in Proc. IEEE ICC-2005, Seoul (Korea), May 2005, vol. 4, pp. 2447–2451

Low-Complexity Soft Demodulation of MIMO-BICM Using the Line-Search Detector

Dominik Seethaler, Gerald Matz, and Franz Hlawatsch*

Institute of Communications and Radio-Frequency Engineering, Vienna University of Technology
Gusshausstrasse 25/389, A-1040 Vienna, Austria
phone: +43 1 58801 38958, fax: +43 1 58801 38999, email: dominik.seethaler@tuwien.ac.at

*Currently on leave with Laboratoire des Signaux et Systèmes, Ecole Supérieure d'Electricité
3 Rue Joliot-Curie, F-91190 Gif-sur-Yvette, France

***Abstract*— *Bit-interleaved coded modulation* (BICM) is an attractive transmission scheme for MIMO wireless communications over fast fading channels. BICM receivers employing maximum-likelihood decoding require a soft demodulator (demapper) that calculates log-likelihood ratios (LLRs) for the coded bits. Because in the MIMO case the calculation of LLRs tends to be excessively complex, there is a strong demand for efficient soft demodulation algorithms that calculate approximate LLRs.**

Here, we develop the novel *soft line-search demodulator* (SLSD) by extending the recently introduced line-search detector (LSD) such that approximate LLRs are obtained with little extra computations. We show that the SLSD's BER performance is close to that of the list extension of the sphere decoding algorithm even though the complexity is significantly smaller.

I. Introduction

Bit-interleaved coded modulation (BICM) [1] is an attractive transmission scheme for multiple-input multiple-output (MIMO) wireless communications [2–4]. MIMO-BICM has been shown to outperform space-time trellis coding in fast fading environments [3]. A fast fading model is often adopted for MIMO systems using orthogonal frequency division multiplexing (OFDM) with frequency interleaving (e.g., [3, 5]), and for block-fading channels with temporal interleaving.

For MIMO-BICM, a two-stage receiver is usually employed that consists of a demodulator (demapper) and a channel decoder, separated by a deinterleaver. The demodulator computes log-likelihood ratios (LLRs)—i.e., soft values—for the coded bits; these are used by the decoder as bit metrics. In the MIMO case, LLR calculation tends to be excessively complex, even if the log-sum approximation is used [3]. Thus, there is a strong demand for efficient MIMO-BICM soft demodulation algorithms with near-optimum performance. Existing approaches use the list extension of the Fincke-Phost sphere decoding algorithm [6] (abbreviated as LFPSD in what follows), as well as algorithms based on ZF equalization [5] or MMSE equalization [7, 8].

In this paper, we develop the *soft line-search demodulator* (SLSD) by extending the *line-search detector* (LSD) recently introduced in [9, 10]. We show how intermediate calculations of the LSD, which is an efficient *hard*-output MIMO detector achieving near-ML performance, can be used for an approximate computation of LLRs with small extra complexity. Even though the SLSD is much less complex than the LFPSD, simulation results demonstrate that the performance of the SLSD is very close to that of the LFPSD and much better than that of equalization-based demodulation.

Funding by FWF grants P15156 and J2302.

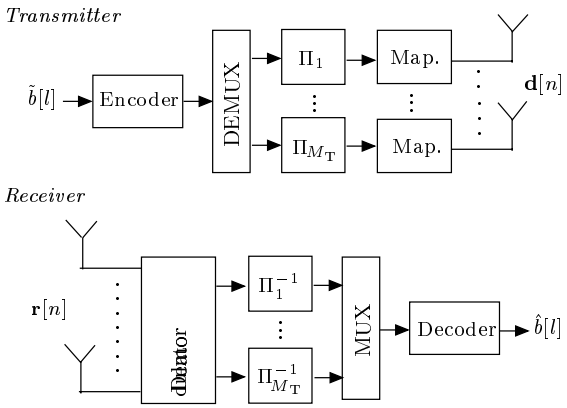

Fig. 1. Block diagram of a MIMO-BICM system [3].

This paper is organized as follows. In Section II, we provide some background on MIMO-BICM. The novel SLSD is presented in Sections III and IV. Finally, simulation results for fast-fading MIMO channels are presented in Section V.

II. MIMO-BICM

A. System Model

We consider a MIMO-BICM system as proposed in [3], with M_T transmit antennas and $M_\mathrm{R} \geq M_\mathrm{T}$ receive antennas (see Fig. 1). A sequence of information bits $\tilde{b}[l]$ is encoded using a convolutional code and cyclically demultiplexed into M_T layers. The coded bits of the kth layer are scrambled by an interleaver Π_k. Subsequently, the interleaved bits are Gray mapped onto complex data symbols $d_k[n] \in \mathcal{A}$ that are transmitted on the kth transmit antenna. The symbols $d_k[n]$ are assumed to have zero mean and unit variance.

