Incremental Reformulated Automatic Relevance Determination

Dmitriy Shutin, Sanjeev R. Kulkarni, and H. Vincent Poor

***Abstract*—In this work, the relationship between the incremental version of sparse Bayesian learning (SBL) with automatic relevance determination (ARD)—a fast marginal likelihood maximization (FMLM) algorithm—and a recently proposed reformulated ARD scheme is established. The FMLM algorithm is an incremental approach to SBL with ARD, where the corresponding objective function—the marginal likelihood—is optimized with respect to the parameters of a single component provided that the other parameters are fixed; the corresponding maximizer is computed in closed form, which enables a very efficient SBL realization. Wipf and Nagarajan have recently proposed a reformulated ARD (R-ARD) approach, which optimizes the marginal likelihood using auxiliary upper bounding functions. The resulting algorithm is then shown to correspond to a series of reweighted ℓ_1-constrained convex optimization problems. This correspondence establishes and analyzes the relationship between the FMLM and R-ARD schemes. Specifically, it is demonstrated that the FMLM algorithm realizes an incremental approach to the optimization of the R-ARD objective function. This relationship allows deriving the R-ARD pruning conditions similar to those used in the FMLM scheme to analytically detect components that are to be removed from the model, thus regulating the estimated signal sparsity and accelerating the algorithm convergence.**

***Index Terms*—Automatic relevance determination, fast marginal likelihood maximization, sparse Bayesian learning.**

I. Introduction

During the last decade research on sparse signal representations has received considerable attention [1]–[5]. With a few minor variations, the general goal of sparse reconstruction is to optimally estimate the parameters of the following canonical model:

$$\boldsymbol{t} = \boldsymbol{\Phi}\boldsymbol{w} + \boldsymbol{\xi} \tag{1}$$

where $\boldsymbol{t} \in \mathbb{R}^N$ is a vector of targets, $\boldsymbol{\Phi} = [\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_L]$ is a dictionary matrix with L columns corresponding to component vectors $\boldsymbol{\phi}_l \in \mathbb{R}^N$, $l = 1, \ldots, L$, and $\boldsymbol{w} = [w_1, \ldots, w_L]^T$ is a vector of unknown weights with only K nonzero entries, i.e., $\boldsymbol{w}$ is assumed to be K-sparse. The additive perturbation $\boldsymbol{\xi}$ is typically assumed to be a white Gaussian random vector with zero mean and covariance matrix $\tau^{-1}\boldsymbol{I}$, where τ is a noise precision parameter. Imposing constraints on the model parameters $\boldsymbol{w}$ is a key to sparse signal modeling (see, e.g., [4]).

Sparse Bayesian learning (SBL) [5]–[9] is a family of empirical Bayes techniques that find a sparse estimate of $\boldsymbol{w}$ by modeling the weights using a hierarchical prior $p(\boldsymbol{w}|\boldsymbol{\alpha})p(\boldsymbol{\alpha}) = \prod_{l=1}^{L} p(w_l|\alpha_l)p(\alpha_l)$,[1] where $p(w_l|\alpha_l)$ is a Gaussian probability density function (pdf) with zero mean and precision parameter α_l—also called the sparsity parameter— that regulates the width of this pdf. Different approaches to SBL vary mainly in the way the hyperprior $p(\alpha_l)$ is specified (see, e.g., [6] and [9]).

Automatic relevance determination (ARD) represents a class of SBL algorithms in which the hyperprior $p(\alpha_l)$ is assumed to be flat or noninformative. The weights $\boldsymbol{w}$ are estimated using the posterior pdf $p(\boldsymbol{w}|\boldsymbol{t}, \boldsymbol{\alpha}, \tau)$, which can be computed analytically; the sparsity parameters $\boldsymbol{\alpha}$ and the noise precision τ are typically found by maximizing the marginal likelihood function[2] $p(\boldsymbol{t}|\boldsymbol{\alpha}, \tau) = \int p(\boldsymbol{t}|\boldsymbol{w}, \tau)p(\boldsymbol{w}|\boldsymbol{\alpha})\mathrm{d}\boldsymbol{w}$. In the original relevance vector machine (RVM) algorithm the latter optimization is realized iteratively using the expectation-maximization (EM) algorithm [6]. Unfortunately, the EM algorithm is known to converge rather slowly and a number of improvements have been proposed in [11] and [12] to address the slow convergence of the RVM algorithm.

In [11], an efficient incremental scheme has been proposed to maximize the marginal likelihood. The authors demonstrate that the maximizer of the marginal likelihood function with respect to the sparsity parameter α_l of a single component can be computed analytically provided the sparsity parameters for the components and the parameter τ are fixed; moreover, this maximizer is shown to take either finite or infinite values. This observation has lead to two important practical consequences. First, it allows constructing an efficient algorithm, termed fast marginal likelihood maximization (FMLM), that incrementally maximizes the marginal likelihood and prunes components[3] by cycling through them in a round-robin fashion. It is important to note that the same mechanism can be used to incrementally build up the model complexity by incorporating new components in the model. Second, the analytical analysis of the conditions that detect the divergence of sparsity parameters—pruning conditions—reveals the dependency of the estimated signal sparsity on the amount of additive noise [13]; this allows for an empirical adjustment of these conditions, which has been shown to significantly accelerates the convergence rate of the sparse inference.

