El Chall *et al. EURASIP Journal on Wireless Communications and Networking* (2015) 2015:69
DOI 10.1186/s13638-015-0305-6

EURASIP Journal on
Wireless Communications and Networking
a SpringerOpen Journal

RESEARCH Open Access

Iterative receivers combining MIMO detection with turbo decoding: performance-complexity trade-offs

Rida El Chall[1*], Fabienne Nouvel[1], Maryline Hélard[1] and Ming Liu[1,2*]

Abstract

Recently, iterative receiver combining multiple-input multiple-output (MIMO) detection with channel decoding has been widely considered to achieve near-capacity performance and reliable high data rate transmission, for future wireless communication systems. However, such iterative processing increases the computational complexity at the receiver. In this paper, the computational complexity of MIMO detection algorithms combined with turbo decoding is investigated. We first present an overview of the family of MIMO detection algorithms based on sphere decoding, *K*-Best decoding, and interference cancellation. A recently proposed low-complexity *K*-Best decoder (LC-*K*-Best) is also presented. Moreover, we analyze the convergence of combining these detection algorithms with the turbo decoder using the extrinsic information transfer (EXIT) chart. Consequently, a new scheduling order of the number of iterations for the iterative process is proposed. Several system configurations are developed and compared in terms of performance and complexity. Simulations and analytical results show that the new scheduling provides good performance with a large saving in the complexity. Additionally, the LC-*K*-Best decoder shows a good performance-complexity tradeoff, and it is therefore suitable for parallel and pipeline architectures that can meet high throughput requirements.

Keywords: Iterative receiver; MIMO; Sphere decoder; *K*-Best decoder; MMSE-IC; VBLAST; Turbo decoder

1 Introduction

Multiple-input multiple-output (MIMO) technology is an effective solution to increase the channel capacity and to improve the link reliability of wireless communications systems [1]. Actually, MIMO technology is combined with orthogonal frequency division multiplexing (OFDM) and advanced channel coding schemes such as turbo codes to support the increase of reliable data transmission. These techniques have been incorporated into the latest standards such as IEEE 802.11n/ac, IEEE 802.16e/m, and 3GPP-Long Term Evolution (LTE).

In such coded MIMO-OFDM systems, the optimal way to decode the received signal would be the joint detection-decoding that reveals to be very complex and infeasible for practical implementation. An alternative solution is to perform the detection and the decoding steps iteratively with soft information exchanging. Such method, commonly referred to as iterative or turbo processing, was initially proposed for turbo decoding [2] where two component decoders exchange soft information to improve the system performance. The turbo principle was rapidly extended to the turbo equalization [3], where equalization and channel decoding were performed iteratively to overcome inter-symbol interference (ISI) [4]. Turbo equalization principle was then applied to several transmission systems, such as systems with multi-user interference [5] and multi-antenna interference [6].

For the coded MIMO-OFDM systems, the optimal detection relies on maximum *a posteriori* probability (MAP) and maximum likelihood (ML) algorithms, which present an exponentially increased complexity with respect to the number of transmit antennas and modulation orders. Therefore, a number of sub-optimal approaches have been proposed in the literature [7-15]. These approaches are based on linear equalization, interference cancellation, and tree-search-based detection.

*Correspondence: rida.el-chall@insa-rennes.fr; mingliu@bjtu.edu.cn
[1] INSA, IETR, CNRS UMR 6164, F-35708, Rennes, France
[2] School of Computer and Information Technology, Beijing Jiaotong University, Beijing, China

Linear equalization consists of a linear filter according to the zero forcing (ZF) or the minimum mean square error (MMSE) criteria [7]. These algorithms need low complexity but suffer from unsatisfactory performance. On the other hand, interference cancellation-based algorithms use an estimation of the previous detected symbols to cancel their interference from the received signal such as ordered successive interference cancellation (OSIC) also referred to as VBLAST [8]. However, their performances suffer from error propagation in the decision feedback loop. The signal detection can also be transformed into a tree-search problem [9-13]. The sphere decoder (SD) is an efficient tree-search-based method that limits the search space of the ML solution to the symbols that lie inside a hyper-sphere. The sphere decoder performs a depth-first search to efficiently find the best solution and achieve near optimal performance. However, it suffers from variable throughput depending on the noise levels and channel conditions [16,17]. Moreover, the sequential nature of the tree search makes it unsuitable for parallel implementation. The breath-first based K-Best decoder [14] and fixed sphere decoder (FSD) [15] are thus proposed to obtain a constant throughput and to reduce the hardware complexity at a cost of certain performance loss.

Recently, many efforts have been made in the design of soft-input soft-output (SISO) MIMO detectors in order to achieve high throughput and low computational complexity. An improved VBLAST (I-VBLAST) for SISO detection was proposed in [18,19]. In addition, a SISO detector based on MMSE interference cancellation (MMSE-IC) was proposed in [20,21]. The list sphere decoder (LSD) was proposed in [22] as a variant of the sphere decoder to provide soft outputs. Consequently, in [23], a list sequential decoder based on metric-first search strategy was proposed for the iterative process. The single tree-search (STS) algorithm [24,25] was proposed to find the MAP hypothesis and the corresponding counter hypotheses during one tree-search process. In [26], the tuple search detector (TSD) was introduced to improve the trade-off between STS-SD and LSD. Furthermore, soft versions of K-Best decoder and FSD decoder for iterative receiver were proposed in [27-29] and [30], respectively. The implementation of MIMO detectors have also been widely discussed in the literature. In [21], an implementation of a SISO detector based on MMSE-IC algorithm was presented. However, this algorithm is not able to fully exploit the spatial diversity of MIMO system. Several implementations of SISO STS-SD were then reported in [31,32] to exploit the spatial diversity. A VLSI architecture of TSD was proposed in [26]. Their main issue is their prohibitive worst-case complexity. In [33], a trellis-search-based SISO decoder and its VLSI architecture have been proposed. Such trellis-based decoder provides a high throughput at the cost of a large hardware area. Several implementations of SISO FSD were proposed in [34-36]. Several implementations of K-Best decoder were also reported [37-39]. Despite these efforts, it is still very challenging to develop a high speed iterative receiver with efficient MIMO detector to meet the high throughput requirements at an affordable complexity and implementation cost.

In this paper, we first present an overview of the main existing soft-input soft-output MIMO detection algorithms. Consequently, a low-complexity K-Best (LC-K-Best) decoder is presented [40]. We then analyze the convergence of the iterative receiver which combines MIMO detection with turbo decoding. The extrinsic information transfer (EXIT) chart [41] is adopted for analyzing the convergence behavior of the iteratively decoded system. A new scheduling of the number of iterations is therefore obtained. Moreover, the complexity of several MIMO detection algorithms is evaluated. We compare the performance and the analytical complexities of STS-SD, LC-K-Best, I-VBLAST, and MMSE-IC-based receivers using the original and the new scheduling orders of the number of iterations in different system configurations. Simulation results show that the LC-K-Best decoder and the new schedule give the best performance-complexity trade-off among existing solutions.

The rest of this paper is organized as follows. Section 2 introduces the system model and the principle of iterative detection-decoding process. Section 3 provides an overview of MIMO detection algorithms based on sphere decoding, K-Best decoding and interference cancellation, followed by a description of the LC-K-Best decoder. Section 4 illustrates the convergence behavior of the iterative process. In Section 5, the performance of these MIMO detectors is compared with different system configurations. In Section 6, the computational complexity versus performance trade-offs are explicitly discussed. We conclude the paper in Section 7.

2 MIMO system model and turbo principle

2.1 System model

We consider a MIMO system based on bit-interleaved coded modulation (BICM) scheme [42] with M transmit antennas and N receive antennas ($N \geq M$) as depicted in Figure 1.

At the transmitter, the data stream is first encoded and punctured with a coding rate R_c. The turbo encoder is constituted by a parallel concatenation of two recursive systematic convolutional encoders separated by an interleaver. The first encoder processes the original data while the second processes the interleaved version of data. Then, the encoded stream is randomly interleaved and mapped into complex symbols of a 2^Q quadrature amplitude modulation (QAM) constellation, where Q is the number of bits per symbol. The resulting sequence of symbols is mapped into M dimensional symbol vectors

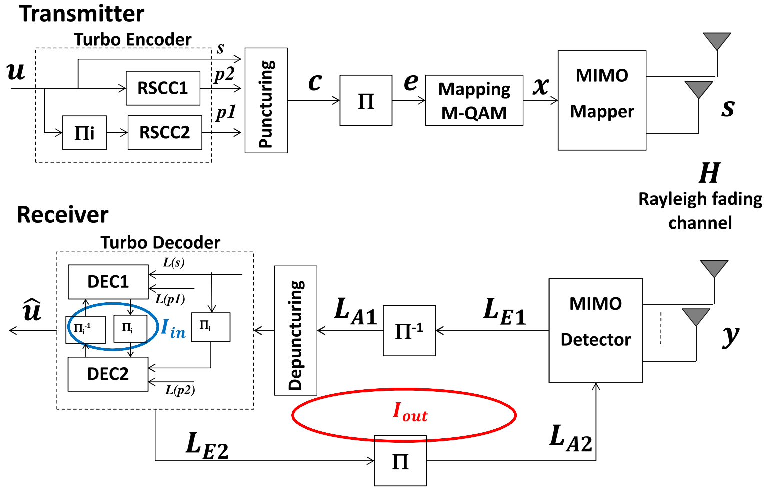


Figure 1 MIMO system block diagram using bit-interleaved coded modulation with iterative detection and decoding.

$\mathbf{s} \in 2^{Q \cdot M}$ using either space-time block coding (STBC) schemes or spatial multiplexing (SM) schemes. Herein, the SM-based MIMO system is considered without loss of generality. It is assumed that the channels experience independent Rayleigh fading, and the transmitter does not require any channel state information (CSI). The transmit power is normalized so that $\mathbb{E}\left\{\mathbf{s}\mathbf{s}^H\right\} = E_s/M\mathbf{I}_M$, where $\mathbf{I}_M$ is the $M \times M$ identity matrix. The transmission information rate is $R_c \cdot M \cdot Q$ bits per channel use. The received vector $\mathbf{y} = [y_1, y_2, ..., y_N]^T$ can be represented by:

$$\mathbf{y} = \mathbf{H}\mathbf{s} + \mathbf{n}, \tag{1}$$

where $\mathbf{H}$ is an $N \times M$ channel matrix, assumed to be perfectly known at the receiver, with independent elements h_{ij} of zero mean and unit variance complex Gaussian random variables; $\mathbf{n} = [n_1, n_2, ..., n_N]^T$ is an independent and identically distributed (i.i.d.) additive white Gaussian noise (AWGN) vector with zero mean and σ_n^2 variance $(N_0 = \sigma_n^2)$.

