Identification en ligne avec régularisation ℓ_1

Algorithme et analyse de convergence en environnement non-stationnaire

Jie CHEN[1,2], Cédric RICHARD[1+], Jose-Carlos M. BERMUDEZ[3], Paul HONEINE[2]

[1] Université de Nice Sophia-Antipolis, UMR Lagrange, CNRS, Observatoire de la Côte d'Azur
[+] Institut Universitaire de France

[2] Université de Technologie de Troyes, Institut Charles Delaunay, UMR STMR, CNRS

[3] Department of Electrical Engineering, Federal University of Santa Catarina, Florianópolis, SC, Brazil

jie.chen@unice.fr, cedric.richard@unice.fr, j.bermudez@ieee.org, paul.honeine@utt.fr

Résumé – Cet article présente une méthode d'identification en ligne de système linéaire avec régularisation ℓ_1. L'analyse de convergence est effectuée pour des environnements non-stationnaires. Ce travail est une extension significative des algorithmes de la famille *non-negative LMS*, à la fois dans la forme de l'algorithme et de l'analyse de convergence. Par ses performances et son coût calculatoire réduit, l'algorithme présente des caractéristiques comparables à l'algorithme LMS tout en prenant en compte la régularisation ℓ_1. Des simulations valident la méthode proposée et le modèle de convergence.

Abstract – This paper presents an online system identification method with ℓ_1-norm regularization. Convergence analysis is performed for non-stationary environments. This work is a significant extension of the non-negative LMS in both aspects of algorithm derivation and convergence analysis. According to its performance and computational cost, the proposed algorithm performs similarly as the LMS algorithm but incorporates ℓ_1-regularization. Experiments validate the proposed algorithm and its convergence analysis.

1 Introduction

Several applications have recently shown the need for online sparse identification techniques. For instance, a particular driving force behind the development of such algorithms is the channel estimation problem, due to the fact that although the number of coefficients of the impulse is large, only a small portion has significant values. Compressive sensing theory provides a robust framework to estimate sparse signals. Instead of the accurate count of non-zero elements by ℓ_0-norm, which leads to NP hard optimization problems, other sparse-induced norms can be used to overcome the difficulty caused by the ℓ_0-norm. The use of the ℓ_1-norm constraint or regularization is a popular choice.

Many approaches to solve the ℓ_1-norm related problem have been described in the literature. Interior-point methods transfer these problems to a convex quadratic problem [10]. Other recent methods include coordinate-wise descent methods [7], iterated shrinkage methods [5], gradient methods [9], gradient projection for sparse reconstruction algorithm [6], and Bregman iterative method [1].

However, the above methods all operate in batch mode. To identify sparse systems in an online way, several adaptive algorithms have also been proposed, including proportionate adaptive filters which incorporate the importance of the individual components by weights, ℓ_0-norm constraint LMS algorithm which approximates ℓ_0-norm by a differentiable function [8], sparse LMS which uses the sign function as the subgradient of the ℓ_1-norm [4]. The last one of these three algorithms directly deals with system identification problems subject to ℓ_1-norm constraint. However, the convergence of subgradient method is not guaranteed, even in batch mode, unless the step size is carefully chosen.

Considering that the ℓ_1 term can be rewritten as the sum of two non-negative vectors, the ℓ_1-regularized problem can be transformed into a minimization problem under non-negativity constraints. Benefiting from this fact, we can solve this problem in an online manner efficiently using our previously proposed constrained system identification method, the so-called non-negative least-mean-square algorithm [3]. In this paper, we derive this algorithm and we propose models that characterize its mean-weight behavior and its second-order behavior.

