

Linköping University Pre-Print

**Partial Marginalization Soft MIMO Detection
with Higher Order Constellations**

Daniel Persson and Erik G. Larsson

N.B.: When citing this work, cite the original article.

©2010 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Daniel Persson and Erik G. Larsson, Partial Marginalization Soft MIMO Detection with Higher Order Constellations, 2010, IEEE Transactions on Signal Processing.
<http://dx.doi.org/10.1109/TSP.2010.2068293>

Postprint available at: Linköping University Electronic Press
<http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-58524>

Manuscript for IEEE Transactions on Signal Processing, (Correspondence Item)

Partial marginalization soft MIMO detection with higher order constellations

Daniel Persson* and Erik G. Larsson

Mailing address:	Department of Electrical Engineering Linköping University 581 83 Linköping Sweden
Email:	danielp@isy.liu.se egl@isy.liu.se
Phone:	+46 13 281351
Fax:	+46 13 139282
OPERATING SYSTEM:	Ubuntu Linux
SOFTWARE FOR FORMATTING:	TeX, BibTeX, Kile

Abstract: A new method for multiple-input multiple-output (MIMO) detection with soft-output, the partial marginalization (PM) algorithm, was recently proposed. Advantages of the method are that it is straightforward to parallelize, and that it offers a fully predictable runtime. PM trades performance for computational complexity via a user-defined parameter. In the limit of high computational complexity, the algorithm becomes the MAP demodulator.

The PM algorithm also works with soft-input, but until now it has been unclear how to apply it for other modulation formats than binary phase-shift keying (BPSK) per real dimension. In this paper, we explain how to generalize PM with soft-input to general signaling constellations, while maintaining the low complexity advantage of the original algorithm.

Partial marginalization soft MIMO detection with higher order constellations

Daniel Persson and Erik G. Larsson,

Department of Electrical Engineering, Linköping University, 581 83 Linköping, Sweden
phone: +46 13 281351, +46 13 281312, email: danielp@isy.liu.se, egl@isy.liu.se.

Abstract

A new method for multiple-input multiple-output (MIMO) detection with soft-output, the partial marginalization (PM) algorithm, was recently proposed. Advantages of the method are that it is straightforward to parallelize, and that it offers a fully predictable runtime. PM trades performance for computational complexity via a user-defined parameter. In the limit of high computational complexity, the algorithm becomes the MAP demodulator.

The PM algorithm also works with soft-input, but until now it has been unclear how to apply it for other modulation formats than binary phase-shift keying (BPSK) per real dimension. In this paper, we explain how to generalize PM with soft-input to general signaling constellations, while maintaining the low complexity advantage of the original algorithm.

Index Terms

Detection, multiple-input multiple-output (MIMO), soft-input, soft-output

I. INTRODUCTION

We are concerned with multiple-input multiple-output (MIMO) communication, where several antennas are used both at the sender and at the receiver side [1]. Specifically, we study the problem of soft demodulation for the case where all antennas transmit independent symbols. Our focus is on systems that use capacity-achieving codes, i.e., turbo and low-density parity-check (LDPC) codes. On the receiving side of the system, iterative demodulation and decoding [2] is employed, see Fig. 1. In these systems, the demodulator and the decoder are exchanging information concerning the likelihood of code bits being 0 and 1, which is referred to as soft information. Both the demodulator and the decoder must thus be able to handle soft-input and soft-output information.

Optimum soft demodulation has a computational complexity that is exponential in the number of transmit antennas, and polynomial in the size of the signal constellation. Several methods have been

devised to approximate the optimal soft demodulator [3]. Low-complexity solutions, such as zero-forcing (ZF) and zero-forcing with decision feedback (ZF-DF), usually provide rather poor performance in most scenarios of practical interest. A more sophisticated method that delivers very good performance is the sphere decoder [4], but its complexity fluctuates substantially from one frame to another, and its expected complexity is exponential in the number of transmit antennas [5]. There are also more recent flavors of the sphere decoder that operate at fixed complexity, both for hard detection [6], and soft demodulation [7].

The soft demodulation method of interest in this paper is the recently proposed partial marginalization (PM) algorithm [8], originally proposed for approximative demodulation without soft-input. The approximation in the PM algorithm consists of two steps. In the first step, a carefully chosen set of marginalization sums is approximated by their largest terms. In the second step, a low-complexity method (ZF-DF, preferably) is used to find these largest terms. The main advantages of PM over the sphere decoder [4] are that it offers a constant and fully predictable runtime, and that it is straightforward to parallelize. PM trades performance for computational complexity via a user-defined parameter, and differently from [6], [7], PM is not based on the Max-log approximation. When setting the user parameter to its largest possible value, the algorithm becomes the optimal (exact) demodulator, and by setting the parameter to zero, one obtains the ZF-DF solution.

