a-priori knowledge into the algorithm to speed up its learning rate. Recently also sparsity concepts have become of interest for such algorithms. In this contribution we follow a different path by focusing on the separability of linear operators, a typical property of interest when dealing with tensors. Once such separability property is given, a gradient type algorithm can be derived with significant increase in learning rate. Even if separability is only given to a certain extent, we show that the algorithm can still provide gains. We derive quality and quantity measures to describe the algorithmic behavior in such contexts and evaluate its properties by Monte Carlo simulations.

Index Terms— Tensor, LMS algorithm, Separability

1. INTRODUCTION

The Least Mean Square (LMS) algorithm [1–3] as a canonical form of a gradient type algorithm has received much attention over more than five decades. In practice it is the algorithm that is implemented in myriads of variants to achieve parametric learning.

1.1. Relation to Prior Work

While it exhibits many desirable properties for its implementations, it suffers greatly if the parameter space becomes too large. At best the learning rate is about $20dB/5M$ [3,4] if M denotes the number of estimation parameters. Depending on the problem, many ideas have been proposed to adapt the step-size [5, 6] and to include a-priori knowledge into the update equations with the goal to achieve faster convergence [7]. Also sparsity has been discovered as beneficial property [8, 9] offering potentials to increase learning rates. Recent approaches include sparsity constraints directly in the formulation of the algorithm exploiting l_0-norms [10, 11]. We on the other hand will not assume sparsity but that a repetitive but not periodic structure is imposed. Such structure can typically be described in form of Kronecker products of vectors, matrices or tensors. Tensors and their decomposition have attracted much interest recently [12] due to the multitude of potential applications of so-called *Big Data* [13, 14].

[*]This work was supported by the Austrian Science Fund (FWF) under Awards S10611-N13 within the National Research Network SISE.

1.2. Our Contribution

In this contribution we will follow a new path, not employing sparsity. We assume that the impulse response $\mathbf{v}$ that needs to be identified can be described by separable partitions, e.g., in form of a three-way tensor $\mathbf{v} = \mathbf{c} \otimes \mathbf{b} \otimes \mathbf{a}$. Such repetitive but not periodic behavior typically occurs if reflections dominate the impulse response as if often the case in wireless transmissions. However, our method does not rely on a perfect separability. We show that even if only parts of the impulse response are separable, the algorithm can gain performance.

1.3. Paper Structure

This paper provides in Section 2 a short introduction into separability in the Least Squares (LS) sense. Based on these findings, the proposed tensor LMS algorithm is derived in Section 3. We demonstrate the algorithm behavior on some experimental examples in Section 4 and eventually we conclude the contribution in Section 5.

1.4. Notation

We describe the transpose of a vector or matrix by upperscript T and denote the Kronecker product by $\otimes$. All signals are considered real valued. The operator vec() realigns a matrix column by column into a vector. Further we denote the set of right side eigenvectors of a matrix by evec().

2. LINEARLY SEPARABLE OPERATORS

Linearly separable operators are well known in the context of fast algorithmic implementations such as Fast Hadamard Transformation (FHT) or Fast Fourier Transformations (FFT). We provide here a copy of the definition of separability for the convenience of the reader, see, e.g., [15].

Definition 2.1 *A linear operator* $\mathbf{A}$ *is said to be separable if* $\mathbf{A} = \mathbf{A}_1 \otimes \mathbf{A}_2$ *for some* $\mathbf{A}_1$ *and* $\mathbf{A}_2$.

Obviously due to the product term, there is not a unique solution, if it exists. While it is straightforward to show how to save complexity once such separability is present (typically the savings are from M^2 to $M \log_2(M)$ which can be considerable if M is large), to show if a vector or matrix is separable at all is not a simple step. We thus show in this section how it can be achieved in the LS sense, given vectors of length M that take on the position of linear operators. This is in fact the same problem as decomposing a rank one tensor. In case of Definition 2.1 it is a two-way tensor, but generalizations to, e.g., three way tensors are straightforward. The presented

approach here follows the general concept of [16] that is based on perfectly separable matrices. It was adapted to make it suitable for long adaptive filters under general impulse responses.

