Separable Least-Mean Squares Beamforming

Lucas N. Ribeiro, Bruno Sokal, André L. F. de Almeida, João César M. Mota
Wireless Telecommunications Research Group (GTEL)
Universidade Federal do Ceará (UFC)
Fortaleza, Brazil
{nogueira, brunosokal, andre, mota}@gtel.ufc.br

***Abstract*—Large-scale antenna systems have attracted much research efforts due to its application to modern mobile communications systems. Designing such arrays, however, can be challenging due to the computational efforts of classical signal processing methods. To tackle this issue, we investigate Kronecker-separable adaptive filtering methods. Simulation results suggest that this approach exhibits better convergence properties in some scenarios compared to the classical adaptive filtering benchmark.**

***Index Terms*—Beamforming, Adaptive Filtering, Tensor Product**

I. Introduction

Large-scale (massive) sensor arrays systems have become popular among different applications such as wireless communications [1] and biomedical signal processing [2], for example. Such large arrays bring desirable features such as increased spatial resolution, large array gain, deep attenuation outside passband, among others. Nevertheless, a price must be paid to benefit from these attractive properties. Massive array systems are usually expensive in many senses: signal processing becomes involved, more efficient hardware implementations are necessary to achieve the promised gains, and higher energy consumption is observed in some applications [3]. In the present contribution, we are interested in addressing the large computational cost of beamforming in massive antenna arrays. To this end, we exploit the separability property of uniform rectangular arrays (URA) and we put forward the design of separable adaptive beamforming filters.

Separable filters have been investigated as a relatively inexpensive implementation of large filters. In [4], we have introduced a separable beamformer that exploits the multi-dimensionality of volumetric arrays. Simulation results show that the proposed filter, which is based on an alternating minimization strategy, exhibits modest computational complexity reduction compared to the classical Wiener beamformer [5]. In the follow-up work [6], we obtain analytical expressions for the separable filter of [4], which is based on sample estimates, and conduct an asymptotic performance analysis. In [7], the authors propose a separable least mean squares (LMS) algorithm to identify a second-order separable system, i.e., a linear and time-invariant system whose impulse response $\mathbf{h}$ can be well approximated as a Kronecker product $\mathbf{h} \approx \mathbf{a} \otimes \mathbf{b}$. In [8], we study the identification of third-order separable systems by comparing the LMS strategy of [7] to that of [4]. We observe that our approach yields better system identification accuracy. Other classical signal processing algorithms such as the generalized sidelobe canceler and the minimum variance distortionless beamformers have been implemented by separable filters in [9] and [10], respectively. The works previously mentioned reveal that separable filters can drastically reduce the computational costs with small performance degradation at massive filtering problems.

This present work is a follow-up of our contributions [4], [6] and should be regarded as a proof-of-concept paper, in which we sketch the potential of separable adaptive filters to the large-scale array beamforming problem. In [4], [6], we have employed *deterministic gradients* in the calculation of the separable beamformers. We now adopt the *stochastic gradient* technique, which leads to the LMS solution. More specifically, we investigate the performance of the tensor LMS (TLMS) algorithm of [7] and we propose a different implementation named alternating tensor LMS (ATLMS). Simulation results show that the separable beamformers enjoy from faster convergence compared to the conventional (non-separable) normalized LMS (NLMS) benchmark. Our simulation results also reveal some properties of the proposed ATLMS method.

We introduce our system model in Section II, present the separable beamforming methods in Section III, show and discuss the numerical results in Section IV and conclude the work in Section V.

A. Notation

The transposed and the conjugated transposed (Hermitian) of $\mathbf{X}$ are denoted by $\mathbf{X}^{\mathsf{T}}$ and $\mathbf{X}^{\mathsf{H}}$, respectively. The $(M \times M)$-dimensional identity matrix is represented by $\mathbf{I}_M$. The absolute value, the ℓ_2 norm, and the expected value operator are respectively represented by $|\cdot|$, $\|\cdot\|_2$, and $\mathbb{E}[\cdot]$. The Kronecker product is denoted by $\otimes$, $O(\cdot)$ represents the Big-O notation, and $\lfloor \cdot \rfloor$ the floor operator.

