Estimation of a Gaussian Source with Memory in Bursty Impulsive Noise

Armando Vannucci*, Giulio Colavolpe*, Riccardo Pecori†, and Luca Veltri*
*Department of Engineering and Architecture, Università di Parma, 43124 Parma, Italy
†SMARTEST Research Centre, eCAMPUS University, Novedrate (CO), Italy
(contact: armando.vannucci@unipr.it)

***Abstract*—We consider the problem of estimating correlated Gaussian samples in (correlated) impulsive noise, through message-passing algorithms. This is a meaningful theoretical framework to model signal transmission on power-line communication systems. Due to the mixture of Gaussian variables (the samples) and Bernoulli variables (the impulsive noise switches), the complexity of messages increases exponentially with the number of samples. By adopting a Parallel Iterative Scheduling, with properly constrained messages, it turns out that each iteration of the proposed algorithm is equivalent to the parallel run of a classical Kalman Smoother and a binary sequence detection through the BCJR algorithm. Results demonstrate the effectiveness of the receiver along with its performance, in terms of mean square estimation error.**

***Index Terms*—Factor graphs; Impulsive noise; Variational Bayesian inference.**

I. Introduction

Environments subject to electromagnetic interference (EMI), like power substations or any sensing device or network in the vicinity of an EMI source, generate noise whose power strongly fluctuates in time. Such impulsive noise induces observation errors that affect the acquisition of a sequence of data samples. This occurs in power-line communication (PLC) systems, both for the detection of discrete symbols [1], and in sensing applications for the estimation of Gaussian sources [2]. Impulsive interference in PLC, e.g., due to switching devices or to (dis)connection of loads from mains, occurs in bursts, so that noise samples are correlated, resulting in a Markov-Middleton model [3]. A simplified two-state Markov process is adopted in [4], in the context of digital communications, where the discrete symbols are correlated through a code, as well as in [5], for the estimation of a memoryless Gaussian source. In this work, we extend the approach of [5] to a much more general and complicated scenario.

We adopt the same noise model with memory as [4], [5], while assuming a correlated Gaussian source for the samples of interest, which is, e.g., an accurate representation of the signal distribution in multicarrier systems, used in PLC or in asynchronous digital subscriber lines [2]. Besides bursty impulsive noise, we also consider memory in the observed signal sequence, through a simple autoregressive model. The resulting estimation problem is modelled with factor graphs (FGs), in an attempt to solve it with the sum-product algorithm (SPA) [6], a message passing algorithm the reader is assumed to be familiar with. Our main objective is thus the design of an optimal receiver for the correlated gaussian samples on a PLC system affected by bursty impulsive noise.

In the present case of mixed discrete (the impulsive noise switches) and continuous (signal) variables, the direct application of the SPA involves messages with exponentially growing complexity, so that we must resort to approximate variational inference techniques. Whenever either the noise sequence or the signal sequence is memoryless, the present problem degenerates into two classical estimation problems with known exact solutions (namely, the Kalman smoother [7] and the BCJR algorithm [8], [5]), that are—in terms of FGs and involved messages—the mirror image of each other. Exploiting this symmetry and by a proper merging of the two subproblems, we propose an algorithm, called *parallel iterative scheduling*, to approximate the optimal solution, based on hard decisions on the impulsive noise state.

II. System model

We refer to the channel model sketched in Fig. 1. We observe a frame of K samples $\{y_k\}$, expressed as

$$y_k = s_k + n_k^G + i_k n_k^I \qquad (k = 0, 1, ..., K-1) \tag{1}$$

where the signal sequence $\mathbf{s}$ and the two noise sequences $\mathbf{n}^G$ and $\mathbf{n}^I$ are independent of each other; the noise sequences are made of real Gaussian independent and identically distributed (i.i.d.) samples while the signal samples are obtained by filtering a real white Gaussian noise process $\mathbf{w}$ with zero mean value. Hence, $w_k \sim \mathcal{N}(0, \sigma_w^2)$, $n_k^G \sim \mathcal{N}(0, \sigma_G^2)$, and $n_k^I \sim \mathcal{N}(0, \sigma_B^2 - \sigma_G^2)$, all of them being $i.i.d.$ sequences. For the sake of simplicity, we analyze an *autoregressive model of order one* (AR(1))

$$s_k = a_1 s_{k-1} + w_k \tag{2}$$

i.e., s_k is obtained from w_k through a simple single-pole infinite impulse response (IIR) digital filter, with $|a_1| < 1$. Its mean value and variance are thus: $\eta_s = 0$; $\sigma_s^2 = \sigma_w^2/(1-a_1^2)$. Since it is the signal variance (compared to that of additive noise) that determines the performance of a linear estimator, we shall take the variance of s_k as a reference, and consider the driving white noise as $w_k \sim \mathcal{N}(0, (1-a_1^2)\sigma_s^2)$.

