



HAL
open science

System identification under non-negativity constraints

Jie Chen, Cédric Richard, Paul Honeine, Henri Lantéri, Céline Theys

► **To cite this version:**

Jie Chen, Cédric Richard, Paul Honeine, Henri Lantéri, Céline Theys. System identification under non-negativity constraints. European Signal Processing Conference, Aug 2010, Aalborg, Denmark. pp.1728-1732. hal-02111262

HAL Id: hal-02111262

<https://hal-normandie-univ.archives-ouvertes.fr/hal-02111262>

Submitted on 25 Apr 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

SYSTEM IDENTIFICATION UNDER NON-NEGATIVITY CONSTRAINTS

Jie Chen ^{*†}, Cédric Richard ^{*}, Paul Honeine [†], Henri Lanteri ^{*} and Céline Theys ^{*}

^{*} Laboratoire Fizeau, Université de Nice Sophia-Antipolis
Observatoire de la Côte d'Azur, UMR CNRS 6525, Parc Valrose, 06108 Nice France

E-mail: cedric.richard@unice.fr, henri.lanteri, celine.theys

[†] Institut Charles Delaunay, Université de Technologie de Troyes, FRE CNRS 2848, 10010 Troyes France
E-mail: paul.honeine@utt.fr

ABSTRACT

Dynamic system modeling plays a crucial role in the development of techniques for stationary and non-stationary signal processing. Due to the inherent physical characteristics of systems usually under investigation, non-negativity is a desired constraint that can be imposed on the parameters to estimate. In this paper, we propose a general method for system identification under non-negativity constraints. We derive additive and multiplicative weight update algorithms, based on (stochastic) gradient descent of mean-square error or Kullback-Leibler divergence. Experiments are conducted to validate the proposed approach.

1. INTRODUCTION

In many real-life phenomena including biological and physiological ones, due to the inherent physical characteristics of systems under investigation, non-negativity is a desired constraint that can be imposed on the parameters to estimate. For instance, in the study of a concentration field or a thermal radiation field, any observation is described with non-negative values (ppm, joule). Non-negativity as a physical constraint has received growing attention from the signal processing community during the last decade. For instance, consider the following non-negative least-square problem

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{A}\mathbf{x} - \mathbf{b}\|^2 \quad (1)$$

$$\text{subject to } \mathbf{x} \geq 0 \quad (2)$$

where $\|\cdot\|$ denotes the Euclidean 2-norm. It has recently been addressed within different contexts, with applications ranging from imaging deblurring [2] to biological data processing [4]. Another similar problem is non-negative matrix factorization, which is now a popular dimension reduction technique [7]. Given a matrix \mathbf{V} with non-negative entries, it consists of seeking a factorization of the form

$$\mathbf{V} \approx \mathbf{W}\mathbf{H} \quad (3)$$

where \mathbf{W} and \mathbf{H} are non-negative matrices. This problem is closely related to the blind deconvolution one.

Two classes of algorithms have been proposed in the literature to solve optimization problems with non-negativity constraints. The first one encompasses the so-called alternating non-negative least squares or projected gradient [1, 3]. These are based on successive projections onto the feasible region, resulting in low-cost operations for simple constraints. Their low-memory requirement makes them attractive for large scale problems. The second one, investigated more recently, consists of multiplicative weight update methods. They provide natural and computationally efficient schema to a large number of problems involving non-negativity constraints [5, 7, 8]. In particular, the application described in [5] refers to the maximum-likelihood

signal restoration for image deblurring. Non-negativity constraints arise from the nature of image data and blurring masks. These are satisfied by a proper choice, at each iteration, of the step size of a gradient descent method. Setting the latter to one leads to a multiplicative form.

In this paper, we study the problem of system identification with non-negativity constraints. We derive additive and multiplicative algorithms based on gradient descent of convex cost functions. In order to deal with non-stationary systems and reduce the computational requirements, we also develop stochastic gradient algorithms that update parameters in an online way.

