

# Iterative detection in MIMO channels using particle filtering

Tanya Bertozzi <sup>\*†</sup> and Didier Le Ruyet<sup>†</sup>

<sup>\*</sup> DIGINEXT, 45 Impasse de la Draille, 13857 Aix en Provence Cedex 3, France,

Telephone: +33 (0)4 42 90 82 82, Email: bertozzi@diginext.fr

<sup>†</sup> CNAM, 292 rue Saint Martin, 75141 Paris Cedex 3, France

Telephone: +33 (0)1 58 80 84 91, Email: leruyet@cnam.fr

## Abstract

To achieve near-capacity in systems employing multiple antennas, an iterative detection scheme must be used. A full A Posteriori Probability (APP) MIMO detector is associated with a convolutional code or a turbo code. However, the APP inner detector presents quickly a prohibitive complexity and schemes known as list sphere decoders have been developed to reduce this complexity. In this paper, we provide a new reduced-complexity MIMO list detector based on the Particle Filtering (PF) methods. This detector can be seen as a tree-search detector, which uses the a priori information to explore the paths of the tree. We show that the average number of paths analyzed by the PF detector decreases at each iteration. Moreover, it is adapted according to the signal-to-noise ratio. We compare the performance and the complexity of the PF detector and the list sphere decoder.

## I. INTRODUCTION

In a Multiple-Input-Multiple-Output (MIMO) system, the optimal detector that minimizes the average error probability is the Maximum Likelihood (ML) detector, if the channel matrix is perfectly known at the receiver. However, this detector cannot be often realized, because its exponential complexity becomes quickly prohibitive [1]-[2].

The ML detection can be interpreted as a search problem of the closest point in a lattice [1] and different algorithms, generally known as sphere decoders [3]-[4]-[5] -[6], have been developed to achieve this search with a polynomial complexity. For a very complete survey, see [7]. In [8], a list version of sphere decoder is considered to iteratively detect and decode the data stream in a Vertical-Bell Labs layered Space-Time (V-BLAST) scheme [9], combined with an outer channel code. This method approximates the optimal joint detector-decoder to achieve near-capacity with low complexity. In [2], the detection problem analyzed by

[8] is seen as a search of the most likely path in a tree. Hence, a list sequential decoding technique based on the stack algorithm is developed and compared with the list sphere decoder of [8].

In this paper, we propose to solve the detection problem of [8] through the tree described in [2], by using the Particle Filtering (PF) methods [10]-[11]. In fact, the PF methods applied to a discrete state space can be interpreted as a reduced-complexity tree-search algorithm, as shown in [12]-[13]. Furthermore, the PF methods present the advantage to exploit the a priori information and to provide the extrinsic information in a natural way [14]. In [15], the PF methods have been already applied to a V-BLAST structure in a non-iterative manner. Moreover, the absence of the tree-search interpretation prevents the comparison with the existing algorithms.

This paper is organized as follows. In Section II the system model is presented. The structure of the PF detector is derived in Section III. Finally, the performance and the computational complexity of the proposed algorithm are given and compared in Section IV.

## II. SYSTEM MODEL

Fig. 1 shows the considered iterative V-BLAST scheme with  $N_T$  transmit and  $N_R$  receive antennas. We consider  $N_R \geq N_T$ . The data stream  $\mathbf{u}$  is encoded with an error correcting code and interleaved to obtain the vector  $\mathbf{c} = [\mathbf{c}_1, \dots, \mathbf{c}_{N_T}]^T$  of coded bits. Each element of  $\mathbf{c}$  is  $\mathbf{c}_i = [c_{i1}, \dots, c_{iM}]$  for  $i = 1, \dots, N_T$ . We assume that the binary digits  $c_{ij}$  take independent values from  $\{+1, -1\}$ , where  $+1$  is the *null* element in GF(2). The sequence  $\mathbf{c}$  is mapped into the complex vector  $\mathbf{s} = [s_1, \dots, s_{N_T}]^T$  of symbols. Each element  $s_i = \text{map}(\mathbf{c}_i)$  is a symbol of a complex modulation (such as BPSK, QPSK, 16-QAM, ...) and  $M$  is the number of bits per symbol. For each transmit antenna, the transmitted energy is  $E_s/N_T$ , where  $E_s$  is the energy per symbol, such that the total transmitted energy through the  $N_T$  transmit antennas is independent from  $N_T$ .