The sequence $\mathbf{i}$ is a *two-state Markov process* with binary values $i_k \in \{0, 1\}$ associated, respectively, with a "good" channel condition (G), with only background noise n^G present, or a "bad" one (B), with both background and impulsive noise

978-1-7281-0289-4/19/$31.00

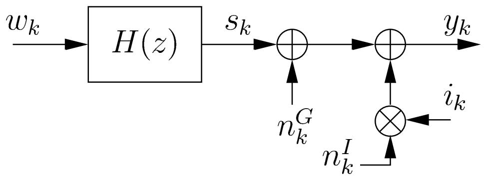

Figure 1. Channel model affected by Gaussian background noise n^G and impulsive noise n^I, that is switched by the two-state Markov process i_k.

n^I. The sum of background plus impulsive noise corresponds to a noise sequence $n_k^B = n_k^G + n_k^I \sim \mathcal{N}(0, \sigma_B^2)$, if $i_k = 1$, whose variance is the sum of σ_G^2 and $\sigma_I^2 = \sigma_B^2 - \sigma_G^2 = (R-1)\sigma_G^2$, where we define the ratio $R \stackrel{\triangle}{=} \sigma_B^2/\sigma_G^2 > 1$ (with σ_B^2 often much larger than the background noise variance σ_G^2).

The Markov process is characterized by the $[2 \times 2]$ *one-step transition probabities matrix* Π with entries $\pi_{r,c} = P\{i_{k+1} = c-1 \mid i_k = r-1\}$, $(r, c \in \{1,2\})$. The probability that the channel is in the "bad" condition, $p_B = P\{i_k = 1\} = \gamma\pi_{12}$, where $\gamma = (\pi_{12} + \pi_{21})^{-1}$ quantifies the memory of the Markov process [4]. In fact, the average duration of 1's sequences is $T_B = \gamma/p_G$, whereas it would be $T_B = 1/p_G$ for a memoryless process.

III. Factor Graph and message passing

The solution of the aforementioned estimation problem through message passing algorithms (Sum Product, Belief Propagation or others [6]) requires the expression of the joint probability distribution function[1] of the signal samples $\mathbf{s} = \{s_k\}$ and parameters $\mathbf{i} = \{i_k\}$ to be estimated given the observed samples $\mathbf{y} = \{y_k\}$, so that their marginals can be employed for the minimum mean square error (MMSE) or maximum a posteriori (MAP) estimation.

Disregarding the joint pdf $p(\mathbf{y})$ of the observations, which is a constant term, we seek to estimate the marginals of

$$\begin{aligned} p(\mathbf{s}, \mathbf{i} \mid \mathbf{y}) &\propto p(\mathbf{s}, \mathbf{i}, \mathbf{y}) = p(\mathbf{y} \mid \mathbf{s}, \mathbf{i}) p(\mathbf{s}) P(\mathbf{i}) \qquad (3) \\ &= \left[\prod_{k=1}^{K-1} p(y_k \mid s_k, i_k) p(s_k \mid s_{k-1}) P(i_k | i_{k-1}) \right] \\ &\quad \cdot p(y_0 \mid s_0, i_0) p(s_0) P(i_0) \end{aligned}$$

factorized as above thanks to the independence of signal $(\mathbf{s})$ and impulsive noise $(\mathbf{i})$. The FG that represents the joint probability distribution function is thus made of $K-1$ identical stages, each modelling one of the factors of the product in (3), where the *variable nodes* s_k and i_k are connected by the factor nodes representing the pdf/pmf in (3). In Fig. 2, we highlight the k-th stage to show the labels of the factor-to-variable node messages that travel on the FG edges. The stages are connected through a double chain of factor nodes: on the bottom line, the one-step transition probabilities of the Markov process, $P(i_k \mid i_{k-1}) = \pi_{(i_{k-1}+1),(i_k+1)}$ and on the top line, the conditional pdfs of the AR(1) process, directly obtained from the system model (2),

$$p(s_k \mid s_{k-1}) = g\left(s_k - a_1 s_{k-1}, (1-a_1^2)\sigma_s^2\right), \qquad (4)$$

where $g(x-\eta, \sigma^2)$ is the standard Gaussian pdf with mean η and variance σ^2. As for the 0-th stage, the factor node

$$P(i_0) = p_G \delta(i_0) + p_B \delta(i_0 - 1) \qquad (5)$$

sets the a-priori pmf of the initial value for the binary variable i_0. In the same way,

$$p(s_0) = g(s_0, \sigma_s^2) \qquad (6)$$

is initialized at its stationary Gaussian distribution. The conditional pdfs of the observed samples are again Gaussian:

$$p(y_k \mid s_k, i_k) = g(y_k - s_k, \sigma_G^2 + i_k \sigma_I^2) \qquad (7)$$

with mean equal to the conditioning s_k and variance dictated by the channel condition: $\sigma_{n,k}^2 = \sigma_G^2 + i_k\sigma_I^2 \in \{\sigma_G^2, \sigma_B^2\}$, depending on the impulsive noise switch at time k.