2. GENERAL PROBLEM FORMULATION

Consider an unknown system, only characterized by a set of real-valued discrete-time responses to known stationary inputs. The problem is to derive a transversal filter model

$$y(n) = \boldsymbol{\alpha}^\top \mathbf{x}(n) + e(n), \quad (4)$$

where $\mathbf{x}(n)$ and $y(n)$ denote the input and noisy output data at time n . Due to the inherent physical characteristics of systems under investigation, non-negativity is a desired constraint that can be imposed on the coefficient vector $\boldsymbol{\alpha}$. Therefore, the problem of identifying the optimum model can be formalized as follows

$$\boldsymbol{\alpha}^\circ = \arg \min_{\boldsymbol{\alpha}} J(\boldsymbol{\alpha}) \quad (5)$$

$$\text{subject to } \boldsymbol{\alpha} \geq 0 \quad (6)$$

with $J(\boldsymbol{\alpha})$ a convex cost function.

2.1 Additive weight update algorithm

In order to solve the problem (5)-(6), let us consider its Lagrangian function $Q(\boldsymbol{\alpha}, \boldsymbol{\lambda})$ given by

$$Q(\boldsymbol{\alpha}, \boldsymbol{\lambda}) = J(\boldsymbol{\alpha}) - \boldsymbol{\lambda}^\top \boldsymbol{\alpha},$$

where $\boldsymbol{\lambda}$ is the vector of non-negative Lagrange multipliers. The Karush-Kuhn-Tucker conditions must necessarily be satisfied at the optimum defined by $\boldsymbol{\alpha}^\circ$, $\boldsymbol{\lambda}^\circ$, namely,

$$\begin{aligned} \nabla_{\boldsymbol{\alpha}} Q(\boldsymbol{\alpha}^\circ, \boldsymbol{\lambda}^\circ) &= 0 \\ \lambda_i^\circ \alpha_i^\circ &= 0, \forall i \end{aligned}$$

where the symbol $\nabla_{\boldsymbol{\alpha}}$ stands for the gradient operator with respect to $\boldsymbol{\alpha}$. These equations can be combined into the following expression

$$\alpha_i^\circ [-\nabla_{\boldsymbol{\alpha}} J(\boldsymbol{\alpha})]_i = 0, \quad (7)$$

where the extra minus sign is just used to make a gradient descent of $J(\boldsymbol{\alpha})$ apparent. To solve equation (7) iteratively,

two important points have to be noticed. The first point is that $\mathbf{D}(-\nabla_{\alpha} J(\boldsymbol{\alpha}))$ is also a gradient descent of $J(\boldsymbol{\alpha})$ if \mathbf{D} is a symmetric positive definite matrix. In this paper, \mathbf{D} denotes a diagonal matrix with i -th diagonal entry $[\mathbf{D}]_{ii} = f_i(\boldsymbol{\alpha})$, where $f_i(\boldsymbol{\alpha}) > 0$. The second point is that equations of the form $\varphi(u) = 0$ can be solved with a fixed-point algorithm, under some conditions on function φ , by considering the problem $u = u + \varphi(u)$. Implementing this fixed-point strategy with equation (7) leads to the component-wise gradient descent algorithm

$$\alpha_i(k+1) = \alpha_i(k) + \eta_i(k) f_i(\boldsymbol{\alpha}(k)) \alpha_i(k) [-\nabla_{\alpha} J(\boldsymbol{\alpha}(k))]_i \quad (8)$$

with $\eta_i(k)$ a positive step size that allows to control convergence. Suppose that $\alpha_i(k) \geq 0$. Non-negativity of $\alpha_i(k+1)$ is guaranteed if

$$1 + \eta_i(k) f_i(\boldsymbol{\alpha}(k)) [-\nabla_{\alpha} J(\boldsymbol{\alpha}(k))]_i \geq 0. \quad (9)$$

If $[\nabla_{\alpha} J(\boldsymbol{\alpha}(k))]_i \leq 0$, condition (9) is clearly satisfied and non-negativity does not impose any restriction on step size. Conversely, if $[\nabla_{\alpha} J(\boldsymbol{\alpha}(k))]_i > 0$, non-negativity of $\alpha_i(k+1)$ holds if

$$0 \leq \eta_i(k) \leq \frac{1}{f_i(\boldsymbol{\alpha}(k)) [\nabla_{\alpha} J(\boldsymbol{\alpha}(k))]_i}. \quad (10)$$

