Bayesian Sparse Blind Deconvolution Using MCMC Methods Based on Normal-Inverse-Gamma Prior

Burak C. Civek and Emre Ertin

Abstract—Bayesian estimation methods for sparse blind deconvolution problems conventionally employ Bernoulli-Gaussian (BG) prior for modeling sparse sequences and utilize Markov Chain Monte Carlo (MCMC) methods for the estimation of unknowns. However, the discrete nature of the BG model creates computational bottlenecks, preventing efficient exploration of the probability space even with the recently proposed enhanced sampler schemes. To address this issue, we propose an alternative MCMC method by modeling the sparse sequences using the Normal-Inverse-Gamma (NIG) prior. We derive effective Gibbs samplers for this prior and illustrate that the computational burden associated with the BG model can be eliminated by transferring the problem into a completely continuous-valued framework. In addition to sparsity, we also incorporate time and frequency domain constraints on the convolving sequences. We demonstrate the effectiveness of the proposed methods via extensive simulations and characterize computational gains relative to the existing methods that utilize BG modeling.

Index Terms—Blind deconvolution, sparse recovery, Bayesian estimation, Markov Chain Monte Carlo, Gibbs sampler.

I. INTRODUCTION

THE problem of reconstructing an unknown signal from an observation, where the observation is modeled as the filtered version of the unknown signal with a given distortion filter, is known as the deconvolution problem. When the impulse response of the distortion filter is not known and to be estimated along with the unknown signal, it evolves into the Blind Deconvolution (BD) problem. This problem setting emerges in a variety of real-life applications including, but not limited to, seismic exploration [1], [2], image deblurring [3], [4], digital communication [5], [6], and biomedical signal reconstruction [7], [8]. The class of BD problems is inherently ill-posed and suffers from the identifiability issues without the prior knowledge about the convolving signals [9]. This is due to the fact that many distinct signal pairs can yield the same observation. To overcome this issue, several different assumptions are made on the signal

pairs in an attempt to constrain the solution space. Typical examples of these include restricted support in frequency or time domain [7], [8], availability of a sparse representation [10]–[12], or existence of a generative subspace [6], [13].

Solution of BD problems has been extensively studied under both deterministic and probabilistic frameworks [14]. Deterministic approaches typically construct a constrained optimization problem, where the goal is to minimize an objective function corresponding to a likelihood term that measures the quality of fit. The constraints on the variables are imposed either explicitly or by augmenting the objective function. Because a closed form solution is usually not available, gradient-descent search algorithms are employed to converge to a local minimum that minimizes the objective function. However, due to the structure of the constraints, the constructed optimization problem usually has a non-convex formulation, and therefore, the successful recovery relies on finding a good initial estimate [11]. Even though there exist convex formulations, which eliminate the effect of initialization, their ability to impose a variety of constraints simultaneously, e.g., frequency and time domain constraints at the same time, is limited [8], [10], [12]. Notably, the lifting approach proposed in [13], and its extension to sparse sequences [15], enable imposing time/frequency constraints jointly by assuming the existence of a generative subspace. However, the number of available measurements is limited when the impulse response of the distortion filter has a bandlimited structure, which might prevent successful recovery with lifting approaches.

Probabilistic approaches, on the other hand, provide effective alternatives by modeling the unknown quantities as random variables and producing estimates based on the posterior distribution. Appropriate prior distributions are assigned to the unknown variables in order to increase the posterior value of points that satisfy the constraints in the probability space. Similar to deterministic models, analytic solution is not tractable in many cases, where the solution is obtained through numerical iterative methods such as Markov Chain Monte Carlo (MCMC) [16]. Different from gradient-descent searches, which seek for a local minimizer point, MCMC methods sample from the posterior distribution, exploring the space to perform density estimation, which mitigates the effect of initialization [17]. This enables probabilistic schemes to incorporate more complex constraints simultaneously without suffering from the highly non-convex structure of the posteriors shaped by these constraints.

In this paper, we adopt the probabilistic framework and present computationally efficient Bayesian methods for the solution of the regularized BD problem, in which the desired signal

is a sparse sequence and the unknown impulse response of the distortion filter is time and/or bandlimited. This problem setting is of great importance due to its applicability to a wide range of scenarios. In many cases, even if the desired signal itself is not sparse, it is possible to find a sparse representation in a suitable transform domain. Moreover, physical realizations of real-world systems are well modeled by exactly or approximately bandlimited system responses with finite impulse responses.

Bayesian BD methods conventionally employ a Bernoulli-Gaussian (BG) prior for modeling the sparse sequences [18], [19]. According to the BG model, the sparse sequence is represented by a Bernoulli distributed binary latent sequence, indicating the nonzero positions, and an amplitude sequence, representing the amplitudes corresponding to those positions under the Gaussian law. The posterior distribution constructed in this manner has a discrete structure, prohibiting efficient analytical solutions and leading to iterative approaches for making inference on the posterior. Bayesian methods typically use the MCMC simulations, whose potential in the sparse BD literature was presented by the pioneering work of Cheng *et al.* [1]. The Gibbs sampler in particular is a powerful iterative tool for Bayesian inference problems, even when the number of unknowns is quite high [20]. Due to its simplicity, it has practical advantages over more sophisticated and efficient sampling methods, such as the Hamiltonian Monte Carlo (HMC) [21], which also has applications to sparse sampling from non-differentiable distributions [22]. Unlike the HMC methods, for which the sampling performance strictly depends on fine-tuning the hyper-parameters, which is quite complicated for high-dimensional spaces, Gibbs sampling does not require any fine-tuning and still provides a significantly fast mixing rate, especially if the sampled variables are independent [20]. However, the mixing performance degrades considerably when there are strong dependencies between variables [23], [24].

It has been shown that the original Gibbs sampler constructed based on BG prior for sparsity causes implicit dependencies between the consecutive sampling steps of the latent indicator sequence, which prohibits the sampler to explore the probability space efficiently, causing it to get stuck on a local optimum for a long time [23]. In order to eliminate this deficiency, more efficient sampling methods were proposed in [23] and [25], accounting for the statistical dependence between the neighboring variables in the indicator sequence. An alternative idea, which was first introduced in [26], and revisited in [24] within the Bayesian sparse blind deconvolution framework, is to use the blocked Gibbs sampler scheme that enables sampling adjacent variables jointly. Despite the considerable improvement in the convergence rate, their application is usually limited to very short blocks due to exponentially increasing computational complexity with the block length. Another attempt, also presented in [24], to improve the mixing rate was to integrate the partially collapsed Gibbs (PCG) sampler, a generalization of the block Gibbs samplers, into the BG model. PCG samplers explore the probability space more effectively by means of marginalization and trimming operations [27]. However, although the PCG sampling scheme outperforms the blocked sampler schemes in terms of the number of iterations needed for convergence, iterations

are considerably more costly [24]. As an alternative to the BG model, the Bernoulli-Laplacian (BL) prior is also widely studied in the literature in an attempt to produce sparser solutions by the inclusion of the Laplacian term [28], [29]. However, both the BG and BL models share the same computational issues described above, which are mainly due to the discrete nature of the incorporated Bernoulli prior creating a computational bottleneck even for the highly efficient sampling schemes.

In order to address this deficiency, in this work, we present alternative MCMC methods utilizing the Normal-Inverse-Gamma (NIG) prior, which is a well-known continuous surrogate of the BG prior for modeling sparsity. The use of continuous NIG prior is well-established in the Bayesian sparse learning literature [30] and emerges in a variety of applications including, but not limited to, the scale mixture of Gaussians [31], where the mixing distribution is an Inverse-Gamma, and the sparse Bayesian linear regression [32], [33]. In our problem setting, unlike the BG model, the NIG prior models the sparse sequence as a zero-mean multivariate Gaussian distribution with an unknown diagonal covariance matrix, where the individual variances marginally follow an Inverse-Gamma (IG) distribution. Therefore, the marginal distribution for the sparse sequence corresponds to a multivariate t-distribution, whose heavy tails encourage sparsity [34]–[36]. With this setting, the problem is transformed into a completely continuous valued domain, i.e., all unknowns including the latent variables are continuous valued.

Our contributions in this work are listed as follows: i) To the best of our knowledge, we present the first application of NIG prior within the MCMC framework for the Bayesian sparse BD problem with time/frequency domain constraints. ii) We also present a PCG sampler variant to address the inefficiencies of the straightforward implementation of the classical Gibbs sampling approach. iii) We designed the samplers in a hierarchical manner to infer all other model parameters, including the noise variance and the hyper-parameters of the selected priors, simultaneously and solely from the measurements. Hence, the samplers require no fine-tuning. iv) We present an in depth numerical study to assess the performance of the proposed samplers and demonstrate that they achieve state-of-the-art performance of the previously proposed blocked [24] Gibbs samplers with BG prior at a fraction of their computational cost.

The paper is organized as follows. We first introduce the problem setting and define the prior distributions associated with each variable in Section II. Then, in Section III, we present the proposed MCMC methods for the solution of sparse BD problems, followed by their validation through numerical experiments in Section IV. We finalize the discussion with the concluding remarks in Section V.

II. PROBLEM STATEMENT

We consider a real-valued time-domain observation sequence y_n of length N, which is modeled by linear convolution of two finite sequences, a relatively shorter pulse sequence h_n of length $T \ll N$, modeling the impulse response of the distortion filter, and a sparse sequence x_n of length $K < N$. The output of the

convolution is then corrupted by additive noise v_n, yielding the following measurement model

$$y_n = \sum_{k=0}^{T-1} h_k x_{n-k} + v_n \text{ for } n = 0, \ldots, N-1, \quad (1)$$

where $x_n = 0$ for any $n \notin \{0, \ldots, K-1\}$. Here, we consider the scenario where both the pulse shape h_n and the sparse sequence x_n are unknown and to be estimated from the observation y_n. This problem is well-known as the blind deconvolution problem and usually ill-posed if there is no prior information about the pair of convolving sequences. Here, we focus on the setting where the problem is regularized by sparsity and time-frequency domain constraints, which enables the successful recovery of both sequences.

For notational convenience, we define the vector valued variables $y, v \in \mathbb{R}^N$, $x \in \mathbb{R}^K$, and $h \in \mathbb{R}^T$, and rewrite (1) as

$$y = Hx + v, \quad (2)$$

where $H \in \mathbb{R}^{N \times K}$ is the Toeplitz matrix of h. Here, the first column and row of H are given as $[h_0, h_1, \ldots, h_{T-1}, 0, \ldots, 0]^T$ and $[h_0, 0, \ldots, 0]$ respectively. Defining $x \in \mathbb{R}^{N \times T}$ as the Toeplitz matrix having $[x_0, x_1, \ldots, x_{K-1}, 0, \ldots, 0]^T$ as its first column and $[x_0, 0, \ldots, 0]$ as its first row, we can alternatively rewrite (2) as $y = xh + v$. We will make use of both of these formulations in the rest of the paper.

A. Prior Distributions

Following the Bayesian framework, the unknowns are modeled as random variables with specific prior distributions. This section provides a complete description of the prior distributions assigned to each variable. We begin with review of the BG distribution, which is the conventional prior used in Bayesian settings for sparse sequences. We then introduce the NIG model, which is a *soft* alternative to the BG model promoting sparsity of x and will constitute the basis of our proposed estimator in the next section.

