

Bayesian Linear Regression with Cauchy Prior and Its Application in Sparse MIMO Radar

Jun Li, Member, IEEE
NXP Semiconductors, San Jose, CA, USA

Ryan Wu, Member, IEEE
NXP Semiconductors, San Jose, CA, USA

I-Tai Lu, Senior Member, IEEE
New York University, NY, USA

Dongyin Ren, Member, IEEE
NXP Semiconductors, San Jose, CA, USA

Abstract—In this paper, a sparse signal recovery algorithm using Bayesian linear regression with Cauchy prior (BLRC) is proposed. Utilizing an approximate expectation maximization (AEM) scheme, a systematic hyper-parameter updating strategy is developed to make BLRC practical in highly dynamic scenarios. Remarkably, with a more compact latent space, BLRC not only possesses essential features of the well-known sparse Bayesian learning (SBL) and iterative reweighted l_2 (IR- l_2) algorithms but also outperforms them. Using sparse array (SPA) and coprime array (CPA), numerical analyses are first performed to show the superior performance of BLRC under various noise levels, array sizes, and sparsity levels. Applications of BLRC to sparse multiple-input and multiple-output (MIMO) radar array signal processing are then carried out to show that the proposed BLRC can efficiently produce high-resolution images of the targets.

Index Terms—Automotive Radar, Sparse MIMO array, High-resolution Radar, Sparse Signal Recovery (SSR), Cauchy, Sparse Bayesian Learning (SBL), sparse array (SPA), coprime array (CPA), array signal processing.

Manuscript received XXXXX 00, 0000; revised XXXXX 00, 0000; accepted XXXXX 00, 0000.

This work was supported by NXP Semiconductors, CA, USA
(Corresponding author: Jun Li)

Author's addresses: Jun Li, Ryan Wu, and Dongyin Ren are with NXP Semiconductors, San Jose, CA, USA. Jun Li was formerly with NYU WIRELESS and the Department of Electrical and Computer Engineering, Tandon School of Engineering, New York University, NY, USA (e-mail: j17333@nyu.edu; ryan.wu@nxp.com; dongyin.ren@nxp.com). I-Tai Lu is with the Department of Electrical and Computer Engineering, Tandon School of Engineering, New York University, NY, USA (e-mail: itl211@nyu.edu).

I. INTRODUCTION

With the development of modern advanced driver-assistance systems (ADAS) and autonomous driving (AD) applications, accurate perception and interpretation of the surrounding environment are highly desirable for automotive radar systems. To satisfy the stringent perception and interpretation requirement, high-resolution automotive radar is being developed to provide point cloud of the surrounding environment in four-dimensions (4D), i.e., range, Doppler, and azimuth and elevation angles [1].

In modern automotive radar system, the Size, Weight, Power, and Cost (SWaP-C) requirement needs to be satisfied. For instance, the size of automotive radar sensor array has to be small enough to be placed behind the vehicle bumper. Thus, it is obvious that the resolution of automotive radar cannot increase easily due to the size constraint. To break loose from this constraint, MIMO technology [2]–[4] is used in the state-of-the-art high-resolution automotive radar to increase the resolution of the system when the number of physical antennas is fixed. For a 3-transmitter/4-receiver MIMO radar system, one may construct a virtual array of 12 elements using only 7 physical antennas. To further improve the resolution, sparse array designs [5]–[7] are often coupled with MIMO virtual array approach to increase the effective array aperture while reducing the hardware cost and mutual coupling among antennas [8]–[10]. Fig. 1 depicts a sparse array of 12 virtual elements spanning an aperture of the size of 28 elements. Sub- 1° angular resolution has been achieved in practical 4D imaging radar systems by employing MIMO sparse array approach.

One main challenge in sparse array design is to deal with angular ambiguity and sidelobes. Moreover, in highly dynamic scenarios, angle estimation has to be conducted based on single snapshot measurement [11]. Many conventional angle estimation methods fail to operate under such conditions [12]–[14]. In this paper, we focus on high-resolution direction-of-arrival (DoA) estimation [15] with MIMO sparse array using only a single snapshot in automotive radar applications.

In a typical automotive frequency-modulated continuous-wave (FMCW) radar processing chain, the targets are first separated in range and Doppler domains. Due to the wide bandwidth feature of current automotive radars, the number of targets that falls in the same range-Doppler bin is small [2]. Therefore, high-resolution results in angle space can be obtained by solving a sparse signal recovery (SSR) problem [16]. Note that sparse signal recovery provides a framework to effectively handle sparse signals encountered in many signal processing applications, such as spectral analysis [17], sparse channel estimation [18], and multiuser detection [19]. It is also a common problem found in machine learning, such as feature selection [20], subspace clustering [21], and sparse representation [22].

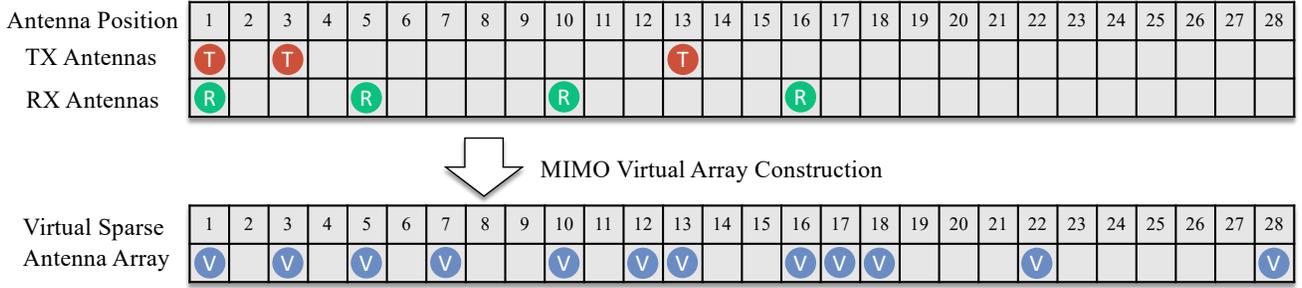


Fig. 1. Forming a sparse array of 12 virtual elements spanning an aperture of the size of 28 elements using a 3-transmitter/4-receiver MIMO radar system.

The general linear model for the sparse signal recovery problem [23] can be presented as:

$$\mathbf{y} = \mathbf{A}\mathbf{c} + \boldsymbol{\epsilon} \quad (1)$$

where \mathbf{y} represents the M observed measurements, \mathbf{c} is the N unknown parameters with $N \gg M$, \mathbf{A} is the basis matrix, and $\boldsymbol{\epsilon}$ is the noise vector. Equation (1) is under-determined and generally has infinite numbers of solutions. Various approaches have been proposed to find the sparsest solution of (1), including but not limited to, Greedy-based approaches [24]–[26], Convex relaxation approaches [27]–[29], Iterative reweighted approaches [30]–[33], and Bayesian linear regression approaches [34]–[38].

Among the aforementioned approaches, Bayesian linear regression approaches are superior in scalability and interpretability. There are different ways to classify these approaches. Here, we consider two related frameworks: the maximum a posteriori (MAP) and the hierarchical frameworks. The main difference between these two frameworks lies in how the sparsity-inducing prior is used. In the latter, the prior is used in a hierarchical manner, while in the former, it is used directly. Two particular approaches are of particular interest here. The first approach is the Cauchy-Gaussian (CG) approach in [37], which also belongs to the iterative reweighted algorithms from the implementation perspective [39]. It uses the MAP framework with the sparsity-inducing Cauchy prior

$$p(c_i; \gamma) = \frac{1}{\pi\gamma(1 + \frac{c_i^2}{\gamma^2})} \quad (2)$$

where γ in (2) is the scale parameter. CG is very effective in suppressing spurious targets if its hyper-parameters are chosen properly. However, the problem of solving the posterior in CG is intractable, so systematic strategies for updating the CG hyper-parameters are missing. Without such strategies, the optimal hyper-parameters can only be found by trial and error in a suboptimal way.

The second approach is the sparse Bayesian learning (SBL) approach which has been widely used in radar signal processing [40]–[42]. SBL uses the hierarchical framework with the conditional Gaussian prior [38] of the unknown vector \mathbf{c} :

$$p(\mathbf{c}|\boldsymbol{\tau}) = \frac{1}{\sqrt{(2\pi)^N |\boldsymbol{\Sigma}^{-1}|}} \exp\left(-\frac{1}{2}\mathbf{c}^T \boldsymbol{\Sigma} \mathbf{c}\right) \quad (3)$$

where the diagonal matrix $\boldsymbol{\Sigma} \equiv \text{diag}(\boldsymbol{\tau})$ is the inverse of the covariance matrix of prior \mathbf{c} conditional on the $N \times 1$ hyper-parameter $\boldsymbol{\tau}$. Compared to the aforementioned algorithms, SBL excels in sparse signal recovery performance with uniform linear array and sufficient signal-to-noise ratio (SNR), and its parameters are self-tuned. However, recent studies have highlighted several areas in which SBL can be improved, particularly when sparse array measurement is used as input. Firstly, SBL involves N hyper-parameters in $\boldsymbol{\tau}$ to define N Gaussian distributions corresponding to the N elements of \mathbf{c} . As elements of $\boldsymbol{\tau}$ approach infinity at various speeds during the iterations, the condition number of the matrix to be inverted in SBL increases. Coupled with limited machine precision in real-world applications (e.g., single-precision computing environments in most automotive radar processors), this results in numerical errors and instability [38]. Thus, the iteration process has to be stopped before numerical error takes place. Secondly, the noise parameter σ_n estimated by SBL converges incorrectly resulting in suboptimal solution [43]–[45] and negatively impacting the sparse recovery performance in practice [46]. Lastly, because of the above-mentioned reasons, SBL tends to produce more spurious solutions [38], [47], [48], which impairs its usefulness when high dynamic range is required. This issue becomes increasingly pronounced as the sparse array becomes sparser. It is essential to highlight that the primary focus of this work is on applications that employ sparse array measurements as input. Numerous SBL applications also employ a uniform linear array as input. For instance, reference [49] offers a valuable example of the SBL application that utilizes a uniform linear array.

In this paper, a sparse signal recovery algorithm, Bayesian linear regression with Cauchy prior (BLRC), is proposed to improve the performance of CG and SBL, specifically when sparse array measurements serve as input data. Like CG, BLRC has only one hyper-parameter γ for all N prior distributions for the N elements of the unknown \mathbf{c} . Unlike CG, we use Laplace approximation in BLRC to approximate the intractable posterior locally so that the hyper-parameters can be learned from the observed data through the proposed approximate expectation maximization (AEM) scheme. In this sense, BLRC is a

significant improvement over the CG approach because BLRC provides a systematic updating scheme for the hyper-parameters which is absent from the CG approach.

BLRC can also be considered as a significant improvement over SBL in several aspects. Like SBL, BLRC is a Bayesian linear regression approach. Both BLRC and SBL have the same iterative updating steps for the solution and hyper-parameters. However, there are some key differences between BLRC and SBL. Primarily, only two hyper-parameters need to be handled in BLRC, while the number of hyper-parameters to be handled is large for SBL. This is because BLRC uses a long-tailed but proper Cauchy distribution that requires only one scale parameter to characterize all N elements of the unknown \mathbf{c} . On the contrary, SBL uses the Gaussian distribution as the intermediate conditional prior where its true prior is improper [38]. Accordingly, SBL needs N variances to characterize the N elements of the unknown \mathbf{c} . As the number of unknowns is typically large, the number of hyper-parameters for SBL is usually large. Therefore, BLRC is superior to SBL in latent space compactness, which is a major advantage of BLRC over SBL.

Using randomly placed Sparse Array (SPA) and Coprime Array (CPA), it is shown in this paper by extensive numerical simulations that BLRC has the following advantages over SBL: numerical robustness, spurious targets suppression, target resolution, system flexibility, noise tolerance, and noise variance estimation. This is originated from the fact that one scale parameter γ is sufficient to describe the Cauchy prior of BLRC while N hyper-parameters $\{\tau_i\}$ are required to describe the conditional Gaussian priors of SBL. Furthermore, the performance of BLRC and SBL are compared using sparse automotive radar image recovery examples. Using physical optics (PO) approximation, MIMO radar signals are generated by our automotive imaging radar simulator. It is shown that BLRC outperforms SBL by producing clearer radar images with better resolution.

The rest of this paper is organized as follows. In Section II, we give a summary of the Bayesian linear regression model, CG approach, and SBL approach. In Section III, we use AEM to develop the proposed BLRC approach and discuss its computational efficiency. In Section IV, comparisons under variational interpretation are provided to explain the improvement of BLRC over CG. In Section V, comparisons based on the iterative reweighted l_2 (IR- l_2) interpretation are provided to explain the superior performance of BLRC over SBL. In Section VI, numerical analyses including initial values of hyper-parameters, numerical stability, convergent properties, spurious targets suppression, noise variance estimation, resolution and sensitivities study are provided. In Section VII, we introduce the automotive image radar signal processing application and our automotive radar signal simulator. The direction of arrival (DoA) estimation result using coprime array is shown to demonstrate the superiority of our proposed BLRC approach. At last, the conclusion is made in Section VIII.

II. Sparse Bayesian Linear Regression Approaches

In this section, we introduce the MAP framework and the hierarchical framework for sparse Bayesian linear regression approaches. Then we briefly summarize the Cauchy Gaussian (CG) algorithm, which uses the MAP framework with Cauchy prior, and the well-known Sparse Bayesian Learning (SBL), which uses the hierarchical framework.

A. Sparse Bayesian Linear Regression

Using the general linear model for the sparse signal recovery in (1) and assuming the M noise elements of ϵ are independent zero-mean Gaussian random variables with the same variance σ_n^2 , the likelihood function of the observation \mathbf{y} is:

$$p(\mathbf{y}|\mathbf{c}, \sigma_n) = \left(\frac{1}{\sqrt{2\pi\sigma_n^2}}\right)^M \exp\left(-\frac{\|\mathbf{y} - \mathbf{A}\mathbf{c}\|^2}{2\sigma_n^2}\right) \quad (4)$$

Using the MAP framework, we are to find the mode of the following posterior distribution:

$$\mathbf{c} = \arg \max_{\mathbf{c}} p(\mathbf{c}|\mathbf{y}) = \arg \max_{\mathbf{c}} \ln p(\mathbf{y}|\mathbf{c})p(\mathbf{c}) \quad (5)$$

The prior distribution $p(\mathbf{c})$ is a sparsity-inducing prior which can typically be written in the following general form:

$$p(\mathbf{c}) \propto \exp\left(-\frac{1}{2} \sum_{i=1}^N g(c_i)\right) \quad (6)$$

where $g(c_i) = h(c_i^2)$, and it has been shown that the choice of a concave and nondecreasing $h(c_i)$ in $[0, \infty)$ could lead to a sparse solution of \mathbf{c} [50]. Then (5) leads to the canonical regularized optimization problem:

$$\mathbf{c} = \arg \min_{\mathbf{c}} \|\mathbf{y} - \mathbf{A}\mathbf{c}\|^2 + \sigma_n^2 \sum_{i=1}^N g(c_i) \quad (7)$$

For example, $h(c_i) = \sqrt{c_i}$ results in the Laplacian prior which is a well-known sparsity-inducing prior, and the corresponding $g(c_i) = |c_i|$ leads to the well-known regularized optimization problem with l_1 norm.

The hierarchical framework is proposed to form the full Bayesian inference by introducing extra hyper-parameters (e.g., τ in SBL [38]). Then, not only \mathbf{c} , but also the hyper-parameter τ and noise variance σ_n^2 are all inferred from observation \mathbf{y} :

$$\begin{aligned} \{\tilde{\mathbf{c}}, \tilde{\tau}, \tilde{\sigma}_n\} &= \arg \max_{\mathbf{c}, \tau, \sigma_n} p(\mathbf{c}, \tau, \sigma_n|\mathbf{y}) \\ &= \arg \max_{\mathbf{c}, \tau, \sigma_n} p(\mathbf{c}|\mathbf{y}, \tau, \sigma_n)p(\tau, \sigma_n|\mathbf{y}) \end{aligned} \quad (8)$$

To solve the above inference problem, several algorithms are proposed, such as the evidence maximization (also known as type-II maximum likelihood) [38], expectation maximization (EM) [51], and variational Bayes (VB) approximation [52].

Basically, (8) can be solved by the following iterative approach. Firstly, the hyper-parameter $\tilde{\tau}$ and $\tilde{\sigma}_n$

are learned based on observation \mathbf{y} using the following optimization:

$$\begin{aligned} \{\tilde{\boldsymbol{\tau}}, \tilde{\sigma}_n\} &= \arg \max_{\boldsymbol{\tau}, \sigma_n} p(\boldsymbol{\tau}, \sigma_n | \mathbf{y}) \\ &= \arg \max_{\boldsymbol{\tau}, \sigma_n} p(\mathbf{y} | \boldsymbol{\tau}, \sigma_n) \\ &= \arg \max_{\boldsymbol{\tau}, \sigma_n} \int p(\mathbf{y} | \mathbf{c}, \sigma_n) p(\mathbf{c} | \boldsymbol{\tau}) d\mathbf{c} \end{aligned} \quad (9)$$

with the assumption that $\boldsymbol{\tau}$ and σ_n are ‘‘improper’’ hyper-parameters with flat distributions ($p(\boldsymbol{\tau}) \propto 1$ and $p(\sigma_n) \propto 1$) or non-informative Jeffreys distributions (uniform under logarithmic scale).

Secondly, the estimation of $\tilde{\mathbf{c}}^{(k+1)}$ of \mathbf{c} for the $(k+1)^{th}$ iteration is obtained from maximizing the posterior distribution:

$$\tilde{\mathbf{c}}^{(k+1)} = \arg \max_{\mathbf{c}} p(\mathbf{c} | \mathbf{y}, \boldsymbol{\tau}(\tilde{\mathbf{c}}^{(k)}), \tilde{\sigma}_n(\mathbf{c}^{(k)})) \quad (10)$$

If the posterior distribution is Gaussian, the solution of (10) is the mean of the posterior distribution. Using (9) and (10), the \mathbf{c} and $\{\boldsymbol{\tau}, \sigma_n\}$ are updated in an iterative manner until a convergence criterion is satisfied.

Note that the ‘‘true’’ prior distribution $p(\mathbf{c})$ can be obtained from the parameterized prior $p(\mathbf{c} | \boldsymbol{\tau})$ in the hierarchical framework:

$$p(\mathbf{c}) = \int p(\mathbf{c} | \boldsymbol{\tau}) p(\boldsymbol{\tau}) d\boldsymbol{\tau} \quad (11)$$

It is shown that the ‘‘true’’ prior $p(\mathbf{c})$ is actually a sparsity-inducing prior, and this kind of representation is also known as scale mixtures [53].

