Detection with particle filtering in BLAST systems

Yufei Huang
Department of Electrical Engineering
The University of Texas at San Antonio
San Antonio, TX 78249-0669
E-mail: yhuang@utsa.edu

Jianqiu Zhang
Department of Electrical and Computer Engineering
University of New Hampshire
Durham, NH 03824
Email: jianqiu.zhang@unh.edu

Petar M. Djurić
Department of Electrical and Computer Engineering
State University of New York at Stony Brook
Stony Brook, NY, 11794
Email: djuric@ece.sunysb.edu

***Abstract*— This work demonstrates the use of particle filtering for detection in BLAST systems. A novel dynamic state-space model (DSSM) is constructed for BLAST systems that is crucial for development of particle filtering algorithms. The proposed DSSM is based on QR decomposition and the output of the feed-forward filter, and it evolves in space. The particle filtering solution does not suffer from error propagation, and our simulations show that it greatly outperforms the V-BLAST and provides near optimum performance.**

I. INTRODUCTION

Recent studies on bandwidth efficient transmission for broadband wireless communications have been focused on the exploitation of spatial diversity. It has been shown that the use of multiple transmitting and receiving antennas in rich scattered multipath communication environments can provide enormous capacity gain. The thrust of the work came with an architecture called BLAST (Bell Laboratories Layered Space-Time) [1], [2].

The maximum likelihood (ML) or minimum mean-square error (MMSE) criteria provide optimum detection for BLAST systems. However, the complexity of optimum detection exponentially increases with the number of transmitting antennas, and thus imposes a prohibitive price for practical implementation of BLAST systems with many transmitting antennas. To seek certain balance between complexity and performance, a detection algorithm employing ordered successive interference cancellation (OSI) was proposed and named V-BLAST (vertical BLAST) [3]. Although the V-BLAST system is rather simple for implementation, its performance is limited due to error propagation. To alleviate the problem, various new schemes have been discussed, but the performance improvement has often been only marginal [4], [5], [6].

In this paper, we propose to use a sequential Monte Carlo sampling algorithm, also known as particle filtering (PF) [7], [8], for detection in BLAST systems. A distinct advantage in detection by PF is that it can prevent error propagation and hence achieve near optimum performance. To employ PF, we demonstrate the possibility of constructing a dynamic state space model (DSSM) for BLAST systems. It is based on QR decomposition of the channel matrix and the output of the feed-forward filter, and it evolves in space.

This work was supported by the National Science Foundation under Award CCR-0082607.

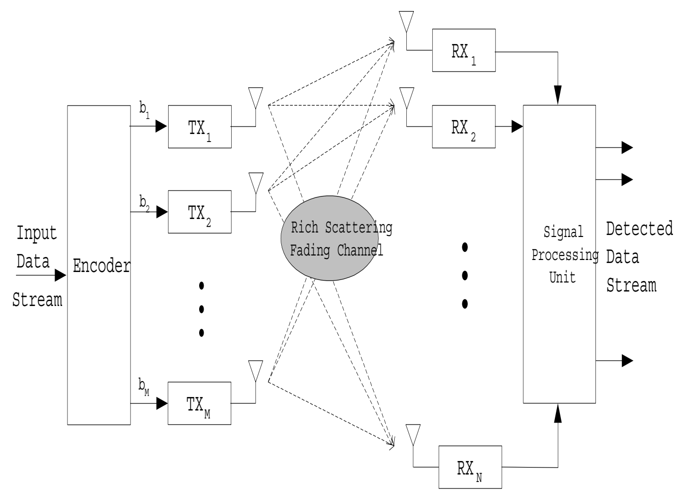

Fig. 1. BLAST system diagram.

The remaining parts of the paper are organized as follows. In Section II, we describe the system model, review the V-BLAST system, and state the adopted methodology. In Section III, we briefly review Monte Carlo and importance sampling, demonstrate the DSSM of BLAST systems, and develop a PF solution for detection. In Section IV, we present simulation results, and in Section V, we make concluding remarks.