Before going further, note that the right-hand side member of (8) is a descent direction because $\eta_i(k) f_i(\boldsymbol{\alpha}(k)) \alpha_i(k) \geq 0$. The general algorithm can thus be written as follows

$$\boldsymbol{\alpha}(k+1) = \boldsymbol{\alpha}(k) + \eta(k) \mathbf{d}(k), \quad (11)$$

where $\mathbf{d}(k)$ is the descent direction defined by

$$\mathbf{d}(k) = -\text{diag}(f_i(\boldsymbol{\alpha}(k)) \alpha_i(k)) \nabla_{\alpha} J(\boldsymbol{\alpha}(k)). \quad (12)$$

Step size $\eta(k)$ is now independent of index i , and can be obtained via a line search procedure in the interval $]0, \eta_{\max}(k)[$ defined by

$$\eta_{\max}(k) = \min_i \frac{1}{f_i(\boldsymbol{\alpha}(k)) [\nabla_{\alpha} J(\boldsymbol{\alpha}(k))]_i}. \quad (13)$$

2.2 Multiplicative weight update algorithm

Let us now decompose the gradient $-\nabla_{\alpha} J(\boldsymbol{\alpha}(k))$ as follows

$$[-\nabla_{\alpha} J(\boldsymbol{\alpha}(k))]_i = [\mathbf{U}(\boldsymbol{\alpha}(k))]_i - [\mathbf{V}(\boldsymbol{\alpha}(k))]_i \quad (14)$$

where $[\mathbf{U}(\boldsymbol{\alpha}(k))]_i$ and $[\mathbf{V}(\boldsymbol{\alpha}(k))]_i$ are strictly positive components. Obviously, such a decomposition is not unique but always exists. Using $f_i(\boldsymbol{\alpha}(k)) = 1/[\mathbf{V}(\boldsymbol{\alpha}(k))]_i$, the updating equation (8) becomes

$$\alpha_i(k+1) = \alpha_i(k) + \eta_i(k) \alpha_i(k) \left(\frac{[\mathbf{U}(\boldsymbol{\alpha}(k))]_i}{[\mathbf{V}(\boldsymbol{\alpha}(k))]_i} - 1 \right). \quad (15)$$

Let us determine the maximum value for the step size in order that $\alpha_i(k+1) \geq 0$, given $\alpha_i(k) \geq 0$. As shown previously, note that a restriction may only apply if

$$[\mathbf{U}(\boldsymbol{\alpha}(k))]_i - [\mathbf{V}(\boldsymbol{\alpha}(k))]_i < 0 \quad (16)$$

According to condition (10), the maximum step size which ensures the positivity of $\alpha_i(k+1)$ is given by

$$\eta_i(k) \leq \frac{1}{1 - \frac{[\mathbf{U}(\boldsymbol{\alpha}(k))]_i}{[\mathbf{V}(\boldsymbol{\alpha}(k))]_i}}, \quad (17)$$

which is strictly greater than 1. The use of a step size equal to 1 leads to the very simple form

$$\boldsymbol{\alpha}(k+1) = \text{diag} \left(\frac{[\mathbf{U}(\boldsymbol{\alpha}(k))]_i}{[\mathbf{V}(\boldsymbol{\alpha}(k))]_i} \right) \boldsymbol{\alpha}(k). \quad (18)$$

This expression is referred to as the multiplicative weight update algorithm.

2.3 Stochastic-gradient formulation

An important issue that has not been discussed up to now is the evaluation of the gradient. This requires knowledge of some statistical properties of the signal at hand, which cannot often be achieved in real-life applications. In order to circumvent this difficulty, we propose to use an instantaneous estimate of the gradient, in the same manner as for conventional LMS-based algorithms. Substituting $\nabla_{\alpha} J$ with its instantaneous estimate in (8) leads to

$$\alpha_i(k+1) = \alpha_i(k) + \eta_i(k) f_i(\boldsymbol{\alpha}(k)) \alpha_i(k) [-\tilde{\nabla}_{\alpha} J(\boldsymbol{\alpha}(k))]_i.$$

As we will see in the next section, coefficient update at each iteration only uses an instantaneous pair of input/output data. This form is particularly appropriate for online signal processing.