B. Cauchy Gaussian (CG) Approach

The CG approach [37] uses the MAP framework with the Cauchy prior given in (2). Assuming the scale parameter γ and the noise standard deviation σ_n are given (which are denoted as $\hat{\gamma}$ and $\hat{\sigma}_n$, respectively), the MAP estimator for \mathbf{c} is to minimize the following cost function.

$$J_{cg}(\mathbf{c}) = \|\mathbf{y} - \mathbf{A}\mathbf{c}\|^2 + h_{cg}(\mathbf{c}) \quad (12)$$

with the log-sum regularized term

$$h_{cg}(\mathbf{c}) = \hat{\sigma}_n^2 \sum_{i=1}^N 2 \ln \left(1 + \frac{c_i^2}{\hat{\gamma}^2} \right) \quad (13)$$

Omitting the derivations, \mathbf{c} is computed iteratively using steps described in Algorithm 1.

C. Sparse Bayesian Learning (SBL)

In sparse Bayesian Learning (SBL), the prior is represented in a hierarchical framework. The conditional prior distribution of the unknown vector \mathbf{c} is Gaussian as defined in (3). Although there are different explanations and optimization algorithms for SBL (e.g., evidence maximization, expectation maximization, and variational Bayes), their updating strategies for the sparse vector \mathbf{c} are similar.

For convenience, let the i^{th} element of $\boldsymbol{\tau}$, $\tau_i = 1/\sigma_i^2$, where σ_i^2 is the variance of c_i , the i^{th} element of \mathbf{c} . Additionally, let $\boldsymbol{\Gamma}$ be the covariance matrix of the posterior

Algorithm 1: Cauchy Gaussian (CG) Approach

Input: \mathbf{y} , \mathbf{A} , $\hat{\sigma}_n$, $\hat{\gamma}$, K
Output: $\hat{\mathbf{c}}$
1 Initialization: $\hat{\mathbf{c}}^{(0)}$
2 for $k = 1 : K$ **do**
3 $\hat{\mathbf{Q}}^{(k)} = \text{diag} \left(1 + \frac{(\hat{c}_1^{(k-1)})^2}{\hat{\gamma}^2}, 1 + \frac{(\hat{c}_2^{(k-1)})^2}{\hat{\gamma}^2}, \dots, 1 + \frac{(\hat{c}_N^{(k-1)})^2}{\hat{\gamma}^2} \right)^{-1}$
4 $\hat{\mathbf{c}}^{(k)} = [2\hat{\sigma}_n^2 \hat{\mathbf{Q}}^{(k)} + \mathbf{A}^T \mathbf{A}]^{-1} \mathbf{A}^T \mathbf{y}$
5 if $\hat{\mathbf{c}}^{(k)}$ converges, $k = K$ and $\hat{\mathbf{c}}^{(K)} = \hat{\mathbf{c}}^{(k)}$
6 end
7 return $\hat{\mathbf{c}} = \hat{\mathbf{c}}^{(K)}$

of \mathbf{c} conditional on \mathbf{y} . Denote the estimate of \mathbf{c} , $\boldsymbol{\tau}$, $\boldsymbol{\Sigma}$, $\boldsymbol{\Gamma}$ and σ_n as $\tilde{\mathbf{c}}$, $\tilde{\boldsymbol{\tau}}$, $\tilde{\boldsymbol{\Sigma}}$, $\tilde{\boldsymbol{\Gamma}}$ and $\tilde{\sigma}_n$, respectively. The typical implementation of SBL as shown in [35] and [38] for finding $\tilde{\mathbf{c}}$ is summarized in Algorithm 2.

Algorithm 2: Sparse Bayesian Learning (SBL)

Input: \mathbf{y} , \mathbf{A} , K , $\text{size}(\mathbf{A})=[M, N]$
Output: $\tilde{\mathbf{c}}$
1 Initialization: $\tilde{\sigma}_n^{(0)} \leftarrow 0.1$, $\tilde{\boldsymbol{\tau}}^{(0)} \leftarrow \mathbf{1}$, $k \leftarrow 0$
2 for $k = 1 : K$ **do**
3 $\tilde{\boldsymbol{\Sigma}}^{(k)} = \text{diag} \left(\tilde{\tau}_1^{(k-1)}, \tilde{\tau}_2^{(k-1)}, \dots, \tilde{\tau}_N^{(k-1)} \right)$
4 $\tilde{\boldsymbol{\Gamma}}^{(k)} = \left(\frac{1}{(\tilde{\sigma}_n^{(k-1)})^2} \mathbf{A}^T \mathbf{A} + \tilde{\boldsymbol{\Sigma}}^{(k)} \right)^{-1}$
5 $\tilde{\mathbf{c}}^{(k)} = \frac{1}{(\tilde{\sigma}_n^{(k-1)})^2} \tilde{\boldsymbol{\Gamma}}^{(k)} \mathbf{A}^T \mathbf{y}$
6 $(\tilde{\sigma}_n^{(k)})^2 = \frac{\|\mathbf{y} - \mathbf{A}\tilde{\mathbf{c}}^{(k)}\|^2}{M - \text{Tr}(\mathbf{I} - \tilde{\boldsymbol{\Gamma}}^{(k)} \tilde{\boldsymbol{\Sigma}}^{(k)})}$
7 for $i = 1 : N$ **do**
8 $\tilde{\tau}_i^{(k)} = \frac{1 - \tilde{\tau}_i^{(k-1)} \tilde{\Gamma}_{ii}^{(k)}}{(\tilde{c}_i^{(k)})^2}$
9 end
10 if $\tilde{\mathbf{c}}^{(k)}$ converges, $k = K$ and $\tilde{\mathbf{c}}^{(K)} = \tilde{\mathbf{c}}^{(k)}$
11 end
12 return $\tilde{\mathbf{c}} = \tilde{\mathbf{c}}^{(K)}$

III. Bayesian Linear Regression with Cauchy Prior (BLRC)

When the parameters σ_n and γ are given for the CG approach, one can directly compute \mathbf{c} and \mathbf{Q} iteratively as shown in Algorithm 1. However, σ_n and γ are usually unknown for the CG approach and are typically found by trial and error under practical conditions. In this section, using the Laplace approximation to obtain a local Gaussian approximation of the posterior distribution of the random vector \mathbf{c} , we present a novel Approximate Expectation Maximization (AEM) algorithm to update these parameters automatically. Benefiting from the general convergence property of EM algorithm, the proposed algorithm is to converge to a local minimum. The combination of the CG approach for estimating \mathbf{c} and \mathbf{Q} and the AEM approach for updating σ_n and γ is named as the

Bayesian Linear Regression with Cauchy Prior (BLRC) approach.

A. Expectation Maximization (EM) Formulation

To obtain the maximum likelihood estimation of parameter $\{\gamma, \sigma_n\}$, we need to maximize $\ln\{p(\mathbf{y}; \gamma, \sigma_n)\}$:

$$\{\hat{\gamma}, \hat{\sigma}_n\} = \arg \max_{\{\gamma, \sigma_n\}} \ln\{p(\mathbf{y}; \gamma, \sigma_n)\} \quad (14)$$

Since $p(\mathbf{y}; \gamma, \sigma_n)$ does not have a closed-form expression with the Cauchy prior model, we propose to maximize its lower bound (i.e., the generalized EM formulation [54]). Then the estimator for $\{\gamma, \sigma_n\}$ in the k^{th} iteration can be expressed as

$$\{\hat{\gamma}^{(k)}, \hat{\sigma}_n^{(k)}\} = \arg \max_{\gamma, \sigma_n} E_{\mathbf{c}|\mathbf{y}}[\ln p(\mathbf{y}, \mathbf{c}; \gamma, \sigma_n)] \quad (15)$$

where $E_{\mathbf{c}|\mathbf{y}}$ is the expectation with respect to the random variable \mathbf{c} given the observation \mathbf{y} and the parameters from the previous iteration $\{\gamma^{(k-1)}, \sigma_n^{(k-1)}\}$

For convenience, the parameter γ in Cauchy prior distribution is mapped to a new parameter τ with $\gamma = \frac{1}{\sqrt{\tau}}$ for further mathematical manipulations. Noting that $p(\mathbf{y}, \mathbf{c}; \tau, \sigma_n) = p(\mathbf{y}|\mathbf{c}; \sigma_n)p(\mathbf{c}; \tau)$ and using the likelihood distribution of observation $p(\mathbf{y}|\mathbf{c}; \sigma_n)$ in (4) and the Cauchy prior distribution $p(\mathbf{c}; \tau)$ in (2), the expectation in (15) can be expressed as

$$\begin{aligned} & E_{\mathbf{c}|\mathbf{y}}[\ln p(\mathbf{y}, \mathbf{c}; \tau, \sigma_n)] \\ &= -\frac{M}{2} \ln(2\pi\sigma_n^2) - \frac{1}{2\sigma_n^2} E_{\mathbf{c}|\mathbf{y}}[\|\mathbf{y} - \mathbf{A}\mathbf{c}\|^2] \\ & \quad - N \ln\left(\frac{\pi}{\sqrt{\tau}}\right) - \sum_{i=1}^N E_{\mathbf{c}|\mathbf{y}}[\ln(1 + \tau c_i^2)] \end{aligned} \quad (16)$$

where the computation of the original expectation can be carried out by computing the two new easier expectations. Define the mean $E_{\mathbf{c}|\mathbf{y}}[\mathbf{c}]$ as $\hat{\mathbf{c}}$ and the covariance $E_{\mathbf{c}|\mathbf{y}}[(\mathbf{c} - \hat{\mathbf{c}})(\mathbf{c} - \hat{\mathbf{c}})^T]$ as $\hat{\mathbf{\Gamma}}$. The first expectation in (16) can be further simplified following Appendix A:

$$E_{\mathbf{c}|\mathbf{y}}[\|\mathbf{y} - \mathbf{A}\mathbf{c}\|^2] = \|\mathbf{y} - \mathbf{A}\hat{\mathbf{c}}\|^2 + \text{Tr}(\mathbf{A}^T \mathbf{A} \hat{\mathbf{\Gamma}}) \quad (17)$$

Regarding the second expectation in (16), it is necessary to simplify $\ln(1 + \tau c_i^2)$ in order to get a closed-form expression. Consider the second-order Taylor series expansion of $\ln(1 + \tau c_i^2)$ at $\tau c_i^2 = E_{\mathbf{c}|\mathbf{y}}[\tau c_i^2]$:

$$\begin{aligned} \ln(1 + \tau c_i^2) &\approx \ln(1 + E_{\mathbf{c}|\mathbf{y}}[\tau c_i^2]) + \frac{\tau c_i^2 - E_{\mathbf{c}|\mathbf{y}}[\tau c_i^2]}{(1 + E_{\mathbf{c}|\mathbf{y}}[\tau c_i^2]) \cdot 1!} \\ & \quad - \frac{(\tau c_i^2 - E_{\mathbf{c}|\mathbf{y}}[\tau c_i^2])^2}{(1 + E_{\mathbf{c}|\mathbf{y}}[\tau c_i^2])^2 \cdot 2!} \end{aligned} \quad (18)$$

Taking expectation on (18), we have:

$$E_{\mathbf{c}|\mathbf{y}}[\ln(1 + \tau c_i^2)] \approx \ln(1 + \tau E_{\mathbf{c}|\mathbf{y}}[c_i^2]) - \frac{\tau^2 \text{Var}_{\mathbf{c}|\mathbf{y}}[c_i^2]}{2(1 + \tau E_{\mathbf{c}|\mathbf{y}}[c_i^2])^2} \quad (19)$$

For convenience, define $\eta_i = c_i^2$, $\hat{\eta}_i = E_{\mathbf{c}|\mathbf{y}}[\eta_i]$ and $\hat{\xi}_i = \text{Var}_{\mathbf{c}|\mathbf{y}}[\eta_i]$. Then (19) becomes

$$E_{\mathbf{c}|\mathbf{y}}[\ln(1 + \tau \eta_i)] \approx \ln(1 + \tau \hat{\eta}_i) - \frac{\tau^2 \hat{\xi}_i}{2(1 + \tau \hat{\eta}_i)^2} \quad (20)$$

B. Approximate Expectation

Closed-form expressions of the mean and variance of \mathbf{c} (i.e., $\hat{\mathbf{c}}$ and $\hat{\mathbf{\Gamma}}$, respectively) in (17) and the mean and variance of η_i (i.e., $\hat{\eta}_i$ and $\hat{\xi}_i$, respectively) in (20) are difficult to derive since there is no explicit expression of the posterior distribution $p(\mathbf{c}|\mathbf{y}; \tau, \sigma_n)$.

Although Cauchy distribution is non-log-concave, the Hessian matrix of the posterior can be shown to be negative-definite everywhere when being solved in an iterative reweighted manner. Thus, the iteratively solved posterior $p(\mathbf{c}|\mathbf{y})$ is log-concave and unimodal. Then, the Laplace approximation, a general methodology that approximates a probability density function locally in terms of a Gaussian distribution, can be employed to approximate $p(\mathbf{c}|\mathbf{y}; \tau, \sigma_n)$.

Performing Taylor series expansion on the previously defined cost function in (12) around the mode $\hat{\mathbf{c}}$, we obtain:

$$J_{cg}(\mathbf{c}) \approx J_{cg}(\hat{\mathbf{c}}) + \frac{1}{2}(\mathbf{c} - \hat{\mathbf{c}})^T \hat{\mathbf{\Gamma}}^{-1}(\mathbf{c} - \hat{\mathbf{c}}) \quad (21)$$

where $\hat{\mathbf{c}}$ is given in line 4 of Algorithm 1 and

$$\hat{\mathbf{\Gamma}}^{-1} = \nabla_{\mathbf{c}} \nabla_{\mathbf{c}} J_{cg}(\mathbf{c}) \Big|_{\mathbf{c}=\hat{\mathbf{c}}} \quad (22)$$

Then, using the Laplace approximation, we obtain the local Gaussian approximation of the posterior distribution of the random vector \mathbf{c} from (21):

$$\begin{aligned} p(\mathbf{c}|\mathbf{y}; \tau, \sigma_n) &\propto p(\mathbf{y}|\mathbf{c}; \sigma_n)p(\mathbf{c}; \tau) \\ &\simeq \frac{1}{(2\pi)^{\frac{N}{2}} |\hat{\mathbf{\Gamma}}|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{c} - \hat{\mathbf{c}})^T \hat{\mathbf{\Gamma}}^{-1}(\mathbf{c} - \hat{\mathbf{c}})\right) \end{aligned} \quad (23)$$

Note that the covariance matrix $\hat{\mathbf{\Gamma}}$ in (23) can be obtained from (22). Since $\hat{\mathbf{Q}}$ in line 3 of Algorithm 1 is a function of $\hat{\mathbf{c}}$, we expressed $\hat{\mathbf{\Gamma}}^{(k)}$ of the k^{th} iteration in terms of the following iterative formula:

$$\hat{\mathbf{\Gamma}}^{(k)} = \left[\frac{1}{\sigma_n^2} \mathbf{A}^T \mathbf{A} + 2\tau \hat{\mathbf{Q}}^{(k)} \right]^{-1} \quad (24)$$

So we can rewrite the approximate posterior mean under Cauchy prior in line 4 of Algorithm 1 as:

$$\hat{\mathbf{c}}^{(k)} = \frac{1}{\sigma_n^2} \hat{\mathbf{\Gamma}}^{(k)} \mathbf{A}^T \mathbf{y} \quad (25)$$

Regarding the mean and variance of η_i (i.e., $\hat{\eta}_i$ and $\hat{\xi}_i$) in (20), we can use the identity:

$$\begin{aligned} \hat{\eta}_i &= E_{\mathbf{c}|\mathbf{y}}[c_i^2] = \text{Var}_{\mathbf{c}|\mathbf{y}}[c_i] + (E_{\mathbf{c}|\mathbf{y}}[c_i])^2 \\ \hat{\xi}_i &= \text{Var}_{\mathbf{c}|\mathbf{y}}[c_i^2] = E_{\mathbf{c}|\mathbf{y}}[c_i^4] - (E_{\mathbf{c}|\mathbf{y}}[c_i^2])^2 \end{aligned}$$

Note that $\text{Var}_{\mathbf{c}|\mathbf{y}}[c_i] = \hat{\mathbf{\Gamma}}_{ii}$, $(E_{\mathbf{c}|\mathbf{y}}[c_i])^2 = \hat{c}_i^2$, and $E[c_i^4] = \hat{c}_i^4 + 6\hat{c}_i^2 \hat{\mathbf{\Gamma}}_{ii} + 3\hat{\mathbf{\Gamma}}_{ii}^2$. Finally we obtain

$$\begin{aligned} \hat{\eta}_i &= \hat{\mathbf{\Gamma}}_{ii} + \hat{c}_i^2 \\ \hat{\xi}_i &= 4\hat{c}_i^2 \hat{\mathbf{\Gamma}}_{ii} + 2\hat{\mathbf{\Gamma}}_{ii}^2 \end{aligned} \quad (26)$$

C. Maximization

Setting the derivatives of (16) with respect to σ_n^2 and τ to zeros, one can obtain the maximization results for (15). Firstly, taking the derivative of (16) with respect to σ_n^2 and utilizing (17), we have

$$\frac{\partial E_{\mathbf{c}|\mathbf{y}}[\ln p(\mathbf{y}, \mathbf{c})]}{\partial(\sigma_n^2)} = -\frac{M}{2\sigma_n^2} + \frac{\|\mathbf{y} - \mathbf{A}\hat{\mathbf{c}}\|^2 + \text{Tr}(\mathbf{A}^T \mathbf{A} \hat{\mathbf{\Gamma}})}{2(\sigma_n^2)^2} \quad (27)$$

From (27), the estimate of σ_n^2 , denoted as $\hat{\sigma}_n^2$, can be obtained as shown in line 9 of Algorithm 3. Noting that the $\hat{\mathbf{c}}$ and $\hat{\mathbf{\Gamma}}$ in (27) are functions of $\hat{\sigma}_n^2$, so $\hat{\sigma}_n^2$ has to be solved iteratively.

Secondly, taking derivative of (16) with respect to τ and utilizing (19), we have

$$\frac{\partial E_{\mathbf{c}|\mathbf{y}}[\ln p(\mathbf{y}, \mathbf{c})]}{\partial \tau} = \frac{N}{2\tau} - \sum_{i=1}^N \frac{\hat{\eta}_i}{1 + \tau \hat{\eta}_i} + \sum_{i=1}^N \frac{\tau \hat{\xi}_i}{(1 + \tau \hat{\eta}_i)^3} \quad (28)$$

Thus, we can solve iteratively for the update of τ , denoted as $\hat{\tau}$, as shown in line 8 of Algorithm 3. In the proposed BLRC algorithm, the update of $\{\hat{\mathbf{c}}, \hat{\mathbf{\Gamma}}\}$ and the update of $\{\hat{\sigma}_n^2, \hat{\tau}\}$ are carried out iteratively, which is summarized in Algorithm 3.