II. PROBLEM FORMULATION

A. System model

We consider a flat fading MIMO system as illustrated in Figure 1. At the transmitter, a single data stream is first divided into

M substreams or layers, and they are then encoded, mapped, and transmitted in parallel on the M transmitting antennas. The receiver consists of N receiving antennas (assume $N \geq M$) and at time t, the sampled discrete signal vector $\mathbf{y}$ can be written as

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{n} \tag{1}$$

where $\mathbf{H}$ is an $N \times M$ channel matrix which is known at the receiver, $\mathbf{s}$ is an $M \times 1$ vector that represents the transmitted signal, and $\mathbf{n}$ is an $N \times 1$ noise vector. The data are assumed to have narrow bands, and therefore the channels are considered flat Rayleigh fading channels. Thus, the entries of $\mathbf{H}$ are independent identically distributed (i.i.d.) zero mean complex Gaussian random variables of equal variance σ_n^2. The total signal power $E[\mathbf{s}^{\mathsf{H}}\mathbf{s}]$ is P ($^{\mathsf{H}}$ represents the Hermitian transpose), and $\mathbf{n}$ is zero mean complex additive white Gaussian noise vector with covariance matrix $E[\mathbf{n}\mathbf{n}^H] = \sigma_n^2 \mathbf{I}_N$, where $\mathbf{I}_N$ is the identity matrix of dimension N. We are here concerned with detection of the transmitted signal $\mathbf{s}$ from the receiving observations $\mathbf{y}$.

B. Review of the V-BLAST scheme

An optimum solution to the detection problem is based on the maximum likelihood (ML) principle. However, since the complexity of the ML detection is exponential with the number of transmitting antennas M, it is prohibitive to use it in practice. To achieve a reasonable trade-off between complexity and performance, Foschini proposed a suboptimal algorithm based on OSI and the receiver applying this scheme has been refereed to as the V-BLAST receiver.

In a V-Blast receiver, the detection proceeds along the signal layers in a decreasing order of their signal-to-noise ratio. In the detection of each layer, a two-step scheme with cancellation and nulling is carried out. First, the estimated interference from the previous layers is subtracted out from the receiving observations using the detected signals and then nulling is performed to extract the desired signal layer. In fact, the V-BLAST receiver was shown to be equivalent to a generalized decision feedback equalizer (DFE) [9] and that its implementation based on DFE requires less computational effort [10]. In the following, we describe the V-BLAST algorithm from a DFE perspective. In a later section, the proposed algorithm is developed based on this representation.

In a V-BLAST receiver, first the channel matrix is decomposed according to the Gram-Schmidt QR decomposition [11] as

$$\mathbf{H} = (\mathbf{Q}\ \bar{\mathbf{Q}}) \begin{pmatrix} \mathbf{R} \\ \mathbf{0} \end{pmatrix} \tag{2}$$

where $\mathbf{Q}$ and $\bar{\mathbf{Q}}$ are $N \times M$ and $N \times (N-M)$ unitary matrices, $\mathbf{R}$ is an $M \times M$ lower triangular matrix, and $\mathbf{0}$ is an $(N-M) \times M$ matrix with all entries equal to 0. Then, in the feedforward filter, we right multiply $\mathbf{y}$ with $\mathbf{Q}^{\mathsf{H}}$ and obtain

$$\begin{aligned} \bar{\mathbf{y}} &= \mathbf{Q}^{\mathsf{H}}\mathbf{y} \\ &= \mathbf{R}\mathbf{s} + \bar{\mathbf{n}} \end{aligned} \tag{3}$$

or

$$\begin{matrix} \bar{y}_1 \\ \bar{y}_2 \\ \vdots \\ \bar{y}_M \end{matrix} = \begin{pmatrix} r_{11} & 0 & \cdots & 0 \\ r_{21} & r_{22} & \ddots & 0 \\ \vdots & \vdots & \ddots & 0 \\ r_{M1} & r_{M2} & \cdots & r_{MM} \end{pmatrix} \begin{pmatrix} s_1 \\ s_2 \\ \vdots \\ s_M \end{pmatrix} + \begin{matrix} \bar{n}_1 \\ \bar{n}_2 \\ \vdots \\ \bar{n}_M \end{matrix} \tag{4}$$

where r_{ij} and $\bar{\mathbf{n}}_i$ are the ijth and ith element of the matrix $\mathbf{R}$ and vector $\bar{\mathbf{n}}$, respectively, and $\bar{\mathbf{n}}$ is still an i.i.d. Gaussian random vector with the same mean and variance as $\mathbf{n}$.