2 Presentation of the method

Consider an unknown linear system, parameterized by the discrete response vector $\boldsymbol{\alpha}^*$ of length N, with input $x(n)$ and desired reference $y(n)$. We intend to determine the coefficients of the system by minimizing the mean-square error with the sparsity-induced ℓ_1-norm

$$\boldsymbol{\alpha}^o = \arg\min_{\boldsymbol{\alpha}} \frac{1}{2}\, E\{[\boldsymbol{\alpha}^\top \boldsymbol{x}(n) - y(n)]^2\} + \lambda\, \|\boldsymbol{\alpha}\|_1 \qquad (1)$$

where the parameter λ provides a tradeoff between data fidelity and solution sparsity. This ℓ_1-regularized problem can be easily rewritten as a standard non-negative least-square problem, by introducing two N-length non-negative vectors $\boldsymbol{\alpha}^+$ and $\boldsymbol{\alpha}^-$ which satisfy the following relations

$$\begin{aligned}&\boldsymbol{\alpha} = \boldsymbol{\alpha}^+ - \boldsymbol{\alpha}^-\\ &\text{with} \quad \boldsymbol{\alpha}^+, \boldsymbol{\alpha}^- \succeq 0\end{aligned} \tag{2}$$

where $\succeq$ is the component-wised symbol greater than or equal. These relations are satisfied by $\alpha_i^+ = \{\alpha_i\}_+$ and $\alpha_i^- = \{-\alpha_i\}_-$ for all $i = 1, 2, \ldots, N$, where $\{\cdot\}_+$ denotes the positive-part operator defined as $\{x\}_+ = \max\{0, x\}$. For simplicity, let us define a new vector $\tilde{\boldsymbol{\alpha}}$ of length $2N$ by associating $\boldsymbol{\alpha}^+$ and $\boldsymbol{\alpha}^-$ as follows

$$\tilde{\boldsymbol{\alpha}} = [\boldsymbol{\alpha}^{+\top} \;\; \boldsymbol{\alpha}^{-\top}]^\top. \tag{3}$$

We also define the extended input vector

$$\tilde{\boldsymbol{x}}(n) = [\boldsymbol{x}^\top(n) \;\; -\boldsymbol{x}^\top(n)]^\top. \tag{4}$$

The problem (1) can thus be reformulated with respect to vector $\tilde{\boldsymbol{\alpha}}$ by

$$\begin{aligned}\tilde{\boldsymbol{\alpha}}^o = &\arg\min_{\tilde{\boldsymbol{\alpha}}} \frac{1}{2} E\left\{[\tilde{\boldsymbol{\alpha}}^\top \tilde{\boldsymbol{x}}(n) - y(n)]^2\right\} + \lambda\, \mathbf{1}_{2N}^\top \tilde{\boldsymbol{\alpha}}\\ &\text{subject to } \tilde{\boldsymbol{\alpha}} \succeq 0\end{aligned} \tag{5}$$

with $\mathbf{1}_{2N}$ an all-one vector with $2N$ elements. Although the decomposition (2) is not unique, one can observe that the ℓ_2-term is unaffected if we set

$$\boldsymbol{\alpha}^+ \leftarrow \boldsymbol{\alpha}^+ + \boldsymbol{s}$$

and

$$\boldsymbol{\alpha}^- \leftarrow \boldsymbol{\alpha}^- + \boldsymbol{s},$$

where $\boldsymbol{s} \succeq 0$ is a shift vector. However such a shift increases the regularization term in (5) by $\lambda\, \mathbf{1}_{2N}^\top \boldsymbol{s}$. It follows that, at the optimum for the problem (5), either $\alpha_i^+ = 0$ or $\alpha_i^- = 0$, for $i = 1, 2, \ldots, N$ so that in fact $\alpha_i^+ = \{\alpha_i\}_+$ and $\alpha_i^- = \{-\alpha_i\}_+$.

