Particle Filtering Applied to Turbo Equalization with Imperfect Channel Estimation

Alberto Gaspar Guimarães and Ernesto Leite Pinto

Abstract— In this article we propose applying the particle filtering technique to derive new turbo equalization schemes under conditions of imperfect channel estimation. We develop a Soft Input-Soft Output (SISO) equalizer that exploits the soft information provided in the iterative process in order to perform fixed-lag smoothing of the transmitted bits, on the grounds of Sequential Importance Sampling (SIS). Preliminary simulation results of performance evaluation and comparison against another alternative recently proposed are provided, for some scenarios of channel estimation errors. It is shown that the scheme herein presented performs significantly better when the estimation error is high.

Keywords— Equalization, Turbo Equalization, Particle Filtering, Sequential Importance Sampling, Fixed-lag Smoothing.

Resumo— Neste trabalho propõe-se aplicar a técnica de filtragem de partículas para o desenvolvimento de novos esquemas de turbo equalização, sob a condição de estimação imperfeita da resposta do canal. Apresenta-se um equalizador SISO (*Soft Input-Soft Output*), que explora a informação suave produzida no processo iterativo para realizar o amaciamento de *lag* fixo dos bits transmitidos. Resultados preliminares de simulação são apresentados para avaliação do desempenho e comparação com outra alternativa recentemente proposta, considerando diferentes cenários de erro na estimação do canal. É mostrado que o esquema aqui desenvolvido apresenta resultados signficativamente superiores quando o erro de estimação é alto.

Palavras-Chave— Equalização, Turbo Equalização, Filtragem de Partículas, Amostragem por Importância Sequencial, Amaciamento de *lag* Fixo.

I. Introduction

Turbo equalization was proposed in [1] and it has been shown to be an effective technique for combating intersymbol interference (ISI). The original scheme of [1] employed a SISO equalizer and a SISO decoder, both based on the maximization of a posteriori probability (MAP) of symbols, exploiting the whole sequence of received samples within a frame. It was shown that the exchange of soft information between these two processors, after some iterations, improves the transmitted bits estimates, giving rise to a bit error rate (BER) close to 10^{-5} for a signal-to-noise ratio (SNR) of 3 dB and severe ISI.

However, the complexity of a MAP equalizer grows exponentially with the modulation constellation size and with the channel memory length. In order to circumvent this, a class of linear SISO equalizers was proposed in [2] by replacing the MAP equalizer with a transversal filter whose coefficients were

adjusted at the baud rate, using the minimum mean square error (MMSE) criterion. The proposal of [2] also includes algorithms for mapping the transversal filters outputs on soft outputs to be supplied to the decoder. In spite of presenting a performance degradation typically on the order of 3 dB for a BER of 10^{-5} in relation to the MAP equalizer, significant complexity reduction was obtained.

The aforementioned SISO equalizers proposals were derived on the basis of perfect channel estimation, which may be an unrealistic assumption for several scenarious of application, especially those characterized by time-varying channels. Aiming at dealing with channel estimation errors, a new MMSE scheme was presented in [3]. However, the proposed scheme turns out to be similar to the original MMSE turbo equalizer, the main difference being an artificial increasing in the overall noise variance in order to take into account the effect of channel estimation errors.

In order to obtain a better compromise between performance and computational cost of turbo equalization under conditions of channel estimation error, we propose in this work to exploit the ability of particle filtering to address nontrivial problems of Bayesian estimation (see [4] and the references therein) and derive a new SISO equalizer to be used under such conditions.

In brief, the particle filtering method is based on approximating the posterior probability density function (pdf) of interest by samples that are drawn from another conveniently chosen pdf, usually called *importance density* (or *importance distribution*). The samples, or "particles", are weighted by properly computed *importance weights*, and the pdf so obtained is employed for estimating the random parameters of interest. This method, also called Sequential Importance Sampling (SIS), propagates *recursively* the posterior pdf with the arrival of a new observation data by using a Bayesian update rule .

