EUSIPCO 2013 1569743609

VARIATIONAL BAYESIAN SPARSE ADAPTIVE FILTERING USING A GAUSS-SEIDEL RECURSIVE APPROACH

Konstantinos E. Themelis, Athanasios A. Rontogiannis, and Konstantinos D. Koutroumbas

Institute for Astronomy, Astrophysics, Space Applications and Remote Sensing
National Observatory of Athens, 152 36, Athens, Greece
{themelis,tronto,koutroum}@noa.gr

ABSTRACT

In this work, we present a new sparse adaptive filtering algorithm following a variational Bayesian approach. First, sparsity is imposed by assigning Laplace priors to the filter parameters through a suitably defined hierarchical Bayesian model. Then, a variational Bayesian inference method is presented, which is appropriate for batch processing. In order to introduce adaptivity the Gauss-Seidel iterative scheme is properly embedded in our method. The proposed algorithm is fully automatic and is computationally efficient despite its Bayesian origin. Experimental results show that the algorithm converges to sparse solutions and exhibits superior estimation performance compared to related state-of-the-art schemes.

Index Terms— Adaptive filtering, variational inference, Bayesian data analysis, Gauss-Seidel method.

1. INTRODUCTION

Adaptive filtering is considered a cornerstone in the modern signal processing research field. Owing to its immense number of applications, it has been a major research area in earlier decades. Nowadays, advances in the realm of compressive sensing have revived the interest for the development of novel adaptive filtering techniques that are tailored to the estimation of *sparse* signals.

Many deterministic algorithms have recently been proposed to address the problem of sparse adaptive filtering. Capitalizing on the celebrated Lasso estimator, the sparse analogue of the recursive least squares (RLS) algorithm has been proposed in [1], where an expectation-maximization (EM) algorithm is used to solve the time-varying optimization problem. In [2] the LS cost function is also regularized with the ℓ_1-norm of the parameter vector and coordinate descent solvers are developed to estimate the sparse vector. Greedy RLS-type algorithms have also received considerable attention in the adaptive filtering literature, e.g. [3]. Moreover, least mean square (LMS) type solvers are proposed in [4], that also incorporate the ℓ_1-norm to enhance sparsity.

The main drawback of these methods is that their estimation performance strongly depends on pre-determined or cross-validated parameter values. To overcome this impediment, statistical methods have recently been proposed for the estimation problem at hand. By adopting the Bayesian model of [5] and following a maximum a posteriori probability (MAP) estimation procedure, a regularized version of the RLS algorithm is described in [6]. A Bayesian treatment of the problem is also presented in [7], where a computationally demanding sliding window on line algorithm is proposed.

In this paper, we introduce a fast variational Bayesian inference algorithm for adaptive filtering. To promote sparsity we first define a suitable hierarchical Bayesian model that utilizes Laplace priors for the filter weight parameters, Then, a novel method for adaptive Bayesian inference is developed by combining the iterative approximate variational scheme presented in [8] with the Gauss-Seidel (GS) recursive method [9]. Due to the employment of the GS scheme, matrix inversions are avoided and each coefficient of the adaptive filter is updated separately in a computationally efficient manner. Besides that, the proposed algorithm preserves all the advantages of Bayesian methods, e.g. no need of tuning any parameters, computation of distributions of the parameters and not just point estimates, etc. Experimental results show that the proposed algorithm converges to the true sparse signal and exhibits better estimation performance than other related state-of-the-art algorithms.

Notation: Vectors are represented as boldface lowercase letters, e.g. $\mathbf{x}$, and matrices as boldface uppercase letters, e.g. $\mathbf{X}$, $(\cdot)^T$ denotes transposition, $\|\cdot\|$ stands for the standard ℓ_2-norm, $\mathcal{N}(\cdot)$ is the Gaussian distribution, $|\cdot|$ denotes the determinant of a matrix or absolute value in case of a scalar, $\Gamma(\cdot)$ is the Gamma distribution or Gamma function, $\mathrm{diag}(\mathbf{x})$ denotes a diagonal matrix whose diagonal entries are the elements of $\mathbf{x}$. Finally the i-th component of vector $\boldsymbol{\alpha}$ is denoted by a_i and the ij-th element of matrix $\mathbf{A}$ by a_{ij}.

