Iterative Decoding for MIMO Channels via Modified Sphere Decoding

H. Vikalo[†], B. Hassibi[†], and T. Kailath[‡]

Abstract

In recent years, soft iterative decoding techniques have been shown to greatly improve the bit error rate performance of various communication systems. For multi-antenna systems employing space-time codes, however, it is not clear what is the best way to obtain the soft-information required of the iterative scheme with low complexity. In this paper, we propose a modification of the Fincke-Pohst (sphere decoding) algorithm to estimate the maximum a posteriori probability of the received symbol sequence. The new algorithm solves a nonlinear integer least-squares problem and, over a wide range of rates and signal-to-noise ratios, has polynomial-time complexity. Performance of the algorithm, combined with convolutional, turbo, and low-density parity check codes is demonstrated on several multi-antenna channels. For systems that employ space-time modulation schemes, a major conclusion is that the best performing schemes are those that support the highest mutual information between the transmitted and received signals, rather than the best diversity gain.

Index Terms—Sphere decoding, wireless communications, multi-antenna systems, turbo codes, LDPC codes, space-time codes, iterative decoding, expected complexity, polynomial-time complexity

[†](hvikalo,hassibi)@systems.caltech.edu; California Institute of Technology, Pasadena, CA 91125

This work was supported in part by the NSF under grant no. CCR-0133818, by the Office of Naval Research under grant no. N00014-02-1-0578, and by Caltech's Lee Center for Advanced Networking.

[‡]Information Systems Laboratory, Stanford University, Stanford, CA 94309

1 Introduction

Recently, the pursuit of high-speed wireless data services has generated a significant amount of activity in the communications research community. The physical limitations of the wireless medium present many challenges to the design of high-rate reliable communication systems. As shown in [1], multi-antenna wireless communication systems are capable of providing data transmission at potentially very high rates. In multi-antenna systems, space-time [2, 4] (along with traditional error-correcting) codes are often employed at the transmitter to induce diversity. Furthermore, to secure high reliability of the data transmission, special attention has to be payed to the receiver design. However, good decoding schemes may result in high complexity of the receiver.

A low-complexity detection scheme for multi-antenna systems in a fading environment has been proposed in [4]. This detection scheme (so-called "nulling-and-canceling"), depending on the adopted criterion, essentially performs zero-forcing or minimum-mean-square-error decision feedback equalization on block transmissions. In [5], a technique referred to as the "sphere decoding" (based on the Fincke-Pohst algorithm [6]) was proposed for lattice code decoding and further adapted for space-time codes in [7]. The sphere decoder provides the maximum-likelihood (ML) estimate of the transmitted signal sequence and so often significantly outperforms heuristic nulling and cancelling. Moreover, it was generally believed that sphere decoding requires much greater computational complexity than the cubic-time nulling and cancelling techniques. However, in [8] an analytic expression for the expected complexity of the sphere decoding has been obtained where it is shown that, over a wide range of rates and signal-to-noise ratios (SNRs), the expected complexity is polynomial-time (often sub-cubic). This implies that in many cases of interest ML performance can be obtained with complexity similar to nulling and cancelling.

Another area of intense research activity is that of soft iterative decoding. Such techniques have been reported to achieve impressive results for codes with long codeword length. Following the seminal paper by Berrou et al. [9], there have been many results on turbo decoding of concatenated codes, with performances approaching the Shannon limit on single-input single-output systems (see [10] and the references therein). More recently, low-density parity check (LDPC) codes, long neglected since their introduction by Gallager [11], have also been resurrected (see, e.g., [12, 13]).

Crucial to both turbo and LDPC decoding techniques is the use of the probabilistic ("soft") information about each bit in the transmitted sequence. For multi-antenna systems employing space-time codes it is not clear what is the best way to obtain this soft-information with low complexity. As noted in [14], where turbo-coded modulation for multi-antenna systems has been studied, if soft information is obtained by means of an exhaustive search, the computational complexity grows exponentially in the number of transmit antennas and in the size of the constellation. Hence, for high-rate systems with large number of antennas, the exhaustive search proves to be practically infeasible. Therefore, heuristics are often employed to obtain soft channel information [14]. [Also, see [15] and the references therein for a related work in the context of multi-user detection.] In [16, 17], two variations of the sphere decoding algorithm have been proposed for estimating the soft information. In [17], sphere decoding has been employed to obtain a list of bit sequences that are "good" in a likelihood sense. This list is then used to generate soft information, which is subsequently updated by iterative channel decoder decisions.

In this paper, we propose a MIMO detector, based on a modification of the original Fincke-Pohst algorithm, which efficiently obtains soft information for the transmitted bit sequence. This modified Fincke-Pohst algorithm essentially performs a maximum a posteriori (MAP) search and thus provides soft information for the channel decoder. The channel decoder's output is then fed back to the Fincke-Pohst MAP (FP-MAP) for the next iteration. [Note that the channel decoder may be iterative as well, as in the cases when the channel code is turbo or LDPC.] Our method differs from that of [17] in that the sphere decoder is modified (to allow for the introduction of soft information from the iterative decoder), that the detector performs MAP search, and that FP-MAP is repeated for each iteration. Furthermore, in addition to the schemes with traditional modulation techniques, we study the multi-antenna systems employing space-time codes – in particular, linear-dispersive (LD) codes of [19]. We show that the LD codes allow for an efficient implementation of the FP-MAP algorithm, and illustrate excellent performance of the proposed scheme via simulations. The LD codes are designed to optimize the mutual information between the transmitted and received signals. Maximizing the mutual information is a necessary condition to obtain the excellent performances promised by the powerful channel codes. We illustrate this by means of comparison with a space-time modulation scheme that does not optimize the above mentioned mutual information – in particular, an orthogonal design [3].

The paper is organized as follows. The channel model and problem statement are in Section 2. The Fincke-Pohst algorithm is described and the calculation of its expected complexity is outlined in Section 3. In Section 4, we introduce the MAP modification of the Fincke-Pohst algorithm, discuss its complexity and present simulation results of the performance of the FP-MAP algorithm in systems using convolutional, turbo, and LDPC codes. In Section 6, we study iterative decoding in multi-antenna systems employing LD space-time modulation and forward channel coding. Finally, we conclude the paper in Section 7.

2 System Model

We assume a discrete-time block-fading multi-antenna channel model, where the channel is known to the receiver. This is a reasonable assumption for communication systems where the signaling rate is much faster than the pace at which the propagation environment changes, so that the channel may be learned, e.g., via transmitting known training sequences.

During any channel use the transmitted signal $\mathbf{s} \in \mathcal{Z}^{M\times 1}$ and received signal $\mathbf{x} \in \mathcal{C}^{N\times 1}$ are related by

$$\mathbf{x} = \sqrt{\frac{\rho}{M}}\mathbf{H}\mathbf{s} + \mathbf{v}, \tag{1}$$

where $\mathbf{H} \in \mathcal{C}^{N\times M}$ is the known channel matrix, and $\mathbf{v} \in \mathcal{C}^{N\times 1}$ is the additive noise vector, both comprised of independent, identically distributed complex-Gaussian entries $\mathcal{C}(0,1)$. If we assume that the entries of $\mathbf{s}$ and $\mathbf{H}$ have, on the average, unit variance, then ρ is the expected received signal-to-noise ratio (SNR). The channel is used multiple times to transmit a vector of data.