1) Bernoulli-Gaussian Prior for Sparsity: The BG model introduces a latent binary sequence $s = [s_0, s_1, \ldots, s_{K-1}]^T$ with $s_n \in \{0, 1\}$ and defines the conditional distribution of x_n given s_n as

$$p(x_n|s_n) = \begin{cases} \delta(x_n) & \text{if } s_n = 0 \\ \mathcal{N}(x_n; 0, \sigma_x^2) & \text{if } s_n = 1 \end{cases}, \quad (3)$$

where $\delta(\cdot)$ is the Dirac delta function and $\mathcal{N}(\cdot; \mu, \sigma^2)$ denotes the Gaussian distribution with mean μ and variance σ^2. Assuming s_n are independent and identically distributed (i.i.d.) according to a Bernoulli distribution with parameter $\pi_0 = P(s_n = 0)$, the prior for s takes the form

$$p(s) = \prod_{n=0}^{K-1} p(s_n) = \binom{K}{|\mathcal{K}_1|} \pi_0^{|\mathcal{K}_1|} (1-\pi_0)^{K-|\mathcal{K}_1|}, \quad (4)$$

where $\mathcal{K}_1$ denotes the set of indices giving the locations of 1's and $|\mathcal{K}_1|$ represents the cardinality of $\mathcal{K}_1$, i.e., $|\mathcal{K}_1| = \sum s_n$. Assuming different pairs of (x_n, s_n) are statistically independent,

the joint distribution of x and s becomes

$$p(x, s) = p(s) \prod_{n \in \mathcal{K}_1} \mathcal{N}(x_n; 0, \sigma_x^2) \prod_{n \in \mathcal{K}_0} \delta(x_n), \quad (5)$$

where $\mathcal{K}_0$ is the complement of $\mathcal{K}_1$ in $\{0, 1, \ldots, K-1\}$. The parameter π_0 reflects our *a priori* knowledge about the expected rate of 1's in s, limiting the total number of nonzero entries in x, and hence, leading to a sparse sequence.

2) The Normal-Inverse-Gamma Prior for Sparsity: Instead of introducing a latent binary sequence, which forms a discrete probability space, the sparsity can also be imposed by a diagonal covariance matrix $\Sigma_x = \text{diag}(\sigma_x^2)$, where $\sigma_x^2 = [\sigma_{x_0}^2, \sigma_{x_1}^2, \ldots, \sigma_{x_{K-1}}^2]^T$. Here, σ_x^2 consists of continuous valued unknown variances of each element in x. Under a zero-mean multivariate Gaussian law with the covariance matrix Σ_x, the conditional prior distribution of x given σ_x^2 becomes

$$p(x|\sigma_x^2) = \mathcal{N}(x; 0, \Sigma_x), \quad (6)$$

where $\mathcal{N}(\cdot; \mu, \Sigma)$ represents the multivariate Gaussian distribution with mean μ and covariance Σ. The idea is that an individual element x_n can be made arbitrarily small by setting a sufficiently low variance $\sigma_{x_n}^2$. Therefore, the unknown variance vector σ_x^2 is also assumed to be a random sequence and to be estimated along with h and x.

We assign i.i.d. IG[1] prior on σ_x^2 with shape and scale parameters α_x and β_x

$$p(\sigma_x^2|\alpha_x, \beta_x) = \prod_{n=0}^{K-1} \mathcal{IG}(\sigma_{x_n}^2; \alpha_x, \beta_x). \quad (7)$$

The IG distribution is conjugate prior for the unknown variance of the Gaussian distribution, which enables analytical calculation of the posterior. Moreover, note that the marginal distribution of any element x_n becomes

$$p(x_n) = \int p(x_n|\sigma_{x_n}^2) p(\sigma_{x_n}^2) d\sigma_{x_n}^2$$

$$= \frac{\beta_x^{\alpha_x}}{\sqrt{2\pi} \Gamma(\alpha_x)} \frac{\Gamma(\alpha_x + 0.5)}{(0.5 x_n^2 + \beta_x)^{\alpha_x + 0.5}}, \quad (8)$$

which corresponds to a generalized Student's t-distribution with degree of freedom $2\alpha_x$ and scale β_x/α_x. As shown in Fig. 1, appropriate selection of parameters (α_x, β_x) leads to a family of distributions that are highly concentrated around zero with significantly heavier tails compared to standard Gaussian distribution, justifying the use of NIG model for sparse sequences as an alternative to BG model. However, as opposed to the simple interpretation of π_0 within the BG model, the effect of parameters α_x and β_x is complicated, preventing the availability of prior estimates beforehand. Therefore, we construct a hierarchical Bayesian model and learn the prior distribution parameters α_x and β_x from the measurement as well. Assuming no prior information on α_x and β_x, we assign Jeffreys prior, given by $p(\alpha_x) \propto 1/\alpha_x$ and $p(\beta_x) \propto 1/\beta_x$, which forms an improper prior exhibiting non-informative structure. Here, $\propto$

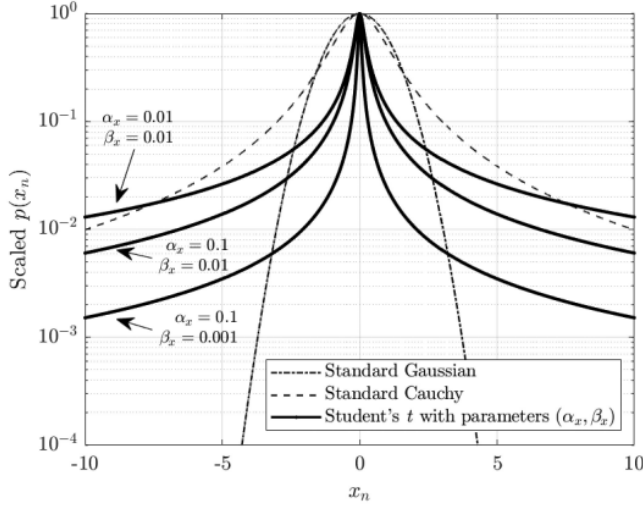[1] Please see the supplement for the explicit definition of IG distribution.

Fig. 1. Marginal prior distribution of x_n in logarithmic scale for different parameter values (α_x, β_x). Standard Gaussian and Cauchy distributions are included for comparison. All densities are scaled such that the maximum is 1.

denotes the proportionality. We should note that the effect of prior distributions $p(\alpha_x)$ and $p(\beta_x)$ on the posterior will indeed be dominated by $p(x|\sigma_x^2)$ and $p(\sigma_x^2|\alpha_x, \beta_x)$ as the length of the sequence K increases. Hence, the constructed hierarchical model is not sensitive to the selection of prior distributions for the variables α_x and β_x.

3) Prior for Short Pulse Sequence: Time and frequency domain constraints for pulse sequences are widely used in blind deconvolution framework to regularize the problem [7], [8], [37]. One common practice is to assume that the pulse sequence belongs to a known subspace, i.e., $h = A\gamma$, where $A \in \mathbb{R}^{T \times L}$ represents a lower dimensional subspace with $L \leq T$ and $\gamma \in \mathbb{R}^L$ represents the unknown orientation of h in the subspace, which is to be estimated. The duration of h in time domain is explicitly enforced by the dimension of A, and the frequency domain restrictions can be applied by constructing A using the first L sequence of either Discrete Prolate Spheroidal (DPS) Sequences or Hermite Functions [38]. Whenever there is no specific frequency domain restriction, A can be set as identity, i.e., $A = I$. We should note that due to the scaling ambiguity inherent in BD problems, the scale of the pulse sequence h must be restricted. This can be achieved by assigning an appropriate prior on γ. Therefore, we assign a zero-mean i.i.d. Gaussian distribution with variance σ_γ^2, i.e.,

$$p(\gamma) = \mathcal{N}(\gamma; 0, \sigma_\gamma^2 I). \tag{9}$$

Here, the variance σ_γ^2 is a fixed hyperparameter to avoid scaling ambiguity.

4) Prior for Noise Variance: We assume that measurements are corrupted by additive white Gaussian noise with unknown variance σ_v^2. Assuming no prior information, similar to the other parameters, we assign Jeffreys prior, i.e., $p(\sigma_v^2) \propto 1/\sigma_v^2$.

B. Estimation Problem

The estimation problem consists of estimating the actual variables of interest x and γ, along with the noise variance σ_v^2, the latent variables σ_x^2, and the corresponding prior distribution parameters α_x, β_x from the given measurement y. Given the prior distributions for all variables, the posterior distribution follows

$$p(\theta|y) \propto p(y|\theta)p(\theta), \tag{10}$$

where we set $\theta = [x, \sigma_x^2, \gamma, \sigma_v^2, \alpha_x, \beta_x]$ for more compact notation. Assuming all variables are statistically independent, the prior distribution $p(\theta)$ is given by

$$p(\theta) = p(x|\sigma_x^2)p(\sigma_x^2|\alpha_x, \beta_x)p(\alpha_x)p(\beta_x)p(\gamma)p(\sigma_v^2), \tag{11}$$

where the expressions for the right hand side are given in Section II-A. Assumed Gaussian noise model yields the following likelihood term

$$p(y|\theta) \propto \left(\frac{1}{\sigma_v^2}\right)^{N/2} \exp\left(-\frac{\|y - XA\gamma\|^2}{2\sigma_v^2}\right), \tag{12}$$

where $\| \cdot \|$ denotes ℓ_2 norm of a vector. Note that the likelihood term is only a function of x, γ and σ_v^2.

For estimation of the variables, we consider the minimum mean-square-error (MMSE) estimator, i.e.,

$$\theta^* = E[\theta|y], \tag{13}$$

which is equivalent to the expected value of the posterior distribution given in (10). However, explicit calculation of (13) is not possible since it requires analytically intractable integrations. This leads us to an approximate solution, which can be obtained using MCMC simulations. MCMC methods are widely used in Bayesian inference problems, including blind deconvolution literature, when the posterior distribution is too complicated to obtain exact analytical solutions [16], [17]. The first step of MCMC methods consists of generating a set of, say, J random samples, denoted by $\{\theta^{(i)}\}_{i=1}^J$, with $\theta^{(i)} = [x^{(i)}, \sigma_x^{2(i)}, \gamma^{(i)}, \sigma_v^{2(i)}, \alpha_x^{(i)}, \beta_x^{(i)}]$ being the ith sample, from the posterior distribution using an appropriate sampler. Once the sampling process is completed, the actual solution of (13) is approximated by the sample mean, i.e.,

$$\theta^* \simeq \frac{1}{J - J'} \sum_{i=J'+1}^{J} \theta^{(i)}, \tag{14}$$

where the first J' samples are discarded as part of the burn-in process. Here, the crucial part is to have an effective sampler, which can converge to the true target distribution quickly. To this end, in the next section, we construct the proposed valid sampling schemes to be used within MCMC simulations.

III. PROPOSED SAMPLERS FOR SPARSE BLIND DECONVOLUTION

Samplers are essential components of MCMC methods for complex target distributions. They construct Markov Chains whose stationary distributions converge to the target distribution (which corresponds to the posterior distribution (10) in our case)

in the long run [39]. The effectiveness of the sampler is directly associated with the mixing rate, which represents how fast the stationary distribution is achieved. In this work, we employ both Gibbs and PCG sampling schemes with improved mixing rates.

In this section, we first briefly review the idea of classical Gibbs sampling for multivariate distributions, followed by the construction of our proposed classical Gibbs sampler, which utilizes the alternative NIG model for sparsity. Then, we propose a PCG based sampler to improve upon the classical Gibbs sampler to obtain a faster mixing rate with a slightly increased computational complexity.

A. Gibbs Sampler

Let $\pi(\boldsymbol{\theta})$ denote the target distribution that we want to sample from, with $\boldsymbol{\theta}$ being a vector valued random variable of arbitrary length, say M, i.e., $\boldsymbol{\theta} = [\theta_0, \theta_1, \ldots, \theta_{M-1}]^T$. When direct sampling from $\pi(\boldsymbol{\theta})$ is not feasible, the idea of Gibbs sampling suggests that we can sample each of the scalar variables, θ_m, in turn from their conditional distributions with all other variables are fixed at their current values. Hence, in the ith iteration, $\theta_m^{(i)}$ is obtained by sampling from $p(\theta_m^{(i)}|\theta_{m^-}^{(i)}, \theta_{m^+}^{(i-1)})$, where m^- and m^+ represents the indices $\{1, \ldots, m-1\}$ and $\{m+1, \ldots, M\}$ respectively. One iteration of Gibbs sampling is completed once every single variable is updated. All variables are initialized, usually by sampling from their prior distributions, before the first iteration, which has a strong effect on the first few realizations. In order to eliminate this effect, a burn-in process is incorporated, where the realizations generated until convergence to the target distribution are discarded.

