

NEAR-OPTIMAL SOFT-OUTPUT FIXED-COMPLEXITY MIMO DETECTION VIA SUBSPACE MARGINALIZATION AND INTERFERENCE SUPPRESSION

Mirsad Čirkić and Erik G. Larsson

Communication Systems Division, Department of Electrical Engineering (ISY), Linköping University, SE-581 83 Linköping, Sweden. {mirsad,erik.larsson}@isy.liu.se

ABSTRACT

The fundamental problem of our interest here is soft MIMO detection. We propose a method that yields excellent performance, at low and at fixed (deterministic) complexity. Our method provides a well-defined tradeoff between computational complexity and performance. Apart from an initial step consisting of selecting columns, the algorithm involves no searching nor algorithmic branching; hence the algorithm has a completely predictable run-time, and it is readily and massively parallelizable.

Index Terms— soft MIMO detection, MMSE, interference suppression, log-likelihood ratio, fixed complexity

1. INTRODUCTION

We consider multiple-input multiple-output (MIMO) systems, which are known to substantially increase both the spectral efficiency in rich scattering environments [1] and the link robustness. A major difficulty in the implementation of MIMO systems is the signal separation (detection) problem, which is generally computationally expensive. This problem can be especially pronounced in large MIMO systems [2]. The main reason for why MIMO detection is difficult is the occurrence of ill-conditioned MIMO channels. For instance, the complexity of the optimal detector, which computes the log-likelihood ratio (LLR) values exactly and therefore solves the MIMO detection problem optimally, grows exponentially with the number of transmit antennas and polynomially with the size of the signal constellation. Suboptimal and fast methods, such as zero-forcing perform well only for well-conditioned channels.

Many different methods have been proposed that aim to perform close to the optimal detector with reduced computational complexity [3]–[7]. Some of today’s state-of-the-art detectors provide the possibility of trading complexity for performance via the choice of some user parameter. One important advantage of such detectors is that the tradeoff parameter can be adapted to the effective channel conditions in order to improve the overall performance. Amongst these detectors, there are two main subcategories. The first consists of detectors that do not have fixed complexity and perform a reduced tree-search, such as the sphere-decoding (SD) aided max-log method and its relatives [3]–[5]. One of the more recent ones is the reduced dimension maximum-likelihood search (RD-MLS) of [5]. Unfortunately, the methods in this category have an exponential worst-case complexity unless a suboptimal termination criterion is used. The

This work was supported in part by the Swedish Research Council (VR), the Swedish Foundation of Strategic Research (SSF), and the Excellence Center at Linköping-Lund in Information Technology (ELLIIT). E. G. Larsson is a Royal Swedish Academy of Sciences (KVA) Research Fellow supported by a grant from the Knut and Alice Wallenberg Foundation.

other subcategory of detectors are the ones that have fixed complexity. These are much more desirable from an implementation point of view in order to avoid over-dimensioned hardware. Examples of such detectors are the soft-output via partial marginalization (PM) method [6] and the fixed-complexity SD (FCSD) [7] aided max-log method. These fixed-complexity detectors provide a simple and well-defined tradeoff between computational complexity and performance, they have a fixed and fully predictable run time, and they are highly parallelizable. Note that the FCSD is equivalent to the PM method with an additional max-log approximation.

We propose a new method that is inspired by the ideas in [5]–[7] of partitioning the original problem into smaller problems. As in the PM method, we perform marginalization over a few of the bits when computing the LLR values. The approximate LLRs that enter the marginalization are simpler than those in PM, and this substantially reduces the complexity of our algorithm which will be clear in Sec. 3. In addition to that, we suppress the interference on the considered subspace by performing soft interference suppression (SIS). The SIS procedure, which is one of the constituents of our algorithm is inspired by the work in [8]–[10]. The main difference between the SIS procedure in our work and that in [8]–[10] is that we allow for the signal subspace (and the interfering subspace) to have varying dimensionality. The additional differences are: (i) we perform the SIS in a MIMO setting internally without the need of a priori information from the decoder as opposed to [8] and (ii) we do not iterate the internal LLR values nor do we ignore the correlation between the interfering terms over the different receive antennas as in [9], [10].

Summary of contribution: We propose a novel MIMO detection method that runs at fixed complexity, provides a clear and well-defined tradeoff between computational complexity and performance, and is highly parallelizable. The ideas behind it are fundamentally simple and allow for very simple algorithmic implementations. We refer to the new method as subspace marginalization with interference suppression (SUMIS).

