Sparse Spike Train Deconvolution Using the Hunt Filter and a Thresholding Method

Vincent Mazet, David Brie, and Cyrille Caironi

***Abstract*—A new deconvolution method of sparse spike trains is presented. It is based on the coupling of the Hunt filter with a thresholding. We show that a good model for the probability density function of the Hunt filter output is a Gaussian mixture, from which we derive the threshold that minimizes the probability of errors. Based on an interpretation of the method as a maximum *a posteriori* (MAP) estimator, the hyperparameters are estimated using a joint MAP approach. Simulations show that this method performs well at a very low computation time.**

***Index Terms*—Bernoulli–Gaussian, Hunt filter, sparse spike train deconvolution, thresholding.**

I. Introduction

We consider the problem of sparse spike train deconvolution which is classically stated as follows (in the sequel, $\boldsymbol{v}_i$ will represent the ith element of vector $\boldsymbol{v}$ and $\boldsymbol{M}_{i,j}$ the element (i,j) of the matrix $\boldsymbol{M}$): given some observation sequence $\boldsymbol{y} \in \mathbb{R}^N$, find the sparse spike train sequence $\boldsymbol{x} \in \mathbb{R}^N$ such as

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n}$$

where $\boldsymbol{H} \in \mathbb{R}^{N \times N}$ is a known impulse response matrix, and $\boldsymbol{n} \in \mathbb{R}^N$ a noise term assumed to be Gaussian and independent, identically distributed (i.i.d.) whose probability density function (pdf) is $p(\boldsymbol{n}) = \mathcal{N}(\boldsymbol{0}, r_n \boldsymbol{I}_N)$, r_n being the noise variance. To handle the sparse nature of $\boldsymbol{x}$, it is modeled as an i.i.d. Bernoulli–Gaussian (BG) sequence [1], [2], i.e.,

$$\forall k, \quad p(\boldsymbol{x}_k) = \lambda \mathcal{N}(0, r_x) + (1-\lambda)\delta(\boldsymbol{x}_k)$$

where r_x is the pulse variance, $\lambda \ll 1$ is the Bernoulli parameter (ratio between pulse and sample number), and δ is the Dirac mass centered on zero. In the sequel, $\boldsymbol{\theta} = \{\lambda, r_x, r_n\}$ gathers the hyperparameters of the problem.

Among the many applications of sparse spike deconvolution, we would like to mention partial discharge analysis, which has motivated this work [3]. The final goal was to develop algorithms that can be embedded to monitor the insulation of high-voltage AC motors. In addition, in this particular application, one has to process very large signals (typically $N = 2^{16}$ points). These are the reasons that have led us to develop an approach fast and simple to implement needing only low memory space.

On the one hand, combinatory algorithms like single mst likely replacement (SMLR) [1], [2], iterated window maximization (IWM) [4], or others given in [5] are well-suited to the problem. Although they give very satisfactory estimations, they are very slow and difficult to implement on embedded systems.

On the other hand, the Hunt filter [6] is very fast and simple to implement, but it does not yield satisfactory results, mainly because of the underlying prior, which assumes that the restored signal is Gaussian. To overcome this drawback, we propose to associate the Hunt filter with a thresholding procedure to actually restore a sparse spike train signal resulting in the so-called Hunt–threshold (HT) method.

The letter is organized as follows. Section II presents the HT method: first, the Hunt filter is briefly recalled; then, we derive the approximate pdf of the Hunt filter output from which we determine the threshold that minimizes the probability of errors. In Section III, based on an interpretation of the HT method as a maximum *a posteriori* (MAP) estimator, we propose to estimate the hyperparameters using a joint MAP (JMAP) approach [7]. In Section IV, the performances of the HT method are evaluated and compared to those of the SMLR of [1]. Finally, Section V gives the conclusions and perspectives of this letter.

II. HT Method

A. Hunt Filter

The Hunt filter [6] results from the Phillips and Twomey criterion minimization, which corresponds to the discrete time formulation of the Tikhonov criterion

$$\mathcal{J}_{\mathrm{PT}} = (\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x})^T(\boldsymbol{y} - \boldsymbol{H}\boldsymbol{x}) + \alpha \boldsymbol{x}^T \boldsymbol{D}^T \boldsymbol{D}\boldsymbol{x}$$

where $\boldsymbol{D} = \boldsymbol{I}_N$ so as to favor the restoration of zero-value signal.