In [8], an extension of PM that can exploit soft input, for the case of binary phase-shift keying (BPSK) per real dimension, was also presented. This extension was based on the fact that the logarithmic prior probabilities of the information bits are linear in the modulated symbols, an observation originally made in [9]. A consequence of this is that the soft input can be algebraically incorporated into the problem by performing a completion of squares operation. This results in a computational problem that has the same form as the corresponding demodulation problem without soft input, but with a larger channel matrix. Such an operation can be interpreted in terms of adding virtual antennas to the system, where the virtual antennas carry the soft input information. This extension of the PM method is however only possible for BPSK modulation per real dimension.

This paper's main contribution is an extension of PM that allows *operation with soft input and higher-order constellations of arbitrary size and shape, and with arbitrary mappings between the channel bits and the signal constellation points*. The key idea is to incorporate the soft input into the part of the PM algorithm where the ZF-DF scheme is invoked to find the largest term in a sum. The computational complexity of our proposed technique is essentially the same as that of the original method without soft input in [8].

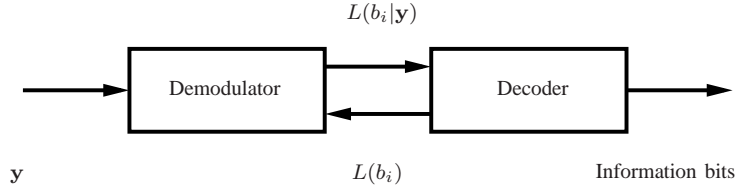


Fig. 1. The MIMO receiver considered in this paper.

II. PRELIMINARIES

We consider a real-valued discrete-time channel model of the form

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

where \mathbf{s} is the N_t -dimensional transmitted vector, consisting of scalar symbols s_n that belong to the M -ary constellation \mathcal{S} , \mathbf{y} is the N_r -dimensional received vector, and the channel matrix $\mathbf{H} \in \mathbb{R}^{N_r \times N_t}$ is completely known at the receiver side. The N_r -dimensional noise vector \mathbf{e} has independent and identically distributed (i.i.d.) Gaussian elements with zero mean and variance $N_0/2$. Hence, we have that

$$p(\mathbf{y}|\mathbf{s}) = \frac{1}{\sqrt{\pi^{N_r} N_0^{N_r}}} \exp\left(-\frac{1}{N_0} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2\right). \quad (2)$$

The model in (1) may be the result of rewriting a complex-valued model with a separable constellation.

We consider the receiver structure in Fig. 1. The signal vector \mathbf{s} carries transmitted bits $b_i \in \{-1, 1\}$, $i = 1, \dots, N_t K$, where $K \triangleq \log_2(M)$ is the number of bits per symbol.¹ Our receiver is of “turbo”-type, i.e., the final estimate of the transmitted codeword is obtained by iterating between the demodulator and the channel decoder. In each iteration, the decoder supplies a log-likelihood ratio (LLR)

$$L(b_i) = \log\left(\frac{P(b_i = 1)}{P(b_i = -1)}\right) \quad (3)$$

for each bit b_i , $i = 1, \dots, N_t K$, of the symbol vector. This LLR is used as a priori information by the demodulator. The demodulator in turn outputs the a posteriori LLR value $L(b_i|\mathbf{y})$, which is employed as input to the decoder. Calculation of this LLR value is the topic of the next section.

III. DEMODULATION

The a posteriori LLR for detection of b_i given \mathbf{y} is

$$L(b_i|\mathbf{y}) = \log\left(\frac{P(b_i = 1|\mathbf{y})}{P(b_i = -1|\mathbf{y})}\right) \quad (4)$$

¹Throughout the paper, we let “bits” take on the values $\{-1, +1\}$ instead of $\{0, 1\}$. This convention simplifies some of the equations in Section III-B.

$$= \log \left(\frac{\sum_{s_1 \in \mathcal{S}} \cdots \sum_{s_j \in \mathcal{S}: b_i=1} \cdots \sum_{s_{N_t} \in \mathcal{S}} \exp \left(-\frac{1}{N_0} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 \right) P(\mathbf{s})}{\sum_{s_1 \in \mathcal{S}} \cdots \sum_{s_j \in \mathcal{S}: b_i=-1} \cdots \sum_{s_{N_t} \in \mathcal{S}} \exp \left(-\frac{1}{N_0} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 \right) P(\mathbf{s})} \right) \quad (5)$$

$$= \log \left(\frac{\sum_{\mathbf{s}: b_i=1} \exp \left(-\frac{1}{N_0} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 \right) P(\mathbf{s})}{\sum_{\mathbf{s}: b_i=-1} \exp \left(-\frac{1}{N_0} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 \right) P(\mathbf{s})} \right), \quad (6)$$

where (5) follows from a standard calculation [8]. In (5), the bit to be detected b_i is carried by symbol s_j . Equation (5) is rewritten as (6) to simplify notation. The probability of each signal vector \mathbf{s}

$$P(\mathbf{s}) = \prod_{k=1}^{N_t K} P(b_k) = \prod_{n=1}^{N_t} P(s_n), \quad (7)$$

is given by assuming independent bits b_k , $k = 1, \dots, N_t K$. The symbols per real dimension s_n , $n = 1, \dots, N_t$ are thus also independent. In fact, a hard decision that b_i equals 1 for $L(b_i|\mathbf{y}) \geq 0$, and that b_i equals -1 otherwise, is a maximum a posteriori (MAP)-optimal decision [10]. Because of this property, we name (6) the MAP detector. However, it is important to observe that no hard decisions are taken by the demodulator unit, its purpose is to supply the decoder with an LLR for each bit b_i .