2. PRELIMINARIES

We consider the real-valued MIMO-channel model

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

where $\mathbf{H} \in \mathbb{R}^{N_r \times N_t}$ is the MIMO channel matrix, $\mathbf{s} \in \mathcal{S}^{N_t}$ is the transmitted vector. We assume that $\mathcal{S} = \{-1, +1\}$ (BPSK modulation per real dimension), hence referring to a “symbol” is equivalent to referring to a “bit”. With some extra expense of notation, it is straightforward to extend all results that we present to higher order constellations. Further, $\mathbf{e} \in \mathbb{R}^{N_r} \sim \mathcal{N}(\mathbf{0}, \frac{N_0}{2} \mathbf{I})$ denotes the noise

vector and $\mathbf{y} \in \mathbb{R}^{N_R}$ is the received vector. The channel is perfectly known to the receiver and in what follows, we assume that $N_R \geq N_T$ since this is typical in practice and simplifies the mathematics performed in this paper. With separable complex symbol constellations, every complex-valued model of type (1) can be posed as a real-valued model of the same type, see [6].

2.1. Optimal Soft MIMO Detection

The optimal soft information desired by the channel decoder is the a posteriori log-likelihood ratio

$$l(s_i|\mathbf{y}) \triangleq \log \left(\frac{p(s_i = +1|\mathbf{y})}{p(s_i = -1|\mathbf{y})} \right), \quad (2)$$

where s_i is the i :th bit of the transmitted vector \mathbf{s} . The quantity in (2) tells us how likely it is that the i :th bit of \mathbf{s} is equal to minus or plus one, respectively. By using Bayes' rule, performing marginalization over all bits except the i :th bit, and assuming uniform a priori probabilities, the log-likelihood ratio (LLR) becomes

$$l(s_i|\mathbf{y}) = \log \left(\frac{\sum_{\mathbf{s}: s_i(\mathbf{s})=+1} \exp \left(-\frac{1}{N_0} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 \right)}{\sum_{\mathbf{s}: s_i(\mathbf{s})=-1} \exp \left(-\frac{1}{N_0} \|\mathbf{y} - \mathbf{H}\mathbf{s}\|^2 \right)} \right). \quad (3)$$

where the notation $\sum_{\mathbf{s}: s_i(\mathbf{s})=x}$ means the sum over all possible vectors $\mathbf{s} \in \mathcal{S}^{N_T}$ for which the i :th bit is equal to x . In (3), there are 2^{N_T} terms that need to be evaluated and added. This exponential complexity is the main problem in MIMO detection that needs to be addressed. Thus, many approximate methods have been proposed.

In order to explain our method and the competing state-of-the-art methods, for fixed $n_s \in \{1, \dots, N_T\}$, we define the following partitioning of the model in (1)

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{e} = \underbrace{\begin{bmatrix} \bar{\mathbf{H}} & \tilde{\mathbf{H}} \end{bmatrix}}_{\text{col. perm. of } \mathbf{H}} \underbrace{\begin{bmatrix} \bar{\mathbf{s}}^T & \tilde{\mathbf{s}}^T \end{bmatrix}^T}_{\text{perm. of } \mathbf{s}} + \mathbf{e} = \bar{\mathbf{H}}\bar{\mathbf{s}} + \tilde{\mathbf{H}}\tilde{\mathbf{s}} + \mathbf{e}, \quad (4)$$

where $\bar{\mathbf{H}} \in \mathbb{R}^{N_R \times n_s}$, $\tilde{\mathbf{H}} \in \mathbb{R}^{N_R \times (N_T - n_s)}$, $\bar{\mathbf{s}} \in \mathcal{S}^{n_s}$ contains the i :th bit s_i in the original vector \mathbf{s} , and $\tilde{\mathbf{s}} \in \mathcal{S}^{N_T - n_s}$. The choice of partitioning involves the choice of a permutation, and how to make this choice (for $n_s > 1$) is not obvious. In fact, for each bit in \mathbf{s} , there are $\binom{N_T - 1}{n_s - 1}$ possible partitionings in (4). How we perform this partitioning is explained in Sec. 3. Note that for different detectors, the choice of partitioning serves different purposes.

