SPARSE SIGNAL RECOVERY USING MPDR ESTIMATION

Maher Al-Shoukairi and Bhaskar D. Rao

Dept. of ECE, University of California, San Diego, 9500 Gilman Dr, San Diego, CA, 92103

ABSTRACT

Utilizing the array processing minimum power distortionless response (MPDR) beamformer framework, we present a new perspective on the sparse Bayesian learning (SBL) algorithm used in sparse signal recovery. In addition to providing more insight into the SBL algorithm, this new perspective allows us to extend the algorithm to more general non-Gaussian priors. Finally, we use the connection between the MPDR and the LMMSE estimator to lower the complexity of the algorithm using a generalized approximate message passing (GAMP) based LMMSE estimator. The result is a GAMP based algorithm with improved convergence properties.

Index Terms**—** Compressed sensing, Bayesian learning, array processing, approximate message passing

1. INTRODUCTION

We consider recovering a vector $\boldsymbol{x} \in \mathbb{R}^N$ from a noisy single measurement vector $\boldsymbol{y} \in \mathbb{R}^M$:

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{e}, \tag{1}$$

where $\boldsymbol{A} \in \mathbb{R}^{M\times N}$ is assumed to be known, and the noise is modeled by white Gaussian noise as $\boldsymbol{e} \sim \mathcal{N}(0, \sigma^2\boldsymbol{I})$. We focus on the sparse signal recovery (SSR) problem, where the undersampled regime is considered with $M < N$. When $\boldsymbol{x}$ is sufficiently sparse, and the matrix $\boldsymbol{A}$ satisfies certain conditions, accurate recovery of $\boldsymbol{x}$ is possible through different techniques. Among the successful approaches of solving the SSR problem are Bayesian techniques [1–6], where the distribution $p(\boldsymbol{x})$ from which the signal $\boldsymbol{x}$ is drawn is used with the system model in (1) to obtain a maximum a posteriori (MAP) or a minimum mean squared error (MMSE) estimate of $\boldsymbol{x}$ [4–7]. In the case that $p(\boldsymbol{x})$ is not known, empirical Bayesian approaches [5–9] can be used to learn the prior while estimating the signal $\boldsymbol{x}$. In the first part of this paper we focus on an empirical Bayesian approach known as the sparse Bayesian learning (SBL) algorithm [5], the algorithm imposes a hierarchical prior on $p(\boldsymbol{x})$, by imposing a Gaussian scale mixture (GSM) prior controlled by a hyperparameter $\boldsymbol{\gamma}$ with distribution $p(\boldsymbol{\gamma})$. The expectation maximization (EM) algorithm is used to estimate the hyperparameter, where the SBL has the advantage of a closed form E-step because of its GSM prior. Once an estimate $\hat{\boldsymbol{\gamma}}$ is obtained an MMSE estimate of $\boldsymbol{x}$ can be obtained. The main ideas we present in this paper are:

1 We provide a different perspective on the E-step of the SBL algorithm for a fixed $\boldsymbol{\gamma}$. The new perspective uses the system model to design an estimator that maximizes the signal to noise and interference (SINR) ratio, a receiver which is known in the array processing literature as minimum power distortionless response (MPDR) beamformer [10]. Using the MPDR receiver the measurement vector is decoupled into AWGN estimates of the elements of $\boldsymbol{x}$. Once decoupled MMSE estimation can be performed to obtain an estimate $\hat{\boldsymbol{x}}$.

2 The MPDR perspective of SBL allows us to exploit the decoupling of $\boldsymbol{x}$ by the MPDR to acquire MMSE estimates for more general non-Gaussian priors. To demonstrate that, we examine the non-negative SSR problem. When the MPDR approach is applied to the the non-negative SSR problem, the resulting algorithm is identical to one of the versions of rectified SBL (RSBL) from [8]. This result provides some insight into why the approximation used in [8] to derive the algorithm is justified.

3 To further demonstrate the capabilities of the MPDR framework, we consider the case when the prior $p(\boldsymbol{x})$ is fully known. In this context we again demonstrate how the MPDR decoupling can be used to obtain an MMSE estimate.