Minimizing $\mathcal{J}_{\mathrm{PT}}$ yields a first estimation $\tilde{\boldsymbol{x}}$ of $\boldsymbol{x}$

$$\tilde{\boldsymbol{x}} = \boldsymbol{G}\boldsymbol{y} \quad \text{where} \quad \boldsymbol{G} = (\boldsymbol{H}^T\boldsymbol{H} + \alpha \boldsymbol{I}_N)^{-1}\boldsymbol{H}^T. \tag{1}$$

The matrix $\boldsymbol{H}$ being supposed circulant, its (discrete) Fourier transform is a diagonal matrix. Thus, the algorithm may be efficiently implemented using the fast Fourier transform, $\tilde{\boldsymbol{x}}$ being then obtained by an inverse (discrete) Fourier transform. This results in the so-called Hunt filter.

A Bayesian interpretation [5] shows that the Hunt filter is equivalent to a MAP approach with a Gaussian prior: it yields $\alpha = r_n/r_x$, where r_x is the variance of the Gaussian signal to restore. In the sequel, we will also take $\alpha = r_n/r_x$ with r_x

Manuscript received April 17, 2003; revised July 3, 2003. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Dimitris A. Pados.

V. Mazet and D. Brie are with the Centre de Recherche en Automatique de Nancy, Centre National de la Recherche Scientifique, UMR 7039, Université Henri Poincaré, 54506 Vandoeuvre-lès-Nancy Cedex, France (e-mail: vincent.mazet@cran.uhp-nancy.fr; david.brie@cran.uhp-nancy.fr).

C. Caironi is with Alstom Moteurs, 54250 Champigneulles, France (e-mail: cyrille.caironi@tde.alstom.com).

Digital Object Identifier 10.1109/LSP.2004.826655

the variance of the pulses of the BG sequence to favor the pulse amplitude estimation.

B. Approximate PDF of $\tilde{\boldsymbol{x}}$

The pdf $p(\tilde{\boldsymbol{x}})$ of $\tilde{\boldsymbol{x}} = \boldsymbol{G}\boldsymbol{y} = \boldsymbol{G}\boldsymbol{H}\boldsymbol{x} + \boldsymbol{G}\boldsymbol{n}$ is needed to calculate the threshold t (Section II-C).

As $(\boldsymbol{G}\boldsymbol{n})_k = \sum_{i=1}^{N} \boldsymbol{G}_{k,i}\boldsymbol{n}_i$, we have

$$p\left((\boldsymbol{G}\boldsymbol{n})_k\right) = p\left(\sum_{i=1}^{N} \boldsymbol{G}_{k,i}\boldsymbol{n}_i\right) = \overset{\star}{\prod_{i=\{1,\ldots,N\}}} p(\boldsymbol{G}_{k,i}\boldsymbol{n}_i).$$

$\overset{\star}{\prod}$ is introduced to improve the readibility of the letter

$$\overset{\star}{\prod_{i=\{1,\ldots,N\}}} f_i(x) = [f_1 \star \ldots \star f_N](x).$$

Since $p(\boldsymbol{n}_k) = \mathcal{N}(0, r_n)$ and $\mathcal{N}(0,\sigma_1^2) \star \mathcal{N}(0,\sigma_2^2) = \mathcal{N}(0,\sigma_1^2 + \sigma_2^2)$, we get

$$p\left((\boldsymbol{G}\boldsymbol{n})_k\right) = \mathcal{N}\left(0, r_n \sum_{i=1}^{N} \boldsymbol{G}_{k,i}^2\right). \tag{2}$$

Similarly, since $p(\boldsymbol{x}_k) = \lambda\mathcal{N}(0, r_x) + (1-\lambda)\delta(\boldsymbol{x}_k)$ and $(\boldsymbol{G}\boldsymbol{H}\boldsymbol{x})_k = \sum_{i=1}^{N} (\boldsymbol{G}\boldsymbol{H})_{k,i}\boldsymbol{x}_i$, we have