Algorithm 3: Bayesian Linear Regression with Cauchy Prior (BLRC)

Input: \mathbf{y} , \mathbf{A} , K , $\text{size}(\mathbf{A})=[M,N]$

Output: $\hat{\mathbf{c}}$

1 **Initialization:** $\hat{\sigma}_n^{(0)} \leftarrow 0.1$, $\hat{\gamma}^{(0)} \leftarrow 1$, $\hat{\mathbf{c}}^{(0)}$

2 **for** $k = 1 : K$ **do**

3 $\hat{\mathbf{Q}}^{(k)} = \text{diag}\left(1 + \frac{(\hat{c}_1^{(k-1)})^2}{(\hat{\gamma}^{(k-1)})^2}, \dots, 1 + \frac{(\hat{c}_N^{(k-1)})^2}{(\hat{\gamma}^{(k-1)})^2}\right)^{-1}$

4 $\hat{\mathbf{\Gamma}}^{(k)} = \left[\frac{1}{(\hat{\sigma}_n^{(k-1)})^2} \mathbf{A}^T \mathbf{A} + \frac{2}{(\hat{\gamma}^{(k-1)})^2} \hat{\mathbf{Q}}^{(k)} \right]^{-1}$

5 $\hat{\mathbf{c}}^{(k)} = \frac{1}{(\hat{\sigma}_n^{(k-1)})^2} \hat{\mathbf{\Gamma}}^{(k)} \mathbf{A}^T \mathbf{y}$

6 $\hat{\eta}_i^{(k)} = \hat{\mathbf{\Gamma}}_{ii}^{(k)} + (\hat{c}_i^{(k)})^2$

7 $\hat{\xi}_i^{(k)} = 4(\hat{c}_i^{(k)})^2 \hat{\mathbf{\Gamma}}_{ii}^{(k)} + 2(\hat{\mathbf{\Gamma}}_{ii}^{(k)})^2$

8 $(\hat{\gamma}^{(k)})^2 = \frac{2}{N} \left\{ \sum_{i=1}^N \left[\frac{\hat{\eta}_i^{(k)}}{1 + \frac{1}{(\hat{\gamma}^{(k-1)})^2}} - \frac{\frac{1}{(\hat{\gamma}^{(k-1)})^2} \hat{\xi}_i^{(k)}}{(1 + \frac{1}{(\hat{\gamma}^{(k-1)})^2} \hat{\eta}_i^{(k)})^3} \right] \right\}$

9 $(\hat{\sigma}_n^2)^{(k)} = \frac{1}{M} \left\{ \|\mathbf{y} - \mathbf{A}\hat{\mathbf{c}}^{(k)}\|^2 + \text{Tr}(\mathbf{A}^T \mathbf{A} \hat{\mathbf{\Gamma}}^{(k)}) \right\}$

10 if $\hat{\mathbf{c}}^{(k)}$ converges, $k = K$ and $\hat{\mathbf{c}}^{(K)} = \hat{\mathbf{c}}^{(k)}$

11 **end**

12 **return** $\hat{\mathbf{c}} = \hat{\mathbf{c}}^{(K)}$

D. Computational Efficiency and Pruning

Computational complexities of the three Bayesian regression approaches (BLRC, CG and SBL) are more or less of the same order. This is because matrix inversion is the most computationally intense step (see step 4 of

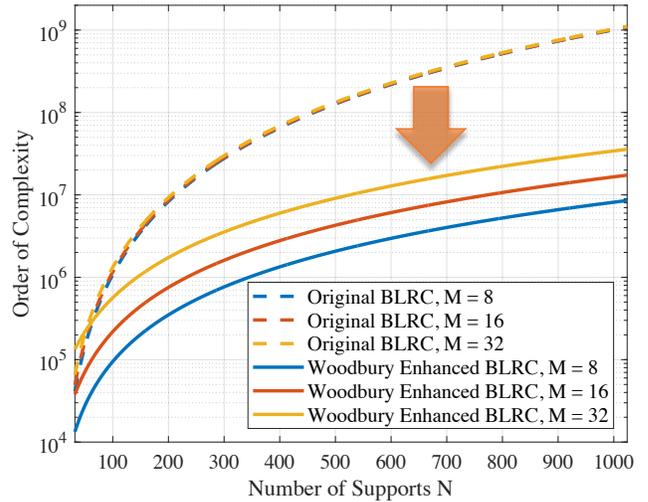


Fig. 2. Computational Efficiency Enhancement via using Woodbury Matrix Identity

Algorithms 1, 2 and 3) and the matrices to be inverted are of the same size, $N \times N$, where N is usually large.

In order to be used in real-time processing applications such as automotive radar, speed-up methods are required for CG, SBL, and BLRC to improve their computational efficiency. Outlined below are three standard strategies. Firstly, the Woodbury matrix identity can be employed to reduce the computational complexity of inverting the $N \times N$ matrices because $\hat{\mathbf{Q}}$ and $\hat{\Sigma}$ are diagonal matrices and the system matrix \mathbf{A} is $M \times N$, $M \ll N$. As shown in Fig. 2, computation speed has been improved by one to two orders of magnitude. Secondly, further increase of the computational efficiency is to use pruning to reduce the effective size of \mathbf{A} in each iteration. If a pruning threshold $t_p^{(k)}$ for the k^{th} iteration is determined, the elements of $\mathbf{c}^{(k-1)}$ with magnitudes smaller than $t_p^{(k)}$ are marked. These elements and their corresponding columns in \mathbf{A} are excluded. Consequently, the size of the matrix to be inverted is reduced in the k^{th} iteration. Thirdly, further improvement on computational efficiency can be made by using efficient matrix inversion algorithms such as Cholesky decomposition for the matrix inversion.

In such a manner, the computation is not only more efficient but also more robust against numerical issues. However, pruning leads to sub-optimal results [38]. Moreover, selecting the pruning threshold $t_p^{(k)}$ itself is not a trivial problem. The larger the pruning threshold is, the more numerically efficient the three Bayesian regression approaches will be, but with an increased risk of discarding weak targets. If the pruning threshold is too small, it has little effect on the computation efficiency. Thus, for fully understanding the convergent property of BLRC, we do not use pruning in numerical analyses in this paper.

IV. Comparison under Variational Interpretation

In this section, we firstly compare the true prior used in SBL and the Cauchy prior used in both CG and

BLRC. Secondly, in order to directly compare different algorithms in the same c -space, the variational bounds of the priors in SBL and BLRC are shown. Note that the core of deriving variational bound is to represent a convex function using its dual form [55], and the variational bound of SBL has been derived in [50] which will be omitted here. Finally, cost functions of l_p , CG, SBL and BLRC in c -space are compared. It is shown that, like SBL, BLRC has the capability to reduce the number of local minimums which explains the superior performance of BLRC over CG.

A. True Prior Comparison

The intrinsic true prior used in SBL can be obtained by integrating $p(c_i|\tau_i)p(\tau_i)$ with respect to the hyper-parameter τ_i . Using (11), the integration result can be represented as a special case of the Student-t distribution $St(c_i; 0, \sqrt{\frac{b}{a}}, 2a)$ [38]:

$$\begin{aligned} p(c_i; a, b) &= \int p(c_i|\tau_i)p(\tau_i)d\tau_i \\ &= \int \mathcal{N}(c_i|0, \tau_i^{-1})\text{Gamma}(\tau_i|a, b)d\tau_i \quad (29) \\ &= \frac{b^a \Gamma(a + \frac{1}{2})}{(2\pi)^{\frac{1}{2}} \Gamma(a)} \left(b + \frac{c_i^2}{2}\right)^{-(a + \frac{1}{2})} \end{aligned}$$

where a, b are the parameters of Gamma distribution. Specifically, the parameters are set to $a = 1$ and $b = 0$ to get an improper uniform hyper-prior [56] or set to $a = b = 0$ to get the Jeffreys prior [38]. Then the true prior degenerates from the Student-t distribution to an improper prior $p(c_i) \propto \frac{1}{|c_i|^3}$ or $p(c_i) \propto \frac{1}{|c_i|}$, respectively. Note that they are improper priors because the areas under these probability density curves cannot be equal to one in the absence of properly defined scaling factors. Although it is hard to directly use these improper priors for Bayesian inference, the MAP estimation adopting these improper priors leads to the IR- l_1 algorithm.

The non-standard Cauchy prior $Cauchy(c_i; 0, \gamma)$ in (2) for both CG and BLRC is also a special case of non-standard Student-t distribution $St(c_i; 0, \gamma, \nu)$ with $\nu = 1$:

$$St(c_i; 0, \gamma, \nu) = \frac{\Gamma(\frac{\nu+1}{2})}{\Gamma(\frac{\nu}{2})} \left(\frac{1}{\pi\nu\gamma^2}\right)^{\frac{1}{2}} \left[1 + \frac{c_i^2}{\nu\gamma^2}\right]^{-\frac{\nu+1}{2}} \quad (30)$$

where γ is the scale parameter and ν is the degree of freedom. Note that (30) becomes (29) if $\nu = 2a$ and $\gamma^2 = \frac{b}{a}$. And the Cauchy prior used in CG and BLRC can be obtained by setting $a = \frac{1}{2}$ which is between 0 and 1.

Unlike the improper prior for SBL, the obtained Cauchy prior for BLRC is a proper distribution. Compared with the Student-t distribution which has a parameter a embedded in the gamma function, Cauchy distribution does not have the parameter a and is much easier to be used for Bayesian inference. Moreover, since the τ has been integrated out in (29), there is no need to estimate the large $N \times 1$ vector τ for Bayesian inference as in the case of SBL. Instead, in the BLRC approach,

one needs to estimate only the scalar parameter γ which is related to the parameter b in (29).

B. Conjugate Dual and Variational Bound

The cost functions of algorithms with MAP framework (e.g., CG) lie in the c -space, whereas the cost functions of algorithms with hierarchical framework (e.g., SBL and BLRC) are not directly represented in c -space. Fortunately, as shown in [50], it provides a way to derive the cost function of algorithms with hierarchical framework in c -space using conjugate dual and variational bound.

From the Student-t prior of a scalar c_i in (29) and omitting the subscript i for convenience, we have

$$\ln p(c) = -(a + \frac{1}{2}) \ln(b + \frac{c^2}{2}) + \ln \frac{b^a \Gamma(a + \frac{1}{2})}{(2\pi)^{\frac{1}{2}} \Gamma(a)} \quad (31)$$

Define the function $f(x)$:

$$f(x) = -(a + \frac{1}{2}) \ln(b + \frac{x}{2}) + \ln \frac{b^a \Gamma(a + \frac{1}{2})}{(2\pi)^{\frac{1}{2}} \Gamma(a)} \quad (32)$$

where $\ln p(c) = f(c^2)$ and $f(x)$ is convex. Also define the conjugate function of $f(x)$ as:

$$f^*(\xi) = \sup_x \left\{ \frac{-\xi}{2} x - f(x) \right\}, \quad \xi > 0 \quad (33)$$

where we use $\frac{-\xi}{2}x$ instead of ξx for notational convenience. Substituting (32) into (33), it can be shown that

$$x^* = \frac{2a + 1}{\xi} - 2b \quad (34)$$

is the solution to $f^*(\xi)$ in (33). Let $x = x^*$ in (34), the conjugate function in (33) becomes

$$\begin{aligned} f^*(\xi) &= -(a + \frac{1}{2}) + \xi b + (a + \frac{1}{2}) \ln \frac{2a + 1}{2\xi} \\ &\quad - \ln \frac{b^a \Gamma(a + \frac{1}{2})}{(2\pi)^{\frac{1}{2}} \Gamma(a)}, \quad \xi > 0, a > 0, b > 0 \end{aligned} \quad (35)$$

Using (33) and (35), we then derive the lower bound of $f(x)$:

$$\begin{aligned} f(x) &\geq \frac{-\xi}{2} x - f^*(\xi) \\ &= -\frac{\xi x}{2} + a + \frac{1}{2} - \xi b - (a + \frac{1}{2}) \ln \frac{2a + 1}{2\xi} \\ &\quad + \ln \frac{b^a \Gamma(a + \frac{1}{2})}{(2\pi)^{\frac{1}{2}} \Gamma(a)} \end{aligned} \quad (36)$$

Since $p(c) = e^{f(c^2)}$, we can use (36) to get the lower bound of the Student-t prior $p(c)$ as

$$p(c) \geq \frac{1}{\sqrt{2\pi\xi^{-1}}} e^{-\frac{c^2}{2\xi^{-1}}} \cdot \phi(\xi, a, b), \quad \forall \xi > 0, a > 0, b > 0 \quad (37)$$

where

$$\phi(\xi, a, b) = \xi^a e^{-b\xi} \cdot \frac{b^a \Gamma(a + \frac{1}{2})}{\Gamma(a)} \cdot \left(\frac{2a + 1}{2e}\right)^{-(a + \frac{1}{2})} \quad (38)$$

If $a \rightarrow 0$ and $b \rightarrow 0$, it is easy to see that a Jeffrey's non-informative prior is obtained. The evidence

maximization can be applied and the cost function of SBL in \mathbf{c} -space can be derived as shown in [50] which is also provided in Appendix B:

$$\min_{\mathbf{c}} J_{sbl}(\mathbf{c}), \quad J_{sbl}(\mathbf{c}) = \|\mathbf{y} - \mathbf{A}\mathbf{c}\|^2 + h_{sbl}(\mathbf{c}) \quad (39)$$

where the minimization with respect to τ and σ_n is done as following (in deriving the regularization $h_{sbl}(\mathbf{c})$ to encourage a sparse solution):

$$h_{sbl}(\mathbf{c}) = \min_{\tau, \sigma_n} \sigma_n^2 \left(\mathbf{c}^T \Sigma \mathbf{c} + \ln |\sigma_n^2 \mathbf{I} + \mathbf{A} \Sigma^{-1} \mathbf{A}^T| \right) \quad (40)$$

C. BLRC Cost Function in \mathbf{c} -space

By setting $a = \frac{1}{2}$ and $b = \frac{\gamma^2}{2}$ in (37), a lower bound of the Cauchy prior $p(c)$ is shown to be:

$$p(c) \geq \frac{1}{\sqrt{2\pi\xi^{-1}}} e^{-\frac{c^2}{2\xi^{-1}}} \cdot \phi(\xi, \gamma) = \mathcal{N}(0, \xi^{-1}) \phi(\xi, \gamma) \quad (41)$$

where $\phi(\xi, \gamma) = \gamma e \sqrt{\frac{\xi}{2\pi}} e^{-\frac{\xi\gamma^2}{2}}$, $\xi > 0$, $\gamma > 0$. Note that the bound in (41) is tight when

$$\xi = \frac{2}{c^2 + \gamma^2}. \quad (42)$$

Replacing c and ξ by c_i and ξ_i , respectively, in both (41) and (42), we can find $p(c_i)$ from (41). Then, using $p(c_i)$, $\forall i$, we can extend the lower bound in (41) of a scalar c to a lower bound of a vector \mathbf{c} :

$$p(\mathbf{c}; \gamma) = \prod_{i=1}^N p(c_i) \geq \mathcal{N}(\mathbf{0}, \Theta^{-1}) \prod_{i=1}^N \phi(\xi_i, \gamma) \quad (43)$$

where

$$\Theta = \text{diag}(\xi_1, \xi_2, \dots, \xi_N) \quad (44)$$

In (43), the dependence on γ is explicitly shown. Given that ξ_i and c_i are related according to (42), the lower bound in (43) is tight.

With the likelihood function in (4) and the lower bound of Cauchy prior in (43) at hand, a lower bound of the evidence distribution $p(\mathbf{y}; \gamma, \sigma_n)$ can be expressed as the following integral:

$$\begin{aligned} p(\mathbf{y}; \gamma, \sigma_n) &= \int p(\mathbf{y}|\mathbf{c}; \sigma_n) p(\mathbf{c}; \gamma) d\mathbf{c} \\ &\geq \int \mathcal{N}(\mathbf{A}\mathbf{c}, \sigma_n^2 \mathbf{I}) \mathcal{N}(\mathbf{0}, \Theta^{-1}) \prod_{i=1}^N \phi(\xi_i, \gamma) d\mathbf{c} \end{aligned} \quad (45)$$

where the lower bound is tight. Unfortunately, we cannot obtain a closed-form expression of the above integral. In Section III, we have already used AEM to derive an alternative lower bound of $p(\mathbf{y}; \gamma, \sigma_n)$ (see (15) and subsequent equations).

Here, we will approximate the integral in (45) based on the iterative procedure of BLRC. Note that the estimate of c_i for the $(k-1)^{th}$ iteration, $c_i^{(k-1)}$, is a known constant, so the following corresponding parameters

$$\begin{aligned} \xi_i^{(k-1)} &= \frac{2}{(c_i^{(k-1)})^2 + \gamma^2} \\ \Theta^{(k-1)} &= \text{diag}(\xi_1^{(k-1)}, \xi_2^{(k-1)}, \dots, \xi_N^{(k-1)}) \end{aligned} \quad (46)$$

are also known constants. Then, $\phi(\xi_i^{(k-1)}, \gamma)$, $\forall i$, derived from the $(k-1)^{th}$ iteration are independent of the integration variable \mathbf{c} and can be brought out of the integral in (45). Thus, (45) becomes

$$p(\mathbf{y}; \gamma, \sigma_n) \geq \mathcal{N}(\mathbf{0}, \Theta_y^{-1}) \cdot \prod_{i=1}^N \phi(\xi_i^{(k-1)}, \gamma) \quad (47)$$

with

$$\mathcal{N}(\mathbf{0}, \Theta_y^{-1}) = \int \mathcal{N}(\mathbf{A}\mathbf{c}, \sigma_n^2 \mathbf{I}) \mathcal{N}(\mathbf{0}, (\Theta^{(k-1)})^{-1}) d\mathbf{c} \quad (48)$$

where

$$\Theta_y^{-1} = \sigma_n^2 \mathbf{I} + \mathbf{A} (\Theta^{(k-1)})^{-1} \mathbf{A}^T \quad (49)$$

In (47), the lower bound may not be tight because of the iterative approximation.

Note that maximizing $\ln p(\mathbf{y}; \gamma, \sigma_n)$ is equivalent to minimizing the negative logarithm of the lower bound of $p(\mathbf{y}; \gamma, \sigma_n)$. The cost function to be minimized for the k^{th} iteration for BLRC is then defined as

$$L_{blrc} = \ln |\Theta_y^{-1}| + \mathbf{y}^T \Theta_y \mathbf{y} - \sum_{i=1}^N \ln \phi(\xi_i^{(k-1)}, \gamma) \quad (50)$$

Recall that $c_i^{(k-1)}$ is defined by the previous iteration, so L_{blrc} is not shown to be an explicit function of \mathbf{c} in (50). However, using the same procedure given in Appendix C, it can be shown that

$$\mathbf{y}^T \Theta_y \mathbf{y} = \min_{\mathbf{c}} \left(\frac{1}{\sigma_n^2} \|\mathbf{y} - \mathbf{A}\mathbf{c}\|^2 + \mathbf{c}^T \Theta^{(k-1)} \mathbf{c} \right) \quad (51)$$

Substituting (51) into (50), L_{blrc} can then be viewed as a function of \mathbf{c} as shown below:

$$\begin{aligned} L_{blrc}(\mathbf{c}) &= \min_{\mathbf{c}} \left(\frac{1}{\sigma_n^2} \|\mathbf{y} - \mathbf{A}\mathbf{c}\|^2 + \mathbf{c}^T \Theta^{(k-1)} \mathbf{c} \right) \\ &\quad + \ln |\Theta_y^{-1}| - \sum_{i=1}^N \ln \phi(\xi_i^{(k-1)}, \gamma) \end{aligned} \quad (52)$$

Note that the last two terms in (52) do not depend on the unknown \mathbf{c} for the current iteration. Substituting (49) into (52), minimizing L_{blrc} with respect to γ and σ_n can be rewritten in \mathbf{c} -space as:

$$\min_{\mathbf{c}} J_{blrc}(\mathbf{c}), \quad J_{blrc}(\mathbf{c}) = \|\mathbf{y} - \mathbf{A}\mathbf{c}\|^2 + h_{blrc}(\mathbf{c}) \quad (53)$$

where the minimization with respect to γ and σ_n is carried out in executing the regularization $h_{blrc}(\mathbf{c})$:

$$\begin{aligned} h_{blrc}(\mathbf{c}) &= \min_{\gamma, \sigma_n} \sigma_n^2 \left(- \sum_{i=1}^N \ln \phi(\xi_i^{(k-1)}, \gamma) \right. \\ &\quad \left. + \mathbf{c}^T \Theta^{(k-1)} \mathbf{c} + \ln |\sigma_n^2 \mathbf{I} + \mathbf{A} (\Theta^{(k-1)})^{-1} \mathbf{A}^T| \right) \end{aligned} \quad (54)$$

Here, the regularization $h_{blrc}(\mathbf{c})$ is to encourage a sparse solution.