II. System Model

Let us consider a receiver system equipped with a URA of N omni-directional antennas, with N_h columns and N_v rows along the yz plane, as depicted in Figure 1. The antenna array is tuned to operate at wavelength λ, and the inter-element spacing in both horizontal and vertical directions is $\lambda/2$.

This work is supported by the National Council for Scientific and Technological Development CNPq, CAPES/PROBRAL Proc. no. 88887.144009/2017-00, and FUNCAP.

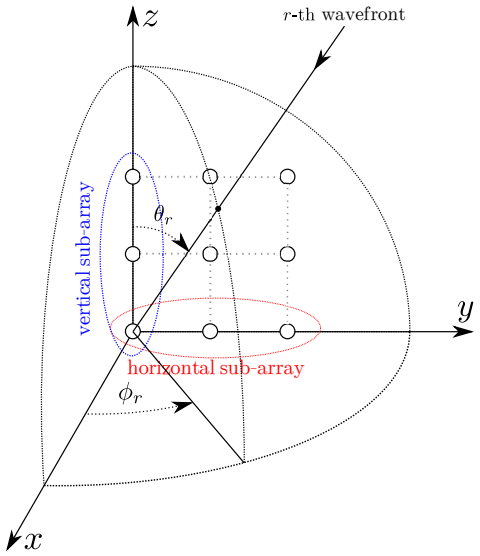

Fig. 1. Uniform rectangular array (URA) with $N = 3 \times 3$ elements.

We assume that R independent narrow-band signals in far-field propagation illuminate the URA. The sampled complex-envelope of the r-th signal is denoted as $s_r[k]$ and it impinges on the array from the azimuth ϕ_r and polar θ_r angles. The signals are assumed to be mutually uncorrelated with zero mean and unit variance, i.e.,

$$\mathbb{E}\left[s_i[k]s_j^*[k]\right] = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} .$$

Let $x_n[k]$ represent received signal at antenna n and instant k, $p_r = \sin\phi_r \sin\theta_r$ and $q_r = \cos\theta_r$ the horizontal and vertical direction cosines [5], respectively, then the received signal vector $\mathbf{x}[k] = [x_1[k], \ldots, x_N[k]]^\mathsf{T}$ can be written as

$$\mathbf{x}[k] = \sum_{r=1}^{R} \mathbf{a}(p_r, q_r)s_r[k] + \mathbf{b}[k] = \mathbf{A}\mathbf{s}[k] + \mathbf{b}[k], \quad (1)$$

where $\mathbf{a}(p_r, q_r) = [a_1(p_r, q_r), \ldots, a_N(p_r, q_r)]^\mathsf{T}$ denotes the array steering vector, $\mathbf{A} = [\mathbf{a}(p_1, q_1), \ldots, \mathbf{a}(p_R, q_R)] \in \mathbb{C}^{N\times R}$ the array manifold matrix, $\mathbf{s}[k] = [s_1[k], \ldots, s_R[k]]^\mathsf{T}$ the impinging signals vector with autocorrelation matrix $\mathbf{R}_{ss} = \mathbb{E}\left[\mathbf{s}[k]\mathbf{s}^\mathsf{H}[k]\right] = \mathbf{I}_{N_s}$, and $\mathbf{b}[k] \in \mathbb{C}^N$ the complex additive white Gaussian noise vector with zero mean and variance σ_b^2. From our URA assumptions, it follows that

$$a_n(\phi_r, \theta_r) = e^{j\pi[(n_h-1)p_r + (n_v-1)q_r]}, \quad (2)$$

with $n = n_h + (n_v - 1)N_h$, $n_h \in \{1, \ldots, N_h\}$, and $n_v \in \{1, \ldots, N_v\}$.

It is well-known that URA array steering vectors can be Kronecker-separated into a horizontal and a vertical factors as a result of array bi-dimensionality [5]. One can factorize (2) as

$$a_n(p_r, q_r) = a_{n_h}^{(h)}(p_r)a_{n_v}^{(v)}(q_r), \quad (3)$$

where $a_{n_h}^{(h)}(p_r) = e^{j\pi(n_h-1)p_r}$ and $a_{n_v}^{(v)}(q_r) = e^{j\pi(n_v-1)q_r}$. From (3) and defining $\mathbf{a}_h(p_r) = \left[a_1^{(h)}(p_r), \ldots, a_{N_h}^{(h)}(p_r)\right]^\mathsf{T}$, $\mathbf{a}_v(q_r) = \left[a_1^{(v)}(q_r), \ldots, a_{N_v}^{(v)}(q_r)\right]^\mathsf{T}$, the steering vectors can finally be factorized as $\mathbf{a}(p_r, q_r) = \mathbf{a}_v(q_r) \otimes \mathbf{a}_h(p_r)$. In the next section, we present adaptive beamforming filters that exploit this algebraic structure to enhance its performance.

