SPARSE DECONVOLUTION: COMPARISON OF STATISTICAL AND DETERMINISTIC APPROACHES

Sébastien Bourguignon[*], *Charles Soussen*[†], *Hervé Carfantan*[‡], *Jérôme Idier*[††]

[*] University of Nice Sophia Antipolis, CNRS, Observatoire de la Côte d'Azur, F-06304 Nice, France
[†]CRAN, UMR 7039, Nancy-University, CNRS, F-54506 Vandœuvre-lès-Nancy, France
[‡]Université de Toulouse, UPS-OMP/CNRS, IRAP, F-31400 Toulouse, France
[††]IRCCyN (UMR CNRS 6597), F-44321 Nantes, France

ABSTRACT

Sparse spike train deconvolution is a classical inverse problem which gave rise to many deterministic and stochastic algorithms since the mid-80's. In the past decade, sparse approximation has been an intensive field of research, leading to the development of a number of algorithms including greedy strategies and convex relaxation methods. Spike train deconvolution can be seen as a specific sparse approximation problem, where the observation matrix contains highly correlated columns and where the focus is set on the exact recovery of the spike locations. The objective of this paper is to evaluate the performance of algorithms proposed in both fields in terms of detection statistics, with Monte-Carlo simulations of spike deconvolution problems.

Index Terms— Sparse spike train deconvolution; detection statistics; greedy algorithms; convex relaxation; Bernoulli-Gaussian model; MCMC algorithms.

1. INTRODUCTION

Spike train deconvolution occurs in a variety of areas such as geophysics, astronomy, and non destructive evaluation [1, Chap. 5]. In these problems, the separation between spikes may be arbitrarily small – in particular, smaller than the size of the convolution kernel – so that the convolved spikes overlap, making the deconvolution problem a non trivial one. Historically, spike train deconvolution has been addressed as an inverse problem where the peaky shape of the solution is enforced by two kinds of strategies. The first – deterministic – one minimizes a data-misfit criterion, penalized by a sparsity-enhancing function [2]. A second well-known approach considers a statistical description of sparsity *via*, *e.g.,* a Bernoulli-Gaussian model combined with deterministic optimization [3] or stochastic sampling [4].

Obviously, spike deconvolution can be linked with the now famous sparse approximation principle [5]. The underlying idea is to approximate a signal with few elementary *atoms* taken from a redundant *dictionary* [6, 7]. Finding the sparsest approximation under some limited error is generally an intractable combinatorial problem. The main sub-optimal methods can be divided into two groups, namely greedy selection algorithms [6, Chap. 3] which yield sparsity by construction, and convex relaxation methods. In particular, using the ℓ^1-norm results in Basis Pursuit denoising (BPDN) for which efficient algorithms are available [7]. In practice, the choice of a relevant algorithm depends on the structure of the dictionary and the desired degree of sparsity. Typically, when the dictionary is close to orthogonal, simple algorithms from both groups are efficient. For deconvolution, however, the dictionary is *imposed* by the instrument response and atoms can be highly correlated. Hence, optimality of usual sparse approximation algorithms is generally not guaranteed.

In this paper, we compare sparsity-enhancing methods issued from both deconvolution and sparse approximation domains. We consider sparsity in the strict sense, *i.e.,* a large number of spike amplitudes are *exactly* zero. Deterministic and stochastic algorithms are evaluated in terms of detection performance on simulated noisy deconvolution problems. Algorithmic accuracy is expressed in terms of correct detection and false alarm rates by performing Monte-Carlo simulations at several noise levels. Note that the evaluation of algorithms in terms of computational cost is out of the scope of this study. In Section 2, we detail the formalism of spike deconvolution and sparse approximation and highlight the differences between the two problems. Section 3 describes state-of-the-art deterministic algorithms and a stochastic approach based on a Monte-Carlo Markov Chain (MCMC) sampler. Finally, Section 4 is dedicated to simulation results and conclusions.

