Bayesian Detection for BLAST

Yufei Huang, *Member, IEEE*, Jianqiu (Michelle) Zhang, *Member, IEEE*, and Petar M. Djurić, *Senior Member, IEEE*

***Abstract*—This work demonstrates the use of the Bayesian methodology for detection in Bell Laboratories Layered Space-Time (BLAST) systems. First, we introduce a procedure for constructing prior distributions and propose the use of two types of prior distributions for the problem. From the corresponding posterior distributions, we obtain the Bayesian linear and decision-feedback detectors and show their equivalence to the popular zero forcing and minimum mean square error (MMSE)-based detectors. Then, we establish an equivalent whitening filter output system model whose unique structure lends itself to constructing a dynamic state space model (DSSM) for BLAST systems, which evolves in space. This DSSM allows for the application of sequential Monte Carlo sampling, or particle filtering (PF), for detection in BLAST systems. We introduce two different particle filtering detectors: the generic particle filtering detector and the stochastic M algorithm. The stochastic M algorithm exploits the discrete nature of the problem in the implementation and, therefore, is much more efficient. Overall, a distinct advantage of the PF detectors is that they can greatly reduce error propagation and thereby achieve near optimum performance. In addition, since they aim at the approximation of the posterior distribution using weighted samples, they can provide soft (probabilistic) information about the unknowns.**

***Index Terms*—Bell Laboratories Layered Space-Time (BLAST) systems, Gibbs sampling, Monte Carlo sampling, particle filtering, Space-time processing, Stochastic M-algorithm.**

I. Introduction

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

In BLAST systems, different data streams are transmitted on different transmitting antennas simultaneously. At the receiver, detection is performed by separating and extracting the streams from the received signals. Although the maximum likelihood (ML) criterion provides optimum performance, its complexity increases exponentially with the number of transmitting antennas. Thus, its practical implementation is prohibitive for systems with large number of transmitting antennas. To achieve manageable complexity, linear and recursive decision feedback algorithms have been proposed, most of which are based on either zero-forcing (ZF) or the minimum mean square error (MMSE) principle. A notable algorithm based on ZF and the use of ordered successive interference cancellation (OSI) has been developed and named vertical BLAST (V-BLAST) [3]. The V-BLAST system is rather simple for implementation, but its performance is limited due to error propagation. To alleviate the error propagation, various new schemes based on hard decision have been proposed, but the performance improvement has often been only marginal [4]–[6]. New algorithms based on soft decision as in [7], [8], however, show promising improvement over those based on hard decisions. Note that all these methods assume that the channels are estimated through, for example, pilot transmissions, and so is the case in this work.

In our paper, we study the detection problem under the Bayesian paradigm. The advantage of the Bayesian methodology is its ability to combine prior knowledge with information collected from the data. We introduce a procedure for constructing of prior distributions for VBLAST systems and propose the use of two types of priors. From the corresponding posterior distributions, we obtain the Bayesian linear and decision feedback (DF) detectors and show their equivalence to the ZF and MMSE based detectors. Moreover, we establish an equivalent whitening filter output (WFO) system model. Based on the WFO model, we develop Bayesian decision feedback detectors as well as an M-detector, which is based on the principle of the M-algorithm. Particularly, the unique structure of the model enables the construction of a dynamic state space model (DSSM) for BLAST systems that evolves in space. This DSSM allows for the application of sequential Monte Carlo sampling, or particle filtering (PF) [9]–[11], for detection in BLAST systems. We introduce two different PF detectors: the generic particle filtering detector and the stochastic M algorithm. The stochastic M algorithm exploits the discrete nature of the problem in the implementation and therefore is much more efficient. A distinct advantage of detection by PF is that the error propagation is greatly reduced and that near-optimum performance is achieved. In addition, since PFs aim at the approximation of the posterior distribution using weighted samples, they can provide soft (probabilitic) information about the unknowns, which can be used in turbo BLAST algorithms [12], [13].

The remaining of the paper is organized as follows. In Section II, we describe the system model and state the detection objective. We derive the posterior distributions and discuss

Manuscript received January 4, 2003; revised January 23, 2004. This work was supported by the National Science Foundation under Awards CCR-9903120 and CCR-0082607. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Zhi Ding.

Y. Huang is with the Department of Electrical Engineering, University of Texas at San Antonio, San Antonio, TX 78249-0669 USA (e-mail: yhuang@utsa.edu).

J. Zhang is with the Department of Electrical and Computer Engineering, University of New Hampshire, Durham, NH 03824 USA (e-mail: jianqiu.zhang@unh.edu).

P. M. Djurić is with the Department of Electrical and Computer Engineering, State University of New York at Stony Brook, Stony Brook, NY 11794 USA (e-mail: djuric@ece.sunysb.edu).

Digital Object Identifier 10.1109/TSP.2004.842210

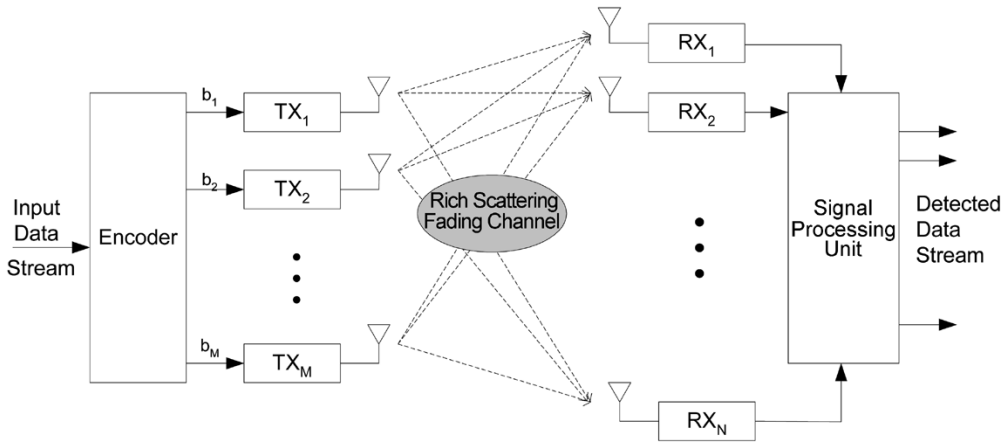


Fig. 1. BLAST system diagram.

the Bayesian decision-feedback detectors in Section III. In Section IV, after briefly reviewing Monte Carlo and importance sampling, we demonstrate the DSSM of BLAST systems and develop a PF solution for detection. We present simulation results in Section V and provide final remarks in Section VI.

II. Problem Formulation

We consider a flat-fading MIMO system, as illustrated in Fig. 1. At the transmitter, a single data stream is first divided into N_t substreams or layers, and they are then encoded, mapped, and transmitted in parallel by N_t transmitting antennas. The receiver consists of N_r receiving antennas (assume $N_r \geq N_t$) 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 $N_r \times N_t$ channel matrix that is known at the receiver, $\mathbf{s}$ is an $N_t \times 1$ vector that represents the transmitted signal, and $\mathbf{n}$ is an $N_r \times 1$ 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. The total signal power $E[\mathbf{s}^H\mathbf{s}]$[1] is N_tP, where P is the power of a single substream, and $\mathbf{n}$ is a zero mean complex additive white Gaussian noise vector with covariance matrix $E[\mathbf{n}\mathbf{n}^H] = \sigma_n^2\mathbf{I}_{N_r}$, where $\mathbf{I}_{N_r}$ is the identity matrix of dimension N_r. We are concerned with detection of the transmitted signal $\mathbf{s}$ from the received observations $\mathbf{y}$. Note that problems of multiuser detection [14] and equalization also have the structure as model (1), and therefore, the algorithms discussed hereafter are applicable to these problems as well.

