

PARTICLE FILTERING FOR ITERATIVE DATA AND PHASE ESTIMATION

Didier Le Ruyet[†], Tanya Bertozzi^{} and Nicolas Paul[†]*

[†] Electronique et Communications, CNAM, 292 rue Saint Martin, 75141 Paris Cedex 3, France

Telephone: 0033 1 40 27 27 98, Fax: 0033 1 40 27 29 94, Email: leruyet@cnam.fr

^{*} Diginext, 45 Impasse de la Draille, 13857 Aix en Provence Cedex 3, France,

Telephone: 0033 4 42 90 82 82, Fax: 0033 4 42 90 82 80, Email: bertozzi@diginext.fr

ABSTRACT

In this paper we consider the iterative decoding of channels with strong phase noise. We propose to estimate the phase using a message passing algorithm based on particle filtering. Instead of estimating the phase directly we use the whole estimated probability density function for the channel decoder. Three versions of phase estimators are derived. The simulation results are given for binary LDPC codes. We compare the proposed iterative data and phase estimators with another solution where the forward backward algorithm is performed over the trellis obtained from the discretization of the phase. The results show that the proposed algorithm achieves a good compromise between performances and complexity.

1. INTRODUCTION

Phase tracking is an important issue in coherent digital communications. Since estimating jointly the phase and the data is generally intractable, a number of suboptimal algorithms have been proposed using a phase estimator followed by a data detector. In order to improve the performance, it is possible to perform an iterative estimation of the phase and data using for example the Expectation Maximisation (EM) algorithm [1]. However, this algorithm is efficient only if the phase is constant during the frame.

In this paper we consider iterative decoding for channels with strong phase noise. Several iterative decoding algorithms based on a Bayesian approach have been considered in the literature for this case. For example, it is possible to perform the the MAP algorithm over the trellis obtained using a phase discretization approach [2].

Particle filtering (PF) [3] represents the most powerful approach for the sequential estimation of the hidden state of a nonlinear dynamic model. During these last years, PF has been widely applied to digital communication (channel estimation, tracking, synchronisation, ...) [4]

In [5], the authors have proposed to apply the PF for the phase estimation. Instead of using a particle filter to estimate

the phase, we propose in this paper to use the whole estimated probability density function of the phase for the channel decoder. Compared to the phase discretization approach, the particle filter performs a discretization of the phase space around the most interesting region. Consequently, it allows to reduce the complexity of the algorithm.

2. SYSTEM MODEL

We consider the following equivalent baseband complex channel model at the receiver:

$$r_k = c_k e^{j\theta_k} + n_k \quad k = 0, 1, \dots, K-1, \quad (1)$$

where c_k is the channel input symbol at time k taken from a M-PSK constellation \mathcal{X} , n_k is a complex gaussian noise with variance σ^2 . θ_k is the phase noise and can be modelled with a random walk:

$$\theta_k = (\theta_{k-1} + \Delta_k) \bmod 2\pi, \quad (2)$$

where Δ_k is a white Gaussian noise with variance σ_Δ^2 . As a consequence, we have:

$$\begin{aligned} p(\theta_k | \theta_{k-1}, \dots, \theta_0) &= p(\theta_k | \theta_{k-1}) \\ &\propto \exp \left\{ -\frac{1}{2\sigma_\Delta^2} (\theta_k - \theta_{k-1})^2 \right\} \end{aligned} \quad (3)$$

We also add an efficient code (turbo code or LDPC).

3. MESSAGE PASSING ALGORITHM

The decision problem is given by:

$$\begin{aligned} \hat{c}_k &= \arg \max_{c_k} p(c_k | \mathbf{r}) \\ &= \arg \max_{c_k} \int_{\boldsymbol{\theta}} p(c_k, \boldsymbol{\theta} | \mathbf{r}) d\boldsymbol{\theta} \\ &= \arg \max_{c_k} \int_{\boldsymbol{\theta}} \sum_{\mathbf{c}: c_i \neq k} p(\mathbf{c}, \boldsymbol{\theta} | \mathbf{r}) d\boldsymbol{\theta} \end{aligned} \quad (4)$$

The work of Nicolas Paul was supported by France Telecom R&D

Factor graphs [7] are an useful tool to describe and derive message passing algorithms for digital communication The factor graph corresponding to the considered problem is given in figure 1.