2.2. Today's State-of-the-Art MIMO Detectors

The PM Method in [6]: PM offers a tradeoff between exact and approximate computation of (3), via a parameter $r = n_s - 1 \in \{0, \dots, N_T - 1\}$. We present the slightly modified version in [11] of the method in [6], which is simpler than that in [6] but without comprising performance. The PM method implements a two-step approximation of (3). More specifically, in the first step it approximates the sums of (3) that correspond to $\tilde{\mathbf{s}}$ with a maximization,

$$l(s_i|\mathbf{y}) \approx \log \left(\frac{\sum_{\bar{\mathbf{s}}: s_i(\bar{\mathbf{s}})=+1} \max_{\tilde{\mathbf{s}}} \exp \left(-\frac{1}{N_0} \|\mathbf{y} - \bar{\mathbf{H}}\bar{\mathbf{s}} - \tilde{\mathbf{H}}\tilde{\mathbf{s}}\|^2 \right)}{\sum_{\bar{\mathbf{s}}: s_i(\bar{\mathbf{s}})=-1} \max_{\tilde{\mathbf{s}}} \exp \left(-\frac{1}{N_0} \|\mathbf{y} - \bar{\mathbf{H}}\bar{\mathbf{s}} - \tilde{\mathbf{H}}\tilde{\mathbf{s}}\|^2 \right)} \right). \quad (5)$$

In the second step, the maximization in (5) is approximated with a linear filter with quantization (clipping), such as the zero-forcing

with decision-feedback (ZF-DF) detector [6]. The ZF-DF method is computationally much more efficient than exact maximization, but it performs well only for well-conditioned matrices. However, the max problems in (5) are generally well-conditioned since the matrices $\tilde{\mathbf{H}}$ are tall. For PM, when forming the partitioning in (4), the original bit-order in $\mathbf{s} = [s_1, \dots, s_{N_T}]^T$ is permuted in (5) in a way such that the condition number of $\tilde{\mathbf{H}}$ is minimized, see [6]. Notably, PM performs ZF-DF aided max-log detection for $r = 0$ and computes the exact LLR values (as defined by (3)) for $r = N_T - 1$.

The FCSD Method in [7]: FCSD essentially performs the same procedure as the PM method except that it introduces an additional approximation by employing the max-log approximation on the remaining sums (sums over $\{\tilde{\mathbf{s}} \in \mathcal{S}^{n_s} : s_i(\tilde{\mathbf{s}}) = x\}$) in (5) in the PM method. Hence, instead of performing summations over $\{\tilde{\mathbf{s}} \in \mathcal{S}^{n_s} : s_i(\tilde{\mathbf{s}}) = x\}$ for each x as in PM, it picks the best candidate from $\{\tilde{\mathbf{s}} \in \mathcal{S}^{n_s} : s_i(\tilde{\mathbf{s}}) = x\}$ for each x .

The RD-MLS Method in [5]: RD-MLS performs further the same procedure as FCSD except that it does not perform clipping after the linear filtering and uses an SD type of algorithm to perform a reduced tree-search over $\{\tilde{\mathbf{s}} \in \mathcal{S}^{n_s} : s_i(\tilde{\mathbf{s}}) = x\}$ for each x . Although this method reduces the number of layers in the tree, it does not necessarily improve the conditioning of the reduced problem, as the PM and FCSD methods do. This is due to the unquantized linear filtering operation that essentially results in performing a projection of the original space \mathbf{s} onto the orthogonal complement of the column space of $\tilde{\mathbf{H}}$. Therefore, for an ill-conditioned matrix $\tilde{\mathbf{H}}$, it is unclear if the RD-MLS algorithm would visit significantly fewer branches in the reduced space $\tilde{\mathbf{s}}$ than in the original space \mathbf{s} .

3. PROPOSED SOFT MIMO DETECTOR

In our proposed method, which we refer to as the subspace marginalization with interference suppression (SUMIS) method, there are two main stages. In stage I, a first approximation to the LLR for each bit is computed. In stage II, these approximate LLRs are used in an interference suppression mechanism, whereafter the LLR values are calculated based on the resulting "purified" model.