4 Finally, we use the connection between the MPDR and the LMMSE estimator to propose a low complexity version of the algorithm. This low complexity version is based on a generalized approximate message passing (GAMP) [11] implementation of the LMMSE. The resulting algorithm enhances the convergence properties of GAMP. While the algorithm shares some properties with the vector approximate message passing (VAMP) algorithm [12], we point out some the differences between the two algorithms in section 5.

2. SPARSE BAYESIAN LEARNING

The SBL algorithm imposes a GSM prior on $\boldsymbol{x}$.

$$p(x_n) = \int \mathcal{N}(x_n; 0, \gamma_n)p(\gamma_n)d\gamma_n, \tag{2}$$

where the prior of the hyperparameter $p(\gamma_n)$ controls the actual prior on x_n. We will focus on the case considered in [5], when $p(\gamma_n)$ is a non-informative prior, and the prior on $\boldsymbol{x}$ becomes a parametrized Gaussian, with $\boldsymbol{\gamma}$ as its variance. The hyperparameter γ_n can be iteratively estimated using evidence maximization, where the EM algorithm was chosen for the SBL algorithm in [5]. Once an estimate $\hat{\boldsymbol{\gamma}}$ is obtained the posterior $p(\boldsymbol{x}|\boldsymbol{y})$ is approximated by $p(\boldsymbol{x}|\boldsymbol{y};\hat{\boldsymbol{\gamma}})$, and the mean of this posterior is used as a point estimate for $\boldsymbol{x}$. Based on the model in (1) and the non-informative prior on γ_n,

$$p(\boldsymbol{y}|\boldsymbol{x};\sigma^2) = \frac{1}{(2\pi\sigma^2)^{\frac{M}{2}}}\exp\left(-\frac{1}{2\sigma^2}\|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\|^2\right) \tag{3}$$

$$p(x_n|\gamma_n) = \prod_{n=1}^{N}\frac{1}{(2\pi\gamma_n)^{\frac{1}{2}}}\exp\left(-\frac{x_n^2}{2\gamma_n}\right) \tag{4}$$

We will assume that the noise variance σ^2 is known throughout the paper. The E-step has a closed form solution, which happens to be

This work was supported in part by NSF Grant CCF-1617365 and ONR Grant No. N00014-18-1-2038

the LMMSE estimate of $p(\boldsymbol{x}|\boldsymbol{y};\boldsymbol{\gamma})$.

$$p(\boldsymbol{x}|\boldsymbol{y};\boldsymbol{\gamma}) = \mathcal{N}(\boldsymbol{x};\hat{\boldsymbol{x}},\boldsymbol{\Sigma_x}) \tag{5}$$
$$\hat{\boldsymbol{x}} = \Gamma \boldsymbol{A}^\top(\sigma^2\boldsymbol{I}+\boldsymbol{A}\boldsymbol{\Gamma}\boldsymbol{A}^\top)^{-1}\boldsymbol{y} \tag{6}$$
$$\boldsymbol{\Sigma_x} = \boldsymbol{\Gamma} - \boldsymbol{\Gamma}\boldsymbol{A}^\top(\sigma^2\boldsymbol{I}+\boldsymbol{A}\boldsymbol{\Gamma}\boldsymbol{A}^\top)^{-1}\boldsymbol{A}\boldsymbol{\Gamma}, \tag{7}$$

where $\boldsymbol{\Gamma}$ is a diagonal matrix created using $\boldsymbol{\gamma}$. We will refer to the mean estimate of $\boldsymbol{x}$ by $\hat{\boldsymbol{x}}$, we also refer to the diagonal entries of Σ_x by τ_{x_n}. Once the E-step is completed the M-step is carried out as:

$$\gamma_n = \hat{x}_n^2 + \tau_{x_n} \tag{8}$$

The algorithm then iterates between these two steps until convergence. We will refer to this algorithm by the EM-SBL.