An iterative decoding scheme is shown in Figure 1. The vector of information bits $\mathbf{b}$ is encoded with an error-correcting code to obtain the vector of coded bits $\mathbf{c}'$, which is then interleaved to result in the vector $\mathbf{c}$. The vector $\mathbf{c}$ is modulated onto a QAM-constellation. Assume that each constellation symbol represents p_m modulated bits (e.g., for a Q-QAM constellation, $p_m = \log_2 Q$). Then the modulation is performed by taking blocks of vector $\mathbf{c}$ of length $p_m M$ and mapping them (e.g., by means of a simple gray mapping) into M-dimensional symbol vectors. The resulting symbols are transmitted across the channel as described by model (1). Therefore, a block of $p_m M$ coded bits (corresponding to a single symbol vector) is transmitted per each channel use. Let us denote these blocks of coded bits as $\mathbf{c}^{[1]}, \mathbf{c}^{[2]}, \ldots \mathbf{c}^{[p_c]}$. Assume, for simplicity, that the total length of the vector $\mathbf{c}$ is $p_c p_m M$. Then the entire vector $\mathbf{c}$ can be blocked as

$$\mathbf{c} = [\mathbf{c}^{[1]} \quad \mathbf{c}^{[2]} \quad \ldots \quad \mathbf{c}^{[p_c]}], \tag{2}$$

and transmitted in p_c channel uses.

Consider the k^{th} channel use (i.e., the block $\mathbf{c}^{[k]}$ has been modulated onto symbol vector $\mathbf{s}$ and transmitted across the channel). On the receiver side, the received vector $\mathbf{x}$ and a priori probabilities of the components of the symbol vector $\mathbf{s}$, $\{p(\mathbf{s}_1), p(\mathbf{s}_2), \ldots, p(\mathbf{s}_M)\}$, are processed by a MIMO detector in order to obtain both the estimated bits in the current block $\mathbf{c}^{[k]}$ and the reliability information about those decisions. Let us denote bits in the block $\mathbf{c}^{[k]}$ by c_i, $i = 1, 2, \ldots, p_m M$. The reliabilities of the decisions for the coded bits c_i can be expressed in the form of a log-likelihood ratio (LLR) as

$$L_1(c_i|\mathbf{x}) = \log \frac{p[c_i = +1|\mathbf{x}]}{p[c_i = -1|\mathbf{x}]}. \tag{3}$$

[Note: we will represent logical 0 with amplitude level -1, and logical 1 with amplitude level $+1$.] Let us denote the reliability information for the block $\mathbf{c}^{[k]}$ by

$$\mathbf{L}_1^{[k]} = [L_1(c_1|\mathbf{x}) \quad L_1(c_2|\mathbf{x}) \quad \ldots \quad L_1(c_{p_m M}|\mathbf{x})],$$

and let $\mathbf{L}_1$ denote a vector of concatenated blocks of reliabilities,

$$\mathbf{L}_1 = [\mathbf{L}_1^{[1]} \quad \mathbf{L}_1^{[2]} \quad \ldots \quad \mathbf{L}_1^{[p_c]}],$$

collected over all p_c uses of the channel. Then $\mathbf{L}_1$ is a vector of LLRs corresponding to all the bits in the vector $\mathbf{c}$.

The vector $\mathbf{L}_1$ is de-interleaved to obtain vector $\mathbf{L}_1'$, which is then used by a channel decoder to form the estimate of the information bit vector, $\widehat{\mathbf{b}}$, as well as to provide $\mathbf{L}_2'$, the a posteriori reliability information for the coded bits vector $\mathbf{c}'$. A posteriori reliability information for the vector $\mathbf{c}$ is obtained by de-interleaving $\mathbf{L}_2'$ into $\mathbf{L}_2$. Let us denote the a posteriori reliability information for the block $\mathbf{c}^{[k]}$ by $\mathbf{L}_2^{[k]}$. Furthermore, assume that the bits c_i, $i = 1, 2, \ldots, p_m M$, in the block $\mathbf{c}^{[k]}$ are independent (which, for a long vector $\mathbf{c}$ and an efficient interleaver is a valid approximation). Then the a posteriori probabilities for the components of the symbol vector $\mathbf{s}$ (symbol vector corresponding to the block $\mathbf{c}^{[k]}$) can easily be found from $\mathbf{L}_2^{[k]}$ using the modulator mapping function. These probabilities, $\{p(\mathbf{s}_1), p(\mathbf{s}_2), \ldots, p(\mathbf{s}_M)\}$, can now be used to run the MIMO detector algorithm (i.e., evaluate (3)) once again. Hence, the MIMO detector is an iterative one, and we use the described scheme for iterative joint detection and decoding in a MIMO system. [Note that for the first iteration of the MIMO detector, we assume that all symbols are equally likely.]

The structure of the channel decoder depends upon the choice of the error-correcting code. For a simple convolutional code, the channel decoder is a simple soft-in soft-out decoder, such as the BCJR algorithm of [18]. When the channel code is a powerful turbo code, then the channel decoder is iterative itself ([10]). If the channel code is an LDPC, the channel decoder is an iterative one, employing message passing algorithms of [11].

The computational complexity of traditional algorithms for evaluating (3) can be prohibitive for applications in multi-antenna systems. Since the sphere decoding algorithm of Fincke and Pohst can supply us with the ML estimate of $\mathbf{s}$ with reasonable complexity, one may speculate whether a modification can be devised to yield soft information with low complexity. To show that this can be done, and to describe how to efficiently approximate the LLRs in (3), it is instructive to review the original Fincke-Pohst algorithm [6].

3 Fincke-Pohst Algorithm

We assume that the components of the transmitted symbol vector $\mathbf{s}$ in (1) are from a complex-valued QAM constellation. To state the ML detection as an integer least-squares problem, we first find the real-valued equivalent of the equation (1). To this end, let $m = 2M$, $n = 2N$, and let s, x, and v denote real vectors obtained from $\mathbf{s}$, $\mathbf{x}$, and $\mathbf{v}$, respectively, as

$$s = \left[\mathcal{R}(\mathbf{s})^T \quad \mathcal{I}(\mathbf{s})^T\right]^T, \quad x = \left[\mathcal{R}(\mathbf{x})^T \quad \mathcal{I}(\mathbf{x})^T\right]^T, \quad \text{and} \quad v = \left[\mathcal{R}(\mathbf{v})^T \quad \mathcal{I}(\mathbf{v})^T\right]^T.$$

Furthermore, let $H \in \mathcal{R}^{n\times m}$ be given by

$$H = \sqrt{\frac{\rho}{M}} \begin{bmatrix} \mathcal{R}(\mathbf{H}) & \mathcal{I}(\mathbf{H}) \\ -\mathcal{I}(\mathbf{H}) & \mathcal{R}(\mathbf{H}) \end{bmatrix}.$$

Then the real-valued equivalent of (1) is given by

$$x = Hs + v.$$

The ML detector maximizes the likelihood that x was received given that s was sent,

$$\max_{s\in\mathcal{D}_L^m} p_{x|s}(x|s). \tag{4}$$

The search space $\mathcal{D}_L^m$ is a finite subset of the (shifted) m-dimensional integer lattice $\mathcal{Z}^m$, which reflects the fact that the unknown (complex) symbols $\mathbf{s}$ are from a QAM constellation. Therefore, $\mathcal{D}_L^m$ is an L-PAM constellation,

$$\mathcal{D}_L^m = \left\{-\frac{L-1}{2}, -\frac{L-3}{2}, \ldots, \frac{L-3}{2}, \frac{L-1}{2}\right\}^m,$$

where L is usually a power of 2.