Theorem 2.1 (LS Approximation) *Consider a vector* $\mathbf{v} \in I\!R^{(M\times 1)}$ *with* $M = NP$*, i.e.,* $\mathbf{v} = [\mathbf{v}_1; \mathbf{v}_2; ...; \mathbf{v}_P]$ *with* $\mathbf{v}_l \in I\!R^{(N\times 1)}; l = 1, 2, ..., P$*. There uniquely exist two vectors* $\mathbf{a} \in I\!R^{N\times 1}, \mathbf{b} \in I\!R^{P\times 1}$ *with* $\|\mathbf{a}\|_2 = 1$ *such that*

$$\{\mathbf{a}, \mathbf{b}\} = \arg\min_{\mathbf{a},\mathbf{b}} \|\mathbf{v} - \mathbf{b} \otimes \mathbf{a}\|_2^2$$

in the LS sense.

Proof: We first compute the elements of the vector $\mathbf{b}$ by LS and obtain

$$\frac{\partial}{\partial b_k} \sum_{l=1}^{P} \|\mathbf{v}_l - b_l\mathbf{a}\|_2^2 = 0$$

that is

$$b_k = \frac{\mathbf{a}^T\mathbf{v}_k}{\mathbf{a}^T\mathbf{a}}; k = 1, 2, ..., P.$$

Substituting b_k into the original formulation we find

$$\min_{\mathbf{a}} \sum_{l=1}^{P} \left\| \mathbf{v}_l - \frac{\mathbf{a}\mathbf{a}^T}{\mathbf{a}^T\mathbf{a}}\mathbf{v}_l \right\|_2^2$$

which is due to the projection operation equivalent to

$$\min_{\mathbf{a}} \sum_{l=1}^{P} \mathbf{v}_l^T \left[\mathbf{I}_N - \frac{\mathbf{a}\mathbf{a}^T}{\mathbf{a}^T\mathbf{a}} \right] \mathbf{v}_l.$$

The equivalent expression $\sum_{l=1}^{P} \mathbf{v}_l^T\mathbf{v}_l - \frac{\mathbf{a}^T\mathbf{v}_l\mathbf{v}_l^T\mathbf{a}}{\mathbf{a}^T\mathbf{a}}$ is minimized by selecting $\mathbf{a}$ the eigenvector to the largest eigenvalue of

$$\mathbf{a} = \arg\max \text{evec} \sum_{l=1}^{P} \mathbf{v}_l\mathbf{v}_l^T.$$

Theorem 2.2 (Orthogonality) *Given a set of vectors* $\{\mathbf{a}, \mathbf{b}\}$ *according to Theorem 2.1 to separate a vector* $\mathbf{v}$ *in the LS sense,* $\mathbf{v}$ *can exactly be represented by*

$$\mathbf{v} = \mathbf{b} \otimes \mathbf{a} + \mathbf{w}$$

with error vector $\mathbf{w}$ *being orthogonal onto the tensor product*

$$\mathbf{w}^T(\mathbf{b} \otimes \mathbf{a}) = 0.$$

Proof: The approximation $\mathbf{b}\otimes\mathbf{a} = \left[\mathbf{I}_P \otimes \frac{\mathbf{a}\mathbf{a}^T}{\mathbf{a}^T\mathbf{a}}\right] \mathbf{v}$. The error vector $\mathbf{w} = \mathbf{v} - \mathbf{b} \otimes \mathbf{a}; \mathbf{w} \in I\!R^{M\times 1}$ is indeed orthogonal to the approximation $\mathbf{b} \otimes \mathbf{a}$ as required for an LS estimation:

$$\mathbf{w} = \left(\mathbf{I}_M - \mathbf{I}_P \otimes \frac{\mathbf{a}\mathbf{a}^T}{\mathbf{a}^T\mathbf{a}} \right) \mathbf{v}$$

We write the error also in concatenated form $\mathbf{w} = [\mathbf{w}_1; \mathbf{w}_2; ...; \mathbf{w}_P]$; $\mathbf{w}_k \in I\!R^{N\times 1}; k = 1, 2, ..., P$ and find for each partition $\mathbf{w}_k = \left[\mathbf{I}_N - \frac{\mathbf{a}\mathbf{a}^T}{\mathbf{a}^T\mathbf{a}}\right] \mathbf{v}_k$ which is orthogonal onto the approximation $\frac{\mathbf{a}\mathbf{a}^T}{\mathbf{a}^T\mathbf{a}}\mathbf{v}_k$.