This research is cofinanced by the European Union (European Social Fund ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) Research Funding Program THALIS: Secure wireless nonlinear communications at the physical layer.

2. PROBLEM FORMULATION

Let us consider the conventional adaptive filtering setup consisting of a transversal filter of order N and an adaptation block. Using standard notation, the input of the filter at time M is denoted by $\mathbf{x}(M) = [x(M), x(M-1), \ldots, x(M-N+1)]^T$ and the filter weight vector at time M is $\hat{\mathbf{w}}(M) = [\hat{w}_1(M), \hat{w}_2(M), \ldots, \hat{w}_N(M)]^T$. The output of the filter at every time instant M is $\psi(M) = \hat{\mathbf{w}}^T(M)\mathbf{x}(M)$, and hence, the instantaneous error of the filter is $e(M) = y(M) - \hat{\mathbf{w}}^T(M)\mathbf{x}(M)$, where $y(M)$ stands for the noisy observed data. We assume that the data generating process is described by

$$y(M) = \mathbf{w}^T\mathbf{x}(M) + \epsilon(M), \tag{1}$$

where $\mathbf{w}$ is the true parameter vector that may or may not depend on time and $\epsilon(M)$ is assumed to be zero mean Gaussian with precision β, $\epsilon(M) \sim \mathcal{N}(\epsilon(M)|0, \beta^{-1})$. The cost function typically minimized in LS filtering is an exponentially weighted sum of squared $e(i)$, i.e.,

$$\min_{\hat{\mathbf{w}}(M)} \sum_{i=1}^{M} \lambda^{M-i}|e(i)|^2, \tag{2}$$

where $0 \ll \lambda < 1$ is the forgetting factor. Let $\mathbf{X}(M)$ be a $M \times N$ data matrix whose ith row is $\mathbf{x}^T(i)$, i.e.,

$$\mathbf{X}(M) = \begin{bmatrix} \mathbf{x}^T(1) \\ \mathbf{x}^T(2) \\ \vdots \\ \mathbf{x}^T(M) \end{bmatrix}, \tag{3}$$

$\mathbf{y}(M) = [y(1), y(2), \ldots, y(M)]^T$, and $\mathbf{\Lambda}(M) = \mathrm{diag}([\lambda^{M-1}, \lambda^{M-2}, \ldots, \lambda])$. Then, the least squares optimization function (2) can be written in vector notation as

$$\min_{\hat{\mathbf{w}}(M)} \|\mathbf{\Lambda}^{1/2}(M)\mathbf{y}(M) - \mathbf{\Lambda}^{1/2}(M)\mathbf{X}(M)\hat{\mathbf{w}}(M)\|^2. \tag{4}$$

Based on the input $\mathbf{x}$ and the data y, we wish to estimate the parameter vector $\mathbf{w}$, under the assumption that it is sparse, i.e., only a few of its N parameters are nonzero. To this end, we formulate the problem in the Bayesian framework and propose a fast adaptive variational Bayesian algorithm.