Our guess is that a well designed particle filtering scheme for SISO equalization with channel estimation errors may efficiently take advantage of the soft information provided by the SISO decoder and sequentially give back extrinsic information to this decoder. Following this reasoning, we derive in this work a fixed-lag smoothing of the transmitted symbols taking into account the statistics of channel estimation errors, on the grounds of SIS theory. In respect of the importance distribution, we derive for the problem of interest the importance function proposed in [5], which was shown to be optimum in the sense of minimizing the conditional variances of the importance weights.

Some preliminary results of bit error rate (BER) performance evaluation of the propoed turbo equalizer are presented, using Kalman filtering for channel estimation, under different

conditions of signal-to-noise ratio and duration of the estimation period. These results show that the proposed SISO equalizer is an effective tool for dealing wih imperfect channel estimation within turbo equalizers. It is also shown that the scheme herein proposed outperforms the MMSE equalizer proposed in [3], when the channel estimation errors get more significant.

The remaining of the paper is organized as follows. The communication system model adopted in this work is presented in Section II. A description of the proposed SISO equalizer is given in Section III. A performance evaluation and comparison with the MMSE turbo equalizer of [6] is provided in Section IV. Finally, concluding remarks are given in section V.

II. SYSTEM MODEL

We consider the following baseband transmission system model. The binary data sequence to be transmitted is encoded using a Forward Error Correction (FEC) code and randomly interleaved before being mapped to a Q-ary signal constellation. The resulting data sequence is partitioned into blocks (frames) of length P and known symbols (training sequence) are added to the encoded sequence in order to enable the receiver to estimate the channel impulse response.

The symbol frames are transmitted over a channel which introduces additive noise and ISI. The relationship between the receiver input y_n and the transmitted symbols can then be given by

$$y_n = \mathbf{x}_n^T \mathbf{h} + \omega_n, \qquad (1)$$

where $\mathbf{h} = [h_1, \ldots, h_L]^T$ represents the baseband channel impulse response which is assumed to be of finite length L and with a fading rate sufficiently slow so that $\mathbf{h}$ may be considered as time-invariant during the frame duration. The vector $\mathbf{x}_n = [x_n, x_{n-1}, \ldots, x_{n-L+1}]^T$ contains the transmitted symbols from time instant n up to $n - L + 1$, and ω_n is a zero-mean circularly complex white Gaussian noise, independent of $\{x_n\}$, with variance denoted by σ_ω^2. We assume that the probability mass function (pmf) $p(x_n)$ of each transmitted data symbol is known to the equalizer, corresponding to the soft information provided by the decoder. Moreover, assuming that an ideal interleaver is used, the transmitted symbols x_n are considered to be mutually independent.

At the receiver, the channel impulse response is not available and a channel estimation technique, using the received training sequence, is employed to provide to the equalizer an estimate $\hat{\mathbf{h}}$ of the channel response and the error covariance matrix of the estimation given by $\mathbf{Q_h} = E[(\mathbf{h} - \hat{\mathbf{h}})(\mathbf{h} - \hat{\mathbf{h}})^T]$.

III. SIS FIXED-LAG SMOOTHING FOR TURBO EQUALIZATION WITH ESTIMATED CHANNEL IMPULSE RESPONSE

In this section our aim is to develop a novel SISO equalization scheme based on a particle filtering algorithm. We consider as inputs the set of observations $\{y_n\}_0^P$, the a priori probability of symbols $p(x_n)$, the estimated channel response $\hat{\mathbf{h}}$ and its corresponding error covariance matrix $\mathbf{Q_h}$. As output, the equalizer forwards to the decoder the *extrinsic*

information by computing the *a posteriori* pmf (APP) of each symbol given the estimated channel impulse response. That is, provided that the true channel impulse response is not available to the receiver, our approach is to compute the APP conditioned to the estimate of $\mathbf{h}$ obtained from the training period.