The received vector is described by the complex MIMO matrix model:

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{n}, \quad (1)$$

where  $\mathbf{y} = [y_1, \dots, y_{N_R}]^T$  is the vector of received complex symbols,  $\mathbf{H}$  is the channel matrix of dimension  $N_R \times N_T$  with each element being an independent realization of a zero-mean complex Gaussian random variable of variance  $1/2$  per real dimension and  $\mathbf{n} = [n_1, \dots, n_{N_R}]^T$  is a vector of independent zero-mean complex Gaussian noise with variance  $\sigma^2$  per real dimension. We assume that  $\mathbf{H}$  is perfectly known at the receiver. The model (1) is equivalent to the following real-valued equation:

$$\mathbf{r} = \mathbf{B}\mathbf{x} + \mathbf{v}, \quad (2)$$

where  $\mathbf{r}$  is a  $2N_R \times 1$  vector obtained by substituting each element of  $\mathbf{y}$  with  $[\text{Re}(y_i) \text{ Im}(y_i)]^T$ ,  $\mathbf{x}$  and  $\mathbf{v}$  are built in a similar way from  $\mathbf{s}$  and  $\mathbf{n}$  respectively and  $\mathbf{B}$  is a  $2N_R \times 2N_T$  matrix obtained from  $\mathbf{H}$  by replacing each element  $h_{ij}$  by the matrix:

$$\begin{bmatrix} \text{Re}(h_{ij}) & -\text{Im}(h_{ij}) \\ \text{Im}(h_{ij}) & \text{Re}(h_{ij}) \end{bmatrix}. \quad (3)$$

In the search of the closest lattice point, the expression (2) is typically rewritten in a different manner. By performing the Gram-Schmidt orthonormalization of the columns of  $\mathbf{B}$  or, equivalently, by applying the QR decomposition on  $\mathbf{B}$ ,  $\mathbf{B}$  assumes the form:

$$\mathbf{B} = [\mathbf{Q}\tilde{\mathbf{Q}}] \begin{bmatrix} \mathbf{R} \\ \mathbf{0} \end{bmatrix}, \quad (4)$$

where  $\mathbf{Q}$  and  $\tilde{\mathbf{Q}}$  are unitary matrices of dimensions  $2N_R \times 2N_T$  and  $2N_R \times (2N_R - 2N_T)$  respectively,  $\mathbf{R}$  is a  $2N_T \times 2N_T$  upper triangular matrix with positive diagonal element and  $\mathbf{0}$  is a  $(2N_R - 2N_T) \times 2N_T$  zero matrix. By applying (4) in (2) and by multiplying by  $\mathbf{Q}^T$ , the model (2) becomes:

$$\tilde{\mathbf{r}} = \mathbf{R}\mathbf{x} + \tilde{\mathbf{v}}, \quad (5)$$

where  $\tilde{\mathbf{r}} = \mathbf{Q}^T \mathbf{r}$  and  $\tilde{\mathbf{v}} = \mathbf{Q}^T \mathbf{v}$ . Thanks to this decomposition, we can first detect  $\text{Im}(s_{N_T})$ , then  $\text{Re}(s_{N_T})$  given  $\text{Im}(s_{N_T})$ ,  $\text{Im}(s_{N_T-1})$  given  $\text{Im}(s_{N_T})$  and  $\text{Re}(s_{N_T})$  and so on, up to  $\text{Re}(s_1)$ . Hence, the ML detection of the symbols  $s_i$  can be performed on a tree with  $2N_T$  levels [2].

### III. THE MIMO PARTICLE DETECTOR

The PF methods, even known as sequential Monte Carlo methods, represent the most powerful approach for the sequential estimation of the hidden state of a nonlinear dynamic model [10]-[11]. The solution to this problem depends on the knowledge of the Posterior Probability Density (PPD) of the hidden state given the observations. Except in a few special cases, it is impossible to analytically calculate a sequential expression of this PPD. It is necessary to adopt numerical approximations. The PF methods provide an iterative approximation of the PPD by weighted points or particles that evolve in the state space. Therefore, these methods give a discrete approximation of the continuous space of the hidden state.

According to (5), the aim of the PF methods is to perform the Maximum A Posteriori (MAP) criterion on the sequence  $x_{1:2N_T} = \{x_1, \dots, x_{2N_T}\}$ :

$$\hat{x}_{1:2N_T} = \arg \max_{x_{1:2N_T}} p(x_{1:2N_T} | \tilde{\mathbf{r}}_{1:2N_R}), \quad (6)$$

where  $x_1 = \text{Im}(s_{N_T}), x_2 = \text{Re}(s_{N_T}), \dots, x_{2N_T} = \text{Re}(s_1)$  and  $\tilde{r}_{1:2N_R} = \{\tilde{r}_1, \dots, \tilde{r}_{2N_R}\}$  is the sequence of the element of  $\tilde{\mathbf{r}}$  taken in an inverse order. We assume that  $N_R = N_T = N/2$  and we use the QPSK modulation. Then, each symbol  $x_n$  of the sequence  $x_{1:2N_T}$  takes its value in the finite set  $\{+1/\sqrt{2}, -1/\sqrt{2}\}$ . The extensions to the case  $N_R > N_T$  and to more complex modulations are straightforward. The PF methods approximate sequentially the PPD in (6) with weighted particles:

$$p(x_{1:N}|\tilde{r}_{1:N}) \approx \sum_{i=1}^{N_p} \tilde{w}_N^{(i)} \delta(x_N - x_N^{(i)}) \cdots \delta(x_1 - x_1^{(i)}), \quad (7)$$

where  $N_p$  is the number of particles,  $\tilde{w}_N^{(i)}$  is the normalized importance weight at the level  $N$  associated with the particle  $i$ ,  $\delta(x_n - x_n^{(i)})$  denotes the Dirac delta centered in  $x_n = x_n^{(i)}$  and  $x_n^{(i)}$  is the value of  $x_n$  associated with the particle  $i$ . This approximation is achieved by using the Sequential Importance Sampling and Resampling (SISR) technique [10].

