Data-Selective Conjugate Gradient Algorithm

Paulo S. R. Diniz, Marcele O. K. Mendonça, Jonathas O. Ferreira
Signals, Multimedia, and Telecommunications Lab.
Universidade Federal do Rio de Janeiro
DEL/Poli & PEE/COPPE/UFRJ
P.O. Box 68504,
Rio de Janeiro, RJ, 21941-972, Brazil,
{diniz, marcele.kuhfuss, jonathas.ferreira}@smt.ufrj.br

Tadeu N. Ferreira
Fluminense Federal University
Engineering School
R. Passo da Patria, 156, room E-406
24210-240, Niteroi, RJ, Brazil,
tadeu_ferreira@id.uff.br

***Abstract*—The conjugate gradient (CG) adaptive filtering algorithm is an alternative to the more widely used Recursive Least Squares (RLS) and Least Mean Square (LMS) algorithms, where the former requires more computations, and the latter leads to slower convergence. In recent years, some adaptive filtering algorithms have been equipped with data selection mechanism to classify if the data currently available consists of an outlier or if it brings about enough innovation. In both cases the data could be discarded avoiding extra computation and performance degradation. This paper proposes a data selection strategy to the CG algorithm and verifies its effectiveness in simulations utilizing synthetic and real data.**

I. Introduction

The conjugate gradient optimization algorithms aim at minimizing approximately quadratic functions by reducing the cost function through line searches in linear independent vector directions [1]-[2]-[7]-[12]-[13]-[14]. Typically online estimation demands fast convergence, low computational complexity, as well as small misadjustment. In highly correlated input signal situations, the RLS algorithms are known to provide faster convergence than the LMS-like algorithms, but the computational cost of RLS might be considered high [8]. An underrated compromise solution is Conjugate Gradient (CG) method [1]. When CG is adapted to streaming adaptive filtering, it attains fast convergence and small misadjustment in comparison to the RLS algorithm [3]-[4]. The CG algorithm does not require matrix inversion, which is another attractive feature. A special case of CG can arise from the Majorize-Minimize Subspace algorithm [5] by restricting the minimization space to the gradient subspace spanned by a small number of vectors, which are the conjugate directions. In this work, we propose the DS CG algorithm which incorporates only innovative data to the adaptation process, avoiding the use of outliers and non-innovative information. The data selection is achieved by prescribing a probability of update that establishes the threshold level to classify the data quality. This parameter is utilized in the MSE inherent to the CG algorithm. A data selective (DS) scheme based on set-membership aproach was previously proposed in [6]. Nevertheless, unlike the set-membership adaptive filtering algorithms which provide a set as solution [8]-[9]-[15], the proposed DS-CG algorithm provides a point solution at each iteration along the line of [10].

The structure of the paper is as follows. Section 2 reviews the online CG adaptive filtering algorithm and discusses how to estimate the MSE after convergence. Section 3 shows how to configure a data-selection strategy by prescribing a probability of update to determine the error thresholds defining the data innovation. Section 4 presents simulation results to corroborate the validity of proposed strategy. Section 5 includes some concluding remarks.

II. The Online Conjugate Gradient

In the adaptive filtering theory, the general objective function of the coefficients aims at minimizing the following cost function:

$$\min \frac{1}{2}\mathbf{w}(k)^T\mathbf{R}\mathbf{w}(k) - \mathbf{p}^T\mathbf{w}(k) \tag{1}$$

in which $\mathbf{R}$ is the $N \times N$ autocorrelation matrix of the input signal, $\mathbf{p}$ is the cross-correlation between the input and reference signals, and $\mathbf{w}(k)$ is the adaptive coefficient vector. Finding the solution of equation (1) corresponds to solve the linear equation:

$$\mathbf{R}\mathbf{w}(k) = \mathbf{p} \tag{2}$$

in which we can apply the CG method. In this method, the coefficient vector can be obtained through equation (3) by a linear combination of the directions $\mathbf{c}(i)$ for $i = 0, \ldots, N-1$ which present $\mathbf{R}$-conjugacy, that is, $\mathbf{c}^T(i)\mathbf{R}\mathbf{c}(j) = 0$, for all $i \neq j$.

$$\mathbf{w}_\text{o} = \sum_{i=0}^{N-1} \alpha(i)\mathbf{c}(i) \tag{3}$$

By premultiplying equation (3) by $\mathbf{c}^T(k)\mathbf{R}$, one can achieve the expression for the constant α in the kth iteration :

$$\alpha(k) = \frac{\mathbf{c}^T(k)\mathbf{p}}{\mathbf{c}^T(k)\mathbf{R}\mathbf{c}(k)} \tag{4}$$

due to conjugate definition and by replacing $\mathbf{R}\mathbf{w}_\text{o} = \mathbf{p}$.