3. FORMULATION WITH MEAN-SQUARE ERROR AND K-L DIVERGENCE

In the previous section, we derived a general framework for solving problem (5)-(6). In particular, we proposed an additive (8) and a multiplicative (18) weight update algorithm for minimizing convex criterion under non-negativity constraints. We shall now present the updating rules to deal with, more specifically, the mean-square error and the Kullback-Leibler divergence. In anticipation of the numerical experiments that will be presented next, we introduce these criteria within the context of maximum-likelihood estimation of Gaussian and Poisson noise corrupted linear systems.

3.1 Minimising the mean-square error

Consider the problem of estimating $\boldsymbol{\alpha}$ in

$$y(n) = \boldsymbol{\alpha}^{\top} \mathbf{x}(n) + e(n), \quad (19)$$

subject to the constraint $\boldsymbol{\alpha} \geq 0$, where system outputs are corrupted by a zero-mean white Gaussian noise $e(n)$ of variance σ^2 . The likelihood function based on N samples is

$$F(\boldsymbol{\alpha}) = \prod_{n=1}^N \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(\boldsymbol{\alpha}^{\top} \mathbf{x}(n) - y(n))^2}{2\sigma^2}\right).$$

The negative log-likelihood function is given by

$$-\log F(\boldsymbol{\alpha}) = \frac{1}{2} \sum_{n=1}^N \frac{(\boldsymbol{\alpha}^{\top} \mathbf{x}(n) - y(n))^2}{\sigma^2}. \quad (20)$$

Dropping the terms that do not depend on $\boldsymbol{\alpha}$, we obtain the cost function to be minimized with respect to $\boldsymbol{\alpha}$, that is,

$$J(\boldsymbol{\alpha}) = \sum_{n=1}^N \left(\boldsymbol{\alpha}^{\top} \mathbf{x}(n) - y(n) \right)^2 \quad (21)$$

subject to $\boldsymbol{\alpha} \geq 0$

where we have included the non-negativity constraints. The gradient of $J(\boldsymbol{\alpha})$ is easily computed as follows

$$\nabla J_{\alpha}(\boldsymbol{\alpha}) = \sum_{n=1}^N \left(\mathbf{x}(n) \mathbf{x}(n)^{\top} \boldsymbol{\alpha} - y(n) \mathbf{x}(n) \right). \quad (22)$$

Using expression (8) with $f_i(\boldsymbol{\alpha}) = \frac{1}{N}$, the additive updating rule for minimising mean-square error under non-negative constraints is given by

$$\boldsymbol{\alpha}(k+1) = \text{diag}(\boldsymbol{\alpha}(k)) \left(\mathbf{1} + \eta(k) (\hat{\mathbf{p}}_{xy} - \hat{\mathbf{R}}_x \boldsymbol{\alpha}(k)) \right) \quad (23)$$

where

$$\eta(k) \leq \min_i \frac{1}{[\hat{\mathbf{R}}_x \boldsymbol{\alpha}(k) - \hat{\mathbf{p}}_{xy}]_i}. \quad (24)$$

In the above expressions, $\hat{\mathbf{R}}_x$ is the estimated autocorrelation matrix of $\mathbf{x}(n)$, and $\hat{\mathbf{p}}_{xy}$ denotes the intercorrelation vector between $\mathbf{x}(n)$ and $y(n)$, defined as

$$\hat{\mathbf{R}}_x = \frac{1}{N} \sum_{n=1}^N \mathbf{x}(n) \mathbf{x}(n)^\top \quad \hat{\mathbf{p}}_{xy} = \frac{1}{N} \sum_{n=1}^N y(n) \mathbf{x}(n). \quad (25)$$

Let us turn now to multiplicative weight update algorithms. Gradient decomposition (14) always exists but is not unique. A possible solution is to consider the following two positive vectors

$$[\mathbf{U}(\boldsymbol{\alpha}(k))] = \max\{[\hat{\mathbf{R}}_x \boldsymbol{\alpha}(k) - \hat{\mathbf{p}}_{xy}]_i, 0\} + \eta \quad (26)$$

$$[\mathbf{V}(\boldsymbol{\alpha}(k))] = -\min\{[\hat{\mathbf{R}}_x \boldsymbol{\alpha}(k) - \hat{\mathbf{p}}_{xy}]_i, 0\} + \eta \quad (27)$$

with η a positive constant to avoid the denominator in (18) to become 0.