Next, in the feedback filter, the symbols are detected successively from s_1 to s_M and at the m-th step, the decision statistic is computed as

$$x_m = \bar{y}_m - r_{m1}\hat{s}_1 - \ldots - r_{m(m-1)}\hat{s}_{m-1} = r_{mm}s_m + \bar{n}_m \tag{5}$$

where $\hat{s}_{m+1}$, $\hat{s}_{m+2}$, $\cdots$, and, $\hat{s}_M$ are the decisions made from the previous steps and they can take either soft or hard decisions. Notice that decision errors made in early steps propagate and result in imperfect interference cancellation in later steps. As a result, the V-BLAST receiver is not near optimum, and to reduce the error in early steps, an ordering on $\mathbf{y}$ with respect to the SNR is applied [5].

C. Decision criterion

We approach the problem from a Bayesian perspective and in particular, we are interested in the marginalized minimum mean square error (MMMSE) decision criterion. According to MMMSE, the decision statistic for s_m is expressed as

$$z_m = E[s_m|\mathbf{y}] = \sum_{\mathbf{s} \in \mathcal{A}} s_m p(\mathbf{s}|\mathbf{y}) \tag{6}$$

where $\mathcal{A} = \{a_1, a_2, \cdots, a_K\}$ is the alphabet set of the constellation in use and $p(\mathbf{s}|\mathbf{y})$ is the full posterior distribution. A final decision on s_m can be made by mapping z_m to the nearest constellation point in $\mathcal{A}$.

The complexity of the MMMSE approach is the same as that of the ML receiver since K^M terms must be evaluated on the posterior distribution, which makes the MMMSE method prohibitive for practical use. In the following sections, we propose to use particle filtering for estimation of z_ms.

III. The PF detection scheme

A. Review of Monte Carlo sampling

Monte Carlo sampling is a powerful method for calculating high dimensional integrals [12]. It has been intensively studied by the statistics community in the past decade, and it has become of great interest to researchers in the area of signal processing in the past few years [13].

The use of the Monte Carlo method in computing the MMMSE estimate (6) requires generation of random samples

$\{\mathbf{s}^{(j)}\}_{j=1}^{J}$ from the posterior distribution $p(\mathbf{s}|\mathbf{y})$, where J indicates the sample size. Then (6) can be approximated by the sample average as

$$z_m \approx \frac{1}{J}\sum_{j=1}^{J} s_m^{(j)} \tag{7}$$

and the approximation can be shown to converge to z_m as J increases [14]. For our problem, J is usually much smaller than K^M, which makes the computation of (7) manageable.

One difficulty associated with the Monte Carlo method is the direct sampling from $p(\mathbf{s}|\mathbf{y})$ because the calculation of the normalizing constant of $p(\mathbf{s}|\mathbf{y})$ requires evaluation of all the K^M terms in the variable space, which again is intractable for large M. To circumvent the difficulty, various sampling schemes can be applied, and we describe here the importance sampling. In the implementation of importance sampling, one first draws samples $\{\mathbf{s}^{(j)}\}_{j=1}^{J}$ from a trial importance distribution $\pi(\mathbf{s}|\mathbf{y})$, which must be easy for sampling. Then, the weights of the samples are calculated by

$$w^{(j)} = \frac{p(\mathbf{s}^{(j)}|\mathbf{y})}{\pi(\mathbf{s}^{(j)}|\mathbf{y})} \quad \forall j. \tag{8}$$

Next, the weights are normalized by $\bar{w}^{(j)} = w^{(j)}/\sum_j^J w^{(j)}$. It should be noted that this normalization process eliminates the necessity of knowing the normalizing constant of $p(\mathbf{s}|\mathbf{y})$ and $\pi(\mathbf{s}|\mathbf{y})$ in computing the weights (8). These samples and weights approximate $p(\mathbf{s}|\mathbf{y})$, and using them, we can estimate z_i in (6) by

$$z_m \approx \sum_{j=1}^{J} \bar{w}^{(j)} s_m^{(j)}. \tag{9}$$

The effectiveness of importance sampling is affected by the choice of the trial importance distribution. In general, the more similar $\pi(\mathbf{s}|\mathbf{y})$ to $p(\mathbf{s}|\mathbf{y})$ is, the less samples are needed to achieve the same performance.

B. The PF detector

As every Monte Carlo sampling algorithm, PF also requires generation of random samples from a desired posterior distribution. However PF allows for producing samples sequentially as the unobserved states of the system evolve. One important approach to implementing PF is by sequential importance sampling (SIS).