Equation (3) can be evaluated as an iterative process in which a portion $\alpha(k)\mathbf{c}(k)$ is added at the kth step. In this way, we can express the coefficient update as follows:

$$\mathbf{w}(k) = \mathbf{w}(k-1) + \alpha(k)\mathbf{c}(k) \tag{5}$$

ISBN 978-90-827970-1-5 © EURASIP 2018

As mentioned in [11], after N steps the CG algorithm yields convergence, that is, $\mathbf{w}(N) = \mathbf{w}_\mathrm{o}$, so that we can reach equation (6) from (5).

$$\begin{aligned}\mathbf{w}(N) - \mathbf{w}(0) &= \mathbf{w}_\mathrm{o} - \mathbf{w}(0)\\ &= \alpha(0)\mathbf{c}(0) + \ldots + \alpha(N-1)\mathbf{c}(N-1)\end{aligned} \tag{6}$$

In a similar way, one can premultiply equation (6) by $\mathbf{c}^T(k)\mathbf{R}$ and after some manipulations it is possible to reach another expression for $\alpha(k)$ in equation (7). Observe that such an expression can be evaluated without calculating the cross-correlation vector $\mathbf{p}$, resulting in less computational error.

$$\alpha(k) = \frac{\mathbf{c}^T(k)\mathbf{g}(k)}{\mathbf{c}^T(k)\mathbf{R}\mathbf{c}(k)} \tag{7}$$

In the above equation, $\mathbf{g}(k) = \mathbf{p} - \mathbf{R}\mathbf{w}(k)$ is the negative gradient of the objective function and can be evaluated as:

$$\mathbf{g}(k) = \mathbf{g}(k-1) - \alpha(k)\mathbf{R}\mathbf{c}(k) \tag{8}$$

Thus, the next conjugate direction $\mathbf{c}(k+1)$ can be obtained as the current negative gradient $\mathbf{g}(k)$ corrected by a term comprising a linear combination of the previous direction vectors:

$$\mathbf{c}(k+1) = \mathbf{g}(k) + \beta(k)\mathbf{c}(k) \tag{9}$$

in which the constant $\beta(k)$ is calculated as in equation (10) in order to guarantee $\mathbf{R}$-conjugacy and improve performance as well.

$$\beta(k) = \frac{(\mathbf{g}(k) - \mathbf{g}(k-1))^T\mathbf{g}(k)}{\mathbf{g}^T(k-1)\mathbf{g}(k-1)} \tag{10}$$

As observed in [4], the estimation of the matrix $\mathbf{R}$ and vector $\mathbf{p}$ can be both computed using the exponentially decaying window, giving rise to equations (11) and (12), respectively.

$$\mathbf{R}(k) = \lambda\mathbf{R}(k-1) + \mathbf{x}(k)\mathbf{x}^T(k) \tag{11}$$

$$\mathbf{p}(k) = \lambda\mathbf{p}(k-1) + d(k)\mathbf{x}(k) \tag{12}$$

Both estimations are also employed in RLS algorithm where λ represents a forgetting factor.

By applying the line search method as done in [4], a new more attractive expression for $\alpha(k)$ can be achieved:

$$\alpha(k) = \eta\frac{\mathbf{c}^T(k)\mathbf{g}(k-1)}{\mathbf{c}^T(k)\mathbf{R}\mathbf{c}(k)} \tag{13}$$

with $(\lambda - 0.5) \leq \eta \leq \lambda$ to assure convergence. From equations (5), (11) and (12), we can obtain another expression for the negative gradient $\mathbf{g}(k)$:

$$\begin{aligned}\mathbf{g}(k) &= \mathbf{p}(k) - \mathbf{R}(k)\mathbf{w}(k)\\ &= \lambda\mathbf{p}(k-1) + d(k)\mathbf{x}(k)\\ &\quad -[\lambda\mathbf{R}(k-1) + \mathbf{x}(k)\mathbf{x}^T(k)][\mathbf{w}(k-1) + \alpha(k)\mathbf{c}(k)]\\ &= \lambda\mathbf{g}(k-1) - \alpha(k)\mathbf{R}(k)\mathbf{c}(k) + \mathbf{x}(k)e(k)\end{aligned} \tag{14}$$

where $e(k) = d(k) - \mathbf{x}^T(k)\mathbf{w}(k-1)$.