D. Comparison in c-space

Consider the following maximally sparse signal recovery problem:

$$\min_{\mathbf{c}} \|\mathbf{c}\|_0, \quad s.t. \mathbf{y} = \mathbf{A}\mathbf{c} \quad (55)$$

Here, the l_0 norm, $\|\mathbf{c}\|_0$, represents the number of non-zero elements in vector \mathbf{c} . Although the cost function in (55) is ideal for many applications, particularly when exact reconstruction [57] is desired, finding its global minimum is an NP-hard problem.

Given that $\|\mathbf{c}\|_0 \equiv \lim_{p \rightarrow 0} \sum_i |c_i|^p$, we employ the l_p norm (i.e., $\|\mathbf{c}\|_p^p$ with $0 < p < 1$) as a benchmark in this section:

$$\min_{\mathbf{c}} \|\mathbf{c}\|_p^p, \quad s.t. \mathbf{y} = \mathbf{A}\mathbf{c} \quad (56)$$

We then compare optimization landscapes of the cost functions in (56), (12), (39), and (53) corresponding to l_p , CG, SBL, and BLRC, respectively. An effective optimization landscape is expected to possess the same global minimum as (55), while exhibiting fewer local minimums.

In order to visualize the differences of optimization landscapes defined in (56), (12), (39), and (53), consider the example shown in [50] where $N = M + 1$. Thus, the null-space of \mathbf{A} , denoted by \mathbf{a}_{null} , has only one dimension. If \mathbf{c}_{op} is the optimum solution achieving the global minimum of (55), \mathbf{c}_{op} shows maximum sparsity and

$$\mathbf{y} = \mathbf{A}\mathbf{c} = \mathbf{A}\mathbf{c}_{op}, \quad \mathbf{c} = \mathbf{c}_{op} + v \cdot \mathbf{a}_{null} \quad (57)$$

where v is an arbitrary constant. Choosing an \mathbf{A} where \mathbf{c}_{op} is known, we can then plot the regularization terms in (56), (12), (39), and (53), with respect to \mathbf{c} by changing the scalar v , to view the optimization landscape (see Fig. 3). Since the constraint $\mathbf{y} = \mathbf{A}\mathbf{c}$ is always satisfied with different v , the optimization landscape of regularization term is also the optimization landscape of the corresponding optimization problem. Note that we scale the regularization terms such that all the regularization terms equal to 1 at $v = 0$.

Since l_p is close to the l_0 if p is small. In Fig. 3, the $p = 0.01$ curve is similar to that for the l_0 norm, where the large pit at $v = 0$ denotes the global minimum. Unfortunately, there exist two other pits at $v = -2$ and $v = 0.71$, which represent two local minimums. As expected, the large pit at $v = 0$ is widened while maintaining its position as global minimum when p increases to 0.1. The two other pits at $v = -2$ and $v = 0.71$ are also widened with the increase in p . Nonetheless, the existence of these local minimums persists with various p value in $(0, 1)$, which complicates the search for the global minimum.

As shown in Fig. 3, it is remarkable that h_{cg} in (13) with a small $\hat{\gamma}$ value for the CG approach behaves like l_p norm with a small p value. As $\hat{\gamma}$ increases, all pits get smoother. Thus, in order to get a more sparse solution, we need to choose a smaller $\hat{\gamma}$ for the CG approach. However, the CG iteration is more likely to be trapped in one of

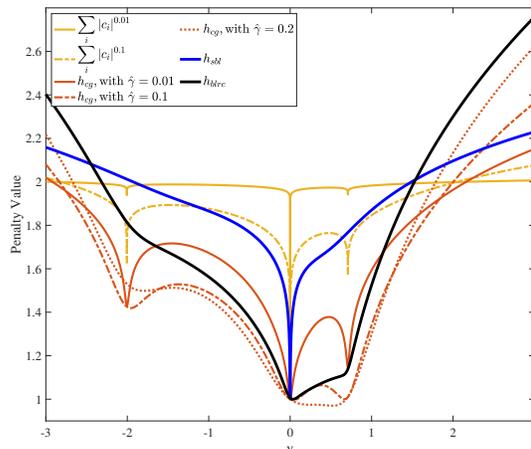


Fig. 3. Normalized penalty values of (56), (12), (39), and (53) with respect to the parameter v in (57).

the local minimums when $\hat{\gamma}$ is small. On the other hand, if we choose a large $\hat{\gamma}$ for iteration, we may avoid being trapped in a local minimum. However, the solution is no longer accurate because the global minimum may shift to a different location. For example, when $\hat{\gamma} = 0.2$, the global minimum shifts from $v = 0$ to $v = 0.52$ as shown in Fig. 3. Therefore, it is unclear how to choose $\hat{\gamma}$ in general for the CG approach.

To plot the penalty $h_{sbl}(\mathbf{c})$ in (40) for SBL, the \mathbf{c} is still controlled by the scalar v using (57). However, the hyper-parameters τ and σ_n within $h_{sbl}(\mathbf{c})$ cannot be predetermined because they depend on the value of \mathbf{c} . Thus for a given v , we firstly compute \mathbf{c} using (57). Then, use line 6 and 8 in Algorithm 2 to compute σ_n and τ iteratively until minimum $h_{sbl}(\mathbf{c})$ is obtained. In Fig. 3, the SBL curve shows the correct global minimum but no local minimums. Note that the h_{sbl} curve has been shown in [50] to produce fewer local minimums than l_p approach as well.

Similar to SBL, the hyper-parameters γ and σ_n of BLRC depend on the value of \mathbf{c} . Thus, to plot the penalty $h_{blrc}(\mathbf{c})$ in (54), we firstly compute \mathbf{c} using (57) for a given v . Then, noting that $\gamma = \frac{1}{\sqrt{\tau}}$, use line 8 and 9 in Algorithm 3 to compute σ_n and τ iteratively until minimum $h_{blrc}(\mathbf{c})$ is reached. In Fig. 3, BLRC presents the correct global minimum without any local minimums, which indicates that it has similar capability as SBL to reduce the number of local minimums. This explains the superior performance of BLRC over CG seen in numerical examples in Section VI.

V. Comparisons under IR- l_2 Interpretations

In this section, we will compare BLRC, SBL, and CG based on the IR- l_2 interpretation which will be used to explain the superior performances of BLRC over SBL in Section VI.

A. IR- l_2 Formulation of CG and BLRC

Since both CG and BLRC use the Cauchy prior in the MAP framework, they have exactly the same steps for updating \mathbf{c} (see step 4 in Algorithm 1 and steps 4 and 5 of Algorithm 3). These updating formulas are derived from solving the log-sum regularized optimization problem

$$\mathbf{c} = \arg \min_{\mathbf{c}} \|\mathbf{y} - \mathbf{A}\mathbf{c}\|^2 + \sigma_n^2 \sum_{i=1}^N 2 \ln(c_i^2 + \hat{\gamma}^2). \quad (58)$$

It is well-known that the log-sum penalty encourages a sparse solution. We typically use the following IR- l_2 approach to minimize this objective function in (58):

$$\mathbf{c}^{(k+1)} = \arg \min_{\mathbf{c}} \{\|\mathbf{y} - \mathbf{A}\mathbf{c}\|_2^2 + \sigma_n^2 \sum_{i=1}^N w_i^{(k)} c_i^2\} \quad (59)$$

where the weight for the k^{th} iteration is

$$w_i^{(k)} = \frac{2}{(\hat{\gamma}^{(k)})^2 + (\hat{c}_i^{(k)})^2} \quad (60)$$

Note that $\{\hat{c}_i^{(k)}\}$ in (60) are obtained from the k^{th} iteration and are considered as constants in (59). The noise variance σ_n^2 in (59) and the scale parameter $\hat{\gamma}^{(k)}$ in (60) are constants with respect to k in CG, but are updated at each k in BLRC.

B. IR- l_2 Formulation of SBL

For the \mathbf{c} updating step in SBL, it is remarkable that line 5 of Algorithm 2 coincides with the solution of the following log-sum regularized optimization problem:

$$\min_{\mathbf{c}} \|\mathbf{y} - \mathbf{A}\mathbf{c}\|_2^2 + \sigma_n^2 \sum_{i=1}^N \ln(c_i^2 + \tilde{\Gamma}_{ii}) \quad (61)$$

The IR- l_2 updating scheme for solving the above optimization problem is given in (59) where the weight

$$w_i^{(k)} = \frac{1}{\tilde{\Gamma}_{ii}^{(k)} + (\hat{c}_i^{(k)})^2} \quad (62)$$

Let $\tau_i^{(k)} = w_i^{(k)}$. It can be shown that (62) is equivalent to the updated hyper-parameter $\tilde{\tau}$ at the k^{th} iteration in step 8 of Algorithm 2. Thus, in SBL, we actually “learn” the hyper-parameter $\tau_i^{(k)}$ using the IR- l_2 updating scheme.

C. The IR- l_2 Point of View

From the IR- l_2 point of view, SBL, BLRC and CG solve the same equation (59) except the following difference. CG and BLRC use a common factor $(\hat{\gamma}^{(k)})^2$ for all weights in (60). However, SBL needs different factors $\tilde{\Gamma}_{ii}$ for different weights in (62).

As shown in the left subplot of Fig. 4, Cauchy in (2) is a long-tailed distribution. When the parameter γ approaches zero, most of the probability mass will concentrate at zero, which implies most c_i 's will have high probabilities to be zero. However, no matter how small γ is, the mean and variance of Cauchy distribution are still undefined. This implies that a small number of

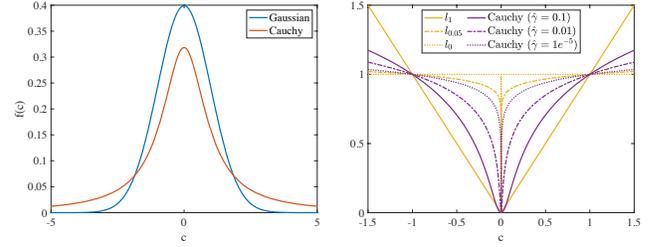


Fig. 4. (Left) Comparison of Gaussian ($\sigma = 1$) and Cauchy ($\gamma = 1$) Distributions; (Right) Illustration of Regularization Functions for CG and BLRC

c_i 's can be non-zeros. Thus, a single universal parameter γ of Cauchy prior can be used to model all c_i 's as seen in (60).

This is not the case for the conditional Gaussian prior. As shown in the left subplot of Fig. 4, Gaussian in (3) is not a long-tailed distribution. When a particular c_i is near zero, the variance σ_i^2 (i.e., $1/\tau_i$) of the corresponding zero-mean Gaussian distribution must approach zero, and vice versa. Similarly, when a particular c_i is non-zero, the variance σ_i^2 of the corresponding zero-mean Gaussian distribution must not be near zero, and vice versa. Therefore, with Gaussian prior, $N \tilde{\Gamma}_{ii}$'s are needed for the N elements of \mathbf{c} as seen in (62).

Recall that the log-sum penalty is equivalent to a Cauchy prior. So, from the IR- l_2 point of view, SBL's formulation in (61) uses “Cauchy priors” with N parameters $\tilde{\Gamma}_{ii}$'s, while CG and BLRC's formulation in (58) uses Cauchy priors with only one parameter $\hat{\gamma}$. In other words, we need to determine N hyper-parameters (i.e., $\tilde{\tau}_i$'s) for characterizing the N parameters $\tilde{\Gamma}_{ii}$'s in SBL, but only one hyper-parameter (i.e., $\hat{\gamma}$) for characterizing the Cauchy prior. Thus, BLRC leads to a more compact latent space. This is a major advantage of BLRC over SBL, which can be explained as follows.

Suppose both c_j and c_l in (1) are 0 and to be recovered using SBL and BLRC. During the SBL iteration process, one can see from (62) that both $\tilde{\Gamma}_{jj}^{(k)}$ and $\tilde{\Gamma}_{ll}^{(k)}$ need to approach 0 in order to increase the penalties for non-zero $c_j^{(k)}$ and $c_l^{(k)}$ as the iteration index k increases. However, in practice, $\tilde{\Gamma}_{jj}^{(k)}$ and $\tilde{\Gamma}_{ll}^{(k)}$ are independently controlled by hyper-parameters $\tau_j^{(k)}$ and $\tau_l^{(k)}$. It is very likely that $\tilde{\Gamma}_{jj}^{(k)}$ and $\tilde{\Gamma}_{ll}^{(k)}$ approaches zero with different speeds and therefore the resulting $c_j^{(k)}$ and $c_l^{(k)}$ are not both near 0. This is not the case with BLRC. In BLRC, as k increases, the only hyper-parameter $\hat{\gamma}^{(k)}$ decreases. It will cause the penalties for non-zero $c_j^{(k)}$ and $c_l^{(k)}$ to increase with more or less the same rate (see (60)). Thus, the resulting $c_j^{(k)}$ and $c_l^{(k)}$ will both be near 0 when the BLRC iteration process converges.

The implication of reducing $\hat{\gamma}^{(k)}$ is shown in the right subplot of Fig. 4 where $\ln(1 + \frac{|c|^2}{\hat{\gamma}^2})$ is plotted for various values of $\hat{\gamma}$. In addition, the l_0 norm, $l_{0.05}$ norm and l_1 norm of c are also plotted. It can be seen that $\ln(1 + \frac{|c|^2}{\hat{\gamma}^2})$ approaches $\|c\|_0$ as $\hat{\gamma}^{(k)}$ approaches zero [57]. This shows

TABLE I
SIMULATION EXAMPLE: SIX RAYS

ray indexes	[1, 2, 3, 4, 5, 6]
Frequencies	[0.1212, 0.1413, 0.3132, 0.331, 0.41, 0.465]
Amplitudes	[1, 0.9254, 0.7331, 0.5678, 0.6, 0.8]
Phases	[5.1191, 5.6913, 0.7979, 5.7389, 3.9732, 0.6129]

that BLRC approximately uses the l_0 norm as $\hat{\gamma}^{(k)}$ in the regularization term approaches zero.

VI. Numerical Analyses

In this section, numerical analyses on numerical stability, convergent properties, spurious targets suppression, noise variance estimation, sensitivities and resolutions of CG, SBL and BLRC are provided. The OMP [24], which is a popular greedy-based approach for the sparse signal recovery problem, is used as a performance benchmark here.

Consider a sum of K rays for numerical simulation:

$$y(i) = \sum_{l=1}^K a_l \exp\{j(2\pi f_l \frac{i}{N} + \phi_l)\} + \epsilon(i) \quad (63)$$

where $i \in S$ and $S \subset \{1, 2, \dots, N\}$. The number of elements in S is M which is smaller than N . Here, $\{f_l\}$, $\{a_l\}$ and $\{\phi_l\}$ are spatial frequencies, amplitudes, and phases, respectively. The additive noise $\{\epsilon(i)\}$ are independent identical distributed zero-mean Gaussian random variables with variance σ_n^2 .

A. A Six-Ray Example

Without loss of generality, we choose $N = 256$. The arbitrarily selected spatial frequencies f_l , amplitudes a_l , and phases ϕ_l for $K = 6$ are listed in Table I. The standard deviation of the additive noise $\epsilon(i)$ is $\sigma_n = 0.1$. Consider a large sparse antenna array (SPA) with $M = 80$ and a small coprime array (CPA) with $M = 16$. The 80 random samples observed by the large SPA, the 16 samples observed by the small CPA, and the 256 full samples of $\{y(i)\}$ are shown in the top subplot of Fig. 5. Fourier spectra of SPA and CPA are shown in the bottom left and right subplots, respectively, of Fig. 5 where the six rays are marked as ground truth. It is seen that the six rays are fully resolved by the large SPA. However, many small noise-like side lobes are generated. For the small CPA, two rays are missing (unresolved). In addition, several large spurious rays appeared.

Using the large SPA and small CPA shown in Fig. 5, OMP, CG, SBL, and BLRC are employed to recover the sparse signal defined in (63) and Table I. Note that the Bayesian approaches (CG, SBL, and BLRC) summarized in Section II have been formulated in terms of real variables and parameters. The extension of these approaches to deal with complex variables and parameters is straightforward as shown in [44]. At first, the $(\cdot)^2$

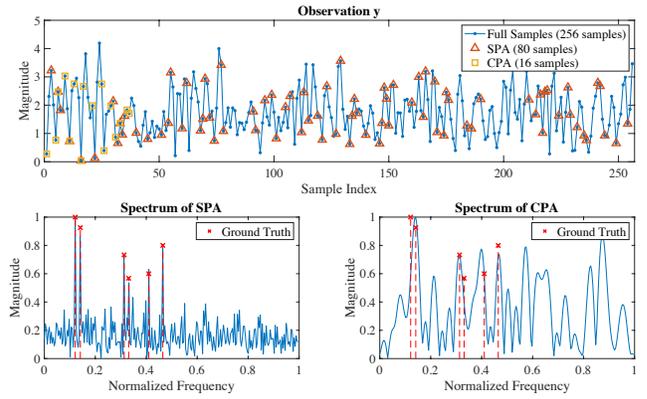


Fig. 5. Time samples and Spectra of sparse array (SPA) and coprime array (CPA)

in Cauchy distribution (2) is replaced by $|(\cdot)|^2$, and the likelihood distribution in (4) is changed to complex Gaussian. Then, the derivations are the same as before except that we need to change transpose $(\cdot)^T$ to Hermitian transpose $(\cdot)^H$ and replace $(\cdot)^2$ by $|(\cdot)|^2$.

B. Initial Values of Hyper-parameters

SBL is robust with respect to the selection of initial values. Here, the initial values are $\tilde{\sigma}_n = 1$ and $\tilde{\sigma}_i = 0.1$, $\forall i$ for both SPA and CPA.