3. BAYESIAN MODELING

In the following in order to simplify derivations we omit the time index M, which shall be retrieved when introducing time recursions in Section 4.1. To establish our Bayesian approach, we define the likelihood function of the observations and introduce appropriate prior distributions over the unknown parameters of the model in (1). The likelihood function can be written as

$$p(\mathbf{y}|\mathbf{w}, \beta) = (\frac{\beta}{2\pi})^{\frac{M}{2}} |\mathbf{\Lambda}|^{-\frac{1}{2}} \exp\left[-\frac{\beta}{2}\|\mathbf{\Lambda}^{\frac{1}{2}}\mathbf{y} - \mathbf{\Lambda}^{\frac{1}{2}}\mathbf{X}\mathbf{w}\|^2\right]. \tag{5}$$

First, a conjugate Gamma distribution is chosen as prior distribution for the noise precision β,

$$p(\beta|\kappa, \theta) = \Gamma(\beta|\kappa, \theta) = \frac{\theta^\kappa}{\Gamma(\kappa)}\beta^{\kappa-1}\exp\left[-\theta\beta\right], \tag{6}$$

with mean $\mathrm{E}[p(\beta|\kappa, \theta)] = \kappa/\theta$ and variance $var[p(\beta|\kappa, \theta)] = \kappa/\theta^2$. To introduce sparsity to our Bayesian model, Laplace priors are used over the weight vector $\mathbf{w}$. The Laplace distribution is equivalent to a two-levels hierarchical prior, namely a multivariate Gaussian prior with exponentially distributed variances. More specifically, the Gaussian prior of $\mathbf{w}$ is

$$p(\mathbf{w}|\boldsymbol{\gamma}, \beta) = \mathcal{N}(\mathbf{w}|0, \beta^{-1}\mathbf{\Gamma}), \tag{7}$$

where $\mathbf{\Gamma} = \mathrm{diag}(\boldsymbol{\gamma})$, with $\boldsymbol{\gamma} = [\gamma_1, \gamma_2, \ldots, \gamma_N]^T$, while a Gamma prior is selected for the vector of variances $\boldsymbol{\gamma}$, i.e.,

$$p(\boldsymbol{\gamma}|\boldsymbol{\alpha}) = \prod_{i=1}^{N} \Gamma(\gamma_i|1, \frac{\alpha_i}{2}) = \prod_{i=1}^{N} \left[\frac{\alpha_i}{2}\exp\left[-\frac{\alpha_i}{2}\gamma_i\right]\right], \tag{8}$$

where $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_N]^T$. This two-levels ((7) and (8)) hierarchical prior assignment leads to the multivariate Laplace prior,

$$\begin{aligned} p(\mathbf{w}|\boldsymbol{\alpha}) &= \prod_{i=1}^{N} p(w_i|\alpha_i) = \prod_{i=1}^{N} \int p(w_i|\gamma_i, \beta)p(\gamma_i|\alpha_i)d\gamma_i \\ &= \prod_{i=1}^{N} \frac{\sqrt{\beta\alpha_i}}{2} \exp\left[-\sqrt{\beta\alpha_i}\,|w_i|\right], \end{aligned} \tag{9}$$

whose potential in sparse signal recovery is explored in [10]. To estimate the sparsity-promoting parameters $\alpha_i, i = 1, 2, \ldots, N$ from the data, a Gamma hyperprior is used,

$$p(\alpha_i|\rho, \delta) = \Gamma(\alpha_i|\rho, \delta) = \frac{\delta^\rho}{\Gamma(\rho)}{\alpha_i}^{\rho-1}\exp\left[-\delta\alpha_i\right], \tag{10}$$

where $\rho > 0$ and $\delta > 0$ are hyperparameters. Note that both pairs of hyperparameters, κ, θ and ρ, δ, of the distributions for β and $\boldsymbol{\alpha}$ respectively are set to values close to zero, to account for non-informative Jeffrey's priors, $p(\eta) \propto \frac{1}{\eta}$.