III. Data Selection Strategy

In many applications based on adaptive Filtering, a certain MSE level can be acceptable, so that updating the filter coefficients 100% of time may be avoided. Thus, only the suitable data under some necessary conditions leads to an update of the filter coefficients, reducing the computational cost. Such a method presented as Data Selection in [10] will be applied here in order to develop the Data Selective Conjugate Gradient (DSCG) algorithm.

In both system identification and prediction problems, which are analyzed in this work, we can formulate the filter output signal as:

$$y(k) = \mathbf{w}^T(k)\mathbf{x}(k) \tag{15}$$

where $\mathbf{x}(k) = [x_0(k)\ x_1(k) \ldots x_{N-1}(k)]^T$ is the input signal and $\mathbf{w}(k) = [w_0(k)\ w_1(k) \ldots w_{N-1}(k)]$ is the filter coefficients. From the desired signal $d(k)$, one can compute the error $e(k)$ as

$$e(k) = d(k) - \mathbf{w}(k)^T\mathbf{x}(k) = d(k) - y(k) \tag{16}$$

where $d(k)$ will be different for each application.

In the system identification application, the desired signal is the output of the unknown system and can be formulated as

$$d(k) = \mathbf{w}_\mathrm{o}{}^T(k)\mathbf{x}(k) + n(k) \tag{17}$$

where $\mathbf{w_o}(k)$ is the optimal coefficient and $n(k)$ is AWGN with zero mean and variance σ_n^2.

By substituting equation (17) in (16), the MSE can be expressed as:

$$\begin{aligned}\xi(k) = \mathbb{E}[e^2(k)] &= \mathbb{E}[n^2(k)] - 2\mathbb{E}[n(k)\Delta\mathbf{w}^T(k)\mathbf{x}(k)]\\ &+ \mathbb{E}[\Delta\mathbf{w}^T(k)\mathbf{x}(k)\mathbf{x}^T(k)\Delta\mathbf{w}(k)]\end{aligned} \tag{18}$$

where we define $\Delta\mathbf{w}(k) = \mathbf{w}(k) - \mathbf{w}_o$. As the noise and coefficients are uncorrelated, the second term in (18) is zero.

Thus obtaining the following expression for the MSE

$$\xi(k) = \sigma_n^2 + \xi_\mathrm{exc}(k) \tag{19}$$

where $\xi_\mathrm{exc}(k)$ is the excess MSE and $\mathbb{E}[n^2(k)] = \sigma_n^2$.

By knowing that $\mathbb{E}[e(k)] = 0$, the MSE expression coincides with the variance of the error signal:

$$\sigma_e^2 = \mathbb{E}[e^2(k)] = \xi(k) = (1+\rho)\sigma_n^2 \tag{20}$$

in which the excess MSE is rewritten as $\rho\sigma_n^2$ in order to isolate σ_n^2. The expression of ρ depends on the adaptive algorithm and must reflect how often the algorithm updates the coefficients in steady-state, i.e, after the learning process in the beginning.

Hence, one can define the desired probability of update, in steady-state as:

$$P_\mathrm{up} = P\left[\frac{|e(k)|}{\sigma_n} > \sqrt{\tau}\right] - P\left[\frac{|e(k)|}{\sigma_n} > \sqrt{\tau_\mathrm{max}}\right] \tag{21}$$

in which, the coefficient update is only performed when:

$$\sqrt{\tau} < \frac{|e(k)|}{\sigma_n} \leq \sqrt{\tau_\mathrm{max}} \tag{22}$$

ISBN 978-90-827970-1-5 © EURASIP 2018

and will be analyzed hereafter.

Considering that the error signal has normal distribution, the probability of update is modeled as:

$$P_{\text{up}} = 2Q\left(\frac{\sigma_n\sqrt{\tau}}{\sigma_e}\right) - 2Q\left(\frac{\sigma_n\sqrt{\tau_{\max}}}{\sigma_e}\right) \tag{23}$$

where Q is the complementary Gaussian cumulative distribution function defined as $Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^\infty \exp^{-t^2/2} dt$.