III. Bayesian Detection

A. Objective

We approach the problem from a Bayesian perspective, and in particular, we are interested in obtaining soft information, i.e., the marginalized posterior distribution of s_m, which can be expressed by

$$p(s_m|\mathbf{y}) = \sum_{\mathbf{s}_{-m}\in\mathcal{A}^{N_t-1}} p(\mathbf{s}|\mathbf{y}) \tag{2}$$

where $\mathcal{A} = \{a_1, a_2, \ldots, a_K\}$ is the alphabet set of the constellation in use, $\mathbf{s}_{-m}$ is the vector of unknown signals except s_m, and $p(\mathbf{s}|\mathbf{y})$ is the full posterior distribution.

Once we obtain $p(s_m|\mathbf{y})$, a final decision on s_m can be calculated according to the maximum *a posteriori* decision as

$$\hat{s}_m = \arg\max_{s_m\in\mathcal{A}} p(s_m|\mathbf{y}). \tag{3}$$

Note that $p(s_m|\mathbf{y})$ can be also the extrinsic information in performing the Turbo BLAST.

When evaluating $p(s_m|\mathbf{y})$, the full posterior distribution $p(\mathbf{s}|\mathbf{y})$ must be calculated, and the complexity of the calculation is exponential. Therefore, exact evaluation of $p(s_m|\mathbf{y})$ is prohibitive in practice. In Sections III-B–D, we develop suboptimum Bayesian detectors with reduced complexity.

B. Posterior Distributions

As indicated above, the posterior distribution of $\mathbf{s}$ is needed for calculating $p(s_m|\mathbf{y})$. The desired posterior distribution of $\mathbf{s}$ is obtained according to Bayes' rule as

$$p(\mathbf{s}|\mathbf{y}) \propto p(\mathbf{y}|\mathbf{s})p(\mathbf{s}) \tag{4}$$

where $p(\mathbf{y}|\mathbf{s})$ and $p(\mathbf{s})$ are the likelihood function and the prior distribution, respectively. Since the likelihood function is defined by the system model (1), the posterior $p(\mathbf{s}|\mathbf{y})$ varies according to the different choices of the prior distribution. In the following, we unfold our derivation of $p(\mathbf{s}|\mathbf{y})$ based on two particular choices for $p(\mathbf{s})$.

1) Noninformative Prior: When nothing is known about $\mathbf{s}$ *a priori*, it is natural to set

$$p(\mathbf{s}) \propto C \tag{5}$$

[1]The superscript ^{H} represents Hermitian transpose.

where C is a constant. This choice leads to a noninformative prior. The corresponding posterior distribution is

$$\begin{aligned} p(\mathbf{s}|\mathbf{y}) &\propto p(\mathbf{y}|\mathbf{s})p(\mathbf{s}) \\ &\propto e^{-\left(1/\sigma_n^2\right)(\mathbf{y}-\mathbf{H}\mathbf{s})^H(\mathbf{y}-\mathbf{H}\mathbf{s})} \\ &\propto e^{-\left(1/\sigma_n^2\right)\left(\mathbf{s}-\mathbf{R}_f\mathbf{H}^H\mathbf{y}\right)^H\mathbf{R}_f^{-1}\left(\mathbf{s}-\mathbf{R}_f\mathbf{H}^H\mathbf{y}\right)} \end{aligned} \tag{6}$$

where $\mathbf{R}_f = (\mathbf{H}^H\mathbf{H})^{-1}$.

2) Gaussian-Type Prior: In cases when we know the covariance matrix $\mathbf{R}_s$ of the transmitted signals, for $p(\mathbf{s})$, we can choose a Gaussian-type function, i.e.,

$$p(\mathbf{s}) \propto e^{-\mathbf{s}^H\mathbf{R}_s^{-1}\mathbf{s}}. \tag{7}$$

Then, the corresponding posterior distribution is

$$\begin{aligned} p(\mathbf{s}|\mathbf{y}) &\propto p(\mathbf{y}|\mathbf{s})p(\mathbf{s}) \\ &\propto e^{-\left(1/\sigma_n^2\right)(\mathbf{y}-\mathbf{H}\mathbf{s})^H(\mathbf{y}-\mathbf{H}\mathbf{s})}e^{-\mathbf{x}^H\mathbf{R}_s^{-1}\mathbf{x}} \\ &\propto e^{-\left(1/\sigma_n^2\right)\left(\mathbf{s}-\mathbf{R}_g\mathbf{H}^H\mathbf{y}\right)^H\mathbf{R}_g^{-1}\left(\mathbf{s}-\mathbf{R}_g\mathbf{H}^H\mathbf{y}\right)} \end{aligned} \tag{8}$$

where $\mathbf{R}_g = \left(\mathbf{H}^H\mathbf{H} + \sigma_n^2\mathbf{R}_s^{-1}\right)^{-1}$. For uncoded systems, $\mathbf{R}_s = P\mathbf{I}_{N_t}$, where P is defined in Section II.

We observe that the above two posterior distributions have similar structures, and for convenience of exposition, we work with a unified posterior distribution of $\mathbf{s}$, which is expressed as

$$p(\mathbf{s}|\mathbf{y}) \propto e^{-\left(1/\sigma_n^2\right)(\mathbf{s}-\mathbf{R}\mathbf{H}^H\mathbf{y})^H\mathbf{R}^{-1}(\mathbf{s}-\mathbf{R}\mathbf{H}^H\mathbf{y})}. \tag{9}$$

This general posterior distribution corresponds to (6) and (8) with $\mathbf{R}$ replaced by $\mathbf{R}_f$ and $\mathbf{R}_g$, respectively. Notice that the maximum *a posteriori* (MAP) rule on (9) with the noninformative prior is equivalent to the ML criterion. Optimum detection based on (6) and (8) is exponentially complex, and therefore, suboptimum linear detectors are constructed from (9) by treating $\mathbf{s}$ as continuous variables. These linear detectors are expressed as

$$\hat{\mathbf{s}}_f = Q\left(\mathbf{R}_f\mathbf{H}^H\mathbf{y}\right) \tag{10}$$

and

$$\hat{\mathbf{s}}_g = Q(\mathbf{R}_g\mathbf{H}^H\mathbf{y}) \tag{11}$$

where $Q(x)$ is the quantization function appropriate to the constellation in use which, for instance, maps x to the nearest constellation point in $\mathcal{A}$. It is not surprising that $\hat{\mathbf{s}}_f$ and $\hat{\mathbf{s}}_g$ obtained within the Bayesian framework are the same as the conventional ZF and MMSE detectors, respectively. We want to point out, however, that the conventional MMSE detector is different from the Bayesian MMSE decision rule described here. The popular ZF and MMSE detectors can be viewed as linear Bayesian MMSE detectors that use noninformative and Gaussian-type priors, respectively. The Bayesian MMSE is equal to the mean of the posterior distribution, and it depends on the applied prior. It is shown in the literature that the MMSE detector outperforms the ZF detector, which is to be expected from a Bayesian standpoint because the MMSE detector exploits a more informative prior. Like in many applications, the two discussed priors lead to mathematically tractable posterior distributions, and thus, they are the most common choices. However, when more specific prior information is available, such as the applied modulation scheme, functions other than the above two might be more desirable and could produce better solutions.