Stochastic-gradient formulation

We restrict ourselves to the use an instantaneous pair of input/output data at each time instant k . This can be done by only taking into account $\mathbf{x}(n)$ and $y(n)$ in (28), that is,

$$\hat{\mathbf{R}}_{x,k} = \mathbf{x}(k) \mathbf{x}(k)^\top \quad \hat{\mathbf{p}}_{xy,k} = y(k) \mathbf{x}(k). \quad (28)$$

In this case, the updating rule becomes an instantaneous approximation of (23) defined by

$$\boldsymbol{\alpha}(k+1) = \text{diag}(\boldsymbol{\alpha}(k)) \left(\mathbb{1} + \eta(k) (\hat{\mathbf{p}}_{xy,k} - \hat{\mathbf{R}}_{x,k} \boldsymbol{\alpha}(k)) \right) \quad (29)$$

To ensure non-negativity of the solution, the step size must satisfy at each iteration the upper-bound expression

$$\eta(k) \leq \min_i \frac{1}{[\hat{\mathbf{R}}_{x,k} \boldsymbol{\alpha}(k) - \hat{\mathbf{p}}_{xy,k}]_i}.$$

3.2 Minimizing the Kullback-Leibler divergence

Consider the problem of estimating $\boldsymbol{\alpha}$ from noisy measurements $y(n)$ drawn from a Poisson law, as in many applications involving photon-counting detectors,

$$y(n) \sim \mathcal{P}(\boldsymbol{\alpha}^\top \mathbf{x}(n)), \quad (30)$$

subject to the constraint $\boldsymbol{\alpha} \geq 0$. In this case, $\mathbf{x}(n)$ is non-negative, and $y(n)$ is an integer. If these random variables are independent, the likelihood function based on N samples can be written as [9]

$$F(\boldsymbol{\alpha}) = \prod_{n=1}^N \frac{(\boldsymbol{\alpha}^\top \mathbf{x}(n))^{y(n)}}{y(n)!} \exp(-\boldsymbol{\alpha}^\top \mathbf{x}(n)). \quad (31)$$

Using Stirling's formula, the negative log-likelihood function can be approximated by

$$-\log F(\boldsymbol{\alpha}) \approx \sum_{n=1}^N \left((\boldsymbol{\alpha}^\top \mathbf{x}(n) - y(n)) + y(n) \log \frac{y(n)}{\boldsymbol{\alpha}^\top \mathbf{x}(n)} \right).$$

Dropping the terms that do not depend on $\boldsymbol{\alpha}$, we obtain a generalized Kullback-Leibler divergence to be minimized with respect to $\boldsymbol{\alpha}$ under non-negativity constraints

$$J(\boldsymbol{\alpha}) = \sum_{n=1}^N \left[(\boldsymbol{\alpha}^\top \mathbf{x}(n)) - y(n) \log(\boldsymbol{\alpha}^\top \mathbf{x}(n)) \right] \quad (32)$$

subject to $\boldsymbol{\alpha} \geq 0$

The gradient of $J(\boldsymbol{\alpha})$ is easily computed as follows

$$\nabla J(\boldsymbol{\alpha}) = \sum_{n=1}^N \left(\mathbf{x}(n) - \frac{y(n)}{\boldsymbol{\alpha}^\top \mathbf{x}(n)} \mathbf{x}(n) \right) \quad (33)$$