2. SPARSE APPROXIMATION AND SPARSE DECONVOLUTION

2.1. Sparse approximations for denoising

Sparsity-based denoising applications rely on the noisy model

$$\boldsymbol{y} = \mathbf{H}\boldsymbol{x} + \boldsymbol{\varepsilon}, \tag{1}$$

This work was partially supported by PPF-ISSO, University of Nice Sophia Antipolis.

where $\boldsymbol{y}$ stands for the noisy observations and $\mathbf{H}$ is a redundant dictionary chosen such as the noise-free signal ($\mathbf{H}\boldsymbol{x}$) has a sparse representation, *i.e.,* $\boldsymbol{x}$ has a small number of non-zero elements. Then, sparse approximation amounts to solving:

$$\min_{\boldsymbol{x}} \|\boldsymbol{y} - \mathbf{H}\boldsymbol{x}\|^2 + \lambda_0 \|\boldsymbol{x}\|_0 \quad (2)$$

where $\lambda_0 > 0$ controls the trade-off between a low sparsity measure $\|\boldsymbol{x}\|_0 = \text{Card}\{k, x_k \neq 0\}$ and a low approximation error.

2.2. Spike train deconvolution

In applied physics, many data acquisition processes can be modeled with a "convolution plus noise" model:

$$y_n = \sum_{k=0}^{K} h_k x_{n-k} + \varepsilon_n, \quad n = 1, \ldots, N$$

where h is the (finite) impulse response of the system and ε represents noise and model errors. In the following, we consider the *full* convolution model, that is, the boundary condition $x_m = 0$ for $m \notin \{1, \ldots, N-K\}$. With matrix-vector notations, the observation model takes the form (1), where $\boldsymbol{y}$ collects the N observations, $\boldsymbol{x}$ gathers the $M = N - K$ samples to be retrieved, and $\mathbf{H}$ is the $N \times M$ Toeplitz matrix whose columns are shifted versions of the impulse response.

Under the assumption that noise samples ε_n are independent and identically distributed (i.i.d.) Gaussian with variance σ_ε^2, the likelihood function reads:

$$\mathcal{L}(\boldsymbol{y}; \boldsymbol{x}, \sigma_\varepsilon^2) = g_N(\boldsymbol{y} - \mathbf{H}\boldsymbol{x}, \sigma_\varepsilon^2 \mathbf{I}_N), \quad (3)$$

where $g_N(\boldsymbol{u}, \boldsymbol{\Sigma})$ denotes the N-variate Gaussian centered distribution with covariance $\boldsymbol{\Sigma}$ and $\mathbf{I}_N$ is the identity matrix of size N. The maximum likelihood estimator reads as the unconstrained least-squares solution $(\mathbf{H}^t\mathbf{H})^{-1}\mathbf{H}^t\boldsymbol{y}$. This solution is known to perform poorly, however, because $\mathbf{H}$ is badly conditioned.

Now, consider the problem of *spike train* deconvolution, *i.e.,* $\boldsymbol{x}$ is composed of many zeros. In the Bayesian setting, maximum *a posteriori* (MAP) estimation amounts to solving:

$$\min_{\boldsymbol{x}} \frac{\|\boldsymbol{y} - \mathbf{H}\boldsymbol{x}\|^2}{2\sigma_\varepsilon^2} + \mu \sum_m \varphi(|x_m|) \text{ for some } \mu > 0, \quad (4)$$

where the prior distribution $p(x_m) \propto \exp(-\mu\varphi(|x_m|))$ enforces sparsity if φ is a nondecreasing function on $\mathbb{R}^+$ that is not differentiable at 0 [8]. As an alternative to priors relying on a continuous function, the Bernoulli-Gaussian (BG) model explicitly expresses sparsity by introducing a point mass distribution at zero [3]. BG estimation can be done whether by computing the MAP estimate, or by more sophisticated methods involving MCMC techniques [4] (see § 3.2 for details).

Both (2) and (4) formulate penalized least-squares problems. However, the discrete valued ℓ^0 norm cannot be associated to a well-defined sparsity prior, although it corresponds to a limit case of a Bernoulli-Gaussian model [5].