It is also useful to note that this process is not restricted to sampling a single scalar variable at a time. An extension of the Gibbs sampler, called blocked Gibbs sampler, allows sampling blocks of variables at one step through their joint distribution conditioned on others. It helps to achieve considerably improved convergence rates compared to sampling a scalar valued variable at a time by reducing the autocorrelation between the successive samples. This is especially useful when applied to variables with strong dependencies.

B. The Proposed Classical Gibbs Sampler

We begin with constructing the classical Gibbs sampler based on the selected prior distributions. Then, in the next section, we propose an alternative sampling scheme that introduces additional intermediate steps to enhance the mixing rate. One iteration of our classical Gibbs sampler is described in Table I. In each step, we sample a variable from its full conditional posterior distribution. The reason for dropping some of the variables from the conditions is not because of marginalization, but due to conditional independence. For instance, in Step 1, we have $p(\alpha_x|\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{\sigma}_x^2, \boldsymbol{\gamma}, \sigma_v^2, \beta_x) = p(\alpha_x|\boldsymbol{\sigma}_x^2, \beta_x)$. Analogous situations apply to the other steps.

This sampling scheme can be viewed as a blocked Gibbs sampler, since the variables being sampled in steps 3, 4, and 5 are vector valued. Nevertheless, each variable is sampled exactly once using the corresponding posterior distribution conditioned on the current values of all other variables, hence, it is a valid

TABLE I
PROPOSED CLASSICAL GIBBS SAMPLER

Step 1. Sample α_x from $p(\alpha_x|\boldsymbol{\sigma}_x^2, \beta_x)$
Step 2. Sample β_x from $p(\beta_x|\boldsymbol{\sigma}_x^2, \alpha_x)$
Step 3. Sample $\boldsymbol{\sigma}_x^2$ from $p(\boldsymbol{\sigma}_x^2|\boldsymbol{x}, \alpha_x, \beta_x)$
Step 4. Sample $\boldsymbol{x}$ from $p(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\sigma}_x^2, \boldsymbol{\gamma}, \sigma_v^2)$
Step 5. Sample $\boldsymbol{\gamma}$ from $p(\boldsymbol{\gamma}|\boldsymbol{y}, \boldsymbol{x}, \sigma_v^2)$
Step 6. Sample σ_v^2 from $p(\sigma_v^2|\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{\gamma})$

Gibbs sampler. Due to our selection of conjugate priors, each conditional posterior distribution is analytically tractable. We now present the closed-form expressions for the sampling distributions in each step of Table I. The derivations are provided in the supplemental material.

1) Sampling Distributions for Step 1 and 2: The prior distribution parameters α_x and β_x for the latent variable σ_x^2 are sampled respectively from

$$p(\alpha_x|\boldsymbol{\sigma}_x^2, \beta_x) \propto \frac{\beta^{K\alpha_x}}{\Gamma(\alpha_x)^K}\left(\prod_{n=0}^{K-1}\frac{1}{\sigma_{x_n}^2}\right)^{\alpha_x+1}p(\alpha_x) \quad (15)$$

and

$$p(\beta_x|\boldsymbol{\sigma}_x^2, \alpha_x) = \mathcal{G}(\beta_x; \tilde{\alpha}_{\beta_x}, \tilde{\beta}_{\beta_x}), \quad (16)$$

where $\tilde{\alpha}_{\beta_x} = K\alpha_x$ and $\tilde{\beta}_{\beta_x} = \sum_{n=0}^{K-1} 1/\sigma_{x_n}^2$. While sampling β_x is straightforward, it is not for α_x due to not well-known form of its sampling distribution. Nevertheless, univariate form of (15) allows us to draw samples efficiently by employing different sampling approaches, such as Metropolis-Hastings or Slice sampling [40].

2) Sampling Distribution for Step 3: The posterior distribution of σ_x^2 conditioned on $\boldsymbol{x}$, α_x, and β_x is given by

$$p(\boldsymbol{\sigma}_x^2|\boldsymbol{x}, \alpha_x, \beta_x) = \prod_{n=0}^{K-1}\mathcal{IG}(\sigma_{x_n}^2; \tilde{\alpha}_x, \tilde{\beta}_{x_n}) \quad (17)$$

with common shape parameter $\tilde{\alpha}_x = \alpha_x + 1/2$ and individual scale parameters $\tilde{\beta}_{x_n} = x_n^2/2 + \beta_x$. Therefore, sampling $\boldsymbol{\sigma}_x^2$ can be achieved by independently sampling its elements from univariate IG distributions.

3) Sampling Distribution for Step 4: The sampling distribution for the sparse sequence $\boldsymbol{x}$ takes the form of a multivariate Gaussian distribution

$$p(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\sigma}_x^2, \boldsymbol{\gamma}, \sigma_v^2) = \mathcal{N}(\boldsymbol{x}; \tilde{\boldsymbol{\mu}}_x, \tilde{\boldsymbol{\Sigma}}_x) \quad (18)$$

with the posterior mean $\tilde{\boldsymbol{\mu}}_x$ and covariance $\tilde{\boldsymbol{\Sigma}}_x$ given by

$$\tilde{\boldsymbol{\mu}}_x = \frac{1}{\sigma_v^2}\tilde{\boldsymbol{\Sigma}}_x \boldsymbol{H}^T\boldsymbol{y}, \qquad \tilde{\boldsymbol{\Sigma}}_x = \left(\frac{1}{\sigma_v^2}\boldsymbol{H}^T\boldsymbol{H} + \boldsymbol{\Sigma}_x^{-1}\right)^{-1}. \quad (19)$$

Here, in order to avoid the explicit matrix inversion required for computing the posterior covariance, we adopt the Cholesky-based sampling procedures as described in [24]. In our application domain, $\boldsymbol{x}$ is typically a moderate length sequence. This allows us to sample the whole sequence at once, while still achieving a significant reduction on the cost per iteration
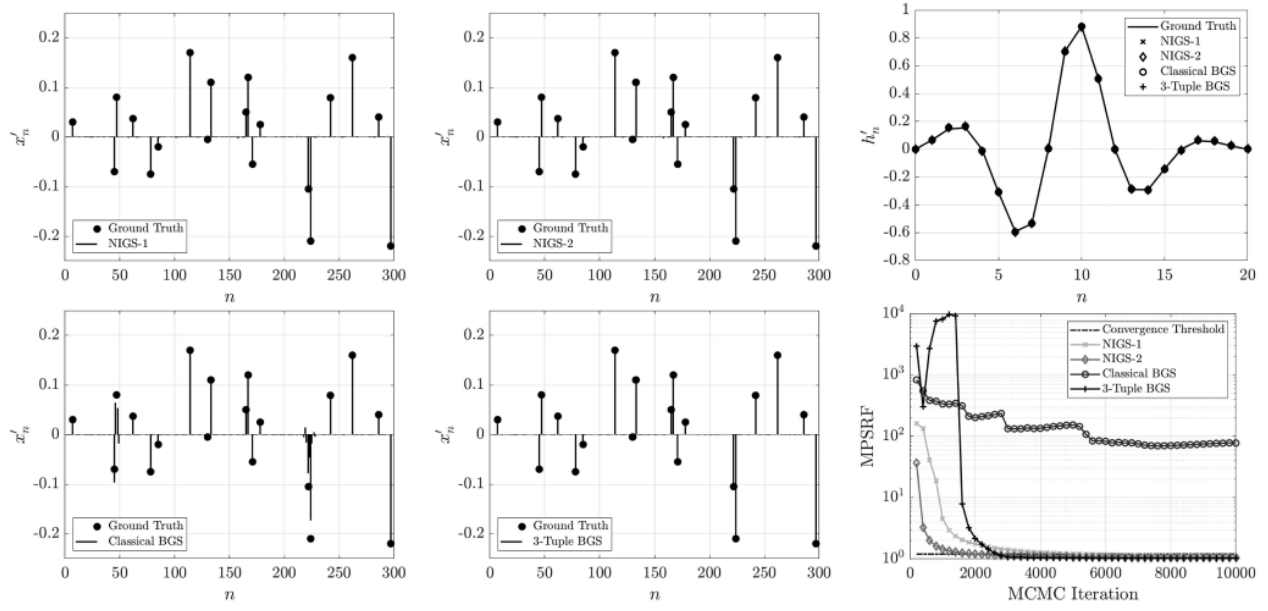
Fig. 2.   Recovery results for the sparse sequence (left and middle columns), pulse sequence (upper right), and convergence rates (lower right).
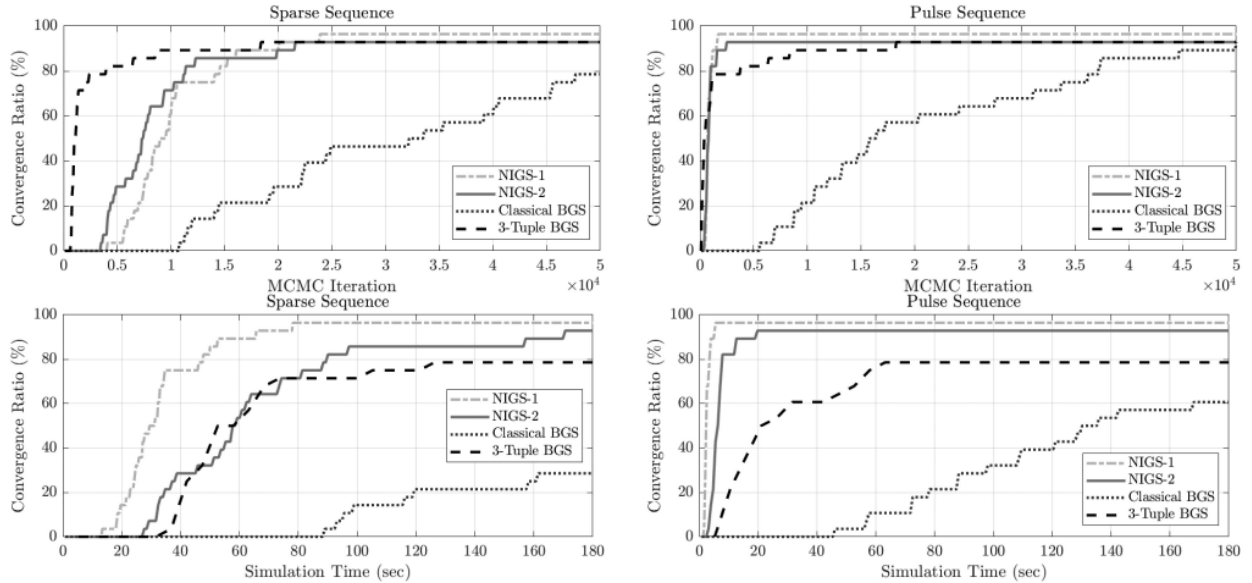

Fig. 3.   Ratio of the number of converged simulations to the total number of simulations as a function of MCMC iteration (top) and CPU time (bottom) for the sparse sequence x and the pulse sequence h.

compared to standard BG models as demonstrated in Fig. 4. However, we should note that, for applications with very large K, such as blind image deconvolution, sampling the whole sequence x at once might be infeasible both in terms of computational and storage costs. This problem has been extensively studied in several studies including, [41], [42] and more recently in [43] and [44]. As an alternative to these approaches, one can also reduce the total computational load of Step 4 by considering a trade-off between the cost per iteration and the mixing rate of the sampler and apply a hierarchical Gibbs sampling procedure

to sample shorter blocks of x in turn conditioned on each other. Since smaller block lengths will reduce the mixing rate by increasing the statistical dependence between the consecutive sub-sampling steps, it must be selected as large as possible within the limits of computational resources.