The computational complexity of (6) is polynomial in the size of the signal constellation M , and exponential in the number of transmit antennas N_t . This is prohibitive in real systems. As an approximation, one may replace (6) with

$$L(b_i|\mathbf{y}) \approx \log \left(\frac{\max_{\mathbf{s}: b_i=1} \left(\exp \left(-\frac{1}{N_0} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 \right) P(\mathbf{s}) \right)}{\max_{\mathbf{s}: b_i=-1} \left(\exp \left(-\frac{1}{N_0} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 \right) P(\mathbf{s}) \right)} \right), \quad (8)$$

which is referred to as the Max-log approximation. Replacing (6) by (8) does not solve the fundamental complexity problem though, since searches for the maximum term in the numerator and in the denominator have to be performed. Several methods that find the maximum terms in (8) have been proposed in the literature. For example, the ZF-DF method is fast, but has poor error probability performance unless \mathbf{H} is very well conditioned [3]. Another possibility is sphere decoding [4], which always finds the maximum, provided that one waits until the algorithm terminates.

A. Review of partial marginalization (PM) in [8]

The MAP problem (6) in the case when the a priori symbol probabilities are uniformly distributed, $P(s_n) = 1/M$, $s_n = 1, \dots, M$, $n = 1, \dots, N_t$, is referred to as the maximum-likelihood (ML) problem

$$L(b_i|\mathbf{y}) = \log \left(\frac{\sum_{\mathbf{s}: b_i=1} \exp \left(-\frac{1}{N_0} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 \right)}{\sum_{\mathbf{s}: b_i=-1} \exp \left(-\frac{1}{N_0} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 \right)} \right). \quad (9)$$

A recent development in MIMO ML demodulation is the soft-output partial marginalization (PM) algorithm [8]. PM combines summation of terms as in (6), and maximization of terms as in (8), via a user-defined parameter. The partial summation, or marginalization, has given the method its name. Maximization is in turn approximated by ZF-DF. The advantages of PM are that it is straightforward to parallelize, that it has a constant and fully predictable runtime, and that it may operate arbitrarily close to the optimal ML solution.

The first step of PM is to let the columns of \mathbf{H} and the elements of \mathbf{s} undergo a permutation in order to reduce the soft detection FER in practice, see Appendix A. The permutation in Appendix A differs from its equivalent in [8], in the way that the column of \mathbf{H} corresponding to b_i after permutation is among the first r columns, and similarly, the symbol per real dimension carrying b_i is among the first r entries of \mathbf{s} . This does not appreciably affect the performance, but it simplifies the description of the PM algorithm. Also, for simplicity of the notation, we continue using \mathbf{H} and \mathbf{s} for the corresponding variables with permutations. We also introduce

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}^A & \mathbf{H}^B \end{bmatrix}, \quad \text{and} \quad \mathbf{s} = \begin{bmatrix} \mathbf{s}^A \\ \mathbf{s}^B \end{bmatrix}, \quad (10)$$

where $\mathbf{H}^A \in \mathbb{R}^{N_r \times r}$, $\mathbf{H}^B \in \mathbb{R}^{N_r \times N_t - r}$, $\mathbf{s}^A \in \mathbb{R}^r$, and $\mathbf{s}^B \in \mathbb{R}^{N_t - r}$. The PM approximation of (6) is given by

$$L(b_i | \mathbf{y}) \approx \log \left(\frac{\sum_{\mathbf{s}^A: b_i=1} \max_{\mathbf{s}^B} \exp \left(-\frac{1}{N_0} \|\mathbf{y} - \mathbf{H}^A \mathbf{s}^A - \mathbf{H}^B \mathbf{s}^B\|^2 \right)}{\sum_{\mathbf{s}^A: b_i=-1} \max_{\mathbf{s}^B} \exp \left(-\frac{1}{N_0} \|\mathbf{y} - \mathbf{H}^A \mathbf{s}^A - \mathbf{H}^B \mathbf{s}^B\|^2 \right)} \right) \quad (11)$$

$$\approx \log \left(\frac{\sum_{\mathbf{s}^A: b_i=1} \exp \left(-\frac{1}{N_0} \|\mathbf{y} - \mathbf{H}^A \mathbf{s}^A - \mathbf{H}^B \hat{\mathbf{s}}^B\|^2 \right)}{\sum_{\mathbf{s}^A: b_i=-1} \exp \left(-\frac{1}{N_0} \|\mathbf{y} - \mathbf{H}^A \mathbf{s}^A - \mathbf{H}^B \hat{\mathbf{s}}^B\|^2 \right)} \right), \quad (12)$$