C. Bayesian Decision Feedback Detectors

The performance of linear detectors is often poor because the detectors treat the discrete variables $\mathbf{s}$ as continuous variables. Their performance can be improved by employing the DF principle. To derive a Bayesian DF (BDF) detector, we first perform Cholesky factorization on $\mathbf{R}^{-1}$ as[2]

$$\mathbf{R}^{-1} = \mathbf{F}^H\mathbf{F} \tag{12}$$

where $\mathbf{F}$ is a lower triangular matrix. Replacing $\mathbf{R}^{-1}$ in (9) by its factors yields

$$\begin{aligned} p(\mathbf{s}|\mathbf{y}) &\propto e^{-(1/\sigma_n^2)(\mathbf{s}-(\mathbf{F}^H\mathbf{F})^{-1}\mathbf{H}^H\mathbf{y})^H\mathbf{F}^H\mathbf{F}(\mathbf{s}-(\mathbf{F}^H\mathbf{F})^{-1}\mathbf{H}^H\mathbf{y})} \\ &= e^{-(1/\sigma_n^2)(\mathbf{F}\mathbf{s}-\mathbf{F}^{-H}\mathbf{H}^H\mathbf{y})^H(\mathbf{F}\mathbf{s}-\mathbf{F}^{-H}\mathbf{H}^H\mathbf{y})} \\ &= e^{-(1/\sigma_n^2)(\mathbf{z}-\mathbf{F}\mathbf{s})^H(\mathbf{z}-\mathbf{F}\mathbf{s})} \end{aligned} \tag{13}$$

where $\mathbf{F}^{-H} = (\mathbf{F}^H)^{-1}$, and $\mathbf{z} = \mathbf{F}^{-H}\mathbf{H}^H\mathbf{y}$. When (13) is the posterior distribution obtained by using the noninformative prior, or $\mathbf{R} = \mathbf{R}_f$, the QR decomposition $\mathbf{H} = \mathbf{Q}\mathbf{F}$ can be applied, and further simplification is possible as $\mathbf{z} = \mathbf{F}^{-H}\mathbf{H}^H\mathbf{y} = \mathbf{Q}^H\mathbf{y}$. The derivation of (13) resembles a noise whitening procedure, and thus, we refer to $\mathbf{z}$ as the WFO. Then, an equivalent system model using WFO can be constructed from (13) as

$$\mathbf{z} = \mathbf{F}\mathbf{s} + \bar{\mathbf{n}} \tag{14}$$

or equivalently

$$\begin{aligned} z_1 &= F_{11}s_1 + \bar{n}_1 \\ z_2 &= F_{21}s_1 + F_{22}s_2 + \bar{n}_2 \\ &\vdots\ \ \vdots \\ z_{N_t} &= F_{N_t1}s_1 + F_{N_t2}s_2 + \cdots + F_{N_tN_t}s_{N_t} + \bar{n}_{N_t} \end{aligned} \tag{15}$$

where F_{ij} and s_i are the ijth and ith elements of $\mathbf{F}$ and $\mathbf{s}$, respectively, and $\bar{n}_m\ \forall\, m$ is white Gaussian noise with zero mean and variance σ_n^2. From (15), a BDF scheme can be readily carried out according to

$$\begin{aligned} \hat{s}_1 &= Q\left(\frac{z_1}{F_{11}}\right) \\ &\text{For } m = 2 \text{ to } N_t \\ \hat{s}_m &= Q\left(\frac{\left(z_m - \sum_{i=1}^{m-1} F_{mi}\hat{s}_i\right)}{F_{mm}}\right). \end{aligned}$$

The BDF detector turns out to be the same as the generalized DF detectors discussed in [15], which are proved to be equivalent to the V-BLAST schemes. In particular, the BDF detector using the noninformative prior is equivalent to the ZF V-BLAST, and the BDF detector using the Gaussian-type prior corresponds to the MMSE V-BLAST. However, the implementation using a DF

[2]Note that the Cholesky factorization is not unique due to the ordering of the received data stream.

principle is computationally more efficient [8], [16]. As before and for the same reason, the MMSE V-BLAST has better performance than the ZF V-BLAST.

A key performance limitation of the DF detector is the error propagation. A popular approach to reduce this effect is to employ reordering of the data stream according to their SNR, where data streams with higher SNR are detected first. It should be noted, however, that reordering only decreases the effect of error propagation and does not overcome it. In [17], a PF algorithm is reported for multiuser detection in CDMA systems, which is very effective in combating error propagation.

D. M-Detectors

The WFO (15) induces a tree structure with N_t levels, and the BDF algorithm can be considered to be a suboptimum tree search algorithm, where, at level m, only the branch with the largest conditional posterior probability $p(s_m|\hat{s}_{1:m-1}, z_{1:m})$ is retained from all candidate branches where $\hat{s}_{1:m-1} = \{\hat{s}_1, \ldots, \hat{s}_{m-1}\}$ and $z_{1:m} = \{z_1, \ldots, z_m\}$. An improvement over the BDF algorithm can be achieved by using the key principle of the M-algorithm [18], [19], where M branches with the largest conditional posterior probability are kept instead. We call this detector the M-detector, and its algorithm can be outlined as follows.

Initialization: set $\mu_0^{(J)} = 0$ with $J = 1$;
At the m-th iteration,

•Breath

For each branch $j \in \{1, 2, \ldots, J\}$, iterate for $k = 1$ to K

−Append a_k to $b_{1:m-1}^{(j)}$ and obtain the candidate branch $x_{1:m}^{((j-1)*K+k)} = \left\{b_{1:m-1}^{(j)}, a_k\right\}$.

−Evaluate $\gamma_k^{(j)} = \left|z_m - \sum_{i=1}^{m} F_{mi} x_i^{((j-1)*K+k)}\right|^2$ and calculate the metric by $\nu_m^{((j-1)*K+k)} = \mu_{m-1}^{(j)} \gamma_k^{(j)}$.

•Selection

−If $J \times K \le M$,

Let $s_{1:m}^{(i)} = x_{1:m}^{(i)}$ and $\mu_m^{(i)} = \nu_m^{(i)}$ $i = 1, \ldots, J \times K$. Set $J = J \times K$

−else

Reorder the branches such that $\nu_m^{(1)} \le \nu_m^{(2)} \le \cdots \le \nu_m^{(J \times K)}$. Select the first M branches and set $b_{1:m}^{(i)} = x_{1:m}^{(i)}$ and $\mu_m^{(i)} = \nu_m^{(i)}$ for $i = 1, \ldots, M$. Set $J = M$

The decision on s_m is made when the above algorithm exits at $m = N_t$ and is taken as $\hat{s}_m = b_m^{(1)}$. Both the M-detector and the BDF detector make hard decisions and cannot provide soft information about s_m. In Sections IV and V, we discuss PRF detectors that aim primarily at calculating soft information.

IV. PF Detectors

A. Review of Monte Carlo Sampling

Monte Carlo (MC) sampling is a powerful methodology for approximating desired distributions and calculating high-dimensional integrals [20]. It has been intensively studied by the statistics community in the past decade and has become of great interest to researchers in the area of statistical signal processing [21] and communications [17], [22]–[27] in the past few years.

The use of the MC method in computing the posterior distribution (2) 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. With the samples, MC methods approximate $p(s_m|\mathbf{y})$ by

$$p(s_m|\mathbf{y}) \approx \frac{1}{J} \sum_{j=1}^{J} \delta\left(s_m - s_m^{(j)}\right) \tag{16}$$

and the approximation can be shown to converge to $p(s_m|\mathbf{y})$ as J increases [28].

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

$$w^{(j)} = \frac{p(\mathbf{s}^{(j)}|\mathbf{y})}{\pi(\mathbf{s}^{(j)}|\mathbf{y})} \quad \forall\, j \tag{17}$$

and normalized according to $w^{(j)} = w^{(j)} / \sum_{j=1}^{J} w^{(j)}$. 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 (17). These samples and weights then approximate $p(s_m|\mathbf{y})$ according to

$$p(s_m|\mathbf{y}) \approx \sum_{j=1}^{J} w^{(j)} \delta\left(s_m - s_m^{(j)}\right). \tag{18}$$

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

B. PF Detector

PF is a sequential MC (SMC) method that allows for producing samples (particles) from a desired posterior distribution of unobserved states of an evolving system. The most appealing approach to implementing PF is the one based on sequential importance sampling (SIS), which is the adopted framework in the paper.

PF is commonly employed to dynamic systems described by DSSM. Recently, it has also been used as an alternative importance sampling method for static systems [27]. Recall that the effectiveness of generic importance sampling is affected by the selected importance distribution. Effective importance distribution is often difficult to obtain, and in the case of static systems, special care is needed in the application of PF.