4) Sampling Distribution for Step 5: Similar to that of x, the sampling distribution for γ is also a multivariate Gaussian distribution

$$p(\gamma|y, x, \sigma_v^2) = \mathcal{N}(\gamma; \tilde{\mu}_\gamma, \tilde{\Sigma}_\gamma), \qquad (20)$$
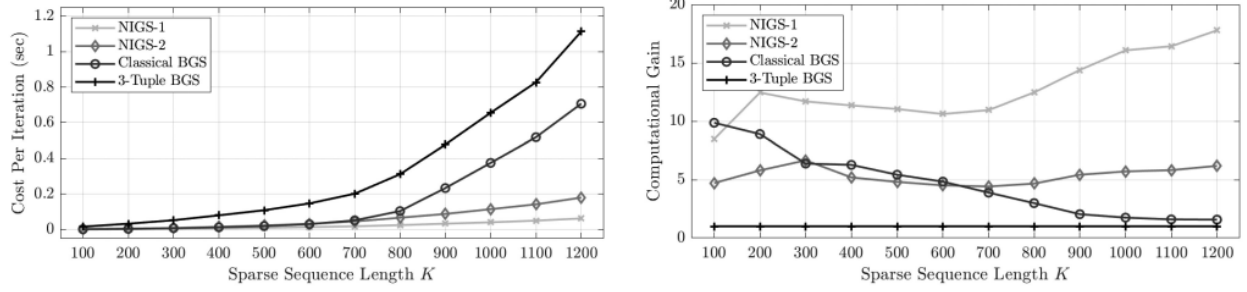
Fig. 4. Comparison of cost per iteration in seconds for sparse sequences with varying lengths (left). Computational gain of the samplers relative to 3-Tuple BGS as a function of sparse sequence length K (right).

where the posterior mean and covariance are now given by

$$\tilde{\mu}_\gamma = \frac{1}{\sigma_v^2}\tilde{\Sigma}_\gamma B^T y, \qquad \tilde{\Sigma}_\gamma = \left(\frac{1}{\sigma_v^2}B^T B + \frac{1}{\sigma_\gamma^2}I\right)^{-1}, \quad (21)$$

with $B = xA$. Unlike the sparse sequence x, the length of γ is usually very small for bandlimited sequences, enabling direct sampling of γ with the Cholesky-based sampling schemes.

5) Sampling Distribution for Step 6: The sampling distribution for the noise variance σ_v^2 takes the form of IG distribution given by

$$p(\sigma_v^2|y, x, \gamma) = \mathcal{IG}(\sigma_v^2; \tilde{\alpha}_v, \tilde{\beta}_v) \qquad (22)$$

with parameters $\tilde{\alpha}_v = N/2$ and $\tilde{\beta}_v = \|y - xA\gamma\|^2/2$.

The use of alternative soft NIG prior for sparsity, instead of the BG model, allows us to introduce Step 3, where the latent variable σ_x^2 is sampled conditioned on the current value of x. The equivalent step for the BG model would be sampling the binary sequence s from $p(s|x) = p(x|s)p(s)$, which would, however, not possible due to the deterministic dependence between s and x. Therefore, s and x are usually sampled jointly. However, since s is a binary sequence of length K, sampling s as a whole at one step, which would require 2^K different probability mass point calculation, is not feasible. This enforces BG models to sample a single tuple (s_n, x_n) at a time conditioned on the others. Since neighboring variables usually have strong dependence, sampling them conditioned on each other causes the sampler to be stuck on a local optimum for a long time. There is an extension proposed in [24] that samples M-tuples at a time to escape from local optimums faster, but it is limited to very short blocks due to exponentially increasing computational load. Therefore, the discrete nature of the BG model creates essential computational burdens. By employing the alternative NIG model, the problem is transformed into a fully continuous valued framework, which not only helps to eliminate the exponential computational complexity but also provides an easier transition between different local optimums.

C. The Proposed Partially Collapsed Gibbs Sampler

There exist two main points worth addressing in an attempt to improve the convergence rate of the proposed classical Gibbs sampler. Firstly, note that Step 3 samples σ_x^2 purely conditioned on x, which may slow down the convergence rate due to their

strong statistical dependence. We address this issue by introducing intermediate proposal steps after Step 3, in which new values for coinciding blocks of x and σ_x^2 are proposed using their joint posterior densities. Secondly, the blind nature of the problem creates many distinct local optimums corresponding to different x and γ pairs. Once a suboptimal[2] configuration has been reached with a corresponding noise level σ_v^2, the sampler can get stuck on this configuration for a long time since x, γ, and σ_v^2 are all sampled conditioned on each other. This problem is especially important when the noise variance σ_v^2 gets smaller because it results in a posterior distribution with sharper and more isolated peaks. One way to resolve this issue is to sample x and γ jointly, which can be realized by marginalizing either x or γ from the sampling distributions given in Step 6 or 4, respectively. However, both of these approaches lead to complicated sampling distributions, from which sampling is not feasible due to high dimensionality. A less effective but more feasible way is to sample either x and σ_v^2, or γ and σ_v^2 jointly, creating more freedom for sampling σ_v^2. This allows σ_v^2 to assume larger values more frequently. Note that as σ_v^2 gets larger, the effect of likelihood is reduced on the conditional posteriors of both x or γ, and it becomes easier to escape from a local optimum. Since $L \ll K$, we choose to sample γ and σ_v^2 jointly, because it is computationally much more efficient compared to sampling x and σ_v^2 jointly.

Let us first introduce the intermediate sampling steps for blocks of x and σ_x^2. We define ℓ_n as the right-hand neighborhood of length Q for index n, i.e., $\ell_n = \{n, n+1, \ldots, n+Q-1\}$ for $n = 0, 1, \ldots, K-Q$, and let $\sim\ell_n$ be the complement of ℓ_n in $\{0, \ldots, Q-1\}$. We represent the variable blocks pointed by the neighborhood ℓ_n as $\sigma_{x_{\ell_n}}^2 = [\sigma_{x_n}^2, \sigma_{x_{n+1}}^2, \ldots, \sigma_{x_{n+Q-1}}^2]^T$ and $x_{\ell_n} = [x_n, x_{n+1}, \ldots, x_{n+Q-1}]^T$. Similarly, $\sigma_{x_{\sim\ell_n}}^2$ and $x_{\sim\ell_n}$ represent the blocks of the remaining $K-Q$ variables. At each iteration of the new sampler, we propose new values for the blocks $\sigma_{x_{\ell_n}}^2$ and x_{ℓ_n} using the proposal distribution $p(\sigma_{x_{\ell_n}}^2, x_{\ell_n}|y, x_{\sim\ell_n}, \sigma_{x_{\sim\ell_n}}^2, \gamma, \sigma_v^2)$, which is the joint posterior distribution of $\sigma_{x_{\ell_n}}^2$ and x_{ℓ_n} conditioned on all the other variables. Since this is actually a valid Gibbs sampling step, the Metropolis-Hastings acceptance probability is always 1 for the proposals. The neighborhood ℓ_n is updated after each iteration

---

[2]Here, by suboptimality, we refer to the configurations of x and γ that achieves a likelihood value as high as the true values of x and γ.

Step 1. Sample α_x from $p(\alpha_x \mid \sigma_x^2, \beta_x)$
Step 2. Sample β_x from $p(\beta_x \mid \sigma_x^2, \alpha_x)$
Step 3. Sample σ_x^2 from $p(\sigma_x^2 \mid x, \alpha_x, \beta_x)$
Step 4. Sample $x_{\sim \ell_n}$ from $p(x_{\sim \ell_n} \mid y, \sigma_x^2, \gamma, \sigma_v^2)$
Step 5. Sample $\sigma_{x_{\ell_n}}^2$ from $p(\sigma_{x_{\ell_n}}^2 \mid y, x_{\sim \ell_n}, \gamma, \sigma_v^2, \alpha_x, \beta_x)$
Step 6. Sample x_{ℓ_n} from $p(x_{\ell_n} \mid y, x_{\sim \ell_n}, \sigma_{x_{\ell_n}}^2, \gamma, \sigma_v^2)$
Step 7. Sample σ_v^2 from $p(\sigma_v^2 \mid y, x)$
Step 8. Sample γ from $p(\gamma \mid y, x, \sigma_v^2)$

as follows. At i^{th} iteration we set $n = \mathrm{mod}(i - 1, K - Q + 1)$, where $\mathrm{mod}(a, b)$ is the modulo operator returning the remainder after division of a by b. This creates a sliding window over σ_x^2 and x that shifts one index to the right at each iteration. Hence, the whole sequence is scanned after every $K - Q + 1$ iterations.

For a given neighborhood ℓ_n, sampling from the joint conditional posterior $p(\sigma_{x_{\ell_n}}^2, x_{\ell_n} \mid y, x_{\sim \ell_n}, \gamma, \sigma_v^2, \alpha_x, \beta_x)$ can be realized in two consecutive steps, i.e.,

- First, sample $\sigma_{x_{\ell_n}}^2$ from $p(\sigma_{x_{\ell_n}}^2 \mid y, x_{\sim \ell_n}, \gamma, \sigma_v^2, \alpha_x, \beta_x)$,
- Then, sample x_{ℓ_n} from $p(x_{\ell_n} \mid y, x_{\sim \ell_n}, \sigma_{x_{\ell_n}}^2, \gamma, \sigma_v^2)$.

These steps can be inserted right after Step 4, as Steps 5 and 6, respectively. Note that the block x_{ℓ_n} is still being sampled in Step 4, which is redundant because it would not be conditioned on in Step 5 and immediately replaced with the new values obtained in Step 6. Therefore, sampling x_{ℓ_n} can be skipped in Step 4, which forms a new step where only $x_{\sim \ell_n}$ is sampled from $p(x_{\sim \ell_n} \mid y, \sigma_x^2, \gamma, \sigma_v^2)$. Although $\sigma_{x_{\ell_n}}^2$ is also re-sampled in Step 5, it is still conditioned on in Step 4, so we cannot skip sampling $\sigma_{x_{\ell_n}}^2$ in Step 3. The first 6 steps of the resulting sampling scheme is given in Table II .

In order to jointly sample σ_v^2 and γ, we need to sample from $p(\sigma_v^2, \gamma \mid y, x)$, which can also be realized in two steps as:

- First, sample σ_v^2 from $p(\sigma_v^2 \mid y, x)$,
- Then, sample γ from $p(\gamma \mid y, x, \sigma_v^2)$.

These constitute the last two steps of the proposed sampler in Table II.

We emphasize that the sampling distribution in Step 4 is not associated with the joint posterior $p(\theta \mid y)$ anymore. Therefore, the proposed sampler is not a classical Gibbs sampler. Instead, it is a PCG sampler since the procedure described above is completely consistent with the *marginalization* and *trimming* operations described in [27]. PCG samplers are generalizations of the block Gibbs sampler, where some of the variables are not sampled from their full conditional posteriors. This provides more freedom for the sampler to jump from one point to another in the sampling space, which usually increases the mixing rate.

Compared to the classical Gibbs sampler in Table I, the first three steps are the same and sampling distribution for Step 8 is already given in (20). The sampling distributions for the remaining steps are presented below. Derivations are provided in the supplemental material.