where $N_t - r$ of the N_t sums, cf. (5), are approximated by maximization over \mathbf{s}^B in (11). In (12), the maximization is approximated by a ZF-DF solution $\hat{\mathbf{s}}^B$. We note that for larger r , the ratio of the number of columns to the number of rows of \mathbf{H}^B becomes smaller, which improves the condition number of $\mathbf{H}^{BT} \mathbf{H}^B$, and thus also the ZF-DF solution, see [8]. In the limit of large r , the ML solution (9) is obtained, and in the limit of small r , the Max-log solution (8) for the case of uniform a priori probabilities, with maximization performed by ZF-DF, is obtained.

B. Partial marginalization with natural mappings of bits to symbols

It was shown in [8] that for BPSK modulation per real dimension, the general MAP problem (6) can be rewritten as an ML problem of the form (9). The reformulation was based on ideas originally

presented in [9]. This strategy can in fact be extended. The MAP problem (6) can be formulated as an ML problem (9) as long as a uniform, separable constellation with a natural mapping of bits to symbols is employed. The key is that the signal vector can be written as a linear function of the bits. Consider a uniform constellation $\mathcal{S} = \{-A, -A + 2\frac{A}{M-1}, -A + 4\frac{A}{M-1}, \dots, A\}$ where A is the maximal signal amplitude. Then we can write

$$\mathbf{s} = \mathbf{W}\mathbf{b}, \quad (13)$$

where

$$\mathbf{b} \triangleq \begin{bmatrix} b_1 \\ \vdots \\ b_{N_t K} \end{bmatrix} \in \mathbb{R}^{N_t K}, \quad (14)$$

$$\mathbf{W} \triangleq \frac{A}{M-1} \begin{bmatrix} 1 & 2 & 4 & \dots & 2^{K-1} & 0 & 0 & 0 & \dots & 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 & 1 & 2 & 4 & \dots & 2^{K-1} & \dots & 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 & \dots & 1 & 2 & 4 & \dots & 2^{K-1} \end{bmatrix} \in \mathbb{R}^{N_t \times N_t K}. \quad (15)$$

For example, for 4-PAM and $A = 3$, the bit sequences $[-1,-1]$, $[1,-1]$, $[-1,1]$, and $[1,1]$ are mapped to the signal constellation points $-3,-1,1$, and 3 respectively.

It is further possible to rewrite

$$\log(P(s_n)) = \sum_{k=(n-1)K+1}^{nK} \frac{1}{2} \left(\log(P(b_k = 1)P(b_k = -1)) + \log\left(\frac{P(b_k = 1)}{P(b_k = -1)}\right) b_k \right), \quad (16)$$

for $n = 1, \dots, N_t$. We introduce

$$\mathbf{\Gamma} \triangleq \text{Diag} \left[\frac{1}{2} \log\left(\frac{P(b_1 = 1)}{P(b_1 = -1)}\right), \dots, \frac{1}{2} \log\left(\frac{P(b_{N_t K} = 1)}{P(b_{N_t K} = -1)}\right) \right] \in \mathbb{R}^{N_t K \times N_t K}, \quad (17)$$

$$\bar{\mathbf{H}} \triangleq \begin{bmatrix} \mathbf{H}\mathbf{W} \\ \frac{N_0}{2}\mathbf{\Gamma} \end{bmatrix} \in \mathbb{R}^{N_r + N_t K \times N_t K}, \quad (18)$$

$$\bar{\mathbf{y}} \triangleq \begin{bmatrix} \mathbf{y} \\ \mathbf{1} \end{bmatrix} \in \mathbb{R}^{N_r + N_t K}, \quad (19)$$

where $\mathbf{1} \in \mathbb{R}^{N_t K}$ is an all-ones vector. By completing the squares with respect to $\frac{N_0}{2}\mathbf{\Gamma}\mathbf{b}$, and rearranging the terms, (6) may be written as

$$L(b_i|\mathbf{y}) = \log \left(\frac{\sum_{\mathbf{b}: b_i=1} \exp\left(-\frac{1}{N_0} \|\bar{\mathbf{y}} - \bar{\mathbf{H}}\mathbf{b}\|^2\right)}{\sum_{\mathbf{b}: b_i=-1} \exp\left(-\frac{1}{N_0} \|\bar{\mathbf{y}} - \bar{\mathbf{H}}\mathbf{b}\|^2\right)} \right). \quad (20)$$

Relation (20) is the MAP estimator (6) on the ML form in (9). The new augmented channel matrix $\bar{\mathbf{H}}$ can be interpreted in terms of having added virtual antennas to the system.

Once the MAP problem is formulated on ML form, the PM method can be used for achieving an approximate solution (12). The linear equation (13) can however not be written for other bit-to-symbol-mappings, and natural mappings are sub-optimal in general [11]. In the following section, we show how to extend the ideas in [8] to the case of a general constellation, and to arbitrary mappings of the channel bits to the constellation.

