Separable Least-Mean Squares Beamforming

Kenneth B. dos A. Benício

Department of Teleinformatics Engineering
Federal University of Ceará

Universidade Federal do Ceará

Fortaleza, 2021

Outline

Problem Statement

Objectives

1. Recover a desired source signal by employing a large antenna array following an Uniform Rectangular Array (URA).
2. Use spatial filter (beamforming) and optimize it according to the Mean square error (MSE) criterion.
3. Solve the problem of slow convergence presented at LMS and NLMS algorithms.

How to do so?

1. Exploiting URA separability.
2. Beamforming filter of the form $\mathbf{w} = \mathbf{w}_v \otimes \mathbf{w}_h$.

Classic Filter Problems I

The Classic Wiener Filter

- The MSE minimization function

$$J_{\mathrm{MSE}}(\boldsymbol{w}) = \mathbb{E}\{(s_d[k] - \boldsymbol{w}^{\mathrm{H}}\boldsymbol{x}[k])^2\} = 0, \tag{1}$$

- The optimal wiener solution

$$\boldsymbol{w}_{\mathrm{opt}} = \boldsymbol{R}_x^{-1}\boldsymbol{p}_{xs}, \tag{2}$$

- Problems with wiener filter

Classic Filter Problems II

Stocasthic Gradient Filters

- Received signal

$$y[k] = \boldsymbol{w}^{\mathrm{H}}\boldsymbol{x}[k], \tag{3}$$

- LMS Adaptative Filter

$$\boldsymbol{w}[k+1] = \boldsymbol{w}[k] + 2\mu\boldsymbol{x}[k]e^{*}[k] \tag{4}$$

- NLMS Adaptative Filter

$$\boldsymbol{w}[k+1] = \boldsymbol{w}[k] + \frac{\mu}{\gamma + \boldsymbol{x}^{\mathrm{T}}[k]\boldsymbol{x}[k]}\boldsymbol{x}[k]e^{*}[k] \tag{5}$$

- Advantages of Adaptative Filtering

Classic Filter Problems III

The Matrix Kronecker Product

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{1,1}\mathbf{B} & a_{1,2}\mathbf{B} & \cdots & a_{1,J}\mathbf{B} \\ a_{2,1}\mathbf{B} & a_{2,2}\mathbf{B} & \cdots & a_{2,J}\mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{I,1}\mathbf{B} & a_{I,2}\mathbf{B} & \cdots & a_{I,J}\mathbf{B} \end{bmatrix} \in \mathbb{C}^{RI \times JJ}. \tag{6}$$

System Model I

- The received signal model follows a geometric channel

$$\boldsymbol{x}[k] = \sum_{r=1}^{R} \boldsymbol{a}(p_r, q_r) s_r[k] + \boldsymbol{b}[k] = \boldsymbol{A}\boldsymbol{s}[k] + \boldsymbol{b}[k], \tag{7}$$

- The vector $\boldsymbol{a}(p_r, q_r)$ represents an Uniform Rectangular Array (URA)

$$\boldsymbol{a}(p_r, q_r) = \boldsymbol{a}_v(q_r) \otimes \boldsymbol{a}_h(p_r) \rightarrow a_n(p_r, q_r) = a_{n_h}^{(h)}(p_r) a_{n_v}^{(v)}(q_r), \tag{8}$$

$$a_{n_h}^{(h)}(p_r) = e^{j\pi(n_h - 1)} p_r, \tag{9}$$

$$a_{n_v}^{(v)}(q_r) = e^{j\pi(n_v - 1)} q_r, \tag{10}$$

System Model II

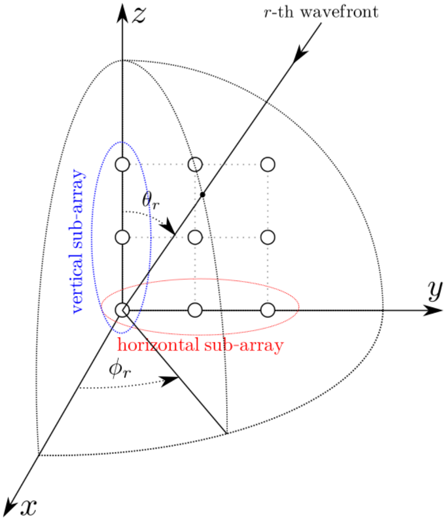

Figure Unifor Rectangular Array (URA) with 3 × 3 elements from [1].