4. BAYESIAN INFERENCE

The posterior distribution of $\mathbf{w}, \beta, \boldsymbol{\gamma}, \boldsymbol{\alpha}$ is given by

$$p\left(\mathbf{w}, \beta, \boldsymbol{\gamma}, \boldsymbol{\alpha}|\mathbf{y}\right) = \frac{p\left(\mathbf{y}|\mathbf{w}, \beta\right)p\left(\mathbf{w}|\boldsymbol{\gamma}, \beta\right)p\left(\boldsymbol{\gamma}|\boldsymbol{\alpha}\right)p\left(\boldsymbol{\alpha}\right)p\left(\beta\right)}{\int p\left(\mathbf{y}, \mathbf{w}, \beta, \boldsymbol{\gamma}, \boldsymbol{\alpha}\right)d\mathbf{w}d\boldsymbol{\gamma}d\boldsymbol{\alpha}d\beta}. \tag{11}$$

However, the computation of the exact posterior (from which an estimate of $\mathbf{w}$ can be obtained) is intractable due to the integration at the denominator. To perform Bayesian inference we choose to approximate (11) utilizing an iterative scheme similar to BI-ICE that was presented in [8]. BI-ICE can be viewed as a first-order approximation to variational Bayesian inference methods, [10]. In analogy to a Gibbs sampling approach, the posterior conditional probabilities of the individual parameters are computed one by one. However, in the resulting Bayesian inference iterative scheme, their mean values are employed instead of random samples. Due to the preservation of conjugacy in our Bayesian model, the conditional posterior distributions can be expressed in closed forms. To begin with $\mathbf{w}$, the conditional posterior distribution of the filter coefficients is easily shown to be multivariate Gaussian,

$$p(\mathbf{w}|\mathbf{y},\boldsymbol{\gamma},\boldsymbol{\alpha},\beta) = \mathcal{N}(\mathbf{w}|\boldsymbol{\mu},\boldsymbol{\Sigma}), \tag{12}$$

where

$$\boldsymbol{\mu} = \beta\boldsymbol{\Sigma}\mathbf{X}^T\boldsymbol{\Lambda}\mathbf{y}, \text{ and} \tag{13}$$

$$\boldsymbol{\Sigma} = \beta^{-1}\left(\mathbf{X}^T\boldsymbol{\Lambda}\mathbf{X} + \boldsymbol{\Gamma}^{-1}\right)^{-1}. \tag{14}$$

The posterior conditional for the precision parameter β is expressed as

$$p(\beta|\mathbf{y},\mathbf{w},\boldsymbol{\gamma},\boldsymbol{\alpha}) = \Gamma\left(\beta\,\middle|\,\frac{M}{2}+\frac{N}{2}+\kappa,\ \frac{1}{2}\left\|\boldsymbol{\Lambda}^{\frac{1}{2}}\mathbf{y}-\boldsymbol{\Lambda}^{\frac{1}{2}}\mathbf{X}\mathbf{w}\right\|^2+\theta+\frac{1}{2}\mathbf{w}^T\boldsymbol{\Gamma}^{-1}\mathbf{w}\right) \tag{15}$$

Also, straightforward computations yield that the conditional pdf of $\gamma_i, i = 1,2,\ldots,N$ given $\mathbf{y}, w_i, \alpha_i, \beta$ is the following generalized inverse Gaussian distribution

$$p(\gamma_i|\mathbf{y},w_i,\alpha_i,\beta) = \left(\frac{\alpha_i}{2\pi}\right)^{\frac{1}{2}}\gamma_i^{-\frac{1}{2}} \exp\left[-\frac{\beta w_i^2}{2\gamma_i}-\frac{\alpha_i}{2}\gamma_i+\sqrt{\beta\alpha_i}\,|w_i|\right]. \tag{16}$$

Finally, the conditional posterior of $\alpha_i, i = 1,2,\ldots,N$ given $\mathbf{y}, w_i, \gamma_i, \beta$ is expressed as

$$p(\alpha_i|\mathbf{y},w_i,\gamma_i,\beta) = \Gamma\left(\alpha_i|1+\rho,\frac{\gamma_i}{2}+\delta\right). \tag{17}$$