C. Partial marginalization with soft input for general constellations

We next present the main contribution of this paper, an extension of [8] to soft-input for general (not necessarily uniform) constellations and arbitrary bit-symbol mappings. The idea is to modify the part of the PM algorithm that invokes the ZF-DF scheme. Similarly to in Section III-A, we start from (6), successively introduce approximations, and rearrange expressions so as to obtain a form suitable for fast computations. We first approximate (6) by replacing $N_t - r$ sums with the maximum term

$$L(b_i|\mathbf{y}) \approx \log \left(\frac{\sum_{\mathbf{s}^A: b_i=1} \max_{\mathbf{s}^B} \left(\exp \left(-\frac{1}{N_0} \|\mathbf{y} - \mathbf{H}^A \mathbf{s}^A - \mathbf{H}^B \mathbf{s}^B\|^2 \right) P(\mathbf{s}^B) \right) P(\mathbf{s}^A)}{\sum_{\mathbf{s}^A: b_i=-1} \max_{\mathbf{s}^B} \left(\exp \left(-\frac{1}{N_0} \|\mathbf{y} - \mathbf{H}^A \mathbf{s}^A - \mathbf{H}^B \mathbf{s}^B\|^2 \right) P(\mathbf{s}^B) \right) P(\mathbf{s}^A)} \right). \quad (21)$$

Thereafter, a QR-factorization of the channel $\mathbf{H}^B = \mathbf{Q}\mathbf{R}$ is performed, where $\mathbf{Q} \in \mathbb{R}^{N_r \times N_r}$ is orthogonal, and $\mathbf{R} \in \mathbb{R}^{N_r \times N_t - r}$ is upper triangular. The Euclidean norm is invariant under an orthogonal transformation, and it is possible to rewrite (21) as

$$L(b_i|\mathbf{y}) \approx \log \left(\frac{\sum_{\mathbf{s}^A: b_i=1} \max_{\mathbf{s}^B} \left(\exp \left(-\frac{1}{N_0} \|\mathbf{R}\mathbf{s}^B - \mathbf{Q}^T(\mathbf{y} - \mathbf{H}^A \mathbf{s}^A)\|^2 \right) P(\mathbf{s}^B) \right) P(\mathbf{s}^A)}{\sum_{\mathbf{s}^A: b_i=-1} \max_{\mathbf{s}^B} \left(\exp \left(-\frac{1}{N_0} \|\mathbf{R}\mathbf{s}^B - \mathbf{Q}^T(\mathbf{y} - \mathbf{H}^A \mathbf{s}^A)\|^2 \right) P(\mathbf{s}^B) \right) P(\mathbf{s}^A)} \right). \quad (22)$$

We further introduce

$$\begin{bmatrix} \tilde{\mathbf{y}} \\ \check{\mathbf{y}} \end{bmatrix} \triangleq \mathbf{Q}^T(\mathbf{y} - \mathbf{H}^A \mathbf{s}^A), \quad \mathbf{R} = \begin{bmatrix} \tilde{\mathbf{R}} \\ \mathbf{0} \end{bmatrix}, \quad (23)$$

where $\tilde{\mathbf{y}}$ is a vector with $N_t - r$ elements, $\check{\mathbf{y}}$ is a vector with $N_r - N_t + r$ elements, $\tilde{\mathbf{R}} \in \mathbb{R}^{N_t - r \times N_t - r}$ is upper triangular, and $\mathbf{0} \in \mathbb{R}^{N_r - N_t + r \times N_t - r}$ is an all-zero matrix.

Relation (23) is reformulated as

$$L(b_i|\mathbf{y}) \approx \log \left(\frac{\sum_{\mathbf{s}^A: b_i=1} \max_{\mathbf{s}^B} \left(\exp \left(-\frac{1}{N_0} \|\tilde{\mathbf{R}}\mathbf{s}^B - \tilde{\mathbf{y}}\|^2 - \frac{1}{N_0} \|\check{\mathbf{y}}\|^2 \right) P(\mathbf{s}^B) \right) P(\mathbf{s}^A)}{\sum_{\mathbf{s}^A: b_i=-1} \max_{\mathbf{s}^B} \left(\exp \left(-\frac{1}{N_0} \|\tilde{\mathbf{R}}\mathbf{s}^B - \tilde{\mathbf{y}}\|^2 - \frac{1}{N_0} \|\check{\mathbf{y}}\|^2 \right) P(\mathbf{s}^B) \right) P(\mathbf{s}^A)} \right). \quad (24)$$

We observe that solving

$$\hat{\mathbf{s}}^B = \operatorname{argmax}_{\mathbf{s}^B} \left(\exp \left(-\frac{1}{N_0} \|\tilde{\mathbf{R}}\mathbf{s}^B - \tilde{\mathbf{y}}\|^2 \right) P(\mathbf{s}^B) \right) \quad (25)$$

in (24) is equivalent to solving

$$\hat{\mathbf{s}}^B = \operatorname{argmax}_{\mathbf{s}^B} \left(-\frac{1}{N_0} \|\tilde{\mathbf{R}}\mathbf{s}^B - \tilde{\mathbf{y}}\|^2 + \sum_{n=1}^{N_t-r} \log(P(s_n^B)) \right), \quad (26)$$

where $\mathbf{s}^B = [s_1^B, \dots, s_{N_t-r}^B]^T$. Inspired by the traditional ZF-DF algorithm [3], we suggest to use Alg. 1 for solving (26) approximatively, and with low computational complexity.