2.3. Structure of the observation matrix

The essential difference between spike train deconvolution and usual sparse approximation concerns matrix $\mathbf{H}$. In sparse approximation, the *design* of a close-to-orthogonal redundant dictionary is crucial to guarantee the ability of algorithms to recover the sparsest solution [9]. In spike deconvolution, matrix $\mathbf{H}$ is badly conditioned since consecutive columns of $\mathbf{H}$ are highly correlated. More precisely, vector $\boldsymbol{x}$ actually results from the discretization of a continuous process at the same sampling rate as the observation $\boldsymbol{y}$. If $\boldsymbol{x}$ was sampled with a finer sampling step, then the conditioning of $\mathbf{H}$ would be even worse. Note that the mutual coherence of $\mathbf{H}$, often used as a measure of quasi-orthogonality for sparse approximation dictionaries [9], also increases as the sampling step decreases.

3. DETERMINISTIC AND STOCHASTIC ALGORITHMS FOR COMPARISON

We briefly describe here algorithms that will be compared for spike train deconvolution. Sparse algorithms include greedy search and ℓ^1 penalization algorithms, while the deterministic and stochastic deconvolution algorithms are based on the Bayesian estimation of Bernoulli-Gaussian models.

3.1. Sparse approximation methods: greedy pursuit and BPDN

Greedy algorithms are acknowledged iterative procedures composed of two steps per iteration: the selection of an atom improving the approximation of the data, and the update of the solution as a linear combination of selected atoms. In this study, we consider the *Orthogonal Matching Pursuit* (OMP) and *Orthogonal Least-Squares* (OLS) algorithms. OLS operates a more sophisticated selection step than OMP and therefore often performs better, but at the price of a higher computational cost. For a thorough description of OMP and OLS, we refer the reader to [10].

We also consider the well known BPDN problem [7] resulting from the relaxation of problem (2) with the ℓ^1-norm:

$$\min_{\boldsymbol{x}} \|\boldsymbol{y} - \mathbf{H}\boldsymbol{x}\|^2 + \lambda_1 \|\boldsymbol{x}\|_1, \text{ with } \|\boldsymbol{x}\|_1 = \sum_m |x_m|. \quad (5)$$

3.2. Algorithms based on the Bernoulli-Gaussian model

The Bernoulli-Gaussian (BG) model is a sparsity-oriented hierarchical probabilistic model which introduces Bernoulli variables $\boldsymbol{q} \in \{0,1\}^M$, where $q_m = 1$ indicates a spike at index m. Spikes are supposed i.i.d. with $\Pr(q_m{=}1) = \rho$. Then, the amplitudes of non-zero components are supposed i.i.d. centered Gaussian with variance σ_x^2: $x_m|q_m \sim g_1(0, q_m\sigma_x^2)$ where by extension, $g_1(0,0)$ stands for the Dirac distribution at 0. Estimation can then be performed in the Bayesian

setting, based on the posterior distribution:

$$p(\boldsymbol{q}, \boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\theta}) \propto \mathcal{L}(\boldsymbol{y}; \boldsymbol{q}, \boldsymbol{x}, \boldsymbol{\theta}) p(\boldsymbol{q}, \boldsymbol{x}|\boldsymbol{\theta}) \quad (6)$$

where $\mathcal{L}(\boldsymbol{y}; \boldsymbol{q}, \boldsymbol{x}, \boldsymbol{\theta})$ is the likelihood (3), $p(\boldsymbol{q}, \boldsymbol{x}|\boldsymbol{\theta})$ is the prior distribution:

$$\begin{aligned} p(\boldsymbol{q}, \boldsymbol{x}|\boldsymbol{\theta}) &= P(\boldsymbol{q}|\boldsymbol{\theta}) p(\boldsymbol{x}|\boldsymbol{q}, \boldsymbol{\theta}) \\ &= \rho^{\|\boldsymbol{q}\|_0} (1-\rho)^{M-\|\boldsymbol{q}\|_0} g_M \left(\mathbf{0}, \sigma_x^2 \mathrm{diag}(\boldsymbol{q})\right) \end{aligned}$$

and $\boldsymbol{\theta} = [\rho, \sigma_x^2, \sigma_\varepsilon^2]$.