The expectations of the posterior conditional distributions needed in the proposed scheme are also easy to derive. The expectation of $\mathbf{w}$, which will be finally used as the estimate of $\mathbf{w}$, is already provided in (13). Similarly, according to their respective posterior distributions, the posterior expectations of β, γ_i, α_i are given by

$$\mathrm{E}\left[p(\beta|\mathbf{y},\mathbf{w},\boldsymbol{\gamma},\boldsymbol{\lambda})\right] = \frac{\frac{M}{2}+\frac{N}{2}+\kappa}{\frac{1}{2}\left\|\boldsymbol{\Lambda}^{\frac{1}{2}}\mathbf{y}-\boldsymbol{\Lambda}^{\frac{1}{2}}\mathbf{X}\mathbf{w}\right\|^2+\theta+\frac{1}{2}\mathbf{w}^T\boldsymbol{\Gamma}^{-1}\mathbf{w}} \tag{18}$$

$$\mathrm{E}\left[p(\gamma_i|\mathbf{y},w_i,\alpha_i,\beta)\right] = \sqrt{\frac{\beta w_i^2}{\alpha_i}}+\frac{1}{\alpha_i} \tag{19}$$

$$\mathrm{E}\left[p(\alpha_i|\mathbf{y},w_i,\gamma_i,\beta)\right] = \frac{1+\rho}{\frac{1}{2}\gamma_i+\delta} \tag{20}$$

Following the workings of BI-ICE, a sparse estimate, $\hat{\mathbf{w}} = \boldsymbol{\mu}$, for $\mathbf{w}$ can be retrieved by iterating among the conditional means of the parameters, as they are expressed in (13), (18), (19), and (20). However, in the adaptive filtering scenario under consideration, as new data become available one would have to solve the system of equations in (13),(14) in each time iteration, which is computationally prohibitive. In the following it is shown that a computationally efficient algorithm can be derived by updating each component of the adaptive filter $\hat{\mathbf{w}}$ separately. This is achieved by suitably adjusting to our probem the Gauss-Seidel (GS) iterative scheme for solving (13), as explained below.

4.1. Gauss-Seidel variational adaptive filtering

It is easily recognized that the posterior mean estimate $\hat{\mathbf{w}} = \boldsymbol{\mu}$ of $\mathbf{w}$ in (13) is the solution of a regularized LS problem obtained from the following system of equations,

$$\mathbf{R}\hat{\mathbf{w}} = \mathbf{z}, \tag{21}$$

where

$$\mathbf{R} = \mathbf{X}^T\boldsymbol{\Lambda}\mathbf{X}+\boldsymbol{\Gamma}^{-1}, \quad \mathbf{z} = \mathbf{X}^T\boldsymbol{\Lambda}\mathbf{y} \tag{22}$$

are the corresponding regularized autocorrelation matrix and crosscorrelation vector of the problem respectively. Let us write $\mathbf{R} = \mathbf{L}+\mathbf{U}$, where $\mathbf{L}$ is the lower triangular component of $\mathbf{R}$ including its main diagonal and $\mathbf{U}$ is its strictly upper triangular component. Then, according to the GS method, (21) can be solved iteratively as follows

$$\mathbf{L}\hat{\mathbf{w}}^{(k+1)} = \mathbf{z}-\mathbf{U}\hat{\mathbf{w}}^{(k)}, \tag{23}$$

where k is the iterations index. From (23) and due to the lower triangular form of $\mathbf{L}$, the elements of $\hat{\mathbf{w}}^{(k+1)}$ can be computed sequentially using forward substitution, [9]

$$\hat{w}_i^{(k+1)} = \frac{1}{r_{ii}}\left(z_i-\sum_{j<i}r_{ij}\hat{w}_j^{(k+1)}-\sum_{j>i}r_{ij}\hat{w}_j^{(k)}\right). \tag{24}$$

By defining $\hat{\mathbf{w}}_{\neg i}^{(k+1)} = [\hat{w}_1^{(k+1)},\ldots,\hat{w}_{i-1}^{(k+1)},\hat{w}_{i+1}^{(k)},\ldots,\hat{w}_N^{(k)}]$ and $\mathbf{r}_{\neg i}^T$ as the i-th row of $\mathbf{R}$ excluding its i-th element, then