PF is commonly used for dynamic systems. Recently it has also been used as an alternative sampling method for static systems [15]. There, the advantage of using PF is that it can produce samples more effectively than the generic importance sampling method. Nonetheless, it is in general not trivial to apply PF for static systems.

When applying PF, especially to static systems, it is instrumental to identify a Markovian factorization of the posterior distribution, or equivalently, to establish a DSSM for the problem. In our problem, the feedforward filter output provides us with the possibility of constructing a DSSM, which evolves in space from $\bar{y}_1$ to $\bar{y}_M$. Particularly, $\mathbf{s}$ is a state variable and considered as the static parameter. By using the Markovian property of the DSSM, the posterior distribution up to step m can be factored as follows

$$\begin{aligned} p(s_{1:m}|\bar{\mathbf{y}}_{1:m}) &= \frac{p(\bar{\mathbf{y}}_m|s_{1:m},\bar{\mathbf{y}}_{1:m-1})p(s_{1:m}|\bar{\mathbf{y}}_{1:m-1})}{p(\bar{\mathbf{y}}_m|\bar{\mathbf{y}}_{1:m-1})} \\ &\propto p(\bar{\mathbf{y}}_m|s_{1:m})p(s_m)p(s_{1:m-1}|\bar{\mathbf{y}}_{1:m-1}) \end{aligned} \tag{10}$$

where the subscript $_{1:m}$ denotes a collection of the variables with subscript of 1 to m, where for instance, $s_{1:m} = \{s_1, s_2, \cdots, s_m\}$. Now, to obtain samples from the posterior distribution, we apply importance sampling with the trial importance distribution chosen according to

$$\pi(s_{1:m}|\bar{\mathbf{y}}_{1:m}) = p(s_m|s_{1:m-1},\bar{\mathbf{y}}_{1:m}) \tag{11}$$

$$p(s_{m-1}|s_{1:m-2},\bar{\mathbf{y}}_{1:m-1})\cdots p(s_1|y_1)$$

$$= p(s_m|s_{1:m-1},\bar{\mathbf{y}}_{1:m})\pi(s_{1:m-1}|\bar{\mathbf{y}}_{1:m-1}). \tag{12}$$

The associated importance weight for the jth samples is calculated by

$$\begin{aligned} w_m^{(j)} &= \frac{p(s_{1:m}^{(j)}|\bar{\mathbf{y}}_{1:m})}{\pi(s_{1:m}^{(j)}|\bar{\mathbf{y}}_{1:m})} \\ &= \frac{p(\bar{\mathbf{y}}_m|s_{1:m}^{(j)})p(s_m^{(j)})p(s_{1:m-1}^{(j)}|\bar{\mathbf{y}}_{1:m-1})}{p(s_m^{(j)}|s_{1:m-1}^{(j)},\bar{\mathbf{y}}_{1:m})\pi(s_{1:m-1}^{(j)}|\bar{\mathbf{y}}_{1:m-1})} \\ &= \frac{p(\bar{\mathbf{y}}_m|s_{1:m}^{(j)})p(s_m^{(j)})}{p(\bar{\mathbf{y}}_m|\bar{\mathbf{y}}_{1:m-1})\frac{p(\bar{\mathbf{y}}_m|s_{1:m}^{(j)})p(s_m^{(j)})}{p(\bar{\mathbf{y}}_m|s_{1:m-1}^{(j)},\bar{\mathbf{y}}_{1:m-1})}} w_{m-1}^{(j)} \\ &\propto p(\bar{\mathbf{y}}_m|s_{1:m-1}^{(j)},\bar{\mathbf{y}}_{1:m-1})w_{m-1}^{(j)} \\ &\propto u_m^{(j)} w_{m-1}^{(j)} \end{aligned} \tag{13}$$