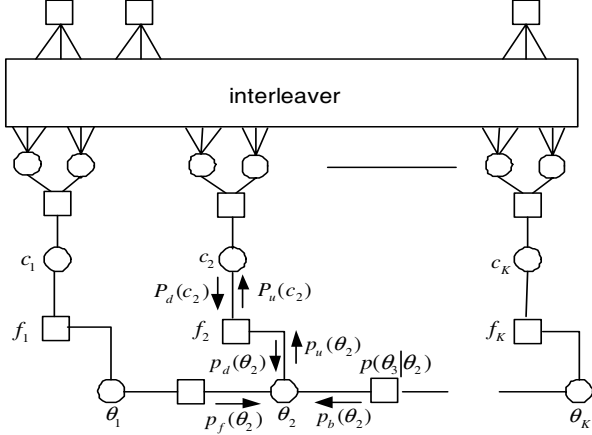


Fig. 1. Factor graph of the LDPC code and the channel model

$p(\mathbf{c}, \boldsymbol{\theta} | \mathbf{r})$ can be factorized as follows:

$$\begin{aligned}
 p(\mathbf{c}, \boldsymbol{\theta} | \mathbf{r}) &\propto p(\mathbf{c})p(\boldsymbol{\theta})p(\mathbf{r} | \boldsymbol{\theta}, \mathbf{c}) \\
 &\propto p(\boldsymbol{\theta}) \prod_{k=0}^{K-1} p(r_k | c_k, \theta_k) \\
 &\propto p(\boldsymbol{\theta}) \prod_{k=0}^{K-1} \exp \left\{ -\frac{1}{2\sigma^2} |r_k - c_k e^{j\theta_k}|^2 \right\} \\
 &\propto p(\theta_0) \prod_{k=1}^K p(\theta_k | \theta_{k-1}) \\
 &\quad \times \prod_{k=0}^{K-1} \exp \left\{ -\frac{1}{2\sigma^2} |r_k - c_k e^{j\theta_k}|^2 \right\}
 \end{aligned} \quad (5)$$

Instead of performing the exact ML decision rule, we will apply the particle filter along the given factor graph.

Since we are not interested in studying the implementation based on the structure of the channel code, we will only concentrate on the message propagation in the lower part of the graph.

Following [8], the message from the variable node c_k to the factor node f_k will be denoted $P_d(c_k)$ and the message from f_k to θ_k will be denoted $p_d(\theta_k)$. The function $f_k(c_k, \theta_k)$ associated to the factor node f_k is given by:

$$f_k(c_k, \theta_k) = p(r_k | c_k, \theta_k) \quad (6)$$

We have:

$$\begin{aligned}
 p_d(\theta_k) &\propto p(r_k | \theta_k) \\
 &\propto \sum_{x \in \mathcal{X}} P_d(c_k = x) f_k(c_k = x, \theta_k).
 \end{aligned} \quad (7)$$

We adopt a forward-backward node activation schedule to describe the phase-noise evolution. In this scheme we need to analyze two types of messages. On the one hand, the forward messages from factor node $p(\theta_k | \theta_{k-1})$ to variable node θ_k are denoted by $p_f(\theta_k)$. These messages can be recursively computed as follows:

$$p_f(\theta_k) \propto \int_0^{2\pi} p_d(\theta_{k-1}) p_f(\theta_{k-1}) p(\theta_k | \theta_{k-1}) d\theta_{k-1} \quad (8)$$

On the other hand, the backward messages sent from factor node $p(\theta_{k+1} | \theta_k)$ to variable node θ_k are denoted by $p_b(\theta_k)$ and can be recursively computed by:

$$p_b(\theta_k) \propto \int_0^{2\pi} p_d(\theta_{k+1}) p_f(\theta_{k+1}) p(\theta_{k+1} | \theta_k) d\theta_{k+1}. \quad (9)$$