2.2 Turbo principle

At the receiver, an iterative detection-decoding process based on the turbo principle is applied as shown in Figure 1. The MIMO detector and the channel decoder can be viewed as serially concatenated blocks. The MIMO detector can employ MAP algorithm or other sub-optimal algorithms like LSD, STS-SD, K-Best decoder, or MMSE-IC. When MIMO equalizer is performed, the iterative process is referred to as turbo equalization [3]. The MIMO detector takes the received symbol vector $\mathbf{y}$ and the *a priori* information L_{A1} of the coded bits from the channel decoder and computes the extrinsic information L_{E1}. This extrinsic information is de-interleaved and serves as the *a priori* information L_{A2} for the turbo decoder. The turbo decoder computes the extrinsic information L_{E2} which is consequently re-interleaved and fed back to the MIMO detector as the *a priori* information L_{A1}. The turbo decoding is performed by two soft-input soft-output (SISO) component decoders that exchange soft information about their data sub-stream. The SISO component decoder can be implemented using BCJR (Bahl, Cocke, Jelinek, and Raviv), Log-MAP algorithms. Each component decoder takes systematic or interleaved information, the corresponding parity information, and the *a priori* information from the other decoder to compute the extrinsic information. This extrinsic information is used by the other decoder as the *a priori* information after interleaving or de-interleaving. In our iterative process, we denote by I_{out} the number of outer iterations (MIMO detector - turbo decoder) and by I_{in} the number of inner iterations (within each turbo decoder).

In the sphere decoder case, the complex model system in Equation 1 is usually transformed into an equivalent real system model as follows:

$$\begin{bmatrix} \text{Re}\,(\mathbf{y}) \\ \text{Im}\,(\mathbf{y}) \end{bmatrix} = \begin{bmatrix} \text{Re}\,(\mathbf{H}) & -\text{Im}\,(\mathbf{H}) \\ \text{Im}\,(\mathbf{H}) & \text{Re}\,(\mathbf{H}) \end{bmatrix} \begin{bmatrix} \text{Re}\,(\mathbf{s}) \\ \text{Im}\,(\mathbf{s}) \end{bmatrix} + \begin{bmatrix} \text{Re}\,(\mathbf{n}) \\ \text{Im}\,(\mathbf{n}) \end{bmatrix}, \tag{2}$$

where Re (.) and Im (.) represent the real and the imaginary parts of the variables, respectively. In this equivalent real system model, the QAM constellation can be viewed as two PAM constellations, and the matrix dimension is hence doubled. In [29], the real model was revealed to be more efficient for the implementation of the sphere decoder and it will be used for the system model in this paper.

3 Soft-input soft-output MIMO detection

This section firstly reviews the optimal MAP detection algorithm. Then, it discusses several sub-optimal detection algorithms. These algorithms can be divided into two main families, namely the tree-search-based detection and the interference cancellation-based detection. The tree-search-based detection generally falls into two main categories, namely depth-first search and breath-first search. The classical sphere decoding is a depth-first approach, while the K-Best decoding and fixed sphere decoding are commonly seen as breath-first approaches. We then present interference cancellation-based detection that performs MMSE filtering in combination with the soft symbol-aided interference cancellation. The interference cancellation can be carried out either in a successive way or in a parallel way. Consequently, the LC-K-Best decoder is also introduced [40].

3.1 Maximum *a posteriori* probability (MAP) detection

The MAP algorithm uses an exhaustive search over all $2^{Q\cdot M}$ possible symbol combinations to compute the exact *a posteriori* probability of each bit. Such probability is usually expressed in terms of log-likelihood ratio (LLR). The sign of LLR value determines the binary decision about the corresponding bit, while its magnitude indicates the reliability of the decision. More concretely, LLR of the b^{th} bit of the i^{th} symbol, $x_{i,b}$, can be computed as:

$$L\left(x_{i,b}\right) = \log\frac{P(x_{i,b}=+1|\mathbf{y})}{P(x_{i,b}=-1|\mathbf{y})} = \log\frac{\sum_{s\in\chi_{i,b}^{+1}} p(\mathbf{y}|\mathbf{s})P(\mathbf{s})}{\sum_{\mathbf{s}\in\chi_{i,b}^{-1}} p(\mathbf{y}|\mathbf{s})P(\mathbf{s})}, \tag{3}$$

where $\chi_{i,b}^{+1}$ and $\chi_{i,b}^{-1}$ denote the sets of symbol vectors corresponding to the i^{th} symbol and having the b^{th} bit of the symbol equal to $+1$ and -1 (representing a logical 1 and a logical 0), respectively. $p(\mathbf{y}|\mathbf{s})$ is the conditioned probability density function given by:

$$p(\mathbf{y}|\mathbf{s}) = \frac{1}{(\pi N_0)^N}\exp\left(-\frac{1}{N_0}\left\|\mathbf{y}-\mathbf{Hs}\right\|^2\right), \tag{4}$$

and $P(\mathbf{s})$ represents the *a priori* information provided by the channel decoder in the form of *a priori* LLRs:

$$\begin{aligned} L_A(x_{i,b}) &= \log\left(\frac{P\left(x_{i,b}=+1\right)}{P\left(x_{i,b}=-1\right)}\right), \quad \forall i,b \\ P(\mathbf{s}) &= \prod_{i=1}^{M} P\left(s_i\right) = \prod_{i=1}^{M}\prod_{b=1}^{Q} P\left(x_{i,b}\right). \end{aligned} \tag{5}$$

To reduce the computational complexity, LLR values can be calculated using the Max-Log-MAP approximation [22]:

$$L\left(x_{i,b}\right) \approx \frac{1}{N_0}\min_{\chi_{i,b}^{-1}}\left\{d_1\right\} - \frac{1}{N_0}\min_{\chi_{i,b}^{+1}}\left\{d_1\right\}, \tag{6}$$

where

$$\begin{aligned} d_1 &= \left\|\mathbf{y}-\mathbf{Hs}\right\|^2 - N_0\log P(\mathbf{s}) \\ &= \left\|\mathbf{y}-\mathbf{Hs}\right\|^2 - N_0\sum_{i=1}^{M}\sum_{b=1}^{Q}\log P(x_{i,b}), \end{aligned} \tag{7}$$

represents the Euclidean distance between the received vector $\mathbf{y}$ and lattice points $\mathbf{Hs}$.

Based on the *a posteriori* LLRs $L(x_{i,b})$ and the *a priori* LLRs $L_A(x_{i,b})$, the detector computes the extrinsic LLRs as $L_E(x_{i,b}) = L(x_{i,b}) - L_A(x_{i,b})$.

The exact computation of LLR using MAP detection can only be used with low-order modulations and a small number of antennas [43] because its complexity increases exponentially with respect to the number of transmit antennas and modulation orders. For example, in the case of a 2 × 2 MIMO system with 4-QAM, $2^{2\times 2} = 16$ possible solutions need to be searched. However, in the case of a 4 × 4 MIMO system with 16-QAM, there are $2^{4\times 4} = 65,536$ possible solutions. A number of MIMO detectors have been therefore proposed with reduced complexity as will be discussed in the following.

3.2 Tree-search-based detection

3.2.1 List sphere decoder (LSD)

The sphere decoder transforms the symbol detection problem into a lattice search problem [9,11,12], which can be represented by the search on a tree. Using the QR decomposition, the channel matrix $\mathbf{H}$ can be transformed into the product of two matrices $\mathbf{Q}$ and $\mathbf{R}$ ($\mathbf{H} = \mathbf{QR}$), where $\mathbf{Q}$ is a $N \times M$ unitary matrix ($\mathbf{Q}^H\mathbf{Q} = \mathbf{I}_M$), and $\mathbf{R}$ is a $M \times M$ upper triangular matrix with real-positive entries on its diagonal. With the modified received symbol vector $\tilde{\mathbf{y}} = \mathbf{Q}^H\mathbf{y}$, the distance in Equation 7 can then be computed as: $\left\|\mathbf{y}-\mathbf{Hs}\right\|^2 = \left\|\tilde{\mathbf{y}}-\mathbf{Rs}\right\|^2$. Exploiting the triangular nature of R, the Euclidean distance metric d_1 can be recursively evaluated through the accumulated partial Euclidean distance d_i with $d_{M+1} = 0$ as follows [25]:

$$\begin{aligned} d_i = d_{i+1} + &\underbrace{\left|\tilde{y}_i - \sum_{j=i}^{M} R_{i,j}s_j\right|^2}_{m_i^C} \\ &+ \underbrace{\frac{N_0}{2}\sum_{b=1}^{Q}\left(\left|L_A\left(x_{i,b}\right)\right| - x_{i,b}L_A\left(x_{i,b}\right)\right)}_{m_i^A}, \quad i = M, ..., 1. \end{aligned} \tag{8}$$

where m_i^C and m_i^A denote the channel-based partial metric and the *a priori*-based partial metric at the i^{th} level, respectively.

The sphere decoder performs a depth-first search in both forward and backward directions. A certain pruning criterion can be used to reduce the number of visited nodes. For example, a sphere radius can be set to limit the search range. The tree is represented with $M+1$ levels, where the level l corresponds to the l^{th} transmit antenna. The tree search starts at the root level with the node at level M corresponding to the symbol transmitted by the M^{th} antenna. The partial Euclidean distance d_M in Equation 8 is computed. If d_M respects the sphere radius constraint, the search continues at level $M-1$ and steps down the tree at level l until finding a valid leaf node at level 1. Subsequently, the search continues by back-tracking to previous levels to find better candidates. Figure 2a illustrates the tree search in the case of $M=2$. Thus, the candidate with the minimum Euclidean distance is chosen as an approximation of the ML solution in the hard-output sphere decoder. Whereas, in the list sphere decoder [22], a list $\mathcal{L}$ of the most promising candidates and their corresponding Euclidean distances are used in the computation of LLR values:

$$L\left(x_{i,b}\right) = \frac{1}{N_0} \min_{\mathcal{L}\cap\chi_{i,b}^{-1}} \{d_1\} - \frac{1}{N_0} \min_{\mathcal{L}\cap\chi_{i,b}^{+1}} \{d_1\}. \tag{9}$$

Although the list sphere decoder is able to approach the theoretical channel capacity, the proximity to the capacity depends on the list size. The list should be large enough to include at least one candidate for both possible hypotheses. However, using an excessively large list size will lead to an increase in computational complexity. Meanwhile, the size of the list should not be too small either. The use of limited list size causes inaccurate approximation due to missing some counter hypotheses where no entry can be found in the list for a particular bit $x_{i,b}=+1$ or -1. The frequently used solution for this problem is to set the LLR to a predefined maximum value [22,27]. Moreover, two methods were used to process the list in the iterative receiver. The first method consists of generating the list during the first iteration and using this list for subsequent iterations to update the soft information [22]. The second method updates the list at each iteration leading to further performance improvements but yielding additional computational complexity [27]. Additionally, several methods can be included for further reduction of the complexity of tree-search algorithms. The Schnorr-Euchner (SE) enumeration [10] proposed as a refinement of Fincke-Pohst (FP) enumeration extends the nodes in ascending order with respect to their Euclidean distance metrics to reduce the average complexities. Layer ordering technique allows to select most reliable symbols at a high layer using the sorted QR (SQR) decomposition [44]. The most reliable symbols are helpful for faster finding the ML solution. MMSE pre-processing might also be used for further reducing through the use of an extended channel matrix for the SQR decomposition [45]. However, this method introduces a biasing factor in the metrics which should be removed in the LLR calculation to avoid performance degradation [46].