From (8) and setting $f_i(\boldsymbol{\alpha})$ to 1, we obtain the update equation for the Poisson noise case

$$\boldsymbol{\alpha}(k+1) = \text{diag}(\boldsymbol{\alpha}(k)) \left(\mathbb{1} + \eta(k) \sum_{n=1}^N \left(\frac{y(n)}{(\boldsymbol{\alpha}(k)^\top \mathbf{x}(n))} \mathbf{x}(n) - \mathbf{x}(n) \right) \right) \quad (34)$$

where

$$\eta(n) \leq \min_i \frac{1}{\left[\sum_{n=1}^N \left(\mathbf{x}(n) - \frac{y(n)}{\boldsymbol{\alpha}^\top \mathbf{x}(n)} \mathbf{x}(n) \right) \right]_i}. \quad (35)$$

A multiplicative form of the algorithm is obtained by decomposing (33) in the form $-\nabla J_\alpha(\boldsymbol{\alpha}(k)) = \mathbf{U}(\boldsymbol{\alpha}(k)) - \mathbf{V}(\boldsymbol{\alpha}(k))$, where the two positive terms can be clearly identified as

$$\mathbf{U}(\boldsymbol{\alpha}(k)) = \sum_{n=1}^N \frac{y(n)}{\boldsymbol{\alpha}(k)^\top \mathbf{x}(n)} \mathbf{x}(n) \quad (36)$$

$$\mathbf{V}(\boldsymbol{\alpha}(k)) = \sum_{n=1}^N \mathbf{x}(n) \quad (37)$$

We obtain the corresponding multiplicative form

$$\alpha_i(k+1) = \alpha_i(k) \frac{\left[\sum_{n=1}^N \frac{y(n)}{\boldsymbol{\alpha}(k)^\top \mathbf{x}(n)} \mathbf{x}(n) \right]_i}{\left[\sum_{n=1}^N \mathbf{x}(n) \right]_i}. \quad (38)$$

Stochastic-gradient algorithm

Let us estimate the gradient instantaneously, using only the available data $\mathbf{x}(k)$ and $y(k)$ at time instant k . We obtain the following algorithm

$$\boldsymbol{\alpha}(k+1) = \text{diag}(\boldsymbol{\alpha}(k)) \left(\mathbb{1} + \eta(k) \left(\frac{y(k)}{\boldsymbol{\alpha}(k)^\top \mathbf{x}(k)} \mathbf{x}(k) - \mathbf{x}(k) \right) \right) \quad (39)$$

In order to get a non-negative solution, at each iteration, the step size must be upper-bounded as follows

$$\eta(k) \leq \min_i \frac{1}{\left[\frac{y(k)}{\boldsymbol{\alpha}(k)^\top \mathbf{x}(k)} \mathbf{x}(k) - \mathbf{x}(k) \right]_i}. \quad (40)$$

4. EXPERIMENTAL RESULTS

The purpose of this section is to illustrate the performance of the proposed approach, and consistency between analytical and numerical results. We shall also see how parameters of the algorithm affect its transient performance.

4.1 Gaussian scenario

Consider the Gaussian scenario (19) where the problem at hand is to estimate $\boldsymbol{\alpha}$. Due to non-negativity constraints, in general, there is no closed-form solution for problem (21). However, in order to illustrate the results and study convergence behavior, we consider a simple special case where inputs are i.i.d. and drawn from a zero-mean unit-variance Gaussian distribution. According to the Wiener-Hopf equations, the optimal solution of the unconstrained problem is obtained by solving

$$\mathbf{R}_x \boldsymbol{\alpha}_{\text{ow}} = \mathbf{p}_{xy}.$$

Table 1: System of reference with non-negative entries, and estimations under Gaussian hypothesis.

α	0.8	0.6	0.5	0.4	0.4	0.3	0.3	0.2	0.1	0.1
$\hat{\alpha}_{\eta=0.04}$	0.8566	0.7197	0.4888	0.2781	0.3062	0.1917	0.3267	0.1320	0.0646	0.0359
$\hat{\alpha}_{\eta=0.004}$	0.7986	0.5680	0.4906	0.4021	0.3898	0.3042	0.2819	0.1875	0.1067	0.0885

In the case where \mathbf{R}_x is diagonal, as considered above, it is easy to show that the solution of the constrained problem is obtained by turning the negative entries of α_{ow} to zero

$$\alpha_o = \{\mathbf{R}_x^{-1} \mathbf{p}_{xy}\}_+ \quad (41)$$

where $\{u\}_+ = \max\{0, u\}$. The minimum mean-square error produced by solution α_o is

$$J_{min} = \sigma^2 - 2\mathbf{p}_{xy} \{\alpha_o\}_+ + \{\alpha_o\}_+^T \mathbf{R}_x \{\alpha_o\}_+. \quad (42)$$

with σ^2 the variance of the additive zero-mean white Gaussian noise $e(n)$. The latter was set to 0.5 in the experiments described below.