3. AN MPDR PERSPECTIVE OF SBL

3.1. MPDR SBL Derivation

Rather than applying the E-step directly to get an estimate of $\boldsymbol{x}$, we consider finding a linear estimator with weight vector w_n for each element of $\boldsymbol{x}$ denoted by x_n. The weight vector is chosen to recover an undistorted version of x_n while minimizing the total output power of the estimator, and therefore maximizing the signal to noise ratio (SNR). Based on the system model in (1), the covariance matrix of $\boldsymbol{y}$ is given by:

$$S_y = (\boldsymbol{A}\boldsymbol{P}\boldsymbol{A}^\top + \sigma^2\boldsymbol{I}) \tag{9}$$

where $\boldsymbol{P}$ is a diagonal matrix, with the diagonal entries p_n that represent the input powers of x_n, we note here that this power is unknown and iteratively estimating it is one of the goals of the proposed algorithm. We also note, that in the case of the parametrized Gaussian prior in (4), this power is the same as the parameter γ_n, however this does not have to be the case for other priors. With this covariance matrix, our optimization problem becomes:

$$\underset{w_n}{\text{argmin}}\; w_n^\top S_y w_n \quad \text{s.t. } w_n^\top a_n = 1$$

where a_n is the nth column of $\boldsymbol{A}$. The solution to this optimization problem can be obtained using Lagrange multipliers, and was previously found in the MPDR receiver [10] to be:

$$w_n = \frac{S_y^{-1}a_n}{(a_n^\top S_y^{-1}a_n)} \tag{10}$$

Next, we divide the proposed approach into three main steps:

A) MPDR Estimation

Applying $w_n^\top$ to the measurement vector $\boldsymbol{y}$, we obtain a noise corrupted version of x_n,

$$w_n^\top\boldsymbol{y} = r_n = x_n + v_n. \tag{11}$$

With enough number of interfering sources (columns of $\boldsymbol{A}$), this noise can be justifiably modeled as Gaussian, $v_n \approx \mathcal{N}(0,\tau_{r_n})$. Where τ_{r_n} is equal to the total output power of the MPDR minus the power of x_n. The total output power of the MPDR is given by [10] as $1/a_n^\top S_y^{-1}a_n$ and:

$$\tau_{r_n} = 1/(a_n^\top S_y^{-1}a_n) - \gamma_n \tag{12}$$

B) MMSE Estimation

An estimate of x_n can be obtained by performing MMSE estimation based on (11) and the prior on x_n. In the case of the parametrized Gaussian prior in (4) this estimate becomes:

$$\hat{x}_n = \mathbb{E}\{x_n|r_n\} = (r_n\gamma_n)/(\tau_{r_n}+\gamma_n) \tag{13}$$
$$\tau_{x_n} = \text{var}\{x_n|r_n\} = (\tau_{r_n}\gamma_n)/(\tau_{r_n}+\gamma_n) \tag{14}$$

C) Power and Prior Updates

Using the estimate $\hat{x}_n$ the power estimate p_n can be updated by:

$$p_n = \mathbb{E}\{x_n^2\} = \hat{x}_n^2 + \tau_{x_n} \tag{15}$$

Since the model assumes an unknown hyperprameter γ_n, the estimates from the MMSE step are used to update it. The update is identical to the p_n power update in (15), because based on the assumed prior the two quantities are the same. The algorithm then iterates between estimating $\hat{\boldsymbol{x}}$ and $\boldsymbol{\tau_x}$, and between updating the powers/hyperparameters until convergence, we refer to this algorithm as MPDR-SBL. It is worth mentioning that in the context of source localization, [13] proposed an algorithm that iteratively estimates the power of the sources using an MPDR estimator, however no prior was assumed for the sources, and there was no MMSE step in the algorithm.

General prior case: In the case that a different prior is assumed on $\boldsymbol{x}$, say for example a Laplacian scale mixture (LSM) [4, 14] was imposed on $\boldsymbol{x}$, then the **MPDR estimation** and **power update** steps remain unchanged. While the **MMSE estimation** in (13) and (14) will change based on the LSM prior.