The messages in Fig. 2 have subscripts "*u,d,f,b*" denoting their (up, down, forward, backward) directions. According to the rules of the SPA [6], the variable-to-factor node messages are simply the product of the incoming messages arriving at the (sender) variable node from all of its neighbouring factor-nodes except the (destination) one to which the message is addressed. For these reasons, we did not label the variable-to-factor node messages in Fig. 2 and consider them implicitly in the factor-to-variable node messages detailed hereafter.

The forward and backward travelling messages, in the top part of the FG are:

$$\begin{aligned} p_f(s_k) &= \int p_f(s_{k-1}) p_u(s_{k-1}) p(s_k \mid s_{k-1}) ds_{k-1} \qquad (8) \\ p_b(s_k) &= \int p_b(s_{k+1}) p_u(s_{k+1}) p(s_{k+1} \mid s_k) ds_{k+1} \qquad (9) \end{aligned}$$

where $k = 1, ..., K-1$ in (8) and, for the purpose of initialization, $p_f(s_0) = p(s_0)$ is the initial message to node s_0, while $k = K-2, ..., 0$ in (9) and $p_b(s_{K-1}) = 1$ is the identity message to the last node.

The messages travelling on the bottom line of the FG are conceptually similar, except that they involve discrete (binary) probabilities, hence the saturation with respect to the previous (forward) or subsequent (backward) variable node is performed as a binary sum:

$$\begin{aligned} P_f(i_k) &= \sum_{i_{k-1}} P_f(i_{k-1}) P_d(i_{k-1}) P(i_k \mid i_{k-1}) \qquad (10) \\ P_b(i_k) &= \sum_{i_{k+1}} P_b(i_{k+1}) P_d(i_{k+1}) P(i_{k+1} \mid i_k) \qquad (11) \end{aligned}$$

where $k = 1, ..., K-1$ in (10), with the initial condition $P_f(i_0) = P(i_0)$ in (5) modelling the stationary unconditional pmf of the impulsive noise switch, while $k = K-2, ..., 0$ in (11), with initial condition $P_b(i_{K-1}) = 1$, modelling the absence of future samples to rely upon. Both update equations use the entries $\pi_{ij} \quad (i, j, = 1, 2)$ of the one-step transition matrix.

The 'vertical' message $P_d(i_k)$ is computed by multiplying the messages in (8) and (9) (as forwarded by variable node

[1]We use the notation $P(\cdot)$ to identify a probability mass function (pmf) of a discrete random variable and $p(\cdot)$ to denote a probability density function (pdf). We use the term probability distribution function to denote a continuous pdf with some discrete probability masses. For a probability distribution function, we still use the symbol $p(.)$.

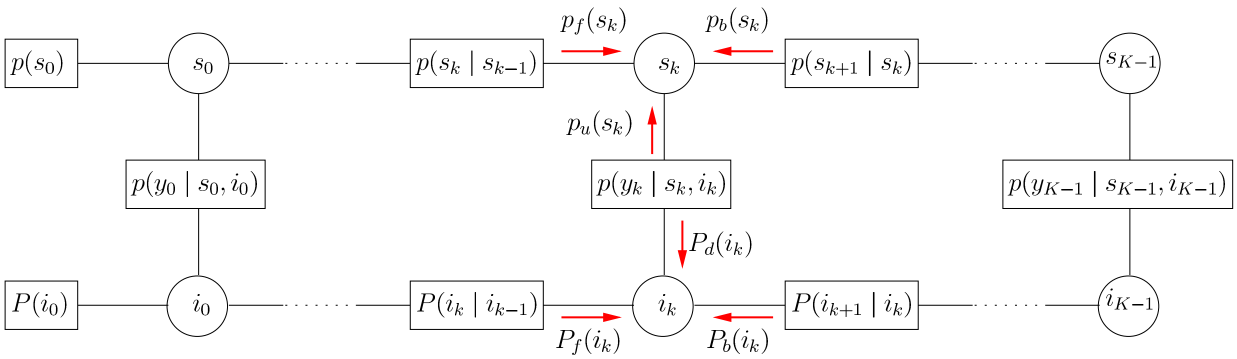

Figure 2. FG modelling the joint probability distribution function $p(\mathbf{s}, \mathbf{i} \mid \mathbf{y})$. The k-th stage is highlighted along with messages sent from factor nodes to variable nodes. The initial ($k = 0$) stage is driven by messages coinciding with the priors, $P_f(i_0) = P(i_0)$ and $p_f(s_0) = p(s_0)$, while the last ($k = K-1$) stage is terminated without a transition factor node, hence $P_b(i_{K-1}) = p_b(s_{K-1}) = 1$.

s_k) by (7), and then integrating (i.e., "summing"), according to the SPA, to saturate with respect to all the variables (s_k only, in this case) except that of interest (i_k); we have thus

$$P_d(i_k) \quad = \quad \int p_f(s_k) p_b(s_k) p(y_k \mid s_k, i_k) ds_k \,. \tag{12}$$

The message $p_u(s_k)$, directed (upwards) towards s_k, entails a saturation with respect to i_k, hence a (binary) summation instead of an integral (being i_k a Bernoulli variable):

$$p_u(s_k) \quad = \quad \sum_{i_k} P_f(i_k) P_b(i_k) p(y_k \mid s_k, i_k) \,. \tag{13}$$