Not taking into consideration the effect of $\tau_{\max}$ in equation (23), the parameter τ is obtained as:

$$\sqrt{\tau} = \sqrt{(1+\rho)}Q^{-1}\left(\frac{P_{\text{up}}}{2}\right) \tag{24}$$

in which $\sqrt{1+\rho} = \sigma_e/\sigma_n$ follows from equation (20).

Since the filter coefficients, in steady-state, satisfy: $\mathbf{w}(k) \approx \mathbf{w}(k-1)$, one can utilize equation (5), to obtain $\alpha(k)\mathbf{c}(k) \approx \mathbf{0}$. Hence, if we premultiply both sides of equation (13) by $\alpha(k)$ it is possible to conclude that $\alpha(k) \approx 0$. As $\alpha(k) \approx 0$, it follows that $\mathbf{g}(k) \approx \mathbf{g}(k-1)$ in equation (8), which implies that $\beta(k) \approx 0$ in (10). Thus, by using equation (9), one can obtain $\mathbf{c}(k+1) \approx \mathbf{g}(k)$. By substituting $\mathbf{c}(k) \approx \mathbf{g}(k-1)$ in equation (13), we obtain $\mathbf{g}(k) \approx 0$. As $\mathbf{g}(k) = \mathbf{p}(k) - \mathbf{R}(k)\mathbf{w}(k)$ in equation (14) and $\mathbf{g}(k) \approx 0$, it is possible to infer that $\mathbf{w}(k) \approx \mathbf{R}^{-1}(k)\mathbf{p}(k)$, which is the same obtained in RLS algorithm. Finally, one can conclude that the CG and RLS algorithms are equivalent in steady-state [8].

According to the discussion above, both CG and RLS algorithms estimate matrix $\mathbf{R}$ and vector $\mathbf{p}$ by using equations (11) and (12). Therefore, in steady-state, the excess MSE present in CG and RLS is equivalent, with derivation detailed in [8], giving rise to the expression of ρ:

$$\rho = \frac{\xi_{\text{exc}}}{\xi_{\min}} \approx (N+1)\frac{P_{\text{up}}(1-\lambda)}{2 - P_{\text{up}}(1-\lambda)}. \tag{25}$$

The derivation is partially included in appendix A.

In the prediction case, which will be properly analyzed in Section 4, $\sigma_e^2 \approx \xi_{\min}$ which leads to the substitution of $\rho = 0$ in equation (24), indicated in the DSCG algorithm outlined in Table I.

As can be observed in Table I, the error signal $e(k)$ is calculated before the coefficient update, avoiding the extra computation related to equations (11), (5), (14), (10) and (9). At each iteration of the data selection algorithm, the error signal $e(k)$ is available so that it is possible to obtain an instantaneous estimate of the MSE as $\xi(k) = |e(k)|^2$. Then according to equation (22) we can propose a data selection strategy by comparing the current MSE level with $\tau(k)\xi_{\min}$, so that if:

$$\xi(k) \leq \tau(k)\xi_{\min} \tag{26}$$

we can consider that the current data does not bring much innovation, and consequently the filter coefficients update is unnecessary. On the other hand, for

$$\xi(k) > \tau_{\max}\xi_{\min} \tag{27}$$

that is, the MSE level is too large, the current data can be considered as an outlier. In this case, the coefficients are not updated and the current data are discarded.

Indeed, only when we have $\tau(k)\xi_{\min} < \xi(k) \leq \tau_{\max}\xi_{\min}$ the filter coefficients will be updated.

We also adopted a method for choosing the $\tau_{\max}$ in order to eliminate some possible outliers: we trained the first 20% of the data without $\tau_{\max}$ and in the following data we calculated a

$$\sqrt{\tau_{\max}} = \text{mean}(|e|/\sigma_e) + 3 * \text{var}(|e|/\sigma_e), \tag{28}$$

since 99.9% of the observations will be below the chosen $\sqrt{\tau_{\max}}$ and the rest we considered an outlier.