Maximizing (6) is essentially a combinatorial problem (in $\boldsymbol{q}$) which is usually addressed with sub-optimal strategies. An acknowledged algorithm is the Single Most Likely Replacement (SMLR) method [3] where each iteration whether includes or removes one element in the support – the chosen replacement is the one that most increases the posterior distribution. Recently, the Single Best Replacement (SBR) algorithm was proposed as an SMLR extension in the limit case $\sigma_x = +\infty$. It can also be viewed as a *forward-backward* extension of OLS, which includes removing steps, aimed toward the minimization of (2) [5].

As an alternative to optimization, one may resort to MCMC methods to avoid problems caused by local minima [4]. Moreover, MCMC methods allow *unsupervised* estimation by jointly sampling $(\boldsymbol{q}, \boldsymbol{x})$ and $\boldsymbol{\theta}$ for a negligible additional cost. Here, we use the implementation based on the partially marginalized Gibbs sampler recently proposed in [4]. ρ is supposed uniformly distributed in $[0, 1]$ and the low-informative conjugate prior $p(\sigma_\varepsilon^2) \propto 1/\sigma_\varepsilon^2$ is set on σ_ε^2 [4]. In order to avoid amplitude shrinkage due to the Gaussian prior on $x_m|q_m = 1$, one would ideally use $\sigma_x = +\infty$. This leads, however, to degenerate distributions in the MCMC sampler. In all our simulations, σ_x is set to 1 since this value is large enough compared to the range of true amplitudes (see Figure 1). Once MCMC samples are obtained, we perform BG estimation by delivering two indicators. First, we compute the posterior mean of the Bernoulli variables, say $\widehat{\boldsymbol{q}} \in [0, 1]^M$, which gives posterior detection probabilities at each location. Then, since averaging produces many small-valued probabilities, a sparse solution is obtained by estimating spike locations at indices where $\widehat{\boldsymbol{q}}$ exceeds some threshold τ. Details related to the empirical setting of τ will be given in the simulation section below.

4. SIMULATION RESULTS AND CONCLUSIONS

Simulations are run using 300-point sequences $\boldsymbol{x}$ with 19 spikes, similar to that introduced in [3]: see Figure 1 for a typical signal together with the impulse response. This impulse response is known to yield a very badly conditioned matrix $\mathbf{H}$ [1, Chap. 5]. It yields a high mutual coherence as well. The OMP, OLS, BPDN, SBR and BG-MCMC algorithms are run for 100 simulated data sets, with different random realizations of both spike locations and noise, according to uniform and Gaussian distributions, respectively.

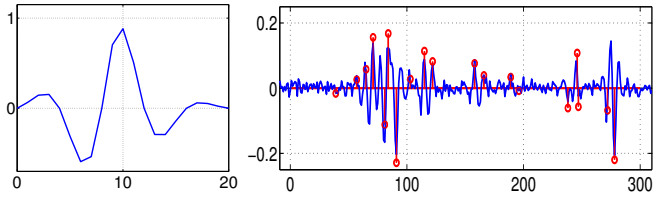


Fig. 1. Impulse response (left) and a realization of the spike process (right): true impulsions $\boldsymbol{x}$ (red circles) and convolved noisy data $\boldsymbol{y}$ (blue line, SNR = 15 dB).