Another approach to maximize the marginal likelihood has been proposed in [12]. In contrast to the incremental FMLM approach, in [12] the marginal likelihood is optimized jointly over the sparsity parameters of all components via minimization of auxiliary upper bounding convex functions [14]. This approach, which the authors in [12] termed a reformulated ARD (R-ARD), exhibits a number of useful features. Specifically, the objective function of the R-ARD scheme is convex; moreover, the R-ARD solution can be computed jointly for all elements of $\boldsymbol{w}$ via a series of weighted ℓ_1-constrained least-squares optimization problems. The latter property is important from a theoretical standpoint as it effectively relates the SBL with ARD to more traditional "non-Bayesian" sparse learning methods, e.g., basis pursuit denoising and minimum ℓ_1-norm methods [3]. The downside of the scheme is a relatively high computational cost as it requires solving a series of ℓ_1-constrained optimization problems. The latter fact has motivated us

Manuscript received October 04, 2011; revised March 12, 2012; accepted May 01, 2012. Date of publication May 21, 2012; date of current version August 07, 2012. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Raviv Raich. This research was supported in part by an Erwin Schrödinger Postdoctoral Fellowship, Austrian Science Fund (FWF) Project J2909-N23, in part by the U.S. Army Research Office under Grant W911NF-07-1-0185, the U.S. Office of Naval Research under Grant N00014-09-1-0342, and in part by the Center for Science of Information (CSoI), an NSF Science and Technology Center, under grant agreement CCF-0939370.

D. Shutin is with the Department of Electrical Engineering, Princeton University Princeton, NJ 08544 USA, and also with the German Aerospace Center, Institute of Communications and Navigation, Wessling 82234, Germany (e-mail: dshutin@gmail.com; dmitriy.shutin@dlr.de).

S. R. Kulkarni is with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08544 USA (e-mail: kulkarni@princeton.edu).

H. V. Poor is with Princeton University, School of Engineering and Applied Science, Princeton, NJ 08544 USA (e-mail: poor@princeton.edu).

Color versions of one or more of the figures in this correspondence are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2012.2200478

[1]It is also possible to extend the SBL prior formulation to priors involving three layers of hierarchy (see, e.g., [9] or [10]).

[2]This is equivalent to assuming a flat hyperpriors $p(\tau)$ and $p(\alpha_l), \forall l$.

[3]An infinite value of the hyperparameter forces the posterior value of the component weight to zero.

to seek for a more efficient realization of the R-ARD scheme. Inspired by the efficiency and analytical tractability of incremental ARD solutions, we investigate if R-ARD scheme can be implemented in an incremental setting.

In this work, we report the results of these investigations. Specifically, we demonstrate that the R-ARD and FMLM algorithms are related; in fact, the FMLM algorithm realizes an incremental optimization of the R-ARD objective function. In other words, FMLM algorithm represents an alternative optimization strategy for the R-ARD objective cost function. It is not unreasonable to assume the existence of the relationship between the two schemes since both R-ARD and FMLM share a similar ARD-motivated effective cost function, yet no study has been performed to formally establish this connection. This relationship allows, on the one hand, for a better understanding of the performance of classical sparse learning schemes in the presence of noise due to the analytical tractability of the FMLM pruning conditions and their dependency on the component's signal-to-noise ratio (SNR) [13], and, on the other hand, for empirical adjustment of R-ARD scheme based on a predefined SNR level that accelerates the convergence rate of the algorithm.

The rest of the correspondence is organized as follows. In Section II, we give an outline of the SBL signal model and the standard RVM algorithm, followed a brief summary of the FMLM scheme. Its relationship to the R-ARD algorithm is outlined in Section III. Finally, in Section IV, we present a small simulation example that illustrates the distinction between the incremental FMLM and R-ARD schemes.

Throughout this correspondence, we shall make use of the following notation. Vectors are represented as boldface lowercase letters, e.g., $\boldsymbol{x}$, and matrices as boldface uppercase letters, e.g., $\boldsymbol{X}$. For vectors and matrices, $(\cdot)^T$ denotes the transpose. Finally, for a random vector $\boldsymbol{x}$, $\mathrm{N}(\boldsymbol{x}|\boldsymbol{a},\boldsymbol{B})$ denotes a multivariate Gaussian pdf with mean $\boldsymbol{a}$ and covariance matrix $\boldsymbol{B}$.

II. Sparse Bayesian Learning

A standard solution to SBL with ARD is a two step procedure that alternates between i) estimating the weight vector $\boldsymbol{w}$ and ii) estimating the corresponding sparsity parameters $\boldsymbol{\alpha}$ and noise precision τ. Given an estimate of the noise precision parameter $\hat{\tau}$ and sparsity parameters $\hat{\boldsymbol{\alpha}}$, an estimate of the weight vector $\widehat{\boldsymbol{w}}$ is obtained as a mode of the posterior pdf $p(\boldsymbol{w}|\boldsymbol{t},\hat{\boldsymbol{\alpha}},\hat{\tau}) = p(\boldsymbol{t}|\boldsymbol{w},\hat{\tau})p(\boldsymbol{w}|\hat{\boldsymbol{\alpha}})$, which can be shown to be a Gaussian pdf $p(\boldsymbol{w}|\boldsymbol{t},\hat{\boldsymbol{\alpha}},\hat{\tau}) = \mathrm{N}(\boldsymbol{w}|\widehat{\boldsymbol{w}},\widehat{\boldsymbol{S}})$ with the mean $\widehat{\boldsymbol{w}}$ and covariance matrix $\widehat{\boldsymbol{S}}$ given by

$$\widehat{\boldsymbol{S}} = \left(\hat{\tau}\boldsymbol{\Phi}^T\boldsymbol{\Phi} + \widehat{\boldsymbol{A}}\right)^{-1} \quad \text{and} \quad \widehat{\boldsymbol{w}} = \hat{\tau}\widehat{\boldsymbol{S}}\boldsymbol{\Phi}^T\boldsymbol{t} \tag{2}$$