3.2.2 Single tree-search sphere decoder (STS-SD)

One of the two minima in Equation 6 corresponds to the MAP hypothesis $\mathbf{s}^{MAP}$ while the other corresponds to the counter hypothesis. The computation of LLR can be done as:

$$L\left(x_{i,b}\right) = \begin{cases} \frac{1}{N_0}\left(d_{i,b}^{\overline{\mathrm{MAP}}} - d^{\mathrm{MAP}}\right), & \text{if } x_{i,b}^{\mathrm{MAP}} = +1 \\ \frac{1}{N_0}\left(d^{\mathrm{MAP}} - d_{i,b}^{\overline{\mathrm{MAP}}}\right), & \text{if } x_{i,b}^{\mathrm{MAP}} = -1. \end{cases}$$

$$d^{\mathrm{MAP}} = \left\|\tilde{\mathbf{y}} - \mathbf{R}\mathbf{s}^{\mathrm{MAP}}\right\|^2 - N_0 P\left(\mathbf{s}^{\mathrm{MAP}}\right), \tag{10}$$

$$d_{i,b}^{\overline{\mathrm{MAP}}} = \min_{s\in\chi_{i,b}^{\overline{\mathrm{MAP}}}} \left\{\left\|\tilde{\mathbf{y}} - \mathbf{R}\mathbf{s}\right\|^2 - N_0 P\left(\mathbf{s}\right)\right\}, \tag{11}$$

$$\mathbf{s}^{\mathrm{MAP}} = \arg\min_{s\in 2^{Q\cdot M}} \left\{\left\|\tilde{\mathbf{y}} - \mathbf{R}\mathbf{s}\right\|^2 - N_0 P\left(\mathbf{s}\right)\right\}, \tag{12}$$

where $\chi_{i,b}^{\overline{\mathrm{MAP}}}$ denotes the bit-wise counter hypothesis of the MAP hypothesis, which is obtained by searching over all the solutions with the b^{th} bit of the i^{th} symbol opposite to the current MAP hypothesis. Originally, the MAP hypothesis and the counter hypotheses can be found through repeating the tree search [47] that requires a large computational complexity cost. To overcome this, the single tree-search algorithm [24,25] was developed to compute all the LLRs concurrently. The d^{MAP} metric and the corresponding $d_{i,b}^{\overline{\mathrm{MAP}}}$ metrics are updated through one tree search. The basic idea of STS-SD is to

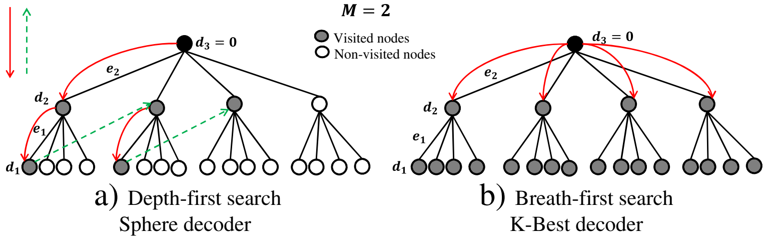


Figure 2 Tree-search strategies. (a) Depth-first search sphere decoder. **(b)** Breath-first search *K*-Best decoder.

search the sub-tree originating from a given node if the Euclidean distance leads to an update of either d^{MAP} or at least one of $d_{i,b}^{\overline{\mathrm{MAP}}}$. Through the use of extrinsic LLR clipping method, the STS-SD algorithm can be tunable between the MAP performance and hard-output performance. Channel matrix regularization and run time constraint may also be used in STS-SD to reduce the decoding complexity at the price of performance degradation. The implementations of STS-SD have been reported in [31,32].

3.2.3 K-Best decoder

K-Best decoder is a breath-first search-based algorithm. Starting from the root node at level $M+1$ with $d_{M+1}=0$, K-Best decoder expands each of the K survival paths to all possible children nodes in the constellation and computes their corresponding partial Euclidean distances. Then, the K-Best decoder sorts all $K\sqrt{2^Q}$ distances and keeps only the K nodes with minimum Euclidean distances until reaching the leaf nodes as illustrated in Figure 2b. The candidate with the minimum Euclidean distance is chosen as an approximate of the ML solution. Whereas, a list of the most likely candidates is retained in the case of iterative receiver. We note that the candidate list does not necessarily correspond to the lowest Euclidean distance.

The major drawbacks of K-Best decoder are the expansion and the sorting operations that are very time consuming. Several proposals have been drawn in the literature to approximate the sorting operations such as relaxed sorting [48], local sorting and merging, and distributed sorting [49]; or even to avoid sorting using on demand expansion scheme [50]. Moreover, K-Best decoder suffers similarly as LSD from missing counter hypothesis problem due to the limited list size. Numerous approaches have been proposed to address this problem such as smart candidates adding [51], bit flipping [52], path augmentation and LLR clipping [22,27].

3.2.4 Fixed sphere decoder (FSD)

Fixed sphere decoder is a breath-first search algorithm proposed to further reduce the complexity of K-Best decoder. It performs two stages of tree search. A full search is performed in the first T levels by expanding all branches per node. Then, a single search is performed in the remaining $M-T$ levels expanding only one branch per node. The parameters T are chosen such as $(N-M)(T+1)+(T+1)^2 > N$ in order to provide an asymptotical ML performance. We note that in FSD, the columns of $\mathbf{H}$ are ordered such as in the first T levels, the signal has the largest post-processing noise amplification. In the soft-output FSD proposed in [30], the search is performed not only to find the ML solution but also to find a set of candidates around the ML solution in order to compute the LLR of all bits. Therefore, a subset S is first chosen, then the ML solution of the subset is used to generate a subset S'. The combined list $S \cup S'$ is finally used to compute an approximation of the extrinsic LLR. Efficient SISO FSD implementations have been proposed in [35,36].

3.3 Interference cancellation (IC)-based detection

Interference cancellation-based detection is commonly used in combination with MMSE liner filtering. In the case of MIMO iterative receiver, the MIMO equalizer and the channel decoder exchange soft information according to the turbo equalization principle [4,6]. The MIMO equalizer produces an equalized symbol vector $\tilde{\mathbf{s}}$ deduced from received signal $\mathbf{y}$. The soft estimated symbol vector $\hat{\mathbf{s}}$ is used to cancel the interference terms in the received signal. The interference cancellation can be carried out either in a successive way as in VBLAST [8] or in a parallel way as in MMSE-IC [20,21].

3.3.1 Minimum mean square error-interference cancellation (MMSE-IC) equalizer

MMSE-IC equalizer can be performed using two filters [20]. The first filter $\mathbf{p}_k$ is applied to the received vector $\mathbf{y}$, and the second filter $\mathbf{q}_k$ is applied to the estimated vector $\hat{s}$ as shown in Figure 3. The equalized symbol $\tilde{\mathbf{s}}_k$ can be written as:

$$\tilde{s}_k = \mathbf{p}_k^H \mathbf{y} - \mathbf{q}_k^H \hat{\mathbf{s}}_k \quad \text{with} \quad k \in [1, M], \tag{13}$$

where $\hat{\mathbf{s}}_k$ denotes the estimated vector given by the previous iteration with the k^{th} symbol omitted: $\hat{\mathbf{s}}_k = \left[\hat{s_1}...\hat{s}_{k-1} \quad 0 \quad \hat{s}_{k+1}...\hat{s}_M\right]$. $\hat{s}_k$ is calculated by the soft mapper as: $\hat{s}_k = \mathbb{E}\left[s_k\right] = \sum_{s\in 2^Q} sP\left(s_k = s\right)$ [53].

The filters $\mathbf{p}_k$ and $\mathbf{q}_k$ are optimized under the MMSE criterion:

$$\left(\mathbf{p}_k^{opt}, \mathbf{q}_k^{opt}\right) = \arg\min_{\mathbf{p}_k, \mathbf{q}_k} \mathbb{E}\left\{|s_k - \tilde{s}_k|^2\right\}, \tag{14}$$

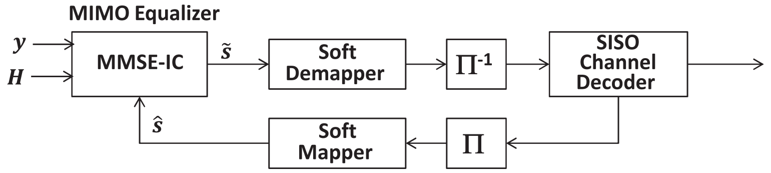


Figure 3 **Turbo equalization with MMSE-IC equalizer.**

and can be computed by [4]:

$$\mathbf{p}_k^{opt} = \sigma_s^2 \left[\mathbf{H}\mathbf{V}_k\mathbf{H}^H + N_0\mathbf{I}_N\right]^{-1} \mathbf{h}_k, \tag{15a}$$

$$\mathbf{q}_k^{opt} = \mathbf{H}^H \mathbf{p}_k^{opt}, \tag{15b}$$

where σ_s^2 is the power of the received signal, $\mathbf{h}_k$ denotes the k^{th} column of the channel matrix $\mathbf{H}$, and $\mathbf{V}_k$ is a diagonal matrix that depends on the residual errors of each estimated symbols:

$$\mathbf{V}_k = \sigma_s^2 e_k e_k^T + \sum_{i=1, i\neq k}^{M} \nu_i^2 e_i e_i^T \tag{16}$$

with ν_k^2 defined as:

$$\nu_k^2 = \mathbb{E}\left\{\left|s_k - \hat{s}_k\right|^2 |L_A\right\}, \tag{17a}$$

$$\nu_k^2 = \sum_{s\in 2^Q} |s|^2 P\left(\hat{s}_k = s\right) - \left|\hat{s}_k\right|^2. \tag{17b}$$

At the first iteration, since no *a priori* information is available, the equalization process is reduced to the classical MMSE solution:

$$\tilde{s}_k = \left[\mathbf{H}^H\mathbf{H} + \frac{\sigma_n^2}{\sigma_s^2}\mathbf{I}_M\right]^{-1} \mathbf{H}^H\mathbf{y}. \tag{18}$$

The equalized symbols $\tilde{s}_k$ are associated with a bias factor β_k in addition to some residual noise plus interferences η_k:

$$\tilde{s}_k = \beta_k s_k + \eta_k \tag{19}$$

These equalized symbols are then used by the soft demapper to compute the LLR values using the Max-Log-MAP approximation [53]:

$$L\left(x_{i,b}\right) = \frac{1}{\sigma_{\eta_k}^2}\left(\min_{s\in\chi_{i,b}^{-1}} |\tilde{s}_k - \beta_k.s_k|^2 - \min_{s\in\chi_{i,b}^{+1}} |\tilde{s}_k - \beta_k.s_k|^2\right). \tag{20}$$

MMSE-IC equalizer requires M matrix inversions for each symbol vector. For this reason, several approximations of MMSE-IC were proposed.