IV. Parallel Iterative Scheduling

As intuitive, being the present problem "fed" by Gaussian pdfs, namely those in (4) and (7), and by the binary variables i_k, the messages passed along the FG edges are represented by *Gaussian mixtures*. For instance, $p_u(s_k)$ in (13) is a linear combination of two Gaussian pdfs, with given (identical) mean and (different) variance, with coefficients corresponding to a pmf $P_u(i_k) = P_f(i_k) P_b(i_k)$. The problem is that we need to propagate $p_u(s_k)$ to the upper part of the FG, through (8) and (9), where the number of components in the mixture doubles at each step of the forward/backward iteration, starting from the single Gaussian $p(s_0)$. Note that the growth in the number of Gaussian components of the mixture (in this case a doubling of their number), at each step, is not due to the presence of cycles in the FG; rather, it simply depends on the presence of discrete variables. In our case, if the Bernoulli variables i_k were all independent of each other, hence did not form a Markov chain, their variable nodes in the FG of Fig. 2 would be only connected to the factor nodes $p(y_k \mid s_k, i_k)$ of the observed samples and would send them messages containing their prior (Bernoulli) distribution. The FG would thus be free of cycles but still the messages $p_u(s_k)$ would be a Gaussian mixture as in (13), weighted by the prior probabilities of i_k, thus implying the proliferation of Gaussian components in the mixture messages propagating along the chain of variable nodes s_k in the upper part of the FG. The only chance to avoid such a rapid increase in the message complexity is to assume that the variables i_k are deterministic, so that each $p_u(s_k)$ consists of a single Gaussian.

Driven by this observation, we adopt a *parallel iterative scheduling* (PISch) for the iterative message passing operations, where the FG is divided in two—upper and lower—halves that operate in parallel, exchanging their own messages 'horizontally' ((8),(9) and (10),(11)), and sending 'vertical' messages (12),(13) to the other half only after completing a forward/backward pass. In particular, the messages (13), passed from the lower to the upper half of the graph, should account for the estimated (Bernoulli) pmfs of individual impulsive noise switches i_k, which would start the generation of Gaussian mixtures with increasing complexity, as discussed above. One rough way to overcome the problem is to take a *hard decision* on $i_k = m \quad (m \in \{0, 1\})$, hence to estimate its pmf as a deterministic $\widetilde{P}(i_k) = \delta(i_k - m)$, which reduces each possible mixture to a simple Gaussian message. Such a constraint implies that the messages passed in the opposite direction, i.e., towards the lower FG half, are simple Gaussian estimates of the pdfs of individual signal samples, $\widetilde{p}(s_k)$ (that can be stored and passed, e.g., through their mean and variance).

A. The upper half of the FG: noise with known variance

If we assume that the variance of the noise affecting the observations is deterministic, the random variables i_k disappear and the FG of Fig. 2 reduces to a cycle-free graph, of which one stage is reported in Fig. 3, that models the joint distribution of samples s_k conditioned on the observations y_k, so that

$$\begin{aligned} p(\mathbf{s} \mid \mathbf{y}) \propto p(\mathbf{s}, \mathbf{y}) = p(\mathbf{y} \mid \mathbf{s}) p(\mathbf{s}) \qquad (14) \\ = \left[\textstyle\prod_{k=1}^{K-1} p(y_k \mid s_k) p(s_k \mid s_{k-1}) \right] p(y_0 \mid s_0) p(s_0) \end{aligned}$$

is clearly a simplified version of (3). In particular, the observed samples are expressed as $y_k = s_k + n_k$, for $k = 0, 1, ..., K-1$, where the noise sequence is made of zero-mean Gaussian samples, $n_k \sim \mathcal{N}(0, \sigma^2_{n,k})$ whose (known) variance can be different, from sample to sample, so that the random sequence $\mathbf{n}$ is not stationary. The signal samples are still those expressed in the AR(1) model in (2). Hence, the transition probabilities of the factor nodes appearing in the upper part of Fig. 3

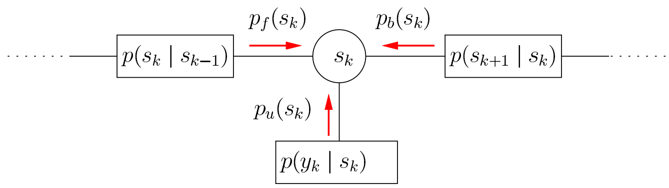

Figure 3. One of the connected stages of the FG modelling the joint pdf $p(\mathbf{s} \mid \mathbf{y})$ of AR(1) correlated Gaussian samples in AWGN with known variance. This model coincides with the Kalman filtering/smoothing problem.

coincide with (4), while the expression of the factor nodes for the observed samples is

$$p(y_k \mid s_k) = g(y_k - s_k, \sigma^2_{n,k}) = p_u(s_k) \tag{15}$$

so that the *upwards* message $p_u(s_k)$ is a single Gaussian. In this scenario, the expressions for the forward/backward travelling messages can be dramatically simplified by noting that all of the pdfs appearing in (8),(9) are Gaussian.