Theorem 2.3 (MMSE) *Separating a two-way rank one tensor in the LS sense as described in the previous Theorems 2.1 and 2.2, its corresponding minimum MSE (MMSE) is given by*

$$MMSE = \sum_{k=1}^{P} \mathbf{w}_k^T\mathbf{w}_k = \sum_{k=1}^{P} \mathbf{v}_k \left(\mathbf{I}_N - \frac{\mathbf{a}\mathbf{a}^T}{\mathbf{a}^T\mathbf{a}} \right) \mathbf{v}_k.$$

Proof: The proof follows directly from the orthogonality property of LS.

Note: if several partitions of $M = NP$ exist, the various MMSE values can be checked and based on those, it can be decided which partitioning is most suitable to linear separation. The normalized MMSE value, i.e.,

$$\gamma(N, P) = \frac{\sum_{k=1}^{P} \mathbf{w}_k^T\mathbf{w}_k}{\sum_{k=1}^{P} \mathbf{v}_k^T\mathbf{v}_k} \leq 1$$

provides a measure for the quality of the separation. We will later use such measure to describe the suitability of the proposed gradient approach for identifying such systems. The smaller the value $\gamma(N, P)$, the better is the separation property of the system. Ideally the norms $\|\mathbf{w}_k\|_2^2$ should be zero to obtain a perfect separability. If however some terms are larger, they indicate outliers or anomalies, thus an easy method to detect them and if necessary treat them with particular methods.

3. A TENSOR LMS ALGORITHM

Let us assume that the impulse response that is to estimate can be separated as $\mathbf{v} = \mathbf{b} \otimes \mathbf{a}$. Then it is sufficient to estimate $\hat{\mathbf{a}}$ and $\hat{\mathbf{b}}$ in order to reproduce $\hat{\mathbf{v}} = \hat{\mathbf{b}} \otimes \hat{\mathbf{a}}$. We assume that $\mathbf{v}$ is of length $M = M_aM_b$, thus $\mathbf{a} \in I\!R^{M_a\times 1}$ and $\mathbf{b} \in I\!R^{M_b\times 1}$. Let us consider the convolution with an input signal at time instant k defined by a vector $\mathbf{x}_k = [\mathbf{x}_{k,1}, \mathbf{x}_{k,2}, ..., \mathbf{x}_{k,M_b}] \in I\!R^{M\times 1}$ which can be split in M_b partitions of length M_a. We find that

$$\mathbf{v}^T\mathbf{x}_k = (\mathbf{b} \otimes \mathbf{a})^T \mathbf{x}_k = \sum_{l=1}^{M_b} b_l\mathbf{a}^T\mathbf{x}_{kl} = \mathbf{a}^T\mathbf{X}_k\mathbf{b},$$

where we reassembled the partitions of $\mathbf{x}_k$ line by line in form of a matrix $\mathbf{X}_k$ (two-way tensor). This convolution can be further formulated as

$$\mathbf{a}^T\mathbf{X}_k\mathbf{b} = \mathbf{y}^T\mathbf{b} = \mathbf{a}^T\mathbf{z},$$

by which we introduced short notations of the partial convolutions $\mathbf{y} = \mathbf{a}^T\mathbf{X}_k$ and $\mathbf{z} = \mathbf{X}_k\mathbf{b}$.

Assume d_k to be the noisy output observation and $\{\hat{\mathbf{a}}_{k-1}, \hat{\mathbf{b}}_{k-1}, \hat{\mathbf{y}}_k = \hat{\mathbf{a}}_{k-1}^T\mathbf{X}_k, \hat{\mathbf{z}}_k = \mathbf{X}_k\hat{\mathbf{b}}_{k-1}\}$ the estimates at time instant k of $\{\mathbf{a}, \mathbf{b}, \mathbf{a}_k, \mathbf{z}_k\}$, respectively. Deriving the LMS algorithm now in partitions with respect to $\mathbf{a}$ and $\mathbf{b}$, we obtain

$$e_k = d_k - \hat{\mathbf{a}}_{k-1}^T\mathbf{X}_k\hat{\mathbf{b}}_{k-1}, \quad (1)$$

$$\hat{\mathbf{a}}_k = \hat{\mathbf{a}}_{k-1} + \mu_{a,k}\hat{\mathbf{z}}_k e_k, \quad (2)$$

$$\hat{\mathbf{b}}_k = \hat{\mathbf{b}}_{k-1} + \mu_{b,k}\hat{\mathbf{y}}_k e_k. \quad (3)$$

Convergence The convergence analysis turns out to be complicated due to the cascade nature of the algorithm. The parameters $\hat{\mathbf{a}}_k$ depend on $\hat{\mathbf{b}}_k$ and vice versa. For cascaded algorithms [17, 18] it is known that they typically show only local [19–22] and not global robustness but behave well in the mean squared error (MSE) sense.