Initialize $\beta, \boldsymbol{\gamma}, \boldsymbol{\alpha}$
for $M = 1, 2, \ldots$
- update the cross-correlation vector $\mathbf{z}(M)$ from (28)
- update the autocorrelation matrix $\mathbf{R}(M)$ from (26)
for $i = 1, 2, N$
- get $\mathbf{r}_{\neg i}(M)$ and $r_{ii}(M)$ from $\mathbf{R}(M)$ and $z_i(M)$ from $\mathbf{z}(M)$
- compute $\hat{w}_i(M)$ from (25)
end for
- $\beta(M) = \frac{\frac{1}{2}+\frac{N}{2}+\kappa}{|y(M)-\hat{\mathbf{w}}^T(M)\mathbf{x}(M)|^2+\theta+\hat{\mathbf{w}}^T(M)\boldsymbol{\Gamma}^{-1}(M)\hat{\mathbf{w}}^T(M)}$
- update $\boldsymbol{\gamma}(M)$ from (19)
- update $\boldsymbol{\alpha}(M)$ from (20)
end for

Table 1. The proposed VBS-RLS algorithm

(24) is rewritten more compactly as[1]

$$\hat{w}_i^{(k+1)} = \frac{1}{r_{ii}} \left(z_i - \mathbf{r}_{\neg i}^T \hat{\mathbf{w}}_{\neg i}^{(k+1)} \right), \tag{25}$$

for $i = 1, 2, \ldots, N$. Since $\mathbf{R}$ is symmetric and positive definite (being the inverse covariance matrix of the posterior distribution in (12) scaled by $\beta^{-1} > 0$), the GS scheme in (25) converges independent of the initial vector $\hat{\mathbf{w}}^{(0)}$, [9].

In an adaptive setting it is reasonable to consider only one cycle of the GS scheme per time iteration M of the adaptive algorithm. Moreover, from (22) the regularized inverse autocorrelation matrix $\mathbf{R}$ and the crosscorrelation vector $\mathbf{z}$ can be efficiently time-updated as follows

$$\begin{aligned} \mathbf{R}(M) &= \lambda \mathbf{R}(M-1) + \mathbf{x}(M)\mathbf{x}^T(M) \\ &\quad -\lambda \boldsymbol{\Gamma}^{-1}(M-1) + \boldsymbol{\Gamma}^{-1}(M) \end{aligned} \tag{26}$$

$$\mathbf{z}(M) = \lambda \mathbf{z}(M-1) + \mathbf{x}(M)y(M). \tag{27}$$

The proposed variational Bayesian sparse RLS (VBS-RLS) adaptive algorithm is summarized in Table 1. The VBS-RLS is a time-recursive regularized LS scheme, where the regularization matrix $\boldsymbol{\Gamma}^{-1}$ changes in time based on (19). It is exactly this variational Bayesian mechanism of producing and updating $\boldsymbol{\Gamma}^{-1}$ that imposes sparsity to our estimate, as also verified by the simulation results presented in the next section. Note from Table 1 that parameter $\beta(M)$ is computed based on the instantaneous squared error $e^2(M)$, instead of the overall squared error, as in (15). We have noticed through extensive experiments that expression (15) for β is not robust especially at the initial convergence phase of the algorithm. The overall complexity of VBS-RLS is $\mathcal{O}(N^2)$ similar to that of the classical RLS algorithm. This is very important for a method of the Bayesian family, where high computational loads are generally required. The most improtant feature of the proposed algorithm though is that in contast to other related sparse adaptive schemes (e.g., [1],[2],[4]), it is fully automatic, alleviating the need for tuning or cross-validating any parameters. This is a highly derirable characteristic in practice especially for on-line processing.