where $\widehat{\boldsymbol{A}} = \mathrm{diag}(\widehat{\boldsymbol{\alpha}})$ a diagonal matrix with the elements of $\widehat{\boldsymbol{\alpha}}$ on the main diagonal. The estimates of $\boldsymbol{\alpha}$ and τ are computed by maximizing the marginal likelihood function [6]

$$p(\boldsymbol{t}|\boldsymbol{\alpha},\tau) = \int p(\boldsymbol{t}|\boldsymbol{w},\tau)p(\boldsymbol{w}|\boldsymbol{\alpha})\mathrm{d}\boldsymbol{w} = \mathrm{N}(\boldsymbol{t}|\boldsymbol{0},\tau^{-1}\boldsymbol{I} + \boldsymbol{\Phi}\boldsymbol{A}^{-1}\boldsymbol{\Phi}^T). \tag{3}$$

The maximizers $\hat{\boldsymbol{\alpha}}$ and $\hat{\tau}$ of (3) can be obtained using the EM algorithm where the weights $\boldsymbol{w}$ are used as complete data (see [6] for more details). The corresponding estimation expressions are then given as

$$\widehat{\alpha}_l = (|\widehat{w}_l|^2 + \widehat{S}_{ll})^{-1} \tag{4}$$

and

$$\hat{\tau} = \frac{N}{\|\boldsymbol{t} - \boldsymbol{\Phi}\widehat{\boldsymbol{w}}\|^2 + \mathrm{tr}(\widehat{\boldsymbol{S}}\boldsymbol{\Phi}^T\boldsymbol{\Phi})} \tag{5}$$

where $\widehat{w}_l$ is the lth element of the vector $\widehat{\boldsymbol{w}}$, and $\widehat{S}_{ll}$ is the lth element on the main diagonal of the matrix $\widehat{\boldsymbol{S}}$. The update expressions (2) and (4)–(5) are then repeatedly evaluated until convergence [6].

In cases when $\boldsymbol{t}$ allows for a sparse representation in terms of $\boldsymbol{\Phi}$, the sparsity parameters of some of the components will diverge, forcing the posterior value of the corresponding weights to converge towards zero and, thus, encouraging a sparse solution. Unfortunately, due to the EM-based maximization of the marginal likelihood the rate at which sparsity parameters diverge is low. Many iterations are needed for the hyperparameters to reach a threshold at which they can be treated as "numerically" infinite.[4] This has motivated to use of alternative, more efficient schemes to maximize (3) with respect to $\boldsymbol{\alpha}$.

A. Fast Marginal Likelihood Maximization

One possible strategy to optimize (3) more efficiently consists of computing its optimum with respect to a single sparsity parameter α_l assuming that the other sparsity parameters $\hat{\boldsymbol{\alpha}}_{\overline{l}} = [\widehat{\alpha}_1,\ldots,\widehat{\alpha}_{l-1},\widehat{\alpha}_{l+1},\ldots,\widehat{\alpha}_L]^T$ and the noise precision parameter $\hat{\tau}$ are fixed. Specifically, the logarithm of $p(\boldsymbol{t}|\alpha_l,\hat{\boldsymbol{\alpha}}_{\overline{l}},\widehat{\tau})$ in (3) can be decomposed as

$$\log p(\boldsymbol{t}|\alpha_l,\hat{\boldsymbol{\alpha}}_{\overline{l}},\hat{\tau}) = \mathcal{L}(\alpha_l;\hat{\boldsymbol{\alpha}}_{\overline{l}},\widehat{\tau}) = \mathcal{L}(\hat{\boldsymbol{\alpha}}_{\overline{l}},\hat{\tau}) + \frac{1}{2}\left(\log(\alpha_l) - \log(\alpha_l - s_l) + \frac{q_l^2}{\alpha_l + s_l}\right) \tag{6}$$

where

$$s_l = \boldsymbol{\phi}_l^T\widehat{\boldsymbol{C}}_{\overline{l}}^{-1}\boldsymbol{\phi}_l, \quad q_l = \boldsymbol{\phi}_l^T\widehat{\boldsymbol{C}}_{\overline{l}}^{-1}\boldsymbol{t}, \tag{7}$$

$$\widehat{\boldsymbol{C}}_{\overline{l}} = \hat{\tau}^{-1}\boldsymbol{I} + \sum_{k\neq l}\widehat{\alpha}_k^{-1}\boldsymbol{\phi}_k\boldsymbol{\phi}_k^T, \tag{8}$$

and $\mathcal{L}(\hat{\boldsymbol{\alpha}}_{\overline{l}},\hat{\tau})$ is a part of the marginal log-likelihood $\log p(\boldsymbol{t}|\alpha_l,\hat{\boldsymbol{\alpha}}_{\overline{l}},\tau)$ that is independent of α_l. Then, the maximum of (6) with respect to α_l is obtained at [11]

$$\widehat{\alpha}_l = \begin{cases} s_l^2(q_l^2 - s_l)^{-1}, & q_l^2 > s_l \\ \infty, & q_l^2 \leq s_l. \end{cases} \tag{9}$$

Using (9), the marginal likelihood $p(\boldsymbol{t}|\alpha_l,\hat{\boldsymbol{\alpha}}_{\overline{l}},\widehat{\tau})$ can be maximized incrementally with respect to the sparsity parameter of one component at a time. Observe that the result (9) not only allows for determining if a particular element in $\widehat{\boldsymbol{w}}$ is zero, but also dramatically accelerates the rate of SBL convergence since the optimum of the marginal likelihood can be computed analytically.