Algorithm 1 ZF-DF-type algorithm for solving (26). The output of the algorithm is $\hat{\mathbf{s}}^B$.

1: Set $k := 1$.

2: Solve

$$\hat{s}_{N_t-r}^B = \operatorname{argmax}_{s_{N_t-r}^B \in \mathcal{S}} \left(-\frac{1}{N_0} (\tilde{R}_{N_t-r, N_t-r} s_{N_t-r}^B - \tilde{y}_{N_t-r})^2 + \log(P(s_{N_t-r}^B)) \right). \quad (27)$$

3: Set $k := k + 1$.

4: Solve

$$\begin{aligned} \hat{s}_{N_t-r-k+1}^B = & \operatorname{argmax}_{s_{N_t-r-k+1}^B \in \mathcal{S}} \left(-\frac{1}{N_0} \left(\tilde{R}_{N_t-r-k+1, N_t-r-k+1} s_{N_t-r-k+1}^B \right. \right. \\ & \left. \left. + \sum_{l=2}^k \tilde{R}_{N_t-r-k+1, N_t-r-k+l} \hat{s}_{N_t-r-k+l}^B - \tilde{y}_{N_t-r-k+1} \right)^2 \right. \\ & \left. + \log(P(s_{N_t-r-k+1}^B)) \right). \quad (28) \end{aligned}$$

5: If $k < N_t - r$, continue from step 3, otherwise terminate.

Both (27) and (28) are scalar optimization problems. Differently from the standard ZF-DF, at each step k , a search over all M points in the scalar signal constellation \mathcal{S} is required. This is because a non-linear equation is solved at each step in Alg. 1, while the standard ZF-DF solves a linear equation at each step.

Once Alg. 1 is performed for all possible \mathbf{s}^A , we may approximate (22) as

$$L(b_i | \mathbf{y}) \approx \log \left(\frac{\sum_{\mathbf{s}^A: b_i=1} \exp \left(-\frac{1}{N_0} \|\tilde{\mathbf{R}}\hat{\mathbf{s}}^B - \tilde{\mathbf{y}}\|^2 - \frac{1}{N_0} \|\check{\mathbf{y}}\|^2 \right) P(\mathbf{s}^A) P(\hat{\mathbf{s}}^B)}{\sum_{\mathbf{s}^A: b_i=-1} \exp \left(-\frac{1}{N_0} \|\tilde{\mathbf{R}}\hat{\mathbf{s}}^B - \tilde{\mathbf{y}}\|^2 - \frac{1}{N_0} \|\check{\mathbf{y}}\|^2 \right) P(\mathbf{s}^A) P(\hat{\mathbf{s}}^B)} \right) \quad (29)$$

As the optimization in Alg. 1 is performed, the expressions $\exp \left(-\frac{1}{N_0} \|\tilde{\mathbf{R}}\hat{\mathbf{s}}^B - \tilde{\mathbf{y}}\|^2 \right) P(\hat{\mathbf{s}}^B)$ in (29) are evaluated simultaneously.

The proposed soft-input demodulator is straightforward to generalize to non-separable constellations. In this case, Alg. 1 has to search through a complex-valued constellation instead of a real-valued constellation

at each step.

We briefly discuss the computational complexity of (29). For each \mathbf{H} , a preprocessing consisting of sorting of \mathbf{H} and \mathbf{s} by means of Alg. 2 in Appendix A, must be performed. In Alg. 2, the computational complexity of the matrix inversions dominates. Also, for each \mathbf{H} , a QR decomposition of \mathbf{H} , and a calculation of $\mathbf{Q}^T \mathbf{H}_A \mathbf{s}_A$ for each \mathbf{s}_A , are required. For each \mathbf{y} , calculation of $\mathbf{Q}^T \mathbf{y}$ requires approximately N_r^2 multiplications and additions, and Alg. 1 needs approximately $2^r M(N_t - r)N_t$ comparisons of scalars. In sufficiently slow fading, the cost of preprocessing each \mathbf{H} can be amortized over many bits. The total number of operations C_{bit} needed for the detection of b_i is then on the order of

$$C_{\text{bit}} \approx \frac{2^r M(N_t - r)N_t + 2N_r^2}{N_t}, \quad (30)$$

and the exponential complexity in N_t of (6) is avoided.