For static systems such as the VBLAST system (15), it is instrumental to identify a Markovian factorization of the posterior distribution or, equivalently, to establish a DSSM for the addressed problem. In our case, the WFO system model (15) allows for construction of a DSSM, which evolves in space from z_1 to z_{N_t},[3] and where $\mathbf{s}$ is considered to be a vector of static state variables. Further, the posterior distribution of interest is $p(\mathbf{s}|\mathbf{z})$. By using the Markovian property of the DSSM, $p(\mathbf{s}|\mathbf{z})$ can be calculated recursively from 1 to N_t as follows:

$$\begin{aligned} p(s_{1:m}|z_{1:m}) &= \frac{p(z_m|s_{1:m}, z_{1:m-1})p(s_{1:m}|z_{1:m-1})}{p(z_m|z_{1:m-1})} \\ &\propto p(z_m|s_{1:m})p(s_m|s_{1:m-1}, z_{1:m-1})p \\ &\quad \times (s_{1:m-1}|z_{1:m-1}) \\ &= p(z_m|s_{1:m})p(s_m)p(s_{1:m-1}|z_{1:m-1}) \quad (19) \end{aligned}$$

where $1 \leq m \leq N_t$, and the subscript $i:j$ denotes a collection of variables with subscripts from i to j, where, for instance, $x_{1:m} = \{x_1, x_2, \ldots, x_m\}$. The last equality is arrived at by using $p(s_m|s_{1:m-1}, z_{1:m-1}) = p(s_m)$, i.e., s_m is independent of data from other antennas and previous observations. The objective here is to obtain samples from the posterior distribution $p(\mathbf{s}|\mathbf{z})$. Since the posterior distributions can be calculated recursively, it is then desirable to produce samples from them recursively. This is possible if we apply importance sampling and choose the importance distribution at step m according to

$$\begin{aligned} \pi(s_{1:m}|z_{1:m}) &= p(s_m|s_{1:m-1}, z_{1:m})p \\ &\quad \times (s_{m-1}|s_{1:m-2}, z_{1:m-1}) \\ &\quad \cdots p(s_1|z_1) \\ &= p(s_m|s_{1:m-1}, z_{1:m})\pi \\ &\quad \times (s_{1:m-1}|z_{1:m-1}). \end{aligned} \quad (20)$$

The associated importance weight for the jth sample is then calculated by (21), shown at the bottom of the page, where the second equality is obtained by using factorization (19), and $u_m^{(j)}$ is called incremental weight. In deriving the above equation, we also ignored the term $p(z_m|z_{1:m-1})$ because it is the same for all samples and is eliminated by weight normalization. The importance distribution (20) 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 $z_{1:1-m}$ [9]. From (20) and (21), it is clear 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 SIS. In the jargon of PF, $s_m^{(j)}$ is called a particle, and $s_{1:m}^{(j)}$ is referred to as a trajectory. If, at step $m-1$, we have the trajectories $\left\{s_{1:m-1}^{(j)}\right\}_{j=1}^{J}$ with weights $\left\{w_{m-1}^{(j)}\right\}_{j=1}^{J}$, the procedure at the mth step can be summarized by the following chart.

For $j = 1$ to J,
- Draw a particle $s_m^{(j)}$ from the trial distribution $p\left(s_m|s_{1:m-1}^{(j)}, z_{1:m}\right)$.
- Append $s_m^{(j)}$ to $s_{1:m-1}^{(j)}$ and obtain the extended trajectory $s_{1:m}^{(j)}$.
- Evaluate the incremental weight $u_m^{(j)}$ and calculate the weight $w_m^{(j)}$ using **(21)**.

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

Implementation of the above PF procedure requires samples from the importance distribution $p\left(s_m|\, s_{1:m-1}^{(j)}, z_{1:m}\right)$ and calculation of the incremental weights $u_m^{(j)}$. The two requirements amount to calculation of the likelihood functions for all $a_k \in \mathcal{A}$, which are easily obtained as

$$\begin{aligned} \lambda_k^{(j)} &= p\left(z_m|s_m = a_k, s_{1:m-1}^{(j)}\right) \\ &= \mathcal{N}\left(z_m - F_{m,m}a_k - \sum_{i=1}^{m-1} F_{m,i}s_i^{(j)}, \sigma_n^2\right). \end{aligned} \quad (22)$$

$$\begin{aligned} w_m^{(j)} &= \frac{p\left(s_{1:m}^{(j)}|z_{1:m}\right)}{\pi\left(s_{1:m}^{(j)}|z_{1:m}\right)} \\ &= \frac{p\left(z_m|s_{1:m}^{(j)}\right)p\left(s_m^{(j)}\right)p\left(s_{1:m-1}^{(j)}|z_{1:m-1}\right)}{p(z_m|z_{1:m-1})p\left(s_m^{(j)}|s_{1:m-1}^{(j)}, z_{1:m}\right)\pi\left(s_{1:m-1}^{(j)}|z_{1:m-1}\right)} \\ &= \frac{p\left(z_m|s_{1:m}^{(j)}\right)p\left(s_m^{(j)}\right)}{p(z_m|z_{1:m-1})\frac{p\left(z_m|s_{1:m}^{(j)}\right)p\left(s_m^{(j)}\right)}{p\left(z_m|s_{1:m-1}^{(j)}, z_{1:m-1}\right)}}w_{m-1}^{(j)} \\ &\propto p\left(z_m|s_{1:m-1}^{(j)}, z_{1:m-1}\right)w_{m-1}^{(j)} \\ &\propto u_m^{(j)}w_{m-1}^{(j)} \end{aligned} \quad (21)$$

[3]The z_ms can be viewed as observations at antennas of the equivalent system. We refer to passing from z_1 to z_m as evolution in space.

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

$$\begin{aligned} p\left(s_m = a_k|s_{1:m-1}^{(j)}, z_{1:m}\right) &\propto p\left(z_m|s_m = a_k, s_{1:m-1}^{(j)}, z_{1:m-1}\right) \\ &\quad \times p(s_m = a_k|s_{1:m-1}, z_{1:m-1}) \\ &= p\left(z_m|s_m = a_k, s_{1:m-1}^{(j)}\right) p(s_m = a_k) \\ &\propto \lambda_k^{(j)}. \end{aligned} \tag{23}$$

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

$$u_m^{(j)} = \sum_{s_k \in \mathcal{A}} p\left(z_m|s_m, s_{1:m-1}^{(j)}\right) p(s_m) = \frac{1}{K}\sum_{k=1}^{K}\lambda_k^{(j)} \tag{24}$$

the incremental weight is also readily obtained from the $\lambda_k^{(j)}$s.

When the algorithm exits at step N_t, the trajectories $\left\{s_{1:N_t}^{(j)}\right\}_{j=1}^{J}$ and their weights $\left\{w_{N_t}^{(j)}\right\}_{j=1}^{J}$ are properly weighted samples from $p(\mathbf{s}|\mathbf{z})$ or, equivalently, $p(\mathbf{s}|\mathbf{y})$, which is the desired posterior distribution. Finally, we can form our decision using the weighted samples according to (18). It is worth stressing that the weighted samples are essentially probability masses that represent the posterior distribution. With them, we can easily calculate not only the MMMSE solution but perform other difficult tasks pertaining to the posterior distribution. For example, we can compute the MAP estimate by choosing the trajectory with the largest cumulative weight,[4] or we can derive extrinsic information from the weights for a Turbo implementation.

As has been pointed out, the advantage of PF over the DF principle is its ability to reduce and even prevent error propagation. To see this, we recall that a PF detector approximates the MMMSE solution, and it approaches the true solution with increase of the sample size J. Since the MMMSE solution is based on the marginalized posterior distribution (MPD) and the MPD is independent of the decision on symbols from other antennas, the MMMSE decision on the symbol of interest is immune to decision errors on other symbols. Consequently, the PF can be very effective in reducing and eventually preventing error propagation. However, when the sample size is limited, there would be error propagation to a certain degree. In those cases an improvement can be achieved by including a Gibbs move at the end of the PF algorithm, and that would invoke additional computation. An easier approach would be to order the observations according to SNR as in [8].