The $(M_\mathrm{T}, M_\mathrm{R})$ MIMO channel is assumed to be flat-fading. At any given time n, it is described by the well-known baseband model (the time index n will hereafter be suppressed)

$$\mathbf{r} \;=\; \mathbf{H}\mathbf{d} + \mathbf{w}\,,$$

with the transmitted data symbol vector $\mathbf{d} \triangleq \big(d_1 \cdots d_{M_\mathrm{T}}\big)^T$, the $M_\mathrm{R} \times M_\mathrm{T}$ channel matrix $\mathbf{H}$, the received vector $\mathbf{r} \triangleq \big(r_1 \cdots r_{M_\mathrm{R}}\big)^T$, and the noise vector $\mathbf{w} \triangleq \big(w_1 \cdots w_{M_\mathrm{R}}\big)^T$. The noise components w_k are assumed statistically independent and circularly symmetric complex Gaussian with variance σ_w^2.

At the receiver, the demodulator uses the received vector $\mathbf{r}$ and knowledge of the channel $\mathbf{H}$ to calculate an LLR $\Lambda_k^{(i)}$ for each bit $b_k^{(i)}$ associated with the symbol vector $\mathbf{d}$. (Here, $b_k^{(i)}$ with $i = 1, \ldots, \log_2|\mathcal{A}|$ denotes the coded and interleaved bits of the kth layer that constitute the label for the symbol $d_k \in \mathcal{A}$. The $b_k^{(i)}$'s are assumed statistically independent and equally likely.) The resulting LLRs are deinterleaved (using deinterleavers Π_k^{-1}) and multiplexed into a single stream that constitutes the soft values to be used by the channel decoder.

B. LLR Calculation

In what follows, let $\mathcal{D}_{k,b}^{(i)} \triangleq \{\mathbf{d} : d_k \in \mathcal{A}_b^{(i)}\}$ where $\mathcal{A}_b^{(i)} \subset \mathcal{A}$ denotes the set of all symbols $a \in \mathcal{A}$ whose label at bit position i equals $b \in \{0, 1\}$. The LLR for $b_k^{(i)}$ is given by

$$\begin{aligned}\Lambda_k^{(i)} \;&\triangleq\; \log\!\left(\frac{f(\mathbf{r}|b_k^{(i)}=1)}{f(\mathbf{r}|b_k^{(i)}=0)}\right)\\ &=\; \log\!\left(\frac{\sum_{\mathbf{d}\in\mathcal{D}_{k,1}^{(i)}} e^{-\frac{1}{\sigma_w^2}\|\mathbf{r}-\mathbf{H}\mathbf{d}\|^2}}{\sum_{\mathbf{d}\in\mathcal{D}_{k,0}^{(i)}} e^{-\frac{1}{\sigma_w^2}\|\mathbf{r}-\mathbf{H}\mathbf{d}\|^2}}\right). \qquad (1)\end{aligned}$$

The computational complexity of (1) is exponential in the number of transmit antennas M_T. A simplification is usually obtained through the log-sum approximation [3, 6]

$$\Lambda_k^{(i)} \;\approx\; \frac{1}{\sigma_w^2}\left[\min_{\mathbf{d}\in\mathcal{D}_{k,0}^{(i)}} \psi^2(\mathbf{d}) \;-\; \min_{\mathbf{d}\in\mathcal{D}_{k,1}^{(i)}} \psi^2(\mathbf{d})\right] \qquad (2)$$

with $\psi(\mathbf{d}) \triangleq \|\mathbf{r}-\mathbf{H}\mathbf{d}\|$. However, the complexity of (2) still is exponential in M_T and can be excessive for practical values of $|\mathcal{A}|$ and M_T [6]. Hence, various approximations to (2) have been proposed [5–8].

For convenience, we introduce the short-hand notation

$$\lambda_{k,b}^{(i)} \;\triangleq\; \min_{\mathbf{d}\in\mathcal{D}_{k,b}^{(i)}} \psi^2(\mathbf{d})\,, \qquad (3)$$

which allows us to rewrite (2) as

$$\Lambda_k^{(i)} \;\approx\; \frac{1}{\sigma_w^2}\Big[\lambda_{k,0}^{(i)} - \lambda_{k,1}^{(i)}\Big]\,. \qquad (4)$$

III. The Novel SLSD: Basics

Let us first consider the ML detector (e.g., [6])

$$\hat{\mathbf{d}}_\mathrm{ML} \;=\; \operatorname*{argmin}_{\mathbf{d}\in\mathcal{D}} \|\mathbf{r}-\mathbf{H}\mathbf{d}\|^2 \;=\; \operatorname*{argmin}_{\mathbf{d}\in\mathcal{D}} \psi^2(\mathbf{d})\,, \qquad (5)$$

where $\mathcal{D} \triangleq \mathcal{A}^{M_\mathrm{T}}$ denotes the set of all possible data vectors $\mathbf{d}$. Assume that the ith bit at the kth layer of $\hat{\mathbf{d}}_\mathrm{ML}$ equals $(\hat{\mathbf{d}}_\mathrm{ML})_k^{(i)} = b$. In this case, the ML vector in (5) can be written as $\hat{\mathbf{d}}_\mathrm{ML} = \arg\min_{\mathbf{d}\in\mathcal{D}_{k,b}^{(i)}} \psi^2(\mathbf{d})$, and hence $\psi^2(\hat{\mathbf{d}}_\mathrm{ML}) = \min_{\mathbf{d}\in\mathcal{D}_{k,b}^{(i)}} \psi^2(\mathbf{d})$. Comparing with (3), we then have