Since H is known and the noise is zero-mean unit-variance Gaussian, the conditional distribution of x given s is

$$p_{x|s}(x|s) = \frac{1}{(2\pi)^{m/2}} e^{-\|x-Hs\|^2}.$$

Hence maximization (4) is equivalent to the optimization problem

$$\min_{s\in\mathcal{D}_L^m} \|x - Hs\|^2. \tag{5}$$

Problem (5) is referred to as an *integer least-squares problem* and it is known to be NP-hard. Geometrically, it corresponds to the search for the "closest" point in a skewed lattice Hs to a given n-dimensional vector x.

The basic idea of the Fincke-Pohst (FP) algorithm is that rather than search over the entire lattice, one should search only over lattice points in a hypersphere of radius r around x. Then the closest lattice point inside the hypersphere is the solution to (5). To perform the search, however, one needs to: (i) determine an appropriate radius r, and (ii) find the lattice points inside the sphere.

The algorithm of Fincke and Pohst does not address the choice of r, but it does propose an efficient way of finding all the points inside the hypersphere. In particular, the algorithm constructs a tree, whose nodes at the k-th level correspond to the lattice points lying inside the sphere of radius r and dimension k. To find the lattice points inside a sphere of radius r and dimension m, the algorithm performs a depth-first tree search over all lattice points of radius r and dimensions $k = 1, 2, \ldots. m$. The nodes in the tree which correspond to the points outside the sphere are pruned. This is illustrated in Figure 2, for an $m = 4$ dimensional lattice which has $L = 4$ points in each dimension (i.e., $s \in \mathcal{D}_4^4$). [Remark: We need the tree-search interpretation for the discussion on the complexity of the algorithms; further details can be found in [8].]

The search radius r can be chosen according to the statistical description of the noise. Note that $||v||^2 = ||x - Hs||^2$ is a χ^2 random variable with n degrees of freedom. We choose the radius r to be a linear function of the variance of $||v||^2$,

$$r^2 = \alpha n,$$

where the coefficient α is chosen in such a way that with a high probability p_{fp} we find a lattice point inside a sphere,

$$\int_0^{\alpha n} \frac{\lambda^{n/2-1}}{\Gamma(n/2)} e^{-\lambda} d\lambda = p_{\text{fp}}. \tag{6}$$

We find α in (6) by a simple table look-up.

3.1 Computational Complexity of Fincke-Pohst Algorithm

As noted above, the FP algorithm performs a search over all lattice points of radius r and dimensions $k = 1, 2, \ldots, m$. Hence the complexity of the algorithm is proportional to the number of lattice points visited. In general, the algorithm has worst-case and average complexity that is exponential in the number of unknowns m (see [8]). However, in communications applications, as it is implied by (1), the vector x is not arbitrary, but is a lattice point perturbed by additive noise with known statistical properties. Hence, in this case, the expected complexity is a relevant figure of merit. The expected complexity of the FP algorithm is proportional to the expected number of lattice points that the algorithm visits. We need two key ingredients to calculate this expected number of lattice points (and, consecutively, the expected complexity):

1. A probability that an arbitrary lattice point s_a belongs to a k-dimensional sphere of radius r around the transmitted point s_t; it was shown in [8] that this probability is given by the following incomplete gamma function:
$$p_{s_a} = \gamma\left(\frac{\alpha n}{1 + \frac{12\rho}{m(L^2-1)}||s_t - s_a||^2}, \frac{k}{2}\right). \tag{7}$$
2. A technique for enumerating points s_a in the lattice with respect to the transmitted point s_t; in [8], an efficient method for counting those lattice points which yield the same argument of the gamma function in (7), based on certain generating functions, is developed.

Using 1., 2., from above, one can find the number of lattice points visited by Fincke-Pohst algorithm and, therefore, the analytic expression for its expected complexity. The details can be found in [8]. E.g., the complexity of the FP algorithm for a 2-PAM constellation is
$$C(m,\rho) = \sum_{k=1}^{m}(2k+17)\sum_{q=0}^{k}\begin{pmatrix} k \\ q \end{pmatrix}\gamma\left(\frac{\alpha m}{1 + \frac{12\rho q}{m(L^2-1)}}, \frac{k}{2}\right). \tag{8}$$
For a 4-PAM constellation it is
$$C(m,\rho) = \sum_{k=1}^{m}(2k+17)\sum_{q}\frac{1}{2^k}\sum_{l=0}^{k}\begin{pmatrix} k \\ l \end{pmatrix} g_{kl}(q)\gamma\left(\frac{\alpha m}{1 + \frac{12\rho q}{m(L^2-1)}}, \frac{k}{2}\right), \tag{9}$$
where $g_{kl}(q)$ is the coefficient of x^q in the polynomial
$$(1 + x + x^4 + x^9)^l(1 + 2x + x^4)^{k-l}.$$
Similar expressions can be obtained for 8-PAM, 16-PAM, etc., constellations.

For a wide range of m, L and ρ, the sphere decoding algorithm has complexity comparable to cubic-time methods such as nulling and cancelling (cubic in m). As a general principle, for a fixed m, the complexity decreases by increasing the SNR ρ or by decreasing L.

4 Modified FP Algorithm for MAP Detection

The MAP detector maximizes the posterior probability $p_{s|x}(s|x)$,

$$\max_{s\in\mathcal{D}_L^m} p_{s|x}(s|x). \tag{10}$$

Using Bayes' rule,

$$\text{arg max } p_{s|x}(s|x) = \text{arg max } \frac{p_{x|s}(x|s)p_s(s)}{p_x(x)} = \text{arg max } p_{x|s}(x|s)p_s(s)$$

Further, by assuming that the symbols $s_1, s_2, \ldots, s_m$ are independent, we can write

$$p_s(s) = \prod_{k=1}^{m} p(s_k) = e^{\sum_{k=1}^{m} \log p(s_k)}.$$

Then, for a known channel in AWGN, (10) is equivalent to the optimization problem

$$\min_{s\in\mathcal{D}_L^m} \left[\|x - Hs\|^2 - \sum_{k=1}^{m} \log p(s_k) \right] \tag{11}$$

For an iterative decoding scheme, we also require soft information, i.e., the probability that each bit is decoded correctly. To this end, consider the LLR defined in (3) and, as in Section 2, consider the k^{th} channel use (that is, the current symbol vector s is obtained by modulating coded block $\mathbf{c}^{[k]} = [c_1 \quad c_2 \quad \ldots \quad c_{p_m M}]$ onto an L-PAM constellation):

$$L_1(c_i|x) \;=\; \log\frac{p[c_i=+1|x]}{p[c_i=-1|x]} = \log\frac{p[x,c_i=+1]}{p[x,c_i=-1]} = \log\frac{\sum_{\mathbf{c}^{[k]}\,:\,c_i\,=\,+1} p[x|\mathbf{c}^{[k]}]p[\mathbf{c}^{[k]}]}{\sum_{\mathbf{c}^{[k]}\,:\,c_i\,=\,-1} p[x|\mathbf{c}^{[k]}]p[\mathbf{c}^{[k]}]}. \tag{12}$$

