DETERMINISTIC PARTICLE FILTERS FOR JOINT BLIND EQUALIZATION AND DECODING ON FREQUENCY SELECTIVE CHANNELS

Claudio J. Bordin Jr. and Luiz A. Baccalá

Escola Politécnica, Universidade de São Paulo, 05508-900, São Paulo, Brazil.
E-mail: {bordin,baccala}@lcs.poli.usp.br.

ABSTRACT

This work proposes deterministic particle filtering structures for joint blindly equalizing/decoding convolutionally coded signals transmitted over frequency selective channels. After describing the proposed structures, we show how to evaluate the weight update functions corresponding to the adopted signal model. Numerical simulations show that the algorithm employing deterministic particle selection greatly outperforms alternative stochastic strategies, even when the latter employ the optimal importance function.

1. INTRODUCTION

Particle filtering techniques have attracted much interest due to their ability to provide approximate solutions to otherwise intractable Bayesian filtering problems. These methods approximate target posterior densities by a weighted sum of Dirac measures centered on the so-called particles - samples of the inferred random variable - which are most frequently obtained by stochastic procedures based on the importance sampling principle.

Recently, several publications [1] [2] pointed out that the raw Monte Carlo procedure used to extend, i.e., draw new elements to the particles at each iteration can be replaced by deterministic procedures when the variable being estimated has a low dimensional discrete distribution. The resulting algorithms, henceforth called deterministic particle filters, generally exhibit greatly improved performance over traditional algorithms.

As described next, deterministic particle filters must perform particle selection (i.e., resample) at each iteration, either via stochastic or deterministic methods [2]. In this work, we provide a new interpretation of deterministic particle filtering algorithms and evaluate their performance in the solution of the joint blind equalization and decoding problem [3], considering now the use of recursive convolutional codes. For the sake of comparison, simulation results of both stochastic and deterministic resampling procedures are provided.

Mr. Bordin work was funded by FAPESP under the grant 02/11457-7.

This work is organized as follows: we introduce particle filtering techniques in Sec. 2, followed by a description (Sec. 3) of the joint equalization and decoding problem and the deduction of the allied densities needed to solve it via particle filtering. In Sec. 4 numerical simulation results illustrate the relative performance of the proposed methods, wherefrom conclusions are summed up in Sec. 5.

2. PARTICLE FILTERS

Let y_n denote the observed output at instant n of a possibly non-linear and time-varying stochastically driven system whose state variable x_n we want to estimate. In a Bayesian filtering framework, one wishes to determine the posterior density $p(x_{0:n}|y_{0:n})$, $x_{0:n} := \{x_0, \ldots, x_n\}$, which collects all statistical information about $x_{0:n}$ embodied by $y_{0:n}$. Particle filters approximate the desired posterior density by

$$\hat{p}_n(x_{0:n}|y_{0:n}) = \sum_{i=0}^{P-1} w_n^{(i)} \delta(x_{0:n} - x_{0:n}^{(i)}) \tag{1}$$

where $x_{0:n}^{(i)}$ are the so-called particles (totalling P of them), $w_n^{(i)}$ their respective weights and $\delta(.)$ denotes a Dirac unit mass.

Particle filters differ from other Monte Carlo estimation methods in that the estimate $\hat{p}_n(x_{0:n}|y_{0:n})$ is determined recursively in time, and is accomplished sequentially by determining new particle elements $\{x_n^{(i)}\}_{i=0}^{P-1}$ and updating their respective weights $\{w_n^{(i)}\}_{i=0}^{P-1}$ recursively.

2.1. Stochastic Particle Filters (SPF)

In this work, stochastic particle filters refer to the general class of algorithms that perform particle extension, i.e., obtain the elements $\{x_n^{(i)}\}_{i=0}^{P-1}$ by means of random draws. Though extensively described in the literature (see [4] for a review), these algorithms are presented here in a slightly modified form so as to set the stage for an easier introduction of deterministic algorithms, so that both approaches can be appreciated from an unified standpoint.