An important issue of PF is the need for resampling. In PF, after several steps, some weights of the samples become trivial and stop contributing to the overall evaluation. Then, resampling is inserted so that samples with negligible weights are replaced by those from the high density area of the desired posterior distribution. The use of resampling moves the mass of the samples closer to the true state and therefore yields a more effective representation of the distribution than the generic importance sampling. Nonetheless, excessive use of resampling impoverishes the sample diversity, and thus, it must be used with care. In practice, resampling can be inserted after a fixed number of steps. There are also many strategies for resampling, and we use the residual resampling procedure as described in [10]. There is a slight difference here with respect to its standard 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 $O(KN_tJ)$, i.e., it is 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.

C. Stochastic M-Detector

Even though the above particle filtering detector has potential to provide accurate soft information and near-optimum performance, the implementation is, in fact, very inefficient and the resampling may also lead to suboptimum solutions. First of all, at the mth iteration of PF, the variable space of the desired posterior distribution $p(s_{1:m}|z_{1:m})$ is of size K^m. For small m, it might be affordable to compute the exact posterior distribution. When this is true, no sampling is needed for the first m steps, and PF should start from the $m+1$th iteration. Second, among the J trajectories, there could be multiple copies of the same trajectory. The original idea of having these multiple copies is that they may produce particles with different values in subsequent steps and therefore increase the diversity of the particles generated from the trajectory. Although this is maybe true for continuous sample spaces since the size of the sample space is infinite, it is, however, not always the case for discrete sample spaces. For our problem, there are only K possible values in the sample space that each trajectory can visit. When the probability mass only concentrates on a few values, the particles generated would be replicates of those few sample values. As a result, multiple copies of the same trajectory at present would produce particles with the same value in subsequent iterations and therefore still remain multiple copies of the same trajectory.

The phenomenon becomes more evident after resampling because more duplicates of the same copy are produced. It is then a waste of resources to record many copies of the same trajectory, which otherwise should be given to process other trajectories. To overcome the problem, these multiple copies should be combined and treated as one trajectory. However, this step reduces the diversity of the generated particles. Another problem is due to resampling, which impoverishes the sample diversity and could discard optimum trajectories in early iterations because of smaller weights. Last, there is another drawback of the

[4]The cumulative weight is the sum of the weights associated with the trajectories carrying the same sample values.

current PF scheme. Suppose that there are no duplicates among the J trajectories. To evolve the jth trajectory from iteration m to $m+1$, K candidates are first calculated, and a new particle is then sampled from them. Accordingly, no matter how large or small the weight $w_m^{(j)}$ is, there is only one candidate of the jth trajectory that is selected for $m+1$. Nevertheless, it may be desirable to retain more than one candidate of a trajectory when its weight is large. Note that overall, there are $K \times J$ candidate trajectories that can be selected for $m+1$. It is, therefore, preferable to sample the J new trajectories from all the $K \times J$ candidates together with the kth candidate of the jth trajectories weighted by $w_m^{(j)}\lambda_k^{(j)}$. This drawback of PF was also noted in [29].

We, therefore, propose to use an alternative implementation of PF, which overcomes the aforementioned drawbacks.

Initialization: Set $J = 1$;
At the m-th iteration,
- **Trajectory expansion**
  For $j = 1$ to J
  For $k = 1$ to K
  –Append a_k to $s_{1:m-1}^{(j)}$ and obtain the extended trajectory $x_{1:m}^{((j-1)K+k)}$.
  –Evaluate $\lambda_k^{(j)}$ according to **(22)** and calculate the weight by $v_m^{((j-1)K+k)} = w_{m-1}^{(j)}\lambda_k^{(j)}$.
- **Trajectory sampling**
  –If $J \times K \le M$,
  Let $s_{1:m}^{(i)} = x_{1:m}^{(i)}$ and $w_m^{(i)} = v_m^{(i)}$ $i = 1, \ldots, J \times K$.
Set $J = J \times K$
  –else
  Sample M trajectories from $J \times K$ extended trajectories $x_{1:m}^{(i)}$ with probability proportional to $v_m^{(i)}$ using the *optimal resampling* algorithm to obtain $s_{1:m}^{(j)}$ and $w_m^{(j)}$ for $j = 1, \ldots, M$. Set $J = M$;
- Perform weight normalization by $w_m^{(j)} = w_m^{(j)} / \sum_{j=1}^{J} w_m^{(j)}$.

The optimal resampling algorithm is proposed in [30], and it is optimal in the sense that the mean square error between the original weights and the sampled weights is minimized. The algorithm for sampling M trajectories from $J \times K$ extended trajectories $x_{1:m}^{(i)}$ weighted by $v_m^{(i)}$ is summarized as follows:

- **Ordering**: Reorder the weights $v_m^{(i)}$ such that $v_m^{(1)} \ge v_m^{(2)} \ge \cdots \ge v_m^{(J\times K)}$
- **Calculating** c: Calculate c as the solution of $\sum_{i=1}^{J\times K} \min(cv_m^{(i)}, 1) = M$.
- **Inheriting:** Determine the largest index l such that $cv_m^{(l)} > 1$. Inherit the trajectory and weights by setting $s_{1:m}^{(j)} = x_{1:m}^{(j)}$ and $w_m^{(j)} = v_m^{(j)}$ for $j = 1, 2, \ldots, l$.
- **Stratified Sampling:**
  –Set $U = (r+l)/M$, $\mathbf{I} = [1, \ldots, J \times K]$, $j = l+1$, and $Q = 0$ where $r \sim U(0,1)$.
  –Do while $U < 1$
  if $Q > U$, then
  Set $U = U + 1/M$, $l = l+1$, $s_{1:m}^{(l+1)} = x_{1:m}^{(i)}$ and $w_m^{(l)} = v_m^{(i)}$.
  else
  Pick an index k for $\{j, j+1, \ldots, J\times K\}$; Set $i = \mathbf{I}_k$ and $\mathbf{I}_k = \mathbf{I}_j$;
  Set $Q = Q + cv_m^{(i)}/M$ and $j = j+1$;

It is shown in [30] that calculating c can be performed in a recursive fashion and that the computational complexity of the optimal sampling is of $O(J \times K)$, which is the same order as the residual sampling. While alternative resampling algorithms such as residual sampling can be used here instead, an attractive feature of the optimal resampling algorithm is that it produces M distinct trajectories and therefore maintains the largest possible diversity of particles. We found through experiments that the diversity of particles is essential in obtaining good performance. A very similar particle filtering algorithm is proposed in [30] in sampling from a Gaussian–Markovian model. However, our algorithm is more efficient. Here, we notice that the above algorithm resembles the M-algorithm discussed in Section III-D. However, there are also clear differences. The M-algorithm is a deterministic algorithm where only the M trajectories (branches) with the largest weights (metrics) are retained, and its objective is to find the best trajectory. By contrast, the selection of trajectories in the above algorithm is based on random sampling, and the goal of it is to obtain best random measures that approximate the posterior distribution. We, therefore, call the algorithm the *stochastic* M-algorithm. Since the stochastic M-algorithm can provide soft information about the unknowns, it is more versatile and can be used in broader applications such as turbo BLAST. However, we want to stress that our stochastic M-algorithm is still a particle filtering algorithm.