where the second equality is arrived by using the factorization (10) and $u_m^{(j)}$ is called the incremental weight. In deriving the above equation, we utilized the fact that $p(s_m^{(j)}|s_{1:m-1}^{(j)},\bar{\mathbf{y}}_{1:m-1}) = p(s_m^{(j)})$, i.e., s_m is independent of data from other antennas and previous observations. We have also ignored the term $p(\bar{\mathbf{y}}_m|\bar{\mathbf{y}}_{1:m-1})$ because it is the same for all samples and in any way it will be eliminated in weight normalization. The importance distribution (11) is known as the optimal importance function in the PF literature because it produces weights with minimal variance conditional on $s_{1:m-1}^{(j)}$ and $\bar{\mathbf{y}}_{1:m}$ [7]. Examining (11) and (13), we notice that the samples and the weights can be obtained recursively based on those acquired at step $m-1$, and this recursive implementation of importance sampling is known as PF. In the jargon of PF, $s_m^{(j)}$ is called a particle and $s_{1:m}^{(j)}$ is referred to as a trajectory. Now suppose that at step $m-1$, we have obtained the trajectories $\{s_{1:m-1}^{(j)}\}_{j=1}^{J}$ and the weights $\{w_{m-1}^{(j)}\}_{j=1}^{J}$. Then the detailed procedure at the mth step can be summarized by the following chart:

For $j = 1$ to J,
1) Draw a particle $s_m^{(j)}$ from the trial distribution $p(s_m|s_{1:m-1}^{(j)}, \bar{\mathbf{y}}_{1:m})$.
2) Append $s_m^{(j)}$ to $s_{1:m-1}^{(j)}$ and obtain the extended trajectory $s_{1:m}^{(j)}$.
3) Evaluate the incremental weight $u_m^{(j)}$ and calculate the weight $w_m^{(j)}$ using (13).

Perform weight normalization by $\bar{w}_m^{(j)} = w_m^{(j)} / \sum_{j=1}^{J} w_m^{(j)}$

In implementing the above PF procedure, we need to draw samples from the trial distribution $p(s_m|s_{1:m-1}^{(j)}, \bar{\mathbf{y}}_{1:m})$ and calculate the incremental weights $u_m^{(j)}$. The essence of the two requirements is the evaluation of the likelihood function

$$\begin{aligned}\lambda_k &= p(\bar{\mathbf{y}}_m|s_m = a_k, s_{1:m-1}^{(j)}) \\ &= \mathcal{N}(\bar{\mathbf{y}}_m - r_{m,m}a_k - \sum_{i=1}^{m-1} r_{m,i}s_i, \sigma_n^2) \end{aligned} \tag{14}$$

for all $a_k \in \mathcal{A}$. Then, a sample a_k from $p(s_m|s_{1:m-1}^{(j)}, z_{1:m})$ is drawn with probability $\lambda_k / \sum_{i=1}^{K} \lambda_i$ which rests on the fact that

$$\begin{aligned} & p(s_m = a_k|s_{1:m-1}^{(j)}, \bar{\mathbf{y}}_{1:m}) \\ &\quad \propto p(\bar{\mathbf{y}}_m|b_m = a_k, b_{1:m-1}^{(j)}, \bar{\mathbf{y}}_{1:m-1}) \\ &\qquad\quad p(b_m = 1|b_{1:m-1}, \bar{\mathbf{y}}_{1:m-1}) \\ &\quad = p(\bar{\mathbf{y}}_m|s_m = a_k, s_{1:m-1}^{(j)})p(s_m = a_k) \\ &\quad = \lambda_k p(s_m = a_k) \\ &\quad \propto \lambda_k \end{aligned} \tag{15}$$

The last proportional relation is arrived from the fact that the prior density of s_m is noninformative and $p(s_m = a_k) = \frac{1}{K}$. Next, since

$$u_m^{(j)} = \sum_{s_m \in \mathcal{A}} p(\bar{\mathbf{y}}_m|s_m, s_{1:m-1}^{(j)})p(s_m) = \frac{1}{K}\sum_{k=1}^{K} \lambda_k \tag{16}$$

the incremental weight is also readily obtained from the λ_ks.

When the algorithm is completed at iteration M, the trajectories $\{s_{1:M}^{(j)}\}_{j=1}^{J}$ and their weights $\{\bar{w}_M^{(j)}\}_{j=1}^{J}$ approximate $p(\mathbf{s}|\bar{\mathbf{y}})$, or equivalently $p(\mathbf{s}|\mathbf{y})$, the desired posterior distribution. Finally, we can form our decision using these weighted samples according to (9).

The advantage of PF is its ability to reduce and even prevent error propagation. For the MMMSE criterion, the marginalized posterior distribution (MPD) is the key entity for inference.