TLMS and ATLMS I

- Filter Problem

$$\mathbb{E}\{(s_d[k] - \boldsymbol{w}^{\mathrm{H}}\boldsymbol{x}[k])^2\} = 0 \tag{11}$$

- Tensor Filters

$$y[k] = (\boldsymbol{w}_v \otimes \boldsymbol{w}_h)^{\mathrm{H}} \boldsymbol{x}[k] \tag{12}$$

- Tensor LMS

$$\boldsymbol{w}_h[k+1] = \boldsymbol{w}_h[k] + \mu[k]\boldsymbol{u}_h[k]e^*[k], \tag{13}$$
$$\boldsymbol{w}_v[k+1] = \boldsymbol{w}_v[k] + \mu[k]\boldsymbol{u}_v[k]e^*[k], \tag{14}$$

- Alternating Tensor LMS

TLMS and ATLMS II

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** $\triangleright$ 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 $\mathbf{w}_v[k+1] \otimes \mathbf{w}_h[k+1]$

Figure TLMS algorithm from [1].

TLMS and ATLMS III

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 $\mathbf{w}_v[k_v + 1] \otimes \mathbf{w}_h[k_h + 1]$

Figure ATLMS algorithm from [1].

TLMS and ATLMS IV

Convergence and Computational Complexity

- The convergence for TLMS in MSE is

$$0 < \mu < \frac{2}{||\boldsymbol{u}_h[k]||_2^2 + ||\boldsymbol{u}_v[k]||_2^2} \tag{15}$$

- The convergence for ATLMS in MSE is

$$0 < \mu < \frac{2}{||\boldsymbol{u}_i[k]||_2^2}, i \in \{h, v\}, \tag{16}$$

- TLMS and ATLMS has a computational complexity of $O(N_h + N_v)$ and NLMS of $O(N)$. Since all the methods are linear in complexity the most important aspect that we must observe is the convergence rate.

Simulation Scenario

Parameters

- It was considered an URA of 4×4 antennas with $R = 4$ multipaths and QPSK information signals.
- The SNR was defined as $\text{SNR} = 1/\sigma_b^2$.
- We set as figure of merit the sample Mean Square Error (MSE) defined and calculated over $K = 10000$ samples

$$\text{MSE}(\boldsymbol{w}) = \frac{1}{K} \sum_{k=1}^{K} \left|\left| s_d[k] - \boldsymbol{w}^{\text{H}} \boldsymbol{x}[k] \right|\right|^2 , \tag{17}$$

NLMS MSE Curve

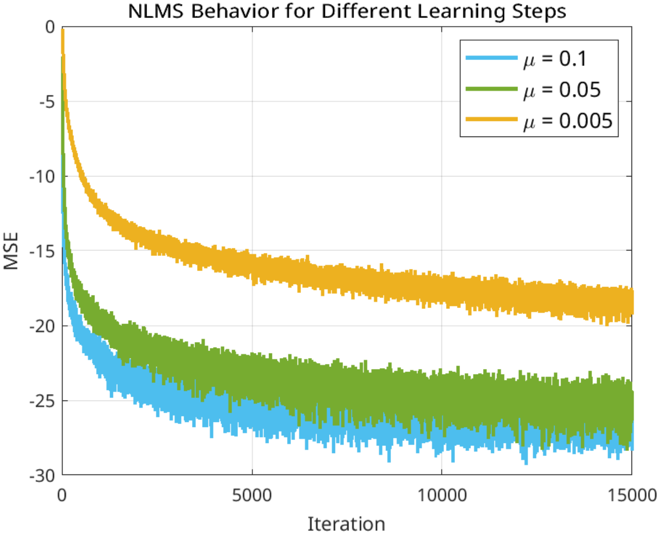

Figure Monter Carlo Experiment with 2500 runs for NLMS algorithm.

TLMS MSE Curve

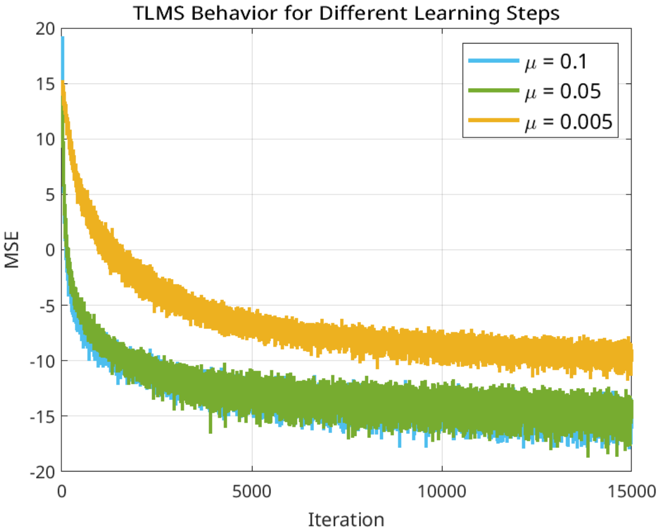

Figure Monter Carlo Experiment with 2500 runs for LMS algorithm.