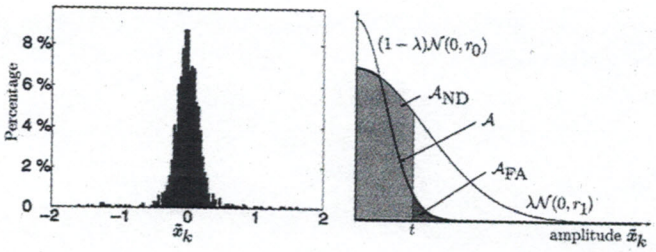
$$\begin{aligned} p\left((\boldsymbol{G}\boldsymbol{H}\boldsymbol{x})_k\right) &= \overset{\star}{\prod_{i=\{1,\ldots,N\}}} p\left((\boldsymbol{G}\boldsymbol{H})_{k,i}\boldsymbol{x}_i\right) \\ &= \overset{\star}{\prod_{i=\{1,\ldots,N\}}} \left[\lambda\mathcal{N}\left(0, r_x(\boldsymbol{G}\boldsymbol{H})_{k,i}^2\right) + (1-\lambda)\delta(\boldsymbol{x}_i)\right]. \end{aligned}$$

To simplify the problem, we now need two assumptions.

1) We suppose that the off-diagonal terms of $\boldsymbol{G}\boldsymbol{H}$ are negligible as compared to the diagonal terms. This is nothing but considering the values $(\boldsymbol{G}\boldsymbol{H}\boldsymbol{x})_k$ decorrelated. We note that the higher the SNR is (i.e., α decreases), the more valid this approximation is. Indeed, when α decreases, $\boldsymbol{G}\boldsymbol{H}$ tends to $\boldsymbol{I}_N$. With this assumption, we have $r_x(\boldsymbol{G}\boldsymbol{H})_{k,i} = 0 \; \forall \, k \neq i$. The validity of this approximation depends both on the SNR and the impulse response. Typically, for the impulse response used in this letter (see Section IV), the approximation is very realistic for a SNR greater than 15 dB. But if the frequency contents of the impulse response becomes more concentrated in low frequencies, the SNR should be greater;
2) We suppose that the diagonal terms of $\boldsymbol{G}\boldsymbol{G}^T$ and $\boldsymbol{G}\boldsymbol{H}\boldsymbol{H}^T\boldsymbol{G}^T$ are constant, i.e., we neglect the boundary effects.

Thus, considering that a Gaussian with zero variance is a Dirac mass, we get

$$p\left((\boldsymbol{G}\boldsymbol{H}\boldsymbol{x})_k\right) = \lambda\mathcal{N}\left(0, r_x(\boldsymbol{G}\boldsymbol{H})_{k,k}^2\right) + (1-\lambda)\delta(\boldsymbol{x}_k). \tag{3}$$



(a) (b)

Fig. 1. PDF of $\tilde{\boldsymbol{x}}$. (a) Histogram of $\tilde{\boldsymbol{x}}$ for 1024 samples. (b) Areas of false alarms and nondetections.

From (2) and (3), we obtain

$$\begin{aligned} p(\tilde{\boldsymbol{x}}_k) &= p\left((\boldsymbol{G}\boldsymbol{H}\boldsymbol{x})_k\right) \star p\left((\boldsymbol{G}\boldsymbol{n})_k\right) \\ &= \lambda\mathcal{N}\left(0, r_x(\boldsymbol{G}\boldsymbol{H})_{k,k}^2 + r_n\sum_{i=1}^{N}\boldsymbol{G}_{k,i}^2\right) \\ &\quad + (1-\lambda)\mathcal{N}\left(0, r_n\sum_{i=1}^{N}\boldsymbol{G}_{k,i}^2\right). \end{aligned}$$