We adopt here a SIS fixed-lag smoothing methodology, i.e., we compute recursively $p(x_n|y_{1:n+M}, \hat{\mathbf{h}})$, where $M > 0$ is some fixed delay, by assuming that the a posteriori *joint* pmf of symbols $x_{1:n} = \{x_1, \ldots, x_n\}^1$ can be approximated by

$$p(x_{1:n}|y_{1:n+M}, \hat{\mathbf{h}}) \approx \sum_{i=1}^{N} \lambda_n^i \delta(x_{1:n} - x_{1:n}^i) \ , \qquad (2)$$

where λ_n^i, $i = 1, \ldots, N$ ($N >> 1$), is the importance weight associated to the i-th particle trajectory $x_{1:n}^i$ generated by importance sampling, and $\delta(.)$ is the Kronecker delta. Each weight at time instant n is computed from [4]

$$\lambda_n^i \propto \frac{p(x_{1:n}^i|y_{1:n+M}, \hat{\mathbf{h}})}{q(x_{1:n}^i|y_{1:n+M}, \hat{\mathbf{h}})} \ , \qquad (3)$$

where $q(.)$ is the importance distribution. Further, if we assume that this density can be factored as

$$q(x_{1:n}|y_{1:n+M}, \hat{\mathbf{h}}) =$$
$$q(x_n|x_{1:n-1}, y_{1:n+M}, \hat{\mathbf{h}})q(x_{1:n-1}|y_{1:n+M-1}, \hat{\mathbf{h}}) \ , \quad (4)$$

and observing that the *posterior* pmf of symbols can be decomposed into

$$p(x_{1:n}|y_{1:n+M}, \hat{\mathbf{h}}) =$$
$$= p(x_n|x_{1:n-1}, y_{1:n+M}, \hat{\mathbf{h}})p(x_{1:n-1}|y_{1:n+M}, \hat{\mathbf{h}})$$
$$= \frac{p(x_n|x_{1:n-1}, y_{1:n+M}, \hat{\mathbf{h}})p(y_{n+M}|x_{1:n-1}, y_{1:n+M-1}, \hat{\mathbf{h}})}{p(y_{n+M}|y_{1:n+M-1}, \hat{\mathbf{h}})}$$
$$\times p(x_{1:n-1}|y_{1:n+M-1}, \hat{\mathbf{h}})$$
$$(5)$$

we may recursively obtain the weight λ_n^i according to

$$\lambda_n^i \propto$$
$$\propto \frac{p(x_n^i|x_{1:n-1}^i, y_{1:n+M}, \hat{\mathbf{h}})p(y_{n+M}|x_{1:n-1}^i, y_{1:n+M-1}, \hat{\mathbf{h}})}{q(x_n^i|x_{1:n-1}^i, y_{1:n+M}, \hat{\mathbf{h}})}$$
$$\times \underbrace{\frac{p(x_{1:n-1}^i|y_{1:n+M-1}, \hat{\mathbf{h}})}{q(x_{1:n-1}^i|y_{1:n+M-1}, \hat{\mathbf{h}})}}_{\lambda_{n-1}^i}$$
$$(6)$$

In order to minimize the conditional variance of the importance weights (as shown in [5]), we choose the following importance distribution

$$q(x_n|x_{1:n-1}^i, y_{1:n+M}, \hat{\mathbf{h}}) = p(x_n|x_{1:n-1}^i, y_{1:n+M}, \hat{\mathbf{h}}) \ . \quad (7)$$

So the importance weight is evaluated by

$$\lambda_n^i \propto p(y_{n+M}|x_{1:n-1}^i, y_{1:n+M-1}, \hat{\mathbf{h}})\lambda_{n-1}^i \ . \qquad (8)$$

---
[1]A similar notation will be further used for other variables, e.g., $y_{1:n} = \{y_1, \ldots, y_n\}$.

We now address the computation of (7) and (8), and next we present the LLR evaluation from the posterior distribution of symbols.