0-7803-9404-6/05/$20.00 ©2005 IEEE

Briefly consider Table 1 which describes the SPF algorithm known as "Bootstrap Filter" whose operation is aimed at recursively generating samples of the joint posterior density $p(x_{0:n}|y_{0:n})$. By factorizing this density as

$$p(x_{0:n}|y_{0:n}) \propto p(x_{0:n-1}|y_{0:n-1}) \\ p(x_n|x_{0:n-1}, y_{0:n-1})p(y_n|x_{0:n}, y_{0:n-1}), \quad (2)$$

one can easily show that, starting from the set

$$\{x_{0:n-1}^{(i)}; w_{n-1}^{(i)}\}_{i=0}^{P-1}$$

that approximates $p(x_{0:n-1}|y_{0:n-1})$ via (1), the set

$$\{x_{0:n}^{(i)}; w_{n-1}^{(i)}\}_{i=0}^{P-1}$$

obtained after step 1 (Table 1) approximates the predictive density $p(x_{0:n}|y_{0:n-1})$, accounting for the factor $p(x_n| x_{0:n-1}, y_{0:n})$ in (2). Likewise step 2 accounts for the factor $p(y_n|x_{0:n}, y_{0:n-1})$ in (2), generating a weighted sample of $p(x_{0:n}|y_{0:n})$, which may be replaced by an unweighted sample if step 3 (resampling) is carried out.

1)	Draw $x_n^{(i)} \sim p(x_n|x_{0:n-1}^{(i)}, y_{0:n-1})$.
2)	Evaluate and normalize the weights $w_n^{(i)} \propto w_{n-1}^{(i)} p(y_n|x_{0:n}^{(i)}, y_{0:n-1})$.
3)	(Optional) Resample particles from the discrete density $(w_n^{(0)}, \ldots, w_n^{(P-1)})$ and set $w_n^{(i)} = 1/P$.

Table 1. Stochastic Particle Filter employing Prior Importance Function (Bootstrap Filter).

2.2. Deterministic Particle Filters (DPF)

Stochastically drawing new particle elements as done in step 1 (Table 1) is not mandatory. In fact, this procedure introduces unwanted variabilities ([4],Chap. 2), which can be mitigated in a general framework by the use of semi-deterministic or deterministic methods, especially when the inferred variable x_n has a discrete distribution. In this special case, a set $\{x_{0:n}^{(i,j)}; \hat{w}_n^{(i,j)}\}_{i=0,j=0}^{P-1,D-1}$ with DP samples of the predictive distribution $p(x_{0:n}|y_{0:n-1})$ can be obtained deterministically as follows:

1a) Extend the trajectories $x_{0:n-1}^{(i)}$ deterministically, obtaining $x_{0:n}^{(i,j)} := \{x_{0:n-1}^{(i)}, x_n^{(j)}\}$, where $x_n^{(j)}$, $0 \le j < D-1$ are all possible values for x_n.

1b) Evaluate and normalize the weights $\hat{w}_n^{(i,j)} \propto w_{n-1}^{(i)} p(x_n^{(i,j)}|x_{0:n-1}^{(i)}, y_{0:n-1})$.

A set $\{x_{0:n}^{(i)}; w_n^{(i)}\}_{i=0}^{P-1}$ with P samples of the updated posterior density can then be obtained according to the following procedure:

2) Update and normalize the weights as $w_n^{(i,j)} \propto \hat{w}_n^{(i,j)} p(y_n|x_{0:n}^{(i,j)}, y_{0:n-1})$.

3) Apply a (deterministic or stochastic) particle selection algorithm to obtain P samples from the discrete density $\{x_{0:n}^{(i,j)}; w_n^{(i,j)}\}_{i=0,j=0}^{P-1,D-1}$.

Note that differently from stochastic algorithms, deterministic particle filters must perform particle selection (step 3) at each iteration to keep the number of particles constant. In the present comparisons, we consider classical stochastic (multinomial and residual [4]) particle selection methods (SPS) and the deterministic method (DPS) proposed in [2] which has very low computational complexity, selecting a new particle set by simply discarding the $(D-1)P$ least weighted particles of the original set and re-normalizing the weights of the remaining ones. As opposed to stochastic methods, the deterministic methods produce bias because, on average, each particle is not selected a number of times proportional to its own weight, a fact, however, that does not affect algorithm convergence properties ([4], Cap. 2).

3. JOINT BLIND EQUALIZATION AND DECODING

To state the joint blind equalization/decoding problem and to deduce the densities needed to apply particle filters to its solution, consider initially a ($1/R$ rate) convolutionally coded digital communication system that transmits BPSK symbols over a frequency selective channel subject to additive gaussian noise. Denoting the transmitted bit sequence by b_k, the coded sequence $c_k^{(m)}$, $0 \le m < R$, is obtained as

$$c_k^{(m)} = \left(\sum_{i=0}^{K} b_{k-i} d_i^{(m)} + \sum_{i=1}^{K} c_{k-i}^{(m)} r_i^{(m)} \right) \mod 2, \quad (3)$$

where K is the convolutional code constraint length and $d_i^{(m)}$ and $r_i^{(m)}$ are the code generating coefficients associated with the direct and the recursive polynomials respectively. The transmitted BPSK signal is then obtained as $s_{Rn+m} = 2c_n^{(m)} - 1$.