Another important consequence of (9) is that the ratio $\widetilde{w}_l^2 = q_l^2/s_l^2$ can be shown to be equal to the squared posterior estimate of the lth weight w_l computed when $\widehat{\alpha}_l = 0$; furthermore, $\widetilde{S}_{ll}$ corresponds to the posterior variance of this weight (see [13] for more details). It follows then that the ratio $\widetilde{w}_l^2/\widetilde{S}_{ll} = q_l^2/s_l$ is an estimate of the component's SNR. The condition $q_l^2 > s_l$ in (9) is then equivalent to the condition $\widetilde{w}_l^2/\widetilde{S}_{ll} > 1$, which has a very intuitive interpretation: components with the estimated SNR below 1 (or equivalently below 0 dB) are removed from the model since according to (9) the corresponding sparsity parameter $\widehat{\alpha}_l$ that maximizes the marginal likelihood $p(\boldsymbol{t}|\alpha_l,\hat{\boldsymbol{\alpha}}_{\overline{l}},\widehat{\tau})$ is infinite. Obviously, this interpretation can be extended by requiring that $\widetilde{w}_l^2/\widetilde{S}_{ll}$ exceeds some other predefined SNR level $\eta_l \geq 1$, which has been shown to improve sparse signal estimation when the actual signal-to-noise ratio is known [13].

[4]For instance, this threshold can be set to 10^{15} or 10^{16}.

III. Reformulated ARD and Incremental ARD Schemes

Let us now analyze the relationship between the R-ARD approach to SBL proposed in [12] and the FMLM scheme. To be consistent with the notation adopted in [12], we define $\boldsymbol{\gamma} = [\gamma_1, \ldots, \gamma_L]^T$ as a prior variance vector with elements $\gamma_l = \alpha_l^{-1}$, $l = 1, \ldots, L$. The marginal likelihood function can then be represented as (see [6], [7], and [12])

$$\log p(\boldsymbol{t}|\boldsymbol{\gamma}, \hat{\tau}) = -\log|\boldsymbol{C}| - \boldsymbol{t}^T\boldsymbol{C}^{-1}\boldsymbol{t} + \text{const} \tag{10}$$

where $\boldsymbol{C} = \hat{\tau}^{-1}\boldsymbol{I} + \sum_{l=1}^{L}\gamma_l\boldsymbol{\phi}_l\boldsymbol{\phi}_l^T$ and the const term collectively represents terms that are independent of $\boldsymbol{\gamma}$. In [12], the authors propose to optimize (10) using auxiliary upper bounding functions. Specifically, they show that the objective function $\mathcal{L}(\boldsymbol{\gamma}) = -\log p(\boldsymbol{t}|\boldsymbol{\gamma}, \widehat{\tau})$ can be upper-bounded as

$$\mathcal{L}(\boldsymbol{\gamma}, z) = z^T\boldsymbol{\gamma} - g^*(z) + \boldsymbol{t}^T\boldsymbol{C}^{-1}\boldsymbol{t} \geq \mathcal{L}(\boldsymbol{\gamma}) \tag{11}$$

where $z = [z_1, \ldots, z_L]^T$ and $g^*(z)$ is the concave conjugate of $\log|\boldsymbol{C}|$ defined by the duality relationship $g^*(z) = \min_{\boldsymbol{\gamma}} z^T\boldsymbol{\gamma} - \log|\boldsymbol{C}|$ [14]. Then, for $\boldsymbol{\gamma}$ fixed at some estimate $\widehat{\boldsymbol{\gamma}}$, the tightest upper bound on $\mathcal{L}(\boldsymbol{\gamma})$ is obtained by minimizing $\mathcal{L}(\boldsymbol{\gamma}, z)|_{\boldsymbol{\gamma}=\widehat{\boldsymbol{\gamma}}} = \mathcal{L}(z; \widehat{\boldsymbol{\gamma}})$ over z. The corresponding minimizer can be found in closed form:

$$\hat{z} = \arg\min_{z} \mathcal{L}(z; \hat{\boldsymbol{\gamma}}) = \text{diag}\{\boldsymbol{\Phi}^T\hat{\boldsymbol{C}}^{-1}\boldsymbol{\Phi}\} \tag{12}$$

where $\text{diag}\{\boldsymbol{\Phi}^T\widehat{\boldsymbol{C}}^{-1}\boldsymbol{\Phi}\}$ is a vector of diagonal entries of $\boldsymbol{\Phi}^T\widehat{\boldsymbol{C}}^{-1}\boldsymbol{\Phi}$ and $\widehat{\boldsymbol{C}} = \hat{\tau}^{-1}\boldsymbol{I} + \sum_{l=1}^{L}\widehat{\gamma}_l\boldsymbol{\phi}_l\boldsymbol{\phi}_l^T$. Now, by fixing z at $\widehat{z}$, the bound $\mathcal{L}(\boldsymbol{\gamma}, z)|_{z=\widehat{z}} = \mathcal{L}(\boldsymbol{\gamma}; \widehat{z})$ is minimized with respect to $\boldsymbol{\gamma}$ by solving

$$\widehat{\boldsymbol{\gamma}} = \arg\min_{\boldsymbol{\gamma}} \mathcal{L}(\boldsymbol{\gamma}; \widehat{z}) = \arg\min_{\boldsymbol{\gamma}} \widehat{z}^T\boldsymbol{\gamma} + \boldsymbol{t}^T\boldsymbol{C}^{-1}\boldsymbol{t}. \tag{13}$$

The R-ARD algorithm then alternates between (12) and (13) until convergence. Note that the objective function (13) is convex; in [12], it has been shown that it can be efficiently optimized using a series of weighted ℓ_1-constrained least squares optimizations. Obviously, a variety of alternative optimization strategies can be derived to optimize $\mathcal{L}(\boldsymbol{\gamma})$ more conveniently. In what follows, we demonstrate that the incremental FMLM approach discussed in Section II-A can also be used to optimize the upper bound $\mathcal{L}(\boldsymbol{\gamma}, z)$ in (11).