As a matter of fact, the assumption of additive Gaussian noise with known variance per sample changes the nature of the system in (1), so that the problem degenerates into the estimation of correlated Gaussian samples (s_k) in the presence of (possibly non-stationary) additive white Gaussian noise (AWGN) with known statistics, which is the classical problem of Kalman filtering/smoothing [7]. Thus, the operations performed by the SPA on the cycle-free FG in Fig. 3 implement its algorithmic steps [6], whose final result is a Gaussian estimate with conditional variance $\widehat{\sigma}^2_k$ and mean $\widehat{\eta}_k$.

The performance of a Kalman smoother with perfect channel state information on impulsive noise is the best that can be achieved, for given noise statistics, hence we shall employ its MSE, as an experimental lower bound in the results that follow (see Figs. 5 and 6).

B. The lower half of the FG: independent signal samples

If we assume that $a_1 = 0$ in (2), the signal samples s_k are uncorrelated, hence each sample is statistically independent of the others. In particular, since $p(s_k \mid s_{k-1}) = p(s_k)$ coincides with the prior pdf of the k-th sample, the factor nodes of the upper line in Fig. 2 are only connected to the following variable node. Each stage of the FG thus reduces to the structure sketched in Fig. 4, where there are no more forward/backward messages to/from s_k, so that the FG is free of cycles and the message sent from factor node $p(s_k)$, now labelled

$$p_d(s_k) = g(s_k - \eta_{s,k}, \sigma^2_{s,k}) = p(s_k) \tag{16}$$

simply coincides with the prior Gaussian distribution of s_k. In (16), we account for possibly different mean and variance for each sample, while in the case of a stationary sequence, as generated by (2), it is simply $(\eta_{s,k}, \sigma^2_{s,k}) = (0, \sigma^2_w)$. Following the "down" direction, the above message (forwarded by variable node s_k) is multiplied by the factor $p(y_k \mid s_k, i_k)$ in (7), whose variance $\sigma^2_{n,k} = \sigma^2_G + i_k\sigma^2_I \in \{\sigma^2_G, \sigma^2_B\}$ depends on the impulsive noise switch at time k. The product is integrated (i.e., summed, according to the SPA), to saturate with respect to s_k to get a convolution of Gaussian pdfs.

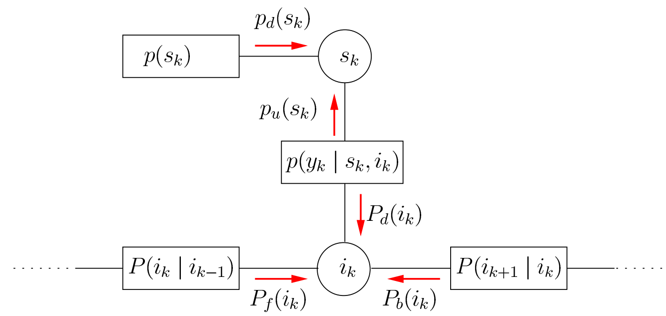

Figure 4. One of the connected stages of the overall FG modelling the joint probability distribution function $p(\mathbf{s}, \mathbf{i} \mid \mathbf{y})$ in the case of independent samples s_k. The initial ($k = 0$) stage is driven by $P_f(i_0) = P(i_0)$ while the last ($k = K - 1$) stage is terminated without a transition factor node, hence $P_b(i_{K-1}) = 1$.

The upper and lower half of the FG are the mirror image of each other. Despite the variables s_k are continuous and i_k are discrete, the probabilistic meaning of $P_f(i_k)$ and $P_b(i_k)$ is in fact similar to that of $p_f(s_k)$ and $p_b(s_k)$: $P_f(i_k) = p(i_k, y_0, ..., y_{k-1})$ and $P_b(i_k) = p(y_{k+1}, ..., y_{K-1} \mid i_k)$, so that their update equations (10),(11) implement *filtering and prediction steps*, similar to those of a Kalman smoother in Sec. IV-A.

In particular, the *filtering step* for the forward message produces $P_f(i_k)P_d(i_k) = p(i_k, y_0, ..., y_k)$ (due to the independence of y_k on past observations, given i_k); this quantity coincides with the probabilities defined as "α" in [8], whereas $P_b(i_k)$ coincide with the "β" probabilities in [8]. As a matter of fact—as it is already known [6]—both the initial conditions and the update rules are such that the SPA applied to the present cycle-free FG implements the well-known BCJR algorithm [5], where i_k are the symbols to be "decoded". At the final, *completion step*, of the BCJR algorithm, the "α" and "β" are multiplied to yield the posterior estimate of i_k. This is the same operation that completes the SPA, where all incoming messages to node i_k are mutiplied to yield

$$\begin{aligned} \overline{P}(i_k) &= P(i_k \mid \mathbf{y}) \propto p(i_k, \mathbf{y}) \\ &= p(i_k, y_0, ..., y_k) p(y_{k+1}, ... y_{K-1} \mid i_k) \\ &= P_f(i_k) P_d(i_k) P_b(i_k) = \widetilde{P}(i_k) \end{aligned} \tag{17}$$

(being future symbols $(y_{k+1}, ..., y_{K-1})$ independent of past and present ones, given i_k), where the marginal pmf of i_k is a normalized version of $\widetilde{P}(i_k)$, i.e., $\overline{P}(i_k) = \widetilde{P}(i_k) \left(\sum_{i_k} \widetilde{P}(i_k)\right)^{-1}$.