Assuming independent bits $c_1, c_2, \ldots, c_{p_m M}$, (12) becomes

$$L_1(c_i|x) = \underbrace{\log\frac{p[c_i=+1]}{p[c_i=-1]}}_{L_{1a}(c_i)} + \underbrace{\log\frac{\sum_{\mathbf{c}^{[k]}\,:\,c_i\,=\,+1} p[x|\mathbf{c}^{[k]}] \prod_{j,j\neq i} p[c_j]}{\sum_{\mathbf{c}^{[k]}\,:\,c_i\,=\,-1} p[x|\mathbf{c}^{[k]}] \prod_{j,j\neq i} p[c_j]}}_{L_{1e}(c_i)},$$

where $L_{1a}(c_i)$ and $L_{1e}(c_i)$ denote so-called *a priori* and *extrinsic* part of the total soft information, respectively. [Note that, when used in an iterative decoding scheme, it is only $L_{1e}(c_i)$ that is passed to the other decoding block(s) in the scheme.] Since the block $\mathbf{c}^{[k]}$ is uniquely mapped into the symbol vector s, it follows that for an AWGN channel

$$L_1(c_i|x) = \log \frac{\sum\limits_{s\,:\,c_i\,=\,+1} p[x|s] \prod\limits_j p[s_j]}{\sum\limits_{s\,:\,c_i\,=\,-1} p[x|s] \prod\limits_j p[s_j]} = \log \frac{\sum\limits_{s\,:\,c_i\,=\,+1} e^{-\|x-Hs\|^2+\sum_j \log p[s_j]}}{\sum\limits_{s\,:\,c_i\,=\,-1} e^{-\|x-Hs\|^2+\sum_j \log p[s_j]}} \tag{13}$$

Computing (13) over the entire signal space $\mathcal{D}_L^m$ is of prohibitive complexity. Instead, we constrain ourselves to those $s \in \mathcal{D}_L^m$ for which the argument in (11) is small. [Note that these are the signal vectors whose contribution to the numerator and denominator in (13) is significant.]

Applying the idea of the Fincke-Pohst algorithm, we search for the points s that belong to the geometric body described by

$$r^2 \geq (s-\hat{s})^* R^* R(s-\hat{s}) - \sum_{k=1}^{n} \log p(s_k), \tag{14}$$

where R is the lower triangular matrix obtained from the QR factorization of H. (Note that this is no longer a hypersphere.) The search radius r in (14) can be chosen according to the statistical properties of the noise and the a priori distribution of s.

A necessary condition for s_m to satisfy (14) readily follows,

$$r_{mm}^2 \left(s_m - \hat{s}_m\right)^2 - \log p(s_m) \leq r^2. \tag{15}$$

Moreover, for every s_m satisfying (15), we define

$$r_{m-1}^2 = r^2 - r_{mm}^2 \left(s_m - \hat{s}_m\right)^2 + \log p(s_m),$$

and obtain a stronger necessary condition for (14) to hold,

$$r_{m-1,m-1}^2 \left(s_{m-1} - \underbrace{\hat{s}_{m-1} + \frac{r_{m-1,m}}{r_{m-1,m-1}}(s_m - \hat{s}_m)}_{\hat{s}_{m-1|m}} \right)^2 - \log p(s_{m-1}) \leq r_{m-1}^2.$$

The procedure continues until all the components of vector s are found. The FP-MAP algorithm can be summarized as follows:

Input: R, x, $\hat{s}$, r, $p_s(\mathbf{s})$.

1. Set $k = m$, $r_m'^2 = r^2 - \|x\|^2 + \|H\hat{s}\|^2$, $\hat{s}_{m|m+1} = \hat{s}_m$
2. (Bounds for s_k) Set $z = \frac{r_k'}{r_{kk}}$, $UB(s_k) = \lfloor z + \hat{s}_{k|k+1} \rfloor$, $s_k = \lceil -z + \hat{s}_{k|k+1} \rceil - 1$.
3. (Increase s_k) $s_k = s_k + 1$. If $r_{kk}^2 \left(s_k - \hat{s}\right)^2 > r_k'^2 + \log p(s_k)$ and $s_k \leq UB(s_k)$, go to 3, else proceed. If $s_k \leq UB(s_k)$ go to 5, else to 4.
4. (Increase k) $k = k + 1$; if $k = m + 1$, terminate algorithm, else go to 3.
5. (Decrease k) If $k = 1$ go to 6. Else $k = k - 1$, $\hat{s}_{k|k-1} = \hat{s}_k + \sum_{j=k+1}^{m} \frac{r_{kj}}{r_{kk}}(s_j - \hat{s}_j)$, $r_k'^2 = r_{k+1}'^2 - r_{k+1,k+1}^2(s_{k+1} - \hat{s}_{k+1|k+2})^2 + \log p(s_{k+1})$, and go to 2.
6. Solution found. Save s and go to 3.

Assume that the search yields the set of points $\mathcal{S} = \{s^{(1)}, s^{(2)}, \ldots, s^{(l_s)}\}$. The vector $s \in \mathcal{S}$ that minimizes (11) is the solution to the MAP detection problem (10). The soft information for each bit c_i can be estimated from (13), by only summing the terms in the numerator and denominator such that $s \in \mathcal{S}$.

4.1 Computational Complexity of FP-MAP Algorithm

The complexity of the FP-MAP algorithm can, in principle, be found following the outline of the calculation of complexity of the original Fincke-Pohst algorithm in Section 3.1. However, the probability that an arbitrary point s_a belongs to a k-dimensional sphere of radius r around the transmitted point s_t (which we need to compute the expected number of points the FP-MAP algorithm visits) now becomes

$$p_{\mathbf{s}_a} = \gamma \left(\frac{\alpha n + \frac{12\rho}{m(L^2-1)} \sum_{j=1}^{m} \log p(s_j)}{1 + \frac{12\rho}{m(L^2-1)} \|s_t - s_a\|^2}, \frac{k}{2} \right). \tag{16}$$

The argument of this probability function is not as simple as the one in (7), and the computation of the expected number of points is much more difficult. First and foremost, (16) is a function of the a priori probabilities which are generally not known in advance to iterations. Second, since each point s_a in a lattice has a distinct a priori probability affiliated with it, argument of the probability function (16) will, in general, be different for each pair of points (s_t, s_a). Thus an efficient enumeration that would help the complexity calculation cannot be done. Hence, to compute the expected number of points, one needs to consider all the possible pairs of points (s_t, s_a) and the corresponding probabilities (16) which, as the size of the problem increases, clearly becomes rather cumbersome. However, we note that since $\log p(s_j) \leq 0,\ j = 1, \ldots, m$, we have

$$\alpha n + \frac{12\rho}{m(L^2-1)} \sum_{j=1}^{m} \log p(s_j) \leq \alpha n.$$

Hence from (7) and (16) it follows that, for the same choice of radius r,

$$p_{s_a}^{FP-MAP} \leq p_{s_a}^{FP},$$

and we conclude that, for the same choice of r, the expected number of points that the FP-MAP algorithm visits is upper bounded by the expected number of points visited by the original sphere decoding algorithm. Thus, the expected complexity of the FP-MAP is roughly upper bounded by the expected complexity of the sphere decoding, for the same choice of r. ["Roughly" upper bounded because since the a priori probabilities enter the algorithm, there are a few (two, to be exact) additional operations per each visited point; this is accounted for by changing $(2k + 17)$ to $(2k + 19)$ in (8) and (9).]