Stage I: We start with the partitioned model in (4)

$$\mathbf{y} = \bar{\mathbf{H}}\bar{\mathbf{s}} + \underbrace{\tilde{\mathbf{H}}\tilde{\mathbf{s}} + \mathbf{e}}_{\text{interference+noise}} \quad (6)$$

and approximate it via $\mathbf{y} \approx \bar{\mathbf{y}} \triangleq \bar{\mathbf{H}}\bar{\mathbf{s}} + \mathbf{n}$ where \mathbf{n} is a Gaussian stochastic vector $\mathcal{N}(\mathbf{0}, \mathbf{Q})$ with $\mathbf{Q} \triangleq \tilde{\mathbf{H}}\tilde{\mathbf{H}}^T + \frac{N_0}{2}\mathbf{I}$. Subsequently, we compute the a posteriori probability $P(\bar{\mathbf{s}}|\bar{\mathbf{y}})$, which with uniform a priori probabilities per bit is proportional to the likelihood function

$$p(\bar{\mathbf{y}}|\bar{\mathbf{s}}) \propto \exp \left(-\frac{1}{2} (\bar{\mathbf{y}} - \bar{\mathbf{H}}\bar{\mathbf{s}})^T \mathbf{Q}^{-1} (\bar{\mathbf{y}} - \bar{\mathbf{H}}\bar{\mathbf{s}}) \right).$$

The computation of this quantity can be performed computationally more efficiently by using the equivalent model

$$(\bar{\mathbf{H}}^T \mathbf{Q}^{-1} \bar{\mathbf{H}})^{-1} \bar{\mathbf{H}}^T \mathbf{Q}^{-1} \bar{\mathbf{y}} = \bar{\mathbf{s}} + \bar{\mathbf{n}}, \quad \bar{\mathbf{n}} \sim \mathcal{N}(\mathbf{0}, (\bar{\mathbf{H}}^T \mathbf{Q}^{-1} \bar{\mathbf{H}})^{-1}). \quad (7)$$

Furthermore, for better numerical stability and faster computation, we use the matrix inversion lemma when we compute the inverse \mathbf{Q}^{-1} .

Next, since bit s_k in the original vector \mathbf{s} in (6) is contained in $\bar{\mathbf{s}}$, the a posteriori probability $P(s_k|\mathbf{y})$ can be approximated with $P(s_k|\bar{\mathbf{y}})$, which is calculated by marginalizing out the remaining bits in $P(\bar{\mathbf{s}}|\bar{\mathbf{y}})$. Due to the assumption on \mathcal{S} being BPSK, we can perform this marginalization in the LLR domain as

$$\lambda_k \triangleq \log \left(\frac{\sum_{\bar{\mathbf{s}}:s_k=+1} \exp \left(-\frac{1}{2}(\mathbf{y} - \bar{\mathbf{H}}\bar{\mathbf{s}})^T \mathbf{Q}^{-1}(\mathbf{y} - \bar{\mathbf{H}}\bar{\mathbf{s}}) \right)}{\sum_{\bar{\mathbf{s}}:s_k=-1} \exp \left(-\frac{1}{2}(\mathbf{y} - \bar{\mathbf{H}}\bar{\mathbf{s}})^T \mathbf{Q}^{-1}(\mathbf{y} - \bar{\mathbf{H}}\bar{\mathbf{s}}) \right)} \right), \quad (8)$$

which can be efficiently computed using the Jacobian logarithm. The a posteriori probabilities of the remaining elements in \mathbf{s} are approximated analogously to (6)-(8) by simply choosing different partitionings (permutations) of \mathbf{H} and \mathbf{s} such that the bit of interest is in $\bar{\mathbf{s}}$. Using the probability approximations (in the LLR domain) in (8), we compute the conditional expected value of bit s_k under the assumption that all the bits in \mathbf{s} conditioned on \mathbf{y} are independent,

$$\mathbb{E}\{s_k|\mathbf{y}\} \triangleq \sum_{s \in \mathcal{S}} sP(s_k=s|\mathbf{y}) \approx \frac{-1}{1+e^{\lambda_k}} + \frac{1}{1+e^{-\lambda_k}} = \tanh\left(\frac{\lambda_k}{2}\right). \quad (9)$$

This stage is performed for all bits s_k in \mathbf{s} , i.e., $k = 1, \dots, N_T$.