TABLE I
DATA SELECTIVE CONJUGATE GRADIENT ALGORITHM

```
DSCG algorithm
Initialization
$\lambda, \eta$ with $(\lambda - 0.5) \leq \eta \leq \lambda$
$\mathbf{w}(0) =$ random vectors or zero vectors
$R_0 = \mathbf{I}$
$\mathbf{g}(0) = \mathbf{c}(1) = zeros(N+1, 1)$
$\gamma =$ small constant for regularization
prescribe $P_{\text{up}}$, and choose $\tau_{\max}$
$\sqrt{\tau} = \sqrt{(1+\rho)}Q^{-1}(\frac{P_{\text{up}}}{2})$
For prediction use $\rho = 0$
For system identification use $\rho = (N+1)\frac{P_{up}(1-\lambda)}{2-P_{up}(1-\lambda)}$.
Do for $k > 0$
acquire $\mathbf{x}(k)$ and $d(k)$
$e(k) = d(k) - \mathbf{w}^T(k-1)x(k)$
$\delta(k) = \begin{cases} 0, & \text{if } -\sqrt{\tau} \leq \frac{|e(k)|}{\sigma_e} \leq \sqrt{\tau} \\ 0, & \text{if } \frac{|e(k)|}{\sigma_e} \geq \sqrt{\tau_{\max}} \\ 1, & \text{otherwise} \end{cases}$
if $\delta(k) = 0$
 $\mathbf{w}(k) = \mathbf{w}(k-1)$
if $\frac{|e(k)|}{\sigma_e} > \sqrt{\tau_{\max}}$
 $e(k) = 0$
 $d(k) = 0$
end if
else
$\mathbf{R}(k) = \lambda\mathbf{R}(k-1) + \mathbf{x}(k)\mathbf{x}^T(k)$
$\alpha(k) = \eta\frac{\mathbf{c}^T(k)\mathbf{g}(k-1)}{[\mathbf{c}^T(k)\mathbf{R}(k)\mathbf{c}(k)+\gamma]}$
$\mathbf{w}(k) = \mathbf{w}(k-1) + \alpha(k)\mathbf{c}(k)$
$\mathbf{g}(k) = \lambda\mathbf{g}(k-1) - \alpha(k)\mathbf{R}(k)\mathbf{c}(k) + \mathbf{x}(k)e(k)$
$\beta(k) = \frac{[\mathbf{g}(k)-\mathbf{g}(k-1)]^T\mathbf{g}(k)}{[\mathbf{g}^T(k-1)\mathbf{g}(k-1)+\gamma]}$
$\mathbf{c}(k+1) = \mathbf{g}(k) + \beta(k)\mathbf{c}(k)$
end if
```

IV. SIMULATION RESULTS

In this section, we present some simulations with generated and real data in which the proposed algorithm in Table 1 is applied in both prediction and system identification problems for $\lambda = 0.98$ and $\eta = 0.48$. In the simulations, the probability of update P_{up} is varied from 0 to 100% in order to verify the data selection power. All the simulations presented in this section are obtained by the average of 200 independent Monte Carlo runs.

A. Simulation 1: Generated Data

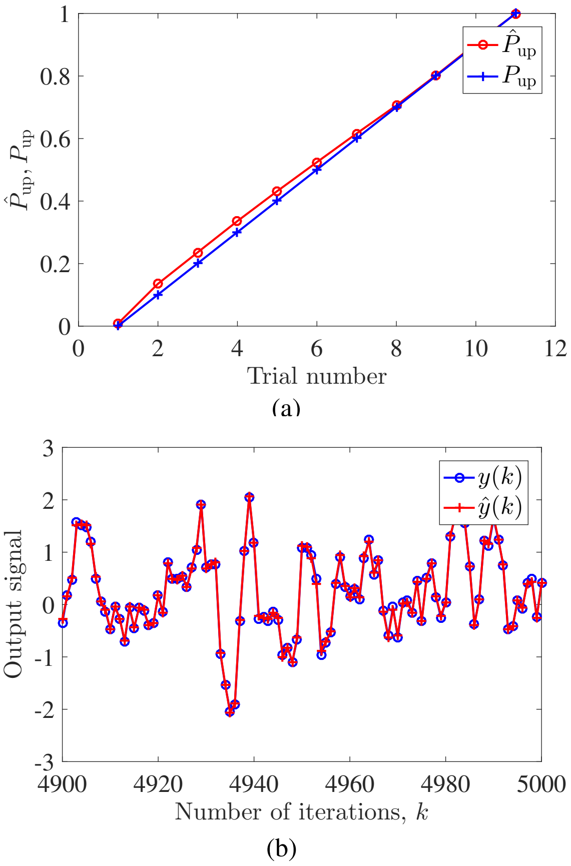
Our main goal in this simulation is to identify an unknown system, that is, a channel impulse response, described as:

$$\mathbf{h} = [0.1\ 0.3\ 0\ -0.2\ -0.4\ -0.7\ -0.4\ -0.2]^T \tag{29}$$

A Gaussian noise, with zero mean and variance $\sigma_n^2 = 10^{-3}$, is added to the unknown system output so that $d(k) = \mathbf{h}^T\mathbf{x}(k)+$