Note that the problem (5) has been reformulated as a system identification problem under non-negativity constraint with respect to $\tilde{\boldsymbol{\alpha}}$. Considering the stochastic gradient approximation in which the correlation matrix $\boldsymbol{R}_x$ and cross-correlation $\boldsymbol{r}_{xy}$ are replaced by their instantaneous estimates $\boldsymbol{x}(n)\boldsymbol{x}^\top(n)$ and $\boldsymbol{x}(n)\, y(n)$, we can now solve this problem in an online manner based on our previously proposed non-negative LMS algorithm [3]

$$\tilde{\boldsymbol{\alpha}}(n+1) = (1 - \eta\,\lambda)\, \tilde{\boldsymbol{\alpha}}(n) + \eta\, \boldsymbol{D}_{\tilde{\boldsymbol{\alpha}}}(n)\, e(n)\, \tilde{\boldsymbol{x}}(n) \tag{6}$$

where η is the step size, $\boldsymbol{D}_{\tilde{\boldsymbol{\alpha}}}(n)$ is the diagonal matrix with the ith diagonal element $\tilde{\alpha}_i(n)$, and $e(n)$ is the estimation error such that $e(n) = y(n) - \boldsymbol{\alpha}^\top(n)\, \boldsymbol{x}(n)$. Using the relation (2), at each time instant the system coefficients $\boldsymbol{\alpha}(n)$ is obtained by

$$\boldsymbol{\alpha}(n) = \boldsymbol{\alpha}^+(n) - \boldsymbol{\alpha}^-(n) \tag{7}$$

This algorithm is based on the NNLMS algorithm. Thus, the variants of NNLMS can also be applied to improve its performance [2].

3 Algorithm behavior modeling in non-stationary environments

We now study the mean-weight behavior of the proposed adaptive algorithm (6) in an arbitrary but quite general time-variant environment. The input $x(n)$ and the desired output $y(n)$ signals are assumed stationary and zero-mean. The signal $z(n) = y(n) - \boldsymbol{x}^\top \boldsymbol{\alpha}^*$ accounts for measurement noise and modeling errors. It assumed that $z(n)$ is stationary, zero-mean with the variance σ_z^2 and statistically independent of any other signal. The time-variant environment is defined with respect to the system coefficients $\boldsymbol{\alpha}^*$ as

$$\boldsymbol{\alpha}^*(n) = \boldsymbol{\alpha}_{\mathrm{o}}^*(n) + \boldsymbol{\xi}(n) \tag{8}$$

where $\boldsymbol{\alpha}_{\mathrm{o}}^*(n)$ is a deterministic time-variant mean, and $\boldsymbol{\xi}(n)$ is a zero-mean random variable with covariance $\boldsymbol{\Xi} = \sigma_\xi^2 \boldsymbol{I}$ and independent of any other signal. This simple model provides some information on how the performance of the proposed algorithms is affected by a time-variant optimal solution which consists of a deterministic trajectory and a random perturbation. The model (8) leads to a tractable analysis and permits inferences about the behavior of the algorithms in time-variant environments by varying the mean value $\boldsymbol{\alpha}_{\mathrm{o}}^*(n)$ and the power σ_ξ^2 of $\boldsymbol{\xi}(n)$.

3.1 Mean weight behavior analysis

We define the weight-error vector with respect to the mean coefficients $\boldsymbol{\alpha}_{\mathrm{o}}^*(n)$ by

$$\tilde{\boldsymbol{v}}(n) = \tilde{\boldsymbol{\alpha}}(n) - \tilde{\boldsymbol{\alpha}}_{\mathrm{o}}^*(n) \tag{9}$$

For the feasibility of the analysis, the following independence assumption is considered in the derivation.

Assumption 1 *(Independence assumption) The input signal $\tilde{\boldsymbol{x}}(n)$ is independent of the weight error vector $\tilde{\boldsymbol{v}}(m)$, for all time index $m \leq n$.*