III. Beamforming Methods

In this work, we are interested in recovering a desired source, hereafter referred to as $s_d[k]$, from the interfering ones by employing a large antenna array. To this end, we make use of a spatial filter (beamformer) and optimize it according to the the minimum mean square error (MMSE) criterion, i.e., we seek the beamforming filter $\mathbf{w} \in \mathbb{C}^N$ which minimizes the mean square error (MSE) function

$$\begin{aligned} J_{\text{MSE}}(\mathbf{w}) &= \mathbb{E}\left[|s_d[k] - \mathbf{w}^\mathsf{H}\mathbf{x}[k]|^2\right] \\ &= \sigma_s^2 - \mathbf{p}_{xs}^\mathsf{H}\mathbf{w} - \mathbf{w}^\mathsf{H}\mathbf{p}_{xs} + \mathbf{w}^\mathsf{H}\mathbf{R}_{xx}\mathbf{w}, \end{aligned} \quad (4)$$

where $\mathbf{R}_{xx} = \mathbb{E}\left[\mathbf{x}[k]\mathbf{x}^\mathsf{H}[k]\right] = \mathbf{A}\mathbf{A}^\mathsf{H} + \sigma_n^2\mathbf{I}_N$ denotes the received signal autocorrelation matrix, $\mathbf{p}_{xs} = \mathbb{E}\left[\mathbf{x}[k]s_d^*[k]\right] = \mathbf{A}\mathbf{e}_d$ denotes the cross-covariance vector, and $\mathbf{e}_r \in \mathbb{C}^R$ the r-th canonical vector in the R-dimensional space. It is well understood that the Wiener filter $\mathbf{w}_{\text{opt}} = \mathbf{R}_{xx}^{-1}\mathbf{p}_{xs}$ yields the global minimum of (4). Its calculation, however, requires knowledge on the array manifold matrix $\mathbf{A}$, which may be expensive to acquire in practice. Adaptive filtering is a classical and computationally inexpensive alternative to the Wiener filter. The LMS algorithm and its normalized flavor are the workhorse solutions for minimizing (4). Nevertheless, they face slow convergence as the filter length grows [11].

To address the computational complexity issue and to exploit the URA separability, we have put forward separable extensions of the classical Wiener filter in [6]. Specifically, we proposed optimizing a separable beamforming filter $\mathbf{w} = \mathbf{w}_v \otimes \mathbf{w}_h$, with $\mathbf{w}_h \in \mathbb{C}^{N_h}$ and $\mathbf{w}_v \in \mathbb{C}^{N_v}$ by calculating *deterministic gradients* of (4). Here, by contrast, we present two adaptive solutions which employ the stochastic gradient technique to optimize the separable filter. The first solution, referred to as TLMS, was first introduced in [7], [12] by Rupp and Schwarz to tackle the slow convergence issue of the NLMS algorithm. The second solution, hereafter called ATLMS consists of a different implementation of the TLMS algorithm inspired by the alternating minimization strategy of [4], [6], [8]. In the remainder of this section, we present the TLMS and ATLMS algorithms and briefly comment on their stability, convergence and computational complexity properties.

A. Tensor LMS Algorithm

We begin by noting that the output signal $y[k] = \mathbf{w}^\mathsf{H}\mathbf{x}[k]$ of the separable filter $\mathbf{w} = \mathbf{w}_v \otimes \mathbf{w}_h$ can be written in two equivalent manners [4], [7]:

$$y[k] = \mathbf{w}_h^\mathsf{H}\mathbf{X}[k]\mathbf{w}_v^* = \mathbf{w}_h^\mathsf{H}\mathbf{u}_h[k], \quad (5)$$

$$= \mathbf{w}_v^\mathsf{H}\mathbf{X}^\mathsf{T}[k]\mathbf{w}_h^* = \mathbf{w}_v^\mathsf{H}\mathbf{u}_v[k], \quad (6)$$

where $\mathbf{X}[k]$ is a $(N_h \times N_v)$-dimensional matrix obtained by reshaping $\mathbf{x}[k]$, and $\mathbf{u}_h[k]$ and $\mathbf{u}_v[k]$ are referred to as the horizontal and vertical sub-array received signals [4], [6].