ISBN 978-90-827970-1-5 © EURASIP 2018

$n(k)$. The entries of input-signal vector $\mathbf{x}(k)$ are obtained from a random Gaussian variable with zero mean and unit variance. Indeed, as the probability of update P_{up} grows from 0 to 100%, one can verify in Figure 1a that the obtained $\hat{P}_{\text{up}}$ follows the prescription closely. The filter order $N = 7$ was chosen to guarantee the convergence of the filter coefficients to the optimal coefficients, detailed in equation (29). The output of the unknown system $y(k)$ and the filter output $\hat{y}(k)$ are depicted in Figure 1b, only for $\hat{P}_{\text{up}} = 0.9\%$ given that for higher prescribed P_{up} the results are very similar. The results shown in Figure 1 were obtained without the presence of outliers. Simulations performed including outliers added to the desired signal, had shown a decrease in the probability of update $\hat{P}_{\text{up}}$ when outliers amount 1% of the samples.



(a)

(b)

Fig. 1. Simulation 1: Comparison between (a) desired P_{up} and achieved $\hat{P}_{\text{up}}$ and (b) $y(k)$ and $\hat{y}(k)$ for = $\hat{P}_{\text{up}} = 0.9\%$.

B. Simulation 2: Real Data

In this simulation, we use the algorithm in Table 1 to predict the temperature from a significantly polluted area in Italy. As in the previous simulation, the probability of update varies from 0 to 100%. The temperature samples were obtained from a data-set provided by University of California at Irvine and recorded between March 2004 and April 2005. At each day, the samples were estimated with intervals of one hour, so that, the full stream of data is composed by one year of estimations. As described before, we need to estimate the predictor error variance σ_e^2 in the prediction problem to verify if a new data is innovative. In order to do so, one can follow the procedure described in [8]. In the prediction problem the desired signal is $x(k)$ whereas the input signal at the filter corresponds a delayed version of $x(k)$. Thus, the MSE can be written as:

$$\xi(k) = E[(x(k+L) - \mathbf{w}^T\mathbf{x}(k))^2] \tag{30}$$

giving rise to a expression for the minimum MSE:

$$\xi_{\min}(k) = r(0) - \mathbf{w}_{\text{o}}^T \begin{bmatrix} r(L) \\ r(L+1) \\ \vdots \\ r(L+N) \end{bmatrix} \tag{31}$$

where $\mathbf{w}_{\text{o}}$ is the optimal coefficients of the predictor and $r(l) = E[x(k)x(k-l)]$. Inspired by equation (31), we can obtain an estimate of σ_e^2 at iteration k by replacing $\mathbf{w}_{\text{o}}^T$ by $\mathbf{w}(k)$ which are the coefficients of the adaptive filter at iteration k. We can estimate $r(l)$ through $r(l) = \theta r(l-1) + (1-\theta)x(k)x(k-l)$ in which θ is a forgetting factor. The value $\theta = 0.99$ was utilized. Since we are using a real dataset, outliers may be present, hence we obtain $\tau_{\max}$ as in (28). In this way, the algorithm performs an estimate of σ_e^2, the estimated prediction error, which is obtained whenever the filter coefficients are updated. In order to verify the minimum MSE, the filter order N is varied. As can be seen in Figure 2a, it was possible to achieve a relatively low MSE level for $N = 20$, around 0.83. Figure 2b compares the average number of iterations where updates occurred $\hat{P}_{\text{up}}$ with the prescribed P_{up}, for $N = 20$ and $L = 1$ sample ahead. As can be observed, the measured $\hat{P}_{\text{up}}$ fits well P_{up}.

In the Figure 3, we illustrate the original and predicted signals between iterations 8800 and 8900. For the rate of updating as small as $\hat{P}_{\text{up}} = 7.59\%$, we already noticed an acceptable prediction performance. For rates higher than 0.173 no improvement on the prediction performance was observed, leading us to conclude that for this particular measured data only 17.31% of the data need to be stored.