Although not true in general, this assumption is commonly used for adaptive filter analysis, and the analytical result is usually not sensitive to this approximation. Now using the relation (9) in the update equation (6), we get an update equation for the weight-error vector

$$\tilde{\boldsymbol{v}}(n+1) = (1 - \eta\,\lambda)\tilde{\boldsymbol{v}}(n) - \boldsymbol{\Delta}(n) + \eta\, \boldsymbol{D}_{\tilde{x}}(n)\tilde{\boldsymbol{\alpha}}(n)e(n) \tag{10}$$

where $\boldsymbol{\Delta}(n) = \tilde{\boldsymbol{\alpha}}_{\mathrm{o}}^*(n+1) - (1 - \eta\,\lambda)\, \tilde{\boldsymbol{\alpha}}_{\mathrm{o}}^*(n)$. Note that the estimation error can be expressed by

$$e(n) = z(n) - \tilde{\boldsymbol{v}}^\top(n)\tilde{\boldsymbol{x}}(n) + \boldsymbol{\xi}^\top(n)\boldsymbol{x}(n)$$

Now taking the expectation of (10), neglecting the statistical dependence of $\tilde{\boldsymbol{x}}(n)$ and $\tilde{\boldsymbol{v}}(n)$, observing that the vector $\boldsymbol{\xi}(n)$ is zero-mean and independent of the other signals, and using $E\{z(n)\boldsymbol{D}_{\tilde{\boldsymbol{x}}}(n)\} = 0$, yields the *mean-weight behavior model*

$$\begin{aligned}E\{\tilde{\boldsymbol{v}}(n+1)\} = &\Big((1 - \eta\,\lambda)\boldsymbol{I} - \eta\, \boldsymbol{D}_{\tilde{\boldsymbol{\alpha}}_{\mathrm{o}}^*}(n)\tilde{\boldsymbol{R}}_x\Big)\, E\{\tilde{\boldsymbol{v}}(n)\}\\ &- \boldsymbol{\Delta}(n) - \eta\, \mathrm{diag}\{\tilde{\boldsymbol{R}}_x\, \tilde{\boldsymbol{K}}(n)\}\end{aligned} \tag{11}$$

where $\tilde{\boldsymbol{K}}(n) = E\{\tilde{\boldsymbol{v}}(n)\tilde{\boldsymbol{v}}^\top(n)\}$ is the covariance matrix of $\tilde{\boldsymbol{v}}(n)$, and $\tilde{\boldsymbol{R}}_x$ is the covariance matrix of $\tilde{\boldsymbol{x}}(n)$. Note that the recursion (11) needs the second-order statistics of $\tilde{\boldsymbol{v}}(n)$. In order to simplify the model, we can make the following assumption $\tilde{\boldsymbol{K}}(n) \approx E\{\tilde{\boldsymbol{v}}(n)\}E\{\tilde{\boldsymbol{v}}^\top(n)\}$ in (11) to provide a simplified model. The first order analysis of adaptive filters is usually insensitive to this kind of approximation. As to be seen in experiments, the simplified model will provide accurate behavior description.

3.2 Excess mean-square error analysis

Neglecting the statistical dependence of $\boldsymbol{x}(n)$ and $\tilde{\boldsymbol{v}}(n)$, and using the properties assumed for $z(n)$ and $\boldsymbol{\xi}(n)$, yields an expression for the *mean-square error* model

$$E\left\{e^2(n)\right\} = E\left\{(z(n) - \tilde{\boldsymbol{v}}^\top(n)\,\tilde{\boldsymbol{x}}(n) + \boldsymbol{\xi}^\top(n)\,\boldsymbol{x}(n))^2\right\}$$
$$= \sigma_z^2 + \text{trace}\{\tilde{\boldsymbol{R}}_x\,\tilde{\boldsymbol{K}}(n)\} + \text{trace}\{\boldsymbol{R}_x\,\boldsymbol{\Xi}\}$$

The *excess mean-square error* (EMSE), which is usually more favorable in the performance analysis, is correspondingly provided by

$$J_{\text{EMSE}}(n) = \text{trace}\{\tilde{\boldsymbol{R}}_x\,\tilde{\boldsymbol{K}}(n)\} + \text{trace}\{\boldsymbol{R}_x\,\boldsymbol{\Xi}\} \qquad (12)$$

The term $\text{trace}\{\boldsymbol{R}_x\,\boldsymbol{\Xi}\}$ is the contribution of the non-stationarity of the system to the excess error caused by random perturbation. In order to determine the excess mean square error due to $\text{trace}\{\tilde{\boldsymbol{R}}_x\tilde{\boldsymbol{K}}(n)\}$, we determine a recursion for $\tilde{\boldsymbol{K}}(n)$, which is provided in Appendix due to its complexity. This result can be used to study the convergence behavior of $E\{e^2(n)\}$ and $J_{\text{EMSE}}(n)$.