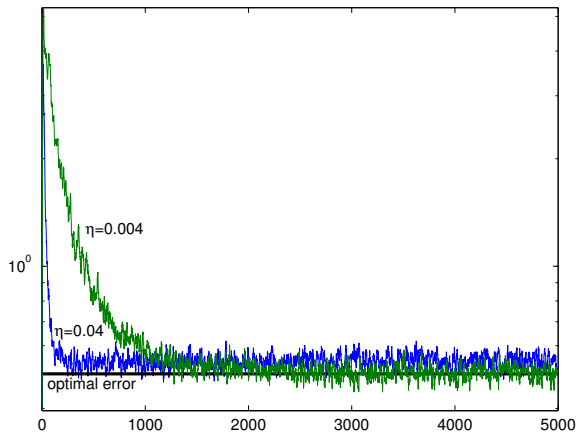


Figure 1: Evolution of mean-square error under Gaussian hypothesis, for several step size values and the system of reference described in Table 1.

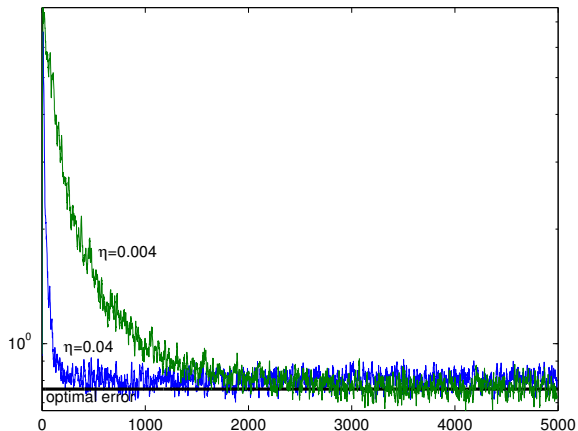


Figure 2: Evolution of mean-square error under Gaussian hypothesis, for several step size values and the system of reference described in Table 2.

First, experiments were conducted to estimate α under non-negativity constraints in the case where the system we used to simulate data was in the set of admissible solutions. See Table 1. From (41) and (42), we have $J_{min} = 0.5$. One hundred realizations of 5000-sample independent sequences were generated in order to compute mean performance. The step size η in (23) was successively fixed at 0.004 and 0.04. The entries of α were initially set to 1. With $\eta = 0.04$, the mean square error over the last 1000 samples was found to be $J_{0.04} = 0.5368$. The algorithm led us to $J_{0.004} = 0.5039$ with $\eta = 0.004$. Table 1 reports the estimated entries of α . Figure 1 presents the ensemble-average learning curves, that is, convergence of the mean-square error. It exhibits a classic behaviour for this kind of algorithms, for which small step size usually means slower convergence but better asymptotic performance.

Next, experiments were conducted in the case where the system of reference was out of the set of admissible solutions. As can be seen in Table 2, this was done by turning 3 entries of α used in the first series of experiments to negative values. In this case, solution of the constrained problem is obtained by setting these negative entries to 0. From (41) and (42), we have $J_{min} = 0.76$. With $\eta = 0.04$, the algorithm led us to $J_{0.04} = 0.8030$ estimated over the last 1000 samples. The mean square error was equal $J_{0.004} = 0.7693$ with $\eta = 0.004$. Table 2 reports the estimated entries of α . The latter shows that those associated to theoretically negative ones are very close to 0, which is consistent with our analysis. Finally, Figure 2 presents the ensemble-average learning curves.