1) Sampling Distribution for Step 4: The sampling distribution for $x_{\sim \ell_n}$ block is obtained by marginalizing x_{ℓ_n} from $p(x \mid y, \sigma_x^2, \gamma, \sigma_v^2)$. The resulting distribution is also a multivariate Gaussian distribution given by

$$p(x_{\sim \ell_n} \mid y, \sigma_x^2, \gamma, \sigma_v^2) = \mathcal{N}(x_{\sim \ell_n}; \tilde{\mu}_{\sim \ell_n}, \tilde{\Sigma}_{\sim \ell_n}). \quad (23)$$

The posterior mean $\tilde{\mu}_{\sim \ell_n}$ and covariance $\tilde{\Sigma}_{\sim \ell_n}$ are defined as

$$\tilde{\mu}_{\sim \ell_n} = \frac{1}{\sigma_v^2} \tilde{\Sigma}_{\sim \ell_n} H_{\sim \ell_n}^T D_{\ell_n}^T y, \quad (24a)$$

$$\tilde{\Sigma}_{\sim \ell_n} = \left(\frac{1}{\sigma_v^2} H_{\sim \ell_n}^T D_{\ell_n} H_{\sim \ell_n} + \Sigma_{\sim \ell_n}^{-1} \right)^{-1}, \quad (24b)$$

where $D_{\ell_n} = I - \frac{1}{\sigma_v^2} H_{\ell_n} \tilde{\Sigma}_{\ell_n} H_{\ell_n}^T$ with $H_{\sim \ell_n}$ and H_{ℓ_n} represent the matrices formed by the columns of H indexed by $\sim \ell_n$ and ℓ_n respectively. The covariance matrices are defined as $\Sigma_{\sim \ell_n} = \mathrm{diag}(\sigma_{x_{\sim \ell_n}}^2)$, $\Sigma_{\ell_n} = \mathrm{diag}(\sigma_{x_{\ell_n}}^2)$, and $\tilde{\Sigma}_{\ell_n} = (\frac{1}{\sigma_v^2} H_{\ell_n}^T H_{\ell_n} + \Sigma_{\ell_n}^{-1})^{-1}$.

2) Sampling Distribution for Step 5: Similarly, the sampling distribution for $\sigma_{x_{\ell_n}}^2$ block is obtained by marginalizing x_{ℓ_n} from $p(\sigma_{x_{\ell_n}}^2, x_{\ell_n} \mid y, x_{\sim \ell_n}, \gamma, \sigma_v^2, \alpha_x, \beta_x)$, which is given by

$$p(\sigma_{x_{\ell_n}}^2 \mid y, x_{\sim \ell_n}, \gamma, \sigma_v^2, \alpha_x, \beta_x)$$

$$\propto \frac{|\tilde{\Sigma}_{\ell_n}|^{1/2}}{|\Sigma_{\ell_n}|^{1/2}} \exp \left(\frac{1}{2} \tilde{\mu}_{\ell_n}^T \tilde{\Sigma}_{\ell_n}^{-1} \tilde{\mu}_{\ell_n} \right) p(\sigma_{x_{\ell_n}}^2). (25)$$

where $\tilde{\mu}_{\ell_n} = \frac{1}{\sigma_v^2} \tilde{\Sigma}_{\ell_n} H_{\ell_n}^T \tilde{y}_{\sim \ell_n}$ and $\tilde{y}_{\sim \ell_n} = y - H_{\sim \ell_n} x_{\sim \ell_n}$. Direct sampling from (25) is not possible, since the form of the distribution is not well-known. Therefore, we employ a MH or Slice sampling step.

3) Sampling Distribution for Step 6: The sampling distribution for x_{ℓ_n} block is given by

$$p(x_{\ell_n} \mid y, x_{\sim \ell_n}, \sigma_{x_{\ell_n}}^2, \gamma, \sigma_v^2) = \mathcal{N}(x_{\ell_n}; \tilde{\mu}_{\ell_n}, \tilde{\Sigma}_{\ell_n}), \quad (26)$$

with $\tilde{\mu}_{\ell_n}$ and $\tilde{\Sigma}_{\ell_n}$ are as defined in Sections III-C1 and III-C2.

4) Sampling Distribution for Step 7: The sampling distribution for the noise variance σ_v^2 is obtained by marginalizing γ from $p(\sigma_v^2, \gamma \mid y, x)$, which is given by

$$p(\sigma_v^2 \mid y, x)$$

$$\propto \left(\frac{1}{\sigma_v^2} \right)^{N/2} |\tilde{\Sigma}_\gamma|^{1/2} \exp \left(\frac{1}{2} \tilde{\mu}_\gamma^T \tilde{\Sigma}_\gamma^{-1} \tilde{\mu}_\gamma - \frac{1}{2\sigma_v^2} y^T y \right) p(\sigma_v^2). \quad (27)$$

Although it is not straightforward to sample from (27), we can efficiently employ MH or Slice sampling methods due to its univariate form. In the next section, we investigate the scaling and time-shifting ambiguities existing in the BD problems and propose two intermediate sampling steps accounting these ambiguities.

D. Scale and Time-Shift Ambiguities

Scaling and time-shift ambiguities are inherent in blind deconvolution problems preventing unique recovery of the pulse shape and the sparse sequence. For a given solution pair (x, h), one can produce infinitely many different solutions consisting of

the !scaled versions $(\alpha x, h/\alpha)$ with $\alpha \in \mathbb{R}$, which constitutes the scale ambiguity. The time-shifted versions $(x * \delta_n, h * \delta_{-n})$, where δ_n is the Kronecker delta with spike at position n and $*$ denotes the convolution operation, is another source of ambiguity constituting the time-shift ambiguity. In practice, recovery of the true parameters up to an arbitrary scale and time-shift does not cause a major problem and is usually sufficient.

1) Scale Ambiguity: In Bayesian framework, assignment of the prior distributions can eliminate the scaling ambiguity only if the scaling of the prior distributions is fixed at an anchor point. Otherwise, with such a scaling transformation $\{ax, a^2\sigma_x^2, \alpha_x, a^2\beta_x, \gamma/a, \sigma_\gamma^2/a^2, \sigma_v^2\}$ with $a > 0$ that leaves the likelihood invariant, one can scale the posterior with a^{L-K+4}. This means that the posterior distribution can be increased arbitrarily by decreasing the scale a, indicating nonexistence of a global optimum. However, this issue can be avoided by setting, for example, σ_γ^2 to a fixed constant. In this case, the sampler eventually converges to a fixed scale α' associated with the value of σ_γ^2. However, convergence might be slow since the parameters are sampled conditioned on each other. A common approach to accelerate the convergence is to introduce an intermediate Metropolis-Hastings (MH) sampling step. Following this approach, once the current values of (x, γ, σ_x^2) are sampled, we first propose the new values for x^* and γ^*, i.e., $(x^*, \gamma^*) = (\alpha x, \gamma/\alpha)$, by sampling the scaling factor α from the proposal distribution $q(\alpha) = \mathcal{N}(\alpha; 0, \sigma_\alpha^2)$ with known variance σ_α^2 and then propose σ_x^{2*} by sampling from $p(\sigma_x^{2*}|x^*, \alpha_x, \beta_x)$, yielding the complete proposal distribution $q(x^*, \gamma^*, \sigma_x^{2*}|x, \gamma, \sigma_x^2) = q(\alpha)p(\sigma_x^{2*}|x^*, \alpha_x, \beta_x)$. The proposed values are accepted with probability p_α, which is given by

$$p_\alpha = \min\left\{1, \alpha^L \exp\left(\frac{\alpha^4 - 1}{2\alpha^2\sigma_\alpha^2}\right) \prod_{n=0}^{K-1}\left[\frac{x_n^2 + 2\beta_x}{\alpha^2 x_n^2 + 2\beta_x}\right]^{\alpha_x + 0.5}\right\} \tag{28}$$

We note that a better strategy would be adjusting the parameters α_x and β_x as well by sampling new values α_x^* and β_x^* from $p(\alpha_x^*, \beta_x^*|\sigma_x^{2*})$. However, achieving a closed form expression for the acceptance probability would not be possible.

2) Time-Shift Ambiguity: Unlike scaling ambiguity, time-shift ambiguity does not fully apply to our case due to the edge effects, i.e., the edges of the reconstructed observation sequence will be corrupted when shifted versions of a given solution are considered. This is because we model the observation sequence using linear convolution of finite length sequences. Therefore, solution pairs with different shifts cannot achieve exactly the same likelihood value. However, although large time-shift ambiguities are explicitly avoided, time-shifts of very short lengths can still correspond to a similar level of likelihood, and hence, needs to be addressed. Since both sequences x and h are sampled conditioned on the current value of the other, jumps between different time-shift configurations rarely occur. To increase the frequency of these jumps, we employ the circular shift compensation method proposed in [45]. Following the scale compensation step, new values of x, σ_x^2 and γ are proposed using the proposal distribution $q(x^*, \sigma_x^{2*}, \gamma^*|x, \sigma_x^2, \gamma) =$

TABLE III
BASELINE CLASSICAL BERNOULLI-GAUSSIAN SAMPLER

Step 1.	For $n = 0, 1, \ldots, K - 1$, sample (s_n, x_n) from $p(s_n, x_n|y, x_{\sim n}, \gamma, \sigma_v^2)$
Step 2.	Sample γ from $p(\gamma|y, x, \sigma_v^2)$
Step 3.	Sample σ_v^2 from $p(\sigma_v^2|y, x, \gamma)$

$q(x^*, \sigma_x^{2*}|x, \sigma_x^2)p(\gamma^*|y, x^*, \sigma_v^2)$ where

$$q(x^*, \sigma_x^{2*}|x, \sigma_x^2) = \begin{cases} 0.5 \text{ if } (x^*, \sigma_x^{2*}) = (x \circledast \delta_{-1}, \sigma_x^2 \circledast \delta_{-1}) \\ 0.5 \text{ if } (x^*, \sigma_x^{2*}) = (x \circledast \delta_1, \sigma_x^2 \circledast \delta_1) \end{cases} \tag{29}$$

and $\circledast$ denotes the circular convolution. It can be shown that the MH acceptance probability p_s is given by

$$p_s = \min\left\{1, \frac{|\tilde{\Sigma}_{\gamma^*}|}{|\tilde{\Sigma}_\gamma|} \exp\left(\frac{1}{2}\tilde{\mu}_{\gamma^*}^T \tilde{\Sigma}_{\gamma^*}^{-1} \tilde{\mu}_{\gamma^*} - \frac{1}{2}\tilde{\mu}_\gamma^T \tilde{\Sigma}_\gamma^{-1} \tilde{\mu}_\gamma\right)\right\} \tag{30}$$

where both $(\tilde{\mu}_{\gamma^*}, \tilde{\Sigma}_{\gamma^*})$ and $(\tilde{\mu}_\gamma, \tilde{\Sigma}_\gamma)$ can be calculated through (21) using x^* and x respectively.

IV. SIMULATIONS

In this section, we present our numerical studies to assess the performance of the proposed samplers and compare our results with the classical BG deconvolution approach, in which the sparsity of x is enforced by the BG prior, as described in (3)–(5), instead of the NIG prior used in this work. The form of the slightly modified version of Cheng *et al.*'s classical Gibbs sampler employing BG model is given in Table III. Step 1 of this sampler consists of K sub-update steps, where pairs of (s_n, x_n) are sampled jointly from their joint posterior distribution at each sub-step. Jointly sampling (s_n, x_n) is also completed in two steps, first sampling s_n from $p(s_n|y, x_{\sim n}, \gamma, \sigma_v^2)$ and then sampling x_n from $p(x_n|y, x_{\sim n}, s_n, \gamma, \sigma_v^2)$. The remaining sampling steps are the same as the proposed samplers. We also consider the M-tuple Gibbs sampler proposed in [24], with $M = 3$. It modifies Step 1 of Table III to sample tuples of length 3 at a time and considerably improves the convergence rate of classical BGS. Hence, it constitutes a stronger baseline for assessing the performance of our samplers. Throughout the simulations, we use these as our baseline samplers and call them, respectively, as the classical Bernoulli-Gaussian Sampler (BGS) and 3-Tuple BGS. We refer to the proposed samplers using the Normal-Inverse-Gamma law as NIGS-1 (for the classical Gibbs sampler) and NIGS-2 (for the partially collapsed Gibbs sampler).