[1]It is easily shown that this procedure is equivalent to a cyclic coordinate-descent algorithm applied to the maximization of the cost function $f(\mathbf{w}) = \frac{1}{2}\left\|\boldsymbol{\Lambda}^{\frac{1}{2}}\mathbf{y} - \boldsymbol{\Lambda}^{\frac{1}{2}}\mathbf{X}\mathbf{w}\right\|^2 + \frac{1}{2}\mathbf{w}^T\boldsymbol{\Gamma}^{-1}\mathbf{w}$

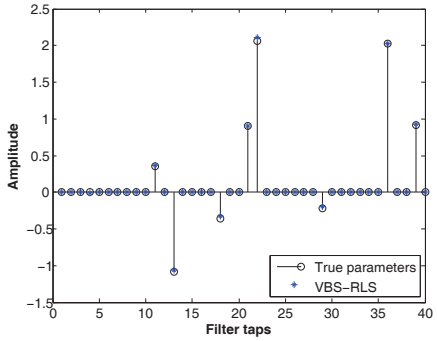

Fig. 1. Sparse estimates of VBS-RLS algorithm.

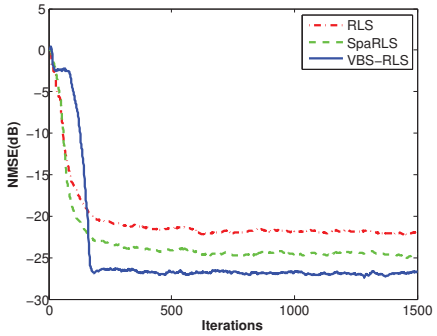

Fig. 2. Convergence and steady-state NMSE.

5. EXPERIMENTAL RESULTS

In this section we demonstrate the performance of the proposed VBS-RLS algorithm. We consider a $N = 40$ taps sparse channel, where only $T = 8$ of its components are nonzero, generated from a standard normal distribution. The positions of the nonzero channel taps are randomly selected. The input sequence consists of randomly generated ± 1 binary symbols, forming frames of length 1500. White Gaussian noise is added to the output of the channel, resulting in an SNR of 15dB. To assess the performance of the proposed scheme a total of 100 frames and channel realizations are used. The normalized mean squared error of the filter taps is employed as performance measure, defined as

$$\mathrm{NMSE(M)} = 10\log_{10}\left\{ \frac{(\mathbf{w} - \hat{\mathbf{w}}(\mathrm{M}))^{\mathrm{T}}(\mathbf{w} - \hat{\mathbf{w}}(\mathrm{M}))}{\mathbf{w}^{\mathrm{T}}\mathbf{w}} \right\}. \tag{28}$$

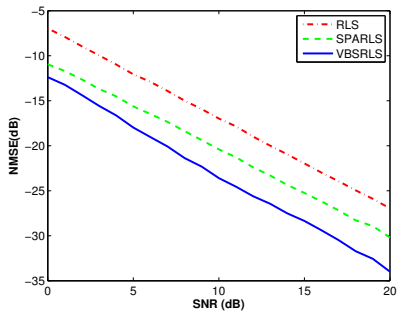

Fig. 3. NMSE versus SNR.

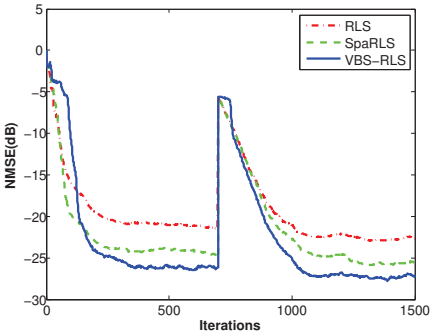

Fig. 4. Tracking of a time-varying channel.