3.2. MPDR-SBL VERSUS EM-SBL

In this section we show the equivalence of the derived algorithm to the original EM-SBL [5]. In the EM-SBL algorithm, the E-step consists of performing an LMMSE estimate on $\boldsymbol{x}$ to obtain $\mathbb{E}\{\boldsymbol{x}|\boldsymbol{y};\boldsymbol{\gamma}\}$. Once an estimate on $\boldsymbol{x}$ is obtained, the M-step in the EM-SBL algorithm given in (8), is equivalent to the updates given in (15) for the powers of $\boldsymbol{p}$ of $\boldsymbol{x}$ and the hyperparameters $\boldsymbol{\gamma}$ of the prior $p(\boldsymbol{x})$. Based on that, we only need to show that the LMMSE estimate used in the EM-SBL is equivalent to the **MPDR estimate**, followed by an **MMSE estimate** of x_n. Because $\boldsymbol{\gamma}$ and $\boldsymbol{p}$ are equivalent in the case of the assumed prior of SBL, in the following steps we will replace p_n in (9) with γ_n to show the equivalence of the MPDR-SBL and the EM-SBL. We start with the mean estimate of the two algorithms. In the MPDR-SBL case, the mean estimate of each x_n is given by (13). We note from (12) that $\tau_{r_n} + \gamma_n = 1/a_n^\top S_y^{-1}a_n$. Therefore the estimate from (13) is rewritten as,

$$\begin{aligned}\hat{x}_n &= (\gamma_n r_n)/(1/a_n^\top S_y^{-1}a_n)\\ &= (\gamma_n w_n^\top\boldsymbol{y})/(1/a_n^\top S_y^{-1}a_n) = \gamma_n a_n^\top S_y^{-1}\boldsymbol{y},\end{aligned} \tag{16}$$

which is identical to the mean estimate of x_n in (6) when $S_y = (\sigma^2\boldsymbol{I}+\boldsymbol{A}\boldsymbol{\Gamma}\boldsymbol{A}^\top)$. Next we show that the variance estimate obtained in the MPDR-SBL by the MPDR then MMSE estimation is equivalent to (7).

$$\tau_{r_n} = \frac{1}{a_n^\top S_y^{-1}a_n} - \gamma_n = \frac{1-\gamma_n a_n^\top S_y^{-1}a_n}{a_n^\top S_y^{-1}a_n} \tag{17}$$
$$\tau_{x_n} = \frac{\tau_{r_n}\gamma_n}{1/a_n^\top S_y^{-1}a_n} = \gamma_n - \gamma_n^2 a_n^\top S_y^{-1}a_n, \tag{18}$$

which is identical to the diagonal of Σ_x given in (7). This shows that the MPDR-SBL is exactly equivalent to the EM-SBL.

The MPDR approach offers a new perspective on the SBL algorithm, where the algorithm starts with an assumption of the input powers of the vector $\boldsymbol{x}$, then uses an estimator to maximize the SNR based on these assumed powers, followed by an MMSE estimator based on the prior. Once an estimate of $\boldsymbol{x}$ is obtained, this estimate is used to update the input power, as well as the unknown prior parameters. We next show how this concept can be extended to more general priors.

4. EXTENSION TO GENERAL PRIORS

4.1. Rectified Sparse Bayesian Learning

The rectified SBL algorithm was proposed in [8] to address the nonegative SSR problem, which is similar to the problem in (1) with non-negativety constraints on $\boldsymbol{x}$. Instead of the parametrized GSM, [8] proposed a "rectified" non-negative GSM prior on $\boldsymbol{x}$, given by:

$$p(x_n|\gamma_n) = \mathcal{N}^R(x_n; 0, \gamma_n), \tag{19}$$

$$\mathcal{N}^R(x_n; 0, \gamma_n) = 2\mathcal{N}(x; 0, \gamma_n)u(x_n), \tag{20}$$

Similar to the EM-SBL, [8] uses the EM algorithm to estimate the hyperparameter $\boldsymbol{\gamma}$, and then uses the approximated posterior $p(\boldsymbol{x}|\boldsymbol{y};\hat{\boldsymbol{\gamma}})$ to get an estimate of $\boldsymbol{x}$. Unlike the SBL algorithm $p(\boldsymbol{x}|\boldsymbol{y};\boldsymbol{\gamma})$ does not have a closed form solution, and therefore [8] used a number of techniques to approximate it,

$$p(\boldsymbol{x}|\boldsymbol{y};\boldsymbol{\gamma}) = c(\boldsymbol{y}) \exp(-\frac{1}{2}(\boldsymbol{x} - \hat{\boldsymbol{x}})^\top \Sigma_x^{-1}(\boldsymbol{x} - \hat{\boldsymbol{x}})), \tag{21}$$