For CG, the selection of the optimal values of $\hat{\sigma}_n$ and $\hat{\gamma}$ is crucial to the performance of CG. Based on the knowledge of the targets in Table I, the optimal values chosen by trial and error are $\hat{\sigma}_n = 0.1$ and $\hat{\gamma} = 0.01$ for both SPA and CPA in our simulation. However, we are not able to know the true targets in practice. Therefore, it is impossible to find the optimal values for $\hat{\sigma}_n$ and $\hat{\gamma}$ for CG in reality. The initial estimate of each element of \mathbf{c} for CG is a random number uniformly distributed in $[0, 1]$.

Like SBL, the selection of initial $\hat{\sigma}_n$ and $\hat{\gamma}$ for BLRC is not critical since they will be updated robustly as the iteration proceeds. The rule of thumb is that the initial $\hat{\gamma}$ needs to be large enough to emphasize the data fitting term $\|\mathbf{y} - \mathbf{A}\mathbf{c}\|^2$ in (58) at the beginning of the iteration process. As the iteration proceeds, $\hat{\gamma}$ will continue to decrease so as to emphasize the regularization term $\hat{\sigma}_n^2 \sum_{i=1}^N 2 \ln(c_i^2 + \hat{\gamma}^2)$ in (58) in order to promote sparsity. However, $\hat{\gamma}$ should not be too large, otherwise it will require more iterations to converge. Since there are fewer measurements with CPA ($M = 16$) compared with SPA ($M = 80$), there exist more local minimums in CPA [45]. Thus, we need to have a larger initial value of $\hat{\gamma}$ with CPA to avoid being trapped in a local minimum in the early stage of iterations. In our simulations, the initial values for BLRC are $\hat{\sigma}_n = 1$ and $\hat{\gamma} = 0.1$ for SPA and $\hat{\sigma}_n = 1$ and $\hat{\gamma} = 1$ for CPA. Like CG, the initial estimate of each element of \mathbf{c} for BLRC is also a random number uniformly distributed in $[0, 1]$.

Using SPA and CPA, respectively, Fig. 6 and Fig. 7 show the values of key parameters versus the iteration

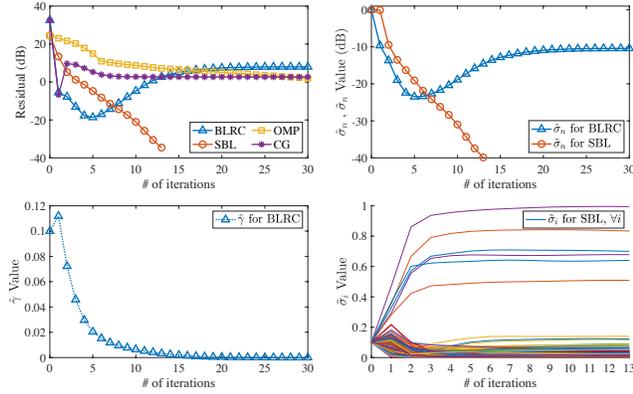


Fig. 6. Convergent Properties of OMP, SBL, CG, and BLRC using SPA

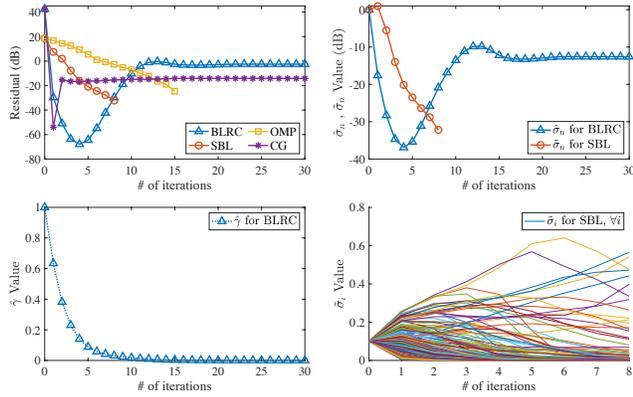


Fig. 7. Convergent Properties of OMP, SBL, CG, and BLRC using CPA

number for OMP, CG, SBL, and BLRC. These results will be used to discuss various numerical aspects of the four considered approaches in the following subsections. Here, in the top left subplots of both figures, the residue in dB is defined as $R_{dB}^{(k)} \equiv 20 \log_{10} \|\mathbf{y} - \mathbf{A}\mathbf{c}^{(k)}\|$ where $\mathbf{c}^{(k)}$ is the estimate of \mathbf{c} using the corresponding approach in the k^{th} iteration.

To further demonstrate that BLRC is robust to the initial values of hyper-parameters, numerical simulations with different initial values of $\hat{\sigma}_n$ and $\hat{\gamma}$ are performed for CPA. In the left subplot of Fig. 8, the residues for BLRC with different initial $\hat{\sigma}_n$ values are plotted against the iteration number while the initial value of $\hat{\gamma}$ is 1. For these initial values, the updated $\hat{\sigma}_n$ values are plotted against the iteration number in the right subplot of Fig. 8. Although different initial values are set for $\hat{\sigma}_n$, identical final residue, final estimate of $\hat{\sigma}_n$, and the recovery result for \mathbf{c} are obtained.

In the left subplot of Fig. 9, the residues for BLRC with different initial $\hat{\gamma}$ values are plotted against the iteration number while the initial value of $\hat{\sigma}_n$ is 1. For these initial values, the updated $\hat{\gamma}$ values are plotted against the iteration number in the right subplot of Fig. 9. Similarly, although we have different initial values for $\hat{\gamma}$, identical final residue, final estimate of $\hat{\gamma}$, and recovery result for \mathbf{c} are obtained.

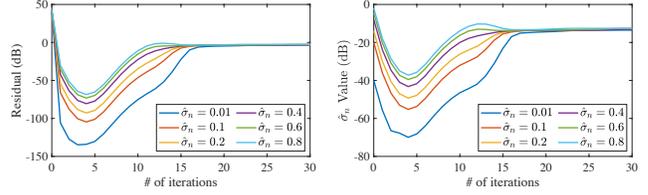


Fig. 8. Convergent Properties of BLRC for CPA with different initial values of $\hat{\sigma}_n$

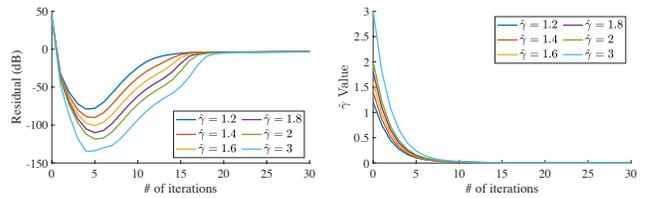


Fig. 9. Convergent Properties of BLRC for CPA with different initial values of $\hat{\gamma}$

TABLE II
CONDITION NUMBERS WITH SPA - SBL AND BLRC

k	5	13	14	25
$\mathbf{H}_{sbl}^{(k)}$	3.4×10^6	8.1×10^{14}	N/A	N/A
$\mathbf{H}_{blrc}^{(k)}$	133	130	236	8.7×10^5

TABLE III
CONDITION NUMBERS WITH CPA - SBL AND BLRC

k	5	8	9	25
$\mathbf{H}_{sbl}^{(k)}$	5.4×10^9	2.9×10^{13}	N/A	N/A
$\mathbf{H}_{blrc}^{(k)}$	1.6×10^4	1.1×10^3	734	8.5×10^6

C. Numerical Stability

In Fig. 6 and Fig. 7, the iterative procedure of SBL is terminated at $k = 13$ and $k = 8$ for SPA and CPA, respectively. This is due to the fact that the matrix $\mathbf{H}_{sbl}^{(k)} = \frac{1}{(\hat{\sigma}_n^{(k-1)})^2} \mathbf{A}^T \mathbf{A} + \hat{\Sigma}^{(k)}$ becomes ill-conditioned for large k 's. Note that $\mathbf{H}_{sbl}^{(k)}$ needs to be inverted so as to obtain the posterior covariance matrix $\hat{\Gamma}^{(k)}$ in step 4 of Algorithm 2. Unlike SBL, the matrix $\mathbf{H}_{blrc}^{(k)}$, defined as $\frac{1}{(\hat{\sigma}_n^{(k-1)})^2} \mathbf{A}^T \mathbf{A} + \frac{2}{(\hat{\gamma}^{(k-1)})^2} \hat{\mathbf{Q}}^{(k)}$, does not become ill-conditioned for the same set of k 's. Note that $\mathbf{H}_{blrc}^{(k)}$ is to be inverted for obtaining the posterior covariance matrix $\hat{\Gamma}^{(k)}$ in step 4 of Algorithm 3. The condition numbers of $\mathbf{H}_{sbl}^{(k)}$ and $\mathbf{H}_{blrc}^{(k)}$ for some k 's are also shown in Table II and Table III for SPA and CPA, respectively.

To understand the reasons why BLRC is numerically more stable than SBL, magnitudes of the IR- l_2 weights $\{w_i^{(k)}\}$ in (60) and (62) for BLRC and SBL, respectively, are sorted in ascending order (see Fig. 10). Note that in Fig. 10 the y -axis is represented in dB so as to emphasize the small $|w_i^{(k)}|$ values. The x -axis is also presented in log scale to emphasize the small i 's.

Recall that the IR- l_2 weights $\{w_i^{(k)}\}$ in (60) are the diagonal elements of $2\hat{\mathbf{Q}}^{(k)}/(\hat{\gamma}^{(k-1)})^2$ in step 4 of Algorithm 3. And the weights $\{w_i^{(k)}\}$ in (62) are equal

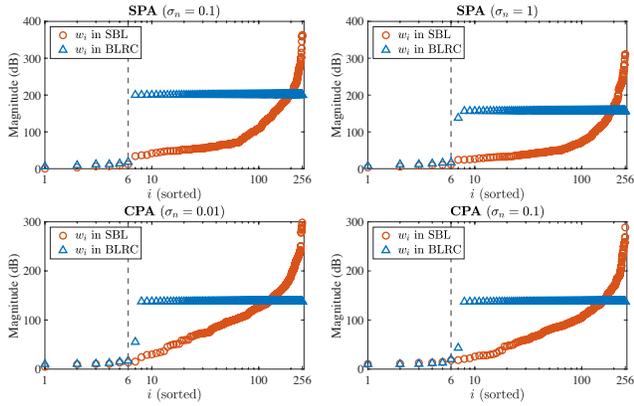


Fig. 10. Sorted magnitudes in dB of the $\text{IR-}l_2$ weights, $20\log_{10}|w_i^{(k)}|$, derived from (60) for BLRC and (62) for SBL, $\forall i = 1, \dots, N$. The dashed vertical line at $i = 6$ marks the number of true targets

to the diagonal elements of $\tilde{\Sigma}^{(k)}$ in step 4 of Algorithm 2. Since $\mathbf{A}^T \mathbf{A}$ is singular, a well-conditioned diagonal loading of $\tilde{\Sigma}^{(k)}$ is desirable so as to make $\mathbf{H}_{sbl}^{(k)}$ non-singular. Similarly, a well-conditioned diagonal loading of $2\hat{\mathbf{Q}}^{(k)}/(\hat{\gamma}^{(k-1)})^2$ is desirable in order to make $\mathbf{H}_{blrc}^{(k)}$ invertible.

From Fig. 10, one can see that the ratio between maximum $|w_i^{(k)}|$ and minimum $|w_i^{(k)}|$ is near or over 10^{15} for SBL. Moreover, $|w_i^{(k)}|$ is more or less uniformly distributed between maximum $|w_i^{(k)}|$ and minimum $|w_i^{(k)}|$ for SBL. This is due to the fact that each element of $\{c_i\}$ is characterized by one hyper-parameter $\tilde{\sigma}_i$ in SBL. When a particular c_i is near zero, the variance σ_i^2 (i.e., $\frac{1}{\tau_i}$ or $\frac{1}{w_i}$) of the corresponding zero-mean Gaussian distribution must approach zero. Thus, τ_i can be seen as the confidence of “the corresponding c_i is zero”. For different elements of $\{c_i\}$, we typically have different confidence levels on whether they are zero or not. In addition, the confidence levels change as iteration increases. Thus, the elements of τ (i.e., the w_i) approach infinity at various speeds during iterations and the diagonal loading for $\mathbf{H}_{sbl}^{(k)}$ (i.e., $\tilde{\Sigma}^{(k)}$) is near singular. As a consequence, $\mathbf{H}_{sbl}^{(k)}$ is also near singular as shown in Table II and Table III, which is also noticed by the SBL original paper [38]. Although we could use σ_i^2 instead of the precision parameters (i.e., τ_i) as in [49], this would not resolve the singularity issue of the matrix $\mathbf{H}_{sbl}^{(k)}$ here. This is because σ_i^2 would still be inverted in the subsequent step of SBL algorithm, leaving the core problem unaddressed.

Unlike SBL, the ratio between maximum $|w_i^{(k)}|$ and minimum $|w_i^{(k)}|$ is near or less than 10^{10} for BLRC. Most importantly, $|w_i^{(k)}|$ is more or less constant for BLRC. This is due to the fact that only one scale parameter $\hat{\gamma}$ is used for characterizing $\{c_i\}$. Since most $\{c_i\}$ are near zeros when BLRC converges, most $\{w_i^{(k)}\}$ are determined only by $\hat{\gamma}$ as shown in (60) and are therefore almost constant. Now that, as the diagonal loading for $\mathbf{H}_{blrc}^{(k)}$ (i.e., $2\hat{\mathbf{Q}}^{(k)}/(\hat{\gamma}^{(k-1)})^2$) is almost constant, $\mathbf{H}_{blrc}^{(k)}$

is not near singular as shown in Table II and Table III. This demonstrates that BLRC is more numerically robust and stable than SBL with sparse array input, making it more suitable for automotive radar applications, which commonly utilize sparse arrays and operate under single-precision computing environments. Moreover, the almost constant diagonal property of $2\hat{\mathbf{Q}}^{(k)}/(\hat{\gamma}^{(k-1)})^2$ helps in noise variance estimation (which will be discussed in Section VI.F).

D. Residues and Convergent Criteria

In the top left subplots in Fig. 6 for SPA and Fig. 7 for CPA, we show the residues in dB, $R_{dB}^{(k)}$, derived from the k^{th} iteration.

The residue for OMP decreases as the iteration number k increases. However, the decreasing rate reduces as $k > K = 6$ since all six rays have been identified at $k = K$. The new rays estimated by OMP after $k = K$ are false targets derived from matching the noises. Since the further decrease of residue does not promote sparsity, we should stop the OMP iteration when the residue is less than an appropriate threshold t_r for practical implementations. Without knowing the number of targets in advance, the OMP estimation will most likely either miss true targets or show false targets. Note that the OMP iteration stops at the 15th iteration in Fig. 7 since the number of measurements $M = 16$ for CPA in our example.

The residue for SBL decreases rapidly as the iteration number k increases. Although the residue is very small for large k 's, it does not mean the estimates of \mathbf{c} obtained for large k 's are better than those obtained for smaller k 's. This is because the posterior covariance matrix $\tilde{\Gamma}$ in step 4 of Algorithm 2 becomes ill-conditioned when $k > 13$ for SPA (see Table II) and when $k > 8$ for CPA (see Table III). Moreover, further decrease of the residue only adjusts the estimated \mathbf{c} to minimize the mismatch between \mathbf{y} and $\mathbf{A}\mathbf{c}$, which is due to the additive noises. It does not promote sparsity. This phenomenon is consistent with the findings in [50] which shows that SBL will converge to a l_0 regularized optimization problem when there is no noise. Thus, for practical implementations, we should stop the SBL iteration when the residue is less than an appropriate threshold t_r . In addition, the SBL iteration stops when the posterior covariance matrix $\tilde{\Gamma}$ in step 4 of Algorithm 2 becomes ill-conditioned (see Table II for SPA and Table III for CPA).

Similar to CG, the residue for BLRC does not always decrease as the iteration number k increases. At the beginning of iterations, like SBL, the residue for BLRC decreases as the iteration number k increases. This is because $\hat{\gamma}$ is large and the data fitting term $\|\mathbf{y} - \mathbf{A}\mathbf{c}\|^2$ in (58) is emphasized in this phase. As the iteration proceeds, $\hat{\gamma}$ continues to decrease (see the bottom left subplots in Fig. 6 for SPA and Fig. 7 for CPA). Then, the regularization term $\hat{\sigma}_n^2 \sum_{i=1}^N 2\ln(c_i^2 + \hat{\gamma}^2)$ in (58) becomes more and more important in order to promote

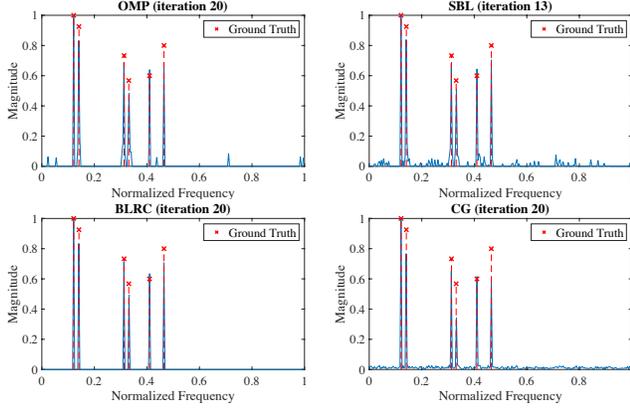


Fig. 11. Recovery of \mathbf{c} at iteration 20 using SPA (Note that the SBL stops at iteration 13) with $\sigma_n = 0.1$

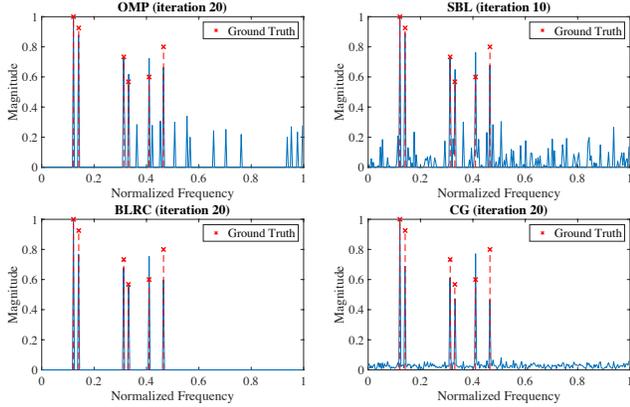


Fig. 12. Recovery of \mathbf{c} at iteration 20 using SPA (Note that the SBL stops at iteration 10) with $\sigma_n = 1$

sparsity. Although the residue could increase in this phase, the rate of residue change commonly decreases as the k increases and the residue gradually becomes nearly a constant. This shows that BLRC not only minimizes the residue but also promotes sparsity. Thus, like CG, we should stop the BLRC iteration if the absolute value of the residue change is less than an appropriate threshold for practical implementations. Note that, unlike SBL, the posterior covariance matrix $\hat{\mathbf{\Gamma}}$ in step 4 of Algorithm 3 does not become ill-conditioned for relatively large k 's (see Table II for SPA and Table III for CPA).