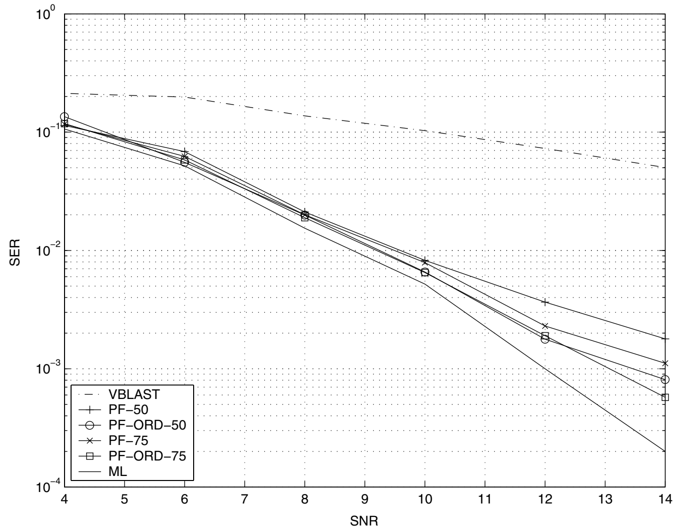

Fig. 2. Plot of SER vs. SNR. The system settings are $M = 4$, $N = 4$, and 4-QAM.

Since the MPD is independent of the decision on symbols from other antennas, the decision on the symbol of interest is immune to decision errors on other symbols. The PF can produce samples which approximate the MPD very closely and therefore can be very effective in reducing and eventually preventing error propagation. However, when the sample size is limited, there would be error propagation in some small degree. Then ordering the observations according to SNR as in [5] could be advantageous.

An important issue of PF is the need for resampling. Namely, after several steps, some weights of the samples become trivial and stop contributing to the overall evaluation. In the literature of PF, resampling is used so that samples with negligible weights are replaced by those from the high density area of the desired posterior distribution. There are many strategies for resampling, and we use the residual resampling procedure as described in [16]. There is a slight difference here with respect to its common implementation. Usually, when a sample trajectory is selected in the resampling, only the present particle in the trajectory is retained and therefore after resampling, the connection between the present weights and previous particles is broken. Note that in our application, the weight must be clearly associated with all the particles in the trajectory at all times because otherwise the MMMSE cannot be performed. As a result, we especially emphasize that a whole trajectory must be taken together as an entity in performing resampling.

The complexity of the algorithm is $O(KMJ)$, i.e., proportional to the product of the size of the alphabet set, the number of samples and the number of transmitting antennas. If the size of the alphabet set and the number of samples are fixed, then the complexity is only linear with respect to the number of transmitting antennas.

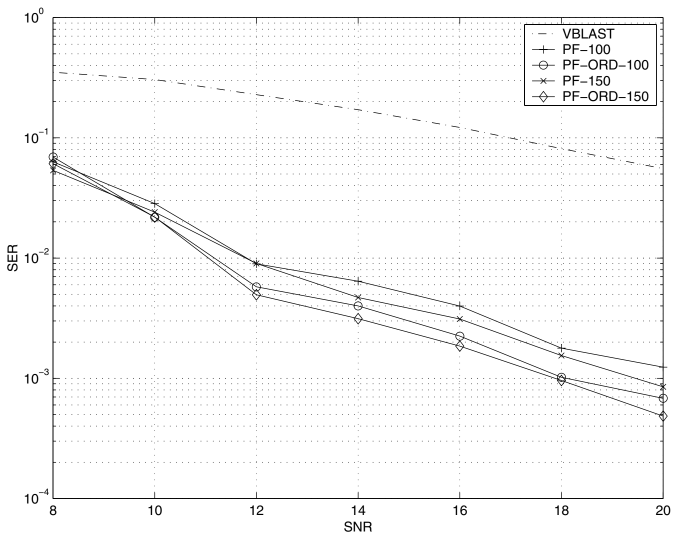

Fig. 3. Plot of SER vs. SNR. The system settings are $M = 8$, $N = 8$, and 16-QAM.

IV. Simulation results

We present several simulation results in this section that show the performance of the proposed PF detection algorithm. In the simulation, the signal is q-QAM modulated and the average power per bit is equal to 1. Thus the symbol energy is $P = 2(q - 1/3)$ and the SNR per receiving antenna per transmitted bit is defined by

$$SNR = 10 \log \frac{MP}{\sigma_n^2 \log_2 q}. \tag{17}$$

We have tested PF detection on the effect of different sample size as well as the ordering. For convenience of presentation, we use PF-ORD-J to represent the PF implementation with the ordering and using J samples.