As $r_n\sum_{i=1}^{N}\boldsymbol{G}_{k,i}^2 = r_n(\boldsymbol{G}\boldsymbol{G}^T)_{k,k}$ and $r_x(\boldsymbol{G}\boldsymbol{H})_{k,k}^2 = r_x\sum_{i=1}^{N}(\boldsymbol{G}\boldsymbol{H})_{k,i}^2 = r_x(\boldsymbol{G}\boldsymbol{H}\boldsymbol{H}^T\boldsymbol{G}^T)_{k,k}$, it follows that the pdf is a Gaussian mixture centered on zero

$$p(\tilde{\boldsymbol{x}}_k) = \lambda\mathcal{N}(0, r_1) + (1-\lambda)\mathcal{N}(0, r_0) \tag{4}$$

with

$$r_1 = r_x(\boldsymbol{G}\boldsymbol{H}\boldsymbol{H}^T\boldsymbol{G}^T)_{k,k} + r_n(\boldsymbol{G}\boldsymbol{G}^T)_{k,k} \tag{5}$$

$$r_0 = r_n(\boldsymbol{G}\boldsymbol{G}^T)_{k,k}, \qquad r_0 < r_1. \tag{6}$$

The histogram of $\tilde{\boldsymbol{x}}$ obtained on a 1024-sample signal [Fig. 1(a)] clearly shows that the Gaussian mixture is a good model for $p(\tilde{\boldsymbol{x}}_k)$.

C. Thresholding

The thresholding separates pulses [term $\lambda\mathcal{N}(0, r_1)$] from noise [term $(1-\lambda)\mathcal{N}(0, r_0)$]. The estimated signal may be expressed as $\widehat{\boldsymbol{x}}_k = \tilde{\boldsymbol{x}}_k$ if $|\tilde{\boldsymbol{x}}_k| > t$, $\widehat{\boldsymbol{x}}_k = 0$ otherwise. Performing such a (hard) thresholding, one can make two kinds of errors. *False alarms* correspond to wrongly detected pulses, while *nondetections* correspond to missed pulses. The threshold t is chosen to minimize the probability of error $\{p_{\mathrm{FA}} + p_{\mathrm{ND}}\}$ where p_{FA} and p_{ND} are the probabilities of false alarm and nondetection. Fig. 1(b) shows the positive part of the two Gaussians. The area $\mathcal{A}_{\mathrm{FA}}$ represents half the probability p_{FA} and $\mathcal{A}_{\mathrm{ND}}$ half the probability p_{ND}. It appears that $\mathcal{A}_{\mathrm{ND}} + \mathcal{A}_{\mathrm{FA}}$ is minimal and equal to $\mathcal{A}$ when t corresponds to the intersection point of the two Gaussians, which gives

$$t = \sqrt{\ln\left(\frac{\lambda}{1-\lambda}\sqrt{\frac{r_0}{r_1}}\right)\frac{2r_1r_0}{r_0 - r_1}}. \tag{7}$$

III. Hyperparameter Estimation

To address the hyperparameter estimation problem, we first need to interpret the HT method in a Bayesian framework. In [8] and [9], it is shown that the MAP estimator remains unchanged

for data varying in a neighborhood if and only if the prior pdf is nonsmooth (i.e., its derivative is discontinuous) in this neighborhood. This results in a thresholding effect of the MAP estimator under nonsmooth priors. Our prior being BG, the MAP estimation implies a thresholding, that allows us to use the HT method as a MAP estimator. The BG estimation $\widehat{\boldsymbol{x}}$ may be interpreted as the pointwise multiplication: $\forall k, \widehat{\boldsymbol{x}}_k = \tilde{\boldsymbol{x}}_k \widehat{\boldsymbol{q}}_k$, $\tilde{\boldsymbol{x}}$ corresponding to the Hunt filter output and $\widehat{\boldsymbol{q}}$ being an estimate of the i.i.d. Bernoulli sequence with parameter λ, controlling the occurrence of pulses in $\boldsymbol{x}$. So, we consider the MAP estimation of $(\boldsymbol{x}, \boldsymbol{q})$, which maximizes the following joint posterior pdf:

$$p(\boldsymbol{x}, \boldsymbol{q}|\boldsymbol{y}, \boldsymbol{\theta}) \propto p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{q}, \boldsymbol{\theta}) p(\boldsymbol{x}|\boldsymbol{q}, \boldsymbol{\theta}) p(\boldsymbol{q}|\boldsymbol{\theta}) \tag{8}$$

where

- $p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{q}, \boldsymbol{\theta}) = p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{H}\boldsymbol{x}, r_n \boldsymbol{I}_N)$ because $p(\boldsymbol{n}|\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{0}, r_n \boldsymbol{I}_N)$ and $\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n}$;
- $p(\boldsymbol{x}|\boldsymbol{q}, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{0}, r_x \boldsymbol{Q})$ where $\boldsymbol{Q} = \text{diag}\{\boldsymbol{q}\}$;
- $p(\boldsymbol{q}|\boldsymbol{\theta}) = \lambda^{N_q}(1-\lambda)^{N-N_q}$, N_q being the number of nonzero samples of $\boldsymbol{q}$.