Initially, the forward and backward messages are uniformly distributed. Then the message $p_u(\theta_k)$ from the variable node θ_k to the factor node f_k is obtained by combining the forward and backward messages:

$$p_u(\theta_k) \propto p_f(\theta_k) p_b(\theta_k). \quad (10)$$

Finally, the message from the factor node to variable node c_k is indicated by $P_u(c_k)$ and is calculated as follows:

$$P_u(c_k) \propto \int_0^{2\pi} p_u(\theta_k) p(r_k | c_k, \theta_k) d\theta_k. \quad (11)$$

After processing these messages, the message $P_u(c_k)$ are processed by the upper part of the factor graph. Then updated messages $P_d(c_k)$ are available for bit decoding or for update estimation of $P_u(\theta_k)$ and $P_u(c_k)$.

In [2], the authors propose to discretize the phase space into L equally spaced phases and then apply the forward backward algorithm [9] to approximate $p_f(\theta_k)$ and $p_b(\theta_k)$. However, at each time in this case the complexity is proportional to L^2 (number of branches of the associated trellis). We thus propose to reduce this complexity by using PF.

4. PARTICLE FILTER FOR THE PHASE ESTIMATION

Particle filtering methods, known also as sequential Monte Carlo methods, are powerful tools for numerical computing optimal estimators when exact solutions cannot be analytically derived [3]. They employ discrete measures with random supports, the particles, for sequentially estimating posterior probability distributions of the unknowns. The particles

allow a continuous exploration of the phase space thanks to their evolution in time. Furthermore, the complexity of PF is directly proportional to the number of particles.

Using a particle filter, $p_f(\theta_k)$ is approximated with weighted particles as follows:

$$p_f(\theta_k) \approx \sum_{i=1}^{N_{part}} \tilde{w}_k^{(i)} \delta(\theta_k - \theta_k^{(i)}) \quad (12)$$

where N_{part} is the number of particles, $\tilde{w}_k^{(i)}$ is the normalized importance weight at time k associated with the particle i and $\delta(\theta_k - \theta_k^{(i)})$ denotes the Dirac delta centered in $\theta_k = \theta_k^{(i)}$.

Initially, all the N_p particles are drawn randomly from $\theta_0 \sim \text{uniform}(-\pi, +\pi)$. At time k , the set of particles from phase space Θ_k is obtained from the particle set Θ_{k-1} . The time evolution of the particles is achieved with an importance sampling distribution.

For simplicity reasons, we choose the a prior importance function [6]:

$$\pi(\theta_k^{(i)} | \theta_{1:k-1}^{(i)}, r_{1:k}) = p(\theta_k^{(i)} | \theta_{k-1}^{(i)}). \quad (13)$$

Considering that the noise Δ_k in (2) is Gaussian, the importance function is a Gaussian distribution with mean $\theta_{k-1}^{(i)}$ and variance σ_Δ^2 . To determine the positions of the particles at time k from the positions at time $k-1$, each particle is drawn according to the importance function (13). The evaluation of the importance function for each particle at time n enables the calculation of the importance weights [6]:

$$w_k^{(i)} = w_{k-1}^{(i)} \frac{p(r_k | \theta_k^{(i)}) p(\theta_k^{(i)} | \theta_{k-1}^{(i)})}{\pi(\theta_k^{(i)} | \theta_{1:k-1}^{(i)}, r_{1:k})}. \quad (14)$$

Using the prior importance function (13), we obtain:

$$\begin{aligned} w_k^{(i)} &= w_{k-1}^{(i)} p(r_k | \theta_k^{(i)}) \\ &\propto w_{k-1}^{(i)} p_d(\theta_k^{(i)}) \end{aligned} \quad (15)$$

A major problem with PF is that the discrete random measure degenerates quickly: all the particles except for a very few are assigned negligible weights. The degeneracy implies that the particle filter performance will deteriorate. In order to reduce this degeneracy, we add a resampling step. Resampling is performed whenever the effective particle size N_{eff} is less than a threshold N_{thresh} [3].