Therefore, the results of [8] suggest that the expected complexity of the FP-MAP algorithm is polynomial in m over a wide range of rates and SNRs. Generally, we chose the search parameter r so that there is sufficiently many points to make a good approximation of (13). Note that the logarithms in (13) can be efficiently computed using the standard Log-MAP implementation [20].

4.2 Simulation Results

We consider bit-error rate (BER) performance of the system with $M = 4$ transmit and $N = 4$ receive antennas and 16-QAM modulation scheme without a space-time coding (i.e., simple Grey mapping is used for modulation). Figure 3 shows the BER performance of the system employing a simple rate $R = 1/2$ convolutional code with length 9216 information bits, memory length 2, and generating polynomials $G_1(D) = 1 + D^2$ (feedforward) and $G_2(D) = 1 + D + D^2$ (feedback). The FP-MAP algorithm is used to obtain the soft information. Figure 4 compares the performance of the FP-MAP with that of the soft nulling and cancelling (N/C) algorithm. For each entry in a transmitted symbol vector, the soft N/C algorithm cancels the previously decoded symbols and obtains the soft information using the distribution of the noise. [The soft N/C algorithm is similar to the soft MMSE equalizer of [21]. Also, see [15] for an application in multi-user context.] Prior to the decoding, symbols are optimally ordered. The complexity of the soft N/C algorithm is roughly cubic (due to the required QR-factorization of the channel matrix). The FP-MAP even with a single iteration outperforms 4 iterations of the soft N/C by 2dB at BER of 10^{-4}.

Figure 5 shows the BER performance of the system with a parallel concatenated turbo code with rate $R = 1/2$ and length 9216 information bits. The constituent convolutional codes are as the one described above. For each iteration of the FP-MAP, the turbo (decoder performs 8 iterations of its own. Figure 6 shows the BER performance of the same system (4×4), with 4-QAM constellation and $8/9$ LDPC code of length 1088, column weight 4. When the LDPC decoder receives soft information from FP-MAP, it performs 8 iterations before passing what it inferred about the coded bits back to FP-MAP. In Figure 5 – Figure 6, the dashed vertical line denotes the capacity limits of the MIMO channel.

The turbo coded scheme in Figure 5 gets 3.3dB close to capacity. At BER of 10^{-5}, it outperforms the convolutional code employed on the same system by approximately 3dB. The rate of the system is 8 bits per channel use. The LDPC code, on the other hand, is about 4.5dB away from capacity of the system in which it is employed; the data rate in this system is 7.1 bits per channel use. Although the LDPC code is outperformed by the turbo code, it proves to be an interesting alternative, especially in light of the complexity exponents $\log_m \mathcal{C}(m, L, \rho)$ shown in Figure 7. At SNR $\approx$ 10dB, both schemes have BER $\approx 10^{-5}$. As indicated in Figure 7, for such SNRs, the complexity of the detection in the system employing the (high rate) LDPC code and 4-QAM modulation is approximately cubic in m, while the complexity in the system with the $(1/2$ rate) turbo code and 16-QAM modulation is significantly higher. So, although the mutual information plots in Figure 7 imply that the system with 16-QAM modulation scheme is more efficient, the system that employs 4-QAM modulation is more favorable from the complexity point of view.

5 Systems employing both channel and S-T codes

Space-time coding has been developed to fully exploit the spatial diversity provided by a wireless link. There has been a tremendous amount of the research activities in the field (see, e.g., [2]-[3]). In this section, we consider powerful channel codes for data encoding in the multi-antenna systems that employ space-time codes. We focus on the LD space-time codes of [19], both for the simplicity of the decoding as well as for the certain optimality properties whose importance will become clear later in the section.

Space-time coding is a modulation technique that imposes spatial and temporal correlation onto a transmitted sequence of modulated symbols. This correlation is generally embedded over a number of channel uses. Therefore, we shall find it useful to model the transmission over a number of channel uses, say T, during which the channel remains constant. In other words, we assume that

$$\mathbf{x}_i = \sqrt{\frac{\rho}{M}} H \mathbf{s}_i + \mathbf{v}_i, \quad i = 1, \ldots, T,$$

and hence we can write

$$X = \sqrt{\frac{\rho}{M}} S H + V, \tag{17}$$

where $X = [\mathbf{x}_1 \;\; \mathbf{x}_2 \;\; \ldots \;\; \mathbf{x}_T]' \in \mathcal{C}^{T \times N}$ is the received signal, $S = [\mathbf{s}_1 \;\; \mathbf{s}_2 \;\; \ldots \;\; \mathbf{s}_T]' \in \mathcal{C}^{T \times M}$ is the transmitted signal, and $V = [\mathbf{v}_1 \;\; \mathbf{v}_2 \;\; \ldots \;\; \mathbf{v}_T]' \in \mathcal{C}^{T \times N}$ is the additive $\mathcal{C}(0,1)$ noise. Furthermore, the transmitted signal matrix satisfies the power constraint $EtrSS^* = TM$.

A *linear-dispersion code* is one for which

$$S = \sum_{q=1}^{Q} (s_q C_q + s_q^* D_q),$$

where $C_q \in \mathcal{C}^{T \times M}$ and $D_q \in \mathcal{C}^{T \times M}$ are fixed matrices, and $s_q, \; q = 1, \ldots, Q$ are complex scalars. Scalars s_q can be chosen from either PSK or QAM constellations. For simplicity, we shall assume that they are chosen from a D-QAM constellation. The particular choice of scalars $s_1, \ldots, s_Q$ determines a specific codeword from the code that is determined by the set of matrices $\{C_q, D_q\}$. The rate of the LD code is then $R = (Q/T)\log_2 r$.

Matrix S can be considered a symbol that is being transmitted (over T channel uses). The total number of matrices S that can be generated by $\{C_q, D_q\}$ and $s_1, \ldots, s_Q$ is 2^{RT} and thus can be very large. This symbol space can easily be generated by the LD codes. However, its enormous size generally prevents any exhaustive search decoding technique and asks for more sophisticated receiver algorithms. For instance, the system employing $M = 8$ transmit antennas coded over $T = M$ channel uses with a rate $R = 16$ LD code has a symbol space with $2^{RT} \approx 3.4 \cdot 10^{38}$ elements. Real-time exhaustive search over such a large set is out of question.

We shall find it convenient for the decoding purposes to represent the scalar s_q by its real and imaginary parts,

$$s_q = \alpha_q + j\beta_q, \;\; q = 1, \ldots, Q.$$

Denoting $A_q = C_q + D_q$, $B_q = C_q - D_q$, one can write

$$S = \sum_{q=1}^{Q} (\alpha_q A_q + j\beta_q B_q), \tag{18}$$

The linearity of the LD codes (18) in the variables $\{\alpha_q, \beta_q\}$ can be exploited to pose the detection problem as an integer least-squares problem. To this end, we write (17) as

$$X = \sqrt{\frac{\rho}{M}} SH + V = \sqrt{\frac{\rho}{M}} \sum_{q=1}^{Q} (\alpha_q A_q + j\beta_q B_q)H + V.$$