4 Experiments

In order to validate the proposed algorithm and the models, we considered non-stationary environments defined by adding two types of time variant terms to stationary coefficients, denoted by "nst.1" and "nst.2", respectively. The system coefficients in these two environments are given by

$$\alpha_i^{*\,(\text{nst.1})}(n) = \alpha_i^{*\,(\text{st.})} + \xi_i(n)$$

and

$$\alpha_i^{*(\text{nst.2})}(n) = \alpha_i^{*\,(\text{st.})} + \frac{|\alpha_{\text{o}_i}^{*\,(\text{st.})}|}{10}\sin\left(\frac{2\,\pi}{T}n + 2\,\pi\frac{i-1}{N}\right) + \xi_i(n)$$

where $\boldsymbol{\alpha}^{*\,(\text{st.})}$ in the above two expressions is the stationary component defined by

$$\alpha_i^{*(\text{stat.})} = \begin{cases} 0.55 - 0.1\,i & i = 1,\dots,5 \\ 0 & i = 6,\dots,25 \\ -0.1\,(i-25) & i = 26,\dots,30 \end{cases}$$

In this first case, there is only one random perturbation added to the stationary unconstrained solutions. Whereas a deterministic sinusoidal time-varying trajectory is also added in the second case. The period T of sinusoidal components was set to 2500. The input signal was an AR process given by $x(n) = 0.5\,x(n-1) + w(n)$, with $w(n)$ i.i.d. zero-mean Gaussian with variance σ_w^2, adjusted to obtain the desired input power $\sigma_x^2 = 1$. The step size was set to $\eta = 0.005$ and regularization parameter was set to $\lambda = 0.06$. The variance of modeling noise error $z(n)$ remained $\sigma_z^2 = 0.01$.

Experiment results are shown in Fig. 1. The blue and red curves show the simulation results, and the theoretical predictions by (11) for the mean weight behavior, and by (12) for the EMSE . Simulation results were obtained by averaging 100 rMonte-Carlo runs. The mean weight behavior curves are illustrated in Figs. 1(a) and 1(b). For the second-order EMSE curves, in addition to these curves, the variance σ_ξ^2 was varied using the values in $\{0, 0.001, 0.005\}$. Corresponding EMSE curves obtained from the theoretical model are illustrated in Figs. 1(c) and 1(d). The simulation results conform with these curves but are not shown for clarity. Effects of the deterministic time-varying trajectory and random perturbations can be clearly observed in these figures. Extra EMSE arises due to tracking of the optimal solution variations.

5 Conclusion

In this paper, we studied the online system identification problem regularized by ℓ_1-norm. This was performed by extending the NN-LMS algorithm. Analytical behavior model of the proposed algorithm was studied in non-stationary environments. Future work may include exploring variants of the algorithm to improve its performance.

Appendix : Recursion of $\tilde{\boldsymbol{K}}(n)$

Post-multiplying (10) by its transpose, taking the expected value, and using the statistical properties of $z(n)$ and $\boldsymbol{\xi}(n)$, yields