$$\lambda_{k,b}^{(i)} \;=\; \psi^2(\hat{\mathbf{d}}_\mathrm{ML})\,. \qquad (6)$$

Thus, $\psi^2(\hat{\mathbf{d}}_\mathrm{ML})$ is either $\lambda_{k,0}^{(i)}$ (if $b = 0$) or $\lambda_{k,1}^{(i)}$ (if $b = 1$). This observation could be exploited for using the ML detector (5) for soft demodulation according to (4). However, the ML detector also has exponential complexity, and furthermore a separate minimization would have to be performed to compute the respective other λ term in (4), i.e., $\lambda_{k,\bar{b}}^{(i)}$ where $\bar{b} \triangleq 1 - b$ denotes bit flipping.

The novel *soft line-search demodulator* (SLSD) is now obtained by replacing the ML detector with the LSD. Because the LSD is a low-complexity approximation to the ML detector, it yields an approximation to $\lambda_{k,b}^{(i)}$ with $b = 0$ or $b = 1$, which is used in (4). We will show that an approximation to the respective other term $\lambda_{k,\bar{b}}^{(i)}$ in (4) can be computed very efficiently from intermediate LSD results. We first provide a brief review of the LSD.

A. Review of the LSD

The LSD [9, 10] is an efficient approximation to the ML detector (5) with complexity $\mathcal{O}(M_\mathrm{T}^3)$. It uses a *reduced search set* $\tilde{\mathcal{D}} \triangleq \{\mathbf{d}^1, \ldots, \mathbf{d}^{|\tilde{\mathcal{D}}|}\} \subset \mathcal{D}$ instead of $\mathcal{D}$, i.e.,

$$\hat{\mathbf{d}}_\mathrm{LSD} \;=\; \operatorname*{argmin}_{\mathbf{d}\in\tilde{\mathcal{D}}} \psi^2(\mathbf{d})\,. \qquad (7)$$

The construction of $\tilde{\mathcal{D}}$ is based on an approximation of the channel $\mathbf{H}$ by an *idealized bad channel* (IBC). The IBC approximation captures essential properties of bad (i.e., poorly conditioned) channel realizations by setting the smallest singular value of the matrix $\mathbf{H}$ equal to zero and the remaining singular values equal to the largest singular value. This is motivated by experimental evidence that for bad channels, the smallest singular value tends to dominate system performance.

In what follows, let $\mathbf{y}_\mathrm{ZF} \triangleq (\mathbf{H}^H\mathbf{H})^{-1}\mathbf{H}^H\mathbf{r}$ denote the ZF-equalized received vector. For an IBC, it can be shown [9, 10] that the total data vector set $\mathcal{D}$ in (5) can be replaced by the set $\tilde{\mathcal{D}}$ of data vectors whose associated ZF decision regions (corresponding to componentwise quantization of $\mathbf{y}_\mathrm{ZF}$) are intersected by the *reference line*

$$\mathcal{L}: \quad \mathbf{y}_\mathrm{ref}(\alpha) \;=\; \alpha\,\mathbf{v}_{M_\mathrm{T}} + \mathbf{y}_\mathrm{ZF}\,, \qquad \alpha \in \mathbb{C}\,.$$

Here, $\mathbf{v}_{M_\mathrm{T}}$ denotes the channel's right singular vector corresponding to the zero singular value of the IBC (i.e., corresponding to the smallest singular value of the actual channel).

The intersection of the ZF decision regions with $\mathcal{L}$ can be calculated efficiently by viewing the complex reference line $\mathcal{L}$ as a 2-D real *reference plane* $\mathcal{P}$ parameterized by $\mathrm{Re}\{\alpha\}$ and $\mathrm{Im}\{\alpha\}$. The intersection of the ZF decision regions with $\mathcal{P}$

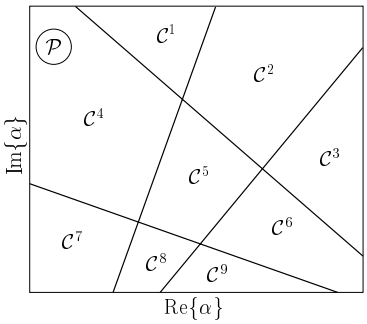

Fig. 2. Cell partitioning of the real reference plane $\mathcal{P}$ for a $(2,2)$ MIMO system and 4-QAM modulation.