The first approximation of MMSE-IC consists of replacing the variable ν_k^2 by its mean $\nu^2 = E\left(\nu_k^2\right) = \sigma_s^2 - \sigma_{\hat{s}}^2$ [4]. Hence, one matrix inversion is computed for all symbols. This approximation is denoted as MMSE-IC1. MMSE-IC1 algorithm reduces significantly the complexity of computing the filter coefficients. However, the coefficients of the equalizer must be recomputed at each iteration.

A second approximation denoted as MMSE-IC2 [20] assumes a perfect estimation of transmitted symbols ($\sigma_{\hat{s}}^2 = \sigma_s^2$) to overcome the matrix inversion at each iteration.

In [21], a low-complexity approach of MMSE-IC is described by performing a single matrix inversion without performance loss. We refer to this algorithm as LC-MMSE-IC.

3.3.2 Successive interference cancellation (SIC) equalizer

The SIC-based detector was initially used in VBLAST systems. In VBLAST architecture [8], a successive cancellation step and interference nulling step are used to detect the transmitted symbols. However, this method suffers from error propagation. Several methods have been proposed to reduce this problem by taking decision errors into account [19,54]. An improved VBLAST for iterative detection and decoding is described in [54]. At the first iteration, an enhanced VBLAST which takes decision errors into account is employed. When the *a priori* LLRs are available from the channel decoder, soft symbols are computed by a soft mapper and are used in the interference cancellation. To describe the enhanced VBLAST algorithm, we assume that the detection order has been made according to the optimal detection order [8]. We define $\hat{\mathbf{s}}_{k-1}$ as $\left[\widehat{s_1} \quad \hat{s}_2 \quad \ldots \quad \hat{s}_{k-1}\right]$, and $\mathbf{H}_{i:j}$ as $\left[\mathbf{h}_i \quad \mathbf{h}_{i+1} \quad \ldots \quad \mathbf{h}_j\right]$, where $\mathbf{h}_i$ denotes the i^{th} column of $\mathbf{H}$. At the step k, the pre-detected symbol vector $\hat{s}_{k-1}$ until step $k-1$ is cancelled out from the received signal:

$$\mathbf{y}_k = \mathbf{y} - \mathbf{H}_{1:k-1}\hat{\mathbf{s}}_{k-1}. \tag{21}$$

In the conventional VBLAST algorithm, the hard estimated symbol vector $\mathbf{s}_{k-1}$ is used in the cancellation step; then the MMSE filtering is applied in the nulling step. The enhanced VBLAST algorithm uses the soft estimated symbol vector $\hat{\mathbf{s}}_{k-1}$ and a nulling matrix $\mathbf{W}_k$ based on the MMSE criterion that takes decision errors into account. $\mathbf{W}_k$ can be expressed by [19,54]:

$$\mathbf{W}_k = \sigma_s^2 \left(\mathbf{H}\Sigma_k\mathbf{H}^H + N_0 I_N\right)^{-1} \mathbf{h}_k, \tag{22}$$

where Σ_k is the decision error covariance matrix defined as:

$$\Sigma_k = \sum_{i=1}^{k-1} \epsilon_i^2 e_i e_i^T + \sum_{i=k}^{M-k+1} \sigma_s^2 e_i e_i^T, \epsilon_i^2 = \mathbb{E}\left\{\left|s_i - \hat{s}_i\right|^2 |\hat{s}_{i-1}\right\}. \tag{23}$$

The estimated symbol $\tilde{s}_k$ can be expressed as:

$$\tilde{s}_k = \mathbf{W}_k^H \mathbf{y}_k = \beta_k s_k + \eta_k. \tag{24}$$

A soft demapper is then used to compute the LLRs as in Equation 20. We refer to this algorithm as improved VBLAST (I-VBLAST) in the following.

3.4 Low-complexity K-Best (LC-K-Best) decoder

The classical K-Best decoder computes $K\sqrt{2^Q}$ Euclidean distances. Then, a sorting operation is done to choose the

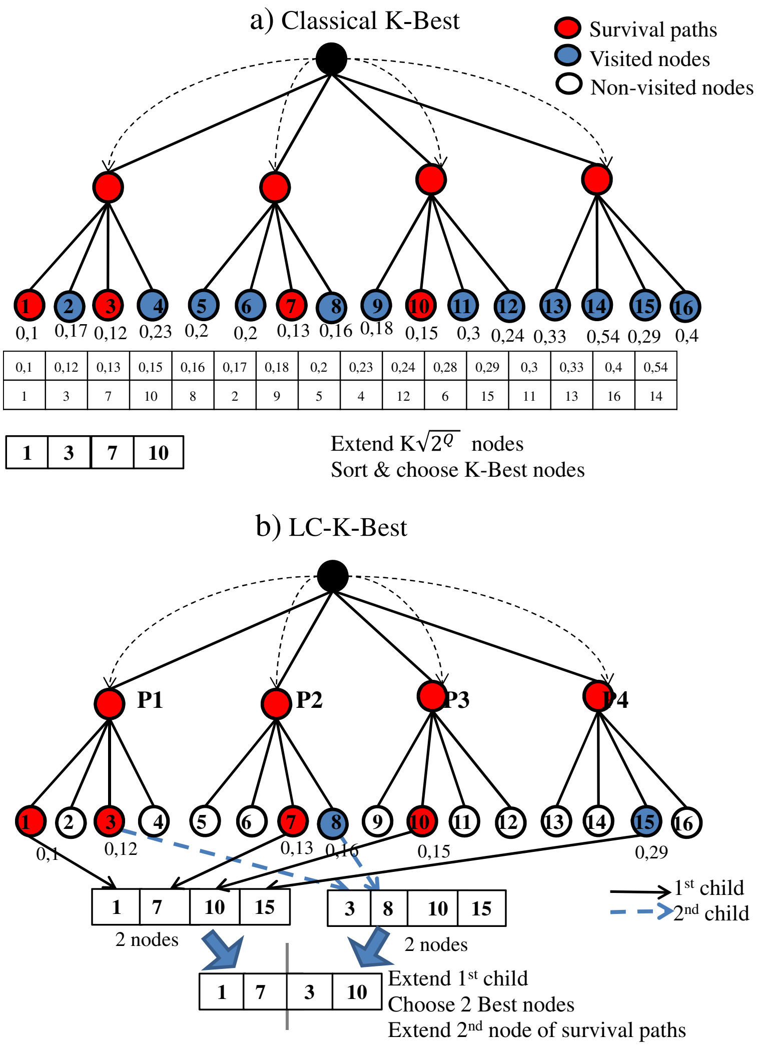
K best candidates as illustrated in Figure 5 with an example of $K = 4$. The LC-K-Best decoder recently proposed in [40] uses two improvements over the classical K-Best decoder for the sake of lower complexity and latency.

Simplified hybrid enumeration

The first improvement simplifies the hybrid enumeration of the constellation points in real system model when the *a priori* information is incorporated into the tree search using two look-up-tables (LUTs). Hybrid enumeration was initially proposed in [55] for soft-input sphere decoder in complex system model. It consists of separating the partial metric into two metrics: the channel metric and the *a priori* metric. To simplify the enumeration, we consider two LUTs. One LUT is used for channel metric m_i^C and the other LUT is used to store the *a priori* metric m_i^A. The enumeration is approximated through the orthogonality of these two metrics. Figure 4 illustrates an example of the enumeration strategy. First, the constellation points are enumerated according to m^C and m^A and stored in the LUTs. Then, the smallest Euclidean distances of m^C and m^A are compared (S2 and S3). The one which has the minimum distance (S2 in m^C) is chosen as the first point. Then, the first point in m^A (S3) is compared to the next point in m^C (S1). Since S3 has a lower distance, it is considered as the second point and so on.

Relaxed on-demand expansion

The second improvement is to use a relaxed on-demand expansion that reduces the need of exhaustive expansion and sorting operations. The on-demand expansion was proposed in [50] for hard-output decoder. It consists in expanding the first children of parent nodes and choosing



0,1	0,12	0,13	0,15	0,16	0,17	0,18	0,2	0,23	0,24	0,28	0,29	0,3	0,33	0,4	0,54
1	3	7	10	8	2	9	5	4	12	6	15	11	13	16	14

Figure 4 Classical K-Best Versus LC-K-Best. (a) Classical K-Best. **(b)** LC-K-Best.

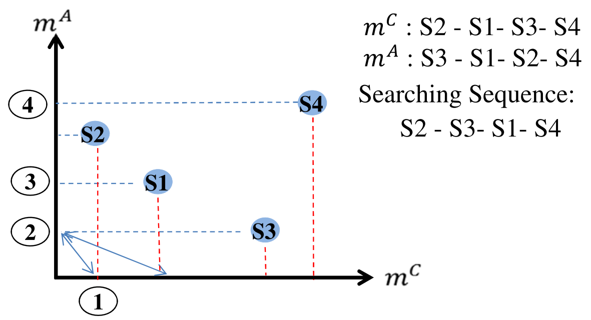


Figure 5 Enumeration strategy based on m^c and m^A.

one minimum between these children. Then, the survival path expands the next child. In our approach, a portion A of the first children is chosen. Then, the corresponding parents expand their next children. This operation is repeated to get K best nodes. The number of the first children A is chosen in order to allow a parent node to extend all its possible children nodes depending on the constellation and on the total number K of retained solutions. Figure 5 shows an example with $K = 4$ and $A = 2$. All parent nodes at the first level expand their first children. The two children that have the smaller Euclidean distances (nodes 1 and 7) are retained. Then, the corresponding parent nodes (P1 and P2) expand their next children (nodes 3 and 8). The distance is compared, and the two nodes (3 and 10) having the lowest distances are retained to get 4 best candidates.