Define

$$\mathcal{A}_q = \begin{bmatrix} A_{R,q} & -A_{I,q} \\ A_{I,q} & A_{R,q} \end{bmatrix}, \;\; \mathcal{B}_q = \begin{bmatrix} -B_{I,q} & -B_{R,q} \\ B_{R,q} & -B_{I,q} \end{bmatrix}, \;\; \underline{h}_n = \begin{bmatrix} h_{R,n} \\ h_{I,n} \end{bmatrix},$$

where the vectors $h_{R,n}$ and $h_{I,n}$, $n = 1, \ldots, N$, denote k^{th} columns of matrices H_R and H_I, respectively. Now, collecting all real and imaginary parts, we can write the expression for the input/output relation as

$$x = \mathcal{H}s + v, \tag{19}$$

where $x = [x_{R,1} \;\; x_{I,1} \;\; \ldots \;\; x_{R,n} \;\; x_{I,N}]'$, $s = [\alpha_1 \;\; \beta_1 \;\; \ldots \;\; \alpha_Q \;\; \beta_Q]'$, $v = [v_{R,1} \;\; v_{I,1} \;\; \ldots \;\; v_{R,N} \;\; v_{I,N}]'$, the $2NT \times 2Q$ real valued equivalent channel matrix is given by

$$\mathcal{H} = \sqrt{\frac{\rho}{M}} H \begin{bmatrix} \mathcal{A}_1 \underline{h}_1 & \mathcal{B}_1 \underline{h}_1 & \ldots & \mathcal{A}_Q \underline{h}_1 & \mathcal{B}_Q \underline{h}_1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathcal{A}_1 \underline{h}_N & \mathcal{B}_1 \underline{h}_N & \ldots & \mathcal{A}_Q \underline{h}_N & \mathcal{B}_Q \underline{h}_N \end{bmatrix},$$

and where $x_{R,k}$, $x_{I,k}$, $v_{R,k}$, and $v_{I,k}$ denote k^{th} column of X_R, X_I, V_R, and V_I, respectively.

The equivalent channel $\mathcal{H}$ is comprised of the dispersion matrices $\{\mathcal{A}_q, \mathcal{B}_q\}$ and the known channel matrix H and is, therefore, known to the receiver.

Note that $s_i \in \mathcal{D}_L$ for $1 \leq i \leq 2Q$, where $\mathcal{D}_L$ denotes an L-PAM constellation ($L = \sqrt{D}$). Then from (19), one can pose the MAP detection problem as

$$\min_s \|x - \mathcal{H}s\|^2 + \sum \log p(s_i), \tag{20}$$

where $p(s_i)$ is the a priori probability of the i^{th} component of the $2Q$-dimensional vector s. Problem (20) allows for efficient implementation of FP-MAP algorithm.

5.1 Performance results

In this section, we present examples that illustrate performance of the LDPC codes in a multi-antenna system with LD space-time modulation. In particular, we use the same $8/9$-rate LDPC code of the length 1088 as in the previous section. Figure 8 shows the performance of the 2×2 system employing a 4-QAM modulation scheme and a rate $R = 4$ LD code (see page 31 in [19] for the construction of the LD code). The significant coding gain due to use of the LDPC code is evident. Even without S-T coding, LDPC code would yield good performance, as illustrated in Section 4. However, S-T coding is necessary in systems with less receive than transmit antennas, to provide as many equations as there are unknowns and provide feasibility of decoding (when $M > N$, the Fincke-Pohst algorithm is exponential in $M - N$). Performance of such a system, with $M = 3$ and $N = 1$, employing the LD code is shown in Figure 9.

In this section, we have focused on LD codes. One can ask whether any S-T code would suffice, i.e., would any modulation scheme support the excellent performance of the powerful channel codes? The answer is negative: S-T modulation schemes need to maximize the mutual information between the input and output signals. Note that the dispersion matrices of the LD code are actually chosen to optimize the mutual information between x and s, i.e., to solve for the optimization problem

$$C_{LD}(\rho, T, M, N) = \max_{A_q, B_q, q=1,\ldots,Q} \frac{1}{2T} E \log\det\left(I_{2NT} + \frac{\rho}{M}\mathcal{H}\mathcal{H}'\right) \tag{21}$$

for a particular ρ of interest and an appropriate power constraint. So, the LD codes satisfy the required criterion. Interestingly, V-BLAST also satisfies this condition. Indeed, as shown in Figure 10, the multi-antenna system employing an LDPC channel code and V-BLAST performs almost as good as the scheme with an LD code (eq. (36) in [19]). Of course, there is an additional gain that the LD code obtains by spreading the signals across space and time more efficiently than V-BLAST. On the other hand, if a S-T code throws away some information (i.e., if it violates (21)), the system performance is inferior in comparison to the S-T code designed with (21) in mind. This is illustrated in Figure 11, where we compare performance of the LD code of Figure 9 with the orthogonal S-T code ([3], also eq. (35) in [19]). The orthogonal design does not satisfy (21) and the performance of the system is clearly much worse than of that employing the LD code.

6 Conclusion

In this paper, we developed a modification of the sphere decoding algorithm to perform the MAP detection and efficiently estimate soft information. When combined with soft iterative decoding schemes, the proposed detection algorithm, FP-MAP, provides close to capacity performances of multi-antenna systems. This was demonstrated on systems employing both turbo and LDPC codes.

We considered the expected complexity of the algorithm and found it is closely related to the expected complexity of the original sphere decoding. In fact, over a wide range of rates and SNRs, the FP-MAP algorithm has complexity that is polynomial in the number of transmit antennas.

Furthermore, we studied the use of powerful channel codes and iterative decoding in multi-antenna systems that employ space-time modulation schemes. We found that in order to obtain the remarkable performance of the iterative decoding, the space-time techniques need to optimize for the mutual information between the transmitted and received symbols. Thus, the design paradigm of [19] that constructs codes based on mutual information appears to be very reasonable.

References

[1] I. E. Telatar, "Capacity of multi-antenna gaussian channels," *Eur. Trans. Telecom.*, 10:585-595, November 1999.

[2] V. Tarokh V, N. Seshadri, and A. R. Calderbank, "Space-time codes for high data rate wireless communication: performance criterion and code construction," *IEEE Trans. on Info. Theory*, 44(2):744-765, March 1998.

[3] B. M. Hochwald and T. L. Marzetta, "Unitary space-time modulation for multiple-antenna communication in Rayleigh flat-fading," *IEEE Trans. Info. Theory*, 46(3):543-564, Mar. 2000.

[4] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs. Tech. J.*, 1(2):41-59, 1996.

[5] E. Viterbo and J. Boutros, "A universal lattice code decoder for fading channels," *IEEE Trans. on Information Theory*, vol. 45, pp. 1639-1642, July 1997.

[6] U. Fincke and M. Pohst, "Improved methods for calculating vectors of short length in a lattice, including a complexity analysis," *Mathematics of Computation*, 44(4):463-471, April 1985.

[7] M. O. Damen, A. Chkeif, and J. C. Belfiore, "Lattice codes decoder for space-time codes," *IEEE Communications Letters*, vol. 4, pp. 161-163, May 2000.

[8] B. Hassibi and H. Vikalo, "Expected complexity of the sphere decoder algorithm," *submitted to the IEEE Trans. on Signal Processing*, 2003.

[9] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding. Turbo codes," in *Proc. Int. Conf. Comm.*, pp. 1064-1070, 1993.

[10] J. P. Woodard and L. Hanzo, "Comparative study of turbo decoding techniques: an overview," *IEEE Trans. on Vehicular Technology*, vol. 49, no. 6, November 2000.

[11] R. G. Gallager, *Low-density parity-check codes*, Cambridge, MA: MIT Press, 1963.