then yields $|\tilde{\mathcal{D}}|$ *cells* $\mathcal{C}^r$, $r = 1, \ldots, |\tilde{\mathcal{D}}|$. This cell partitioning of $\mathcal{P}$ is illustrated in Fig. 2. To each cell $\mathcal{C}^r$, there corresponds a data vector $\mathbf{d}^r \in \tilde{\mathcal{D}}$ that is given componentwise by

$$d_k^r = Q_{\mathcal{A}}\{y_{\mathrm{ref},k}(\alpha)\}\,, \quad \text{for any } \alpha \in \mathcal{C}^r, \tag{8}$$

where d_k^r and $y_{\mathrm{ref},k}(\alpha)$ denote the kth component of $\mathbf{d}^r$ and $\mathbf{y}_{\mathrm{ref}}(\alpha)$, respectively and $Q_{\mathcal{A}}\{\cdot\}$ denotes componentwise quantization according to the alphabet $\mathcal{A}$. To obtain $\mathbf{d}^r$, it thus suffices to determine an arbitrary point $\mathbf{y}_{\mathrm{ref}}(\alpha)$ in $\mathcal{C}^r$.

The efficient procedure used by the LSD to calculate the $\mathbf{d}^r$'s is described in detail in [9, 10]. This procedure is based on the assumption that the symbol alphabet $\mathcal{A}$ is "line-structured," i.e., the boundaries of the symbol quantization regions of $\mathcal{A}$ are P straight lines. Examples include ASK, QAM and PSK alphabets (e.g., $P = 2$ for 4-QAM) but not, e.g., an hexagonal constellation. The LSD algorithm yields the reduced search set $\tilde{\mathcal{D}} = \{\mathbf{d}^1, \ldots, \mathbf{d}^{|\tilde{\mathcal{D}}|}\}$ and the corresponding set of distances $\tilde{\Psi} \triangleq \{\psi^2(\mathbf{d}^1), \ldots, \psi^2(\mathbf{d}^{|\tilde{\mathcal{D}}|})\}$, for use in (7).

It can be shown [9, 10] that the size of $\tilde{\mathcal{D}}$ is bounded as

$$|\tilde{\mathcal{D}}| \leq \frac{(M_{\mathrm{T}}P)^2}{2} + \frac{M_{\mathrm{T}}P}{2} + 1\,.$$

For example, for a (6,6) channel and 4-QAM modulation we have $|\tilde{\mathcal{D}}| \leq 79$ rather than $|\mathcal{D}| = 4096$ data vectors. Furthermore, $\tilde{\mathcal{D}}$ always contains both the result of ML detection for the IBC and the result of ZF detection for the actual channel $\mathbf{H}$. Thus, $\hat{\mathbf{d}}_{\mathrm{LSD}} = \hat{\mathbf{d}}_{\mathrm{ML}}$ if the channel is either an IBC or an idealized "good" channel (i.e., a channel with condition number 1, for which ZF detection is optimum).

B. The SLSD Approach

The LSD can be extended to obtain approximations to $\lambda_{k,0}^{(i)}$ and $\lambda_{k,1}^{(i)}$ in (4). Assume that the ith bit at the kth layer of $\hat{\mathbf{d}}_{\mathrm{LSD}}$ equals $(\hat{\mathbf{d}}_{\mathrm{LSD}})_k^{(i)} = b$. Because $\hat{\mathbf{d}}_{\mathrm{LSD}}$ is an approximation to $\hat{\mathbf{d}}_{\mathrm{ML}}$, the corresponding distance

$$\tilde{\lambda}_{k,b}^{(i)} \triangleq \psi^2(\hat{\mathbf{d}}_{\mathrm{LSD}})$$

is an approximation to $\lambda_{k,b}^{(i)} = \psi^2(\hat{\mathbf{d}}_{\mathrm{ML}})$ (cf. (6)).

It remains to calculate a similar approximation to the respective other λ term in (4), $\lambda_{k,\bar{b}}^{(i)}$. According to (3), $\lambda_{k,\bar{b}}^{(i)} = \min_{\mathbf{d}\in\mathcal{D}_{k,\bar{b}}^{(i)}} \psi^2(\mathbf{d})$. Thus, an approximation to $\lambda_{k,\bar{b}}^{(i)}$ is given by

$$\tilde{\lambda}_{k,\bar{b}}^{(i)} \triangleq \min_{\mathbf{d}\in\tilde{\mathcal{D}}_{k,\bar{b}}^{(i)}} \psi^2(\mathbf{d})\,,$$

where $\tilde{\mathcal{D}}_{k,\bar{b}}^{(i)}$ is the reduced LSD search set corresponding to $\mathcal{D}_{k,\bar{b}}^{(i)}$. The set $\tilde{\mathcal{D}}_{k,\bar{b}}^{(i)}$ is obtained by using $\mathcal{A}_{\bar{b}}^{(i)}$ instead of $\mathcal{A}$ for layer k. This implies a new cell partitioning of $\mathcal{P}$, and thus $\tilde{\mathcal{D}}_{k,\bar{b}}^{(i)}$ is different from $\tilde{\mathcal{D}}$. We could perform a second LSD pass with this new cell partitioning to determine $\tilde{\lambda}_{k,\bar{b}}^{(i)}$, but then the LSD computations would have to be redone for each layer k and each label index i. However, in the next section we will show that for a practically important class of symbol alphabets, *the second LSD pass can be circumvented*.