We observe that the PF methods are applied to the discrete state space of the data. In this case, the PF methods can be seen as a tree-search algorithm [12]-[13], where the particles explore the state space in groups. An example of particle tree is represented in Fig. 2.

Each group of particles at a level  $n$  in the tree is characterized by:

- the sequence  $x_{1:n}^{(i)} = x_1^{(i)}, \dots, x_n^{(i)}$ , where the index  $i$  refers to all the particles contained in the considered group;
- the non-normalized importance weight  $w_n^{(i)}$ ;
- the normalized importance weight  $\tilde{w}_n^{(i)}$ ;
- the number of the particles in the group.

The root of the tree is a group containing all the particles  $N_p$ . The non-normalized importance weights  $w_0^{(i)}$  are initialized to 1 and the normalized importance weights  $\tilde{w}_0^{(i)}$  to  $1/N_p$ . The state space is discrete. Then, the particles can only be moved towards the possible states of the system. In fact, for each group at a level  $n - 1$  two values are possible for the current symbol  $x_n$ :  $+1/\sqrt{2}$  and  $-1/\sqrt{2}$ . Hence, the particles will be divided in two groups. In the conventional SISR algorithm, the evolution of the particles is achieved with an importance sampling distribution. For the transition from level  $n - 1$  to level  $n$ , the particles are drawn according to the importance function  $\pi(x_n^{(i)}|x_{1:n-1}^{(i)}, \tilde{r}_{1:n})$ . We choose the optimal importance function, which minimizes the variance of the importance weights of the particles [10]:

$$\pi(x_n^{(i)}|x_{1:n-1}^{(i)}, \tilde{r}_{1:n}) = p(x_n^{(i)}|x_{1:n-1}^{(i)}, \tilde{r}_n). \quad (8)$$

In this case, the particles cannot be drawn because the states are discrete. However, the particles of a group at level  $n - 1$  are divided proportionally to the importance function in two groups at level  $n$ . Owing to the

indivisibility of the particles, it can occur that some groups are empty. The path associated with an empty group is eliminated.

We calculate (8) for  $x_n = +1/\sqrt{2}$ . The calculation for  $x_n = -1/\sqrt{2}$  is similar. By applying the Bayes theorem, the importance function assumes the form:

$$\begin{aligned} p(x_n^{(i)} = +1/\sqrt{2} | x_{1:n-1}^{(i)}, \tilde{r}_n) \\ = \frac{p(\tilde{r}_n | x_{n,+}^{(i)}) p(x_n^{(i)} = +1/\sqrt{2} | x_{1:n-1}^{(i)})}{\int p(\tilde{r}_n | x_n^{(i)}, x_{1:n-1}^{(i)}) p(x_n^{(i)} | x_{1:n-1}^{(i)}) dx_n^{(i)}}, \end{aligned} \quad (9)$$

where  $x_{n,+}^{(i)} = (x_n^{(i)} = +1/\sqrt{2}), x_{1:n-1}^{(i)}$ . Since we can consider that the symbols transmitted by the different antennas are independent, we can write:

$$P_{n,+} = p(x_n^{(i)} = +1/\sqrt{2}) = p(x_n^{(i)} = +1/\sqrt{2} | x_{1:n-1}^{(i)}). \quad (10)$$

Therefore, (9) becomes:

$$\begin{aligned} p(x_n^{(i)} = +1/\sqrt{2} | x_{1:n-1}^{(i)}, \tilde{r}_n) \\ = \frac{p(\tilde{r}_n | x_{n,+}^{(i)}) P_{n,+}}{p(\tilde{r}_n | x_{n,+}^{(i)}) P_{n,+} + p(\tilde{r}_n | x_{n,-}^{(i)}) P_{n,-}}. \end{aligned} \quad (11)$$

From (5), we observe that the probability density  $p(\tilde{r}_n | x_{n,+}^{(i)})$  is Gaussian with mean  $x_{n,+}^{(i)}$  and variance  $\sigma^2$ . The same observation is valid for  $p(\tilde{r}_n | x_{n,-}^{(i)})$ . Moreover, the probabilities  $P_{n,+}$  and  $P_{n,-}$  are calculated from the a priori information about the bits  $c_{lm}$ , where  $l = 1, \dots, N/2$  and  $m = 1, 2$  for the QPSK modulation and Gray mapping between the bits and the symbols. Then, the exploration of the paths in the tree depends on the a priori information.