Stage II: For each bit s_i , the interfering vector $\tilde{\mathbf{s}}$ in (6) is suppressed using

$$\mathbf{y}' \triangleq \mathbf{y} - \tilde{\mathbf{H}}\mathbb{E}\{\tilde{\mathbf{s}}|\mathbf{y}\} = \bar{\mathbf{H}}\bar{\mathbf{s}} + \underbrace{\tilde{\mathbf{H}}(\tilde{\mathbf{s}} - \mathbb{E}\{\tilde{\mathbf{s}}|\mathbf{y}\})}_{\text{interference+noise}} + \mathbf{e} \approx \bar{\mathbf{H}}\bar{\mathbf{s}} + \mathbf{n}'. \quad (10)$$

where $\mathbf{n}' \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}')$ with $\mathbf{Q}' \triangleq \tilde{\mathbf{H}}\mathbf{D}\tilde{\mathbf{H}}^T + \frac{N_0}{2}\mathbf{I}$ and \mathbf{D} being the conditional covariance matrix of $\tilde{\mathbf{s}}$. Since $\mathcal{S} = \{-1, +1\}$ and the elements in $\tilde{\mathbf{s}}$ are assumed to be independent conditioned on \mathbf{y} , we get

$$\mathbf{D} = \mathbb{E}\{\text{diag}(\tilde{\mathbf{s}})^2|\mathbf{y}\} - \mathbb{E}\{\text{diag}(\tilde{\mathbf{s}})|\mathbf{y}\}^2 = \mathbf{I} - \text{diag}(\mathbb{E}\{\tilde{\mathbf{s}}|\mathbf{y}\})^2$$

where the operator $\text{diag}(\cdot)$ takes a vector of elements as input and returns a diagonal matrix with these elements on its diagonal. After the interfering vector $\tilde{\mathbf{s}}$ is suppressed, we compute the LLRs. The LLRs are computed by performing a full-blown marginalization in (3) over the corresponding subspace $\bar{\mathbf{s}}$ in the approximated model in (10). Hence, the LLR value we compute for the i :th bit is

$$l(s_i|\mathbf{y}) \approx \log \left(\frac{\sum_{\bar{\mathbf{s}}:s_i(\bar{\mathbf{s}})=+1} \exp \left(-\frac{1}{2}(\mathbf{y}' - \bar{\mathbf{H}}\bar{\mathbf{s}})^T \mathbf{Q}'^{-1}(\mathbf{y}' - \bar{\mathbf{H}}\bar{\mathbf{s}}) \right)}{\sum_{\bar{\mathbf{s}}:s_i(\bar{\mathbf{s}})=-1} \exp \left(-\frac{1}{2}(\mathbf{y}' - \bar{\mathbf{H}}\bar{\mathbf{s}})^T \mathbf{Q}'^{-1}(\mathbf{y}' - \bar{\mathbf{H}}\bar{\mathbf{s}}) \right)} \right). \quad (11)$$

The computation of (11) can be rewritten for improved numerical efficiency, similarly to the LLR computation in the first stage.

Choosing the Permutations: The optimal permutation would be the one that minimizes the bit-error-rate after decoding and this permutation is hard to find. There are many ways to choose the permutation via heuristic arguments. We aim to choose the partitioning, for a bit s_k in \mathbf{s} , that suppresses the interfering vector $\tilde{\mathbf{s}}$ in (6) as much as possible. This essentially means that the columns in $\tilde{\mathbf{H}}$ should be as orthogonal as possible to the columns in $\bar{\mathbf{H}}$. Therefore, we base our partitioning on $\mathbf{H}^T\mathbf{H}$, which can be thought of as a covariance matrix

$$\mathbf{H}^T\mathbf{H} = \begin{bmatrix} \sigma_1^2 & \rho_{1,2}\sigma_1\sigma_2 & \dots \\ \rho_{1,2}\sigma_1\sigma_2 & \sigma_2^2 & \\ \vdots & & \ddots \end{bmatrix},$$

and pick for a column or row k in $\mathbf{H}^T\mathbf{H}$ (bit k in \mathbf{s}) the $n_s - 1$ indexes that correspond to the strongest correlation coefficients $\rho_{k,\ell}$. Then, these indexes along with the index k specify the columns from \mathbf{H} that are placed in $\tilde{\mathbf{H}}$. The rest of the columns are placed in $\bar{\mathbf{H}}$.

Computational Complexity: By omitting the pre-processing procedure and assuming that $N_R \approx N_T \gg n_s$, we give a rough complexity count for finding all the bits in one vector \mathbf{y} . For a vector \mathbf{y} , we must compute the inverse \mathbf{Q}'^{-1} for each partitioning. This can be done with $\mathcal{O}(N_T^3)$ operations for all partitionings by using the Woodbury matrix identity. In each marginalization step, we have 2^{n_s} terms that compute matrix-vector multiplications and one matrix inverse of dimension n_s , hence requiring together $\mathcal{O}(n_s^2 2^{n_s})$ operations. Therefore, for $N_T \gg n_s$, our algorithm requires roughly $\mathcal{O}(N_T^3)$ operations for all the bits, which is much lower than the corresponding complexity of the PM method, $\mathcal{O}(N_T^3 2^{n_s})$.