In this paper we will consider three phase estimation versions based on the particle filter:

- MMSE phase estimation,
- forward only phase pdf estimation,
- forward backward phase pdf estimation.

In the MMSE phase estimation, from $p_f(\theta_k)$ approximated by the particle filter, we estimate $\hat{\theta}_k$ using the MMSE estimate:

$$\hat{\theta}_k = \sum_{i=1}^{N_p} \tilde{w}_k^{(i)} \theta_k^{(i)} \quad (16)$$

where $\tilde{w}_k^{(i)}$ are the normalized weight obtained from $w_k^{(i)}$ and $P_u(c_k)$ is simply given by :

$$P_u(c_k) \approx \exp \left\{ -\frac{1}{2\sigma^2} |r_k - c_k e^{j\hat{\theta}_k}|^2 \right\} \quad (17)$$

In the forward only phase pdf (FOPP) estimation, we also do not calculate $p_b(\theta_k)$. We approximate $p_u(\theta_k)$ without considering the backward message :

$$p_u(\theta_k) \approx p_f(\theta_k) \quad (18)$$

From $p_f(\theta_k)$ we evaluate $P_u(c_k)$:

$$P_u(c_k) \approx \sum_{i=1}^{N_p} \tilde{w}_k^{(i)} p(r_k | c_k, \theta_k^{(i)}) \quad (19)$$

In the forward backward phase pdf (FBPP) estimation, we perform a second particle filter in the backward direction in order to calculate $p_b(\theta_k)$:

$$p_b(\theta_k) \approx \sum_{i=1}^{N_{part}} \tilde{w}_k^{(i)} \delta(\theta_k - \theta_k^{(i)}) \quad (20)$$

From (2), since we consider a i.i.d. gaussian noise, we can rewrite the evolution equation as follows

$$\theta_k = (\theta_{k+1} + \Delta_k) \mod 2\pi, \quad (21)$$

The calculation of the non normalized weight is given by:

$$w_k^{f(i)} \propto w_{k+1}^{f(i)} p_d(\theta_k^{(i)}) \quad (22)$$

The calculation of $p_u(\theta_k)$ as given in equation (11) is difficult since the support of the two particle filters are different. A solution is provided in [10] however the computational complexity is quadratic in the number of particles $O(KN_{part}^2)$. In order to reduce the complexity to $O(KN_{part})$, we propose to simplify the calculation of $p_u(\theta_k)$ by combining both supports as follows :

$$p_u(\theta_k) \approx \frac{1}{2} (p_f(\theta_k) + p_b(\theta_k)) \quad (23)$$

Then $P_u(c_k)$ is given by :

$$P_u(c_k) \approx \frac{1}{2} \sum_{i=1}^{N_p} \tilde{w}_k^{(i)} p(r_k | c_k, \theta_k^{(i)}) + \frac{1}{2} \sum_{i=1}^{N_p} \tilde{w}_k^{f(i)} p(r_k | c_k, \theta_k^{f(i)}) \quad (24)$$

This solution is clearly suboptimal but performs well in practice.

5. SIMULATION RESULTS

In this section we present the performance of the proposed PF algorithm. The considered code is a (3,6)-regular LDPC code with codewords of length 4000. The chosen modulation is QPSK. We assume a pilot symbol in every block of 20 transmitted symbols. We considered a severe phase noise model with $\sigma_{\Delta} = 6$ [8]. For all the simulations, we perform 10 global iterations between the channel decoder and the phase estimator. 20 iterations over the LPDC graph are performed at each global iterations.

In Figure 2 we give the performance obtained using the MMSE, FOPB and FBPP phase estimation algorithm. We also add the performance curves of the discretization method using $L = 20$ and $L = 32$ equispaced phases. The performance obtained for $L = 32$ can be considered as the maximum achievable performance [2].