A. Evaluation of Importance pmf

Following the principle of SIS, at each time instant n and for each trajectory $x_{1:n-1}^i$, we must draw a new particle x_n^i according to (7). This pmf can be evaluated by

$$p(x_n|x_{1:n-1}^i, y_{1:n+M}, \hat{\mathbf{h}}) =$$
$$= \frac{p(y_{n:n+M}|x_n, x_{1:n-1}^i, y_{1:n-1}, \hat{\mathbf{h}})p(x_n|x_{1:n-1}^i, y_{1:n-1}, \hat{\mathbf{h}})}{p(y_{n:n+M}|x_{1:n-1}^i, y_{1:n-1}, \hat{\mathbf{h}})}$$
$$\propto p(y_{n:n+M}|x_n, x_{1:n-1}^i, y_{1:n-1}, \hat{\mathbf{h}})p(x_n) . \quad (9)$$

The pmf $p(x_n)$ is known, whereas the computation of the function $p(y_{n:n+M}|x_n, x_{1:n-1}^i, y_{1:n-1})$ requires some algebraic manipulations. To perform it we initially write

$$p(y_{n:n+M}|x_n, \mathbf{x}_{1:n-1}^i, y_{1:n-1}, \hat{\mathbf{h}}) =$$
$$\sum_{x_{n+1}} \cdots \sum_{x_{n+M}} \left[\prod_{k=1}^{M} p(x_{n+k}) \right] \times p(y_{n:n+M}|\hat{\mathbf{h}}, \boldsymbol{\theta}_n^i) \quad (10)$$

where we have denoted $\boldsymbol{\theta}_n^i \triangleq \{x_{n:n+M}, x_{1:n-1}^i, y_{1:n-1}\}$. We assume that $p(\mathbf{h}|\hat{\mathbf{h}}, \boldsymbol{\theta}_n^i) = p(\mathbf{h}|\hat{\mathbf{h}})$ and this density is Gaussian with parameters $(\hat{\mathbf{h}}, \mathbf{Q_h})$ provided by the channel estimation algorithm. Therefore, given eq. (1), we may readily see that[2]

$$p(y_{n:n+M}|\hat{\mathbf{h}}, \boldsymbol{\theta}_n^i) =$$
$$= \mathbb{N}(y_{n:n+M}; \mathbf{X}_{n+M}^i \hat{\mathbf{h}}, \mathbf{X}_{n+M}^i \mathbf{Q_h}(\mathbf{X}_{n+M}^i)^H + \boldsymbol{\Lambda_y}), \quad (11)$$

where

$$\mathbf{X}_{n+M}^i = \begin{bmatrix} (\mathbf{x}_{n+M}^i)^T \\ (\mathbf{x}_{n+M-1}^i)^T \\ \vdots \\ (\mathbf{x}_n^i)^T \end{bmatrix} \quad (12)$$

being $\mathbf{x}_{n+j}^i = [x_{n+j}^i, x_{n+j-1}^i, \ldots, x_{n+j-L+1}^i]$ and $\boldsymbol{\Lambda_y} = \sigma_\omega^2 \mathbf{I}_{M+1}$. The matrix $\mathbf{I}_{M+1}$ is the $(M+1) \times (M+1)$ identity matrix. When defining $\mathbf{X}_{n+M}^i$, we have set $\mathbf{x}_m^i \triangleq [\tilde{x}_m, \tilde{x}_{m-1}, \ldots, \tilde{x}_{m-M+1}]^T$, for $n + M \leq m \leq n$, where $\tilde{x}_k = x_k^i$ if $k \leq n-1$ and $\tilde{x}_k = x_k$ if $n \leq k \leq n+M$.