C. Symbol estimates from soft channel state estimates

The stages in Fig. 4 are connected only through the bottom line in Fig. 2, that models the Markov chain i_k, so that the FG is free of cycles. Its message-passing procedure thus terminates in a single forward+backward pass, converging at the conditional marginal probabilities (17). At this point, the posterior distribution of s_k can be estimated, which is our ultimate goal. To this aim, $P_u(i_k) = P_f(i_k)P_b(i_k)$ is sent upwards to compute the message in (13),

$$\begin{aligned} p_u(s_k) &= \sum_{i_k} p(y_k \mid s_k, i_k) P_u(i_k) \\ &= \sum_{i_k} p(y_k \mid s_k, i_k, \mathbf{y}\backslash y_k) p(i_k, \mathbf{y}\backslash y_k) \\ &= p(y_k, \mathbf{y}\backslash y_k \mid s_k) = p(\mathbf{y} \mid s_k) \end{aligned}$$

which is multiplied by the message travelling in the opposite direction on the same edge, that is the prior pdf $p_d(s_k)$ in (16), to get the posterior estimate

$$\begin{aligned} p(s_k \mid \mathbf{y}) &\propto p(s_k, \mathbf{y}) = p(\mathbf{y} \mid s_k)p(s_k) = p_u(s_k)p_d(s_k) \qquad (18) \\ &= \left[P_u(i_k = G) g(s_k - y_k, \sigma_G^2) \right. \\ &\left. + \; P_u(i_k = B) g(s_k - y_k, \sigma_B^2)\right] g(s_k - \eta_{s,k}, \sigma_{s,k}^2). \end{aligned}$$

This is (as much as $p_u(s_k)$) a binary Gaussian mixture, weighted by the probabilities $P_u(i_k)$. In (18) and hereafter, we use $\{G, B\}$ for the values of the impulsive noise switch i_k, instead of $\{0, 1\}$, for uniformity of notation.

The mean value of (18) is the MMSE estimate

$$\widehat{s}_k = E\left[s_k \mid \mathbf{y}\right] \qquad (19)$$

while the MSE of this estimator

$$\widehat{\sigma}_k^2 = E[(s_k - \widehat{s}_k)^2 \mid \mathbf{y}] = E[s_k^2 \mid \mathbf{y}] - \widehat{s}_k^2 \qquad (20)$$

can be computed from the conditional variance of (18). Eq. (19) expresses the same result as in [2, eq. (8)], there obtained in the case of Middleton's class-A impulsive noise.

Fig. 5 shows the MSE performance of the estimate (19), obtained from the lower FG half, as applied to independent signal samples s_k (label "LO noSI (ind. samples)") observed in correlated impulsive noise ($\gamma = 100$, $p_B = 0.1$, $R = 100$). Its performance is very close to that of an ideal estimator, derived assuming perfect CSI on i_k. This demonstrates that the estimate $\widetilde{P}(i_k)$ of the channel state, resulting from the BCJR algorithm applied to the FG in Fig. 4, is extremely accurate. If the same estimator is applied to correlated ($a_1 = 0.9$) signal samples, with the same noise statistics, instead, its performance (curve labelled "LO noSI (AR1 samples)") significantly degrades.

In order to account for the signal samples' correlation, a more reliable estimate of their distribution—compared to the prior distribution in (16)—should be provided by the factor nodes $p(s_k)$. To that purpose, we can use the estimates provided by the upper FG half discussed in Sec. IV-A, i.e., the Gaussian pdfs estimated by Kalman Smoothing. Since the upper FG half needs itself to have some channel state information on the variables i_k, we assume here an ideal perfect CSI. Fig. 5 shows the great performance enhancement provided by such a setup with *estimated Signal Information* (curves labelled "LO est.SI (AR1 samples)"). For completeness, we applied the same strategy when the signal samples are independent ($a_1 = 0$). In this case (curve labelled "LO est.SI (ind. samples)"), Fig. 5 shows no improvements with respect to the absence of *Signal Information*, clearly due to the fact that there is no side