We assume a linear and time-invariant FIR transmission channel under perfect receiver synchronization, so that baud rate samples y_k of the received signal are expressed by the base-band equivalent model

$$y_k = \sum_{l=0}^{L-1} h_l s_{k-l} + v_k , \quad (4)$$

where h_l is the channel impulse response, L its duration in symbol intervals and v_k additive zero-mean white circular gaussian noise of variance σ_v^2. Our objective is to obtain

MAP estimates $\hat{b}_k$ of the transmitted bits given the observed data, i.e.,

$$\hat{b}_k = \arg\max_{b_k} p(b_k|\underline{y}_{0:k}), \quad (5)$$

where $\underline{y}_{0:k} := (y_0, \ldots, y_{(k+1)R-1})$.

To obtain estimates of the posterior density via both stochastic and deterministic particle filters, we need to evaluate the densities $p(b_k|b_{0:k-1}, \underline{y}_{0:k-1})$ and $p(\underline{y}_k|b_{0:k}, \underline{y}_{0:k-1})$. Under the assumption that the transmitted bits are equiprobable and IID, the first density reduces to $p(b_k)$. Moreover, as the bit sequence b_k uniquely defines the transmitted symbol sequence s_k, the second density can be determined as

$$p(\underline{y}_k|b_{0:k}, \underline{y}_{0:k-1}) = p(\underline{y}_k|S_{0:(k+1)R-1}, \underline{y}_{0:k-1}) \ , \quad (6)$$

where $S_k \triangleq [\ s_k \ \ldots \ s_{k-L+1}\ ]^T$, and $(s_0, \ldots, s_{(k+1)R-1})$ is the symbol sequence corresponding to the bit sequence $b_{0:k}$.

Exploiting the fact that $p(y_j|S_{0:j}, y_{0:j-1}) = p(y_j|S_{0:k}, y_{0:j-1})$, $k > j$, the density on the r.h.s. of (6) further decomposes as

$$p(\underline{y}_k|S_{0:(k+1)R-1}, \underline{y}_{0:k-1}) = \prod_{j=Rk}^{R(k+1)-1} p(y_j|S_{0:j}, y_{0:j-1}) \ . \quad (7)$$

To determine $p(y_j|S_{0:j}, y_{0:j-1})$ note that the above definitions allow rewriting (4) as

$$\begin{cases} S_{j+1} &= FS_j + e_1\, s_{j+1} \\ y_j &= h^H S_j + v_j \end{cases} \quad (8)$$

where F is an $(L \times L)$ shift matrix (all entries zero, except for the first subdiagonal, whose entries are ones), $e_1 = [\ 1\ 0\ \cdots\ 0\ ]^T$ and $h = [\ h_0\ \cdots\ h_{L-1}\ ]^T$.

From (8) one can see that y_j is conditionally gaussian given S_j and h. Under the assumption that the parameters h have a joint gaussian prior distribution, one obtains [5]

$$p(y_j|S_{0:j}, y_{0:j-1}) = \mathcal{N}_C\left(y_j \mid \hat{h}_{j-1}^H S_j \ ; \ S_j^H \Sigma_{j-1} S_j + \sigma_v^2\right), \quad (9)$$

where $\hat{h}_j$ and Σ_j, respectively the conditional mean and variance of h are obtained by means of conventional Kalman filter iterations:

$$\hat{h}_j = \hat{h}_{j-1} + \frac{\underline{y}_j - S_j^H \hat{h}_{j-1}}{S_j^H \Sigma_{j-1} S_j + \sigma_v^2} \Sigma_{j-1} S_j \ . \quad (10)$$

$$\Sigma_j = \Sigma_{j-1} - \frac{\Sigma_{j-1} S_j S_j^H \Sigma_{j-1}}{S_j^H \Sigma_{j-1} S_j + \sigma_v^2} \ . \quad (11)$$