Since we use the optimal importance function, the non-normalized importance weights are given by [10]:

$$w_n^{(i)} = w_{n-1}^{(i)} p(\tilde{r}_n | x_{1:n-1}^{(i)}), \quad (12)$$

where:

$$p(\tilde{r}_n | x_{1:n-1}^{(i)}) = p(\tilde{r}_n | x_{n,+}^{(i)}) P_{n,+} + p(\tilde{r}_n | x_{n,-}^{(i)}) P_{n,-}. \quad (13)$$

The normalized importance weights are calculated by:

$$\tilde{w}_n^{(i)} = \frac{w_n^{(i)}}{\sum_{j=1}^{N_p} w_n^{(j)}}. \quad (14)$$

This algorithm presents a degeneracy phenomenon. After a few iterations of the algorithm, only one particle has a normalized weight almost equal to 1 and the other weights are very close to zero. Therefore, a

large computational effort is devoted to updating paths without almost no contribution to the final estimate. In order to avoid this behavior, different strategies of resampling of the particles have been developed [11]. For this application, we choose the following procedure: if the number of group with one particle is greater than 35% of the particles, if the number of groups is greater than 50% of the particles and we are not at the final level  $N$ , the particles are uniformly redistributed according to their normalized weights. After a resampling phase, the non-normalized weights are reinitialized to 1 and the normalized weights to  $1/N_p$ . For a justification of this choice, see [13].

The PF algorithm can be summarized in the scheme shown in Fig. 3. In this scheme,  $G_n$  denotes the number of groups at level  $n$ ,  $g_n$  is the generic index associated with a group at level  $n$  and  $N_p^{g_n}$  the number of particles in the group  $g_n$ .

At the final level  $N$ , the PF detector provide the extrinsic information about the bits  $c_{lm}$  from their A Posteriori Probabilities (APP). The APP of the bit  $c_{lm}$  is expressed in the form of a log-likelihood ratio (L-value):

$$L(c_{lm}|\tilde{r}_{1:N}) = \ln \frac{p(c_{lm} = 1|\tilde{r}_{1:N})}{p(c_{lm} = 0|\tilde{r}_{1:N})}. \quad (15)$$

The probabilities in (15) are approximated by the particles as follows:

$$p(c_{lm} = 1|\tilde{r}_{1:N}) \approx \sum_{i \in S_-} \tilde{w}_N^{(i)} \delta(x_{N-2l-m+3} - x_{N-2l-m+3}^{(i)}) \quad (16)$$

and

$$p(c_{lm} = 0|\tilde{r}_{1:N}) \approx \sum_{i \in S_+} \tilde{w}_N^{(i)} \delta(x_{N-2l-m+3} - x_{N-2l-m+3}^{(i)}), \quad (17)$$

where  $S_-$  and  $S_+$  represent the sets of particles with  $x_{N-2l-m+3} = -1/\sqrt{2}$  and  $x_{N-2l-m+3} = +1/\sqrt{2}$  respectively and we have taken into account that the PF detector detects first the imaginary part of the symbol transmitted through the antenna  $N/2$ , then its real part, the imaginary part of the symbol transmitted through the antenna  $N/2 - 1$ , its real part and so on. The well known relation that permits the calculation of the extrinsic information  $L_E(c_{lm}|\tilde{r}_{1:N})$  from the APP and the a priori information  $L_A(c_{lm})$  is:

$$L_E(c_{lm}|\tilde{r}_{1:N}) = L(c_{lm}|\tilde{r}_{1:N}) - L_A(c_{lm}). \quad (18)$$

#### IV. SIMULATION RESULTS

In this section, we give simulation results representing the performance of the proposed PF detector. The adopted performance measure is the Bit Error Rate (BER) versus the Signal-to-Noise Ratio (SNR)  $E_b/N_0$ ,

where  $E_b$  denotes the signal energy per transmitted information bit and  $N_0/2 = \sigma^2$  the double-sided noise power spectral density. The relation between  $E_b$  and  $E_s$  is the following [8]:

$$\left. \frac{E_b}{N_0} \right|_{\text{dB}} = \left. \frac{E_s}{N_0} \right|_{\text{dB}} + 10 \log_{10} \frac{N_R}{R N_T M}, \quad (19)$$

where  $R$  is the rate of the outer code. We assume an ergodic channel with 8 transmit and 8 receive antennas. The channel matrix  $\mathbf{H}$ , perfectly known at the receiver, changes independently at each channel use. In order to compare our performance with the results of [8], the transmission is organized in blocks of 9216 information bits. The outer code is a Parallel Concatenated Convolutional (PCC) code composed of two (7,5) convolutional codes. The code rate  $R = 1/2$  is obtained by puncturing. Then, considering QPSK modulation ( $M = 2$ ), (19) becomes:

$$\left. \frac{E_b}{N_0} \right|_{\text{dB}} = \left. \frac{E_s}{N_0} \right|_{\text{dB}}. \quad (20)$$

For each block of information bits, we perform 4 global iteration for the MIMO detection and 8 iteration for the iterative decoding of the PCC code. The full tree of the MIMO detector has  $2^{16}$  possible paths and we apply the PF methods in order to analyze only a subset of these paths. Fig. 4 shows the performance of the PF detector with 1000 particles for the 4 global iterations. We can observe the improvement of the performance due to the turbo scheme. For this detector, in Fig. 5 we give the average number of the paths at the end of the tree considered for the calculation of the extrinsic information. Since the exploration of the tree in the PF detector depends on the a priori information, at each iteration the average number of paths decreases significantly. Moreover, this average number is adapted according to  $E_b/N_0$ .