$$\begin{aligned}\tilde{\boldsymbol{K}}(n+1) = {} & (1-\eta\lambda)^2\,\tilde{\boldsymbol{K}}(n) - (\boldsymbol{P}_1(n) + \boldsymbol{P}_1^\top(n)) \\ & - \eta\,(\boldsymbol{P}_2(n) + \boldsymbol{P}_2^\top(n)) - \eta\,(\boldsymbol{P}_3(n) + \boldsymbol{P}_3^\top(n)) + \boldsymbol{P}_4(n) \\ & + (\boldsymbol{P}_5(n) + \boldsymbol{P}_5^\top(n)) + (\boldsymbol{P}_6(n) + \boldsymbol{P}_6^\top(n)) + \eta^2\,\boldsymbol{P}_7(n) \\ & + \eta\,(\boldsymbol{P}_8(n) + \boldsymbol{P}_8^\top(n)) + \eta^2\,\boldsymbol{P}_9(n) + \eta^2\,\boldsymbol{P}_{10}(n) \\ & + \eta\,(\boldsymbol{P}_{11}(n) + \boldsymbol{P}_{11}^\top(n)) + \eta\,(\boldsymbol{P}_{12}(n) + \boldsymbol{P}_{12}^\top(n)) \\ & + \eta^2(\boldsymbol{P}_{13}(n) + \boldsymbol{P}_{14}(n) + \boldsymbol{P}_{15}(n) + \boldsymbol{P}_{15}^\top(n)) \end{aligned} \qquad (13)$$

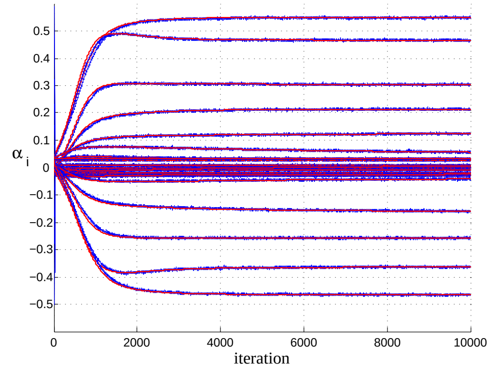
where matrices $\boldsymbol{P}_1(n)$ to $\boldsymbol{P}_{12}(n)$ are calculated by

$$\begin{aligned}
\boldsymbol{P}_1(n) &= (1-\eta\lambda)E\left\{\boldsymbol{v}(n)\right\}\,\boldsymbol{\Delta}^\top(n) \\
\boldsymbol{P}_2(n) &= (1-\eta\lambda)\,\tilde{\boldsymbol{K}}(n)\,\tilde{\boldsymbol{R}}_x\,\boldsymbol{D}_{\tilde{\boldsymbol{v}}}(n) \\
\boldsymbol{P}_3(n) &= (1-\eta\lambda)\,\tilde{\boldsymbol{K}}(n)\,\tilde{\boldsymbol{R}}_x\,\boldsymbol{D}_{\tilde{\boldsymbol{\alpha}}_{\text{o}}^*}(n) \\
\boldsymbol{P}_4(n) &= E\{\boldsymbol{\Delta}(n)\,\boldsymbol{\Delta}^\top(n)\} \\
\boldsymbol{P}_5(n) &= \boldsymbol{\Delta}(n)\big(\text{diag}\{\tilde{\boldsymbol{R}}_x\,\tilde{\boldsymbol{K}}(n)\}\big)^\top \\
\boldsymbol{P}_6(n) &= \boldsymbol{\Delta}\,E\{\tilde{\boldsymbol{v}}^\top(n)\}\,\tilde{\boldsymbol{R}}_x\,\boldsymbol{D}_{\tilde{\boldsymbol{\alpha}}_{\text{o}}^*}(n) \\
\boldsymbol{P}_7(n) &= \sigma_z^2\,(\tilde{\boldsymbol{R}}_x \circ \tilde{\boldsymbol{K}}(n)) \\
\boldsymbol{P}_8(n) &= \sigma_z^2\,E\left\{\boldsymbol{D}_{\tilde{\boldsymbol{v}}}(n)\right\}\,\tilde{\boldsymbol{R}}_x\,\boldsymbol{D}_{\tilde{\boldsymbol{\alpha}}_{\text{o}}^*}(n) \\
\boldsymbol{P}_9(n) &= \sigma_z^2\,\boldsymbol{D}_{\tilde{\boldsymbol{\alpha}}_{\text{o}}^*}(n)\,\tilde{\boldsymbol{R}}_x\,\boldsymbol{D}_{\tilde{\boldsymbol{\alpha}}_{\text{o}}^*}(n) \\
\boldsymbol{P}_{10}(n) &= \boldsymbol{Q}(n) \circ \tilde{\boldsymbol{K}}(n) \\
\boldsymbol{P}_{11}(n) &= E\left\{\boldsymbol{D}_{\tilde{\boldsymbol{v}}}(n)\right\}\,\boldsymbol{Q}(n)\,\boldsymbol{D}_{\tilde{\boldsymbol{\alpha}}_{\text{o}}^*}(n) \\
\boldsymbol{P}_{12}(n) &= \boldsymbol{D}_{\tilde{\boldsymbol{\alpha}}_{\text{o}}^*}(n)\,\boldsymbol{Q}(n)\,\boldsymbol{D}_{\tilde{\boldsymbol{\alpha}}_{\text{o}}^*}(n)
\end{aligned}$$