The NLMS algorithm can be derived with respect to the horizontal and vertical sub-arrays. At instant (iteration) k, we have:

$$e[k] = s_d[k] - (\mathbf{w}_v[k] \otimes \mathbf{w}_h[k])^{\mathsf{H}}\mathbf{x}[k], \tag{7}$$

$$\mathbf{w}_h[k+1] = \mathbf{w}_h[k] + \tilde{\mu}[k]\mathbf{u}_h[k]e^*[k], \tag{8}$$

$$\mathbf{w}_v[k+1] = \mathbf{w}_v[k] + \tilde{\mu}[k]\mathbf{u}_v[k]e^*[k], \tag{9}$$

where the step size $\tilde{\mu}[k]$ is normalized as [7]

$$\tilde{\mu}[k] = \frac{\mu}{\|\mathbf{u}_h[k]\|_2^2 + \|\mathbf{u}_v[k]\|_2^2}. \tag{10}$$

In the original TLMS paper [7], (i) $\mathbf{u}_h[k]$ and $\mathbf{u}_v[k]$ are computed, (ii) the instantaneous error (7) is calculated, then (iii) the sub-filters are updated following (8) and (9). An extra step can be considered to check convergence. TLMS is summarized in Algorithm 1 considering a standard filter initialization and K available samples.

Algorithm 1 Tensor LMS algorithm

Require: Step parameter μ, sample size K

$k \leftarrow 1$
Initialize $\mathbf{w}_h[k]$ and $\mathbf{w}_v[k]$ as $[1, 0, \ldots, 0]^{\mathsf{T}}$
for $k = 1 : K$ **do** ▷ Note we use MATLAB's notation
  $\mathbf{u}_h[k] \leftarrow \mathbf{X}[k]\mathbf{w}_v^*[k]$
  $\mathbf{u}_v[k] \leftarrow \mathbf{X}[k]^{\mathsf{T}}\mathbf{w}_h^*[k]$
  $e[k] \leftarrow s_d[k] - (\mathbf{w}_v[k] \otimes \mathbf{w}_h[k])^{\mathsf{H}}\mathbf{x}[k]$
  $\tilde{\mu}[k] \leftarrow \frac{\mu}{\|\mathbf{u}_h[k]\|_2^2 + \|\mathbf{u}_v[k]\|_2^2}$
  $\mathbf{w}_h[k+1] \leftarrow \mathbf{w}_h[k] + \tilde{\mu}[k]\mathbf{u}_h[k]e^*[k]$
  $\mathbf{w}_v[k+1] \leftarrow \mathbf{w}_v[k] + \tilde{\mu}[k]\mathbf{u}_v[k]e^*[k]$
  Check convergence
end for
return $\mathbf{w}_v[k+1] \otimes \mathbf{w}_h[k+1]$

B. Alternating Tensor LMS Algorithm

Note, however, that one can update $\mathbf{w}_h[k]$ and $\mathbf{w}_v[k]$ in different forms. In the ATLMS algorithm, $\mathbf{w}_h[k]$ and $\mathbf{w}_v[k]$ are computed in the alternating fashion of [6]. More specifically, for a fixed $\mathbf{w}_v[k]$, we (i) compute $\mathbf{u}_h[k]$, (ii) the instantaneous error (7), then (iii) the horizontal sub-filter is updated following (8). Steps (i)-(iii) are repeated over K_h iterations. Next, for a now fixed $\mathbf{w}_h[k]$, we (iv) compute $\mathbf{u}_v[k]$, (v) the instantaneous error (7), then (vi) the vertical sub-filter is updated as in (9). Steps (iv)-(vi) are repeated over K_v iterations. Since each sub-filter is updated according to the classical NLMS fashion, the step size parameters set as

$$\tilde{\mu}_h[k] = \frac{\mu}{\|\mathbf{u}_h[k]\|_2^2}, \quad \tilde{\mu}_v[k] = \frac{\mu}{\|\mathbf{u}_v[k]\|_2^2}. \tag{11}$$

ATLMS is summarized in Algorithm 2 considering a standard filter initialization and K available samples, which are divided in blocks of $K_b = \lfloor \frac{K}{K_h+K_v} \rfloor$ samples.