The undistorted error $e_{a,k}$ can be written in numerous ways:

$$\begin{aligned} e_{a,k} &= \mathbf{a}^T\mathbf{X}_k\mathbf{b} - \hat{\mathbf{a}}^T\mathbf{X}_k\hat{\mathbf{b}}_{k-1} && (4)\\ &= (\mathbf{a}-\hat{\mathbf{a}}_{k-1})^T\mathbf{X}_k\mathbf{b} + \hat{\mathbf{a}}_{k-1}^T\mathbf{X}_k(\mathbf{b}-\hat{\mathbf{b}}_{k-1}) && (5)\\ &= \mathbf{a}^T\mathbf{X}_k(\mathbf{b}-\hat{\mathbf{b}}_{k-1}) + (\mathbf{a}-\hat{\mathbf{a}}_{k-1})^T\mathbf{X}_k\hat{\mathbf{b}}_{k-1} && (6)\\ &= \mathbf{a}^T\mathbf{X}_k(\mathbf{b}-\hat{\mathbf{b}}_{k-1}) + (\mathbf{a}-\hat{\mathbf{a}}_{k-1})^T\mathbf{X}_k\mathbf{b} \\ &\quad -(\mathbf{a}-\hat{\mathbf{a}}_{k-1})^T\mathbf{X}_k(\mathbf{b}-\hat{\mathbf{b}}_{k-1}). && (7)\end{aligned}$$

Introducing parameter error weights $\tilde{\mathbf{a}}_k = \mathbf{a} - \hat{\mathbf{a}}_k$ and $\tilde{\mathbf{b}}_k = \mathbf{b} - \hat{\mathbf{b}}_k$ we find for the algorithmic updates (ignoring the noise):

$$\begin{aligned} \tilde{\mathbf{a}}_k &= \tilde{\mathbf{a}}_{k-1} - \mu_{a,k}\hat{\mathbf{z}}_k[\mathbf{y}_k^T\tilde{\mathbf{b}}_{k-1} + \tilde{\mathbf{a}}_{k-1}^T\hat{\mathbf{z}}_k] && (8)\\ \tilde{\mathbf{b}}_k &= \tilde{\mathbf{b}}_{k-1} - \mu_{b,k}\hat{\mathbf{y}}_k[\mathbf{z}_k^T\tilde{\mathbf{a}}_{k-1} + \tilde{\mathbf{b}}_{k-1}^T\hat{\mathbf{y}}_k] && (9)\end{aligned}$$

or equivalently

$$\begin{bmatrix} \tilde{\mathbf{a}}_k \\ \tilde{\mathbf{b}}_k \end{bmatrix} = \begin{bmatrix} \mathbf{I} - \mu_{a,k}\hat{\mathbf{z}}_k\hat{\mathbf{z}}_k^T & -\mu_{a,k}\hat{\mathbf{z}}_k\mathbf{y}_k^T \\ -\mu_{b,k}\hat{\mathbf{y}}_k\mathbf{z}_k^T & \mathbf{I} - \mu_{b,k}\hat{\mathbf{y}}_k\hat{\mathbf{y}}_k^T \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{a}}_{k-1} \\ \tilde{\mathbf{b}}_{k-1} \end{bmatrix} \quad (10)$$

which almost resembles a desired form $\mathbf{I} - \mu_k\bar{\mathbf{x}}_k\bar{\mathbf{x}}_k^T$ with $\bar{\mathbf{x}}_k = [\hat{\mathbf{z}}_k^T, \hat{\mathbf{y}}_k^T]^T$ but unfortunately only if $\hat{\mathbf{z}}_k = \mathbf{z}_k, \hat{\mathbf{y}}_k = \mathbf{y}_k$ and $\mu_k = \mu_{a,k} = \mu_{b,k}$. We are thus facing additional perturbation terms