[12] D. J. C. MacKay and R. M. Neal, “Near Shannon limit performance of low-density parity-check codes,” *Electron. Lett.*, vol. 32, pp. 1645-1646, 1996.

[13] D. J. C. MacKay, “Good error correcting codes based on very sparse matrices,” *IEEE Trans. Inform. Theory,* vol. 45, pp. 399-431, March 1999.

[14] A. Stefanov and T. M. Duman, “Turbo-coded modulation for systems with transmit and receive antenna diversity over block fading channels: system model, decoding approaches, and practical considerations,” *IEEE J. on Sel. Areas in Comm.*, vol. 19, no. 5, May 2001.

[15] P. D. Alexander, A. J. Grant, and M. C. Reed, “Iterative detection in code-division multiple-access with error control coding,” *European Trans. on Tel.,* 9:419-425, 1998.

[16] H. Vikalo and B. Hassibi, ”Low-complexity iterative decoding over multiple antenna channels via a modified sphere decoder,” in *Proc. Allerton Conf. on Commun., Control, and Computing*, Monticello, IL, October 2001.

[17] B. M. Hochwald and S. ten Brink, “Achieving near-capacity on a multiple-antenna channel,” in *Proc. Allerton Conf. on Commun., Control, and Computing*, Monticello, IL, October 2001.

[18] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, “Optimal decoding of linear codes for minimizing symbol error rate,” *IEEE Trans. Info. Theory*, pp. 284-287, March 1974.

[19] B. Hassibi and B. Hochwald, “High-rate codes that are linear in space and time,” *Transactions on IEEE Trans. Info. Theory*, 48(7):1804-1824, July 2002.

[20] P. Robertson, P. Hoeher and E. Villebrun, ”Optimal and suboptimal maximum a posteriori algorithms suitable for turbo decoding”, *Euro. Trans. Telecommun.*, 8:119-125, Mar-Apr. 1997.

[21] Tuechler, M., Singer, A., Koetter, R.: “Minimum Mean Squared Error Equalization using A-priori Information,” IEEE Trans. on Sign. Proc., vol. 50, pp. 673-683, 2002.

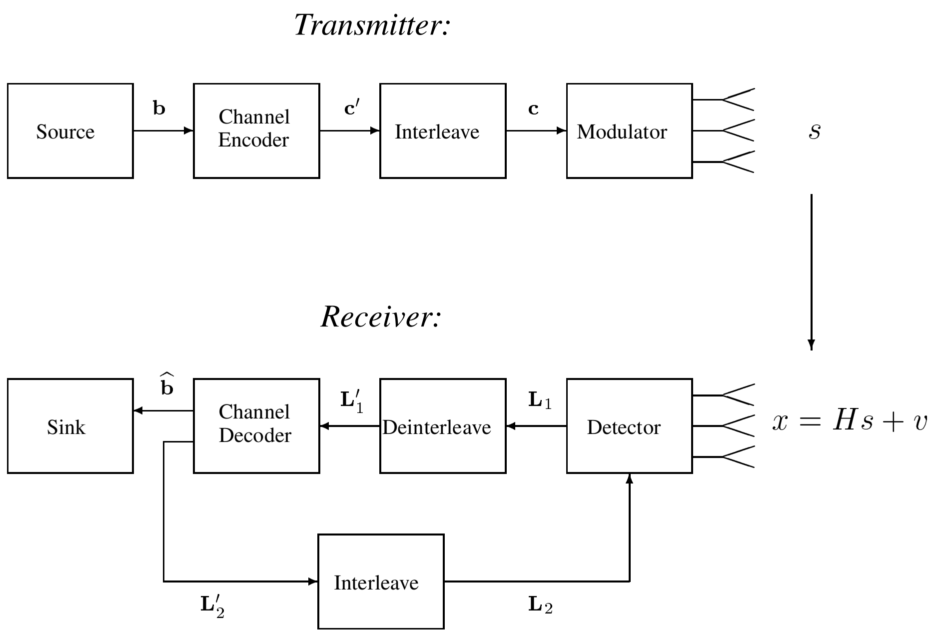

Figure 1: *System model.*

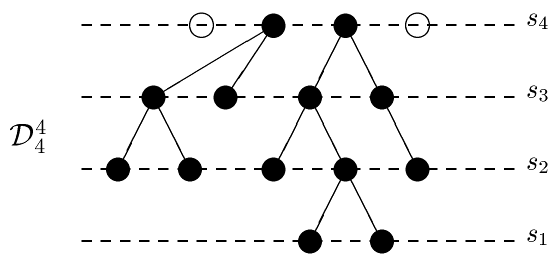

Figure 2: Tree-pruning interpretation of sphere decoding.

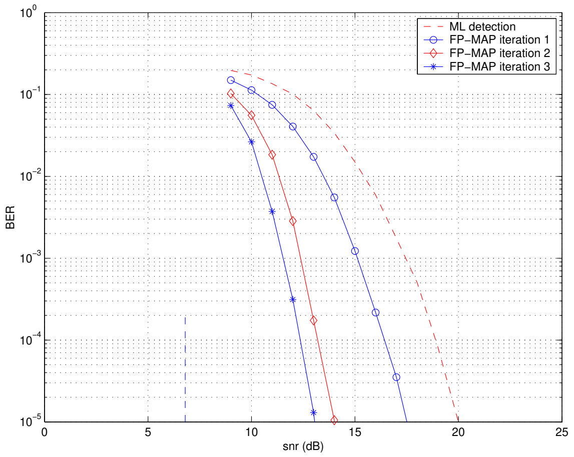

Figure 3: *BER performance of $M = N = 4$ system employing rate $1/2$, 9216 bits long convolutional code, 16-QAM, FP-MAP.*

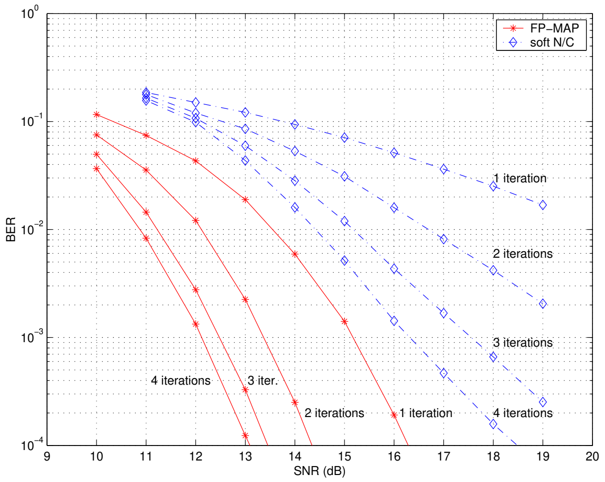

Figure 4: *Comparison of BER performances for FP-MAP and soft N/C employed on $M = N = 4$ system with rate $1/2$, 1000 bits long convolutional code, 16-QAM.*

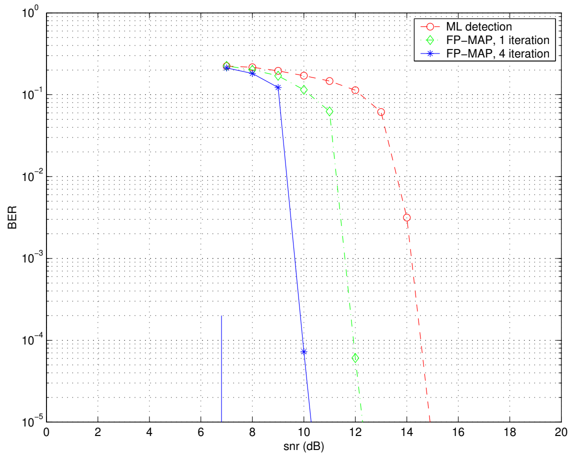

Figure 5: *BER performance of* $M = N = 4$ *system employing rate* $1/2$ *turbo code,* 16-*QAM, FP-MAP.*