E. Sparse Spectrum Reconstruction and Side Lobes

Note that the SPA with $M = 80$ has a much larger aperture than the CPA with $M = 16$. As it has been shown in [45] that larger $\frac{N}{M}$ ratios imply more local minimums, it will be much easier to reconstruct the original spectrum using SPA than CPA. The CPA example puts the four approaches under a severe test.

1. Sparse Array (SPA) with $M = 80$

Using the SPA samples in Fig. 5, Fig. 11 and Fig. 12 show the estimated \mathbf{c} with $\sigma_n = 0.1$ and $\sigma_n = 1$, respectively.

As shown in Fig. 11, OMP, SBL, BLRC, and CG all recover six rays with high frequency accuracy. The magnitudes of the six rays recovered by CG are the most inaccurate among the four approaches. Regarding side lobes, OMP suggests twenty rays at $k = 20$ and therefore generates 14 distinct side lobes. Both SBL and CG have many small side lobes where the side lobes for CG are like white noises and the side lobes for SBL are like weak rays. It is remarkable that most side lobes are suppressed by BLRC.

The sparse signal recovery properties of SBL and BLRC with SPA can be explained using the two bottom subplots in Fig. 6. For SBL in the bottom right subplot, the six top curves with $\tilde{\sigma}_i > 0.4$ (after the 4th iteration) represent the six true targets. The rest of the curves with small but non-negligible $\tilde{\sigma}_i$'s represent the false targets. It can be seen that SBL has many spurious targets. For BLRC in the bottom left subplot, $\hat{\gamma}$ decreases drastically to promote sparsity as k increases. This is due to the fact that a smaller $\hat{\gamma}^{(k)}$ makes the regularization term in (58) closer to the l_0 norm, as shown in the right subplot of Fig. 4.

The sparse signal recovery properties of SBL and BLRC with SPA can also be seen from the two top subplots in Fig. 10. For both SBL and BLRC, the leftmost six points ($i = 1, 2, \dots, 6$) with the six smallest weights represent the six true targets. Those points in the middle and the right side of the two top subplots are with very large weights and will generate essentially zero c_i 's. The rest of the points with moderate weights (which are not large enough to generate negligible c_i 's) will be considered as false targets. It can be seen that SBL has many more spurious targets than BLRC because many weights of SBL are not large enough to generate negligible c_i 's.

When σ_n increases from 0.1 to 1, the magnitudes of false targets derived from OMP, SBL, and CG increase drastically (comparing Fig. 11 with Fig. 12). However, the spurious targets are suppressed by BLRC for both $\sigma_n = 0.1$ and $\sigma_n = 1$. Since the largest ray magnitude is 1 as shown in Table I, it is concluded that BLRC can suppress spurious targets in low signal-to-noise ratio (SNR).

2. Coprime Arrays (CPA) with $M = 16$

Using the CPA samples in Fig. 5, Fig. 13 and Fig. 14 show the estimated \mathbf{c} with $\sigma_n = 0.01$ and $\sigma_n = 0.1$, respectively.

From Fig. 13 and Fig. 14, we observe that OMP cannot resolve the third and fourth rays. OMP also misses the fifth ray when $\sigma_n = 0.1$. In addition, there are quite a few side lobes that will be interpreted as targets in radar applications. Note that the iteration of OMP stops at 15 in this simulation because the number of measurements $M = 16$. Both CG and SBL can resolve the six rays but have large side lobes. The side lobes of CG are larger than those of SBL. Remarkably, BLRC suppresses spurious targets well.

Similar to SPA, the sparse signal recovery properties of SBL and BLRC using CPA can be seen from the two

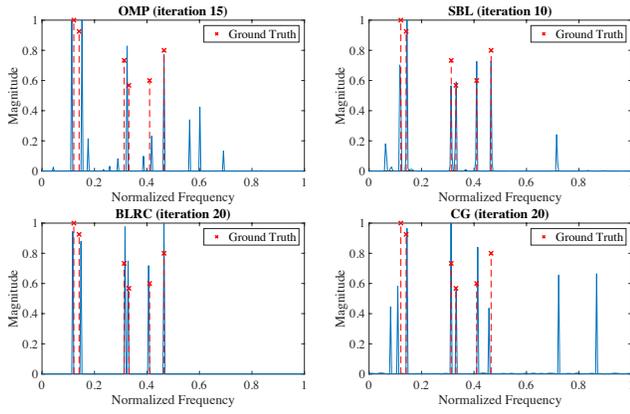


Fig. 13. Recovery of \mathbf{c} at iteration 20 using CPA (Note that the OMP stops at iteration 15 and SBL stops at iteration 10) with $\sigma_n = 0.01$

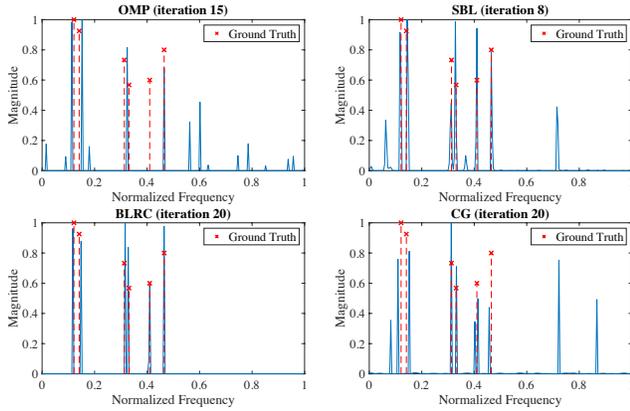


Fig. 14. Recovery of \mathbf{c} at iteration 20 using CPA (Note that the OMP stops at iteration 15 and SBL stops at iteration 8) with $\sigma_n = 0.1$

bottom subplots in Fig. 7. For SBL in the bottom right subplot, the six top curves corresponding to the six largest $\tilde{\sigma}_i$ represent the six true targets. The rest of the curves with smaller $\tilde{\sigma}_i$'s represent the false targets. However, as the $\tilde{\sigma}_i$'s representing some of the false targets are only slightly smaller than the $\tilde{\sigma}_i$'s representing the true targets, the false targets are almost as strong as the true targets. For BLRC in the bottom left subplot, $\hat{\gamma}$ decreases so as to promote sparsity as k increases. The implication of reducing $\hat{\gamma}^{(k)}$ is shown in Fig. 4, which has been discussed previously for SPA and will not be repeated here.

The sparse signal recovery properties of SBL and BLRC using CPA can also be seen from the two bottom subplots in Fig. 10. The first six points with $i = 1, 2, \dots, 6$ with the six smallest weights represent the six true targets. The discussions are the same as those for SPA and are omitted here. Again, SBL has many more spurious targets than BLRC.

When σ_n increases from 0.01 to 0.1, the magnitudes of false targets derived from OMP, SBL, and CG increase (comparing Fig. 13 with Fig. 14). However, the spurious targets are still suppressed by BLRC. It is concluded that BLRC can suppress spurious targets in moderate SNR's even with a small array ($M = 16$).

In summary, BLRC provides sparser solution than SBL with sparse array input. This outcome can be attributed to BLRC's ability to generate a more accurate estimation of the noise standard deviation, σ_n , while SBL tends to produce a much smaller value for σ_n estimation. (Further details can be found in the subsequent section.) This issue is exacerbated when sparse array input is utilized, as non-uniform random sampling further worsens the ambiguity problem. From (58) and (61), we know that σ_n^2 is the weight on the penalty term. When σ_n^2 converges to a reasonable non-zero value, further decrement of $\hat{\gamma}^{(k)}$ in BLRC will make $\ln(1 + \frac{|c_i|^2}{\hat{\gamma}^2})$ in (58) approach $\|c\|_0$, i.e., the sparsest solution.

F. Noise Standard Deviation Estimation

In the top right subplots in Fig. 6 and Fig. 7 for SPA and CPA, respectively, we show the noise standard deviations in dB $20 \log_{10} \tilde{\sigma}_n^{(k)}$ for SBL and $20 \log_{10} \hat{\sigma}_n^{(k)}$ for BLRC) as functions of the iteration index k . As the iteration number k increases, $\tilde{\sigma}_n^{(k)}$ for SBL decreases to very small values. However, unlike SBL, BLRC provides a reasonable estimate of the noise standard deviation. Note that the estimated $\hat{\sigma}_n$ is the combined effect of the original additive white Gaussian noise (AWGN) and the insufficient-sampling noise [58]. Thus, $\hat{\sigma}_n$ estimated by BLRC is greater than the AWGN σ_n . When σ_n is large, it will be shown in the next section that the estimated $\hat{\sigma}_n \approx \sigma_n$ because the sampling noise now is negligible compared to AWGN (see the right subplot in Fig. 15).

From (61), one can see that SBL with a very small $\tilde{\sigma}_n^{(k)}$ de-emphasizes the regularization term and therefore does not promote sparsity at the end of the iterative process (see the top right subplots in Fig. 6 and Fig. 7). However, from (58), one can see that BLRC promotes sparsity since the regularization term remains effective in the entire iteration process as $\hat{\sigma}_n \geq \sigma_n$.

To understand why BLRC has a superior performance over SBL in estimating σ_n , rewrite the σ_n^2 updating strategy of BLRC in step 9 of Algorithm 3 as:

$$\hat{\sigma}_n^2 = \frac{\|\mathbf{y} - \mathbf{A}\hat{\mathbf{c}}\|^2}{M - \text{Tr}(\mathbf{I} - \frac{2}{\hat{\gamma}^2} \hat{\mathbf{Q}}\hat{\mathbf{\Gamma}})} \quad (64)$$

As shown in Fig. 10, $\frac{2}{\hat{\gamma}^2} \hat{\mathbf{Q}}$ in (64) is approximately a constant diagonal loading matrix of \mathbf{H}_{blrc} where $\hat{\mathbf{\Gamma}} = \mathbf{H}_{blrc}^{-1}$. Since there are $N - K$ very large constant diagonal terms in $\frac{2}{\hat{\gamma}^2} \hat{\mathbf{Q}}$, \mathbf{H}_{blrc} is approximately equal to $\frac{2}{\hat{\gamma}^2} \hat{\mathbf{Q}}$. Thus, $\frac{2}{\hat{\gamma}^2} \hat{\mathbf{Q}}\hat{\mathbf{\Gamma}}$ in (64) is approximately a diagonal matrix with $(N - K)$ 1's and (K) 0's. Then, the denominator in (64) is approximately equal to $M - K$. As the numerator in (64) is approximately equal to $M\sigma_n^2$ if $\hat{\mathbf{c}} \approx \mathbf{c}$, we have $\hat{\sigma}_n^2 \approx \frac{M}{M-K}\sigma_n^2$.

Even though the σ_n^2 updating strategy of SBL in step 6 of Algorithm 2:

$$\tilde{\sigma}_n^2 = \frac{\|\mathbf{y} - \mathbf{A}\tilde{\mathbf{c}}\|^2}{M - \text{Tr}(\mathbf{I} - \tilde{\mathbf{\Gamma}}\tilde{\mathbf{\Sigma}})} \quad (65)$$

TABLE IV
Parameter Settings

	Fig. 10 MSE vs. σ_n	Fig. 11 MSE vs. K	Fig. 11 MSE vs. M
SBL	$\tilde{\sigma}_n = 1, \tilde{\sigma}_i = 0.1, \forall i$		
BLRC	$\hat{\sigma}_n = 1, \hat{\gamma} = 0.1$		
CG	$\hat{\sigma}_n = 5 \times 10^{-2}$ $\hat{\gamma} = 5 \times 10^{-3}$	$\hat{\sigma}_n = 5 \times 10^{-2}$ $\hat{\gamma} = 5 \times 10^{-4}$	$\hat{\sigma}_n = 5 \times 10^{-2}$ $\hat{\gamma} = 5 \times 10^{-3}$
Noises	$\sigma_n = .05 \sim 1.0$	$\sigma_n = 0.1$	$\sigma_n = 0.1$
Signal K rays	$K = 6$ rays listed in Table I	$K = 1 \sim 12$ random rays	$K = 6$ rays listed in Table I
SPA M samples	$M = 80$ random samples	$M = 80$ random samples	$M = 32 \sim 192$ random samples

is of the same form as (64), $\tilde{\Sigma}$ is nearly singular and its diagonal terms are far from constant. Thus, SBL does not estimate σ_n^2 accurately. In fact, as $\tilde{\sigma}_n^{(k)}$ becomes smaller, the penalty term for non-sparsity in (61) becomes smaller as well. Then, as the iteration proceeds, the data fitting term $\|y - \mathbf{A}\mathbf{c}\|^2$ in (61) becomes more and more important and the estimated $\tilde{\sigma}_n^{(k)}$ becomes disproportionately small.

G. Sensitivity with respect to σ_n , K and M

Sensitivities of the four spectrum reconstruction results (derived by OMP, SBL, CG, and BLRC) with respect to the noise standard deviation σ_n , number of targets K , and number of samples M are shown in Fig. 15 and Fig. 16. In these figures, the normalized mean squared error (MSE) is defined as

$$\text{MSE (dB)} = 10 \log_{10} \left[\frac{1}{N} \left\| \frac{\mathbf{c}}{\max_i \{ |c_i| \}} - \frac{\bar{\mathbf{c}}}{\max_i \{ |\bar{c}_i| \}} \right\|_2^2 \right] \quad (66)$$

where $N = 256$ and \mathbf{c} is the true complex spectrum of the receive signal and $\bar{\mathbf{c}}$ represents the recovered spectrum using OMP, SBL, CG, or BLRC. Note that MSE may not reflect the spectrum reconstruction well when the estimated ray frequencies are slightly off from the corresponding true ray frequencies. In addition, MSE does not show sparsity characteristics. Thus, this simulation is conducted only for SPA with large M where sparse signal recovery can be done relatively easily.

The MSE at each point plotted in Fig. 15 and Fig. 16 is the average of 100 realizations of random noises, random rays and random samples, respectively. The parameters for the three Bayesian approaches (SBL, CG, and BLRC) are outlined in Table I. The number of samples, ray parameters, and noise standard deviations are also shown in Table I.

The average MSE's of 100 realizations obtained by OMP, SBL, CG, and BLRC as noise standard deviation increases from 0.05 to 1 are shown in the left subplot of Fig. 15. BLRC shows the best performance, which is followed by SBL. CG performs better than OMP when noise is small. But CG performs worse than OMP with

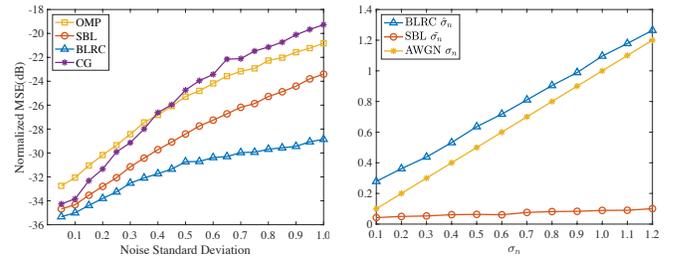


Fig. 15. (Left) The normalized MSE as noise standard deviation increases; (Right) The estimated σ_n by BLRC and SBL

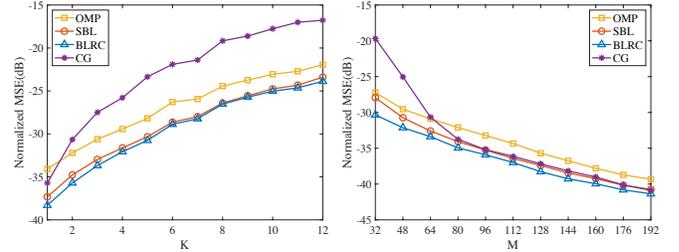


Fig. 16. (Left) The normalized MSE as the number of targets K changes; (Right) The normalized MSE as the number of measurements M changes

large noises, which is due to the fact that the parameters of CG are not adjusted as noise increases.

The right subplot in Fig. 15 shows the average estimated noise standard deviations of 100 realizations obtained by BLRC and SBL for several different noise levels. It can be seen that the SBL estimates are inaccurate. On the other hand, BLRC estimates are larger than, but very close to, the true noise values. The higher the noise level is, the more accurate the BLRC estimate is. This is because, as mentioned before, there are mainly two kinds of noise in sparse signal recovery problems [58]: additive white Gaussian noise (AWGN) and random sampling noise. What BLRC estimate is the combined effect of these two kinds of noise. When AWGN is large, AWGN becomes dominant and the random sampling noise becomes negligible.

The average MSE's of 100 realizations obtained by OMP, SBL, CG and BLRC as the number of rays, K , increases from 1 to 12 are shown in the left subplot in Fig. 16. Table I shows the parameters used for the simulations. Similarly, BLRC shows the best performance followed by SBL. CG performs better than OMP when $K = 1$. But CG performs worse than OMP for all other K 's, which is due to the fact that optimum CG parameters used for simulation need to be found by trial and error in order to get good performances. Here, the parameters are chosen for $K = 1$ and not adjusted further as K increases.

The average MSE's of 100 realizations obtained by OMP, SBL, CG, and BLRC as the number of SPA elements, M , increases from 32 to 192 are shown in the right subplot in Fig. 16. Table I shows the parameters used for simulations. BLRC has the best performance and SBL is the second-best. CG performs better than OMP when M is large. But CG performs worse than OMP for small

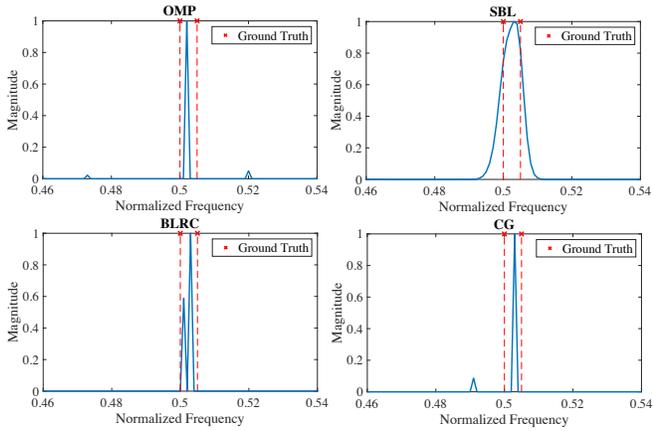


Fig. 17. Resolution of Two Rays of Same Magnitude

M 's. Again, the parameters of CG are not adjusted as M changes.

H. Resolution

Using SPA with $M = 80$, OMP, CG, SBL, and BLRC have very similar resolution performances. Their estimated results are all accurate because the aperture of SPA with $M = 80$ is large enough and sufficient measurements are obtained. Here, the CPA with $M = 16$ is used to test the resolution of the proposed BLRC approach, as high resolution cannot be achieved easily with such a small aperture. It will be shown that BLRC outperforms OMP, CG, and SBL.

In the first example, consider two rays with equal amplitudes. Their normalized frequencies are $\frac{500}{N}$ and $\frac{505}{N}$ where $N = 1000$. Fig. 17 shows that OMP, SBL and CG all fail to resolve the two rays while BLRC can distinguish them. Note that SBL stops at iteration 6 in this example because the matrix to be inverted becomes singular at iteration 7 (see numerical stability in Section VI.C).