It has been shown in [40] that LC-K-Best decoder achieves almost the same performance as the classical K-Best decoder with different modulations. It was shown that the computational complexity in terms of the number of visited nodes can be significantly reduced specially in the case of high-order modulations.

4 Convergence analysis

The extrinsic information transfer (EXIT) chart proposed in [41] is an effective tool to analyze the convergence of the iterative process. It describes the exchange of the mutual information in the iterative process in order to determine the required number of iterations, the convergence threshold, and the average decoding trajectory. Two iterative processes (inside the turbo decoder, and between the MIMO detector and the turbo decoder) are involved in iterative MIMO turbo code receivers.

In our analysis, we separately study the convergence of turbo decoding and MIMO detection. We denote by I_{A1} and I_{A2} the *a priori* mutual information at the inputs of the MIMO detector and the turbo decoder, respectively. I_{E1} and I_{E2} denote their corresponding extrinsic mutual information at the outputs. We model the *a priori* information L_A by applying independent Gaussian random variable n_A with zero mean and σ_A^2 variance such as $\mu_A = \sigma_A^2/2$ [41]:

$$L_A = \mu_A x + n_A. \tag{25}$$

The mutual information I_x (I_A or I_E) can be computed by the means of Monte Carlo simulation using the probability density function p_{L_x}:

$$I_x = \frac{1}{2} \sum_{x=-1,1} \int_{-\infty}^{+\infty} p_{L_x}(L_x|x) \log \times \frac{2p_{L_x}(L_x|x)}{p_{L_x}(L_x|-1)\, p_{L_x}(L_x|+1)} dL_x. \tag{26}$$

A simple approximation of the mutual information is used in our analysis:

$$I_x \approx 1 - \frac{1}{L} \sum_{n=0}^{L-1} \log\left(1 + \exp\left(-xL_x\right)\right), \tag{27}$$

where L is the number of transmitted bits, and L_x is the LLR associated with the bit x ∈ $\{-1, +1\}$. At the beginning, the *a priori* mutual information $I_{A1} = 0$ and $I_{A2} = 0$. Then, the extrinsic mutual information I_{E1} of the MIMO detector becomes the *a priori* mutual information I_{A2} of the turbo decoder and vice versa (i.e, $I_{E1} = I_{A2}$ and $I_{E2} = I_{A1}$). Moreover, when the tunnel is opened, the exchange of the extrinsic information can be visualized as a 'zig-zag' decoding trajectory in the EXIT chart. Jumping from one curve to the other to reach a mutual information near to one determines the convergence point and the required number of iterations.

For our convergence analysis, a 4 × 4 MIMO system with 16-QAM constellation and turbo decoder ($R_c = 1/2$) is considered. Figure 6 shows the EXIT chart of the overall system for different E_b/N_0 values and several MIMO detectors (STS-SD, LC-K-Best, MMSE-IC, and MMSE-IC1). As the I-VBLAST detector performs successive interference cancellation at the first iteration and parallel interference cancellation of the soft estimated symbols for the rest iterations, it is less intuitive to present its convergence in the EXIT chart. Therefore, the convergence analysis of VBLAST is not given in this work.

We notice in Figure 6 that the characteristic of the turbo decoder is independent of E_b/N_0 values. At a low signal to noise ratio (e.g, E_b/N_0 = 1 dB), with one inner iteration, the MIMO detector and the turbo decoder transfer characteristics intersect at low mutual information (≤ 0.4); the tunnel is blocked for all detection algorithms. Therefore, the performance cannot be improved through iterations. With the increase of E_b/N_0, the transfer characteristics of the MIMO detectors are shifted upward, and the tunnel between the two characteristics is then open allowing the iterative process to improve the performance of the system. By comparing the characteristics of STS-SD, LC-K-Best decoder, and MMSE-IC equalizers, we notice that

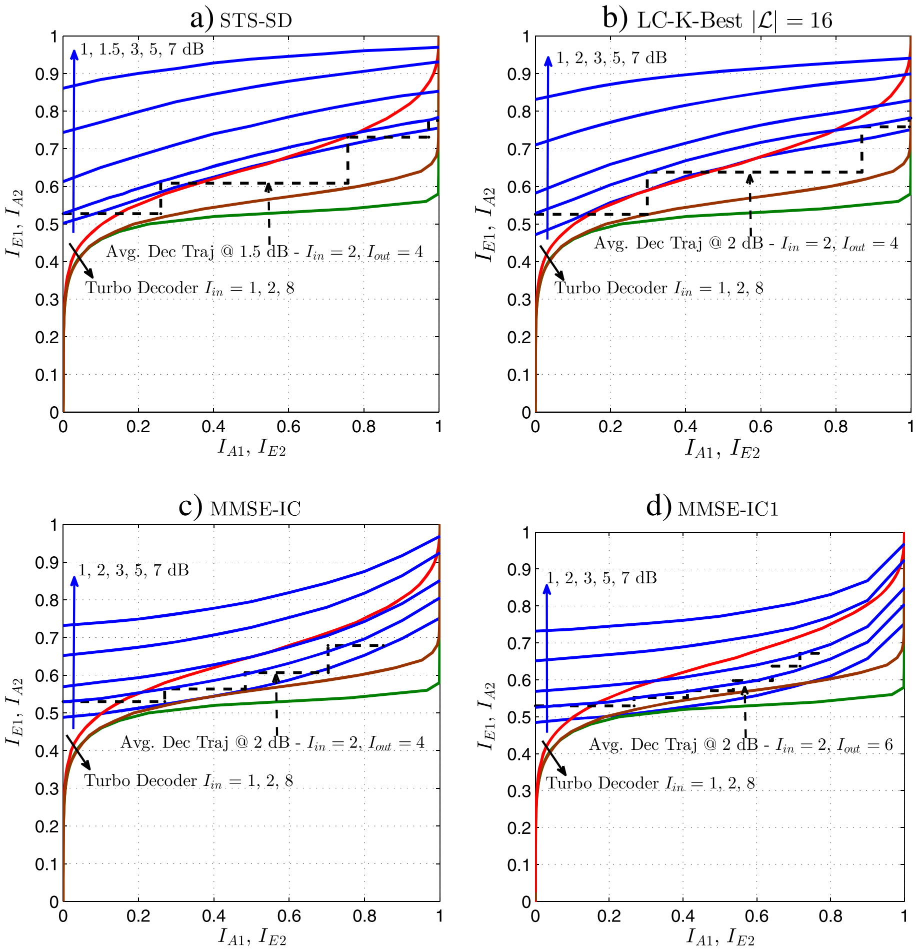


Figure 6 EXIT chart for the turbo decoder (R_c = 1/2) and MIMO detectors. a) STS-SD, **b)** LC-K-Best, **c)** MMSE-IC, **d)** MMSE-IC1, at different E_b/N_0 values (1, 1.5, 2, 3, 5, 7 dB) in a 4×4 MIMO system using 16-QAM.

STS-SD has a larger mutual information at its output. LC-K-Best decoder has slightly less mutual information than STS-SD. MMSE-IC and MMSE-IC1 show low mutual information levels at their outputs compared to other algorithms when $I_{A1} < 0.85$. While for $I_{A1} > 0.85$, the extrinsic mutual information is similar to others.

Furthermore, as shown in Figure 6a, in the case of STS-SD, at $E_b/N_0 = 1$ dB with eight iterations inside the turbo decoder, the tunnel is open and hence applying outer iterations will lead to the intersection of the two characteristic curves at moderate mutual information level. This intersection indicates that the BER performance cannot be further improved with more iterations. At $E_b/N_0 = 1.5$ dB, the tunnel is larger, and three outer iterations are sufficient to converge towards higher mutual information leading to lower BER. However, when performing two inner iterations inside the turbo decoder, the convergence point can be attained by performing four outer iterations. Similarly, LC-K-Best decoder in Figure 6b shows an equivalent performance but slightly higher E_b/N_0 is required. The convergence speed of LC-K-Best decoder is a bit lower than STS-SD, which requires more iterations to get the same performance. The reason is mainly due to the unreliability of the LLRs caused by the small list size ($\mathcal{L} = 16$). In the cases of MMSE-IC and MMSE-IC1 (Figure 6c and Figure 6d, respectively), the characteristics present a lower mutual information than the LC-K-Best decoder when $I_{A1} < 0.85$. Therefore, an equivalent performance can be obtained at higher E_b/N_0 or by performing more iterations.

In addition, the average decoding trajectory resulting from the free-run iterative detection-decoding simulations is illustrated in Figure 6 at $E_b/N_0 = 1.5$ or 2 dB for four or six outer iterations between the MIMO detector and the turbo decoder and two inner iterations. The decoding trajectory closely matches the characteristics in the case of STS-SD and LC-K-Best decoder. The small mismatch is due to the limited interleaver length. In the case of MMSE-IC and MMSE-IC1 equalizers, the decoding trajectory diverges from the characteristics for high mutual information because the equalizer uses the *a posteriori* information to compute soft symbols instead of the extrinsic information.

In conclusion, the iterative process with large tunnel leads to faster convergence. If only one inner iteration is performed for each outer iteration, more than four outer iterations are required since the turbo decoder requires at least six to eight iterations to converge. If otherwise, more inner iteration is carried out (e.g., two), four outer iterations in the global loop are sufficient to reach the convergence threshold. The best trade-off scheduling of the required number of iterations is therefore I_{out} iterations in the outer loop and a total of eight iterations inside the turbo decoder distributed across these I_{out} iterations.

5 Performance comparisons

In this section, we compare the performance of different MIMO detectors namely STS-SD, LC-K-Best decoder, MMSE-IC, and I-VBLAST equalizers. The simulations are based on a 4 × 4 SM MIMO system, QAM constellation with Gray mapping, and Rayleigh fading channel model. The Rayleigh fading coefficients are generated randomly with zero mean and unit variance. The 1/3 rate turbo encoder specified in 3GPP LTE is used in the simulations. Puncturing is performed in the rate matching module to achieve a coding rate R_c. The transmitted frame consists of 2,048 bits. A random interleaver of a size 2,048 is therefore considered. Table 1 summarizes the principle parameters for the simulations. The performance is measured in terms of bit error rate (BER) with respect to SNR per information bit E_b/N_0 defined as:

$$\frac{E_b}{N_0} = \frac{E_s}{N_0} + 10\log_{10}\frac{1}{R_c QM}. \quad [dB] \tag{28}$$

Table 1 Simulation parameters

MIMO system	**4 × 4 Spatial multiplexing**
Channel type	Flat Rayleigh fading
Detector	Single tree search (STS-SD)
	LC-K-Best, $K = 16$ (16-QAM), $K = 32, 64$ (64-QAM),
	I-VBLAST, MMSE-IC, MMSE-IC1
Channel decoder	LTE turbo code K = 4 $[13, 15]_o$
	$R_c = 1/2, 3/4$
	Block Length $L = 2{,}048$ bits
Modulation 2^Q-QAM	16-QAM, 64-QAM
	Gray mapping
Interleaver	Random, size = 2,048

For each E_b/N_0 value, BER is obtained with at least 200 errors. A maximum number of 10,000 frames is transmitted which is largely sufficient for obtaining a BER level of 10^{-5}.