Let us consider the optimization of the bound $\mathcal{L}(\boldsymbol{\gamma}, z)$ in (11) in an "incremental" setting. First, consider the optimization of $\mathcal{L}(\boldsymbol{\gamma}; \widehat{z})$ with respect to a single parameter γ_l assuming that γ_k, $k \neq l$, are known and fixed at their estimated values. By defining $\widehat{\boldsymbol{\gamma}}_{\overline{l}} = [\widehat{\gamma}_1, \ldots, \widehat{\gamma}_{l-1}, \widehat{\gamma}_{l+1}, \ldots, \widehat{\gamma}_L]^T$ and noting that the matrix $\boldsymbol{C}$ can be decomposed as $\boldsymbol{C} = \hat{\tau}^{-1}\boldsymbol{I} + \sum_{k\neq l}\gamma_k\boldsymbol{\phi}_k\boldsymbol{\phi}_k^T + \gamma_l\boldsymbol{\phi}_l\boldsymbol{\phi}_l^T$, we rewrite (13) in the following form:

$$\begin{aligned}\mathcal{L}(\gamma_l; \widehat{\boldsymbol{\gamma}}_{\overline{l}}, \widehat{z}) &= \sum_{k\neq l}\hat{z}_k\widehat{\gamma}_k + \boldsymbol{t}^T\widehat{\boldsymbol{C}}_{\overline{l}}^{-1}\boldsymbol{t} + \hat{z}_l\gamma_l - \frac{q_l^2}{\gamma_l^{-1} + s_l} \\ &= \hat{z}_l\gamma_l - \frac{q_l^2}{\gamma_l^{-1} + s_l} + \text{const}\end{aligned} \tag{14}$$

where the parameters q_l and s_l are defined in (7), $\widehat{\boldsymbol{C}}_{\overline{l}}$ is defined in (8), and const is a part of $\mathcal{L}(\gamma_l; \widehat{\boldsymbol{\gamma}}_{\overline{l}}, \widehat{z})$ that is independent of γ_l. We now compute the value of γ_l that minimizes (14) by computing

$$\frac{\mathrm{d}\mathcal{L}(\gamma_l; \widehat{\boldsymbol{\gamma}}_{\overline{l}}, \widehat{z})}{\mathrm{d}\gamma_l} = \hat{z}_l - \frac{q_l^2}{(1 + \gamma_l s_l)^2}$$

and equating the result to zero. Solving for γ_l gives two stationary points of $\mathcal{L}(\gamma_l; \widehat{\boldsymbol{\gamma}}_{\overline{l}}, \widehat{z})$ at

$$\widehat{\gamma}_{l,1} = \frac{-|q_l| - \sqrt{\hat{z}_l}}{s_l\sqrt{\hat{z}_l}} \quad \text{and} \quad \widehat{\gamma}_{l,2} = \frac{|q_l| - \sqrt{\hat{z}_l}}{s_l\sqrt{\hat{z}_l}}. \tag{15}$$

Notice that the first solution $\widehat{\gamma}_{l,1}$ is always negative and, thus, infeasible[5]; the second solution is positive provided $|q_l| > \sqrt{\hat{z}_l}$. Let us study these stationary points in more detail.

It is known that a stationary point γ_l^* is a local minimum of $\mathcal{L}(\gamma_l; \widehat{\boldsymbol{\gamma}}_{\overline{l}}, \widehat{z})$ if, and only if,

$$\left.\frac{\mathrm{d}^2\mathcal{L}(\gamma_l; \widehat{\boldsymbol{\gamma}}_{\overline{l}}, \widehat{z})}{\mathrm{d}\gamma_l^2}\right|_{\gamma_l=\gamma_l^*} = \left.\frac{2q_l^2 s_l}{(1 + \gamma_l s_l)^3}\right|_{\gamma_l=\gamma_l^*} > 0. \tag{16}$$

By evaluating (16) at $\gamma_l^* = \widehat{\gamma}_{l,1}$, it is easy to see that the sign of the second derivative at this point is negative, i.e., $\widehat{\gamma}_{l,1}$ is not a local minimum of (14). In contrast, $\widehat{\gamma}_{l,2}$ in (15) is a local minimum. However, it is only a feasible optimum provided $|q_l| > \sqrt{\hat{z}_l}$. Thus,

$$\widehat{\gamma}_l = \arg\min_{\gamma_l} \mathcal{L}(\gamma_l; \widehat{\boldsymbol{\gamma}}_{\overline{l}}, \widehat{\boldsymbol{z}}) = \begin{cases} \frac{|q_l| - \sqrt{\hat{z}_l}}{s_l\sqrt{\hat{z}_l}} & |q_l| > \sqrt{\hat{z}_l}, \\ \text{no feasible sol.} & |q_l| \leq \sqrt{\hat{z}_l}. \end{cases} \tag{17}$$

Notice a similarity between the structure of the solution (17) and that in (9). As we will see later, the condition $|q_l| > \sqrt{\hat{z}_l}$ is in fact related to the pruning conditions of the FMLM algorithm.