$$\begin{aligned} \mu_{a,k}\hat{\mathbf{z}}_k\mathbf{y}_k^T &= \mu_{a,k}\hat{\mathbf{z}}_k\hat{\mathbf{y}}_k^T + \mu_{a,k}\hat{\mathbf{z}}_k[\mathbf{y}_k - \hat{\mathbf{y}}_k]^T && (11)\\ &= \mu_{a,k}\hat{\mathbf{z}}_k\hat{\mathbf{y}}_k^T + \mu_{a,k}\hat{\mathbf{z}}_k\tilde{\mathbf{a}}_{k-1}^T\mathbf{X}_k^T. && (12)\end{aligned}$$

We recognize that these perturbation terms are proportional to the parameter errors $\tilde{\mathbf{a}}_{k-1}$. Once the algorithm is close to the solution, it will converge rapidly. The existence of these off-diagonal terms prevents the existence of global robustness but applying expectation operators, they can be considered negligible.

Following along the lines of the analysis in [8], we can conclude convergence in the MSE sense as long as $\mu_k = \mu_{a,k} = \mu_{b,k}$

$$0 < \mu_k < \frac{2}{\|\hat{\mathbf{y}}_k\|_2^2 + \|\hat{\mathbf{z}}_k\|_2^2}. \quad (13)$$

Complexity The algorithmic complexity can be well below the standard complexity of $2M$. As the update equations only require $M_a + M_b$ operations, lots can be saved. Let us assume that $M = 2^{2k}$, then the optimal partitioning would be $M_a = M_b = 2^k$ and we save 2^{k-1} operations, thus we roughly reduce the complexity from M to $\sqrt{M}$. But also the computation in the error term can significantly be reduced from M operations if the filter structure is transversal. Operating in a block mode of $M_a(M_b)$ steps can save considerable complexity. Although such a block mode reduces learning speed, the strong increase in learning due to separability may compensate easily for this effect.

Extensions The algorithm shown here only requires a two-way tensor, that is a matrix. Nevertheless, extensions to arbitrary n-way tensors are straightforward. Take for example a system impulse response described by a three-way tensor. The output sequence reads

$$y_k = \sum_{n=0}^{M_c-1}\sum_{m=0}^{M_b-1}\sum_{l=0}^{M_a-1} a_l b_m c_n x_{k-(m+M_b l+M_a M_b n)}. \quad (14)$$

The regression vectors are simply obtained by a double convolution, omitting the term in (14) for which the gradient is to be computed, e.g., the corresponding terms to compute the $\mathbf{b}$ partitions are obtained by omitting the summation over m. In the following results, we present one experiment with a three-way tensor; the MATLAB code is freely available [23].

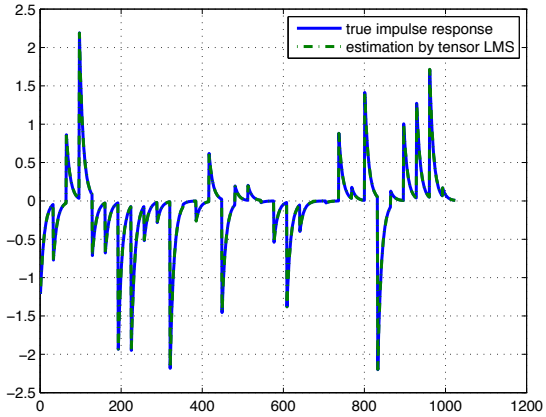

Fig. 1. *Typical impulse response with $M = 1024$ taps.*

4. EXPERIMENTAL RESULTS

In our first Monte Carlo (MC) example we consider $M = 1024$ weights to estimate. The reference model is based on a Kronecker product of $M_a = M_b = 32$ taps each. One set is exponentially decaying: $\mathbf{b}_k = 0.9^k; k = 0, 1, ..., 31$, while the other ($\mathbf{a}$) is selected randomly from $N(0, 1)$. By this we generate independent impulse responses for each MC run. Figure 1 depicts a typical impulse response. We average our results over 10 MC runs. The input signal x_k is simply from $N(0, 1)$. We add noise from $N(0, N_o)$ with $N_o = 10^{-2}$ power at the observed desired signal. We apply a normalized step-size $\mu_k = \alpha/[\|\hat{\mathbf{y}}_k\|_2^2 + \|\hat{\mathbf{z}}_k\|_2^2]$ and vary $\alpha \in (0, 2)$. For all step-sizes in this range we obtain stable behavior. Note that the filter taps need to be initialized with non-zero values as otherwise the adaptive filter stalls. Their actual initial values are uncritical. We simply set $\hat{\mathbf{a}}_{0,1} = \hat{\mathbf{b}}_{0,1} = 1$, i.e., their first value to one.