The performance of these detection algorithms is compared with different system configurations. In the first configuration, an original schedule that performs eight inner iterations inside the turbo decoder for each outer iteration is considered. The second configuration uses a new schedule that performs a total number of eight iterations inside the turbo decoder distributed equally across the outer iterations. Such schedule is chosen based on the convergence behavior of the iterative processing. For this configuration, two schemes are considered. First scheme performs two outer iterations, each with four inner iterations. The other scheme performs four outer iterations, each with two inner iterations. These schemes are considered to investigate the impact of the number of outer and inner iterations on the performance and the complexity of the system. Moreover, in the case of MMSE-IC equalizer, a previous schedule [20] that performs only one inner iteration for each outer iteration is considered. This configuration can be adopted in a low-complexity detector like the equalizer used in [20]. In the case of tree-search-based algorithms, such configuration with eight outer iterations requires a high computational complexity in the MIMO detector.

Figure 7 shows the BER performance of the first configuration with $I_{\text{in}} = 8$ inner iterations and I_{out} = 1, 2, 4, or 8 outer iterations. STS-SD is used without any simplifications which allows us to consider it as a reference close to MAP performance. At the first iteration $I_{\text{out}} = 1$ (Figure 7a), since no *a priori* information is available at the equalizer, a classical MMSE equalization is performed. For $I_{\text{out}} = 2, 4, 8$, an interference canceller can be carried out efficiently. Therefore, I-VBLAST, MMSE-IC equalizer, and its approximation (MMSE-IC1, MMSE-IC2) are considered. From Figure 7a,b, it can be seen that the performance of the system is improved through iterations by about 1.5 dB at a BER level of 1×10^{-5} with all MIMO detection algorithms. At the first iteration, STS-SD outperforms LC-K-Best decoder by about 0.5 dB (Figure 7a). However, this gap is reduced to 0.2 dB at a BER level of 1×10^{-4} with four outer iterations (Figure 7b). In addition, Figure 7a shows that I-VBLAST outperforms LC-K-Best decoder by 0.2 dB and 0.1 dB at a BER level of 1×10^{-5} for $I_{\text{out}} = 1$ and $I_{\text{out}} = 2$, respectively. Moreover, LC-K-Best decoder slightly outperforms MMSE equalizer by about 0.1 dB at a BER level of 1×10^{-5}. However, MMSE-IC and I-VBLAST performances are close to LC-K-Best decoder with four outer iterations (Figure 7b). MMSE-IC1 equalizer shows performance degradation of 0.4 dB compared to MMSE-IC equalizer and LC-K-Best decoder. Whereas, MMSE-IC2 presents a degradation of 0.8 dB at

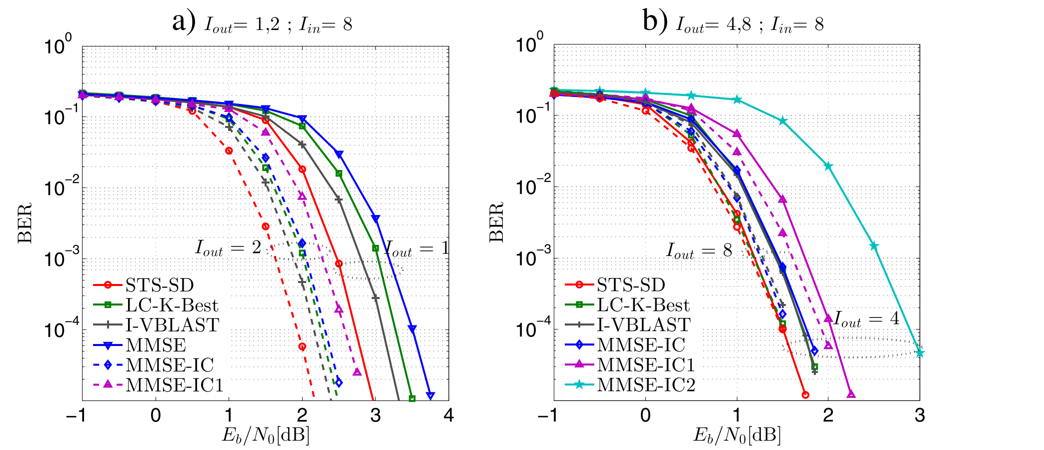


Figure 7 BER performance of a 4 × 4 coded MIMO system with 16-QAM using several MIMO detectors. (STS-SD, LC-***K***-Best, I-VBLAST, MMSE-IC, and MMSE-IC1), **(a)** $I_{\text{out}} = 1, 2$; $I_{\text{in}} = 8$ and **(b)** $I_{\text{out}} = 4, 8$; $I_{\text{in}} = 8$. Turbo code with $R_c = 1/2$ and $L = 2,048$ is used.

a BER level of 1×10^{-4} compared to MMSE-IC1. Furthermore, Figure 7b shows that no significant improvement can be achieved after four outer iterations. This improvement at a BER level of 1×10^{-4} is negligible with eight iterations in the case of STS-SD and less than 0.2 dB with other detectors.

Figure 8 illustrates the BER performance of STS-SD and LC-K-Best decoder with four outer iterations and eight turbo decoder iterations distributed across these four outer iterations. We see that the order of inner iterations has an impact on the performance of the system. For example, performing five inner iterations inside the turbo decoder in the first outer iteration, then one iteration in the remaining outer iterations shows a degradation about $0.2 \sim 0.25$ dB compared to the case when five inner iterations are performed in the last outer iteration. This is explained by the fact that through iterative process, the turbo decoder gets more reliable information at its input which allows faster convergence. However, this scheduling is not the optimal since the turbo decoder does not benefit from the iterations until the end. By varying the order of inner iterations, we find that a good solution is to perform two inner iterations inside the turbo decoder for each of the four outer iterations.

Figure 9 illustrates the performance of MIMO detectors with the second system configuration using two different schemes. Comparing Figure 9a and Figure 9b, it can be seen that the second scheme with $I_{\text{out}} = 2$ and $I_{\text{in}} = 4$ presents a degradation of about 0.5 dB compared to the first scheme with $I_{\text{out}} = 4$ and $I_{\text{in}} = 2$. Moreover, Figure 9a shows that the first scheme presents a degradation of less than 0.1 dB at a BER level of 2×10^{-5} with all detection algorithms compared to the scheme that repeats eight inner iterations at each outer iteration in Figure 7b. By comparing the algorithms, LC-*K*-Best decoder shows a degradation of less than 0.2 dB compared to STS-SD at a BER level of 2×10^{-5}. However, it outperforms MMSE-IC1 equalizer by about 0.4 dB at a BER level of 2×10^{-5}. MMSE-IC and I-VBLAST show almost the same performance as LC-*K*Best decoder with $I_{\text{out}} = 4$ and $I_{\text{in}} = 2$. MMSE-IC2 presents a degradation of 1 dB compared to MMSE-IC1. Moreover, with $I_{\text{out}} = 2$ and $I_{\text{in}} = 4$,

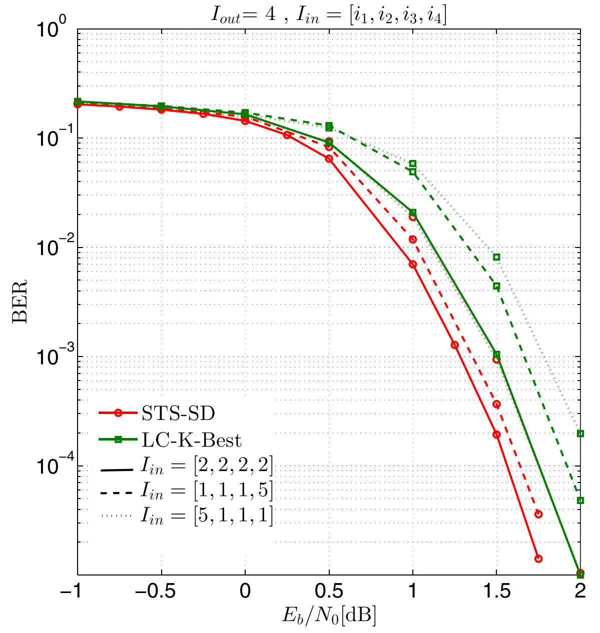


Figure 8 BER performance of a 4 × 4 coded MIMO system with 16-QAM using STS-SD and LC-*K*-Best decoders. Eight turbo decoding iterations are distributed over four outer iterations. I_{in} =[i_1, i_2, i_3, i_4] indicates that i_k inner iterations are performed in the k^{th} outer iteration. Turbo code with $R_c = 1/2$ and $L = 2,048$ is used.

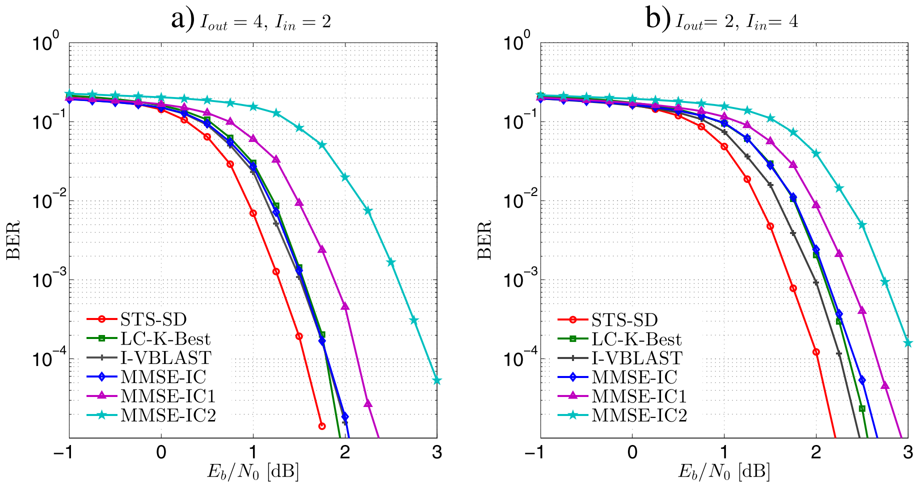


Figure 9 BER performance of a 4 × 4 coded MIMO system with 16-QAM using several MIMO detectors. (STS-SD, LC-***K***-Best, I-VBLAST, MMSE-IC, MMSE-IC1, and MMSE-IC2), **(a)** $I_{\text{in}} = 2$, $I_{\text{out}} = 4$ and **(b)** $I_{\text{in}} = 4$, $I_{\text{out}} = 2$. Turbo code with $R_c = 1/2$ and $L = 2,048$ is used.