4.1. MCMC convergence monitoring

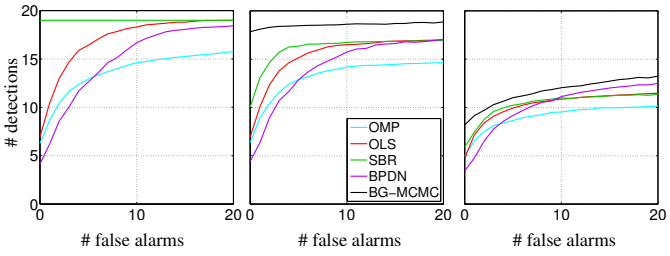
The MCMC algorithm requires care to check that convergence is reached. For any simulated data set, we set $T = 10$ different random initializations, from each of which 5000 MCMC iterations are run. For the t-th sequence, the posterior mean estimate $\widehat{\boldsymbol{q}}^{(t)}$ is computed by averaging the last 1000 MCMC samples. Finally, our convergence test relies on computing the empirical variance of $\widehat{\boldsymbol{q}}_i$ from all T estimates $\widehat{\boldsymbol{q}}_i^{(t)}$ for any location i: the simulations for which all $\widehat{\boldsymbol{q}}_i$ have a variance lower than 10^{-2} are kept, and the BG solution is defined as the average of $\widehat{\boldsymbol{q}}^{(t)}$. The other simulations are discarded.

4.2. Experimental results

We first study the ability of algorithms to locate the true solution support. OMP, OLS and SBR are run until all true spikes are detected with possible wrong detections. The BPDN solution is computed for decreasing values of λ_1 with the homotopy continuation implementation [7]. Similarly, in BG-MCMC estimation, parameter τ is being gradually decreased in order to yield solutions at different sparsity levels. Note that for SNR = 50 dB, posterior distributions involved in BG-MCMC sampling were so sharp that they led to numerical problems. Consequently, no result was obtained in this case. Figure 2 plots the average number of correct detections *vs.* the number of false alarms for different signal-to-noise ratios, with $\mathrm{SNR}_{\mathrm{dB}} = 10 \log\left(\|\mathbf{H}\boldsymbol{x}\|^2 / (N\sigma_\varepsilon^2)\right)$. Note that algorithms may lead to several solutions with the same number of false alarms, but with a different number of correct detections. In this case, the best detection score is kept.

We now make a quantitative comparison corresponding to solutions obtained for the same approximation error $E = \|\boldsymbol{y} - \mathbf{H}\widehat{\boldsymbol{x}}\|$, where $\widehat{\boldsymbol{x}}$ is the estimated sparse sequence. To this aim, all algorithms are run for decreasing sparsity levels (hence, for decreasing E) until E is lower than a threshold $E_{\min}$. This threshold is set relative to a statistical χ^2-test, which amounts to deciding if the approximation error corresponds to noise. Suppose that σ_ε^2 is known. Then, $\|\boldsymbol{\varepsilon}\|^2 / \sigma_\varepsilon^2$ follows a χ_N^2 distribution. Hence, solutions are selected with $E \simeq E_{\min}$, where:

$$E_{\min}/\sigma_\varepsilon^2 = \nu, \text{ where } Pr(u \leq \nu | u \sim \chi_N^2) = \eta \quad (7)$$



(a) SNR = 50 dB (b) SNR = 20 dB (c) SNR = 5 dB

Fig. 2. Average number of detected spikes *vs*. false alarms reached by OMP, OLS, SBR, BPDN and BG-MCMC, for three different noise levels.

for a given probability η, that we fix to $\eta = 95\%$. As BPDN and BG-MCMC solutions suffer from amplitude bias, amplitudes corresponding to each support are re-estimated by least-squares before computing E. Table 1 gives the average number of detections and false alarms for different SNRs. Note that numerical values in Table 1 are averaged both in detection and in false alarms, for the tuning obtained by (7). Since the plots in Figure 2 are obtained by fixing the number of false alarms, the results slightly differ from those of Table 1. Results are commented and conclusions are drawn in § 4.3.

4.3. Conclusions

Let us focus on Figure 2. We remark that SBR always performs at least as good as OLS, which always outperforms OMP. However, the difference between OLS and SBR vanishes as the SNR increases. We also note that, in the almost noise-free case (SNR = 50 dB), SBR is the only algorithm that *always* retrieves the true support (#D = 19, #FA = 0). BPDN has the worst detection rate for low #FA, but it performs better than OMP when #FA increases. BPDN can also achieve better detection scores than OLS and SBR for high #FA, especially at low SNR. Finally, BG-MCMC always gives the best results for noisy data, both in terms of detection and false alarms.