ATLMS MSE Curve

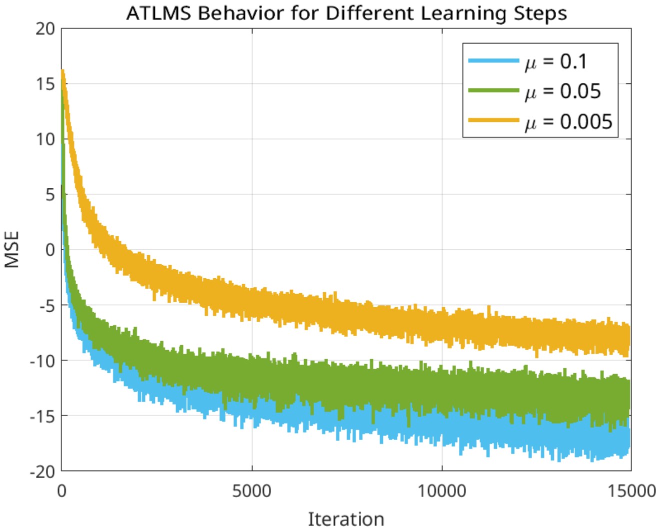

Figure Monter Carlo Experiment with 2500 runs for LMS algorithm.

ATLMS: Different sampling intervals

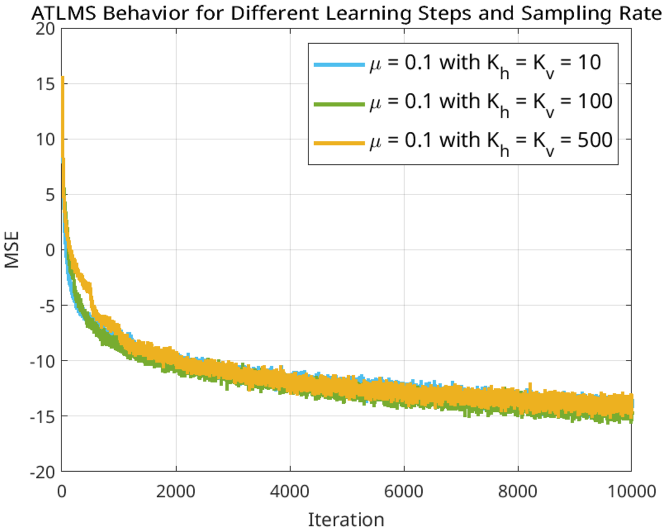

Figure Monter Carlo Experiment with 2500 runs for the ATLMS with different sampling intervals.

Processing Time: TLMS vs. ATLMS

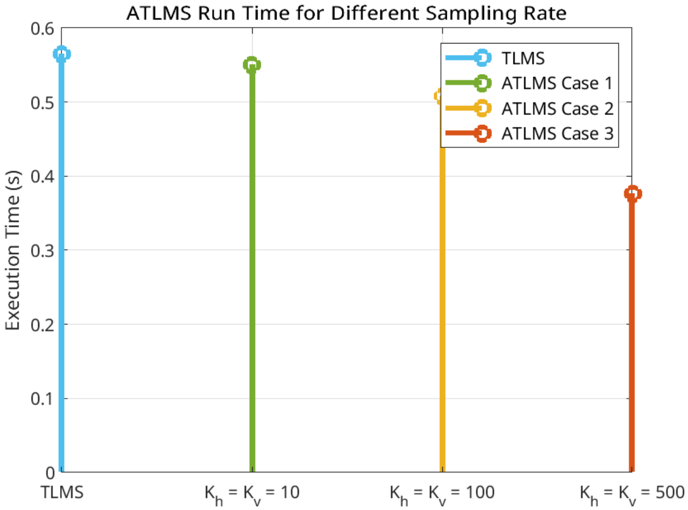

Figure Run time process for ATLMS with different sampling intervals.

Conclusion I

- TLMS and ATLMS algorithms converges faster than the traditional approachs using NLMS.
- TLMS and ATLMS converges to almost the same end, however ATLMS has a greater misadjustment error at the end.
- ATLMS can be slightly faster than the TLMS.

References

[1] L. N. Ribeiro, B. Sokal, A. L. de Almeida, and J. C. M. Mota, "Separable least-mean squares beamforming,"

Thank you for your presence!