Combining the expressions (9), (10) and (11), we finally arrive at the importance distribution

$$p(x_n|\mathbf{x}_{1:n-1}^i, y_{1:n+M}, \hat{\mathbf{h}}) =$$
$$\frac{p(x_n) \sum_{x_{n+1}} \cdots \sum_{x_{n+M}} \left[\prod_{k=1}^{M} p(x_{n+k}) \right] \mathbb{N}(y_{n+M:n}; \boldsymbol{\mu_y}^i, \boldsymbol{\Sigma_y}^i)}{\sum_{x_n} p(x_n) \sum_{x_{n+1}} \cdots \sum_{x_{n+M}} \left[\prod_{k=1}^{M} p(x_{n+k}) \right] \mathbb{N}(y_{n+M:n}; \boldsymbol{\mu_y}^i, \boldsymbol{\Sigma_y}^i)},$$
$$(13)$$

where $\boldsymbol{\mu_y}^i \triangleq \mathbf{X}_{n+M}^i \hat{\mathbf{h}}$ and $\boldsymbol{\Sigma_y}^i \triangleq \mathbf{X}_{n+M}^i \mathbf{Q_h}(\mathbf{X}_{n+M}^i)^H + \boldsymbol{\Lambda_y}$.

[2]$\mathbb{N}(\boldsymbol{\alpha}, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes a Gaussian pdf with argument $\boldsymbol{\alpha}$ and parameters $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$.

B. Evaluation of Importance Weight

We may easily see that the *incremental* weight given by $\tilde{\lambda}_n^i \triangleq p(y_{n+M}|x_{1:n-1}^i, y_{1:n+M-1}, \hat{\mathbf{h}})$ may be computed from

$$\tilde{\lambda}_n^i = \frac{p(y_{n:n+M}|x_{1:n-1}^i, y_{1:n-1}, \hat{\mathbf{h}})}{p(y_{n:n+M-1}|x_{1:n-1}^i, y_{1:n-1}, \hat{\mathbf{h}})} . \quad (14)$$

It is worthy to note that the numerator of (14) is equal to the denominator of (13), i.e.,

$$p(y_{n:n+M}|x_{1:n-1}^i, y_{1:n-1}, \hat{\mathbf{h}}) =$$
$$\sum_{x_n} \cdots \sum_{x_{n+M}} \left[\prod_{k=0}^{M} p(x_{n+k}) \right] \mathbb{N}(y_{n+M:n}; \boldsymbol{\mu_y}^i, \boldsymbol{\Sigma_y}^i) \quad (15)$$

therefore the additional computation burden when evaluating the weight by using (14) is due solely to the computation of $p(y_{n:n+M-1}|x_{1:n-1}^i, y_{1:n-1}, \hat{\mathbf{h}})$, which may be done by

$$p(y_{n:n+M-1}|x_{1:n-1}^i, y_{1:n-1}, \hat{\mathbf{h}}) =$$
$$\sum_{x_n} \cdots \sum_{x_{n+M-1}} \left[\prod_{k=0}^{M-1} p(x_{n+k}) \right] \mathbb{N}(y_{n+M-1:n}; \boldsymbol{\mu_{y-}}^i, \boldsymbol{\Sigma_{y-}}^i) ,$$
$$(16)$$

in which $\boldsymbol{\mu_{y-}}^i$ is the $M \times 1$ subvector of $\boldsymbol{\mu_y}^i$ with the first element missing, and $\boldsymbol{\Sigma_{y-}}^i$ is the $M \times M$ submatrix of $\boldsymbol{\Sigma_y}^i$ with the first row and first column missing.

Substituting (15) and (16) into (14) we then can write the following expression to compute the incremental weights

$$\tilde{\lambda}_n^i = \frac{\sum_{x_n} \cdots \sum_{x_{n+M}} \left[\prod_{k=0}^{M} p(x_{n+k}) \right] \mathbb{N}(y_{n+M:n}; \boldsymbol{\mu_y}^i, \boldsymbol{\Sigma_y}^i)}{\sum_{x_n} \cdots \sum_{x_{n+M-1}} \left[\prod_{k=0}^{M-1} p(x_{n+k}) \right] \mathbb{N}(y_{n+M-1:n}; \boldsymbol{\mu_{y-}}^i, \boldsymbol{\Sigma_{y-}}^i)} .$$
$$(17)$$

In order to lessen the computational complexity when evaluating (17), we may compute both the inverse and the determinant of $\boldsymbol{\Sigma_y}^i$ from the values found for $\boldsymbol{\Sigma_{y-}}^i$ [7].