In the second example, consider two rays with different magnitudes: 1 and 0.2. Let their normalized frequencies be $\frac{500}{N}$ and $\frac{510}{N}$, respectively, where $N = 1000$. Again, Fig. 18 shows that OMP, SBL and CG all fail to resolve the two rays while BLRC can distinguish them. Similarly, SBL stops at iteration 6 due to the near singular issue.

Although the two rays can be distinguished by BLRC in both examples, the estimated frequencies and magnitudes are slightly off because the targets are so close to each other. Even worse when one is weaker than the other, the weak target estimation is biased toward the strong target. Fortunately, for current autonomous driving applications, these results are satisfactory because the ability to separate targets is more crucial (e.g., early detection of crossing pedestrians [59]).

Note that in both Fig. 17 and Fig. 18, SBL has spurious targets while BLRC does not. This is not shown in these two zoom-in figures. The better performances of BLRC over SBL on resolution and sparsity are due to

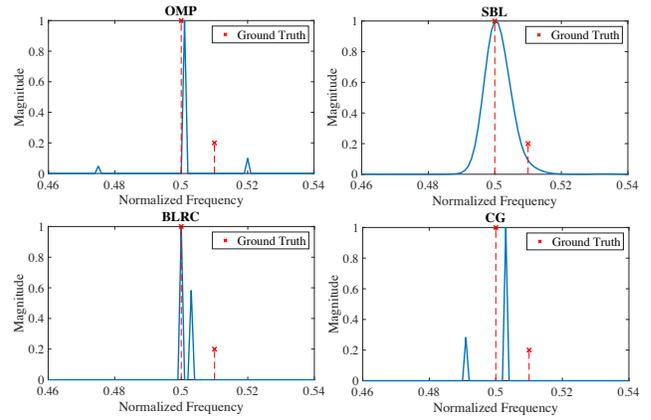


Fig. 18. Resolution of Two Rays with Different Magnitudes

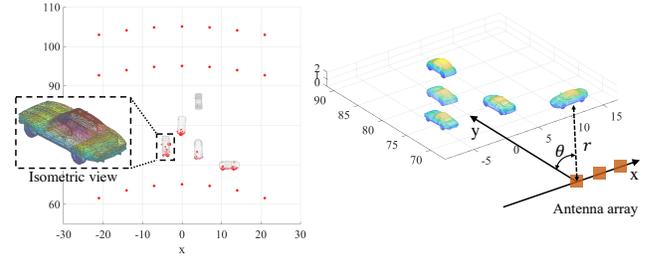


Fig. 19. Simulation scenario for range-azimuth angle estimation (top view and side view)

the fact that BLRC provides a reasonable estimation of the noise standard deviation σ_n^2 , but SBL does not (as discussed in Sections VI.F and VI.G).

VII. Automotive Radar Applications

In this section, simulated automotive radar signals are processed to demonstrate the performance of the proposed BLRC method.

A. Image Radar Signal Processing

In order to locate targets in 2D, we need to receive the back scattered electromagnetic signal in two distinct dimensions [60]–[62]. Consider the following frequency-modulated continuous-wave (FMCW) transmitted pulse:

$$s(t) = e^{j2\pi(f_c + \frac{\alpha}{2})t}, \quad 0 \leq t \leq T_c \quad (67)$$

where f_c is the carrier frequency, α is the chirp slope, and T_c is the duration of one chirp.

Consider Fig. 19 as an example where the first antenna array element is set as the origin. Since the targets are in the far zone, plane wave approximation is valid. Define $P(r_l, \theta_l)$ as the position of l^{th} target in the polar coordinates, then the round-trip propagation time between the l^{th} target and the i^{th} array element is given by

$$t_l = \frac{2r_l - id \sin \theta_l}{v} \quad (68)$$

where d is the antenna spacing and v is the speed of light.

To estimate the range $\{r_l\}$ and angle $\{\theta_l\}$, the t_l -delayed received signal at the i^{th} antenna element is

mixed with $s^*(t)$ by the mixer, filtered by low-pass filter (LPF), and sampled at the sampling rate f_s by analog-to-digital converter (ADC). Then, the n^{th} ADC samples at the i^{th} antenna element is:

$$\begin{aligned} \check{y}(n, i) \approx & \sum_{l=1}^K \check{a}_l \exp \left\{ j2\pi \left[\frac{2r_l}{v} \alpha n \Delta t - \frac{d \sin \theta_l}{\lambda} i + \frac{2r_l}{\lambda} \right] \right\} \\ & + \check{\epsilon}(n, i), \quad n = 1, 2, \dots, N_s \end{aligned} \quad (69)$$

where K is the number of targets, $\Delta t = 1/f_s$ is the sampling interval, N_s is the number of ADC samples, $\check{\epsilon}$ is the additive noise and \check{a}_l is the l^{th} target strength. Note that the received target strength $|\check{a}_l|$ is proportional to the antenna gain, RCS of the target and $1/r_l$.

To obtain the range estimation, performs FFT on the N_s ADC samples in (69):

$$y(p, i) = \mathcal{F}_n \{ \check{y}(n, i) \} = \sum_{n=0}^{N_s-1} \check{y}(n, i) e^{-j2\pi \frac{p}{N_s} n} \quad (70)$$

Representing (70) in the same form as (63) with full samples, we have

$$y(p, i) = \sum_{l=1}^K a_l(p) \exp \{ j(2\pi f_l \frac{i}{N} + \phi_l) \} + \epsilon(p, i) \quad (71)$$

where $a_l(p) = \mathcal{F}_n \{ \check{a}_l e^{j2\pi [\frac{2r_l}{v} \alpha n \Delta t]} \}$, $f_l = -\frac{L \sin \theta_l}{\lambda}$, $\phi_l = \frac{4\pi r_l}{\lambda}$ and $\epsilon(p, i) = \mathcal{F}_n \{ \check{\epsilon}(n, i) \}$. Note that the d in (69) is the minimum element spacing in antenna array and $d = \frac{L}{N}$ where L is the full antenna aperture if none of the N array elements is missing. Therefore, the SPA and CPA used in previous sections can be obtained by selecting array elements based on the corresponding pattern as shown in Fig. 5.

Note that, when $p = \frac{2r_l}{v} \alpha N_s \Delta t$, $|a_l(p)|$ gets its maximum value: $N_s |\check{a}_l|$. Usually, if the maximum of $|a_l(p)|$ occurs at $p = p^*$, a local maximum of $|y(p, i)|$ also occurs at $p = p^*$. Thus, the indexes of local maximums of $|y(p, i)|$ offer the range information $\{ \frac{2r_l}{v} \alpha N_s \Delta t \}$.

To find the angle information, we can apply sparse spectrum reconstruction approaches (with respect to i) to $y(p^*, i)$ in (71). For automotive radar applications, the number of samples M which is usually much less than N . Since the number of dominant targets for a given range is small, there will be no problem to find the spatial frequency f_l corresponding to p^* in (71). From f_l and p^* , we can then find the angle and range of target, respectively. Note that, without loss of generality, the Doppler processing is omitted for simplicity.

B. Automotive Radar Signal Simulator

A physical optics (PO) based electromagnetic simulator inspired by [63] is developed to generate typical FMCW automotive radar signals. In our simulator, targets are composed of many small triangular facets. To generate the received radar signal, the simulator at first computes the induced currents on each facet of all targets illuminated by the incident radar signal based

TABLE V
RADAR SETTING

Parameter	Setting
Carrier Frequency f_c	79GHz
Wavelength λ	3.797 mm
Chirp Slope α	10MHz/ μ sec
Sampling Frequency f_s	20MHz
ADC Samples N_s	1024
Antenna Number M	16
Antenna Element Spacing d	$\lambda/2$
Target Surface Resistivity	0.1
Car1 Position	[12,70,0] in meters
Car2 Position	[-4.1,75,0] in meters
Car3 Position	[-0.1,80.1,0] in meters
Car4 Position	[4.21,86,0] in meters
Car5 Position	[4.15,74,0] in meters
Fixed Target Ranges	65,95,105 in meters
Fixed Target Angles	-21,-14,-7,0,7,14,21 in degrees

on the principle of PO. Then, the simulator sums up, at the radar receive antenna, all back-scattered electromagnetic fields radiated from these induced currents on all facets of all targets. Using this simulator, scattered electromagnetic fields and radar cross section (RCS) of each target in realistic target scenes can be generated. Compared to the computationally intensive and resource consuming full-wave simulation, the PO based simulator is able to highlight dominant wave features and provide computational efficiency without compromising much on numerical accuracy if the high frequency approximation for electromagnetic wave propagation and scattering is valid.

C. Numerical Example

An automotive radar with a 16-element coprime array (as marked in the top subplot of Fig. 5) is considered. A scene with five cars and twenty-one fixed point targets (corner reflectors) is shown in the left subfigure of Fig. 19. The first element of the coprime array is located at the origin as shown in the right subfigure of Fig. 19. Specific parameters for all these targets and the radar are shown in Table V.

Then FMCW signal in (67) with a chirp rate 10MHz/ μ sec is transmitted. In our simulator, all targets are composed of many small facets. For example, the automobile in isometric view in Fig. 19 is composed of 7226 facets. Then we use our proposed automotive radar signal simulator to generate the received signals at the 16 coprime array elements. The additive white Gaussian noise (AWGN) $\check{\epsilon}$ in (69) is generated with two different standard deviations, $\sigma = 0.03$ and $\sigma = 0.3$.

After mixer, LPF, and ADC, the signal received by the i^{th} array element becomes the ADC sample $\check{y}(n, i)$ in (69). Perform FFT on $\check{y}(n, i)$ with respect to n to obtain

$y(p, i) = \mathcal{F}_n\{\tilde{y}(n, i)\}$ as shown in (70). Then, perform OMP, CG, SBL, and BLRC on $y(p, i)$ with respect to i to obtain the 2D radar spectrum $\tilde{y}(p, f)$ for each of these four approaches. Using $r = \frac{pv}{2\alpha N_s \Delta t}$ to convert p to the range variable r and using $\theta = \arcsin \frac{-\lambda f}{L}$ to map f to the angle θ , the 2D radar image plot $|\tilde{y}(r, \theta)|$ recovered by the four approaches mentioned in Section VI with two different AWGN strengths are shown in Fig. 20.

As shown in Fig. 20, with pruning, SBL results still exhibit spurious targets. Moreover, the number of spurious targets increases with noise strength. CG results also show spurious targets at range equal to 65 m, even though the CG parameters have been optimized by trial and error. Note that optimizing parameters by trial and error is not possible in practice, since the targets are supposed to be unknown. For OMP, we set the iteration number to be 8, which is slightly larger than the maximum target number at each range. This is not possible in practical applications either because the targets are unknown. It is remarkable that, BLRC robustly delivers the least spurious solution and provides optimal resolution of the target images for both AWGN strengths. This is consistent with our previous analyses.

VIII. Conclusion

In this paper, we propose a Bayesian linear regression algorithm, BLRC, which uses the non-conjugate Cauchy prior. Then we focus on the comparisons among three Bayesian linear regression approaches (i.e., BLRC, CG, and SBL) when sparse array measurement is used as input.

Firstly, BLRC can be considered as a significant improvement over the CG approach because BLRC provides a systematic updating scheme for the hyper-parameters which is absent from the CG approach. This makes BLRC more feasible than CG as CG is sensitive to the selection of the hyper-parameters. Furthermore, the systematic updating scheme empowers BLRC the capability to reduce the number of local minimums as shown in Section IV. This greatly reduces the chances of BLRC being trapped in a local minimum at the early stage of the iteration process. As is shown in Section VI and Section VII, BLRC outperforms CG in various scenarios even when CG has the best choice of its hyper-parameters. This makes BLRC more practical than CG, especially when it is used in the highly dynamic automotive scenarios.

Secondly, BLRC can also be considered as a significant improvement over the well-known SBL approach. Both BLRC and SBL are Bayesian linear regression approaches and have the same iterative updating steps for the solution and hyper-parameters. The formulas for updating the solutions in BLRC and SBL are also similar to each other. However, there are key differences between BLRC and SBL. Only two hyper-parameters need to be handled in BLRC due to the use of Cauchy prior, while the number of hyper-parameters to be handled is large

for SBL. Thus, compared with SBL, BLRC has a more compact latent space.

Thirdly, comprehensive numerical analyses are conducted to demonstrate the superior performances of BLRC when sparse array measurement is used as input. Special attention has been paid to the comparisons between BLRC and the well-known SBL approach. It is shown that BLRC is more numerically robust than SBL and provides a more accurate estimate of the noise variance. Since BLRC provides a more accurate estimation of the noise variance, it tends to yield the sparsest solution, whereas SBL is susceptible to generating spurious targets. The issue of spurious targets becomes more pronounced in scenarios involving sparse array or low-SNR input, as both factors contribute to exacerbating the ambiguity problem. From the resolution point of view, the sparsest solution tends to separate targets even though they are very close to each other. Thus, BLRC provides a higher resolution result. Based on the IR- l_2 interpretations, it is remarkable to see that all advantages of BLRC over SBL are originated from the simple fact that BLRC uses the long-tailed Cauchy prior which leads to a more compact latent space.

Finally, the application of BLRC to sparse MIMO radar array signal processing is presented. Compared with the reconstructed radar images of other sparse signal recovery algorithms, the performance of BLRC successfully demonstrates its efficiency in producing high-resolution radar images with the least false targets, which is critical to the development of advanced driver-assistance systems (ADAS) and autonomous driving (AD) applications.

Appendix A

The first expectation in (16) is further simplified below:

$$E_{\mathbf{c}|\mathbf{y}}[||\mathbf{y} - \mathbf{A}\mathbf{c}||^2] = |\mathbf{y}|^2 - \hat{\mathbf{c}}^T \mathbf{A}^T \mathbf{y} - \mathbf{y}^T \mathbf{A} \hat{\mathbf{c}} + E_{\mathbf{c}|\mathbf{y}}[\mathbf{c}^T \mathbf{A}^T \mathbf{A} \mathbf{c}] \quad (72)$$

As $\mathbf{c}^T \mathbf{A}^T \mathbf{A} \mathbf{c}$ is a scalar,

$$\mathbf{c}^T \mathbf{A}^T \mathbf{A} \mathbf{c} = \text{Tr}(\mathbf{c}^T \mathbf{A}^T \mathbf{A} \mathbf{c}) = \text{Tr}(\mathbf{A} \mathbf{c} \mathbf{c}^T \mathbf{A}^T)$$

Then,

$$\begin{aligned} E_{\mathbf{c}|\mathbf{y}}[\mathbf{c}^T \mathbf{A}^T \mathbf{A} \mathbf{c}] &= \text{Tr}(\mathbf{A} E_{\mathbf{c}|\mathbf{y}}[\mathbf{c} \mathbf{c}^T] \mathbf{A}^T) \\ &= \text{Tr}(\mathbf{A} \hat{\mathbf{\Gamma}} \mathbf{A}^T) + \text{Tr}(\mathbf{A} \hat{\mathbf{c}} \hat{\mathbf{c}}^T \mathbf{A}^T) \\ &= \text{Tr}(\mathbf{A}^T \mathbf{A} \hat{\mathbf{\Gamma}}) + \hat{\mathbf{c}}^T \mathbf{A}^T \mathbf{A} \hat{\mathbf{c}}. \end{aligned} \quad (73)$$

Substituting the above equation into (72), we have

$$E_{\mathbf{c}|\mathbf{y}}[||\mathbf{y} - \mathbf{A}\mathbf{c}||^2] = ||\mathbf{y} - \mathbf{A}\hat{\mathbf{c}}||^2 + \text{Tr}(\mathbf{A}^T \mathbf{A} \hat{\mathbf{\Gamma}}) \quad (74)$$

Appendix B

Set $a \rightarrow 0$ and $b \rightarrow 0$, then we extend the lower bound in (37) of a scalar c to a lower bound of a vector \mathbf{c} , and use the notation τ_i instead of ξ_i :

$$p(\mathbf{c}|\boldsymbol{\tau}) = \prod_{i=1}^N p(c_i) \geq \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}^{-1}) \quad (75)$$

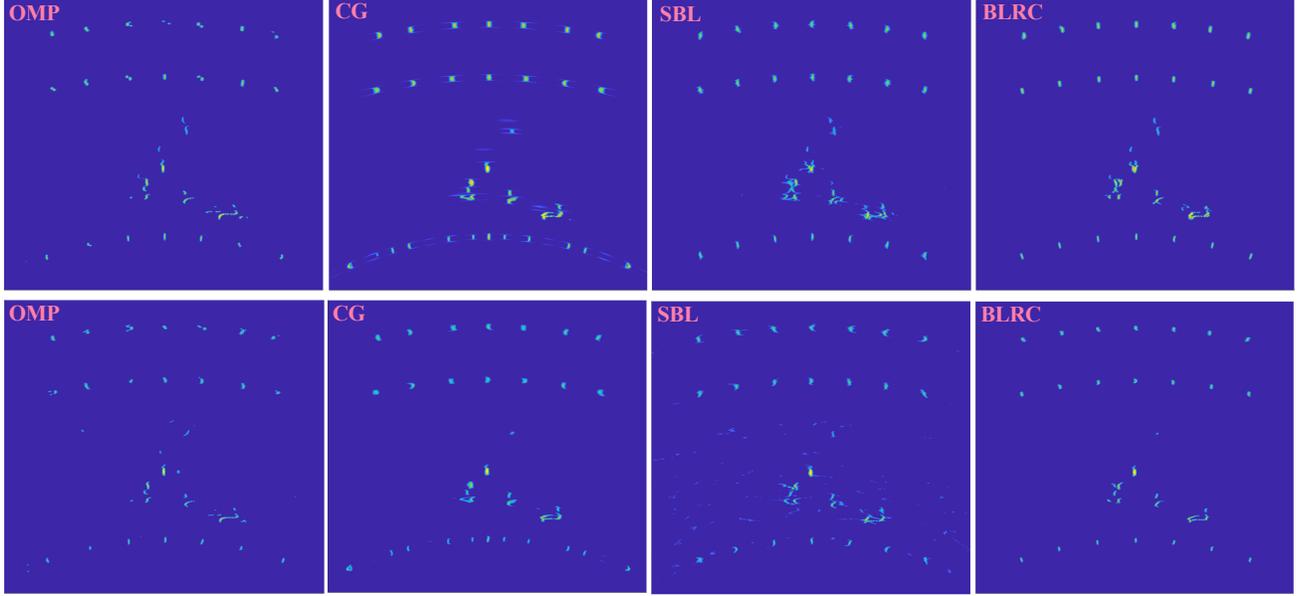


Fig. 20. Radar Imaging Performances for OMP, CG, SBL, and BLRC based on Realistic Target Scene Simulation (Image Dynamic Range: 70dB). Upper Row: $\sigma = 0.03$. Lower Row: $\sigma = 0.3$.

where Σ has been defined as $\text{diag}(\boldsymbol{\tau})$ in (3). In (75), the dependence on $\boldsymbol{\tau}$ is explicitly shown. Recall the optimization problem for hyper-parameters in (9):

$$\begin{aligned} \{\tilde{\boldsymbol{\tau}}, \tilde{\sigma}_n\} &= \arg \max_{\boldsymbol{\tau}, \sigma_n} \ln\{p(\mathbf{y}|\boldsymbol{\tau}, \sigma_n)\} \\ &= \arg \max_{\boldsymbol{\tau}, \sigma_n} \ln\left\{\int p(\mathbf{y}|\mathbf{c}, \sigma_n)p(\mathbf{c}|\boldsymbol{\tau})d\mathbf{c}\right\} \end{aligned} \quad (76)$$

Substituting (4) and (75) into the integral in (76), the lower bound of the evidence distribution $p(\mathbf{y}|\boldsymbol{\tau}, \sigma_n)$ is derived as follows:

$$p(\mathbf{y}|\boldsymbol{\tau}, \sigma_n) \geq \int \mathcal{N}(\mathbf{A}\mathbf{c}, \sigma_n^2\mathbf{I})\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}^{-1})d\mathbf{c} = \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_y^{-1}) \quad (77)$$

with

$$\boldsymbol{\Sigma}_y^{-1} = \sigma_n^2\mathbf{I} + \mathbf{A}\boldsymbol{\Sigma}^{-1}\mathbf{A}^T \quad (78)$$

According the equation (36) in [38], the following is the negative logarithm of the lower bound of $p(\mathbf{y}|\boldsymbol{\tau}, \sigma_n)$:

$$L_{sbl} = \ln|\boldsymbol{\Sigma}_y^{-1}| + \mathbf{y}^T\boldsymbol{\Sigma}_y\mathbf{y} + C \quad (79)$$

where C is a constant. Thus, optimization problem in (76) is equivalent to minimizing the cost function L_{sbl} in (79).