The comparison of Algorithms 1 and 2 gives some insight on their behavior and differences. In the former, both sub-filters are updated at each iteration, whereas, in the latter, the solution iterates through the horizontal and vertical directions in an alternate fashion. This strategy is essentially a block coordinate descent [13]. Although these algorithms are algebraically similar, it is unclear if they yield the same performance in terms of convergence rate and source recovery. The simulations results presented in Section IV shed a light on this question.

Algorithm 2 Alternating Tensor LMS algorithm

Require: Step parameter μ, sample parameters K, K_h, K_v

$k \leftarrow 1$
$K_b \leftarrow \lfloor \frac{K}{K_h+K_v} \rfloor$
Initialize $\mathbf{w}_h[k]$ and $\mathbf{w}_v[k]$ as $[1, 0, \ldots, 0]^{\mathsf{T}}$
for $k = 1 : K_h + K_v : K_b(K_h + K_v)$ **do**
  for $k_h = k : k + K_h - 1$ **do**
    $\mathbf{u}_h[k_h] \leftarrow \mathbf{X}[k_h]\mathbf{w}_v^*[k_h]$
    $e[k_h] \leftarrow s_d[k_h] - (\mathbf{w}_v[k_h] \otimes \mathbf{w}_h[k_h])^{\mathsf{H}}\mathbf{x}[k_h]$
    $\tilde{\mu}_h[k_h] \leftarrow \frac{\mu}{\|\mathbf{u}_h[k_h]\|_2^2}$
    $\mathbf{w}_h[k_h+1] \leftarrow \mathbf{w}_h[k_h] + \tilde{\mu}_h[k_h]\mathbf{u}_h[k_h]e^*[k_h]$
  end for
  for $k_v = k + K_h : k + K_h + K_v - 1$ **do**
    $\mathbf{u}_v[k_v] \leftarrow \mathbf{X}[k_v]^{\mathsf{T}}\mathbf{w}_h[k_v]^*$
    $e[k_v] \leftarrow s_d[k_v] - (\mathbf{w}_v[k_v] \otimes \mathbf{w}_h[k_h+1])^{\mathsf{H}}\mathbf{x}[k_v]$
    $\tilde{\mu}_v[k_v] \leftarrow \frac{\mu}{\|\mathbf{u}_v[k_v]\|_2^2}$
    $\mathbf{w}_v[k_v+1] \leftarrow \mathbf{w}_v[k_v] + \tilde{\mu}_v[k_v]\mathbf{u}_v[k_v]e^*[k_v]$
  end for
  Check convergence
end for
return $\mathbf{w}_v[k_v+1] \otimes \mathbf{w}_h[k_h+1]$

C. Convergence and Computational Complexity

As already mentioned in [7], the theoretical analysis of separable adaptive filters is challenging due to their cascaded nature. It can be shown that TLMS converges in the MSE as long as [7]

$$0 < \mu < \frac{2}{\|\mathbf{u}_h[k]\|_2^2 + \|\mathbf{u}_v[k]\|_2^2}.$$

Since each ATLMS block-descent consists of a standard and independent NLMS algorithm, we choose the step size factors to satisfy $0 < \mu < \frac{2}{\|\mathbf{u}_h[k]\|_2^2}$ and $0 < \mu < \frac{2}{\|\mathbf{u}_v[k]\|_2^2}$. In our experiments, we consider that the adaptive algorithm has converged when $\|\mathbf{w}[k+1] - \mathbf{w}[k]\|_2^2 \leq \epsilon$, where $\epsilon > 0$ is a small tolerance.

Regarding computational complexity, TLMS, and ATLMS perform $O(N_h + N_v)$ multiplications while NLMS $O(N)$. The convergence rate is the most important aspect in this sense, since both strategies are linear with the filter length. In the next section, we plot the learning curve of these algorithms to investigate their convergence properties.