C. LLR Evaluation

Once evaluated the symbol posterior pmf, the equalizer with the help of the demapper [8] is able to forward to the decoder the extrinsic information about each coded bit c_n, i.e.,

$$L_E(c_n) = ln \frac{p(c_n = 1|y_{1:n+M}, \hat{\mathbf{h}})}{p(c_n = 0|y_{1:n+M}, \hat{\mathbf{h}})} - ln \frac{p(c_n = 1)}{p(c_n = 0)}, \quad (18)$$

where the second term in the right-hand side of (18) is the *a priori* LLR received from the decoder. Nevertheless, because of the intrinsic randomness of the particle filtering method, it is possible that at any moment all drawn particles representing the support of the posterior pdf yield $p(c_n = 1|y_{1:n+M}, \hat{\mathbf{h}}) = 1$ or $p(c_n = 0|y_{1:n+M}, \hat{\mathbf{h}}) = 1$, leading to a $L_E(c_n)$ of infinity modulus. In fact, this is likely to happen when the turbo process evolves and the a priori LLR received from the decoder becomes high. The solution adopted here in order to circumvent this problem, whenever it occurs, is to artificially assign

$$L_E(c_n) = 4\phi(c_n) \quad (19)$$

where the function ϕ is defined as $\phi(1) = 1$ and $\phi(0) = -1$. Assigning $L_E(c_n)$ with such value, the corresponding a priori probability of c_n received by the decoder is $P(c_n = \phi^{-1}(L_E(c_n)/4)) = e^4/(1 + e^4) \approx 0.982$. That is, this information to the decoder somehow confirms its belief (expressed by the high LLR value sent to the equalizer) about the likelihood of c_n. We remark that (19) was set empirically and further performance improvement can be attained by a careful investigation of this issue, which will be carried out in the continuation of this work.

IV. SIMULATION AND RESULTS

We present in this section the performance evaluation of the SISO equalizer proposed in this paper, embedded in a turbo equalization receiver which estimates the channel impulse response through a preamble of known symbols. For the sake of comparison, we implement as well a turbo receiver with the MMSE SISO equalizer as proposed in [3], which also incorporates channel estimation errors into the equalization algorithm.

The following setup was used. A set of 256 data bits is randomly generated and encoded using a 1/2-rate convolutional encoder with generator $(1 + D^2, 1 + D + D^2)$. Next the coded bits are interleaved and mapped to ± 1 symbols (BPSK). To the sequence of 512 symbols (plus the coder overhead), we add a preamble of known BPSK symbols. In order to evaluate the receivers performances under different degrees of channel estimation precision, we used 20 and 40-symbol preambles in the simulation runs. The frame is then transmitted over a time-invariant ISI channel of length 3, whose coefficients are independent samples of a zero-mean Gaussian distribution with unitary variance. Finally, zero-mean white Gaussian noise with variance $\sigma_\omega^2 = N_0/2$ is added to the received symbols.

At the receiver, the channel impulse response is estimated by using a Kalman algorithm where the considered process and observation equations are respectively given by $h_{n+1} = h_n$ and by eq. (1). The received preamble is used as the set of observations to the estimation algorithm. Figure 1 depicts the Mean Square Error (MSE) of channel estimation ($MSE = E[(\mathbf{h} - \hat{\mathbf{h}})^T(\mathbf{h} - \hat{\mathbf{h}})]$) when the preamble length is varied at $E_b/N_0 =$ 0, 2 and 3 dB. Since $\mathbf{h}$ is a priori Gaussian , and a Kalman filter is used as the channel estimator with Gaussian observation noise, we remark that $p(\mathbf{h}|\hat{\mathbf{h}})$ is Gaussian and the analysis developed in III is well adjusted to the simulation scenario.