As shown in Appendix C,

$$\mathbf{y}^T\boldsymbol{\Sigma}_y\mathbf{y} = \min_{\mathbf{c}} \left(\frac{1}{\sigma_n^2}\|\mathbf{y} - \mathbf{A}\mathbf{c}\|^2 + \mathbf{c}^T\boldsymbol{\Sigma}\mathbf{c} \right) \quad (80)$$

Thus, minimizing the cost function L_{sbl} with respect to $\boldsymbol{\tau}$ and σ_n can be rewritten in \mathbf{c} -space as [50]:

$$\min_{\mathbf{c}} J_{sbl}(\mathbf{c}), \quad J_{sbl}(\mathbf{c}) = \|\mathbf{y} - \mathbf{A}\mathbf{c}\|^2 + h_{sbl}(\mathbf{c}) \quad (81)$$

where the minimization with respect to $\boldsymbol{\tau}$ and σ_n is done in deriving the regularization $h_{sbl}(\mathbf{c})$ to encourage a sparse solution:

$$h_{sbl}(\mathbf{c}) = \min_{\boldsymbol{\tau}, \sigma_n} \sigma_n^2 \left(\mathbf{c}^T\boldsymbol{\Sigma}\mathbf{c} + \ln|\sigma_n^2\mathbf{I} + \mathbf{A}\boldsymbol{\Sigma}^{-1}\mathbf{A}^T| \right) \quad (82)$$

Appendix C

To find the solution to

$$\min_{\mathbf{c}} \left(\frac{1}{\sigma_n^2}\|\mathbf{y} - \mathbf{A}\mathbf{c}\|^2 + \mathbf{c}^T\boldsymbol{\Sigma}\mathbf{c} \right), \quad (83)$$

set the derivative with respect to \mathbf{c} to be 0:

$$\frac{\partial}{\partial \mathbf{c}} \left[\frac{1}{\sigma_n^2}\|\mathbf{y} - \mathbf{A}\mathbf{c}\|^2 + \mathbf{c}^T\boldsymbol{\Sigma}\mathbf{c} \right]_{\mathbf{c}=\mathbf{c}_o} = 0 \quad (84)$$

so we can get the optimal \mathbf{c}_o :

$$\mathbf{c}_o = (\sigma_n^2\boldsymbol{\Sigma} + \mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{y} \quad (85)$$

Substitute \mathbf{c}_o back to the original objective function in (83), we have

$$\begin{aligned} &\frac{1}{\sigma_n^2}\|\mathbf{y} - \mathbf{A}\mathbf{c}_o\|^2 + \mathbf{c}_o^T\boldsymbol{\Sigma}\mathbf{c}_o \\ &= \frac{1}{\sigma_n^2}\mathbf{y}^T [\mathbf{I} - \mathbf{A}(\sigma_n^2\boldsymbol{\Sigma} + \mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T]\mathbf{y}. \end{aligned} \quad (86)$$

Applying the Woodbury's identity

$$\mathbf{E}^{-1} - \mathbf{E}^{-1}\mathbf{B}(\mathbf{D} + \mathbf{C}\mathbf{E}^{-1}\mathbf{B})^{-1}\mathbf{C}\mathbf{E}^{-1} = (\mathbf{E} + \mathbf{B}\mathbf{D}^{-1}\mathbf{C})^{-1}, \quad (87)$$

to simplify $[\mathbf{I} - \mathbf{A}(\sigma_n^2\boldsymbol{\Sigma} + \mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T]$ in (86), we get

$$\begin{aligned} &\frac{1}{\sigma_n^2}\mathbf{y}^T [\mathbf{I} - \mathbf{A}(\sigma_n^2\boldsymbol{\Sigma} + \mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T]\mathbf{y} \\ &= \mathbf{y}^T (\sigma_n^2\mathbf{I} + \mathbf{A}\boldsymbol{\Sigma}^{-1}\mathbf{A}^T)^{-1}\mathbf{y} \end{aligned} \quad (88)$$

Thus, we have

$$\mathbf{y}^T (\sigma_n^2\mathbf{I} + \mathbf{A}\boldsymbol{\Sigma}^{-1}\mathbf{A}^T)^{-1}\mathbf{y} = \min_{\mathbf{c}} \left(\frac{1}{\sigma_n^2}\|\mathbf{y} - \mathbf{A}\mathbf{c}\|^2 + \mathbf{c}^T\boldsymbol{\Sigma}\mathbf{c} \right). \quad (89)$$

Acknowledgment

The authors would like to thank Prof. Ivan Selesnick (New York University) and Satish Ravindran (NXP Semiconductors) for their helpful suggestions and discussions on this work. Additionally, we would like to extend our gratitude to the editor and anonymous reviewers for their valuable comments which have been instrumental in enhancing the quality and clarity of this work. We appreciate their time and effort in helping us refine this work.

REFERENCES

- [1] F. Engels, P. Heidenreich, M. Wintermantel, L. Stacker, M. Al Kadi, and A. M. Zoubir
Automotive radar signal processing: Research directions and practical challenges
IEEE Journal of Selected Topics in Signal Processing, vol. 15, no. 4, pp. 865–878, 2021.
- [2] S. Sun, A. P. Petropulu, and H. V. Poor
Mimo radar for advanced driver-assistance systems and autonomous driving: Advantages and challenges
IEEE Signal Processing Magazine, vol. 37, no. 4, pp. 98–117, 2020.
- [3] B. Liao
Fast angle estimation for mimo radar with nonorthogonal waveforms
IEEE Transactions on Aerospace and Electronic Systems, vol. 54, no. 4, pp. 2091–2096, 2018.
- [4] H. Ren and A. Manikas
Mimo radar with array manifold extenders
IEEE Transactions on Aerospace and Electronic Systems, vol. 56, no. 3, pp. 1942–1954, 2019.
- [5] P. P. Vaidyanathan and P. Pal
Sparse sensing with co-prime samplers and arrays
IEEE Transactions on Signal Processing, vol. 59, no. 2, pp. 573–586, 2010.
- [6] B. Liao and S.-C. Chan
Direction-of-arrival estimation in subarrays-based linear sparse arrays with gain/phase uncertainties
IEEE Transactions on Aerospace and Electronic Systems, vol. 49, no. 4, pp. 2268–2280, 2013.
- [7] N. Hu, Z. Ye, X. Xu, and M. Bao
Doa estimation for sparse array via sparse signal reconstruction
IEEE Transactions on Aerospace and Electronic Systems, vol. 49, no. 2, pp. 760–773, 2013.
- [8] B. Liao and S.-C. Chan
Adaptive beamforming for uniform linear arrays with unknown mutual coupling
IEEE Antennas and Wireless Propagation Letters, vol. 11, pp. 464–467, 2012.
- [9] B. Liao, Z.-G. Zhang, and S.-C. Chan
Doa estimation and tracking of ulas with mutual coupling
IEEE Transactions on Aerospace and Electronic Systems, vol. 48, no. 1, pp. 891–905, 2012.
- [10] B. Liao and S. C. Chan
Direction finding with partly calibrated uniform linear arrays
IEEE Transactions on Antennas and Propagation, vol. 60, no. 2, pp. 922–929, 2011.
- [11] P. Hacker and B. Yang
Single snapshot doa estimation
Advances in Radio Science, vol. 8, pp. 251–256, 2010.
- [12] B. Liao, Z. Zhang, and S.-C. Chan
A subspace-based method for doa estimation of uniform linear array in the presence of mutual coupling
In *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*. IEEE, 2010, pp. 1879–1882.
- [13] F. Li, H. Liu, and R. J. Vaccaro
Performance analysis for doa estimation algorithms: unification, simplification, and observations
IEEE Transactions on Aerospace and Electronic Systems, vol. 29, no. 4, pp. 1170–1184, 1993.
- [14] R. J. Weber and Y. Huang
Analysis for capon and music doa estimation algorithms
In *2009 IEEE Antennas and Propagation Society International Symposium*. IEEE, 2009, pp. 1–4.
- [15] B. Liao, A. Madanayake, and P. Agathoklis
Array signal processing and systems
Multidimensional Systems and Signal Processing, vol. 29, no. 2, pp. 467–473, 2018.
- [16] S. Fortunati, R. Grasso, F. Gini, and M. S. Greco
Single snapshot doa estimation using compressed sensing
In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 2297–2301.
- [17] D. Malioutov, M. Cetin, and A. S. Willsky
A sparse signal reconstruction perspective for source localization with sensor arrays
IEEE Transactions on Signal Processing, vol. 53, no. 8, pp. 3010–3022, 2005.
- [18] M. Rice, C. Hogstrom, M. S. Afran, and M. Saquib
On sparse channel estimation in aeronautical telemetry
IEEE Transactions on Aerospace and Electronic Systems, vol. 55, no. 5, pp. 2612–2618, 2019.
- [19] B. Shim and B. Song
Multiuser detection via compressive sensing
IEEE Communications Letters, vol. 16, no. 7, pp. 972–974, 2012.
- [20] P. S. Bradley, O. L. Mangasarian, and W. N. Street
Feature selection via mathematical programming
INFORMS Journal on Computing, vol. 10, no. 2, pp. 209–217, 1998.
- [21] E. Elhamifar and R. Vidal
Sparse subspace clustering: Algorithm, theory, and applications
IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 35, no. 11, pp. 2765–2781, 2013.
- [22] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma
Robust face recognition via sparse representation
IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 31, no. 2, pp. 210–227, 2009.
- [23] R. G. Baraniuk
Compressive sensing [lecture notes]
IEEE signal processing magazine, vol. 24, no. 4, pp. 118–121, 2007.
- [24] J. A. Tropp and A. C. Gilbert
Signal recovery from random measurements via orthogonal matching pursuit
IEEE Transactions on Information Theory, vol. 53, no. 12, pp. 4655–4666, 2007.
- [25] D. Needell and J. A. Tropp
Cosamp: Iterative signal recovery from incomplete and inaccurate samples
Applied and computational harmonic analysis, vol. 26, no. 3, pp. 301–321, 2009.
- [26] J. A. Tropp
Greed is good: algorithmic results for sparse approximation
IEEE Transactions on Information Theory, vol. 50, no. 10, pp. 2231–2242, 2004.
- [27] R. Tibshirani
Regression shrinkage and selection via the lasso
Journal of the Royal Statistical Society: Series B (Methodological), vol. 58, no. 1, pp. 267–288, 1996.
- [28] S. S. Chen, D. L. Donoho, and M. A. Saunders
Atomic decomposition by basis pursuit
SIAM review, vol. 43, no. 1, pp. 129–159, 2001.
- [29] J. A. Tropp

- Just relax: convex programming methods for identifying sparse signals in noise
IEEE Transactions on Information Theory, vol. 52, no. 3, pp. 1030–1051, 2006.
- [30] E. J. Candes, M. B. Wakin, and S. P. Boyd
Enhancing sparsity by reweighted l1 minimization
Journal of Fourier analysis and applications, vol. 14, no. 5-6, pp. 877–905, 2008.
- [31] I. F. Gorodnitsky and B. D. Rao
Sparse signal reconstruction from limited data using focuss: A re-weighted minimum norm algorithm
IEEE Transactions on signal processing, vol. 45, no. 3, pp. 600–616, 1997.
- [32] R. Chartrand and W. Yin
Iteratively reweighted algorithms for compressive sensing
In *2008 IEEE international conference on acoustics, speech and signal processing*. IEEE, 2008, pp. 3869–3872.
- [33] D. Wipf and S. Nagarajan
Iterative reweighted 1 and 2 methods for finding sparse solutions
IEEE J. Sel. Top. Signal Process., vol. 4, pp. 317–329, 2010.
- [34] 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, 2004.
- [35] S. Ji, Y. Xue, and L. Carin
Bayesian compressive sensing
IEEE Transactions on Signal Processing, vol. 56, no. 6, pp. 2346–2356, 2008.
- [36] S. D. Babacan, R. Molina, and A. K. Katsaggelos
Bayesian compressive sensing using laplace priors
IEEE Transactions on Image Processing, vol. 19, no. 1, pp. 53–63, 2010.
- [37] M. D. Sacchi, T. J. Ulrych, and C. J. Walker
Interpolation and extrapolation using a high-resolution discrete fourier transform
IEEE Transactions on Signal Processing, vol. 46, no. 1, pp. 31–38, 1998.
- [38] M. E. Tipping
Sparse bayesian learning and the relevance vector machine
Journal of machine learning research, vol. 1, no. Jun, pp. 211–244, 2001.
- [39] S. D. Blunt, T. Chan, and K. Gerlach
Robust doa estimation: The reiterative superresolution (risr) algorithm
IEEE Transactions on Aerospace and Electronic Systems, vol. 47, no. 1, pp. 332–346, 2011.
- [40] P. Chen, Z. Cao, Z. Chen, and X. Wang
Off-grid doa estimation using sparse bayesian learning in mimo radar with unknown mutual coupling
IEEE Transactions on Signal Processing, vol. 67, no. 1, pp. 208–220, 2018.
- [41] S. Nannuru, A. Koochakzadeh, K. L. Gemba, P. Pal, and P. Gerstoft
Sparse bayesian learning for beamforming using sparse linear arrays
The Journal of the Acoustical Society of America, vol. 144, no. 5, pp. 2719–2729, 2018.
- [42] X. Wang, M. Huang, and G. Bi
Sparse bayesian learning for doa estimation in mimo radar with unknown nonuniform noise
In *2016 CIE International Conference on Radar (RADAR)*. IEEE, 2016, pp. 1–5.
- [43] T. A. Srikrishnan and B. D. Rao
Addressing the noise variance problem in sparse bayesian learning
In *2018 52nd Asilomar Conference on Signals, Systems, and Computers*. IEEE, 2018, pp. 1974–1979.
- [44] D. P. Wipf and B. D. Rao
An empirical bayesian strategy for solving the simultaneous sparse approximation problem
IEEE Transactions on Signal Processing, vol. 55, no. 7, pp. 3704–3716, 2007.
- [45] D. P. Wipf
Bayesian methods for finding sparse representations
2006.
- [46] Z. Zhang and B. D. Rao
Sparse signal recovery with temporally correlated source vectors using sparse bayesian learning
IEEE Journal of Selected Topics in Signal Processing, vol. 5, no. 5, pp. 912–926, 2011.
- [47] S. Nannuru, K. L. Gemba, P. Gerstoft, W. S. Hodgkiss, and C. F. Mecklenbräuker
Multi-frequency sparse bayesian learning with uncertainty models
arXiv preprint arXiv:1704.00436, 2017.
- [48] S. Nannuru, P. Gerstoft, and K. L. Gemba
Sparse bayesian learning with uncertain sensing matrix
In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 3964–3968.
- [49] B. Liao, Z. Zhou, and S. Zhang
An map method for closed-loop channel training in massive mimo systems
IEEE Transactions on Vehicular Technology, vol. 71, no. 5, pp. 5534–5539, 2022.
- [50] D. P. Wipf, B. D. Rao, and S. Nagarajan
Latent variable bayesian models for promoting sparsity
IEEE Transactions on Information Theory, vol. 57, no. 9, pp. 6236–6255, 2011.
- [51] A. P. Dempster, N. M. Laird, and D. B. Rubin
Maximum likelihood from incomplete data via the em algorithm
Journal of the Royal Statistical Society: Series B (Methodological), vol. 39, no. 1, pp. 1–22, 1977.
- [52] D. G. Tzikas, A. C. Likas, and N. P. Galatsanos
The variational approximation for bayesian inference
IEEE Signal Processing Magazine, vol. 25, no. 6, pp. 131–146, 2008.
- [53] R. Giri and B. Rao
Type i and type ii bayesian methods for sparse signal recovery using scale mixtures
IEEE Transactions on Signal Processing, vol. 64, no. 13, pp. 3418–3428, 2016.
- [54] C. M. Bishop and N. M. Nasrabadi
Pattern recognition and machine learning. Springer, 2006, vol. 4, no. 4.
- [55] M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul
An introduction to variational methods for graphical models
Machine learning, vol. 37, no. 2, pp. 183–233, 1999.
- [56] M. E. Tipping
The relevance vector machine
In *Advances in neural information processing systems*, 2000, pp. 652–658.
- [57] Y. Shen, J. Fang, and H. Li
Exact reconstruction analysis of log-sum minimization for compressed sensing
IEEE Signal Processing Letters, vol. 20, no. 12, pp. 1223–1226, 2013.
- [58] L. Stankovic, S. Stankovic, and M. Amin
Missing samples analysis in signals for applications to l-estimation and compressive sensing
Signal Processing, vol. 94, pp. 401–408, 01 2014.
- [59] A. Palffy, J. F. Kooij, and D. M. Gavrilu
Occlusion aware sensor fusion for early crossing pedestrian detection
In *2019 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2019, pp. 1768–1774.
- [60] S. M. Patole, M. Torlak, D. Wang, and M. Ali
Automotive radars: A review of signal processing techniques

- IEEE Signal Processing Magazine*, vol. 34, no. 2, pp. 22–35, 2017.
- [61] M. Soumekh
Array imaging with beam-steered data
IEEE Transactions on Image Processing, vol. 1, no. 3, pp. 379–390, 1992.
- [62] S. Patole and M. Torlak
Two dimensional array imaging with beam steered data
IEEE transactions on image processing, vol. 22, no. 12, pp. 5181–5189, 2013.
- [63] F. Chatzigeorgiadis
Development of code for a physical optics radar cross section prediction and analysis application
NAVAL POSTGRADUATE SCHOOL MONTEREY CA, Tech. Rep., 2004.