Due to the scale and minor time-shift ambiguities that are inherent in blind deconvolution problems, all recovery results presented in this section are up to an arbitrary scale and shift factor. The compensation steps introduced in the previous section resolve these ambiguities only for the sampling stage, i.e., the resulting estimated sequences are not necessarily expected to match the scale and shift of the true sequences. Therefore, for any given estimation pair $(\hat{x}, \hat{h})$, we present the corrected versions x' and h', which are given by $x' = (\hat{x} * \delta_{-n})/a$ and

$h' = a(\hat{h} * \delta_n)$, where the correction factors are found using

$$(a, n) = \arg\min_{a \in \mathbb{R}, n \in \mathbb{Z}} \|h - a(\hat{h} * \delta_n)\|. \tag{31}$$

A. A Measure of Convergence Rate

In order to empirically measure the convergence rate, we employ the iterated graphical monitoring approach proposed by Brook and Gelman in [46]. The diagnostic is based on comparing the between and within chain covariance matrices, denoted by S_b and S_w respectively, of different simulation chains that are running simultaneously. More precisely, for multivariate simulations, convergence is declared when the multivariate potential scale reduction factor (MPSRF), denoted by $\hat{R}$, becomes close to 1. A typical threshold in which the MPSRF is expected to fall below is 1.2 as suggested in [46]. The MPSRF is defined as

$$\hat{R} = \frac{i-1}{i} + \frac{q+1}{q}\lambda_{\max}, \tag{32}$$

where i denotes the number of MCMC iterations, q is the total number of distinct simulation chains running simultaneously and $\lambda_{\max}$ is the maximum eigenvalue of the matrix $S_w^{-1}S_b$. The definitions of the between chain covariance matrix S_b and the within chain covariance matrix S_w are given as

$$S_b = \frac{1}{q-1}\sum_{j=1}^{q}(\bar{\theta}_{\cdot j} - \bar{\theta}_{\cdot\cdot})(\bar{\theta}_{\cdot j} - \bar{\theta}_{\cdot\cdot})^T, \tag{33a}$$

$$S_w = \frac{1}{q(i-1)}\sum_{j=1}^{q}\sum_{l=1}^{i}(\theta_{lj} - \bar{\theta}_{\cdot j})(\theta_{lj} - \bar{\theta}_{\cdot j})^T, \tag{33b}$$

where θ_{lj} denotes the lth sample of jth chain. Also, $\bar{\theta}_{\cdot j}$ and $\bar{\theta}_{\cdot\cdot}$ represent the local mean of the jth chain and the global mean of the all chains respectively. Therefore, for a given data, we run several chains from different initial points and monitor the MPSRF value to assess the convergence.

B. Recovery Performance on Mendel's Sequence

As an illustrative example, we present the recovery results of the proposed samplers on a given observation sequence and provide a comparison with the classical BGS described above in order to show its inefficiency. The observation sequence y_n was generated based on the linear convolution model given in (1). As the sparse sequence x_n, we used the well-known Mendel's sequence, which models a 1-D sparse reflectivity profile for seismic blind deconvolution [2]. The sequence is depicted with bullets in sparse recovery plots of Fig. 2. As the short pulse sequence h_n, we used the following sequence

$$h_n = \cos\left((n - 10)\pi/4\right)\exp\left(-|0.225n - 2|^{1.5}\right) \tag{34}$$

for $n = 0, 1, \ldots, 20$, which is the same sequence used in both [1] and [24]. It is represented with solid line in the upper-right plot of Fig. 2 with solid line. The noise variance was set as $\sigma_v^2 = 4.8 \times 10^{-6}$, which corresponds to a Signal-to-Noise Ratio (SNR) of 25 dB.

The lengths of the sparse sequence x and the pulse sequence h are $K = 300$ and $T = 21$, respectively, yielding a length $N = 321$ measurement sequence y. For this experiment, we did not impose any frequency domain constraints on the pulse sequence and set the subspace matrix as identity, i.e., $A = I$. For Step 1 of NIGS-1, and Steps 1, 5, and 7 of NIGS-2, we employed univariate Slice sampling approach. We set the window length for NIGS-2 as[3] $Q = 10$ and used the following fixed values for the parameters of baseline samplers: $\sigma_x^2 = 1$, $\sigma_\gamma^2 = 10$, and $\pi_0 = 1 - |x|_0/K$, where $|x|_0$ denotes the number of nonzero elements in the true sequence x. For all samplers, we generated 10^4 samples and used the last 25% for producing the estimations.

The four plots in the left and middle columns of Fig. 2 illustrate the sparse sequences recovered by each sampler. The recoveries obtained by the proposed samplers and 3-Tuple BGS are almost identical and perfectly match the true sequence. On the other hand, the sparse sequence recovered by the classical BGS contains inaccurate nonzero entries where the actual spikes are located closely. This example shows the main inefficiency of the classical BGS. Since the (s_n, x_n) tuples are sampled conditioned on the current value of adjacent entries, it usually takes a large number of iterations to escape from a local optimum. Therefore, initialization plays an important role in the convergence behavior of BGS. This issue will become clearer once we present the convergence analysis.

The recovered pulses, as shown in middle lower plot in Fig. 2, are almost identical and perfectly match the true pulse shape for all samplers. This is usually the case since the number of parameters to be estimated, i.e., the degree of freedom, is much smaller for the pulse sequence.

In order to compare the convergence rates of the samplers, we illustrate the evolution of the MPSRF corresponding to samples of x and γ for all samplers in the lower-right plot in Fig. 2. We simulated 10 independent chains with distinct initialization for each sampler and updated the MPSRF after every 200 iterations using the last 50% of the generated samples. The MPSRF curves of the proposed samplers NIGS-1 and NIGS-2 consistently decrease and fall below the convergence threshold of 1.2 after around 5000 and 2000 iterations, respectively. This suggests that each of the individual chains of NIGS-1 and NIGS-2 converged to the same local optimum, which is very likely to be the global one, regardless of where they are initialized at the parameter space. However, the MPSRF curve of classical BGS fails to converge and cannot fall below the convergence threshold during the simulation duration. This result confirms that individual chains of BGS get stuck on a local optimum for a much larger number of iterations. Note that unlike classical BGS, 3-Tuple BGS reaches the convergence threshold at around 2500 iterations, which is due to its improved convergence rate.

C. Empirical Analysis of the Convergence Rates

Next, we evaluate the convergence performance of the proposed samplers over a set of randomly generated sparse sequences. The generated sequences have length $K = 300$. The nonzero positions were randomly distributed across the sequence. In order to ensure that there are no localized regions with all nonzero elements, we divided each sequence into multiple

---

[3]We observed that the window length has no significant impact on neither the convergence rate nor the overall success rate, as long as it is selected in the same order of magnitude as the pulse length.

segments of equal lengths and randomly choose a nonzero index for each segment. A 6% sparsity level, defined by the ratio of the number of nonzero entries to the total length of the sequence, was maintained for all generated sequences. The amplitudes for the nonzero positions were independently drawn from a univariate zero-mean Gaussian distribution with a variance of 0.5. A total of 50 distinct observations were generated by convolving each of 50 different sparse sequences with the pulse sequence given in (34), which were then corrupted by additive noise such that the SNR is 20 dB.

For each observation, we run all samplers 10 times with different initialization to obtain 10 independently simulated chains. The MPSRF values are calculated on these chains after every 100 MCMC iterations by using the last 50% of the samples. Same as before, we set the convergence threshold for MPSRF as 1.2. In Fig. 3, we illustrate the ratio of converged sequences over all 50 sequences as a function of the MCMC iteration for all samplers. In order to individually investigate the convergence performance on different unknowns, we calculated the MPSRF curves separately for the sparse sequence x and pulse coefficients γ, which are represented in the upper left and right figures, respectively.

We see in the top left figure that the ratio of converged sequences for the proposed samplers increases quite rapidly compared to classical BGS, while all three are outperformed by 3-Tuple BGS. The outstanding convergence performance of 3-Tuple BGS is mainly due to the fact that many of the configurations leading to a local optimum occur within a short neighborhood, which can be escaped rapidly even with a very small tuple size M. However, the computational complexity of M-Tuple BGS increases exponentially with the tuple size M as shown in [24]. We also observed that NIGS-2 achieves a slightly better convergence rate compared to NIGS-1, which indicates that the intermediate sampling steps introduced in NIGS-2 help accelerating the mixing rate. Comparing the left and right figures, we see that convergence rate is faster for all samplers in the case of pulse coefficients γ. This is indeed an expected result since the degree of freedom is considerably higher for the sparse sequence. We further note that the convergence rates of the proposed samplers are almost as fast as that of 3-Tuple BGS, and they still significantly outperform classical BGS.

D. Computational Complexity Analysis

The study presented above compared the convergence rates of the different algorithms in terms of the number of MCMC iterations. Next, we incorporate the computational complexities of the samplers into the analysis. In order to investigate the empirical computational complexities, we measured the processing times based on unoptimized MATLAB R2019a implementation of the samplers on an Intel Core i5-8265 U processor. For the implementation of classical and 3-Tuple BGS, we used the efficient numerical method presented in [24]. The average elapsed time for one iteration of each sampler as a function of the sparse sequence length K is illustrated in the left plot in Fig. 4. For all considered values of K, the number of spikes was adjusted accordingly to maintain the 6% sparsity level. It can be observed that the cost per iteration in terms of processing

time is significantly higher for 3-Tuple BGS for all values of K. Moreover, the difference between the computational complexities of the proposed samplers and classical BGS increases considerably for larger values of K. In order to better illustrate the difference of computational complexities, we also present the computational gain of all samplers relative to 3-Tuple BGS in the right plot in Fig. 4. We define the gain as the ratio of cost per iteration of 3-Tuple BGS to those of other samplers. It is clear that NIGS-1 and NIGS-2 are, respectively, at least around 10 and 5 times computationally more efficient compared to 3-Tuple BGS. Moreover, the computational gain of the proposed samplers increases for larger values of K. We also observed that classical BGS is computationally more efficient compared to NIGS-2 for smaller values of K, even though its cost per iteration increases significantly for larger values of K.

Based on empirically measured computational costs of the samplers, we can also compare the convergence rates in terms of the simulation time. The comparisons are provided in bottom rows of Fig. 3 for both sparse sequence x and the pulse coefficients γ using length $K = 300$ sparse sequences. The results indicate that the excessive computational cost of 3-Tuple BGS overwhelms its outstanding convergence rate. Due to its lower cost per iteration, NIGS-1 achieves the fastest convergence rate in terms of processing time for both x and γ. NIGS-2 has a slightly higher cost per iteration compared to NIGS-1, but still attains a similar convergence rate to that of 3-Tuple BGS for x, which is even better for γ. Due to increasing computational gain with K, it would also be reasonable to suggest that the difference between convergence rates will be more substantial for larger values of K, favoring the use of proposed samplers in practice.