V. Simulation Results

We present several simulation results that show the performance of the proposed PF detectors. In the simulation, the signal is q-quadrature amplitude modulated (QAM), 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

$$\text{SNR} = 10 \log \frac{N_t P}{\sigma_n^2 \log_2 q}. \tag{25}$$

Additionally, we assume independent channels and the entries of the channel matrix are generated from complex Gaussian distributions with zero mean and variance σ_n^2. The detectors were tested with noninformative (NON) and Gaussian-type (GAU) priors on different sample sizes and with ordering and no ordering. For convenience of presentation, we use ORD-J to represent the implementation with ordering and using J samples. Thus, for instance, PF-GAU-ORD-100 denotes a PF implementation with the Gaussian-like prior, ordering, and 100 samples, and SM-NON-8 stands for the stochastic M-algorithm with noninformative prior, no ordering, and eight samples. Further, for comparisons, we tested the VBLAST detectors, and we use ZF-VBLAST and MMSE-VBLAST to denote the zero

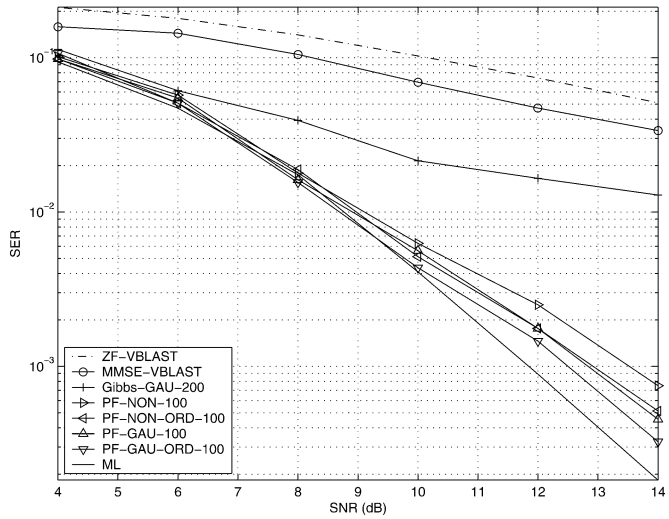


Fig. 2. Plot of SER versus SNR for a system with $N_t = 4$, $N_r = 4$, and 4-QAM. The sample size J is 100.

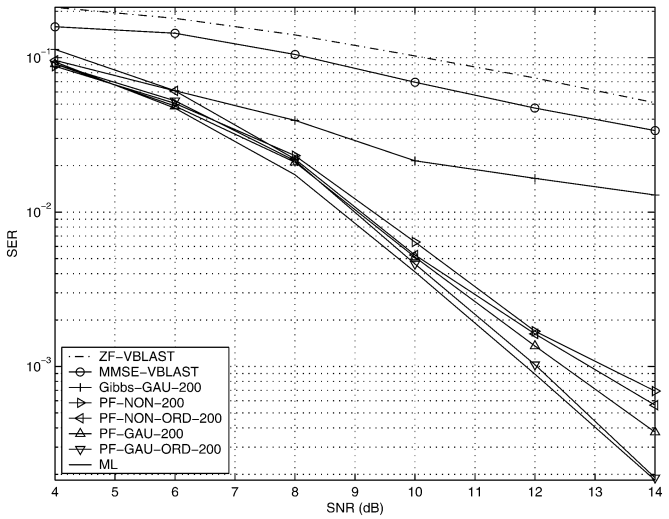


Fig. 3. Plot of SER versus SNR for a system with $N_t = 4$, $N_r = 4$, and 4-QAM. The sample size J is 200.

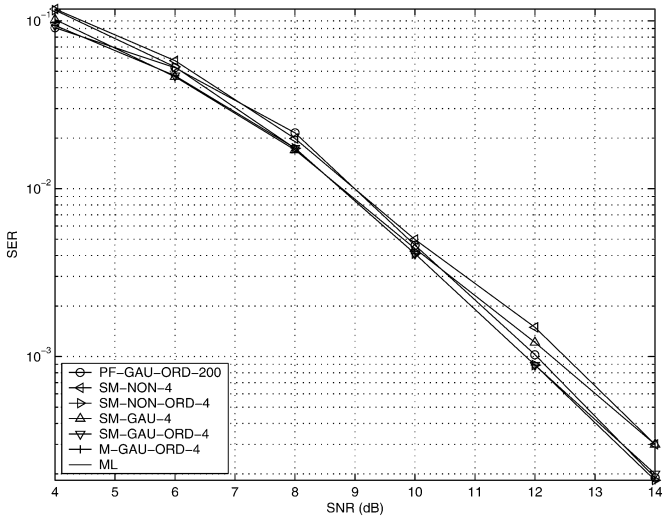


Fig. 4. Plot of SER versus SNR for a system with $N_t = 4$, $N_r = 4$, and 4-QAM.

forcing and MMSE VBLAST detectors, respectively. The sampling-based algorithms make decisions using the MAP rule. In evaluating the performance, we simulated symbol error rates (SERs) for each detector under different system settings. To obtain an SER for a particular setting, Monte Carlo trials were repeated until 200 errors were collected.

In the first experiment, we chose a small system of $N_t = 4$ transmitting and $N_r = 4$ receiving antennas with 4-QAM modulation. For this setting, the optimum solution can be calculated and, thus, used to provide performance lower bound. We simulated both the MAP and MMMSE solutions under the two different priors and found that their performances are almost the same. As a result, we only used the result of the MAP under the noninformative prior or, equivalently, the ML in our simulation. We compared the SER versus SNRs of various PF detectors with the SERs of the ZF-VBLAST, MMSE-VBLAST, and the Gibbs sampling-based detector with Gaussian-type prior (Gibbs-GAU) (the Gibbs sampling based detector is described in the Appendix). In the simulation of the Gibbs sampler, the first 50 samples are used as burn-in, and the next 200 samples are collected for detection. The generic PF detectors with and without ordering were tested for $J = 100$, and resampling was performed at every two steps.

The results are presented in Fig. 2. Overall, the performance gain of the PF detectors over the VBLASTs is considerable. For instance, the PF-NON-100 exhibits about ten fold improvement in SER at 10 dB and more than 50 fold at 14 dB over the MMSE-VBLAST. In comparing the PF detectors, as expected, we notice that the PF-GAUs have a clear edge over the PF-NONs. Given the fact that the performances of the optimum MMMSE under the two priors are the same, the result implies that the PF-GAUs are more effective. Namely, with the same sample size, the PF-GAUs achieve higher performance. We also see that the ordering demonstrates its advantage, especially at high SNR regions. The performance of the PF-GAU-ORD-50 is within less than 1-dB difference to the ML bound. Note that the performance of all these PF detectors can be further improved with the increase of sample size.

The results for $J = 200$ are depicted in Fig. 3. In the figure, the performance relationship remains similar as that for $J = 100$. While the other detectors attained slight performance gain, the PF-GAU-ORD almost achieved the ML bound for $J = 200$. This indicates that even though the performance for $J = 100$ is already satisfactory, the performance of the PFs can be always pushed closer to the bound by increasing J, which is a distinct feature not possessed by deterministic algorithms such as the DFs. However, with a reasonable small sample size J_b, usually, the performance is already close to the bound, and the slight gain by the additional increase in J might not be worth the required extra computation. For example, for this experiment, for PF-GAU-ORD, we can use $J_b = 100$. By comparing the Gibbs-GAU-200 with PF, we see that even though the Gibbs-GAU-200 performs well at low SNRs, it exhibits clear error floor at high SNRs, which is a phenomenon that is also observed in its application to multiuser detection [27].

Next, in Fig. 4, we studied the M-detector and the stochastic M-detector (SM) with different types of priors and compared there performance with PF-GAU-ORD-200. For

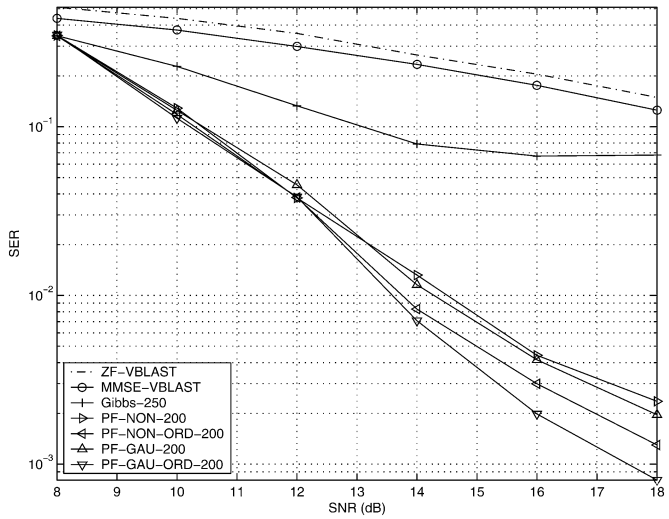


Fig. 5. Plot of SER versus SNR for a system with $N_t = 8$, $N_r = 8$, and 16-QAM. The sample size J is 200.