Summary: The steps of the SUMIS algorithm are summarized in Alg. 1 with generic pseudo-code. Note that due to the fact that most of the N_T permutations of \mathbf{H} and \mathbf{s} will overlap, there is room to optimize the operation of the algorithm much more. Via the adjustable subspace dimensionality, i.e., the n_s -parameter, our method provides a simple and well-defined tradeoff between computational complexity and detector performance. For $n_s = N_T$, there is no interfering vector $\tilde{\mathbf{s}}$ and SUMIS performs exact LLR computation. For $n_s = 1$, SUMIS performs the soft MMSE method with the additional step of suppressing the interfering vector $\tilde{\mathbf{s}}$.

Algorithm 1 Subspace Marginalization with Interference Suppression (SUMIS)

```

Start with some  $\mathbf{H}$ ,  $\mathbf{y}$  and  $n_s \in \{1, \dots, N_T\}$ 
for  $k = 1, \dots, N_T$  do // - First stage - //
    Decide upon a partitioning in (4) based on  $\mathbf{H}^T\mathbf{H}$ 
    Calculate  $\lambda_k$  in (8) (cond. probability of  $s_k$  in terms of LLR)
    Calculate  $\mathbb{E}\{s_k|\mathbf{y}\}$  and  $\text{Var}\{s_k|\mathbf{y}\} = 1 - \mathbb{E}\{s_k|\mathbf{y}\}^2$  in (9)
end for
for each bit in  $\mathbf{s}$  do // - Second stage - //
    Suppress the interfering vector  $\tilde{\mathbf{s}}$  and calculate  $\mathbf{y}'$  in (10)
    Calculate the new covariance matrix  $\mathbf{Q}'$ 
    Calculate the LLR in (11)
end for

```

4. NUMERICAL RESULTS

4.1. Simulation Setup

Using Monte Carlo simulation technique we plot the performance of our new method in terms of frame-error rate (FER) with respect to E_b/N_0 where E_b is the energy per information bit. We use quadrature phase-shift keying (QPSK) modulation with a 4×4 and a 6×6 complex MIMO system, which means that the detection is performed on a real-valued 8×8 and 12×12 MIMO system with binary phase-shift keying (BPSK) modulation, respectively. The channel is chosen to be Rayleigh fading. We consider two different coherence times: slow fading (each codeword sees one channel realization) and fast fading (each codeword spans over 40 channel matrices), respectively. We consider two different channel codes: one bit-interleaved convolution (BIC) code with rate 1/3, and one low-density parity-check (LDPC) code with rate 1/2. Each codeword consists of 10000 bits. For comparison, we also plot the curves of the optimal detector

and of the PM method in [6]. Since the FCSD method is an approximation of the PM method, we refrain from plotting its performance curves. We also ignore plotting the curves of RD-MLS due to the fact the its complexity is not predictable and a fair comparison is difficult to make.