When tuning the algorithms so that the approximation error equals the noise level (see Table 1) we note that they all yield approximately the same detection rate, but they largely differ in terms of false alarms. In particular, in noisy simulations, BG-MCMC yields very few false alarms compared to other methods, while BPDN yields much higher #FA.

Obviously, in practical cases, these results must be linked with the computational cost of algorithms: generally speaking, one has BG-MCMC $\gg$ SBR $\simeq$ OLS $>$ BPDN $\simeq$ OMP. If the noise is very low, then SBR is the best strategy, at the expense of a higher runtime than other deterministic algorithms. For higher noise levels, BG-MCMC is definitely the best choice at the price of a significantly increased computation cost and of critical issues regarding convergence monitoring. As shown in the last row of Table 1, the number of simulations for which convergence was not obtained reaches

SNR	5		10		20		50	
	#D	#FA	#D	#FA	#D	#FA	#D	#FA
OMP	7.6	2.7	10.7	4.0	14.2	10.6	18.5	42.2
OLS	7.9	2.0	11.2	2.6	15.2	4.3	**19.0**	**5.8**
SBR	8.0	1.8	11.3	2.3	15.6	2.4	**19.0**	**0.0**
BPDN	8.4	4.2	11.5	6.1	15.7	9.2	18.8	18.7
BG-MCMC	**8.7**	**1.2**	**12.5**	**1.9**	**17.3**	**0.2**	-	-
(% valid simulations)	(71)		(55)		(51)		(0)	

Table 1. Average number of correct detections (#D) and of false alarms (#FA) as a function of SNR. All algorithms are tuned in order to yield the same approximation error (7). Bold figures highlight the most significant results.

almost 50 %. Note that this problem occurs more frequently with increasing SNR: since for higher SNR more confidence is given to the data, the posterior distributions are sharper, hence MCMC samplers encounter additional difficulties to escape from local modes. In practice, if computational resources do not allow to run BG-MCMC, the faster SBR and OLS algorithms can be used. Last, when resources are even more limited, *e.g.,* for real-time processing, BPDN or OMP shall be chosen, whether a relatively high number of false alarms is allowed or not.

5. REFERENCES

[1] J. Idier, Ed., *Bayesian Approach to Inverse Problems*, ISTE Ltd and John Wiley & Sons Inc, Apr. 2008.

[2] H. L. Taylor, S. C. Banks, and J. F. McCoy, "Deconvolution with the ℓ_1 norm," *Geophysics*, vol. 44, no. 1, pp. 39–52, 1979.

[3] J. Kormylo and J. M. Mendel, "Maximum-likelihood detection and estimation of Bernoulli-Gaussian processes," *IEEE Transactions on Information Theory*, vol. 28, pp. 482–488, 1982.

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

[5] C. Soussen, J. Idier, D. Brie, and J. Duan, "From Bernoulli-Gaussian deconvolution to sparse signal restoration," Tech. Rep., CRAN, Nancy, 2010.

[6] A. J. Miller, *Subset Selection in Regression*, Chapman and Hall, London, UK, 2nd edition, Apr. 2002.

[7] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*, Springer, Aug. 2010.

[8] P. Moulin and J. Liu, "Analysis of multiresolution image denoising schemes using generalized Gaussian and complexity priors," *IEEE Trans. Inf. Theory*, vol. 45, pp. 909–919, Apr. 1999.

[9] J.-J. Fuchs, "On sparse representations in arbitrary redundant bases," *IEEE Trans. Inf. Theory*, vol. 50, no. 6, pp. 1341–1344, Jun. 2004.

[10] T. Blumensath and M. E. Davies, "On the difference between Orthogonal Matching Pursuit and Orthogonal Least Squares," Tech. Rep., University of Edinburgh, Mar. 2007.