IV. Simulation Results

In this section, we present simulation results to investigate the performance of the separable beamformers using NLMS as benchmark. We set as figure of merit the sample MSE, defined as

$$\text{MSE}(\mathbf{w}) = \frac{1}{K}\sum_{k=0}^{K-1} \left| s_d[k] - \mathbf{w}^{\mathsf{H}}\mathbf{x}[k] \right|^2,$$

and calculated over $K = 10^4$ samples. We consider a URA of $N = N_h \times N_v = 8 \times 8$ antennas, which is a reasonable setup for a 5G base-station [14], and $R = 4$ QPSK-modulated impinging signals. The results are obtained by means of Monte Carlo simulations with 10^4 independent experiments, wherein the directional cosines p_r and q_r are uniformly selected from the interval $[-0.9, 0.9]$ and the convergence tolerance is set to $\epsilon = 10^{-9}$. As the QPSK signals have unit power, we define the signal to noise ratio (SNR) as SNR $= 1/\sigma_b^2$. Since massive MIMO systems will likely operate in the low SNR regime, we set SNR $= 0\,\mathrm{dB}$ in our simulations.

We first study the learning curve of ATLMS for different iteration intervals K_h and K_v and step size factors in Figures 2 and 3. In the former figure, the learning curves are obtained for $\mu = 0.5$. In this scenario, we observe that $K_h = K_v = 100$ yields the fastest convergence. In the latter figure, where the learning curves were calculated for $\mu = 0.1$, we note that $K_h = K_v = 10$ presents the best convergence rate. Interestingly, ATLMS converges to the same MSE for all K_h and K_v tuples, approximately $-15\,\mathrm{dB}$ for $\mu = 0.5$ and $-16.5\,\mathrm{dB}$ for $\mu = 0.1$. Therefore, we conclude that the iteration intervals K_h and K_v affect mostly the algorithm convergence rate. If the these parameters are chosen too large, then the convergence rate only worsens. If they are relatively small (up to 100 iterations), then they can be optimized for a given step size parameter. However, Figures 2 and 3 show that the convergence performance difference between $K_h = K_v = 1$, 10, and 100 is not much significant. In the following simulations, we choose $K_h = K_v = 10$, as it provides the fastest convergence for $\mu = 0.1$.

Now we investigate the convergence rate of the adaptive beamformers for different step size μ. In Figure 4, we observe the well-known behavior of the NLMS algorithm: its convergence rate sharply decreases as μ increases. Note that it becomes especially slow for $\mu = 0.1$ and 0.05. In a practical setup, the system designer typically seeks a step size that strikes a good compromise between convergence rate and MSE. Figure 4 reveals that $\mu = 0.5$ achieves this trade-off for the given parameters. The separable beamformers, by contrast, converge much faster for all considered step sizes, as one can see in Figures 5 and 6. We observe a striking result for $\mu = 0.05$: the separable beamformers achieve convergence in about 5000 iterations, while NLMS converge in more than 10000 iterations. These results suggest that the separable beamformers are more efficient in computational terms, as they converge faster and each filter update requires a number of multiplications still linear with the array size.

Although TLMS and ATLMS seem to yield the same performance, one can find some differences. For the given parameters, we see by comparing Figure 5 to 6 that ATLMS has, at convergence, a larger misadjustment and attains a slightly worse MSE level. Moreover, the convergence rate of ATLMS is slightly poorer. Simply put, TLMS performs better than ATLMS for the considered scenario. A natural question rises: is this true in general? Simulations considering different scenarios would be necessary to answer, and, such extensive analysis is out of the scope of this paper. It is important to mention that although ATLMS does not exhibit any significant

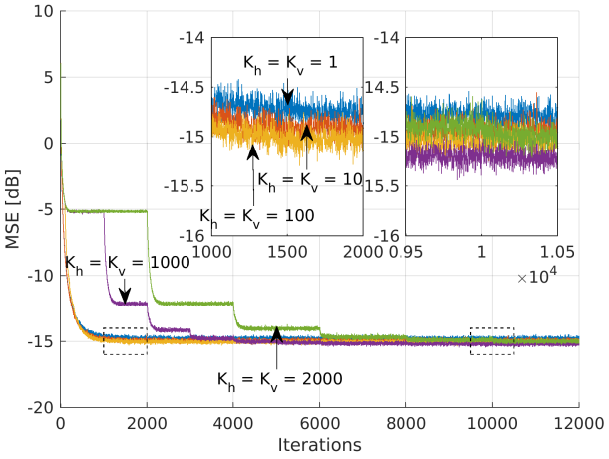

Fig. 2. ATLMS learning curves for different iteration intervals K_h and K_v. $\mu = 0.5$.