4.2 Poisson scenario

Consider the Poisson scenario (30), where the problem at hand is to estimate α subject to non-negativity constraints. Unfortunately, even in simple cases, solving this problem is analytically intractable. Experiments were however conducted in the case described by Table 1, where the system of reference was in the set of admissible solutions. Inputs $\mathbf{x}(n)$ were absolute-valued data drawn from a zero-mean Gaussian distribution of variance 10^4 . One hundred realizations of 5000-sample independent sequences were generated in order to compute mean performance. The step size η in (34) was successively fixed at 0.0004, 0.001 and 0.004. In Figure 3, the convergence behavior of the algorithm with respect to the step clear shows that small step size usually means slower convergence but better asymptotic performance.

5. CONCLUSION

In many real-life phenomena including biological and physiological ones, due to the inherent physical characteristics of systems under investigation, non-negativity is a desired constraint that can be imposed on the parameters to estimate. In this paper, we proposed a general method for system identification under non-negativity constraints. We derived additive and multiplicative weight update algorithms, based on (stochastic) gradient descent of mean-square error or Kullback-Leibler divergence. Two applications were finally considered, involving Gaussian and Poisson noise corrupted linear systems.

Table 2: System of reference with positive and negative entries, and estimations under Gaussian hypothesis.

α	0.8	0.6	0.5	0.4	-0.4	0.3	-0.3	0.2	-0.1	0.1
$\hat{\alpha}_{\eta=0.04}$	0.7818	0.7184	0.4121	0.3918	0.0	0.3078	0.0	0.1727	0.0	0.1043
$\hat{\alpha}_{\eta=0.004}$	0.8076	0.6175	0.4956	0.4623	0.0002	0.3818	0.0005	0.2189	0.0138	0.1182

REFERENCES

- [1] J. Barzilai and J. Borwein. Two point step size gradient methods. *IMA J. Numer. Anal.*, 8(1):141–148, 1988.
- [2] F. Benvenuto, R. Zanella, L. Zanni, and M. Bertero. Non-negative least-squares image deblurring: improved gradient projection approaches. *Inverse Problems*, 26:025004, 2010.
- [3] P. Calamai and J. Moré. Projected gradient methods for linearly constrained problems. *Mathematical Programming*, 39(1):93–116, 1987.
- [4] M. O. E. Zeng. Nonnegative least square - a new look into sage data. In *Proceedings of CSB'09: the LSS Computational Systems Bioinformatics Conference*, Stanford, CA, 2009.
- [5] H. Lanteri, M. Roche, O. Cuevas, and C. Aime. A general method to devise maximum-likelihood signal restoration multiplicative algorithms with non-negativity constraints. *Signal Processing*, 81(5):945–974, 2001.
- [6] D. Lee and H. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.
- [7] D. Lee and H. Seung. Algorithms for non-negative matrix factorization. *Advances in neural information processing systems*, pages 556–562, 2001.
- [8] C. Lin. Projected gradient methods for nonnegative matrix factorization. *Neural Computation*, 19(10):2756–2779, 2007.
- [9] J. Llacer and J. Nunez. Iterative maximum likelihood estimator and Bayesian algorithms for image reconstruction in astronomy. *The Restoration of HST Images and Spectra*, Baltimore, 1990.

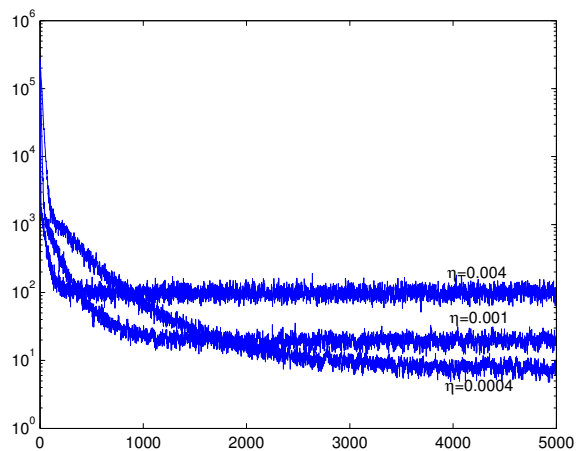


Figure 3: Evolution of mean-square error under Poisson hypothesis, for several step size values and the system of reference described in Table 1.