To estimate the three hyperparameters gathered into $\boldsymbol{\theta} = \{\lambda, r_x, r_n\}$, we consider the following JMAP [7]:

$$\begin{aligned}(\tilde{\boldsymbol{x}}, \widehat{\boldsymbol{q}}, \widehat{\boldsymbol{\theta}}) &= \arg\max_{(\boldsymbol{x}, \boldsymbol{q}, \boldsymbol{\theta})} p(\boldsymbol{x}, \boldsymbol{q}, \boldsymbol{\theta}|\boldsymbol{y}) \\ &= \arg\max_{(\boldsymbol{x}, \boldsymbol{q}, \boldsymbol{\theta})} p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{q}, \boldsymbol{\theta}) p(\boldsymbol{x}|\boldsymbol{q}, \boldsymbol{\theta}) p(\boldsymbol{q}|\boldsymbol{\theta})\end{aligned}$$

where $p(\boldsymbol{\theta})$ does not appear because it is considered as uniform (no *a priori* on the hyperparameters: it is in fact an ML estimator). This optimization problem is solved using an iterative procedure

$$\begin{cases} \left(\tilde{\boldsymbol{x}}^{(i)}, \widehat{\boldsymbol{q}}^{(i)}\right) = \arg\max_{(\boldsymbol{x}, \boldsymbol{q})} p\left(\boldsymbol{x}, \boldsymbol{q}, \lambda^{(i-1)}, r_x^{(i-1)}, r_n^{(i-1)}|\boldsymbol{y}\right) \\ \left(\widehat{\lambda}^{(i)}, \widehat{r_x}^{(i)}, \widehat{r_n}^{(i)}\right) = \arg\max_{(\lambda, r_x, r_n)} p\left(\boldsymbol{x}^{(i)}, \boldsymbol{q}^{(i)}, \lambda, r_x, r_n|\boldsymbol{y}\right). \end{cases}$$

The first optimization problem is approximatively solved using the HT method, and the hyperparameters are estimated using (8), with the assumption that $\boldsymbol{x}$ is a BG signal

$$\begin{aligned}\widehat{\lambda} &= \arg\max_{\lambda} p(\boldsymbol{q}|\lambda) = \frac{N_q}{N} \\ \widehat{r_x} &= \arg\max_{r_x} p(\boldsymbol{x}|\boldsymbol{q}, r_x) = \frac{1}{N_q}\|\widehat{\boldsymbol{x}}_k\|^2 \\ \widehat{r_n} &= \arg\max_{r_n} p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{q}, r_n) = \frac{1}{N}\|\boldsymbol{y} - \boldsymbol{H}\widehat{\boldsymbol{x}}\|^2.\end{aligned}$$

Rather than using (5) and (6) to estimate r_0 and r_1, we propose to estimate them directly from $\tilde{\boldsymbol{x}}$ as

$$(\widehat{r_0}, \widehat{r_1}) = \arg\max_{(r_0, r_1)} p(\tilde{\boldsymbol{x}}|\boldsymbol{q}, r_0, r_1).$$

From (4), we get for all k

$$\begin{aligned}p(\tilde{\boldsymbol{x}}_k|\boldsymbol{q}, r_0, r_1) &= \boldsymbol{q}_k \mathcal{N}(0, r_1) + (1 - \boldsymbol{q}_k)\mathcal{N}(0, r_0) \\ &= \mathcal{N}\left(0, r_1 \boldsymbol{q}_k + r_0(1 - \boldsymbol{q}_k)\right).\end{aligned}$$

Then

$$p(\tilde{\boldsymbol{x}}|\boldsymbol{q}, r_0, r_1) = \mathcal{N}\left(\boldsymbol{0}, r_1 \boldsymbol{Q} + r_0(\boldsymbol{I}_N - \boldsymbol{Q})\right).$$