where $\hat{\boldsymbol{x}}$ and Σ_x^{-1} are the same as the ones given in (6) and (7). The reason $p(\boldsymbol{x}|\boldsymbol{y};\boldsymbol{\gamma})$ cannot be obtained in closed form is because $c(\boldsymbol{y})$ does not have a closed form expression. One of the proposed approximations was to assume that Σ_x^{-1} is a diagonal matrix which allows the posterior distribution to decouple into a product of marginals $p(x_n|\boldsymbol{y};\gamma_n) \propto \mathcal{N}^R(x_n; \hat{x}_n, \tau_{x_n})$. Based on this decoupling, estimates for $\tilde{x}_n = \mathbb{E}\{x_n|\boldsymbol{y};\gamma_n\}$ and $\tilde{\tau}_{x_n} = \text{var}\{x_n|\boldsymbol{y};\gamma_n\}$ can be easily obtained by the mean and variance of the rectified Gaussian distribution and are given in [8]. We now show how we can extend the MPDR-SBL to the non-negative case using the prior in (19). We will show how the derived algorithm is identical the diagonal approximation used in [8]. Therefore, we offer some insight into why the diagonal approximation given in [8] is justified.

Similar to the MPDR-SBL case, we construct the covariance matrix from (9) using p_n values. We point out here that similar to the SBL case, $\boldsymbol{p}$ happens to correspond to the hyperparameter vector $\boldsymbol{\gamma}$ and can be used interchangeably.

$$S_y = (\boldsymbol{A}\boldsymbol{\Gamma}\boldsymbol{A}^\top + \sigma^2\boldsymbol{I}) \tag{22}$$

We apply the **MPDR estimation** step in (11) and (12) to estimate r_n and τ_{r_n}. We then perform the **MMSE estimation** step. Using the prior from (19) and the MPDR output in (11),

$$p(x_n|r_n) \propto \mathcal{N}^R(x_n; 0, \gamma_n)\mathcal{N}(x_n; r_n, \tau_{r_n}) \tag{23}$$

$$\propto \mathcal{N}^R(x_n; \hat{x}_n, \tau_{x_n}) \tag{24}$$

$$\tilde{x}_n = \mathbb{E}\{x_n|r_n\}, \quad \tilde{\tau}_{x_n} = \text{var}\{x_n|r_n\}, \tag{25}$$

where (24) is obtained using the rule for the multiplication of two Gaussian pdfs. We can see that the posterior approximation obtained in (24) is identical to the one obtained in [8] for $p(x_n|\boldsymbol{y};\gamma_n)$. Based on that, the estimates obtained in (25) are also identical to the ones obtained in [8]. Given the approximated posterior $p(x_n|r_n;\gamma_n)$ we update the estimates of the input powers, and the hyperparameters:

$$\gamma_n = p_n = \mathbb{E}\{x^2\} = \tilde{x}_n^2 + \tilde{\tau}_{x_n} \tag{26}$$

Which is also identical to the M-step update in the RSBL given in [8]. Therefore, the decoupling of $\boldsymbol{x}$ elements into AWGN components using the MPDR can provide an explanation as to why the diagonal approximation that decoupled the posterior in RSBL is a justified approximation. We mention here that despite the fact that assumed prior on $\boldsymbol{x}$ is not zero-mean anymore, numerical results suggest that the AWGN assumption in (11) is still valid. This is probably due to the fact that the MPDR estimator of x_n is suppressing the interference from all the other elements of $\boldsymbol{x}$. However, more rigorous analysis of the algorithm's performance and convergence is left for future work.

4.2. Bernoulli-Gaussian Priors

We now consider the case of knowing the prior on $\boldsymbol{x}$, without having to estimate its hyperparametres. An example of such a prior is the Bernoulli-Gaussian prior [6, 15] for the SSR problem, and the non-negative Bernoulli-Gaussian prior for the non-negative SSR problem [7]. First we apply the **MPDR estmation** step to find the estimates for r_n and τ_{r_n} using (11) and (12). Then we exploit the similarity between the decoupling of $\boldsymbol{x}$ into AWGN corrupted x_ns in the proposed MPDR formulation and in the generalized approximate message passing (GAMP) algorithm [11], and so we will not rederive $\mathbb{E}\{x_n|r_n\}$ or var$\{x_n|r_n\}$ for the **MMSE estimation**. Instead we will use the GAMPMatlab package directly from [16] to perform these estimates. Once the estimates are obtained, the **power updates** step is performed, where p_ns are updated according to (15). Based on the power updates another iteration is performed until convergence. we will refer to the algorithm by Bernoulli-Gaussian MPDR (BG-MPDR) and non-negative Bernoulli-Gaussian MPDR (NN-BG-MPDR). The performance of the NN-BG-MPDR algorithm is studied in section 6.