E. Overall Recovery Performance for Different Scenarios

In this section, we investigate the recovery performance of the proposed samplers is investigated under various different SNR and sparsity levels, along with the comparisons with the baseline samplers. We considered 11 different SNR levels ranging between 10 dB and 30 dB with 2 dB separation between each level. For each SNR level, we investigated 26 different scenarios, where the number of spikes is increased from 10 to 60 with 2 increments. The length of the sparse sequence for each scenario was fixed at $K = 300$, yielding sparsity levels ranging from 3% to 20%. We created 20 different random sparse sequences for each one of these 286 scenarios using the same way described in Section IV-C. For the pulse sequence, we used the normalized first derivative of Gaussian pulse, given by

$$h_n = 2(n\pi f_c/f_s - 2) \exp\left(0.5 - 2(n\pi f_c/f_s - 2)^2\right) \quad (35)$$

for $n = 0, 1, \ldots, 22$ with center frequency $f_c = 2$ GHz and sampling rate $f_s = 36$ GHz. This pulse shape constitutes a strictly short duration sequence in the time domain, which is also nearly bandlimited in the frequency domain. Therefore, for these experiments, we construct the columns of the pulse subspace matrix A using the first $L = 8$ DPS sequences of length $T = 23$. We run each sampler for 10^4 iterations and use the last 25% of the samples for estimation of the unknowns. We declare a recovered sparse sequence x' (or pulse sequence h') successful
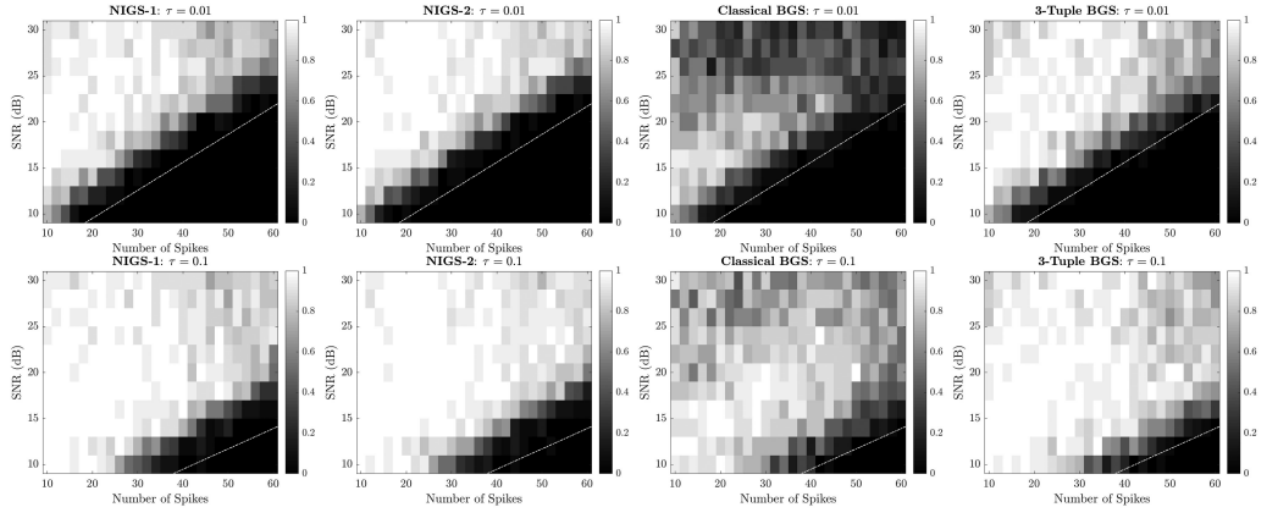
Fig. 5. Empirical successful recovery rates for the sparse sequence x at different SNR and sparsity levels for $\tau = 0.01$ and $\tau = 0.1$. Dashed white line depicts the transition boundary for 3-Tuple BGS.

if the Normalized Mean Squared Error (NMSE) is less than a given threshold τ, i.e.,

$$\frac{\|x - x'\|^2}{\|x\|^2} \le \tau, \qquad \frac{\|h - h'\|^2}{\|h\|^2} \le \tau. \tag{36}$$

In Fig. 5, we illustrate the empirical successful recovery rates of the samplers for the sparse sequence x for two different values of τ. We first focus on the transition boundaries, which identify the feasible regions in the sparsity/SNR plane for which successful recovery, as defined in (36), is possible. In order to better illustrate the differences, we draw the transition boundary of 3-Tuple BGS, associated with the corresponding τ, as a reference at all plots. It can be observed that feasible regions of the proposed samplers are slightly more restricted. This indicates that for a given sparsity level, the proposed samplers require a slightly higher SNR level for successful recovery. Nevertheless, the feasible regions are quite similar for all samplers. On the other hand, the rate of successful recovery within the feasible region is significantly higher for the proposed sampler as opposed to classical BGS. This is due to the fact that classical BGS requires a considerably larger number of iterations than 10^4 to converge to the true stationary distribution. In addition, we also observed that NIGS-2 achieves the highest overall success rate within its feasible region even though 3-Tuple BGS has a slightly larger feasible region. This is also justified by the average success rates, given in Table IV , corresponding to different values of τ, over all 286 scenarios. Despite the slightly smaller feasible region, NIGS-2 achieves a success rate as high as that of 3-Tuple BGS. The results also show that NIGS-2 achieves a higher overall success rate compared to NIGS-1, which provides empirical evidence that the intermediate sampling steps of NIGS-2 improves the convergence rate. Moreover, it is clear that both proposed samplers significantly outperform classical BGS. We present the corresponding success rates for the pulse shape h in Fig. 6. Comparing the transition plots in Figs. 5 and 6, our first observation is that feasible regions for the pulse sequence are

TABLE IV
OVERALL EMPIRICAL SUCCESSFUL RECOVERY RATES FOR THE SPARSE SEQUENCE x (TOP) AND THE PULSE SEQUENCE h (BOTTOM)

	$\tau = 0.01$	$\tau = 0.04$	$\tau = 0.07$	$\tau = 0.1$
NIGS-1	0.54	0.69	0.75	0.79
NIGS-2	0.57	0.72	0.78	0.81
Classical BGS	0.36	0.53	0.62	0.69
3-Tuple BGS	0.57	0.73	0.79	0.82

Pulse Sequence				
	$\tau = 0.01$	$\tau = 0.04$	$\tau = 0.07$	$\tau = 0.1$
NIGS-1	0.82	0.92	0.94	0.96
NIGS-2	0.83	0.93	0.96	0.97
Classical BGS	0.85	0.93	0.95	0.96
3-Tuple BGS	0.86	0.93	0.95	0.96

The NMSE Threshold τ is Varied Between 0.01 and 0.1.

significantly more extensive for all samplers. As it can be seen from $\tau = 0.01$ case, similar to the recovery of sparse sequence, the feasible regions are more extensive for the baseline samplers. Another observation is that the success rates of classical BGS are significantly higher for the pulse sequence. This indicates that the main source of the inefficiency of classical BGS is different configurations of sparse sequences rather than the pulse shape. Table IV shows that all samplers achieve a similar level of success rate for the pulse sequence except for $\tau = 0.01$, which can be explained by the smaller feasible regions of the proposed samplers. Overall, the success rates are consistently higher compared to the recovery of sparse sequences, implying that it is easier to recover the pulse shape in most cases.

Finally, we also compared the overall success rates of the samplers at different SNR levels in Fig. 7. Considering the recovery of sparse sequences, given in the left figure, all samplers perform quite similarly for lower SNRs, and the success rates increase as SNR increases, except for classical BGS. Its success rate reaches a stable level at around 20 dB and then starts
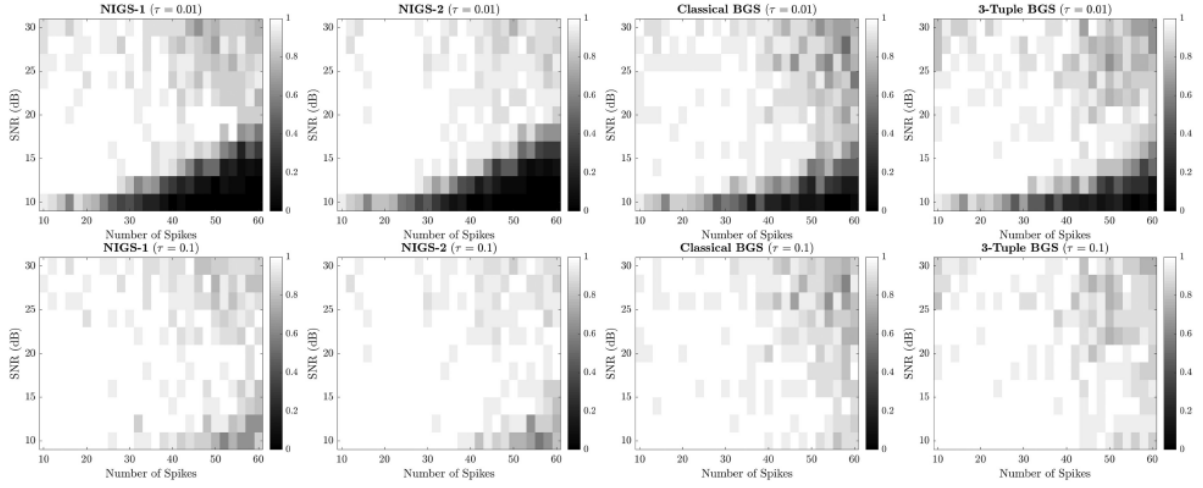
Fig. 6. Empirical successful recovery rates for the pulse sequence h at different SNR and sparsity levels for $\tau = 0.01$ and $\tau = 0.1$.
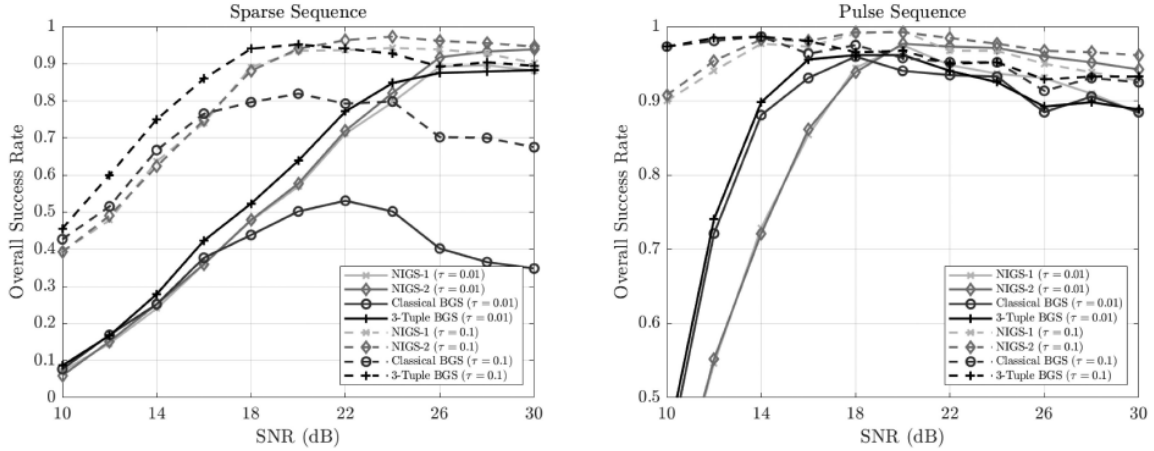

Fig. 7. Overall empirical successful recovery rates for the sparse sequence x (left) and the pulse sequence h (right) at different SNR levels. Solid and dashed lines correspond to $\tau = 0.01$ and $\tau = 0.1$ cases respectively.

slightly decreasing after 24 dB. This seems non-intuitive but at high SNRs, peaks of the likelihood function get sharper, and it becomes overwhelmingly difficult to escape from a locally optimum configuration. The proposed samplers and 3-Tuple BGS are not affected by this effect due to their improved ability to escape local optimums. The figure illustrates the success rate curves for both $\tau = 0.01$ and $\tau = 0.1$. It can be observed that at high SNRs, the number of simulations of NIGS-2 with resulting NMSE less than $\tau = 0.01$ is more than those of 3-Tuple BGS both with NMSE less than $\tau = 0.1$ and $\tau = 0.01$. This suggests that NIGS-2 is more successful at escaping local optimums. However, 3-Tuple BGS outperforms all other samplers at lower SNRs. The right figure in Fig. 7 demonstrates the same analyses for the recovery of the pulse sequence. As expected, at lower SNRs, the baseline samplers achieve higher success rates due to their extensive feasible regions. On the other hand, they are being outperformed by the proposed samplers at high SNRs, because of enhanced convergence characteristics of NIGS-1 and NIGS-2. We also note that it is possible to observe a similar performance reduction effect at high SNRs.