In Fig. 6 we compare the performance of the PF detector with the one of the list sphere decoder presented in [8], both obtained at the fourth iteration. The performance of the list sphere decoder with a number of analyzed paths of 256, 128, 64, 32, 16 and the one of the PF detector with 1000 and 500 particles are considered. The respective average numbers of paths analyzed by the PF detector are shown in Fig. 7.

Thanks to the decreasing of the average number of path at each iteration, the PF detector with 1000 particles reaches the  $\text{BER} = 3.3 \cdot 10^{-5}$  at 2.8 dB by analyzing only 1.0019 paths during the fourth iteration. Instead, the list sphere decoder considers always 128 paths at each iteration for the same BER. If we reduce the number of particles to 500, we can observe at  $\text{BER} = 10^{-4}$  a performance degradation of 0.46 dB with respect to the one of the PF detector with 1000 particles. This loss of performance is due to the decrease of the number of paths analyzed by the PF detector during the first iteration, even if the average number of paths at the fourth iteration is always equal to about 1.

Although the PF detector represents a suboptimal solution, it achieves near-ML performance with a lower complexity with respect to the list sphere decoder.

## V. CONCLUSION

In this paper, we have proposed to use the PF methods as reduced-complexity MIMO detector in an iterative turbo scheme. We have developed a soft-input soft-output PF MIMO detector and we have shown how this algorithm can be seen as a reduced-complexity tree-search algorithm. We have compared the performance and the complexity of the proposed detector with the ones of the list sphere decoder presented in [8]. We have observed that the computational complexity of the PF detector is adapted according to the signal-to-noise ratio. Furthermore, since the exploration of the paths in the tree depends on the a priori information, we have noted that the average number of analyzed paths decreases at each iteration of the turbo detection structure. Then, we have shown that we can reach the same performance of the list sphere decoder with lower complexity.

## REFERENCES

- [1] E. Agrell, T. Eriksson, A. Vardy and K. Zeger, "Closest point search in lattices," *IEEE Trans. Inform. Theory*, Vol. 48, pp. 2201–2214, August 2002.
- [2] S. B  ro, J. Hagenauer and M. Witzke, "Iterative detection of MIMO transmission using a list-sequential (LISS) detector," *Proc. ICC*, Anchorage, Alaska, May 2003.
- [3] U. Fincke and M. Pohst, "Improved methods for calculating vectors of short length in a lattice, including a complexity analysis," *Mathematics of Computation*, Vol. 44, pp. 463–471, April 1985.
- [4] C. P. Schnorr and M. Euchner, "Lattice basis reduction: improved practical algorithms and solving subset sum problems," *Mathematical Programming*, Vol. 66, pp. 181–191, 1994.
- [5] E. Viterbo and J. Boutros, "A universal lattice code decoder for fading channels," *IEEE Trans. Inform. Theory*, Vol. 45, pp. 1639–1642, July 1999.
- [6] M. O. Damen, A. Chkeif and J. -C. Belfiore, "Lattice codes decoder for space-time codes," *IEEE Commun. Lett.*, Vol. 4, pp. 161–163, May 2000.
- [7] M. O. Damen, H. El Gamal and G. Caire, "On maximum likelihood detection and search for the closest lattice point," submitted to *IEEE Trans. Inform. Theory*, 2003.
- [8] B. M. Hochwald and S. ten Brink, "Achieving near-capacity on a multiple-antenna channel," *IEEE Trans. Commun.*, Vol. 51, pp. 389–399, March 2003.
- [9] P. W. Wolniansky, G. J. Foschini, G. D. Golden and R. A. Valenzuela, "V-BLAST: an architecture for realizing very high data rates over the rich-scattering wireless channel," *Proc. of URSI ISSSE*, pp. 295–300, Pisa, Italy, 1998.
- [10] 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.
- [11] A. Doucet, J. F. G. de Freitas and N. J. Gordon, *Sequential Monte Carlo methods in practice*. New York: Springer-Verlag, 2001.
- [12] T. Bertozzi, D. Le Ruyet, G. Rigal and H. Vu-Thien, "Joint data-channel estimation using the particle filtering on multipath fading channels," *Proc. ICT*, Papeete, Tahiti, February 2003.
- [13] T. Bertozzi, D. Le Ruyet, G. Rigal and H. Vu-Thien, "On particle filtering for digital communications," *Proc. SPAWC*, Roma, Italy, June 2003.
- [14] D. Guo and X. Wang, "Blind detection in MIMO systems via sequential Monte Carlo," *IEEE J. Select. Areas Commun.*, Vol. 21, pp. 464–473, April 2003.