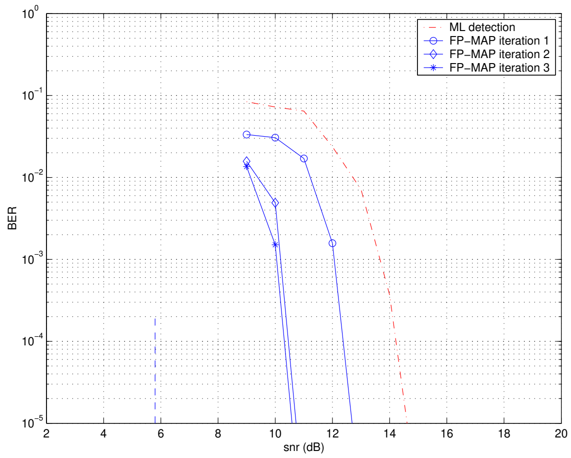

Figure 6: *BER performance of* $M = N = 4$ *system employing rate* $8/9$ *LDPC code,* 4-*QAM, FP-MAP.*

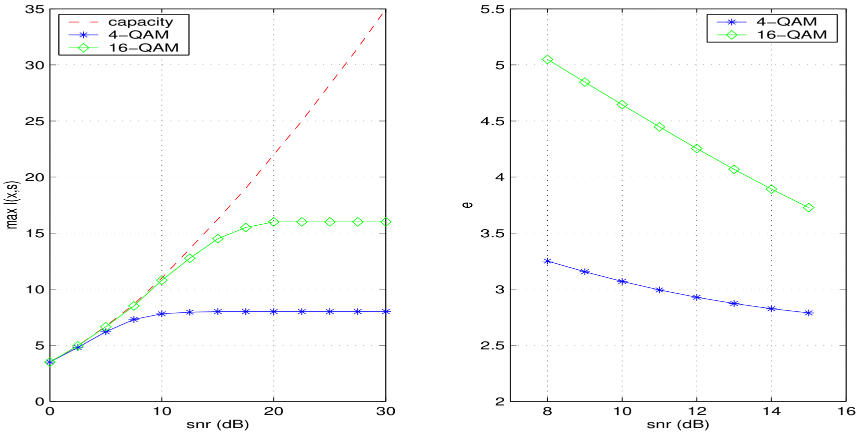

Figure 7: *Maximum mutual information and complexity exponents for 4-QAM and 16-QAM constellations, $M = N = 4$ system.*

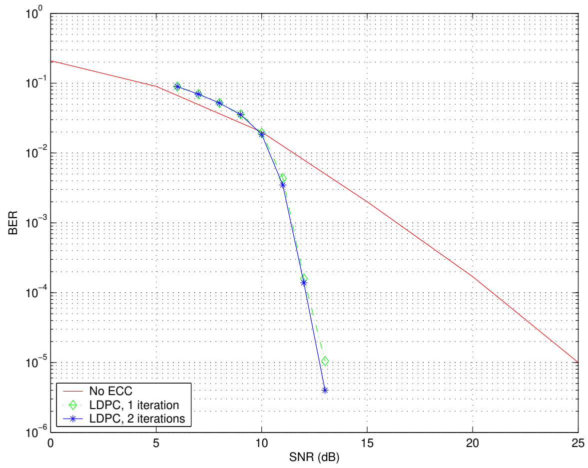

Figure 8: *BER performace of $M = N = 2$ system employing $R = 4$ LD code and rate $8/9$ LDPC code, 4-QAM modulation, FP-MAP.*

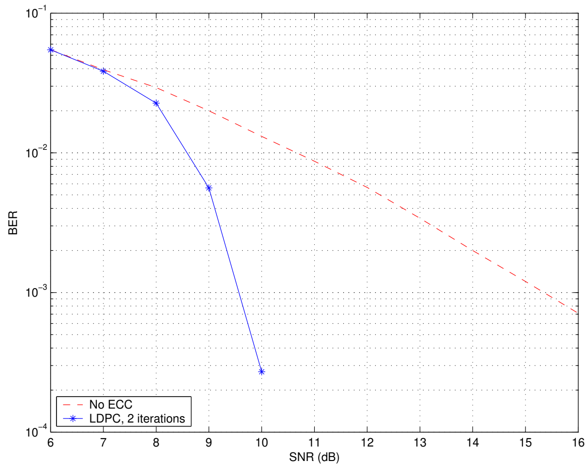

Figure 9: *BER performance of $M = 3$, $N = 1$ system employing $R = 2$ LD code and rate $8/9$ LDPC code, FP-MAP.*

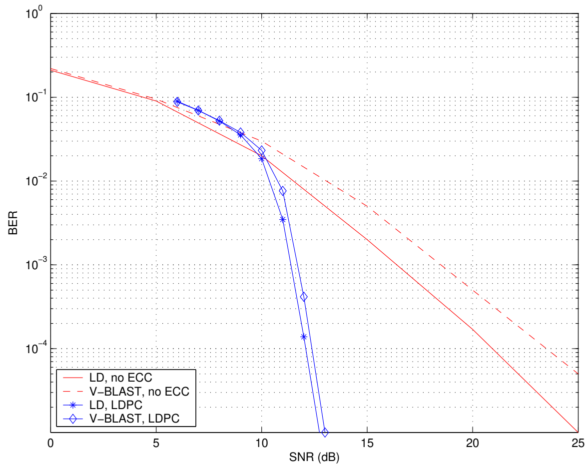

Figure 10: *Comparison of BER performances of LD code vs. V-BLAST (rate $R = 4$) employed on $M = N = 2$ system, using rate $8/9$ LDPC code, FP-MAP.*

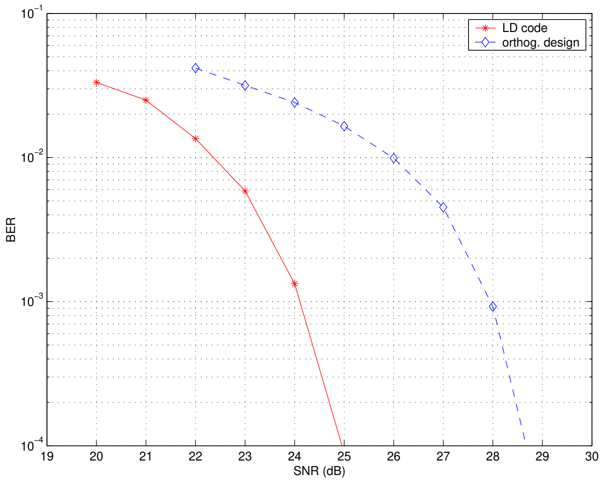

Figure 11: *Comparison of BER performances of LD code vs. orthogonal design on* $M = 3$*,* $N = 1$ *system. Both schemes employ rate* $8/9$ *LDPC code. The system using the LD code uses* 64*-QAM, the system using orthogonal design uses* 256*-QAM so that both have rate* $R = 6$*. The FP-MAP is used for detection.*