V. Concluding Remarks

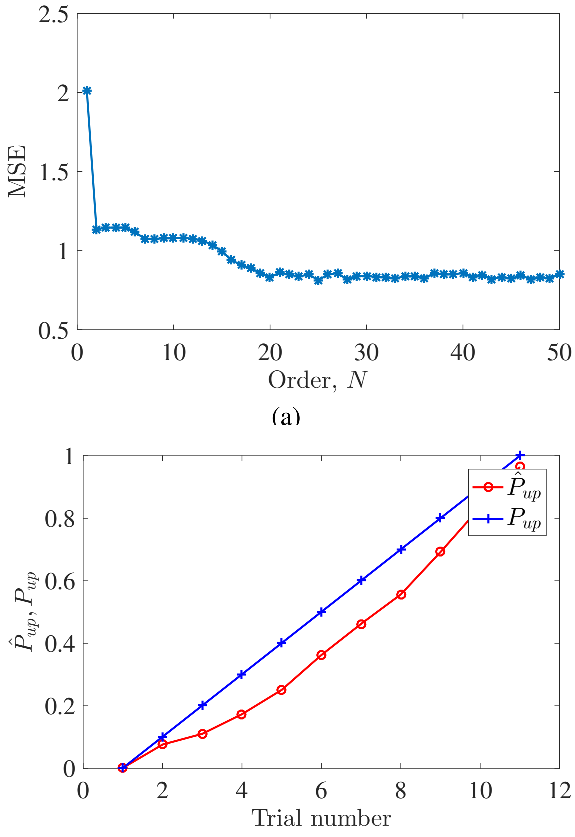
This paper proposes the data-selective conjugate-gradient algorithm enabling a prescribed probability of updating. With the data-selection strategy, it is possible to discard the least innovative information acquired without impairing the quality of the estimation. It is also possible to reject outliers which can affect the learning process temporarily. Simulation results, employing synthetic and measured data, show the benefits of utilizing the DS-CG algorithm.

APPENDIX A

Since equations (11) and (12) are used to estimate $\mathbf{R}$ and vector $\mathbf{p}$ and $\mathbf{w}(k) = \mathbf{R}^{-1}(k)\mathbf{p}(k) = \mathbf{R}\mathbf{p} = \mathbf{w}_o$ when k tends to infinity, we will follow the same steps of the RLS algorithm, see [8], pp. 226,

$$\Delta\mathbf{w}(k) = \lambda\mathbf{S}(k)\mathbf{R}(k-1)\Delta\mathbf{w}(k-1) + \mathbf{S}(k)\mathbf{x}(k)e_o(k) \tag{32}$$

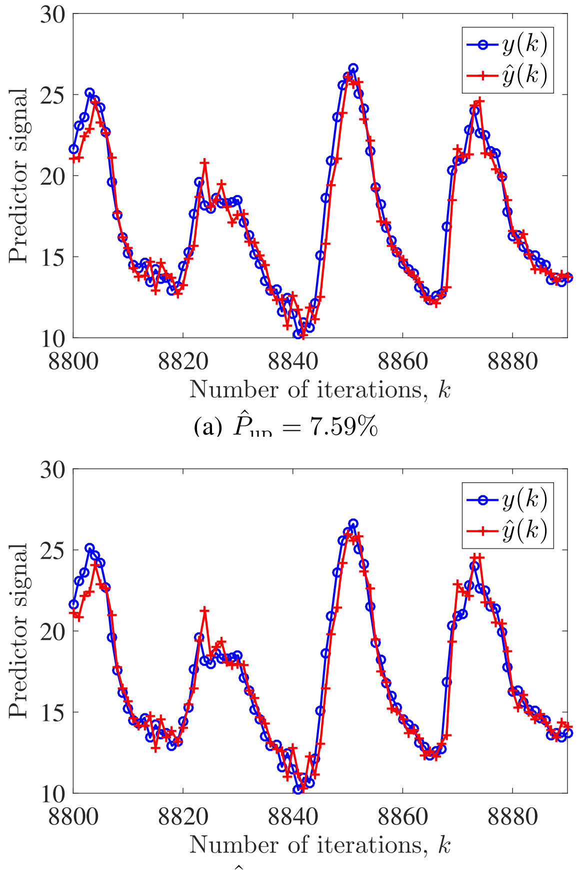
where $\Delta\mathbf{w}(k) = \mathbf{w}(k) - \mathbf{w}_o$ and $\mathbf{S}(k) = \mathbf{R}^{-1}(k)$. Considering that the DSCG updates only when there is relevant



(a)

(b)

Fig. 2. Simulation 2: (a) MSE in steady-state as the filter order N increases and (b) comparison between desired $P_{\rm up}$ and achieved $\hat{P}_{\rm up}$.