We note that the approximations $\tilde{\lambda}_{k,b}^{(i)}$ and $\tilde{\lambda}_{k,\bar{b}}^{(i)}$ become exact, i.e., $\tilde{\lambda}_{k,b}^{(i)} = \lambda_{k,b}^{(i)}$ and $\tilde{\lambda}_{k,\bar{b}}^{(i)} = \lambda_{k,\bar{b}}^{(i)}$, if the channel is either an IBC or an idealized "good" channel because in these two cases LSD detection is equal to ML detection.

IV. The Novel SLSD: Algorithm

A second LSD pass is not required because $\tilde{\lambda}_{k,\bar{b}}^{(i)}$ can be calculated very efficiently from the sets $\tilde{\mathcal{D}}$ and $\tilde{\Psi}$ obtained during the first LSD pass, as explained in the following.

A. Efficient Calculation of $\tilde{\lambda}_{k,\bar{b}}^{(i)}$

The symbol quantization (decision) region associated with a symbol $a_j \in \mathcal{A}$ is (cf. Fig. 3(a) for 4-QAM)

$$\mathcal{Q}_j \triangleq \left\{y \,\middle|\, |y-a_j| \leq |y-a_{j'}|, \text{ for all } j' \neq j\right\}.$$

Of course, $Q_{\mathcal{A}}\{y\} = a_j$ for any $y \in \mathcal{Q}_j$. Consider now the reduced alphabet $\mathcal{A}_{\bar{b}}^{(i)} \subset \mathcal{A}$ of size $|\mathcal{A}_{\bar{b}}^{(i)}| = |\mathcal{A}|/2$ that consists of all symbols $\tilde{a}_j \in \mathcal{A}$, $j = 1, \ldots, |\mathcal{A}|/2$ whose label at bit position i equals $\bar{b} = 1 - b$. The quantization regions $\tilde{\mathcal{Q}}_j$ associated with the symbols $\tilde{a}_j \in \mathcal{A}_{\bar{b}}^{(i)}$ are different from the quantization regions $\mathcal{Q}_j$, as illustrated in Fig. 3(b). To allow for further simplifications, we assume in the following that each $\tilde{\mathcal{Q}}_j$ is the union of certain $\mathcal{Q}_{j'}$, i.e.,

$$\tilde{\mathcal{Q}}_j = \bigcup_{j'\in\mathcal{J}_j} \mathcal{Q}_{j'}, \tag{9}$$

with some disjoint index sets $\mathcal{J}_j \subset \{1, \ldots, |\mathcal{A}|\}$. This assumption holds e.g. for $|\mathcal{A}|$-PSK with $|\mathcal{A}| = 2^l$, $l \in \mathbb{N}$ (e.g., BPSK, QPSK or 4-QAM, and 8-PSK) using Gray labeling. For an illustration see Fig. 3(a), (b). In Section IV-C, we will briefly discuss how this assumption can be avoided.

According to (8), a data vector $\mathbf{d}^r \in \tilde{\mathcal{D}}$ can be obtained by componentwise $\mathcal{A}$-quantization of $\mathbf{y}_{\mathrm{ref}}(\alpha)$ with $\alpha \in \mathcal{C}^r$:

$$d_k^r = Q_{\mathcal{A}}\{y_{\mathrm{ref},k}(\alpha)\}\,, \quad \alpha \in \mathcal{C}^r, \quad k = 1, \ldots, M_{\mathrm{T}}\,.$$

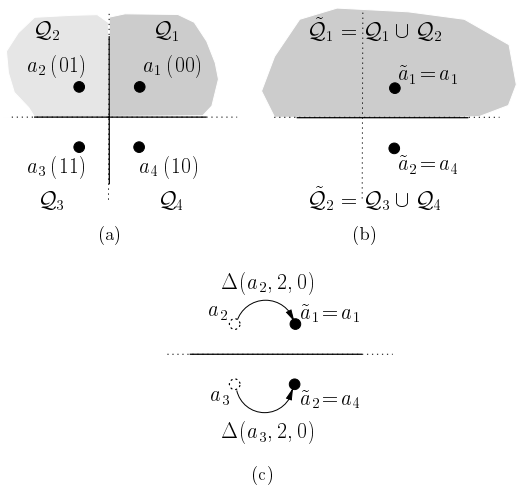

Fig. 3. Reduction of the 4-QAM symbol alphabet $\mathcal{A}$ to $\mathcal{A}_0^{(2)}$: (a) Symbol alphabet $\mathcal{A}$ with quantization regions $\mathcal{Q}_j$, (b) reduced symbol alphabet $\mathcal{A}_0^{(2)}$ with quantization regions $\tilde{\mathcal{Q}}_j$, (c) corresponding symbol re-mapping.