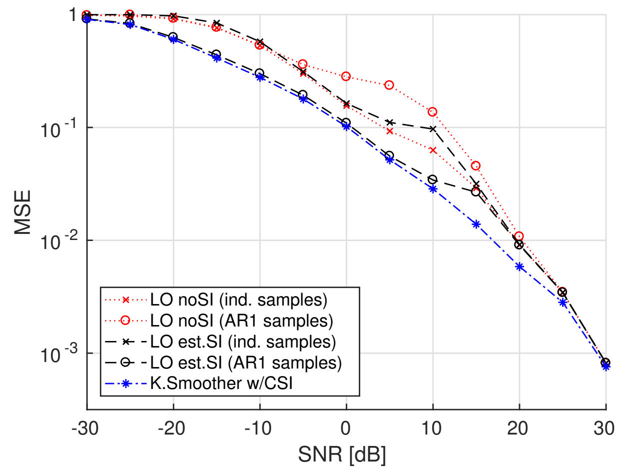

Figure 5. MSE of the estimate in (19) as applied to: independent signal samples ($a_1 = 0$) or AR(1) correlated samples ($a_1 = 0.9$) with unit variance ($\sigma_s^2 = 1$) in correlated impulsive noise ($\gamma = 100$, $p_B = 0.1$, $R = 100$). Estimates with no prior signal information (label "noSI") are compared with those using *estimated signal information* (label "estSI"), provided by the upper FG half with perfect CSI. 10^2 frames of 10^3 samples have been considered, for each SNR value. The experimental lower bound in the case of AR(1) correlated samples (Kalman smoother with CSI, discussed in Sec. IV-A) is reported for comparison.

information to gain: since the signal samples are independent, the prior pdf (16) already describes them faithfully and can only be approximated by the noisy estimates of the upper FG half (hence a slight degradation of performance).

V. Simulation Results

Fig. 6 shows the MSE of the signal estimate obtained by the Parallel Iterative Scheduling (PISch) algorithm described above. We considered the following system parameters: for the AR(1) signal samples, $a_1 = 0.9$ and $\sigma_s^2 = 1$; for correlated noise, we set $\gamma = 100$, $p_B = 0.1$, and $R = 100$.

In particular, the two halves of the FG (upper+lower) are run in parallel and, after each iteration, exchange their estimates for the Gaussians $\widetilde{p}(s_k)$ and a hard decision, i.e., a single Dirac, for $\widetilde{P}(i_k)$. At the first iteration, $\widetilde{P}(i_k)$ is set to the prior pmf $P(i_0)$ in (5), for any i_k (as if they formed an i.i.d. sequence), so that all factor nodes in (7) use the same (average) noise variance $\sigma_{n,k}^2 = \sigma_n^2 = (1-p_B)\sigma_G^2 + p_B\sigma_B^2 = \sigma_G^2\left((1-p_B) + p_B R\right)$; similarly, $\widetilde{p}(s_k)$ is initialized, for any s_k, with the prior pdf (6) of s_0 (as if the signal samples were uncorrelated). The upper and lower halves of the FG complete a forward/backward pass, then send messages to the other half along the vertical edges.

In order to check convergence, the MSE in Fig. 6 is evaluated and plotted for two different signal estimates. Both are the expectation of the estimated marginal $\widetilde{p}(s_k \mid \underline{y}) \propto p_u(s_k)p_d(s_k)$, that is produced on the vertical edge connected to the variable node s_k, but the product of messages is calculated in two ways, right after completing the message passing procedure on each of the two FG halves. For the curves labelled "LO" (or "UP"), in figures, the upwards message $p_u^n(s_k)$ (or the downwards $p_d^n(s_k)$), produced at the present iteration n by the lower (or upper) half of the FG,

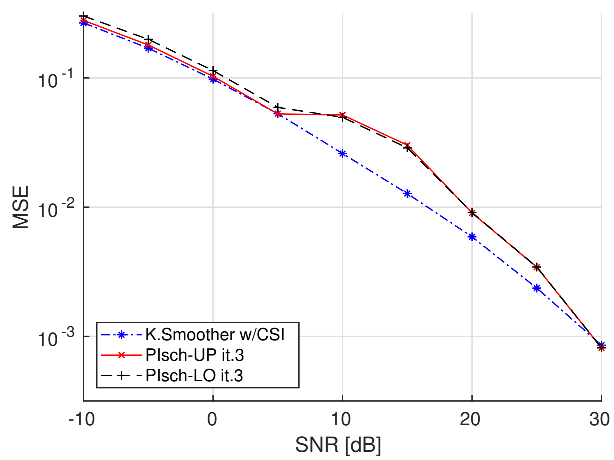

Figure 6. MSE for the estimate of AR(1) correlated signal samples ($a_1 = 0.9$), provided by the Parallel Iterative Scheduling algorithm, at convergence. Parameters' values as in Fig. 5. The experimental lower bound (Kalman smoother with CSI, discussed in Sec. IV-A) is reported for comparison. 10^2 frames of 10^3 samples each have been considered, for each SNR value.

is multiplied by the downwards message $p_d^{n-1}(s_k)$ (or the upwards $p_u^{n-1}(s_k)$) of the previous iteration $(n-1)$.