As a final observation, it is worth mentioning that the algorithm described so far, in addition to providing filtered estimates, can be easily extended to determine fixed-lag smoothed estimates of the transmitted bits, since for $d > 0$ [6]:

$$p(b_{0:k}|\underline{y}_{0:k+d}) \approx \frac{\sum_{i=1}^M w_{k+d}^{(i)} \delta(b_{0:k}^{(i)} - b_{0:k})}{\sum_{i=1}^M w_{k+d}^{(i)}}. \quad (12)$$

4. SIMULATION RESULTS

To compare these methods we carried out Monte Carlo simulations in which we measured mean bit error rates (BER) over 500 independent realizations produced by a communication system that transmits BPSK (± 1) symbols over a frequency selective channel with 3 random coefficients drawn independently in each realization from the multivariate complex gaussian distribution $\mathcal{N}_c(h|0; I)$. We assumed that the additive noise is a zero-mean complex circular gaussian process with variance σ_v^2, so that the signal-to-noise ratio (SNR) is defined as

$$SNR := \|h\|^2 / 2\sigma_v^2.$$

The transmitted BPSK symbols are obtained as the non-interleaved output of the binary $1/2-$rate recursive systematic convolutional code given in polynomial notation by

$$\left[1, \frac{1 + D + D^2}{1 + D^2}\right].$$

To compute the mean BER, the algorithms processed 200 message bits in each realization, discarding the first 100 to allow for algorithm convergence. The initial particle states $S_{-1}^{(i)}$ were drawn from IID equiprobable ± 1 r.v., and we assumed that $\Sigma_{-1}^{(i)} = I$ and $h_{-1}^{(i)} \sim \mathcal{N}_C(h|0; I)$. For comparison, we also show the performance obtained by an alternative scheme, consisting of a MAP decoder fed with the symbol posterior probabilities obtained by a MAP equalizer employing the correct channel parameters.

The mean BER obtained by the proposed methods as a function of the SNR is shown in Fig. 1. In this simulation, all particle filter based algorithms employed 250 particles and a zero smoothing-lag ($d = 0$). As one can verify, the methods based on DPF outperformed their stochastic counterparts, and the algorithm employing deterministic particle selection strategy (DPF-DPS) outperformed all other blind algorithms by a large margin. Notice, however, that the performance of the proposed methods ceases to improve for SNR's larger than 10 dB. This effect was observed to persist even if a much larger number of particles is employed.

In Fig. 2 we show the performance obtained under the same conditions, adopting now a fixed smoothing lag of $d = 10$. As one may notice, the performance of the SPF and of the DPF-SPS algorithms remained largely unchanged. The performance of deterministic filter employing deterministic selection (DPF-DPS), on the other hand, was much improved, coming close to that of the trained scheme.

Finally, in Fig. 3 we evaluate the performance of the deterministic particle filter employing deterministic selection (DPF-DPS) as a function of the number of particles P. As one may notice, the performance of the proposed method is greatly improved when the number of particles P is raised

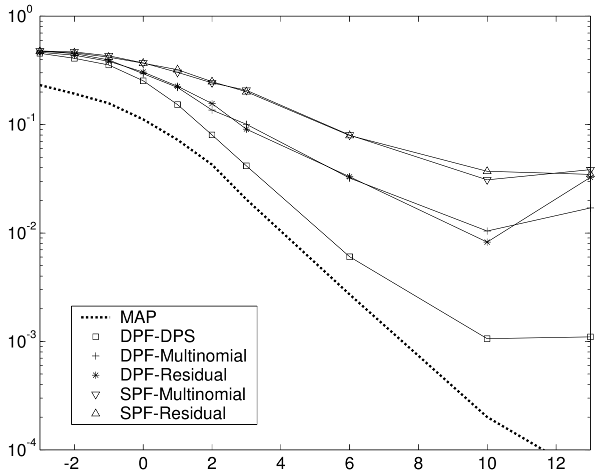

Fig. 1. Performance of the deterministic (DPF) and stochastic (SPF) joint equalization and decoding methods as a function of the SNR, employing multinomial, residual and deterministic (DPS) resampling strategies ($d = 0$), and of the alternative MAP method described in the text. The SPF algorithms employ the optimal importance function and are implemented as described in [3].

to 100, remaining practically unchanged as P is further increased.

As a final remark, it is worth mentioning that the computational complexity of the proposed DPF-DPS scheme is comparable to that of turbo joint blind equalization and decoding techniques [7]. While the proposed DPF-DPS technique requires $\mathcal{O}(PL^2)$ operations per bit, turbo techniques performing MMSE channel re-estimation require $\mathcal{O}(I(L^2 + 2^K))$, where I is the number of iterations performed (usually $I \approx 10$) and K is the code constraint length.