For our first experiment we depict the obtained relative system mismatch $\|\mathbf{v} - \hat{\mathbf{v}}_k\|_2^2/\|\mathbf{v}\|_2^2$ over the iterations k of the tensor LMS for a step-size $\alpha = 1$ (fastest learning for LMS) and compare to the standard LMS algorithm. Figure 2 exhibits the mismatch. As in the standard LMS algorithm the maximal speed is obtained for $\alpha = 1$ and the learning roughly $20dB/5[M_a+M_b]$. The steady state values for both algorithms show similar values at the same step-size α.

In our second example we add more realism by adding a part to the impulse response that is not separable. We achieve this goal by reusing the setup of our first experiment but randomly add impulse response values from $N(0, 10^{-4})$. Figure 3 shows an example of an impulse response together with the part that is estimated by the separable LMS algorithm. Figure 4 depicts the corresponding relative system mismatch. Due to the system mismatch only the separable part can be identified. The steady state gets stuck at around 13dB higher values than before, even a bit more than the theoretically expected 11dB from steady state analysis. The learning of this first part is, however, considerably faster than a classic LMS solution. A further improvement could be obtained by using the tensor structure for initial quick learning and then switching to a more general filter structure. Such combinations of filters with different learning speed have been proposed [24] in form of convex combiners.

Finally in our last example, we show a three-way tensor LMS algorithm for which we selected $\mathbf{a} = [0, ..., 0, 1, 0] \in \mathbb{R}^{32\times 1}, \mathbf{c}_k = 0.9^k; k = 0, 1, ..., 7$ and $\mathbf{b} \in \mathbb{R}^{4\times 1}$ with randomly selected values

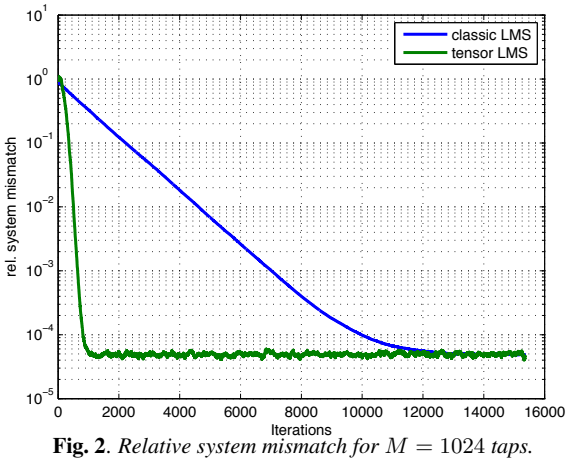

Fig. 2. *Relative system mismatch for* $M = 1024$ *taps.*

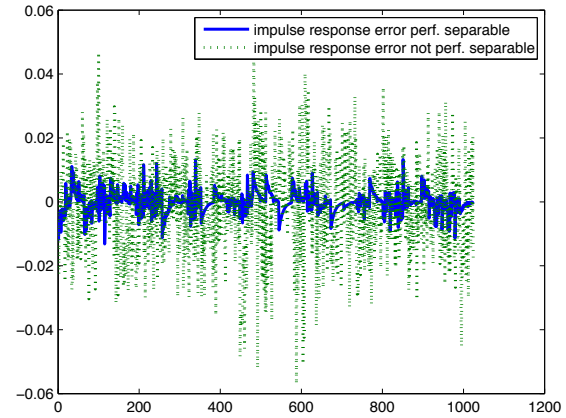

Fig. 3. *Typical impulse response (not perfectly separable) with* $M = 1024$ *taps.*

for each MC run. This setup nicely resembles a telephone hybrid causing electrical echoes $\mathbf{b}$ after a certain long-distance delay defined by the position of one in $\mathbf{a}$ and multiple roundtrip echoes. An example of the impulse response is shown in Figure 5.