- [15] Y. Huang, J. Zhang and P. M. Djurić, "Detection with particle filtering in BLAST systems," *Proc. ICC*, Anchorage, Alaska, May 2003.

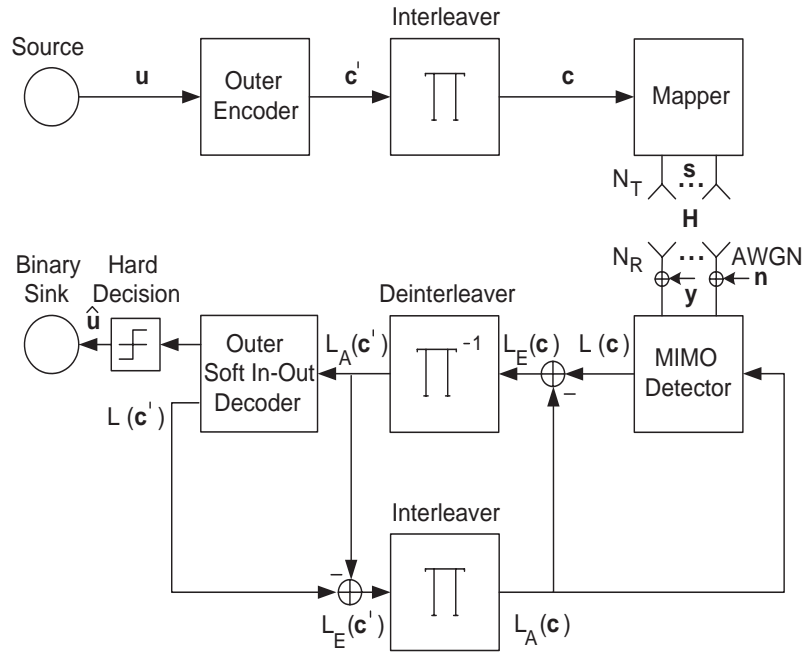


Fig. 1. Iterative V-BLAST transmission model.

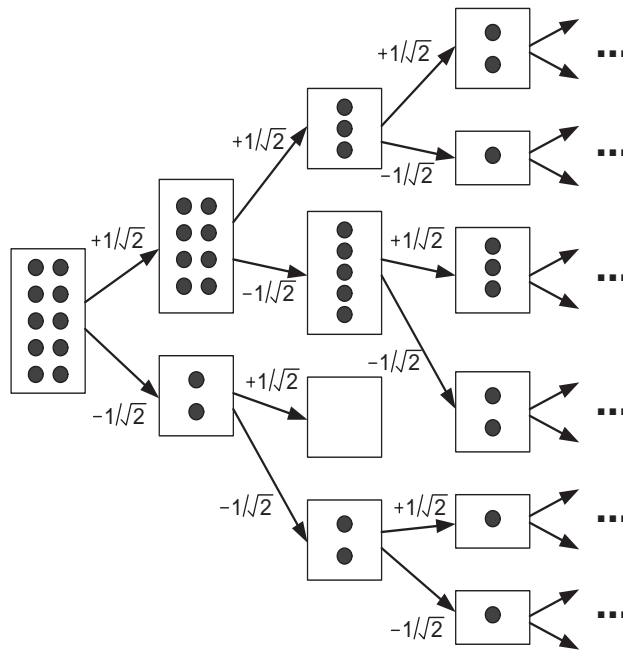


Fig. 2. An example of particle tree for the QPSK modulation.

### The PF algorithm

- *Initialize:*

$$G_0 = 1;$$

$$w_0^{(i)} = 1, \quad \forall i \in 1_0;$$

$$\tilde{w}_0^{(i)} = 1/N_p, \quad \forall i \in 1_0;$$

$$N_p^{1_0} = N_p.$$

For  $n = 1, \dots, N$ :

- *For  $g_{n-1} = 1, \dots, G_{n-1}$ :*

- *Calculate  $p(x_n^{(i)} = +1/\sqrt{2}|x_{1:n-1}^{(i)}, \tilde{r}_n)$  and*

- $p(x_n^{(i)} = -1/\sqrt{2}|x_{1:n-1}^{(i)}, \tilde{r}_n)$  using (11);*

- *Divide the particles of  $g_{n-1}$  in the two groups*

- $(2g-1)_n$  and  $(2g)_n$  proportionally to*

- $p(x_n^{(i)} = +1/\sqrt{2}|x_{1:n-1}^{(i)}, \tilde{r}_n)$  and*

- $p(x_n^{(i)} = -1/\sqrt{2}|x_{1:n-1}^{(i)}, \tilde{r}_n)$ ;*

- *Update  $N_p^{(2g-1)_n}$  and  $N_p^{(2g)_n}$ ;*

- *Update  $x_{1:n}^{(i)} = \{x_{1:n-1}^{(i)}, +1/\sqrt{2}\}$ ,  $\forall i \in$*