Initially, to illustrate the sparsity of the estimates produced by VBS-RLS, its solution for a single channel realization is plotted in Fig. 1. The solution is obviously sparse and almost identical to the true sparse channel coefficients. Next, the convergence and NMSE performance of VBS-RLS is compared to those of the classical RLS algorithm, and a widely used sparse deviation of the RLS, the SpaRLS algorithm, [1]. In this venue, NMSE curves are displayed in Fig. 2, as iterations evolve in time. Although the proposed algorithm has a slightly slower initial convergence rate (most probably owing to the approximation for $\beta(M)$), it reaches faster its steady-state and achieves the lowest steady-state NMSE. It should be noted that the performance of the SpaRLS algorithm strongly depends on a thresholding parameter, which was fine-tuned through exhaustive experimentation. Steady-state NMSE curves of the three algorithms versus SNR is shown in Fig. 3. The curves have been obtained by averaging over the last 500 steady-state NMSE values of the algorithms. We observe that VBS-RLS outperforms the other two algorithms in the whole SNR range. Interestingly enough, the tracking capabilities of the VBS-RLS algorithm are displayed in Fig. 4, where a time-varying sparse channel is utilized. Specifically, a single nonzero tap of unit amplitude is added to the channel at the 700th time instant. Fig. 4 shows that the VBS-RLS easily adapts to the channel modification, and reaches its steady-state faster than RLS and SpaRLS.

6. CONCLUSIONS

A fast variational Bayesian algorithm was proposed for performing sparse adaptive filtering. The algorithm was derived by properly combining a batch sparsity promoting variational Bayes scheme with an efficient iterative GS method. The performance of the proposed algorithm was demonstrated in comparison to state-of-the-art sparse adaptive filtering algorithms. Theoretical study of the overall algorithm's convergence, as well as a mechanism for zero filter taps pruning are currently under investigation.

7. REFERENCES

[1] B. Babadi, N. Kalouptsidis, and V. Tarokh, "SPARLS: The sparse RLS algorithm," *Signal Processing, IEEE Transactions on*, vol. 58, pp. 4013 –4025, Aug. 2010.

[2] D. Angelosante, J.A. Bazerque, and G.B. Giannakis, "Online adaptive estimation of sparse signals: Where RLS meets the ℓ_1 -norm," *Signal Processing, IEEE Transactions on*, vol. 58, pp. 3436 –3447, July 2010.

[3] G. Mileounis, B. Babadi, N. Kalouptsidis, and V. Tarokh, "An adaptive greedy algorithm with application to nonlinear communications," *Signal Processing, IEEE Transactions on*, vol. 58, pp. 2998 –3007, June 2010.

[4] Y. Chen, Y. Gu, and A.O. Hero, "Sparse LMS for system identification," in *ICASSP*, April 2009, pp. 3125 –3128.

[5] M. E. Tipping, "Sparse Bayesian learning and the relevance vector machine," *J. Mach. Learn. Res.*, vol. 1, pp. 211–244, 2001.

[6] H. Koeppl, G. Kubin, and G. Paoli, "Bayesian methods for sparse RLS adaptive filters," in *Thirty-Seventh IEEE Asilomar Conference on Signals, Systems and Computers*. IEEE, 2003, vol. 2, pp. 1273–1278.

[7] T. Buchgraber, D. Shutin, and H. V. Poor, "A sliding-window online fast variational sparse bayesian learning algorithm," in *ICASSP*, May 2011, pp. 2128 –2131.

[8] K.E. Themelis, A.A. Rontogiannis, and K.D. Koutroumbas, "A novel hierarchical bayesian approach for sparse semisupervised hyperspectral unmixing," *Signal Processing, IEEE Transactions on*, vol. 60, no. 2, pp. 585 –599, Feb. 2012.

[9] G.H. Golub and C.F. Van Loan, *Matrix Computations*, Johns Hopkins University Press, 1996.

[10] M.W. Seeger and D.P. Wipf, "Variational bayesian inference techniques," *Signal Processing Magazine, IEEE*, vol. 27, no. 6, pp. 81 –91, Nov. 2010.