We compare the performance also with the PNLMS algorithm [8, 9, 25] which typically comes with a complexity $3M$, while the tensor LMS algorithm runs in the order of $2(M_a + M_b + M_c)$. Due to space constraints the algorithm is not presented here but the procedure can be downloaded from our web page [23]. Figure 6 depicts the relative system mismatch obtained after 100MC runs.

5. CONCLUSIONS

We proposed a novel gradient type algorithm for tensors. Examples of rank one two- and three-way tensors were presented. The adaptive algorithm is of considerably less complexity than alternative algorithms and exhibits very rapid learning speed with comparable steady-state quality. Even for systems that do not match perfectly a rank one tensor, significant savings can be found. We presented the idea only for real-valued signals; extensions to the complex-valued case seem straight-forward.

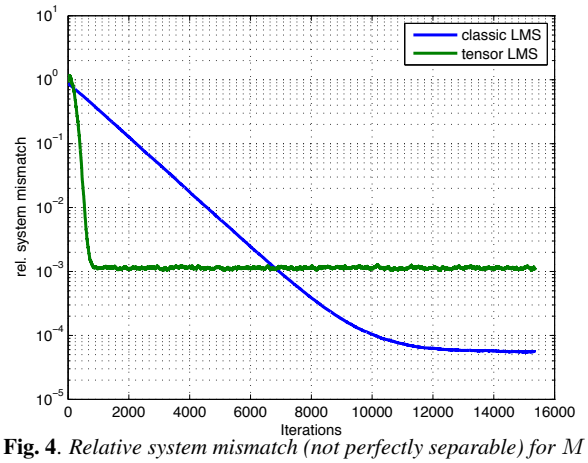

Fig. 4. *Relative system mismatch (not perfectly separable) for* $M = 1024$ *taps.*

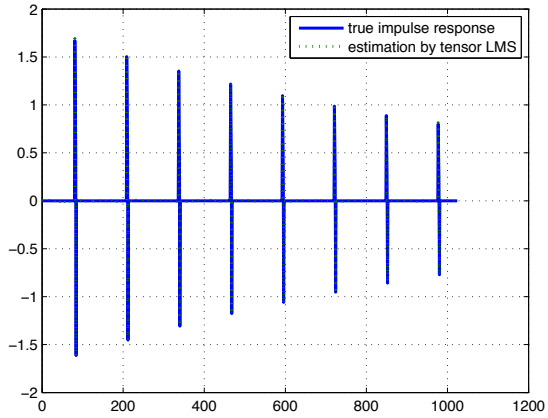

Fig. 5. *Impulse response described by a rank one three-way tensor.*

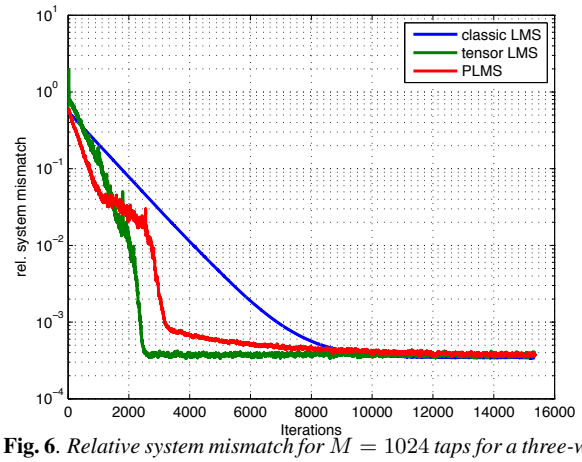

Fig. 6. *Relative system mismatch for* $M = 1024$ *taps for a three-way tensor.*

6. REFERENCES

[1] B. Widrow and M. Hoff Jr., "Adaptive switching circuits," in *IRE WESCON conv. Rec.*, vol. Part 4, 1960, pp. 96–104.

[2] S. Haykin, *Adaptive Filter Theory*. 4. edition, Prentice Hall, 2002.

[3] A. H. Sayed, *Fundamentals of Adaptive Filtering*. Hoboken, NJ, USA: John Wiley & Sons, Inc., 2003.

[4] M. Rupp, "The behavior of LMS and NLMS algorithms in the presence of spherically invariant processes," *IEEE Trans. Signal Processing*, vol. 41, no. 3, pp. 1149–1160, Mar. 1993.