5. CONCLUSION

In this work we evaluated the performance of deterministic particle filtering structures for joint blindly equalizing/decoding convolutionally coded signals transmitted over frequency selective channels. Numerical simulations show that the algorithms based on deterministic particle filters outperform the stochastic alternatives, and that the performance advantage of the deterministic particle filtering method is widened if a deterministic particle selection strategy is employed.

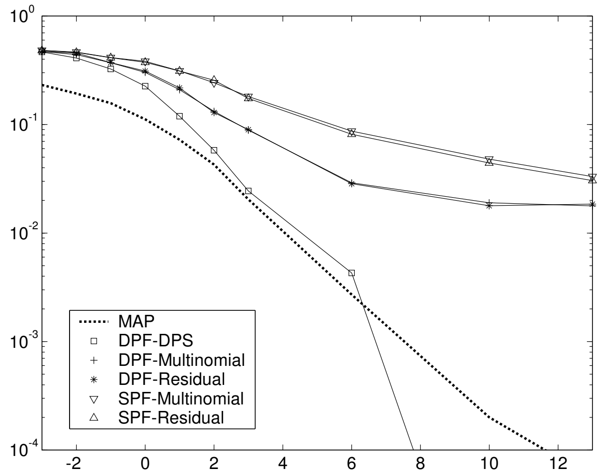

Fig. 2. Performance of the deterministic (DPF) and stochastic (SPF) joint equalization and decoding methods as a function of the SNR, employing multinomial, residual and deterministic (DPS) resampling strategies ($d = 10$), and of the alternative MAP method described in the text. The SPF algorithms employ the optimal importance function and are implemented as described in [3].

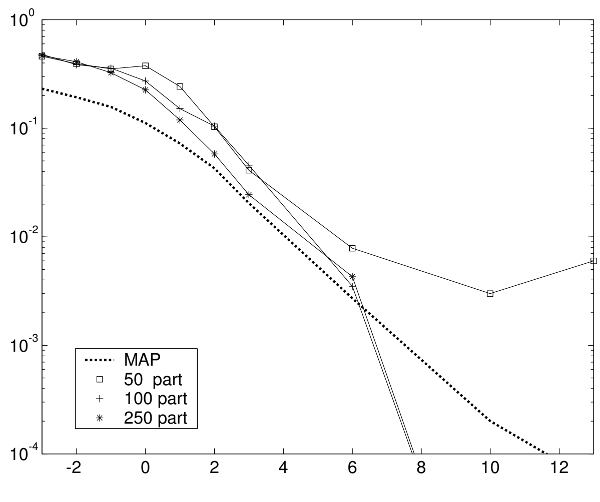

Fig. 3. Performance of the deterministic joint equalization and decoding algorithm (DPF-DPS) and of the alternative MAP method described in the text as a function of the SNR and of the number of particles ($d = 10$).

6. REFERENCES

[1] E. Punskaya, C. Andrieu, A. Doucet, and W. Fitzgerald, "Particle filtering for multiuser detection in fading cdma channels," in *Proceedings of the 11th IEEE Signal Processing Workshop on Statistical Signal Processing, 2001.*, Singapore, august 2001.

[2] B. Dong, X. Wang, and A. Doucet, "A new class of soft mimo demodulation algorithms," *IEEE Transactions on Signal Processing*, vol. 51, no. 11, pp. 2752–2763, november 2003.

[3] C. J. Bordin and L. A. Baccalá, "Particle filter algorithms for joint blind equalization/decoding of convolutionally coded signals," in *Proceedings of the 2005 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP 2005*, Philadelphia, PA - USA, 2005, vol. 3, pp. 497–500.

[4] A. Doucet, J. de Freitas, and N. Gordon, Eds., *Sequential Monte-Carlo Methods in Practice*, Springer-Verlag, Berlin, Germany, 2000.

[5] P. M. Djurić, J. H. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. F. Bugallo, and J. Miguez, "Particle filtering," *IEEE Signal Processing Magazine*, vol. 20, no. 5, pp. 19–38, september 2003.

[6] T. Clapp, *Statistical Methods for the Processing of Communication Data*, Ph.D. thesis, University of Cambridge, 2000.

[7] R. Koetter, A. C. Singer, and M. Tuchler, "Turbo Equalization," *IEEE Signal Processing Magazine*, vol. 21, no. 1, pp. 67–80, January 2004.