- $(2g-1)_n$  and  $x_{1:n}^{(j)} = \{x_{1:n-1}^{(j)}, -1/\sqrt{2}\}$ ,*

- $\forall j \in (2g)_n$ ;*

- *Calculate  $w_n^{(i)}$ ,  $\forall i \in (2g-1)_n$  and  $w_n^{(j)}$ ,*

- $\forall j \in (2g)_n$  using (12);*

- *Eliminate the empty groups and update  $G_n$ ;*

- *Calculate  $\tilde{w}_n^{(i)}$ ,  $\forall i \in g_n = 1, \dots, G_n$  using*

- (14);*

- *If (number of groups with 1 particle  $> 0.35N_p$*

- and  $G_n > 0.5N_p$  and  $n \neq N$ ), resample the*

- particles according to their normalized weights*

- and reinitialize  $w_n^{(i)} = 1$  and  $\tilde{w}_n^{(i)} = 1/N_p$ ,*

- $\forall i = 1, \dots, N_p$ .*

Fig. 3. The steps of the PF algorithm.

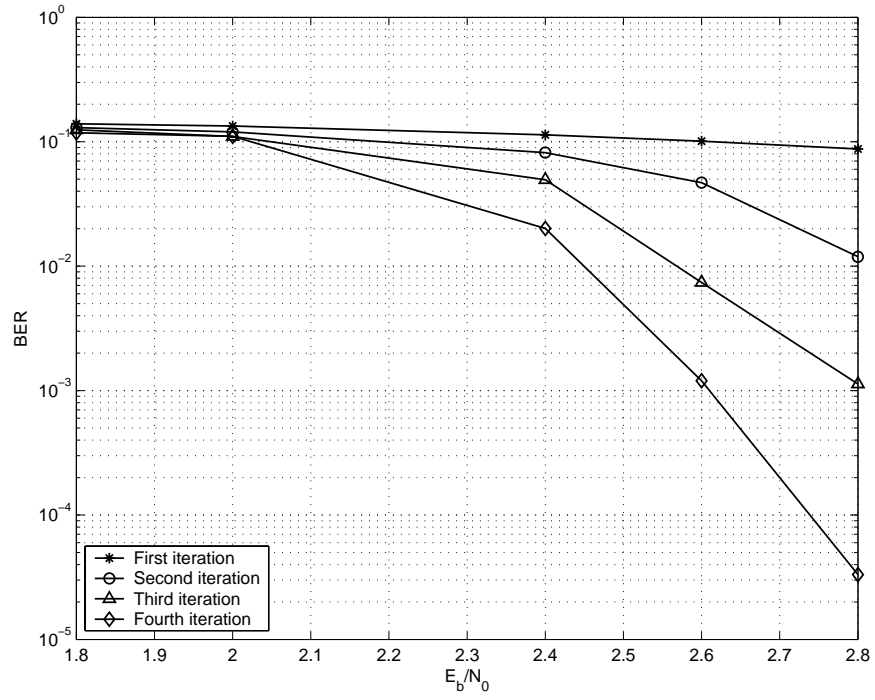


Fig. 4. BER versus  $E_b/N_0$ : PF detector with 1000 particles.

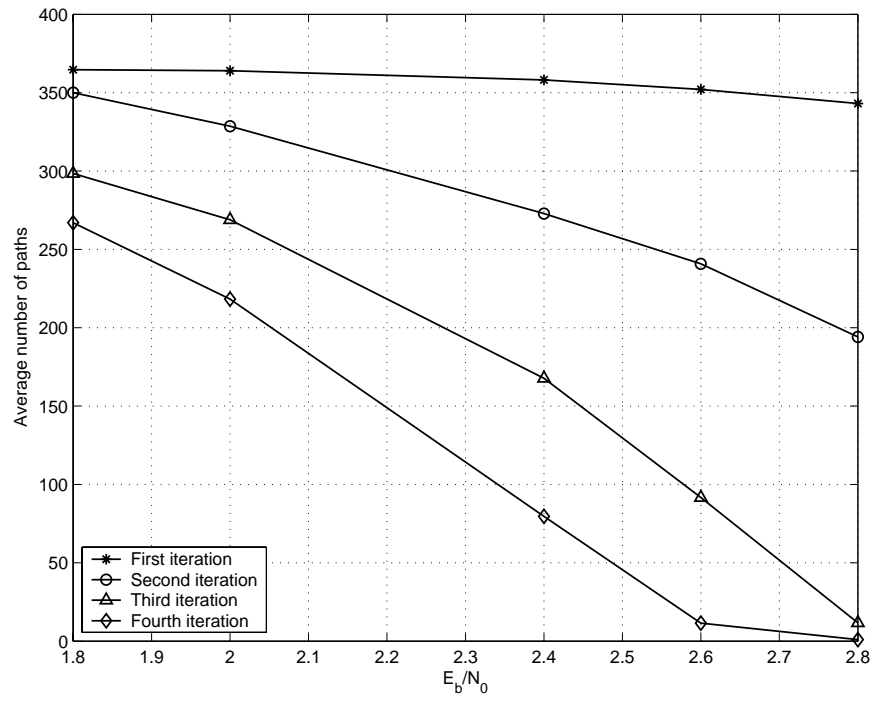


Fig. 5. Complexity versus  $E_b/N_0$ : PF detector with 1000 particles.