In the first experiment, we use $M = 4$ transmitting and $N = 4$ receiving antennas with 4-QAM modulation. We compared the PF detection with the V-BLAST as well as the ML solution. The PF with and without ordering were tested for $J = 50$ and $J = 75$. The symbol error rate (SER) versus SNR is plotted in Figure 2. We can see that there is about 10 fold improvement in SER at 10 dB and more than 20 fold at 14 dB for PF algorithms over the V-BLAST. Comparing among the PF algorithms, we notice that they perform rather similarly at low SNRs and the ordering and increased J are more effective for high SNR regions. Finally, their performance is near optimum, especially the one of PF-ORD-75.

Next, we tested the system with 16-QAM modulation employing $M = 8$ transmitting antennas and $N = 8$ receiving antennas. This time the sample size for PF was set to $J = 100$ and $J = 150$. The simulation results are shown in Figure 3, and big improvements over the V-BLAST can be clearly seen for the PF algorithms. Especially, at 20 dB, PF-ORD-150 achieves almost 100 times gain in SER. Again, as in the previous experiment, the ordering and increasing of sample size show some moderate improvement.

V. Conclusions

We have shown the use of PF for detection in BLAST systems. The particle filtering was constructed on a novel DSSM which evolves in space. The proposed particle filtering scheme was demonstrated to have great improvement over the V-BLAST system and possesses potential for achieving near optimum performance for BLAST systems.

References

[1] G. J. Foschini, "Layered space-time architecture for wireless communication in a fadign environment when using multi-element antennas," *Bell Labs Technical Journal*, vol. 1, no. 2, pp. 41–59, 1996.

[2] G. J. Goschini and M. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Persional Communications*, vol. 45, no. 1, pp. 311–335, 1998.

[3] G. D. Golden, G. J. Foschini, R. A. Valenzuela, and P. W. Wolniasky, "Detection algorithm and initial laboratory results using v-blast space-time communication architecture," *IEE Electronics Letters*, vol. 35, pp. 14–15, Jan 1999.

[4] W. J. Choi, R. Negi, and J. Cioffi, "Combined ML and DFE decoding for the V-BLAST system," in *Proceedings of the IEEE International Conference on Communications 2000*, pp. 1243–1248, 2000.

[5] M. O. Damen, K. Abed-Meraim, and S. Burykh, "Iterative QR detection for BLAST," *Wireless Personal Communications*, vol. 19, no. 3, pp. 179–192, Dec. 2001.

[6] C. Degen, C. Walke, and B. Rembold, "Comparative study of efficient decision-feedback equalization schemes for MIMO systems," in *Proceedings of European Wireless 2002*, Florence, Italy, Feb. 2002.

[7] A. Doucet, S. Godsill, and C. Andrieu, "On sequential monte carlo sampling methods for bayesian filtering," *Statistics and Computing*, vol. 10(3), pp. 197–208, 2000.

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

[9] G. GInis and J. M. Cioffi, "On the relation between V-BLAST and the GDFE," *IEEE Communications Letters*, vol. 5, no. 9, pp. 364–366, Sept. 2001.

[10] D. Wübben, J. Rinas R. Böhnke, V. Kühn, and K.D. Kammeyer, "Efficient algorithm for decoding layered space-time codes," *IEE Electronic Letters*, vol. 37, no. 22, Oct. 2001.

[11] G. H. Golub and C. F. Van Loan, *Matrix Computations*, John Hopkins University Press, 2nd edition, 1989.

[12] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter, *Markov Chain Monte Carlo in Practice*, Chapman and Hall, 1996.

[13] P. M. Djurić and S. J. Goodsill, "Guest editorial special issue on Monte Carlo methods for statistical signal processing," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 173–173, Feb. 2002.

[14] M. A. Tanner, *Tools for Statistical Inference*, Springer Verlag, New York, 1996.

[15] J. Zhang, Y. Huang, and P. M. Djurić, "Multiuser detection with particle filtering," in *Proceedings of EUSIPCO*, Toulouse, France, 2002.

[16] J. Liu and R. Chen, "Sequential Monte Carlo methods for dynamic systems," *J. Amer. Statist. Assoc.*, vol. 93, pp. 1032–1044, 1998.