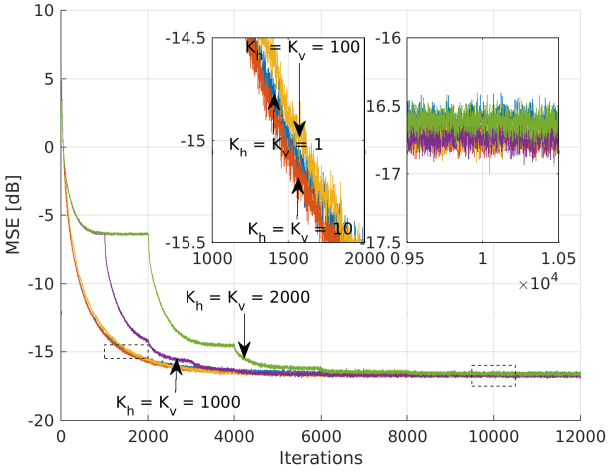

Fig. 3. ATLMS learning curves for different iteration intervals K_h and K_v. $\mu = 0.1$.

improvement over TLMS, there are practical scenarios where this alternating strategy seems promising as it gives more flexibility in the filter optimization, e.g., the filter designer can choose to favor a sub-array over the other by manipulating K_h and K_v, which is impossible with TLMS.

V. Conclusion

Simulations results have shown that the separable beamformers converge faster than the traditional NLMS beamformer. For the considered scenarios and parameters set, ATLMS does not present any significant improvement compared to TLMS. Nonetheless, ATLMS offers more design flexibility, which can be useful to some practical applications in wireless communications systems. Future work include theoretical performance analysis and application to realistic mobile communication scenarios with massive multi-antenna systems.

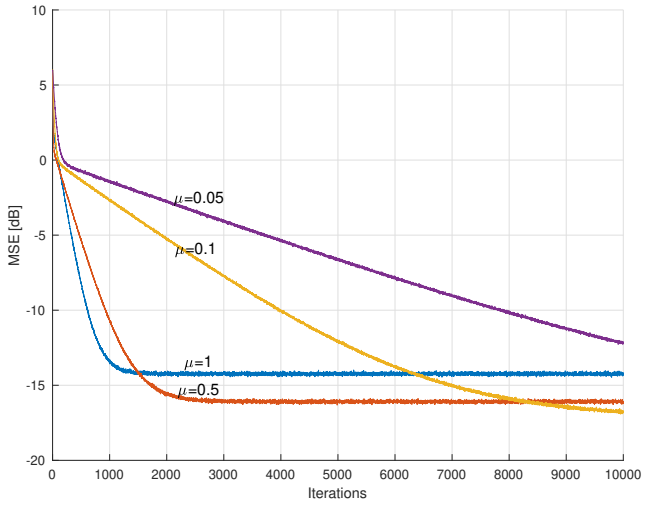

Fig. 4. NLMS learning curves for different step size μ.

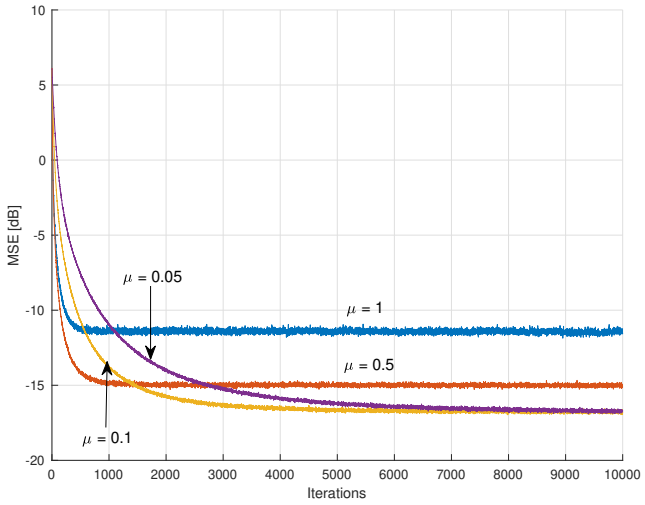

Fig. 6. ATLMS learning curves for different step size μ. $K_h = K_v = 10$.