V. CONCLUSION

In this paper, we studied the problem of sparse blind deconvolution under a Bayesian framework and presented efficient MCMC based estimation methods for jointly recovering two unknown sequences from their noisy convolutions. We derived two different hierarchical Gibbs samplers under a NIG prior enforcing sparsity from, which forms a continuous valued alternative to the conventional BG model. While the first sampler follows a classical Gibbs sampler structure, the second one employs a PCG sampler scheme, which incorporates additional sampling steps to reduce the statistical dependence of the variables in an attempt to enhance the convergence rate. By moving the problem into a completely continuous valued framework, we avoided the computational burdens due to the discrete nature of the BG model. The proposed samplers were evaluated on an empirical basis via extensive numerical simulations and compared with the existing sampling schemes that are based on the BG model. The obtained results demonstrated the effectiveness of the proposed samplers on achieving successful

recovery under various different settings. Comparisons with the baseline samplers demonstrated a significant increase in the convergence rate, along with considerable computational gains. As a result, the proposed methods can be used in a variety of real-life applications involving blind deconvolution problems with sparsity and time/frequency domain constraints.

REFERENCES

[1] Q. Cheng, R. Chen, and T.-H. Li, "Simultaneous wavelet estimation and deconvolution of reflection seismic signals," *IEEE Trans. Geosci. Remote Sens.*, vol. 34, no. 2, pp. 377–384, Mar. 1996.

[2] J. M. Mendel and J. Goutsias, "One-dimensional normal-incidence inversion: A solution procedure for band-limited and noisy data," *Proc. IEEE*, vol. 74, no. 3, pp. 401–414, Mar. 1986.

[3] E. Lam and J. Goodman, "Iterative statistical approach to blind image deconvolution," *J. Opt. Soc. Amer. A, Opt. Image Sci. Vis.*, vol. 17, pp. 1177–1184, 2000.

[4] Y. Huang, E. Chouzenoux, and J.-C. Pesquet, "Unrolled variational Bayesian algorithm for image blind deconvolution," 2021, *arXiv:2110.07202*.

[5] G. Xu, H. Liu, L. Tong, and T. Kailath, "A least-squares approach to blind channel identification," *IEEE Trans. Signal Process.*, vol. 43, no. 12, pp. 2982–2993, Dec. 1995.

[6] E. Moulines, P. Duhamel, J. Cardoso, and S. Mayrargue, "Subspace methods for the blind identification of multichannel FIR filters," *IEEE Trans. Signal Process.*, vol. 43, no. 2, pp. 516–525, Feb. 1995.

[7] J. Gao, E. Ertin, S. Kumar, and M. al'Absi, "Contactless sensing of physiological signals using wideband RF probes," in *Proc. Asilomar Conf. Signals, Syst. Comput.*, 2013, pp. 86–90.

[8] B. C. Civek, N. Sugavanam, S. Baskar, and E. Ertin, "Blind deconvolution methods for estimation of multilayer tissue profiles with ultrawideband radar," in *Proc. IEEE Radar Conf.*, 2019, pp. 1–6.

[9] S. Choudhary and U. Mitra, "Sparse blind deconvolution: What cannot be done," in *Proc. IEEE Int. Symp. Inf. Theory*, 2014, pp. 3002–3006.

[10] Ç. Bilen, G. Puy, R. Gribonval, and L. Daudet, "Convex optimization approaches for blind sensor calibration using sparsity," *IEEE Trans. Signal Process.*, vol. 62, no. 18, pp. 4847–4856, Sep. 2014.

[11] Y. Li, K. Lee, and Y. Bresler, "Blind gain and phase calibration via sparse spectral methods," *IEEE Trans. Inf. Theory*, vol. 65, no. 5, pp. 3097–3123, May 2019.

[12] L. Wang and Y. Chi, "Blind deconvolution from multiple sparse inputs," *IEEE Signal Process. Lett.*, vol. 23, no. 10, pp. 1384–1388, Oct. 2016.

[13] A. Ahmed, B. Recht, and J. Romberg, "Blind deconvolution using convex programming," *IEEE Trans. Inf. Theory*, vol. 60, no. 3, pp. 1711–1732, Mar. 2014.

[14] S. Chaudhuri, R. Velmurugan, and R. Rameshan, "Blind deconvolution methods: A review," in *Blind Image Deconvolution: Methods and Convergence*. Cham, Switzerland: Springer International Publishing, 2014, pp. 37–60.

[15] S. Ling and T. Strohmer, "Self-calibration and biconvex compressive sensing," *Inverse Problems*, vol. 31, no. 11, Sep. 2015, Art. no. 115002.

[16] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*. New York, NY, USA: Springer, 2004.

[17] W. Gilks, S. Richardson, and D. Spiegelhalter, *Markov Chain Monte Carlo in Practice*. London, U.K.: Chapman & Hall/CRC Interdisciplinary Statistics. Taylor & Francis, 1995.

[18] M. Lavielle, "Bayesian deconvolution of Bernoulli-Gaussian processes," *Signal Process.*, vol. 33, no. 1, pp. 67–79, 1993.

[19] C. Soussen, J. Idier, D. Brie, and J. Duan, "From Bernoulli-Gaussian deconvolution to sparse signal restoration," *IEEE Trans. Signal Process.*, vol. 59, no. 10, pp. 4572–4584, Oct. 2011.

[20] S. Geman and D. Geman, "Stochastic relaxation, gibbs distributions, and the Bayesian restoration of images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-6, no. 6, pp. 721–741, Nov. 1984.

[21] R. M. Neal, "MCMC using Hamiltonian dynamics," *Handbook Markov Chain Monte Carlo*, vol. 54, pp. 113–162, 2010.

[22] L. Chaari, J.-Y. Tourneret, C. Chaux, and H. Batatia, "A hamiltonian Monte Carlo method for non-smooth energy sampling," *IEEE Trans. Signal Process.*, vol. 64, no. 21, pp. 5585–5594, Nov. 2016.

[23] S. Bourguignon and H. Carfantan, "Bernoulli-Gaussian spectral analysis of unevenly spaced astrophysical data," in *Proc. IEEE/SP 13th Workshop Stat. Signal Process.*, 2005, pp. 811–816.

[24] D. Ge, J. Idier, and E. L. Carpentier, "Enhanced sampling schemes for MCMC based blind Bernoulli-Gaussian deconvolution," *Signal Process.*, vol. 91, no. 4, pp. 759–772, 2011.

[25] C.-Y. Chi and J. Mendel, "Improved maximum-likelihood detection and estimation of Bernoulli-Gaussian processes (corresp.)," *IEEE Trans. Inf. Theory*, vol. IT-30, no. 2, pp. 429–435, Mar. 1984.

[26] J. S. Liu, W. H. Wong, and A. Kong, "Covariance structure of the Gibbs sampler with applications to the comparisons of estimators and augmentation schemes," *Biometrika*, vol. 81, no. 1, pp. 27–40, 1994.

[27] D. A. V. Dyk and T. Park, "Partially collapsed Gibbs samplers: Theory and methods," *J. Amer. Stat. Assoc.*, vol. 103, no. 482, pp. 790–796, 2008.

[28] L. Chaari, J.-Y. Tourneret, and H. Batatia, "Sparse Bayesian regularization using Bernoulli-Laplacian priors," in *Proc. 21st Eur. Signal Process. Conf.*, 2013, pp. 1–5.

[29] L. Chaari, H. Batatia, and J.-Y. Tourneret, "Sparse Bayesian image restoration with linear operator uncertainties with application to EEG signal recovery," in *Proc. 2nd Middle East Conf. Biomed. Eng.*, 2014, pp. 139–142.

[30] Z. Zhang, S. Wang, D. Liu, and M. I. Jordan, "EP-GIG priors and applications in Bayesian sparse learning," *J. Mach. Learn. Res.*, vol. 13, no. 65, pp. 2031–2061, 2012.

[31] D. F. Andrews and C. L. Mallows, "Scale mixtures of normal distributions," *J. Roy. Stat. Society. Ser. B. (Methodol.)*, vol. 36, no. 1, pp. 99–102, 1974.

[32] F. Caron and A. Doucet, "Sparse Bayesian nonparametric regression," in *Proc. 25th Int. Conf. Mach. Learn.*, New York, NY, USA, 2008, pp. 88–95.

[33] P. J. Brown and J. E. Griffin, "Inference with normal-gamma prior distributions in regression problems," *Bayesian Anal.*, vol. 5, no. 1, pp. 171–188, 2010.

[34] S. Yildirim, A. T. Cemgil, M. Aktar, Y. Ozakin, and A. Ertuzun, "A Bayesian deconvolution approach for receiver function analysis," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 12, pp. 4151–4163, Dec. 2010.

[35] A. Mohammad-Djafari, "Bayesian approach with prior models which enforce sparsity in signal and image processing," *EURASIP J. Adv. Signal Process.*, vol. 2012, 2012, Art. no. 52.

[36] L. Yang, J. Fang, and H. Li, "Sparse Bayesian dictionary learning with a Gaussian hierarchical model," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2016, pp. 2564–2568.

[37] G. Kail, J. Tourneret, F. Hlawatsch, and N. Dobigeon, "Blind deconvolution of sparse pulse sequences under a minimum distance constraint: A partially collapsed Gibbs sampler method," *IEEE Trans. Signal Process.*, vol. 60, no. 6, pp. 2727–2743, Jun. 2012.

[38] F. Hlawatsch, *Time-Frequency Analysis and Synthesis of Linear Signal Spaces: Time-Frequency Filters, Signal Detection and Estimation, and Range-Doppler Estimation*. Boston, MA, USA: Kluwer Academic Publishers, 1998.

[39] W. Gilks, S. Richardson, and D. Spiegelhalter, "Introducing Markov Chain Monte Carlo," in *Markov Chain Monte Carlo in Practice*, Boca Raton, FL, USA: CRC Press, 1996.

[40] R. M. Neal, "Slice sampling," *Ann. Statist.*, vol. 31, no. 3, pp. 705–767, 2003.

[41] G. O. Roberts and S. K. Sahu, "Updating schemes, correlation structure, blocking and parameterization for the Gibbs sampler," *J. Roy. Stat. Society: Ser. B. (Statist. Methodol.)*, vol. 59, no. 2, pp. 291–317, 1997.

[42] C. Gilavert, S. Moussaoui, and J. Idier, "Efficient Gaussian sampling for solving large-scale inverse problems using MCMC," *IEEE Trans. Signal Process.*, vol. 63, no. 1, pp. 70–80, Jan. 2015.

[43] Y. Marnissi, E. Chouzenoux, A. Benazza-Benyahia, and J.-C. Pesquet, "An auxiliary variable method for Markov chain Monte Carlo algorithms in high dimension," *Entropy*, vol. 20, no. 2, 2018, Art. no. 110.

[44] M. Vono, N. Dobigeon, and P. Chainais, "High-dimensional Gaussian sampling: A review and a unifying approach based on a stochastic proximal point algorithm," *SIAM Rev.*, vol. 64, no. 1, pp. 3–56, 2022.

[45] C. Labat and J. Idier, "Sparse blind deconvolution accounting for time-shift ambiguity," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2006, pp. III–III.

[46] S. P. Brooks and A. Gelman, "General methods for monitoring convergence of iterative simulations," *J. Comput. Graphical Statist.*, vol. 7, no. 4, pp. 434–455, 1998.