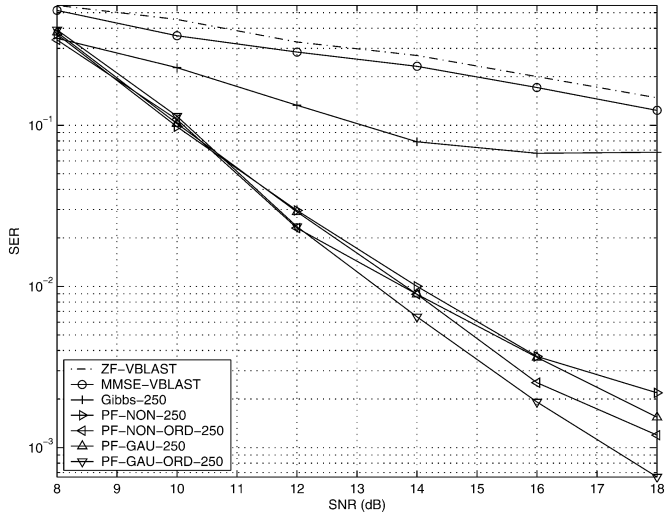


Fig. 6. Plot of SER versus SNR for a system with $N_t = 8$, $N_r = 8$, and 16-QAM. The sample size J is 250.

the two M-detectors, we used $M = 4$ or, equivalently, four trajectories in the PF detectors. We observe that both M-detectors have near optimum performance, and particularly, the SM-GAU-ORD and M-GAU-ORD have exactly the same performance over the tested SNRs. In addition, the SM-NON-ORD and M-NON-ORD reach the optimum solution at high SNRs. The two M-detectors, therefore, can provide similar or even better performance than the PF detectors but with much less complexity. Both the stochastic M-algorithm and the PF detectors can produce soft information.

In the second experiment, we used a larger system with 16-QAM modulation employing $N_t = 8$ transmitting antennas and $N_r = 8$ receiving antennas. This time the sample sizes for PF were set to $J = 200$ and $J = 250$, and resampling was performed at every three steps. For the Gibbs sampler, 300 samples were generated, and the first 50 were used as burn-in. The simulation results are shown in Figs. 5 and 6. One can clearly see big improvements of the PF algorithms over V-BLAST. For example, at 20 dB, PF-GAU-ORDs achieved almost 200 times gain in SER. In addition, the Gibbs sampler shows similar behavior as in the first experiment where an error floor is clearly seen at high SNRs. By comparing the PF detectors, we see again that the PF-GAUs are more efficient than the PF-NONs, and the ordering is again beneficial, especially at high SNRs. Note that the increase of sample size from $J = 200$ to $J = 250$ does not result in clear improvement in performance, and thus, we can choose $J_b = 200$.

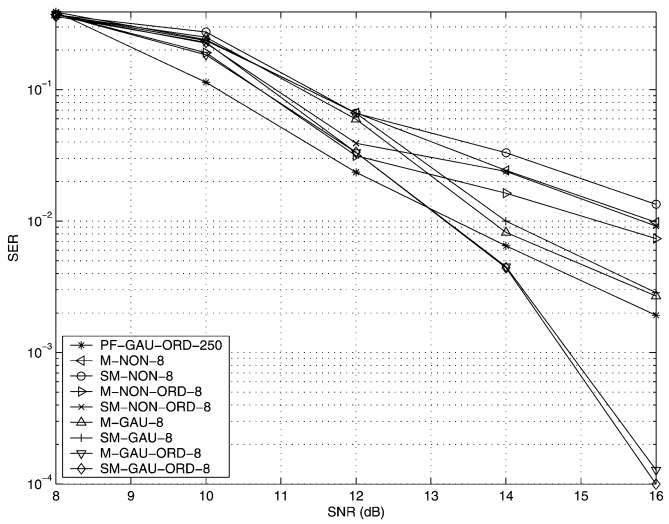


Fig. 7. Plot of SER versus SNR for a system with $N_t = 8$, $N_r = 8$, and 16-QAM.

We then tested the M-algorithm and the stochastic M-algorithm and plotted the results in Fig. 7 along with the results of PF-GAU-ORD-250. This time, we used eight trajectories, or $M = 8$. Overall, the M and SM methods have similar performance, but the SM method can provide soft information. Even though the PF-GAU-ORD-250 performs better than all the M and SM methods at low SNRs (below 13 dB), the M and SM algorithms with Gaussian type prior and ordering stand out clearly as the best methods for high SNRs. At 16 dB, the M and SM algorithms have about ten-fold improvement over the PF-GAU-ORD-250. Note that only eight trajectories were used for the M and SM methods, which is a very small fraction of 250, which obviously makes them more efficient and effective than the PFs.

VI. Conclusions

We studied the detection in BLAST systems using Bayesian methods. We introduced a general way of constructing prior distributions and proposed the use of two priors. The corresponding posterior distributions were derived, and a WFO system model was established. Based on them, the Bayesian linear and DF detectors were obtained, which were shown to be equivalent to the linear and VBLAST detectors under the ZF and MMSE principles. We identified a tree structure of the WFO model and proposed a detector based on the M-algorithm.

We also developed two different PF detectors for BLAST systems: generic PFs and the stochastic M-algorithm. Two PF algorithms were constructed on a novel DSSM that evolves in space. The proposed schemes were demonstrated to have great improvement over the V-BLAST system. Between the two, the

stochastic M-algorithm is much more efficient and can also provide better performance. We also noticed that it is very beneficial to use the Gaussian type prior and ordering. Even though the M-algorithm has similar performance as the stochastic M-algorithm, the stochastic M-algorithm has the advantage of providing soft information.

APPENDIX
DERIVATION OF THE GIBBS SAMPLING DETECTOR

The Gibbs sampler [20], [28], [31], [32] is a Markov chain MC sampling algorithm, and its objective in our paper is to draw samples from the posterior distribution (9). The algorithm can be described as follows.

Given an initial sample $\mathbf{s}^{(0)}$, iterate for $j = 1$ to J,

- Sample $s_1^{(j)}$ from $p\left(s_1|s_2^{(j-1)}, s_3^{(j-1)}, \ldots, s_{N_t}^{(j-1)}, \mathbf{y}\right)$
- Sample $x_2^{(j)}$ from $p\left(s_2|s_1^{(j)}, s_3^{(j-1)}, \ldots, s_{N_t}^{(j-1)}, \mathbf{y}\right)$

$\vdots \quad \vdots$

- Sample $s_{N_t}^{(j)}$ from $p\left(s_{N_t}|s_1^{(j)}, s_2^{(j)}, \ldots, s_{N_t-1}^{(j)}, \mathbf{y}\right)$

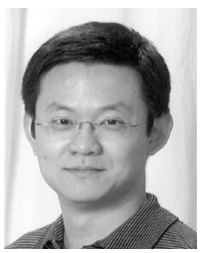
Once J samples are collected, the first κ samples are considered as burn-in and discarded. The remaining samples are used to perform the detection. The choice of the initial sample affects the length of the burn-in period. In our simulations, the results of the zero forcing VBLAST method were used as initial samples.

ACKNOWLEDGMENT

The authors would like to acknowledge the comments of the anonymous reviewers. They were valuable and insightful and helped us in developing the stochastic M-algorithm.