Once the optimum of (14) with respect to γ_l is found, we return to (12) and re-estimate $\widehat{z}$ using an updated vector $\widehat{\boldsymbol{\gamma}}$, in which the lth element has been computed using (17). What will happen if the updates (12) and (17) are repeated *ad infinitum* for some fixed component $\boldsymbol{\phi}_l$? To answer this question we note that the value of $\widehat{\gamma}_l$ in (17) depends explicitly only on the lth element of $\widehat{z}$. Applying the Woodbury matrix identity [15] to the inverse of $\widehat{\boldsymbol{C}} = \hat{\tau}^{-1}\boldsymbol{I} + \sum_{l=1}^{L}\widehat{\gamma}_l\boldsymbol{\phi}_l\boldsymbol{\phi}_l^T$ and combining the result with (12) we can rewrite the lth element of $\widehat{z}$ as

$$\hat{z}_l = \hat{\tau}\boldsymbol{\phi}_l^T\boldsymbol{\phi}_l - \hat{\tau}^2\boldsymbol{\phi}_l^T\boldsymbol{\Phi}\widehat{\boldsymbol{S}}\boldsymbol{\Phi}^T\boldsymbol{\phi}_l. \tag{18}$$

The result (18) allows us to establish a relationship between the value of $\hat{z}_l$ and the parameter s_l in (7). Specifically, it can be shown (see [11, eq. (23) and (24)]) that these parameters are related through the following transformation:

$$\hat{z}_l = \widehat{\gamma}_l^{-1}s_l(\widehat{\gamma}_l^{-1} + s_l)^{-1}. \tag{19}$$

Now, if (19) and (17) are repeatedly evaluated *ad infinitum*, then[6]

$$\hat{z}_l \to \frac{s_l^2}{q_l^2} \quad \text{and} \quad \widehat{\gamma}_l \to s_l^{-2}(q_l^2 - s_l). \tag{20}$$

It can be seen that the value of $\widehat{\gamma}_l$ in (20) coincides with the value of $\widehat{\alpha}_l^{-1}$ in (9). Notice that this result is valid provided $|q_l| > \sqrt{\hat{z}_l}$, i.e., when $\widehat{\gamma}_l$ in (17) is a feasible optimum. However, for $\hat{z}_l = s_l^2/q_l^2$, the condition $|q_l| > \sqrt{\hat{z}_l}$ is equivalent to the condition $q_l^2 > s_l$, which in turn guarantees the positivity of $\widehat{\gamma}_l$ in (20) and coincides with the pruning condition in (9).

Thus, the FMLM algorithm realizes an incremental optimization of the R-ARD objective function. Moreover, due to the convexity of the (13) such incremental optimization is guarantied to converge to a local optimum of the R-ARD objective function. Furthermore, the pruning condition in (9) also determines if the minimum of (13) is achieved at a feasible solution.

Let us point out that the connection between the R-ARD and the FMLM scheme can be exploited to develop SNR-based adjustments of the R-ARD similar to those used for FMLM algorithm. Specifically, by combining (19) and $\widehat{\gamma}_l = s_l^{-2}(q_l^2 - s_l)$ from (20) with the adjusted

[5]Recall that γ_l models the prior variance of the weight w_l and must be non-negative.

[6]We leave this result without proof. It can be easily verified by substituting a feasible solution for $\widehat{\gamma}_l$ from (17) into (19) and solving for $\hat{z}_l$, which gives $\hat{z}_l = s_l^2/q_l^2$. Inserting the latter result in (17) gives (20).

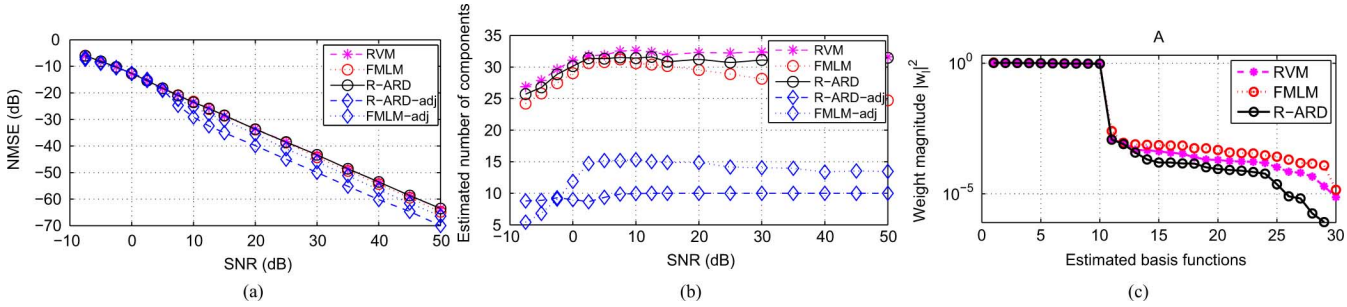

Fig. 1. (a) Estimated NMSE and (b) number of estimated components versus SNR. (c) Nonzero elements of a sparse estimate of a weight vector for 30 dB SNR. The plotted values in (c) correspond to a single realization of the learning algorithms.

pruning condition $q_l^2/s_l > \eta_l$, where $\eta_l \geq 1$ is the pruning SNR level, it can be shown that the FMLM pruning condition $q_l^2/s_l > \eta_l$ is equivalent to

$$\gamma_l > \frac{\eta_l - 1}{\eta_l} z_l^{-1}. \tag{21}$$

Condition (21) can be used as a pruning condition for the R-ARD scheme. In other words, the sparsity level of the R-ARD algorithm can be adjusted based on (21) and some preselected sensitivity $\eta_l \geq 1$ expressed in the units of SNR. In the next experimental section, we demonstrate the usefulness of such an adjustment.