Given the estimated channel response and the error covariance matrix, the equalizer outputs soft information on each unknown symbol, which are then deinterleaved and decoded by using a BCJR algorithm. The fixed-lag particle smoothing algorithm is implemented with $M = 3$, and $N = 50$ samples are used to approximate the posterior symbol pmf. With the aim of combatting the so-called degeneracy problem of particle filtering, we used the method of resampling [4][9] whenever the effective sample size ($N_{eff} \approx (\sum_{i=1}^{N}(\lambda_n^i)^2)^{-1}$) falls below $N/3$. This threshold was set empirically. By its turn, the linear MMSE equalization is carried out using 9 noncausal and 5 causal coefficients.

The BER performances of the turbo receiver with the Particle Filter Fixed Lag (PF-FL) and MMSE equalizers are shown in Figures 2 and 3, respectively for the cases where the frame preamble contains 20 and 40 symbols. The BER plots are made at various E_b/N_0 ratios, corresponding to the results obtained for the first and fourth iterations of the turbo process, after the transmission of 500 blocks (i.e. 128,000 bits). For each transmitted block, the channel impulse response is independently drawn as described before.
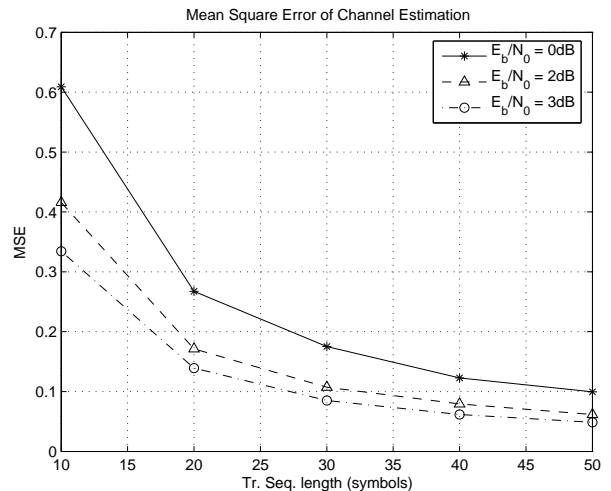

Fig. 1. MSE of channel estimation.

From the Figure 2 it can be seen that the two schemes have similar performance for the first iteration, but the receiver with the PF-based equalizer presents superior performance after four iterations of the turbo process. At a BER of 1.5×10^{-3}, there is a difference of about 1 dB between the two plots, and this gap tends to increase for higher signal-to-noise ratios.

When a preamble of 40 symbols is used to estimate the channel impulse response, the MSE of channel estimation decreases to approximately the half of the value obtained for a 20-symbol preamble, as shown in Figure 1. In this case, Figure 3 shows that after 4 iterations the turbo receiver using the MMSE equalizer outperforms our proposal, with a performance difference of about 0.5 dB at $BER \approx 6 \times 10^{-5}$. That is, when the channel error is less significant, the approximations introduced in [3] have less impact on the overall performance of MMSE equalizer.

From Figure 3 it is also interesting to note that the scheme here proposed presents better results than the MMSE receiver in the first iteration, except at low values of SNR. This behavior may indicate that the LLR mapping defined in eq. (19) is somehow degradating the soft information flow between the equalizer and the decoder. As mentioned before, this point requires a deeper investigation which will be done in future works.

V. CONCLUSIONS

We have developed a novel technique for SISO equalization with imperfect channel estimation, using the particle filtering approach with fixed-lag smoothing. Provided that the channel