(a) $\hat{P}_{\rm up} = 7.59\%$

(b) $\hat{P}_{\rm up} = 17.31\%$

Fig. 3. Simulation 2: Comparison between $y(k)$ and $\hat{y}(k)$ for two values of $\hat{P}_{\rm up}$

information, we can apply an analytical model containing the desired probability of update $P_{\rm up}$:

$$\begin{aligned}\Delta\mathbf{w}(k) \quad &= \quad \Delta\mathbf{w}(k-1) \\ &\quad +P_{\rm up}[\lambda\mathbf{S}(k)\mathbf{R}(k-1)-\mathbf{I}]\Delta\mathbf{w}(k-1) \\ &\quad +P_{\rm up}\mathbf{S}(k)\mathbf{x}(k)e_o(k). \end{aligned} \tag{33}$$

Using the equation above and a similar derivation of the excess MSE in [8], pp. 226-229, we obtain the equation (25).

REFERENCES

[1] A. Antoniou and W.-S. Lu, *Practical Optimization: Algorithms and Engineering Applications*, Springer, New York, NY, 2007.

[2] R. Fletcher, "Practical methods of optimization," 2nd Edition, John Wiley & Sons, Cornwall, UK, 2013.

[3] G. K. Boray and M. D. Srinath, "Conjugate gradient techniques for adaptive filtering," *IEEE Trans. on Circuits and Systems-I: Fundamental Theory and Applications*, Vol. 39, pp. 1-10, n. 1, Jan. 1992.

[4] P. S. Chang and A. N. Willson, Jr., "Analysis of conjugate gradient algorithms for adaptive filtering," *IEEE Trans. on Circuits and Systems-I: Fundamental Theory and Applications*, Vol. 48, pp. 409-418, n. 2, Feb. 2000.

[5] E. Chouzenoux and J.-C. Pesquet., "A Stochastic Majorize-Minimize Subspace Algorithm for Online Penalized Least Squares Estimation, "*IEEE Transactions on Signal Processing*, Vol. 65, No. 18, pages 4770-4783, 2017.

[6] J. A. Apolinário, S. Werner and P. S. R. Diniz, "Conjugate Gradient Algorithm with Data Selective Updating," *Brazilian Telecommunications Symposium-SBT*, 2001.

[7] J. A. Apolinário, Jr., M. L. R. de Campos, and C. P. Bernal O., "The constrained conjugate gradient algorithms," *IEEE Trans. Signal Processing Letters*, Vol. 7, n. 12, pp. 351-354, Dec. 2000.

[8] P. S. R. Diniz, *Adaptive Filtering: Algorithms and Practical Implementation*, Springer, New York, NY, 4th Edition, 2013.

[9] P. S. R. Diniz, "Convergence performance of the simplified set-membership affine projection algorithm," *Circuits Syst. Signal Processing*, Birkäuser, vol. 30, pp. 439-462, April 2011.

[10] P. S. R. Diniz, *Data selective adaptive filtering*, submitted for publication, 2017.

[11] D. G. Luenberger, "Linear and Nonlinear Programming," 2nd. Edition, Reading, MA, Addison-Wesley, 1984.

[12] A. W. Hull and W. K. Jenkins, "Preconditioned conjugate gradient methods for adaptive filtering,"*in Proc. IEEE Int. Symp. Circuits Syst.*, Singapore, June 1991, pp. 540-543.

[13] Z. Chen, H. Li and M. Rangaswamy, "Conjugate gradient adaptive matched filter", *IEEE Trans. on Aerospace and Electronic Systems*, Vol. 51, n. 1, pp 178-191, Jan. 2015.

[14] M. Zhang, A. Zhang and Q. Yang, "Robust Adaptive Beamforming Based on Conjugate Gradient Algorithms", *IEEE Trans. on Signal Processing*, Vol. 4, n. 11, pp 6046-6057, Nov. 2016.

[15] L. Wang and R. C. de Lamare, "Set-membership constrained conjugate gradient adaptive filtering algorithm and its application to beamforming," *in Proc. of the 7th International Conference on Digital Signal Processing*, Corfu, Greece, July 2011, pp. 1–5.

[16] UC Irvine, "Air Quality Data Set" , Machine Learning Reposiory, [Online]. Available: https://archive.ics.uci.edu/ml/datasets/Air+quality, accessed on Sept. 2017.