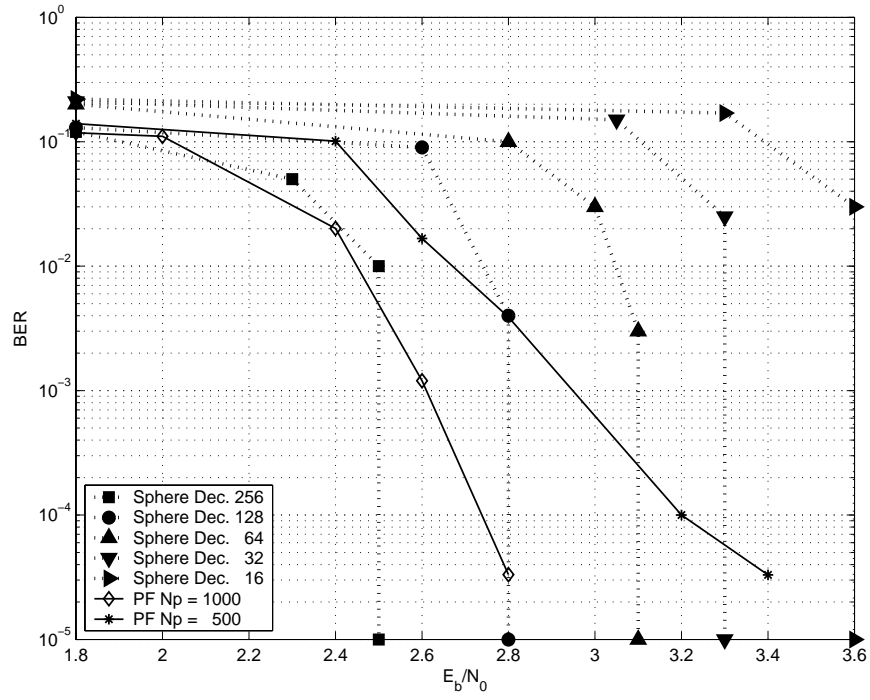


Fig. 6. BER versus  $E_b/N_0$ : Comparison list sphere decoder / PF detector.

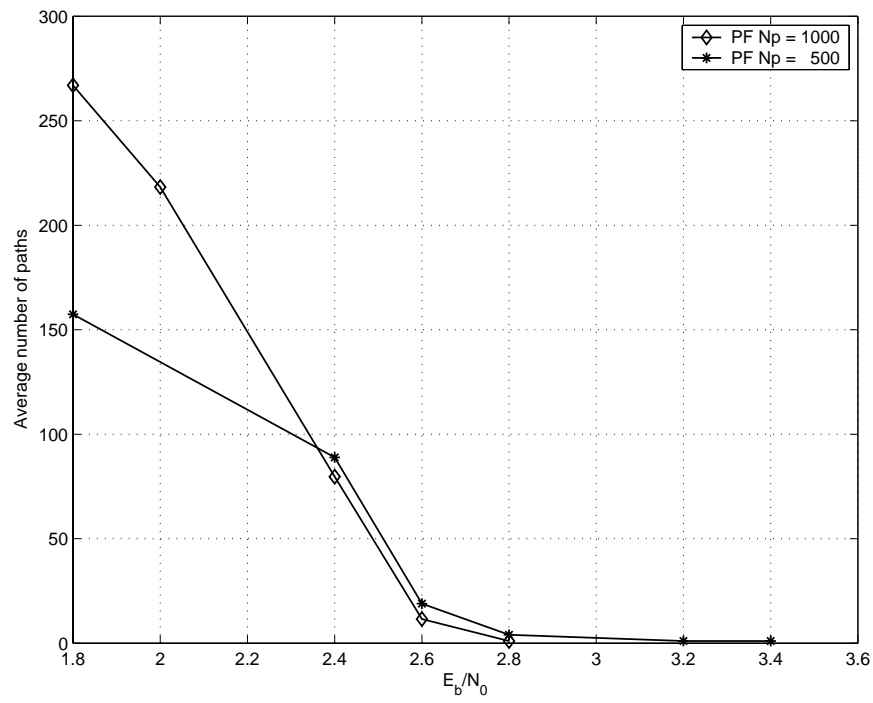


Fig. 7. Complexity versus  $E_b/N_0$ : PF detector with 1000 and 500 particles.