5. GAMP BASED LOW COMPLEXITY ALGORITHMS

A low complexity implementation of the SBL algorithm was proposed in [17]. The algorithm used the GAMP algorithm to efficiently implement the E-step, and was called Gaussian GAMP SBL (GGAMP-SBL). When a non-Gaussian prior is used with GAMP, the algorithm faces convergence issues for non-i.i.d.-Gaussian $\boldsymbol{A}$ [18–20]. Because it uses a parametrized Gaussian prior on $\boldsymbol{x}$, one of the main advantages of the proposed algorithm in [17], was that the GGAMP based E-step was gauranteed to converge to the correct LMMSE mean estimate $\hat{\boldsymbol{x}}$ and an approximate variance $\boldsymbol{\tau_x}$ even when the matrix A is not i.i.d-Gaussian. The convergence gaurantee was proven for Gaussian priors in [20], when a technique called damping is used [19, 20]. We use the connection between the LMMSE estimator and the MPDR to implement the **MPDR estimation** step in (11) and (12), using the GGAMP algorithm [20]. we do not restate the GGAMP LMMSE algorithm here due to space limitation, but details of the algorithm can be found in the E-step of table I in [17]. Assuming the LMMSE estimate μ_{n_L} and an approximate variance τ_{n_L} are obtained using GGAMP, r_n and τ_{r_n} can be found as follows:

$$\tau_{r_n} = (p_n\tau_{n_L})/(p_n - \tau_{n_L}) \quad r_n = (\mu_{n_L}\tau_{r_n})/(\tau_{n_L}) \tag{27}$$

Although the estimates of $\boldsymbol{\tau_r}$ are approximate [21], we still expect the algorithm's performance to be comparable to the higher complexity version. This was previously shown numerically in [17] in the case of GGAMP-SBL, and it will be shown in our numerical results section as well. This allows us to have a GAMP based algorthim that does not diverge for non-i.i.d.-Gaussian $\boldsymbol{A}$ and general priors, an outcome that was only available to Gaussian and Gaussian scale mixture priors [20].

5.1. GGAMP Based MPDR Approach Versus VAMP

The vector approximate message passing algorithm (VAMP) [12] was developed to overcome convergence issues experienced by AMP algorithms. Since VAMP can produce Bayesian estimates based on given priors, while achieving low complexity and improved convergence when $\boldsymbol{A}$ is non-i.i.d.-Gaussian, VAMP holds some similarities to the proposed approach in this section which we refer to by (GGAMP-MPDR). While more extensive studies need to be performed to compare the two algorithms, we mention here two main differences between VAMP and the GGAMP-MPDR:

1 VAMP can only handle i.i.d. priors, therefore it won't be able to have a scale mixture as its assumed prior on $\boldsymbol{x}$, making it unable to handle hierarchical priors.

2 VAMP requires a one time SVD decomposition at the beginning of the algorithm while the proposed GGAMP-MPDR does not. Because GGAMP-MPDR does not need to process the full matrix $\boldsymbol{A}$, the MPDR part can be divided into parallel sub-algorithms, each of which uses sub-columns of $\boldsymbol{A}$. This can be useful when the size of $\boldsymbol{A}$ is larger than memory limitations at hand.

6. NUMERICAL RESULTS

In this section we present some numerical results based on the non-negative SSR problem in section 4.1. The study is conducted with $N = 1000$, $M = 500$ and $K = 200$ non-zero entries, where the non-zero entries are drawn from a rectified Gaussian distribution with zero mean and unit variance. The study is conducted over a range of correlation of the columns of $\boldsymbol{A}$ to show how the new framework improves the convergence properties compared to the GAMP algorithm when $\boldsymbol{A}$ is non-i.i.d..-Gaussian. We use the normalized mean squared error (NMSE) between the original signal $\boldsymbol{x}$ and the recovered estimate $\hat{\boldsymbol{x}}$ as a performance measure for comparison.