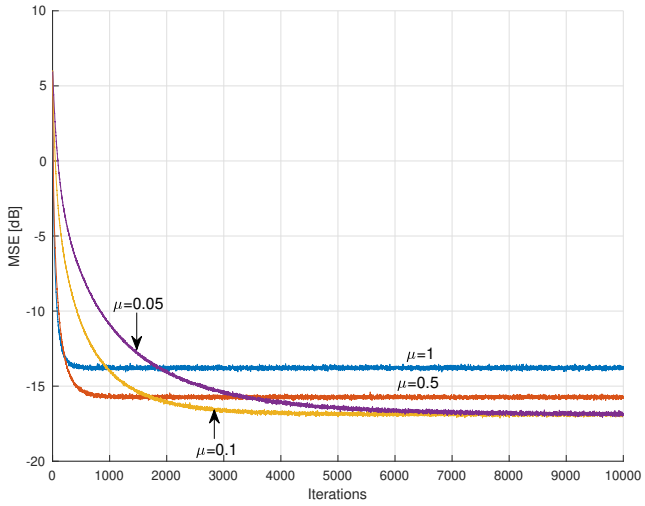

Fig. 5. TLMS learning curves for different step size μ.

REFERENCES

[1] S. Schwarz and M. Rupp, "Society in motion: challenges for LTE and beyond mobile communications," *IEEE Communications Magazine*, vol. 54, no. 5, pp. 76–83, May 2016.

[2] M. Seeck, L. Koessler, T. Bast, F. Leijten, C. Michel, C. Baumgartner, B. He, and S. Beniczky, "The standardized EEG electrode array of the IFCN," *Clinical Neurophysiology*, vol. 128, no. 10, pp. 2070 – 2077, 2017.

[3] L. N. Ribeiro, S. Schwarz, M. Rupp, and A. L. F. de Almeida, "Energy efficiency of mmwave massive mimo precoding with low-resolution dacs," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 2, pp. 1–1, 2018.

[4] L. N. Ribeiro, A. L. F. de Almeida, and J. C. M. Mota, "Tensor beamforming for multilinear translation invariant arrays," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2016, pp. 2966–2970.

[5] H. L. Van Trees, *Optimum array processing: Part IV of detection, estimation and modulation theory*. Wiley Online Library, 2002, vol. 1.

[6] L. N. Ribeiro, A. L. de Almeida, J. A. Nossek, and J. C. M. Mota, "Low-complexity separable beamformers for massive antenna array systems," *arXiv preprint arXiv:1805.00176*, 2018.

[7] M. Rupp and S. Schwarz, "A tensor LMS algorithm," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 3347–3351.

[8] L. N. Ribeiro, A. L. F. de Almeida, and J. C. M. Mota, "Identification of separable systems using trilinear filtering," in *2015 IEEE 6th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, Dec 2015, pp. 189–192.

[9] R. K. Miranda, J. P. C. da Costa, F. Roemer, A. L. de Almeida, and G. Del Galdo, "Generalized sidelobe cancellers for multidimensional separable arrays," in *2015 IEEE 6th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pp. 193–196.

[10] L. Liu, J. Xie, L. Wang, Z. Zhang, and Y. Zhu, "Robust tensor beamforming for polarization sensitive arrays," *Multidimensional Systems and Signal Processing*, pp. 1–22, 2018.

[11] V. H. Nascimento, "Improving the initial convergence of adaptive filters: variable-length LMS algorithms," in *2002 14th International Conference on Digital Signal Processing*, vol. 2. IEEE, 2002, pp. 667–670.

[12] M. Rupp and S. Schwarz, "Gradient-based approaches to learn tensor products," in *2015 23rd European Signal Processing Conference (EUSIPCO)*, pp. 2486–2490.

[13] M. Hong, M. Razaviyayn, Z.-Q. Luo, and J.-S. Pang, "A unified algorithmic framework for block-structured optimization involving big data: With applications in machine learning and signal processing," *IEEE Signal Processing Magazine*, vol. 33, no. 1, pp. 57–77, 2016.

[14] T. L. Marzetta, E. G. Larsson, H. Yang, and H. Q. Ngo, *Fundamentals of Massive MIMO*. Cambridge University Press, 2016.