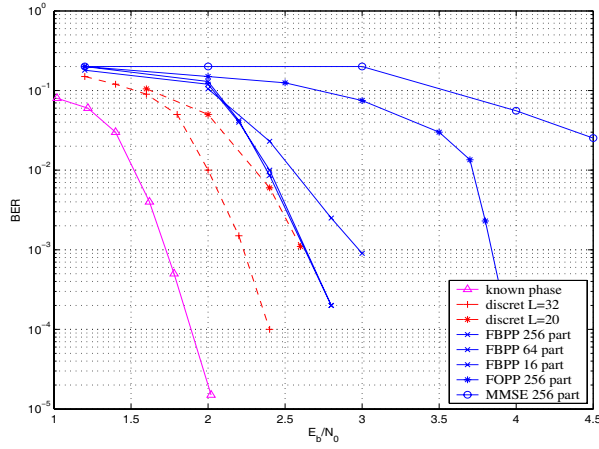


Fig. 2. Performance of the algorithm based on discretization of the phase and PF algorithm

The simulation results show that the $N_{part} = 256$ MMSE and FOPP phase estimation algorithms perform poorly compared to the other algorithms. The performance obtained for the $N_{part} = 64$ FBPP phase estimation algorithm is close to the one of the $L = 20$ discretization method and is about 0.4 dB far from the $L = 32$ discretization method at $BER = 10^{-3}$. However, from the complexity point of view, we only calculate $N_{part} = 64$ branch metrics compared to $L^2 = 400$ branch metrics at each instant for the discretization method. Interestingly, there is only a performance degradation of about 0.25 dB between $N_{part} = 16$ and $N_{part} = 256$ FBPP phase estimation algorithm.

6. CONCLUSIONS

In this paper we have considered the iterative decoding of channels with strong phase noise. We have proposed to estimate the phase using a message passing algorithms based

on particle filtering. The forward backward phase pdf estimation significantly reduces the computational complexity with the same or a slight performance degradation. It is of interest to extend these results by evaluating the two particle filter smoother [10] after having reduce its complexity.

7. REFERENCES

- [1] N. Noels, H. Steendam, M. Moeneclaey, "Carrier phase recovery in turbo receivers: Cramer-Rao bound and synchronizer performance," *Proc. Symposium on turbo codes 2003*. Brest, France. pp. 323-326, Sept 2003.
- [2] M. Peleg, S. Shamai (Shitz), S. Galan, "Iterative decoding for coded noncoherent MPSK communications over phase-noisy AWGN channel," *IEEE Proc. Commun.*, Vol. 147, pp. 87-95, Apr 2000.
- [3] A. Doucet, J. F. G. de Freitas and N. J. Gordon Eds., *Sequential Monte Carlo methods in practice*. New York: Springer-Verlag, 2001.
- [4] P. M. Djuric, J. M. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. F. Bugallo and J. Miguez, "Applications of particle filtering to selected problems in communications" *IEEE Signal Processing Magazine*, Vol. 20, pp. 19-38, Sept. 2003.
- [5] J. Dauwels, H. A. Loeliger, "Phase estimation by message passing", *IEEE Proc. Int. Conf. on Communications*, Vol. 1, pp. 20-24, 2004.
- [6] A. Doucet, S. Godsill and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statistics and Computing*, Vol. 10, No. 3, pp. 197-208, 2000.
- [7] F. R. Kschischang, B.J. Frey, H. A. Loeliger "Factor graphs and the sum product algorithm," *IEEE Trans. on Info. Theory*, Vol. 47, pp. 498-519, feb 2001.
- [8] G. C. Colavolpe, A. Barbieri, Giuseppe Caire, "Iterative decoding in the presence of strong phase noise," *submitted to IEEE Journal on Select. Areas in Communications* 2004.
- [9] L. R. Bahl, J. Cocke, F. Jelinek, J. Raviv. "Optimal decoding of linear codes for minimizing symbol error rate" *IEEE Trans. Inform. Theory*. Vol.20, pp. 284-287, March 1974.
- [10] M. Briers, A. Doucet, S. Maskell "Smoothing algorithms for state space models", *submitted to IEEE trans. On signal processing* 2005