Since, for the PISch algorithm, a hard decision on i_k fixes the noise variance, the "UP" estimate is directly provided by the upper half of the FG as the result of a Kalman Smoother, with noise variance dictated by the estimated i_k; while the "LO" estimate is (19), as derived as a result of lower half of the FG. The two curves in Fig. 6 practically coincide after three iterations, meaning that the iterative algorithm has stabilized at convergence. We numerically checked that, even after 10 iterations, the MSE curves did not change appreciably.

As seen in Fig. 6, the performance of PISch tends to follow a *waterfall* shape, i.e., a sequence of breakpoints where the slope changes (a similar behavior is observed in Fig. 5, when $a_1 = 0$). This is not a numerical artifact, as one may conjecture, but rather an intrinsic feature of a two-states system, like the one implied by the hard decisions of PISch. A theoretical justification for this behavior can be given, but is outside of the scope of the present work.

As seen in the figure, the results of the PISch algorithm significantly deviate from those of a Kalman Smoother with perfect CSI, discussed in Sec. IV-A, which acts as an experimental lower bound, since it implements an optimal 'genie aided' estimator that knows the noise statistics at each time epoch. The maximum deviation is around 5 dB, at intermediate SNR values.

The problem is that the simple PISch algorithm relies on 'hard messages', i.e., on hard decisions on the impulsive noise swithes i_k. A more sophisticated algorithm, such as Expectation Propagation [9], would instead rely on a Gaussian projection of the (mixture) messages containing soft information on the channel state, with an expected enhanced performance, in terms of MSE. Typically, the benefits of exchanging soft information are especially evident at moderate-to-high SNR values, where the performance of PISch, in Fig. 6, converges onto the above metioned 'waterfall' shape.

Investigation on more complicated and performing algorithms is left to future developments. It is however interesting to note that the apparently heuristic approach of PISch can be viewed in a perspective similar to that of EP, i.e., as a divergence-based approximation of messages. In fact, despite employing different metrics from the KL divergence adopted by EP, the Gaussian mixture message $p_u(s_k)$ is approximated, in the PISch algorithm, with only one of its components, hence with a simple Gaussian message $g(s_k)$, chosen to be the one with the largest value of $P_u(i_k)$, hence with the largest mass, as per (13). As discussed in [10], the selection of the component with largest mass in a mixture p_u corresponds to approximating the mixture with a Gaussian g that minimizes the α-Divergence $D_\alpha(p_u \| g)$ with $\alpha \to -\infty$, that is the 'exclusive' or 'zero-forcing' α-Divergence.

VI. Conclusions

The estimation of a correlated Gaussian sequence affected by bursty impulsive noise, considered in this work, is a meaningful theoretical framework to analyze multicarrier signal transmission on power-line communication systems. The design of a receiver based on message-passing algorithms, implementing varational inference techniques, entails messages with exponentially increasing complexity, for which approximate solutions are demanded. By identifying the 'critical messages', we implemented a simple and novel algorithm for its approximate solution. Complexity reduction is achieved by taking a hard decision on the impulsive noise state, which leads to suboptimal performance, as expected. An extension to more computationally demanding algorithms, such as Expectation Propagation, is thus foreseen as future work, in the attempt of achieving a performance closer to optimality.

References

[1] D. Fertonani and G. Colavolpe, "A Robust Metric for Soft-Output Detection in the Presence of Class-A Noise", *IEEE Trans. Commun.*, vol. 57, no. 1, pp. 36-40, Jan. 2009.

[2] P. Banelli, "Bayesian Estimation of a Gaussian Source in Middleton's Class-A Impulsive Noise", *IEEE Signal Proc. Lett.*, vol. 20, no. 10, pp. 956-959, Oct. 2013.

[3] G. Ndo, F. Labeau, and M. Kassouf, "A Markov-Middleton Model for Bursty Impulsive Noise: Modeling and Receiver Design", *IEEE Trans. Power Del.*, vol. 28, no. 4, pp. 2317-2325, Oct. 2013.

[4] D. Fertonani and G. Colavolpe, "On Reliable Communications over Channels Impaired by Bursty Impulse Noise", *IEEE Trans. Commun.*, vol. 57, no. 7, pp. 2024-2030, July 2009.

[5] S. Alam, G. Kaddoum, and B. L. Agba, "Bayesian MMSE Estimation of a Gaussian Source in the Presence of Bursty Impulsive Noise", *IEEE Comm. Lett.*, vol. 22, no. 9, pp. 1846-1849, Sep. 2018.

[6] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor Graphs and the Sum-Product Algorithm", *IEEE Trans. Inform. Theory*, vol. 47, no. 2, pp. 498-519, Feb. 2001.

[7] J. R. Movellan, "Discrete Time Kalman Filters and Smoothers", *MPLab Tutorials*, Univ. of California, San Diego (USA), 2011.

[8] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate", *IEEE Trans. Inform. Theory*, vol. 20, no. 2 , pp. 284-287, Mar. 1974.

[9] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*, MIT Press, 2006. ISBN: 026218253X, http://www.gaussianprocess.org/gpml/

[10] T. Minka, "Divergence measures and message passing", Tech. Report, Microsoft Research Ltd., Cambridge (UK), Jan. 2005.