IV. A Simple Illustrative Example

In this section, we contrast the performance of the incremental ARD scheme, i.e., of the FMLM algorithm, to that of the R-ARD algorithm. Note that while the latter finds a solution via a series of weighted ℓ_1-constrained[7] least-squares optimizations as suggested in [12], the FMLM algorithm optimizes the same R-ARD objective function incrementally, with the optimum at each step computed analytically.

We will also compare the performance of FMLM and R-ARD to the standard RVM algorithm with noninformative hyperpriors that uses EM algorithm to estimate model parameters [6], as well as to the FMLM and R-ARD algorithms that use SNR-adjusted pruning rules $q_l^2/s_l > \eta_l$ in (9) and (21), respectively; we will refer to the latter two schemes as FMLM-adj and R-ARD-adj schemes, respectively. For both FMLM-adj and R-ARD-adj schemes, the adjustment threshold is set to $\eta_l = 3.16$, $l = 1, \ldots, L$, which corresponds to 5 dB SNR. This adjustment has been used in all simulations. Our goal here is to construct a simple scenario where the distinctions between the R-ARD approach to SBL and the incremental FMLM, as well as the impact of SNR-based adjustment can be easily demonstrated. We acknowledge, however, that the performed experiments do not by any means represent a comprehensive comparison of the methods, which cannot be included due to the correspondence length constraints.

As performance measures, we compute the normalized mean-square error (NMSE) and the estimated number of components versus SNR; additionally, we demonstrate the rate at which components are removed from the model and the convergence rate of the estimated weight vector $\boldsymbol{w}$ as a function of the algorithm iteration index. The estimated quantities are averaged over 100 independent algorithm runs. Note that in order to estimate the number of components with the RVM and R-ARD algorithms, it is necessary to specify a pruning threshold for the sparsity parameters $\widehat{\alpha}_l$, $l = 1, \ldots, L$, which is essentially a numerical way of detecting their divergence. Specifically, when for some component $\boldsymbol{\phi}_l$ an estimate of $\widehat{\alpha}_l$ exceeds the pruning threshold, the corresponding component is removed from the model. Let us stress that, in the case of the incremental approach, such a pruning threshold is not needed: the pruning conditions in (9) essentially "detect" the divergence of sparsity parameters; the same also holds for the R-ARD-adj algorithm. In our implementation of the RVM and R-ARD algorithms, we set this threshold to 10^8. We note that, in the case of the R-ARD algorithm, this is also needed since in the presence of noise the solution obtained by ℓ_1-constrained solver might not be exactly sparse; in fact, some elements in $\boldsymbol{w}$ might become very small, but nonetheless numerically still larger than 0. To further simplify the analysis of the simulation results, we will assume the noise precision parameter $\hat{\tau}$ to be known and fixed in all simulations.

A simple compressive sampling toy problem is considered. We assume that the components $\boldsymbol{\phi}_l \in \mathbb{R}^N$, $l = 1, \ldots, L$, are generated by drawing N samples from a Gaussian distribution with zero mean and variance $1/\sqrt{N}$. A sparse vector $\boldsymbol{w}$ is constructed by setting K elements of $\boldsymbol{w}$ to ± 1 at random locations. The target vector $\boldsymbol{t}$ is then generated according to (1). In this experiment, we set $N = 100$, $K = 10$, and $L = 200$.

In Fig. 1, we plot the estimated NMSE and the estimated number of components versus SNR. Notice that the best performance is achieved for R-ARD-adj algorithm, followed closely by the FMLM-adj scheme; the normalized mean square-error (NMSE) performance of the other algorithms is indistinguishable. The reason for such a good performance of the R-ARD-adj is the adjusted pruning condition that leads to the improved estimated signal sparsity, as can be seen in Fig. 1(b): the R-ARD-adj algorithm estimates the correct number of components almost over the whole tested SNR range; in contrast, other schemes underestimate the true signal sparsity. In the case of the FMLM-adj scheme, this adjustment is less effective. This can be explained by the "gain" of the R-ARD scheme due to the joint estimation of the weight vector $\boldsymbol{w}$. Let us point out that the underestimation of the signal sparsity is due to the fact that when noise is present some of the estimated weights take very small, yet nonzero values, as demonstrated in Fig. 1(c). Clearly, with an appropriate thresholding it is possible to remove these weak "noisy" components to recover the true sparsity. Yet in practical situations the discrepancy between zero and nonzero estimated weights might not be so distinct as in Fig. 1(c) and finding an optimal threshold might be quite challenging. The proposed adjustments of the pruning conditions, originally used in [13] for FMLM and extended here in (21) to the R-ARD scheme, readily provide a way to optimally select this threshold based on the analytical analysis of the inference expressions; surprisingly, these adjustments give quite accurate results. Unfortunately, the space constraints prohibit us from a more detailed study of the adjusted pruning conditions.

Now, in Fig. 2, we demonstrate the performance of the compared algorithms versus the number of update iterations; specifically, we compute the ℓ_2-norm of the difference between the estimated vectors $\boldsymbol{w}$ at

[7]The ℓ_1-constrained optimization is implemented using a *simple Matlab solver for ℓ_1 -regularized least squares problems*, which uses a log-barrier method [14] to optimize the corresponding objective function. The duality gap for the log-barrier solver is set to 10^{-10}. The software is available online at http://www.stanford.edu/ boyd/l1_ls/.