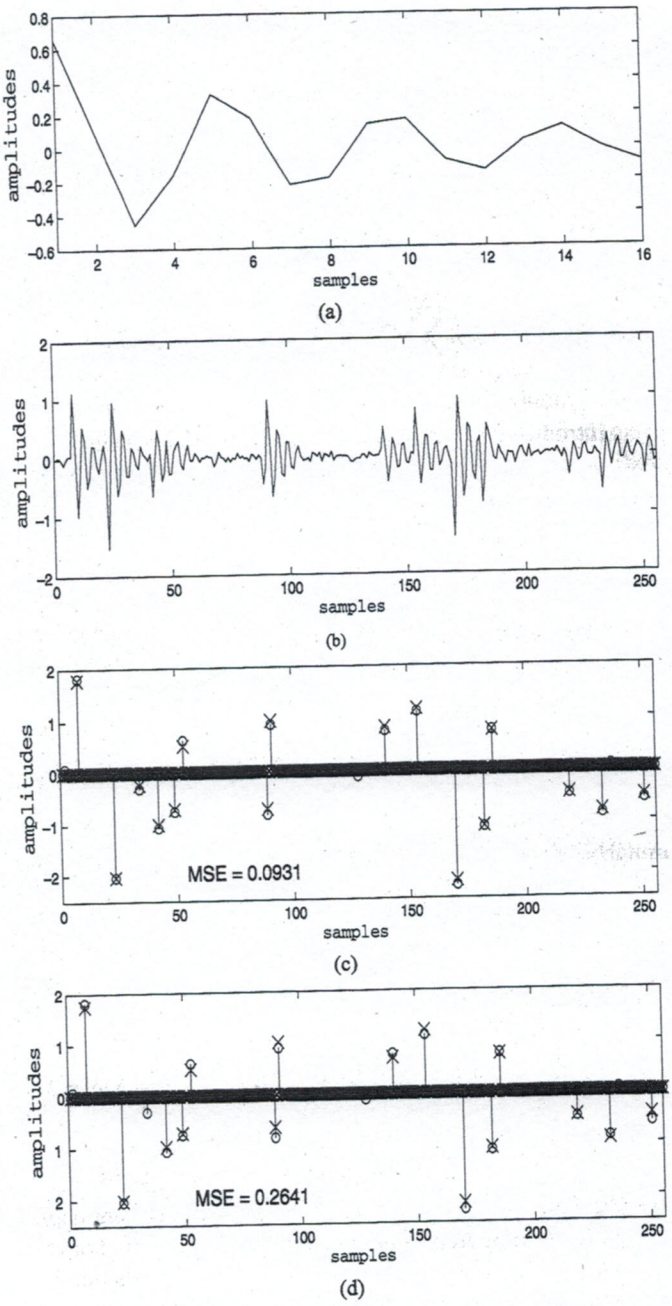


Fig. 2. SMLR and HT estimates. o represents real pulses, and × estimated ones. MSE: mean square error on the detected pulses. (a) Impulse response ($\boldsymbol{h}$). (b) Data ($\boldsymbol{y}$). (c) SMLR estimate. (d) HT estimate.

Straightforward calculations yield the following explicit expressions:

$$\widehat{r_1} = \frac{1}{N_q}\sum_{k=1}^{N} \tilde{\boldsymbol{x}}_k^2 \widehat{\boldsymbol{q}}_k, \quad \widehat{r_0} = \frac{1}{N - N_q}\sum_{k=1}^{N} \tilde{\boldsymbol{x}}_k^2 (1 - \widehat{\boldsymbol{q}}_k).$$

This approach has been chosen because it leads to a faster algorithm and gives better results than those obtained by using (5) and (6).