IV. EXPERIMENTS

In this section, the performance of the proposed algorithm is quantified, and compared to other approaches.

A. Simulation setting

Experiment parameters are chosen as follows:

- **MIMO system:** A 3×3 MIMO complex system was used. This means that $N_t = N_r = 6$. 16-quadrature amplitude modulation (QAM) signaling, which corresponds to $M = 4$ different possible constellation points per real dimension, was employed. We further used a rate-1/2, regular (3,6) LDPC code with codeword size 2000 bits. A parity check matrix was randomly constructed, but some small-loop removal was applied. The resulting graph had girth 8.
- **Channel:** We are considering Rayleigh slow fading channels, where each element of \mathbf{H} is independent and identically distributed (i.i.d.) zero mean with variance 1/2. More precisely, \mathbf{H} is constant over a codeword.
- **Performance comparisons:** We refer to the set of uncoded bits per codeword as a frame. We estimate the frame-error rate (FER) by means of Monte Carlo integration for varying signal-to-noise ratio (SNR). The SNR is defined as E_b/N_0 , where E_b is the transmitted energy per uncoded bit. For each SNR value, we count 1000 frame errors in the Monte Carlo simulation. Comparisons are made with the optimal brute-force MAP detector (6), and the brute-force Max-log detector (8), which both are non-polynomial (NP)-hard problems to solve.

B. Results

In Fig. 2 we see a comparison between brute-force MAP (6) and PM (29) with different numbers of iterations. For PM, the number r of columns over which the sum is performed is equal to 3. For all SNR values, and after every iteration step, the PM algorithm performs almost as well as the MAP detector.

In Fig. 3, the number r of columns over which the sum is performed in the PM method is varied, and comparisons are made with the MAP and Max-log schemes. The number of demodulator-decoder iterations is 3 for all algorithms. For all signal-to-noise ratios, PM gives better performance than the Max-log method, which, similarly to the MAP method, has an exponentially increasing computational complexity with the number of antennas. The observation that PM gives better performance than the Max-log algorithm is important, since the fixed complexity sphere decoder, which is a major competitor to the PM algorithm, is based on the Max-log algorithm [6], [7].

The choice of r as a function of N_t is discussed in more detail in reference [8], which also contains more extensive simulation results for different numbers of antennas (for 4-QAM). The tradeoffs involved in the choice of r do not depend much on whether an iterative receiver (providing the demodulator with soft input) is used. The C++ program used for generating the results can be downloaded from [12].

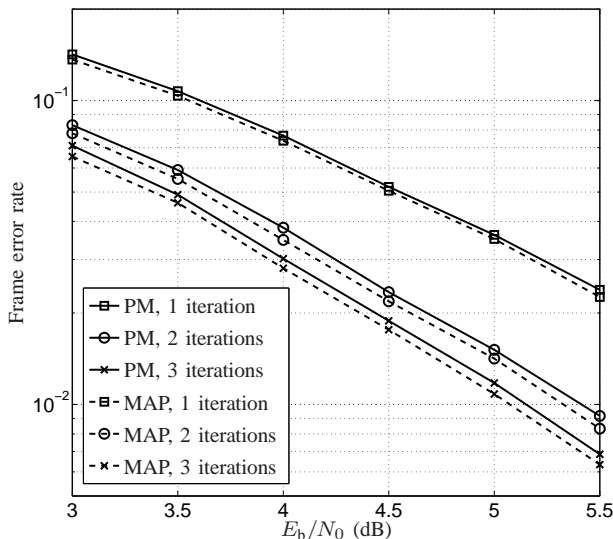


Fig. 2. Performance comparison between PM (29) and brute-force MAP (6) when the number of iterations is varied. We use a 3×3 complex MIMO system, $N_t = N_r = 6$, with 16-QAM modulation $M = 4$, and a rate-1/2 LDPC code where each codeword contains 2000 bits, and spans one channel realization. For PM, the number r of columns over which the sum is performed is equal to 3.

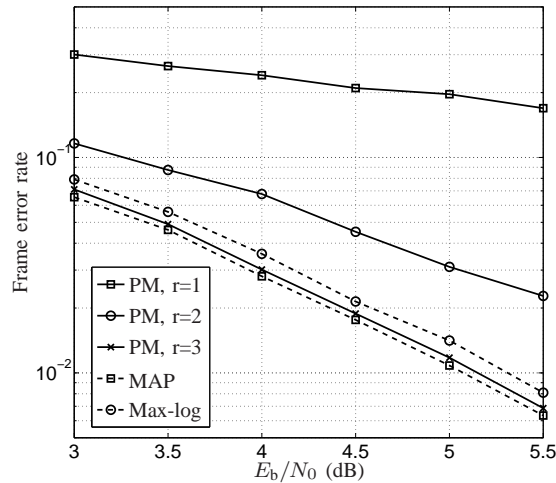


Fig. 3. Performance comparisons for PM (29), with different numbers r of summed columns, brute-force MAP (6), and brute-force Max-log (8). The number of demodulator-decoder iterations is 3 for all algorithms. All other simulation parameters remain the same as in Fig. 2.