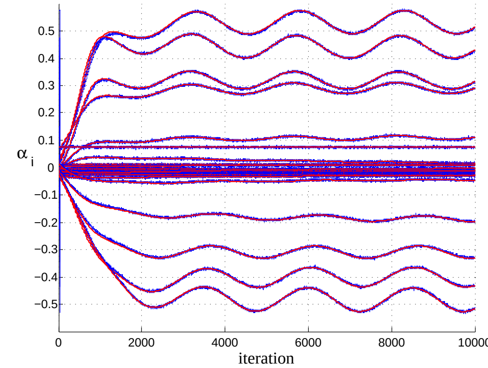
where in the above expressions

$$\boldsymbol{Q}(n) = \text{trace}\{\tilde{\boldsymbol{R}}_x\tilde{\boldsymbol{K}}(n)\}\tilde{\boldsymbol{R}}_x + 2\tilde{\boldsymbol{R}}_x\tilde{\boldsymbol{K}}(n)\tilde{\boldsymbol{R}}_x.$$

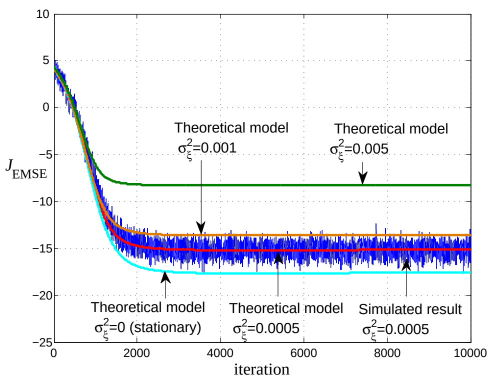
These terms do not involve the random perturbation $\boldsymbol{\xi}(n)$ on the weights. We now derive expressions for $\boldsymbol{P}_{13}(n)$ through $\boldsymbol{P}_{15}(n)$. These terms convey the effect of the environment non-stationarity due to the random variations



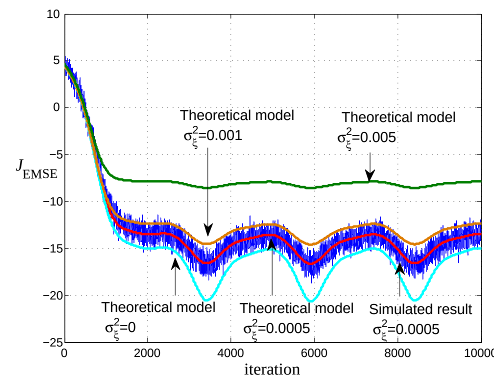
(a) mean behavior with $\boldsymbol{\alpha}^{*\,(\mathrm{nst.1})}$