I-VBLAST slightly outperforms LC-*K*-Best decoder and MMSE-IC equalizer.

In addition, Figure 10 shows the performance of MIMO detection algorithms with $I_{\text{out}} = 2, 4, 8$ and $I_{\text{in}} = 1$. We show that $I_{\text{out}} = 2$ is not sufficient for system convergence. With $I_{\text{out}} = 4$, there is a degradation of about 0.3 to 0.5 dB at the BER level of 10^{-5} compared to Figure 9a. However, with $I_{\text{out}} = 8$, MIMO detection algorithms present an improvement of 0.1 dB at a BER level of 2×10^{-5}. This configuration with $I_{\text{in}} = 1$ and $I_{\text{out}} = 8$ increases the complexity of the receiver especially in the case of tree-search-based algorithms.

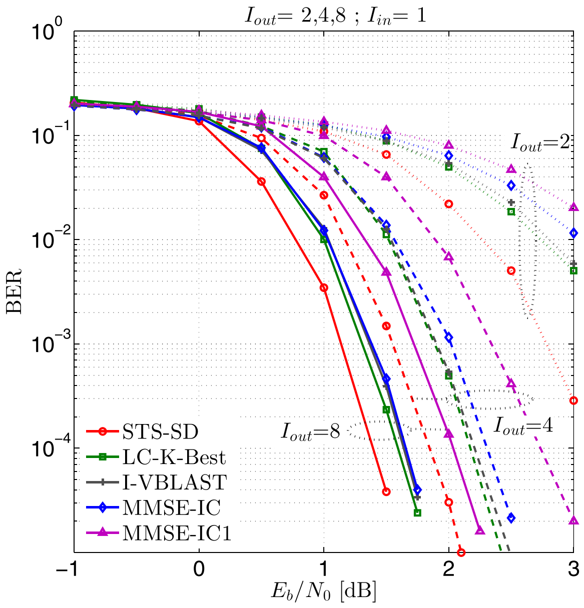


Figure 10 BER performance of a 4 × 4 coded MIMO system with 16-QAM using several MIMO detectors. (STS-SD, LC-***K***-Best, I-VBLAST, MMSE-IC, MMSE-IC1, and MMSE-IC2), $I_{\text{in}} = 1$, $I_{\text{out}} = 2, 4, 8$. Turbo code with $R_c = 1/2$ and $L = 2,048$ is used.

Figure 11 shows the BER performance of 64-QAM with $I_{\text{out}} = 4$, $I_{\text{in}} = 2$, and $R_c = 3/4$. We see that LC-*K*-Best decoder with a list size of 64 presents a similar performance as STS-SD. Moreover, performance degradation of 0.1 dB at a BER level of 1×10^{-4} is observed with a list size of 32. However, I-VLAST equalizer and MMSE-IC equalizer present a degradation of 2 dB at a BER level of 1×10^{-4} compared to LC-*K*-Best decoder. Therefore, LC-*K*-Best decoder is more robust in the case of high-order modulations and high coding rate.

In order to summarize the performance of different detectors with different system configurations, we provide the E_b/N_0 values achieving a BER level of 2×10^{-5} in Table 2. The number used in the parenthesis in the table represents the performance loss over STS-SD.

Simulation results show that iterative receiver substantially improves the performance of coded MIMO systems (Figure 7). Moreover, after a certain number of iterations, the performance of the system becomes saturated and does not show significant improvement anymore. Additionally, Figures 8 and 9 show that the scheduling order and the number of iterations affect the system performance. Their impact on the complexity of the system will be detailed in the next section.

6 Complexity comparisons

In practical systems, the computational complexity impacts the latency, the throughput, and the power consumption of the device. Therefore, the receiver algorithms must be a compromise between performance and cost. In this section, we evaluate the computational complexity of

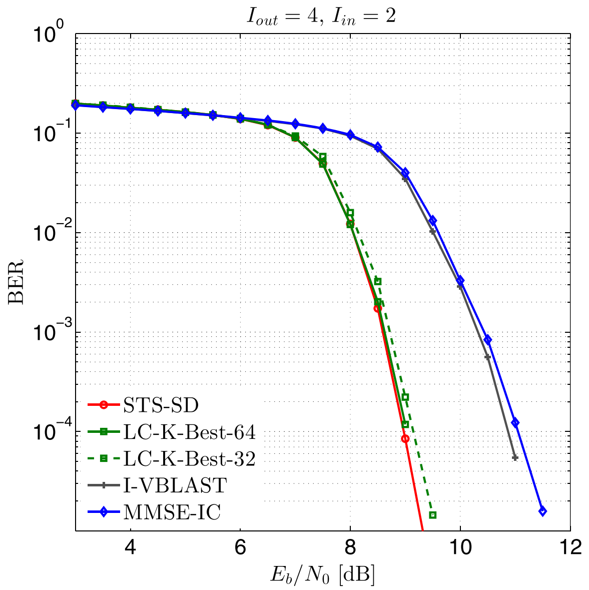


Figure 11 BER performance of a 4×4 coded MIMO system with 64-QAM using several MIMO detectors. (STS-SD, LC-***K***-Best, I-VBLAST, and MMSE-IC), $I_{\text{in}} = 2$, $I_{\text{out}} = 4$. Turbo code with $R_c = 3/4$ and $L = 2{,}048$ is used.

the turbo decoder and the family of MIMO detection algorithms in terms of the number and the type of real float value operations (additions/subtractions, multiplications, divisions). We then compare the overall complexity of the iterative receiver for different system configurations.

6.1 Iterative receiver complexity

For an iterative receiver, the algorithm complexity is related to the channel decoder, the MIMO detector, and the number of iterations. The overall complexity of an iterative receiver can be expressed by:

$$C_{\text{total}} = I_{\text{in}} I_{\text{out}} C_{\text{Turbo}} N_{\text{bit}} + N_{\text{symb}} \left\{ C_{\text{det1}} + (I_{\text{out}} - 1) C_{\text{deti}} \right\}, \tag{29}$$

where C_{det1} denotes the complexity of the first iteration of MIMO detection algorithm per symbol vector without taking into consideration the *a priori* information. While C_{deti} denotes the complexity per iteration per symbol vector taking into consideration the *a priori* information. C_{Turbo} denotes the complexity of the turbo decoder per iteration per information bit. N_{bit} is the number of information bit at the input of the encoder, while N_{symb} is the number of symbol vectors. N_{symb} and N_{bit} are linked by the following relation:

$$N_{\text{symb}} = \frac{N_{\text{bit}}}{QR_c M} = \alpha N_{\text{bit}}, \quad \text{with} \quad \alpha = \frac{1}{QR_c M}. \tag{30}$$

6.2 Turbo decoder complexity

The turbo decoder complexity depends on SISO decoder algorithms and the number of iterations. Herein, Max-Log-MAP algorithm with a correction factor is used [56]. The complexity of Max-Log-MAP decoder corresponds to three principal computations: branch metrics, recursive state metrics, and LLR of the bits. Table 3 summarizes the total number of operations per info bit and per iteration for the LTE turbo decoder with eight states and $n = 2$ output bits. Therefore, the overall complexity of the turbo decoder can be obtained by multiplying it by the block length L and by the number of iterations $I_{\text{in}} I_{\text{out}}$.

6.3 MIMO detection complexity

In the case of tree-search-based algorithms, the commonly used approach to measure the complexity is to count the number of visited nodes in the tree-search process. In the case of the equalizer, the complexity is evaluated in terms of real or complex operations required to compute filter coefficients.

The complexity of tree-search-based algorithms can be divided into two steps: the pre-processing step and the tree-search step. However, the complexity of IC equalizer algorithms are dominated by the computation of the filter coefficients and the matrix inversion. Herein, QR decomposition method is used to help the computation of the matrix inversion. In the case of quasi-stationary channel, the channel matrix is assumed to be constant over a long period of time. The QR decomposition and the matrix inversion can be performed only once over the frame. Therefore, their associated complexity can be substantially reduced in slow fading environment. Moreover, the QR decomposition is performed at the first iteration. The complexity of QR decomposition can be then considered negligible compared to the global complexity of the iterative receiver.

Table 2 E_b/N_0 values achieving a BER level of 2×10^{-5} for different detectors with 4×4 16-QAM, $R_c = 1/2$[a]

	$I_{\text{out}} = 4, I_{\text{in}} = 8$	$I_{\text{out}} = 4, I_{\text{in}} = 2$	$I_{\text{out}} = 8, I_{\text{in}} = 1$
STS-SD	1.69 dB	1.72 dB	1.56 dB
LC-*K*-Best	1.87 dB (−0.18)	1.89 dB (−0.17)	1.76 dB (−0.20)
I-VBLAST	1.88 dB (−0.19)	1.97 dB (−0.25)	1.78 dB (−0.22)
MMSE-IC	1.92 dB (−0.23)	1.89 dB (−0.27)	1.76 dB (−0.24)
MMSE-IC1	2.20 dB (−0.51)	2.28 dB (−0.56)	2.20 dB (−0.64)

[a] The number in the parenthesis corresponds to the performance loss in dB over STS-SD.

Table 3 Complexity of turbo decoder per information bit per iteration

	ADD	Max (2-input)	LUT
Turbo decoder $2^m = 8, n = 2$	232	88	88

The complexity of the sphere decoder depends on the modulation order and on the number of transmit antennas [13,16,17]. The average complexity of the sphere decoder was shown to be polynomial in the number of unknowns roughly $O(M^3)$. However, it presents an exponential complexity in the worst case conditions depending on the noise level and the choice of an initial radius. Due to the sequential nature of the tree search and the statistical effect of the channels, it is very difficult to find an analytical expression of SD complexity. Herein, Monte Carlo simulations were used to measure the number of operations (additions and multiplications) of STS-SD.

6.4 Complexity results

In the case of a 4×4 16-QAM system, the complexity of different detection algorithms in terms of number of operations are summarized in Figures 12 and 13 at the first and ith iteration, respectively. The MAP algorithm presents the highest complexity (4.7×10^6 MUL, 4.6×10^6 ADD). It is used as a reference to view the reduction in the complexity of other algorithms compared to the optimal detector. The complexity of STS-SD depends largely on the SNR. Its average complexity is computed through simulations over all SNR range. The average number of arithmetic operations is 90% lower than the MAP algorithm. However, it still has a larger complexity than other algorithms.