λ, r_x, and r_n are initialized to arbitrary values, while r_0 and r_1 are initialized using (5) and (6). The convergence test is made by comparing the signal and hyperparameter values from one iteration to the next. The procedure stops when they differ no

TABLE I
COMPUTATION TIME, PERCENTAGE OF DETECTED PULSES (DP), AND FALSE ALARMS (FA) FOR SEVERAL SNR

SNR = 20 dB	SMLR	23.39 s	92.0 % DP	2.0 % FA
	HT	0.16 s	87.0 % DP	5.0 % FA
SNR = 15 dB	SMLR	26.94 s	88.17 % DP	2.15 % FA
	HT	0.19 s	81.72 % DP	4.3 % FA
SNR = 10 dB	SMLR	37.59 s	69.07 % DP	3.09 % FA
	HT	0.23 s	61.85 % DP	8.24 % FA
SNR = 5 dB	SMLR	105.64 s	42.9 % DP	3.7 % FA
	HT	0.31 s	40.2 % DP	1.9 % FA

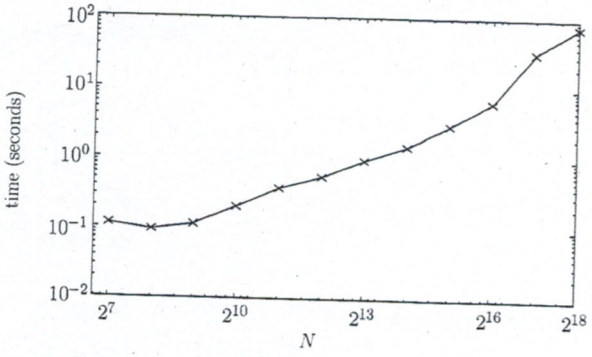


Fig. 3. Mean computation time.

more than 1%. Extensive simulations have shown the very satsfactory behavior of the procedure.

IV. SIMULATIONS RESULTS

Some simulations were carried out to assess the performances of the HT method and to compare them to those of the SMLR of [1]. The two methods were implemented in Matlab 6.5 on a Pentium III at 800 MHz. Fig. 2 shows the results achieved by the two methods on a 256-sample signal with a 16-sample impulse response and the following real hyperparameters: $\lambda = 0.08$, $\sigma_x = 1.5$, SNR = 15 dB. Comparing the results, it appears that the SMLR performs slightly better than the HT method. Table I gives the statistics obtained on ten simulations. They confirm the superiority of the SMLR at the price of a very high computation time as compared to the HT method. We also note that the performances of the two methods become comparable as the SNR increases. Fig. 3 shows the mean computation time evolution of the HT method as a function of N, confirming that the computational burden remains reasonable, even for very large signals.

V. CONCLUSION

The HT method is a sparse spike train deconvolution consisting in coupling the Hunt filter with a thresholding. We show that when the signal $\boldsymbol{x}$ is BG, a Gaussian mixture is a good model for the pdf of the Hunt filter output. From this result, we derive the threshold that minimizes the probability of errors. We then propose to use the HT method as a MAP estimator and to jointly estimate the hyperparameters of the problem. Simulations show that the method performs well at a very low computation time, making this approach very well-suited to process very large signals and to be implemented on embedded systems. Future work will be directed to find which global criterion is minimized.

REFERENCES

[1] F. Champagnat, Y. Goussard, and J. Idier, "Unsupervised deconvolution of sparse spike trains using stochastic approximation," *IEEE Trans. Signal Processing*, vol. 44, pp. 2988–2998, Dec. 1996.

[2] J. J. Kormylo and J. M. Mendel, "Maximum likelihood detection and estimation of Bernoulli–Gaussian processes," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 482–488, May 1982.

[3] V. Mazet, D. Brie, and C. Caironi, "Déconvolution impulsionnelle par filtre de Hunt et seuillage," in *Proc. GRETSI*, Sept. 2003.

[4] K. F. Kaaresen, "Deconvolution of sparse spike trains by iterated window maximization," *IEEE Trans. Signal Processing*, vol. 45, pp. 1173–1183, May 1997.

[5] J. Idier, *Approche bayésienne pour les problèmes inverses*. Paris, France: Hermès Science, 2001.

[6] B. R. Hunt, "The inverse problem of radiography," *Math. Biosci.*, vol. 8, pp. 161–179, 1970.

[7] A. Mohammad-Djafari, "Joint estimation of parameters and hyperparameters in a Bayesian approach of solving inverse problems," in *Proc. ICASSP*, Munich, Germany, Apr. 1997, pp. 2837–2840.

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

[9] M. Nikolova, "Local strong homogeneity of a regularized estimator," *SIAM J. Appl. Math.*, vol. 61, no. 2, pp. 633–658, 2000.