For a data vector $\tilde{\mathbf{d}}^{r'} \in \tilde{\mathcal{D}}_{k,\bar{b}}^{(i)}$, on the other hand, we have $\tilde{d}_k^{r'} \in \mathcal{A}_{\bar{b}}^{(i)}$ instead of $\tilde{d}_k^{r'} \in \mathcal{A}$, and thus we must use

$$\tilde{d}_k^{r'} = Q_{\mathcal{A}_{\bar{b}}^{(i)}}\{y_{\mathrm{ref},k}(\alpha)\}\,, \quad \alpha \in \tilde{\mathcal{C}}^{r'}$$

for the given layer k. Here, $\tilde{\mathcal{C}}^{r'}$ denotes a cell associated with $\tilde{\mathcal{D}}_{k,\bar{b}}^{(i)}$. However, (9) implies that

$$Q_{\mathcal{A}_{\bar{b}}^{(i)}}\{y\} \;=\; Q_{\mathcal{A}_{\bar{b}}^{(i)}}\big\{Q_{\mathcal{A}}\{y\}\big\}\,, \quad y \in \mathbb{C}\,.$$

Thus, we can calculate an LSD candidate data vector $\tilde{\mathbf{d}}^{r'} \in \tilde{\mathcal{D}}_{k,\bar{b}}^{(i)}$ from a data vector $\mathbf{d}^r \in \tilde{\mathcal{D}}$ obtained at the first LSD pass by re-quantizing layer k according to

$$\tilde{d}_k^{r'} = Q_{\mathcal{A}_{\bar{b}}^{(i)}}\{d_k^r\}\,, \tag{10}$$

while leaving the other layers unchanged. This amounts to a *symbol re-mapping* of the form

$$\tilde{d}_k^{r'} = d_k^r + \Delta(d_k^r, i, \bar{b})\,, \tag{11}$$

with a suitable offset $\Delta(d_k^r, i, \bar{b})$. For example, for 4-QAM we have

$$\Delta(d_k^r, i, \bar{b}) \;=\; \begin{cases} -2j\ \mathrm{Im}\{d_k^r\}\,, & \text{if } i=1 \text{ and } d_k^r \notin \mathcal{A}_{\bar{b}}^{(i)} \\ -2\ \mathrm{Re}\{d_k^r\}\,, & \text{if } i=2 \text{ and } d_k^r \notin \mathcal{A}_{\bar{b}}^{(i)} \\ 0\,, & \text{otherwise.} \end{cases}$$

This is illustrated in Fig. 3(c) for $\mathcal{A}_{\bar{b}}^{(i)} = \mathcal{A}_0^{(2)}$.

The structure expressed by (9) has another important consequence. Indeed, (9) implies that each LSD cell $\tilde{\mathcal{C}}^{r'}$ for $\tilde{\mathcal{D}}_{k,\bar{b}}^{(i)}$ is a union of certain LSD cells $\mathcal{C}^r$ for $\mathcal{D}$. Hence, if (10) is calculated for *each* data vector in $\tilde{\mathcal{D}}$, it is guaranteed that the *entire* set $\tilde{\mathcal{D}}_{k,\bar{b}}^{(i)}$ is constructed. Note, however, that some data vectors $\tilde{\mathbf{d}} \in \tilde{\mathcal{D}}_{k,\bar{b}}^{(i)}$ will be multiply obtained.

To calculate $\tilde{\lambda}_{k,\bar{b}}^{(i)}$, we do not need the set of vectors $\tilde{\mathcal{D}}_{k,\bar{b}}^{(i)}$ but just the corresponding set of distances $\tilde{\Psi}_{k,\bar{b}}^{(i)}$. Using (11), it can be shown that each distance $\psi^2(\tilde{\mathbf{d}}^{r'}) \in \tilde{\Psi}_{k,\bar{b}}^{(i)}$ can be efficiently obtained from some $\mathbf{d}^r \in \tilde{\mathcal{D}}$ and $\psi^2(\mathbf{d}^r) \in \tilde{\Psi}$ as

$$\begin{aligned}\psi^2(\tilde{\mathbf{d}}^{r'}) \;=\; \psi^2(\mathbf{d}^r) \;+\; \|\mathbf{h}_k\|^2 |\Delta(d_k^r, i, \bar{b})|^2 \\ -\; 2\,\mathrm{Re}\Big\{\big(\mathbf{r} - \mathbf{H}\mathbf{d}^r\big)^H \mathbf{h}_k\, \Delta(d_k^r, i, \bar{b})\Big\}\,,\end{aligned} \tag{12}$$

where $\mathbf{h}_k$ denotes the kth column of $\mathbf{H}$.

B. Algorithm Summary

The SLSD algorithm can be summarized as follows.