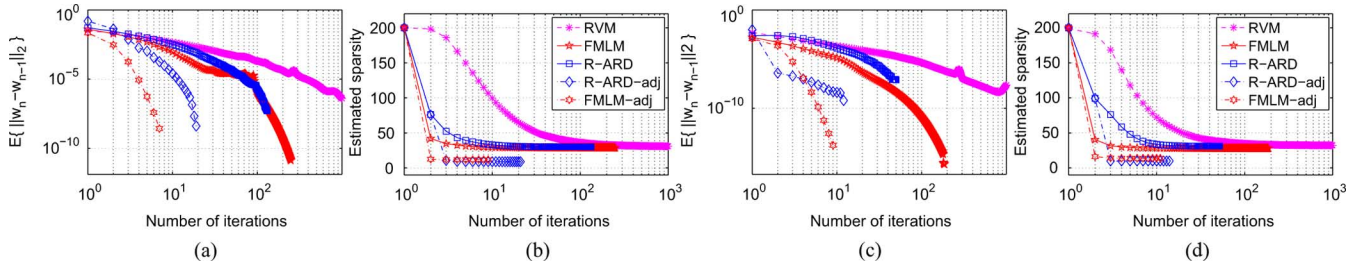

Fig. 2. Convergence of the weight vector $\boldsymbol{w}$ and estimated signal sparsity versus the number of algorithm iterations for (a),(b) 0 dB SNR and (c),(d) 30 dB SNR.

two consecutive iterations as well as the number of estimated components versus the algorithm iteration index for 0 dB and 30 dB SNR. Here, each iteration includes a complete update of all L components present in the model. Note that, for the R-ARD algorithm, a single update iteration includes a single ℓ_1-constrained optimization, which is also an iterative scheme. The reported number of iterations for the R-ARD algorithm is obtained assuming that an ℓ_1-constrained optimization is solved instantaneously.

Again we observe that the adjusted schemes outperform the other tested methods: due to the used adjustment, the irrelevant components are removed already during the early iterations and roughly ten updates are needed for the remaining weights to converge. As expected, the RVM algorithm performs the worst among the compared schemes. The performance of R-ARD and FMLM schemes is comparable, yet the former is computationally more demanding since it requires solving a series of ℓ_1-constrained optimizations; obviously, in this respect the incremental approach is computationally much more attractive. Note that the convergence rate of R-ARD improves as the SNR grows. In contrast, the rate at which the incremental FMLM algorithm removes components seems to be almost independent of the SNR.

V. Conclusion

In this work, a relationship between the incremental fast marginal likelihood maximization (FMLM) algorithm and the reformulated ARD (R-ARD) algorithm has been established.

We have demonstrated that the FMLM algorithm realizes an incremental optimization of the R-ARD objective function. The pruning condition of the FMLM algorithm that determines whether the maximum of the marginal likelihood with respect to a single sparsity parameter is achieved at a finite optimum coincides with the condition that guarantees the existence of a feasible minimum of the R-ARD objective function with respect to this sparsity parameter.

Based on this relationship the empirical adjustments of the pruning conditions, previously proposed for the FMLM algorithm, can be derived for the R-ARD scheme as well. This provides an additional degree of freedom for the R-ARD scheme to control the estimated signal sparsity based on the signal-to-noise ratio. Experimental results based on a simple compressive sampling toy problem indicate that the use of adjustments significantly boosts the convergence rate of the scheme and reduces the underestimation of the true signal sparsity.

References

[1] R. Baraniuk, "Compressive sensing," *IEEE Signal Process. Mag.*, vol. 24, no. 4, pp. 118–121, Jul. 2007.
[2] D. Donoho, "For most large underdetermined systems of linear equations, the minimal ell-1 norm solution is also the sparsest solution," *Commun. Pure Appl. Math.*, vol. 59, no. 6, pp. 797–829, Jun. 2006.
[3] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 33–61, 1998.
[4] M. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, Mar. 2008.
[5] D. G. Tzikas, A. C. Likas, and N. P. Galatsanos, "The variational approximation for Bayesian inference," *IEEE Signal Process. Mag.*, vol. 25, no. 6, pp. 131–146, Nov. 2008.
[6] M. Tipping, "Sparse Bayesian learning and the relevance vector machine," *J. Mach. Learn. Res.*, vol. 1, pp. 211–244, Jun. 2001.
[7] D. Wipf and B. Rao, "Sparse Bayesian learning for basis selection," *IEEE Trans. Signal Process.*, vol. 52, no. 8, pp. 2153–2164, Aug. 2004.
[8] M. Figueiredo, "Adaptive sparseness for supervised learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 9, pp. 1150–1159, 2003.
[9] N. L. Pedersen, C. N. Manchon, D. Shutin, and B. H. Fleury, "Application of Bayesian hierarchical prior modeling to sparse channel estimation," presented at the IEEE Int. Conf. Commun. (ICC), 2012.
[10] A. Lee, F. Caron, A. Doucet, and C. Holmes, "A hierarchical Bayesian framework for constructing sparsity-inducing priors" [Online]. Available: http://arxiv.org/abs/1009.1914v2, preprint arXiv:1009.1914.
[11] M. E. Tipping and A. C. Faul, "Fast marginal likelihood maximisation for sparse Bayesian models," presented at the 9th Int. Workshop Artif. Intell. Stat., Key West, FL, Jan. 2003.
[12] D. Wipf and S. Nagarajan, "A new view of automatic relevance determination," presented at the 21 Annu. Conf. Neural Inf. Process. Syst., Vancouver, BC, Canada, Dec. 2007.
[13] D. Shutin, T. Buchgraber, S. R. Kulkarni, and H. V. Poor, "Fast variational sparse Bayesian learning with automatic relevance determination for superimposed signals," *IEEE Trans. Signal Process.*, vol. 59, no. 12, pp. 6257–6261, Dec. 2011.
[14] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, MA: Cambridge Univ. Press, 2004.
[15] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. Baltimore, MD: The Johns Hopkins Univ. Press, 1996.