V. CONCLUSION

A previously proposed method for MIMO detection with soft-input and output, PM [8], has fixed computational complexity, is straightforward to parallelize, and may operate arbitrarily close to the MAP detector performance-wise, when using more computational power. However, when using soft-input, this scheme has until now only worked with BPSK modulation per real dimension.

This paper offers a new soft-input extension, for arbitrary separable constellations, to the PM method. The computational complexity is not significantly increased. In a range of presented experiments, the new method gives the nearly same FER performance as MAP detection. Moreover, the PM algorithm performs better than the Max-log algorithm. This last experimental comparison is important, since the fixed complexity sphere decoder [6], [7] is based on the Max-log algorithm. The proposed ZF-DF algorithm is straightforward to generalize to non-separable constellations. In this case, the ZF-DF algorithm has to search through a complex-valued constellation instead of a real-valued constellation at each step.

APPENDIX A

COLUMN REORDERING

The purpose of Alg. 2 is to reorder the bits so that the “most difficult” bits are dealt with by the marginalization over s^A in (12). In effect, this minimizes the error propagation in Alg. 1. For a more thorough discussion of Alg. 2, see [8].

Algorithm 2 Reordering of columns of \mathbf{H}

-
- 1: Let $\mathcal{I} = \emptyset$ (empty), and let $\mathcal{I}^c = [1, \dots, N_t]$.
 - 2: Compute $\lambda = \text{diag}((\mathbf{H}^T \mathbf{H})^{-1})$. Let k be the index of the largest element of λ .
 - 3: Set $\mathcal{I} := [\mathcal{I} \ \mathcal{I}_k^c]$.
 - 4: Remove the k th column of \mathbf{H} .
 - 5: If \mathbf{H} is empty, go to step 6, otherwise repeat from step 2.
 - 6: Assume that the bit b_i that is to be detected is carried by the symbol j before permutation. If j is element \mathcal{I}_l of \mathcal{I} , where $l > r$, redefine $\mathcal{I} := [\mathcal{I}_1, \dots, \mathcal{I}_{r-1}, j, \mathcal{I}_r, \mathcal{I}_{r+1}, \dots, \mathcal{I}_{l-1}, \mathcal{I}_{l+1}, \dots, \mathcal{I}_{N_t}]$.
 - 7: Redefine \mathbf{H} with its original element values that it had in step 1.
 - 8: Rearrange the columns of \mathbf{H} and the scalar entries of \mathbf{s} according to \mathcal{I} . For performing the summation in (29), the position of the symbol j that carries the bit i should also be stored.
-

REFERENCES

- [1] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge, U.K.: Cambridge Univ. Press, 2005.
- [2] B. Hochwald and S. ten Brink, "Achieving near-capacity on a multiple-antenna channel," *IEEE Trans. Commun.*, vol. 51, no. 3, pp. 389–399, Mar. 2003.
- [3] E. G. Larsson, "MIMO detection methods: How they work," *IEEE Signal Processing Mag.*, vol. 26, no. 3, pp. 91–95, May 2009.
- [4] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," *IEEE Trans. Inform. Theory*, vol. 48, no. 8, pp. 2201–2214, Aug. 2002.
- [5] J. Jalden and B. Ottersten, "On the complexity of sphere decoding in digital communications," *IEEE Trans. Signal Processing*, vol. 53, no. 4, pp. 1474–1484, Apr. 2005.
- [6] L. Barbero and J. Thompson, "Fixing the complexity of the sphere decoder for MIMO detection," *IEEE Trans. Wireless Commun.*, vol. 7, no. 6, pp. 2131–2142, June 2008.
- [7] —, "Extending a fixed-complexity sphere decoder to obtain likelihood information for turbo-MIMO systems," *IEEE Trans. Veh. Technol.*, vol. 57, no. 5, pp. 2804–2814, Sept. 2008.
- [8] E. G. Larsson and J. Jalden, "Fixed-complexity soft MIMO detection via partial marginalization," *IEEE Trans. Signal Processing*, vol. 56, no. 8, pp. 3397–3407, Aug. 2008.
- [9] B. Steingrimsson, Z.-Q. Luo, and K. M. Wong, "Soft quasi-maximum-likelihood detection for multiple-antenna wireless channels," *IEEE Transactions on Signal Processing*, vol. 51, no. 11, pp. 2710–2719, Nov 2003.
- [10] H. L. Van Trees, *Detection, Estimation, and Modulation Theory, Part I*. Wiley-Interscience, Sept. 2001.
- [11] H. Futaki and T. Ohtsuki, "Low-density parity-check (LDPC) coded OFDM systems with M-PSK," in *Vehicular Technology Conf.*, May 2002, pp. 1035–1039, vol.2.
- [12] Code repository, Division of Communication Systems, Linköping University. [Online]. Available: <http://www.commsys.isy.liu.se/en/publications>