[5] R. W. Harris, D. M. Chabries, and F. A. Bishop, "A variable step (VS) adaptive filter algorithm," *IEEE Transactions on Signal Processing*, vol. 34, pp. 309–316, Apr. 1986.

[6] R. H. Kwong and E. W. Johnston, "A variable step size LMS algorithm," *IEEE Transactions on Signal Processing*, vol. 40, no. 7, pp. 1633–1642, Jul. 1992.

[7] S. Makino, Y. Kaneda, and N. Koizumi, "Exponentially weighted step-size NLMS adaptive filter based on the statistics of a room impulse response," *IEEE Transactions on Speech and Audio Processing*, vol. 1, Jan. 1993.

[8] D. L. Duttweiler, "Proportionate normalized least mean square adaptation in echo cancellers," *IEEE Trans. Speech Audio Processing*, vol. 8, no. 5, pp. 508–518, Sept. 2000.

[9] J. Benesty and S. Gay, "An improved PNLMS algorithm," in *Proc. of the 27th International Conference on Acoustics, Speech, and Signal Processing (ICASSP'02)*, 2002.

[10] Y. Gu, J. Jin, and S. Mei, "l_0 norm constraint LMS algorithm for sparse system identification," *Signal Processing Letters, IEEE*, vol. 16, no. 9, pp. 774–777, Sept 2009.

[11] Y. Chen, Y. Gu, and A. Hero, "Sparse LMS for system identification," in *Proc. of the 34th International Conference on Acoustics, Speech, and Signal Processing (ICASSP'09)*, April 2009, pp. 3125–3128.

[12] P. Comon, "Tensors – a brief introduction," *IEEE Signal Processing Magazine*, pp. 44–53, May 2014.

[13] V. Mayer-Schoenberger and K. Cukier, *Big Data: A Revolution That Will Transform How We Live, Work and Think*. John Murray, 2013.

[14] ——, *Learning With Big Data: The Future of Education*. Houghton Miffling Harcourt Publishing Company, 2014.

[15] T. K. Moon and W. C. Stirling, *Mathematical Methods and Algorithms for Signal Processing*. Prentice Hall, 2000.

[16] C. van Loan and N. Pitsianis, "Approximation with Kronecker products," in *Linear Algebra for Large Scale and Real Time Applications*. Norwell, MA: Kluwer, 1993, p. 293–314.

[17] R. Flanagan and J.-J. Werner, "Cascade echo canceler arrangement," *U.S. Patent 6,009,083*, Dec. 28, 1999.

[18] D. Huang, X. Su, and A. Nallanathan, "Characterization of a cascade LMS predictor," in *Proc. of the 30th International Conference on Acoustics, Speech, and Signal Processing (ICASSP'05)*, vol. 3, Singapore, Singapore, Mar. 2005, pp. 173–176.

[19] E. Aschbacher and M. Rupp, "Robustness analysis of a gradient identification method for a nonlinear Wiener system," in *Proc. IEEE SSP 2005*, Bordeaux, France, Jul. 2005, pp. 103–108.

[20] R. Dallinger and M. Rupp, "On robustness of coupled adaptive filters," in *Proc. of the 34th International Conference on Acoustics, Speech, and Signal Processing (ICASSP'09)*, Taipei, Taiwan, Apr. 2009, pp. 3085–3088.

[21] ——, "Stability analysis of an adaptive Wiener structure," in *Proc. of the 35th International Conference on Acoustics, Speech, and Signal Processing (ICASSP'10)*, Dallas, TX, USA, Mar. 2010, pp. 3718–3721.

[22] ——, "On the robustness of LMS algorithms with time-variant diagonal matrix step-size," in *Proc. of the 38th International Conference on Acoustics, Speech, and Signal Processing (ICASSP'13)*, May 2013.

[23] [Online]. Available: https://www.nt.tuwien.ac.at/downloads/featured-downloads

[24] M. T. M. Silva and V. H. Nascimento, "Improving the tracking capability of adaptive filters via convex combination," *IEEE Transactions on Signal Processing*, vol. 56, no. 7, pp. 3137–3149, Jul. 2008.

[25] J. Benesty and Y. A. Huang, "The LMS, PNLMS, and exponentiated gradient algorithms," in *In Proceedings of EUSIPCO*, 2007.