The study compares a number of algorithms, where it considers the MPDR-RSBL algorithm proposed in 4.1, also referred to as the diagonal approximation DA-RSBL in [8]. In addition to that, we consider using the proposed MPDR framework with complete knowledge of the prior by using the NN-BG-MPDR algorithm from 4.2. We also consider the GAMP based low complexity versions of the two algorithms where we refer to them by MPDR-RSBL-LC and NN-BG-MPDR-LC. We compare these proposed algorithms against the GAMP Based non-negative Bernoulli-Gaussian algorithm from [7] referred to by NNGMAMP, and also against the VAMP based Bernoulli-Gaussian algorithm, where both algorithms assume full knowledge of the prior. For reference we also include a "Genie" algorithm, that is provided with the true support of $\boldsymbol{x}$ so it can perform an MMSE estimate on the non-zero entries.

There are a number of observations we can see in Fig. 1. First, when studying the convergence behavior against the column correlation of $\boldsymbol{A}$, we are particularly interested in the GAMP based low complexity algorithms, MPDR-RSBL-LC and BG-MPDR-LC. We can clearly see that both algorithms offer enhanced convergence compared to the GAMP based NNGMAMP algorithm which diverges at higher correlation. We compare the performance of BG-MPDR-LC to VAMP since they are both based on the same prior. We notice that BG-MPDR-LC is able to offer similar, and in some cases better performance compared to VAMP.

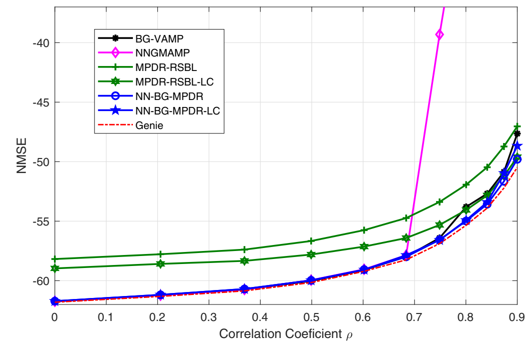

Fig. 1. NMSE Versus Column Correlation Comparison

We can also see that the lower complexity approximations of the algorithms provide comparable results to the full complexity ones. Comparing BG-MPDR-LC with BG-MPDR, there is a slight degradation in performance at high correlations. On the other hand, the Low complexity MPDR-RSBL-LC algorithm is performing slightly better than the full complexity MPDR-RSBL.

Finally, we notice how the full knowledge of the prior gives an advantage in performance over the hierarchical model that assumes no knowledge of the prior.

We also highlight the complexity reduction offered by the GGAMP implementation by plotting the runtimes of different algorithms. From Fig. 2 we can see that the GGAMP-MPDR implementation offers significant runtime reduction compared to the full MPDR algorithms. Based on previous results in [17], this reduction is expected to become more significant as the problem size increases. The runtimes of the two low complexity algorithms are comparable to VAMP and are slightly better than NNGMAMP.

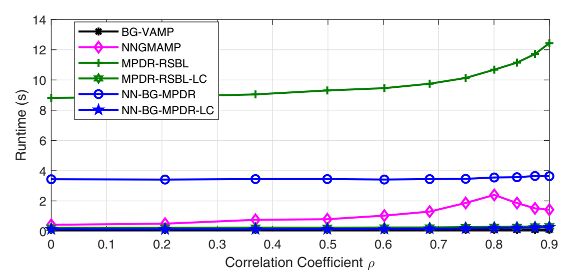

Fig. 2. Runtime Versus Column Correlation Comparison

7. CONCLUSIONS

In this paper we presented a different perspective on the SBL algorithm. The new perspective uses an MPDR estimator to decouple the measurements into AWGN components. With this perspective we were able to use the MPDR concept with more general priors. We were able to use the relationship between the MPDR and the LMMSE estimator to introduce lower complexity algorithms, resulting in a GAMP based algorithm with improved convergence. Future work should include a more extensive comparison of the low complexity algorithm to VAMP to better understand the similarities and differences.

8. REFERENCES

[1] Shihao Ji, Ya Xue, and Lawrence Carin, "Bayesian compressive sensing," *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2346–2356, 2008.