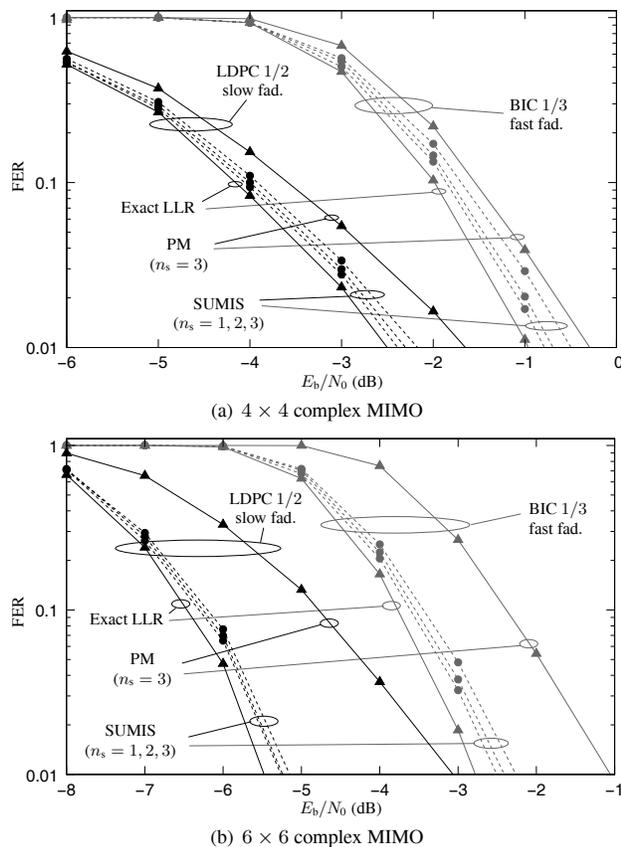


Fig. 1. Performance comparison. The figures show the frame error rate (FER) as a function of E_b/N_0 . We have one figure for 4×4 and one for 6×6 complex MIMO with QPSK. In each figure, there are two groups of plots: one for slow fading (with 1/2-rate LDPC) and one for fast fading (with 1/3-rate BIC). The shown performance curves are: (i) dashed curves for the SUMIS method with $n_s = 1, 2, 3$ spanning from right to left and (ii) solid curves for the optimal method and the PM method with $n_s = r + 1 = 3$.

4.2. Results

We can observe in Fig. 1 that the SUMIS detector performs close to the optimal soft detector. It outperforms the PM method, and it does so at a much lower complexity. Note that the complexity of SUMIS with $n_s = 3$ is much lower than that of PM with $n_s = r + 1 = 3$ even though the partitioned problem in (4) is of the same size. The reason is that the PM method performs the ZF-DF procedure for each summation term whereas the SUMIS method does not.

5. CONCLUSIONS

We have proposed a novel MIMO detection method that outperforms today's state-of-the-art detectors, runs at fixed-complexity, provides a clear and well-defined tradeoff between computational complexity and performance, and is highly parallelizable. The ideas behind it are fundamentally simple and allow for very simple algorithmic implementations. The proposed method has a complexity that is of the same order of magnitude as the linear methods. It opens the door for a whole new class of detectors that can be utilized in the future. Several extensions, which did not fit within the scope of this paper, can be made. One example is iterative decoding.

6. REFERENCES

- [1] E. Telatar, "Capacity of multi-antenna Gaussian channels," *European Transactions on Telecommunications*, vol. 10, no. 6, pp. 585–596, Jun. 2001.
- [2] F. Rusek, D. Persson, B. K. Lau, E. G. Larsson, T. L. Marzetta, O. Edfors, and F. Tufvesson, "Scaling up MIMO: Opportunities and challenges with very large arrays," *IEEE Signal Processing Magazine*, (accepted) 2011.
- [3] E. Viterbo and J. Boutros, "A universal lattice code decoder for fading channels," *IEEE Transactions on Information Theory*, vol. 45, no. 5, pp. 1639–1642, Jul. 1999.
- [4] Z. Guo and P. Nilsson, "Algorithm and implementation of the K-best sphere decoding for MIMO detection," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 3, pp. 491–503, Mar. 2006.
- [5] J. W. Choi, B. Shim, A.C. Singer, and N. I. Cho, "Low-complexity decoding via reduced dimension maximum-likelihood search," *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1780–1793, Mar. 2010.
- [6] E. G. Larsson and J. Jaldén, "Fixed-complexity soft MIMO detection via partial marginalization," *IEEE Transactions on Signal Processing*, vol. 56, no. 8, pp. 3397–3407, Aug. 2008.
- [7] L.G. Barbero and J.S. Thompson, "Extending a fixed-complexity sphere decoder to obtain likelihood information for turbo-MIMO systems," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 5, pp. 2804–2814, Sept. 2008.
- [8] X. Wang and H. V. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Transactions on Communications*, vol. 47, no. 7, pp. 1046–1061, Jul. 1999.
- [9] P. Schniter, "Low-complexity equalization of OFDM in doubly selective channels," *IEEE Transactions on Signal Processing*, vol. 52, no. 4, pp. 1002–1011, Apr. 2004.
- [10] W.-J. Choi, K.-W. Cheong, and J. M. Cioffi, "Iterative soft interference cancellation for multiple antenna systems," in *IEEE Wireless Communications and Networking Conference (WCNC)*, Sept. 2000, vol. 1, pp. 304–309.
- [11] D. Persson and E. G. Larsson, "Partial marginalization soft MIMO detection with higher order constellations," *IEEE Transactions on Signal Processing*, vol. 59, no. 1, pp. 453–458, Jan. 2011.