(b) mean behavior with $\boldsymbol{\alpha}^{*\,(\mathrm{nst.2})}$



(c) 2nd-order behavior with $\boldsymbol{\alpha}^{*\,(\mathrm{nst.1})}$



(d) 2nd-order behavior with $\boldsymbol{\alpha}^{*\,(\mathrm{nst.2})}$

FIGURE 1 – Algorithm behavior in non-stationary environments. First row : first order results with $\sigma_\xi^2 = 0.005$. Second row : Second order results with various levels of σ_ξ^2.

of system weights. These terms are expressed by

$$\boldsymbol{P}_{13}(n) = \sigma_\xi^2\, \tilde{\boldsymbol{K}}(n) \circ (\tilde{\boldsymbol{R}}_x\, \mathrm{trace}\{\boldsymbol{R}_x\} + 2\, \boldsymbol{R}'_x\, \boldsymbol{R}'^{\top}_x)$$
$$\boldsymbol{P}_{14}(n) = \sigma_\xi^2 \left(\boldsymbol{\alpha}_\mathrm{o}^*(n)\, \boldsymbol{\alpha}_\mathrm{o}^{*\top}(n)\right) \circ (\tilde{\boldsymbol{R}}_x\, \mathrm{trace}\{\boldsymbol{R}_x\} + 2\, \boldsymbol{R}'_x \boldsymbol{R}'^{\top}_x)$$
$$\boldsymbol{P}_{15}(n) = \sigma_\xi^2 \left(E\{\tilde{\boldsymbol{v}}(n)\}\boldsymbol{\alpha}_\mathrm{o}^{*\top}(n)\right) \circ (\tilde{\boldsymbol{R}}_x \mathrm{trace}\{\boldsymbol{R}_x\} + 2\, \boldsymbol{R}'_x \boldsymbol{R}'^{\top}_x)$$

with $\boldsymbol{R}'_x = [\boldsymbol{R}_x, -\boldsymbol{R}_x]^\top$. Using the expected values $\boldsymbol{P}_1(n)$ to $\boldsymbol{P}_{15}(n)$ in (13), we finally obtain a recursive analytical model for the behavior of $\tilde{\boldsymbol{K}}(n)$. More detailed derivations are similar to those of NN-LMS algorithm and its variants, and can be referred to [2, 3].

Références

[1] J. F. Cai, S. Osher, and Z. Shen. Linearized bregman iterations for compressed sensing. *Mathematical of Computation*, 78(267) :1515–1536, 2009.

[2] J. Chen, C. Richard, J.-C. M. Bermudez, and P. and Honeine. A modified non-negative lms algorithm and its stochastic behavior analysis. In *Proc. of Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, 2011.

[3] J. Chen, C. Richard, J.-C. M.Bermudez, and P. Honeine. Non-negative least-mean-square algorithm. *IEEE Transactions on Signal Processing*, 59(11) :5525–5535, 2011.

[4] Y. Chen, Y. Gu, and A. O. Hero. Sparse lms for system identification. In *Proc. of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2009.

[5] I. Daubechies, M. Defrise, and C. De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics*, 57(11) :1413–1457, 2004.

[6] M. Figueiredo, R. D. Nowak, and S. J. Wright. Gradient projection for sparse reconstruction : Application to compressed sensing and other inverse problems. *IEEE Journal of Selected Topics in Signal Processing*, 1(4) :586–597, 2007.

[7] J. Friedman, T. Hastie, H. Höfling, and R. Tibshirani. Pathwise coordinate optimization. *The Annals of Applied Statistics*, 2(1) :302–332, 2007.

[8] Y. Gu, J. Jin, and S. Mei. ℓ_0-norm constraint lms algorithm for sparse system identification. *IEEE Signal Processing Letters*, 16(7) :774–777, 2009.

[9] Y. Nesterov. Gradient methods for minimizing composite objective function. Technical report, Université catholique de Louvain, Center for Operations Research and Econometrics (CORE), 2007.

[10] Y. Nesterov and A. Nemirovsky. Interior-point polynomial methods in convex programming. In *Studies in Applied Mathematics*, volume 13. Philadelphia, PA : SIAM, 1994.