[2] S Derin Babacan, Rafael Molina, and Aggelos K Katsaggelos, "Bayesian compressive sensing using Laplace priors," *IEEE Transactions on Image Processing*, vol. 19, no. 1, pp. 53–63, 2010.

[3] Dror Baron, Shriram Sarvotham, and Richard G Baraniuk, "Bayesian compressive sensing via belief propagation," *IEEE Transactions on Signal Processing*, vol. 58, no. 1, pp. 269–280, 2010.

[4] Ritwik Giri and Bhaskar D Rao, "Type I and type II Bayesian methods for sparse signal recovery using scale mixtures," *arXiv preprint arXiv:1507.05087*, 2015.

[5] D. P. Wipf and B. D. Rao, "Sparse Bayesian learning for basis selection," *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2153–2164, Aug 2004.

[6] Jeremy P Vila and Philip Schniter, "Expectation-maximization Gaussian-mixture approximate message passing," *IEEE Transactions on Signal Processing*, vol. 61, no. 19, pp. 4658–4672, 2013.

[7] Jeremy P Vila and Philip Schniter, "An empirical-Bayes approach to recovering linearly constrained non-negative sparse signals," *IEEE Transactions on Signal Processing*, vol. 62, no. 18, pp. 4689–4703, 2014.

[8] Alican Nalci, Igor Fedorov, and Bhaskar D Rao, "Rectified Gaussian scale mixtures and the sparse non-negative least squares problem," *arXiv preprint arXiv:1601.06207*, 2016.

[9] Maher Al-Shoukairi and Bhaskar D Rao, "A GAMP based algorithm with hierarchical priors for recovering non-negative sparse signals," in *Signals, Systems, and Computers, 2017 51st Asilomar Conference on*. IEEE, 2017, pp. 435–439.

[10] Harry L Van Trees, *Optimum array processing: Part IV of detection, estimation, and modulation theory*, John Wiley & Sons, 2004.

[11] S. Rangan, "Generalized approximate message passing for estimation with random linear mixing," in *Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on*, July 2011, pp. 2168–2172.

[12] Philip Schniter, Sundeep Rangan, and Alyson K Fletcher, "Vector approximate message passing for the generalized linear model," in *Signals, Systems and Computers, 2016 50th Asilomar Conference on*. IEEE, 2016, pp. 1525–1529.

[13] Tarik Yardibi, Jian Li, Petre Stoica, Ming Xue, and Arthur B Baggeroer, "Source localization and sensing: A nonparametric iterative adaptive approach based on weighted least squares," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 46, no. 1, 2010.

[14] Pierre Garrigues and Bruno A Olshausen, "Group sparse coding with a Laplacian scale mixture prior," in *Advances in neural information processing systems*, 2010, pp. 676–684.

[15] Philip Schniter, "Turbo reconstruction of structured sparse signals," in *Information Sciences and Systems (CISS), 2010 44th Annual Conference on*. IEEE, 2010, pp. 1–6.

[16] "Generalized approximate message passing," SourceForge.net project GAMPmatlab, available on-line at http://gampmatlab.sourceforge.net/.

[17] Maher Al-Shoukairi, Philip Schniter, and Bhaskar D Rao, "A GAMP based low complexity sparse Bayesian learning algorithm," *IEEE Transactions on Signal Processing*, vol. 66, no. 2, pp. 294–308, 2018.

[18] F. Caltagirone, L. Zdeborova, and F. Krzakala, "On convergence of approximate message passing," in *Information Theory (ISIT), 2014 IEEE International Symposium on*, June 2014, pp. 1812–1816.

[19] J. Vila, P. Schniter, S. Rangan, F. Krzakala, and L. Zdeborov, "Adaptive damping and mean removal for the generalized approximate message passing algorithm," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, April 2015, pp. 2021–2025.

[20] S. Rangan, P. Schniter, and A. Fletcher, "On the convergence of approximate message passing with arbitrary matrices," in *Information Theory (ISIT), 2014 IEEE International Symposium on*, June 2014, pp. 236–240.

[21] Sundeep Rangan, Philip Schniter, Erwin Riegler, Alyson Fletcher, and Volkan Cevher, "Fixed points of generalized approximate message passing with arbitrary matrices," in *Information Theory Proceedings (ISIT), 2013 IEEE International Symposium on*. IEEE, 2013, pp. 664–668.