REFERENCES

[1] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs Tech. J.*, vol. 1, no. 2, pp. 41–59, 1996.
[2] G. J. Foschini and M. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Pers. Commun.*, 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," *Electron. Lett.*, 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 *Proc.IEEE International Conference on Communications 2000*, 2000, pp. 1243–1248.
[5] A. Benjebbour, H. Murata, and S. Yoshida, "Performance of iterative successive detection algorithm with space-time transmission," in *Proc. VTC*, 2001, pp. 1287–1291.
[6] C. Degen, C. Walke, and B. Rembold, "Comparative study of efficient decision-feedback equalization schemes for MIMO systems," in *Proc. Eur. Wireless*, Florence, Italy, Feb. 2002.
[7] W. Choi, K. Cheong, and J. M. Cioffi, "Iterative soft interference cancellation for multiple antenna systems," in *Proc. IEEE Wireless Commun. Networking Conf.*, 2000, pp. 304–309.
[8] M. O. Damen, K. Abed-Meraim, and S. Burykh, "Iterative QR detection for BLAST," *Wireless Pers.Commun.*, vol. 19, no. 3, pp. 179–192, Dec. 2001.
[9] A. Doucet, S. Godsill, and C. Andrieu, "On sequential Monte Carlo sampling methods for Bayesian filtering," *Statist. Comput.*, vol. 10, no. 3, pp. 197–208, 2000.
[10] J. Liu and R. Chen, "Sequential Monte Carlo methods for dynamic systems," *J. Amer. Statist. Assoc.*, vol. 93, pp. 1032–1044, 1998.
[11] A. Doucet, J. de Freitas, and N. E. Gordon, *Sequential Monte-Carlo Methods in Practice*. New York: Springer-Verlag, 2001.
[12] M. Sellathurai and S. Haykin, "Turbo-BLAST for wireless communications: Theory and experiments," *IEEE Trans. Signal Process.*, vol. 50, pp. 2538–2546, Oct. 2002.
[13] S. Bäro and G. Bauch, "Improving blast performance using space-time block codes and turbo decoding," in *Proc. IEEE Global Telecommun. Conf.*, vol. 2, 2000, pp. 1067–1071.
[14] S. Verdú, *Multiuser Detection*. New York: Cambridge Univ. Press, 1998.
[15] G. Ginis and J. M. Cioffi, "On the relation between V-BLAST and the GDFE," *IEEE Commun. Lett.*, vol. 5, pp. 364–366, Sep. 2001.
[16] D. Wübben, J. Rinas, R. Böhnke, V. Kühn, and K. D. Kammeyer, "Efficient algorithm for decoding layered space-time codes," *IEE Electron. Lett.*, vol. 37, no. 22, Oct. 2001.
[17] J. Zhang, Y. Huang, and P. M. Djurić, "Joint channel estimation and multiuser detection using particle filtering," in *Proc. Conf. Inf. Sci. Syst.*, Feb. 2002.
[18] J. B. Anderson and S. Mohan, "Sequential coding algorithms: A survey and cost analysis," *IEEE Trans. Commun.*, vol. COM–32, no. 2, pp. 169–177, Feb. 1984.
[19] A. J. Simmons, "Breath-first trellis decoding with adaptive effort," *IEEE Trans. Commun.*, vol. 38, no. 1, pp. 3–12, Jan. 1990.
[20] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter, *Markov Chain Monte Carlo in Practice*. London, U.K.: Chapman and Hall, 1996.
[21] P. M. Djurić and S. J. Goodsill, "Guest editorial special issue on Monte Carlo methods for statistical signal processing," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 173–173, Feb. 2002.
[22] P. M. Djurić, J. M. Kotecha, J. Zhang, Y. Huang, T. Ghirmai, M. F. Bugallo, and J. Míguez *et al.*, "Particle filtering," *IEEE Signal Process. Mag.*, vol. 20, no. 5, pp. 19–38, Sep. 2003.
[23] R. Chen, X. Wang, and J. S. Liu, "Adaptive joint detection and decoding in flat-fading channels via mixture Kalman filtering," *IEEE Trans. Inf. Theory*, vol. 46, no. 6, pp. 2079–2094, Sep. 2000.
[24] X. Wang and R. Chen, "Adaptive Bayesian multiuser detection for synchronous CDMA with Gaussian and impulsive noise," *IEEE Trans. Signal Process.*, vol. 47, no. 7, pp. 2013–2027, Jul. 2000.
[25] Y. Huang and P. M. Djurić, "Multiuser dection of synchronous code-division multiple-access signals by perfect sampling," *IEEE Trans. Signal Process.*, vol. 50, no. 7, pp. 1724–1734, Jul. 2002.
[26] ——, "A blind particle filtering detector for joint channel estimation, tracking and data detection over flat fading channels," in *Proc. Eur. Signal Process. Conf.*, Sept. 2002.
[27] J. Zhang, Y. Huang, and P. M. Djurić, "Multiuser detection with particle filtering," in *Proc. EUSIPCO*, Toulouse, France, 2002.
[28] M. A. Tanner, *Tools for Statistical Inference*. New York: Springer-Verlag, 1996.
[29] E. Punskaya, A. Doucet, and W. J. Fitzgerald, "On the use and misuse of particle filtering in digital communications," in *Proc. EUSIPCO*, Sept. 2002.
[30] P. Fearnhead, "Sequential Monte Carlo Method in Filter Theory," Ph.D., Oxford Univ., Oxford, U.K., 1998.
[31] A. E. Gelfand and A. F. M. Smith, "Sampling-based approaches to calculating marginal densities," *J. Amer. Statist. Assoc.*, vol. 85, pp. 398–409, 1990.
[32] G. Casella and E. I. George, "Explaining the Gibbs sampler," in *Amer. Statist.*, 1992, vol. 46, pp. 167–174.



Yufei Huang (M'03) was born in Shanghai, China, in 1973. He received the B.S. degree in applied electronics from Northwestern Polytechnic University, Xi'an, China, in 1995 and the M.S. and Ph.D. degrees in electrical engineering from the State University of New York (SUNY) at Stony Brook in 1997 and 2001, respectively.

He was a Post-Doctoral Researcher with the Department of Electrical and Computer Engineering, SUNY Stony Brook. He is now with the Department of Electrical Engineering, University of Texas at San Antonio. His current research interests include Bayesian Monte Carlo methods, graphical modeling, and their applications to wireless communications and bioinformatics

Jianqiu (Michelle) Zhang (M'03) received the B.S. degree in 1992 from Zhejiang University, Hangzhou, China, the M.S. degree in 1995 from Zhongshan University, Guangzhou, China, and the Ph.D. degree in 2002 from the State University of New York at Stony Brook, all in electrical engineering.

From 1995 to 1997, she was a software engineer with the Research and Development Department of Guangdong Nortel, Guangzhou. Since 2002, she has been with University of New Hampshire, Durham, where she is an assistant professor with the Department of Electrical and Computer Engineering. Her current research interests include fading channel capacity, combined channel estimation and data detection, channel coding and decoding, particle filtering, multiuser detection, and space-time coding.

Peter M. Djurić (SM'99) received the B.S. and M.S. degrees in electrical engineering from the University of Belgrade, Belgrade, Yugoslavia, in 1981 and 1986, respectively, and the Ph.D. degree in electrical engineering from the University of Rhode Island, Kingston, in 1990.

From 1981 to 1986, he was Research Associate with the Institute of Nuclear Sciences, Vinca, Belgrade, Yugoslavia. Since 1990, he has been with the State University of New York at Stony Brook, where he is Professor with the Department of Electrical and Computer Engineering. He works in the area of statistical signal processing, and his primary interests are in the theory of modeling, detection, estimation, and time series analysis and its application to a wide variety of disciplines, including telecommunications, bio-medicine, and power engineering.

Prof. Djuric has served on numerous technical committees for the IEEE and SPIE and has been invited to lecture at universities in the United States and overseas. He is the Area Editor of Special Issues of the IEEE SIGNAL PROCESSING MAGAZINE and Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING. He is also Chair of the IEEE Signal Processing Society Committee on Signal Processing—Theory and Methods and is on the Editorial Board of *Digital Signal Processing*, the *EURASIP Journal on Applied Signal Processing*, and the *EURASIP Journal on Wireless Communications and Networking*. He is a Member of the American Statistical Association and the International Society for Bayesian Analysis.