1) Use the LSD to calculate the reduced search set $\tilde{\mathcal{D}}$, the associated distance set $\tilde{\Psi}$, and $\psi^2(\hat{\mathbf{d}}_{\mathrm{LSD}})$, the minimum element of $\tilde{\Psi}$.
2) For each layer $k \in \{1, \ldots, M_{\mathrm{T}}\}$ and each bit index $i \in \{1, \ldots, \log_2|\mathcal{A}|\}$, perform the following steps:
   a) determine $b = (\hat{\mathbf{d}}_{\mathrm{LSD}})_k^{(i)}$ and set $\tilde{\lambda}_{k,b}^{(i)} = \psi^2(\hat{\mathbf{d}}_{\mathrm{LSD}})$;
   b) calculate $\tilde{\Psi}_{k,\bar{b}}^{(i)}$ by evaluating (12) for each $\mathbf{d}^r \in \tilde{\mathcal{D}}$;
   c) obtain $\tilde{\lambda}_{k,\bar{b}}^{(i)}$ as the minimum of $\tilde{\Psi}_{k,\bar{b}}^{(i)}$;
   d) finally, calculate the approximate LLR

$$\tilde{\Lambda}_k^{(i)} \;=\; \frac{1-2b}{\sigma_w^2}\Big[\tilde{\lambda}_{k,b}^{(i)} - \tilde{\lambda}_{k,\bar{b}}^{(i)}\Big]\,,$$

   where the factor $1-2b \in \{-1, 1\}$ is necessary to adjust the sign of $\tilde{\Lambda}_k^{(i)}$ (cf. (4)).

We note that in the case of an IBC or an idealized good channel, $\tilde{\Lambda}_k^{(i)}$ becomes equal to the log-sum approximation for $\Lambda_k^{(i)}$ (the right hand side of (2) or (4)).

C. Extension to General Alphabets

The calculation of $\tilde{\lambda}_{k,\bar{b}}^{(i)}$ was based on the assumption (9). This assumption does not hold if $\mathcal{A}$ is a QAM alphabet with $|\mathcal{A}| > 4$ (e.g., 16-QAM). However, we can extend $\mathcal{A}$ to a *virtual* symbol alphabet $\mathcal{A}_{\mathrm{v}}$ whose associated *virtual* quantization regions $\mathcal{Q}_{\mathrm{v},j}$ satisfy (9), and then apply the LSD assuming $d_k \in \mathcal{A}_{\mathrm{v}}$. From the obtained virtual sets $\tilde{\mathcal{D}}_{\mathrm{v}}$ and $\tilde{\Psi}_{\mathrm{v}}$, $\tilde{\lambda}_{k,b}^{(i)}$ and $\tilde{\lambda}_{k,\bar{b}}^{(i)}$ can be calculated by evaluating (12) for every $\mathbf{d}^r \in \tilde{\mathcal{D}}_{\mathrm{v}}$. Obviously, the complexity is increased since $|\mathcal{A}_{\mathrm{v}}| > |\mathcal{A}|$ and additional distances have to be calculated. For example, for 16-QAM we obtain $|\mathcal{A}_{\mathrm{v}}| = 36$.

V. Simulation Results

We now assess the bite error rate (BER) performance and computational complexity of the SLSD by means of simulation results. We considered a MIMO-BICM transmitter employing a rate-$1/2$ 16-state convolutional code with octal generators $(23, 35)$ and trellis termination using 4 bits, followed by a random block interleaver. A 4-QAM (QPSK) symbol alphabet with Gray labeling was used. The MIMO channel of size $(4, 4)$, $(6, 6)$, or $(8, 8)$ had iid Gaussian matrix entries with unit variance. To simulate fast fading, the channel was independently generated for each time instant. A Viterbi decoder with a traceback depth of 25 was employed for channel decoding.

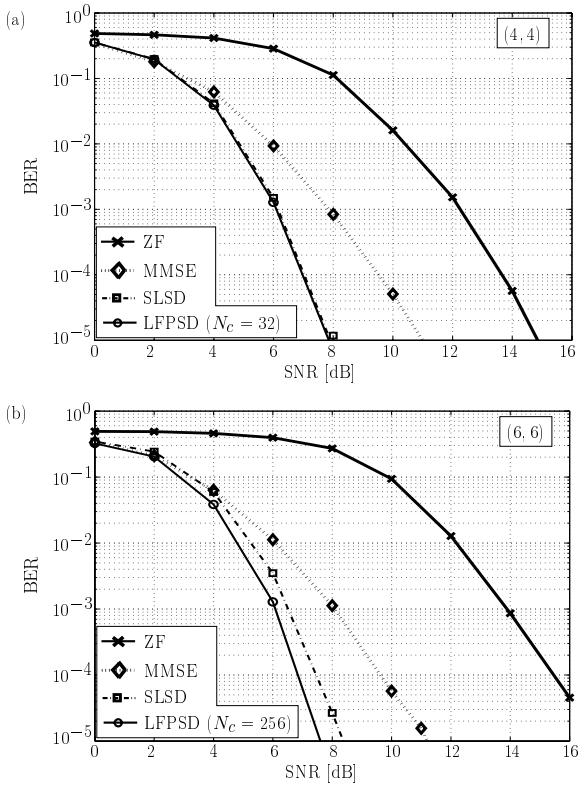

Fig. 4. BER performance of the proposed SLSD, the ZF-based and MMSE-based demodulators, and the LFPSD demodulator for 4-QAM (QPSK) modulation: (a) $(4, 4)$ channel, (b) $(6, 6)$ channel.