estimation error is a Gaussian random vector with known covariance matrix, the algorithm is able to incorporate the estimation error in the the evaluation of the posterior symbol pmf. By contrast, other proposals such as the MMSE-based equalizer shown in [3], deal with the channel errors considering some simplifying assumptions into the original method. The simulation results showed that when the channel estimation error is significant, the scheme presented here outperforms the proposal of [3]. When the error decreases, our approach looses performance when compared to the MMSE-based receiver, possibly due to a misadjusted LLR mapping. This is an initial work which have shown promising results, and further developments and investigations will be soon carried out in future submissions.
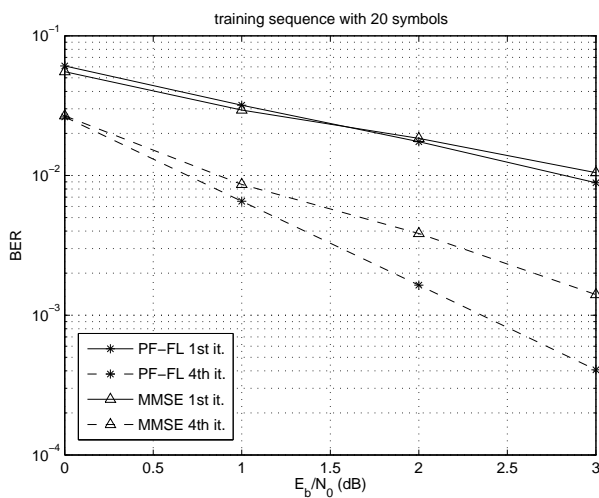

Fig. 2.   BER performance of turbo receivers with PF-FL and MMSE equalizers when the training sequence has 20 symbols.
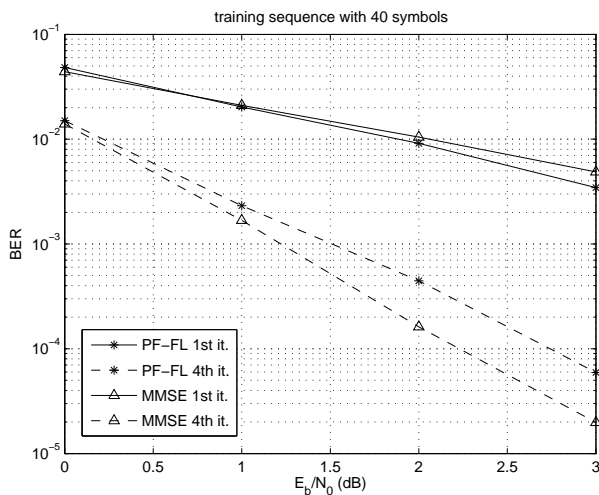

Fig. 3.   BER performance of turbo receivers with PF-FL and MMSE equalizers when the training sequence has 40 symbols.

REFERENCES

[1] C. Douillard, M. Jezequel, C. Berrou, A. Picart, P. Didier, and A. Glavieux, "Iterative correction of intersymbol interference: Turbo equalization," in *European Trans. Telecomm.*, Aug. 1995, pp. 507–511.
[2] M. Tüchler, A. C. Singer, and R. Koetter, "Minimum mean squared error equalization using a priori information," *IEEE Trans. Signal Processing*, vol. 50, no. 3, pp. 673–683, Mar. 2002.
[3] R. Otnes and M. Tüchler, "Block SISO linear equalizers for turbo equalization in serial-tone HF modems," in *Proc. Norwegian Signal Processing Symposium (NORSIG-2001)*, Oct. 2001.
[4] M. S. Arulampalam, S. Maskell, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Processing*, vol. 50, pp. 174–188, Feb. 2002.
[5] A. Doucet, "On sequential Monte Carlo sampling methods for Bayesian filtering," Dept. Eng. Univ. Cambridge, UK, Tech. Rep., 1998.
[6] R. Otnes and M. Tüchler, "Low-complexity turbo equalization for time-varying channels," in *Proc. IEEE Veh. Technol Conf (Spring)*, May 2002, pp. 140–144.
[7] F. A. Graybill, *Matrices with Applications in Statistics.*   USA: Wadsworth, Inc., 1983.
[8] M. Tüchler, R. Koetter, and A. Singer, "Turbo equalization," *IEEE Signal Processing Mag.*, pp. 67–80, 2004.
[9] N. Gordon, D. Salmond, and A. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEE Proceedings-F*, vol. 140, pp. 107–113, Apr. 1993.