For comparison, we also considered the LFPSD, ZF-based, and MMSE-based demodulators [5–8]. The LFPSD used LLR clipping to thresholds ± 8; the number of canditate data vectors inside the hypersphere was $N_c = 32$ for the $(4, 4)$ channel and $N_c = 256$ for the $(6, 6)$ and $(8, 8)$ channels.

A. BER Performance

For the $(4, 4)$ and $(6, 6)$ channels, Fig. 4 shows the BER obtained with the various demodulators versus the SNR (the SNR is defined as $\mathrm{E}\{\|\mathbf{Hd}\|^2\}/\mathrm{E}\{\|\mathbf{w}\|^2\} = M_{\mathrm{T}}/\sigma_w^2$). It is seen that the BER performance of the proposed SLSD is practically identical to the performance of the LFPSD for the $(4, 4)$ channel, and within about 0.7 dB of LFPSD performance for the $(6, 6)$ channel. Furthermore, the SLSD significantly outperforms ZF-based and MMSE-based demodulation.

B. Computational Complexity

To convey a rough picture of the computational complexity of the various demodulators, Table I displays kflop estimates that were determined using a MATLAB V5.3 implementation. Because the complexity of LFPSD strongly depends on the actual channel realization and the SNR, Table I also shows the maximum LFPSD complexity (obtained during 1000 simulation runs at an SNR of 6 dB) in addition to the average LFPSD complexity. (The complexity of the proposed SLSD does not depend on the channel realization and SNR.)

channel	LFPSD		SLSD	ZF, MMSE
	av.	max.		
$(4, 4)$	46	147	34	2.5
$(6, 6)$	534	2053	112	7
$(8, 8)$	1848	12173	285	15

TABLE I
MEASURED COMPUTATIONAL COMPLEXITY IN KFLOPS OF THE VARIOUS DEMODULATORS.

It is seen that the SLSD is more complex than the ZF and MMSE demodulators, which was to be expected in view of its substantially improved BER performance. On the other hand, the SLSD's complexity is significantly smaller than both the average and the maximum complexity of LFPSD.

VI. Conclusions

The novel *soft line-search demodulator* (SLSD) is an efficient demodulation technique for MIMO systems using bit-interleaved coded modulation (BICM). The SLSD exploits intermediate calculation results of the line-search detector (LSD), a recently proposed hard-decision algorithm with near-ML performance, to obtain approximate log-likelihood ratios for the coded bits with little extra computational effort. Simulation results demonstrated that the performance of the SLSD is close to that of the list extension of the Fincke-Phost sphere decoding algorithm although the computational complexity is significantly smaller.

References

[1] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Trans. Inf. Theory*, vol. 44, pp. 927–945, May 1998.

[2] J. J. Boutros, F. Boixadera, and C. Lamy, "Bit-interleaved coded modulations for multiple-input multiple-output channels," in *Proc. IEEE Symp. Spread Spectrum Techn. Appl.*, Parsippany, NJ, Sept. 2000, pp. 123–126.

[3] S. H. Müller-Weinfurtner, "Coding approaches for multiple antenna transmission in fast fading and OFDM," *IEEE Trans. Signal Processing*, vol. 50, pp. 2442–2450, Oct. 2002.

[4] R. Visoz, A. O. Berthet, and J. J. Boutros, "Reduced-complexity iterative decoding and channel estimation for space time BICM over frequency-selective wireless channels," in *Proc. IEEE PIMRC-02*, Lisbon, Portugal, Sept. 2002, pp. 1017–1022.

[5] M. Butler and I. Collings, "A zero-forcing approximate log-likelihood receiver for MIMO bit-interleaved coded modulation," *IEEE Commun. Letters*, vol. 8, no. 2, pp. 105–107, Feb. 2004.

[6] B. M. Hochwald and S. ten Brink, "Achieving near-capacity on a multiple-antenna channel," *IEEE Trans. Inf. Theory*, vol. 51, no. 3, pp. 389–399, March 2003.

[7] K.-B. Song and S. A. Mujtaba, "A low complexity space-frequency BICM MIMO-OFDM system for next-generation WLANs," in *Proc. IEEE Globecom 2003*, San Francisco, CA, Dec. 2003, pp. 1059–1063.

[8] D. Seethaler, G. Matz, and F. Hlawatsch, "An efficient MMSE-based demodulator for MIMO bit-interleaved coded modulation," in *Proc. IEEE Globecom 2004*, vol. IV, Dallas, Texas, Dec. 2004, pp. 2455–2459.

[9] H. Artés, D. Seethaler, and F. Hlawatsch, "Efficient detection algorithms for MIMO channels: A geometrical approach to approximate ML detection," *IEEE Trans. Signal Processing, Special Issue on MIMO Communications Systems*, vol. 51, no. 11, pp. 2808–2820, Nov. 2003.

[10] D. Seethaler, H. Artés, and F. Hlawatsch, "Efficient approximate-ML detection for MIMO spatial multiplexing systems by using a 1-D nearest neighbor search," in *Proc. IEEE ISSPIT 2003*, Darmstadt, Germany, invited paper, Dec. 2003, pp. 290–293.