The complexity of the equalizer comprises the complexity of soft mapping and soft demapping. The complexity of K-Best and LC-K-Best decoders includes the complexity of SQR decomposition at the first iteration and LLR computation. The complexity of classical K-Best decoder is about 50% higher than that of LC-K-Best decoder. LC-K-Best complexity is approximately 30% higher than that of the MMSE equalizer and 50% lower than that of I-VBLAST. I-VBLAST requires more complexity due to the matrix inversion for each detected symbol. For ith iteration, MMSE-IC shows 56% higher complexity than MMSE-IC1 due to the matrix inversion for each detected symbol. However, MMSE-IC2 has 62% lower complexity than MMSE-IC1 since MMSE-IC2 approximation does not contain a matrix inversion. Comparing the complexity of MMSE-IC1 equalizer and LC-K-Best decoder for the ith iteration, we found that LC-K-Best decoder presents a lower number of arithmetic operations (22% MUL, 10% ADD). This is due to the fact that the equalizer repeats the matrix inversion for each iteration. However, LC-MMSE-IC algorithm proposed in [21] has a lower complexity than the LC-K-Best decoder in terms of MUL (7%) and ADD (19%) with additional DIV and SQRT operations required by the matrix inversion. It is important to note that in LC-K-Best decoder, there is a number of comparisons to choose the best candidates that are not taken into consideration in the complexity comparisons. MMSE-IC2 presents the lowest complexity but with more performance degradation, it presents a reduction of 58% MUL and 52% ADD with a degradation of 1.5 dB compared to LC-K-Best decoder. Moreover, if the channel matrix is assumed to be quasi-stationary, the SQR decomposition as well as the matrix inversion associated with MMSE will be reused within the signal frame.

As we saw in Section 5, the scheduling order and the number of iterations have a great impact on the system performance. The best trade-off schedule when performing I_{out} outer iterations is to perform at least eight turbo decoding iterations distributed equally into $I_{\text{in}} = 8/I_{\text{out}}$. For this reason, we consider the two schemes having a difference of 0.5 dB at a BER level of 10^{-5} to view the impact of this performance degradation into the overall complexity. In the first scheme (scheme 1), $I_{\text{out}} = 4$ and $I_{\text{in}} = 2$ iterations are performed. The second scheme (scheme 2) has $I_{\text{out}} = 2$ and $I_{\text{in}} = 4$. These two schemes present an equal number of turbo decoding iterations. Therefore, the complexity in terms of number of operations for the turbo decoder will be the same. However, the access to the memory will be changed.

Figure 14 summarizes the overall complexity of the receiver for one transmitted frame with these two schemes using different detection algorithms. As shown in Figure 14, a significant reduction of MUL operations between 40% ∼ 60% is obtained with scheme 2, and a reduction of ADD/SUB operations between 5% ∼ 25% is observed. However, the number of Max

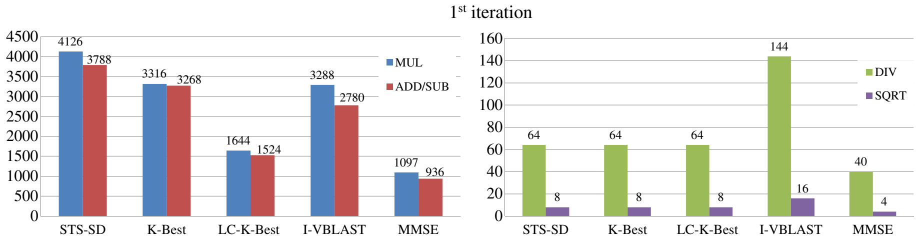


Figure 12 Complexity of a 4×4 16-QAM system for different detection algorithms. Evaluated in terms of the number of operations per symbol vector at the first iteration.

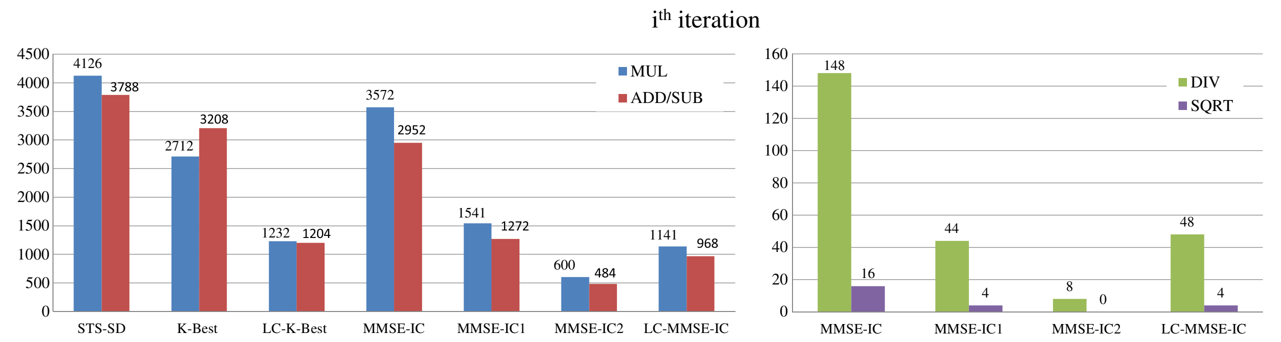
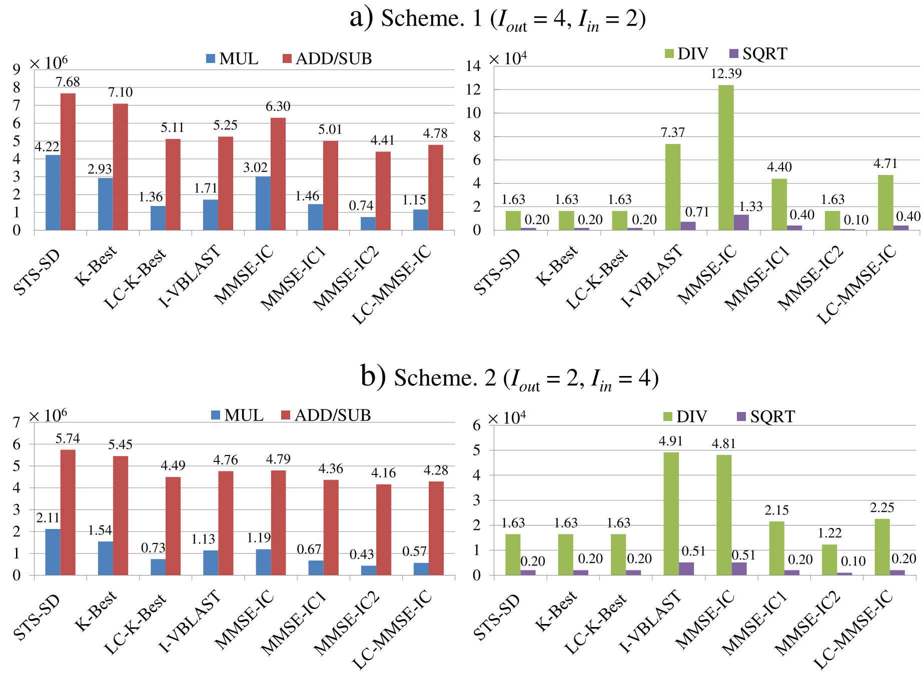

ith iteration

Figure 13 Complexity of a 4 × 4 16-QAM system for different detection algorithms. Evaluated in terms of the number of operations per symbol vector at the *i*th iteration.

operations remains the same since the same number of turbo decoding iterations is used. DIV and SQRT operations are more reduced in the case of MMSE-IC, MMSE-IC1, and LC-MMSE-IC by about 50% ∼ 60% due to reduction of matrix inversions. MMSE-IC2 presents a reduction of the DIV operations by 25%. Moreover, comparing the overall complexity in the same scheme, we show that the complexity of STS-SD is much higher than LC-K-Best decoder (65% ∼ 67% MUL, 21% ∼ 33% ADD) at the expense of only 0.4 ∼ 0.25 dB of performance improvement at a BER level of 1×10^{-5}. K-Best decoder presents lower complexity than STS-SD but higher than LC-K-Best decoder. LC-K-Best decoder shows approximately the same complexity as MMSE-IC1 equalizer. In addition, LC-K-Best presents a reduced complexity than I-VBLAST (20% ∼ 35% MUL, 2% ∼ 5% ADD, approximately 50% DIV, approximately 50% SQRT). The reason is that I-VBLAST requires multiple matrix inversions at the first iteration. However, LC-MMSE-IC has a lower complexity than LC-K-Best decoder in terms of MUL (15% ∼ 22% less) and ADD (4% ∼ 6% less) but requires more DIV (20% ∼ 50% more) and SQRT (approximately 50% more) operations. Furthermore, the complexity of MMSE-IC2 equalizer is much lower than LC-K-Best decoder (40 ∼ 45% MUL less, 7% ∼ 13% ADD less). However, MMSE-IC2 presents a significant degradation of 1 ∼ 1.25 dB at 1×10^{-5} compared to LC-K-Best decoder.


a) Scheme. 1 ($I_{out} = 4$, $I_{in} = 2$)

b) Scheme. 2 ($I_{out} = 2$, $I_{in} = 4$)

Figure 14 Overall computational complexity of a 4 × 4 16-QAM system. Two distinct system configurations: **(a)** scheme 1 and **(b)** scheme 2.

It is therefore worthy to compare the overall complexity of different receivers with approximately the same performance. We consider different system configurations for different detection algorithms given approximately 1×10^{-5} at $E_b/N_0 = 2$ dB at the expense of MMSE-IC1 and MMSE-IC2 that have a degradation of 0.25 and 1 dB, respectively. Figure 15 illustrates the obtained results. As Figure 15 shows, LC-K-Best decoder, LC-MMSE-IC, and MMSE-IC2 equalizers have the lowest computational complexity. However, MMSE-IC2 presents more performance degradation. We remind that LC-K-Best decoder requires many comparison operations that are not considered in the complexity. MMSE-IC (8 to 1) has the higher complexity even higher than STS-SD, due to matrix inversion for each symbol and for each iteration. MMSE-IC (4 to 2) has lower complexity than MMSE-IC (8 to 1) (55% MUL, DIV, SQRT; 32 % ADD) with small degradation (0.1 $\sim$ 0.2 dB). Furthermore, MMSE-IC1 and I-VBLAST present a higher complexity than LC-K-Best since the matrix inversion is repeated at each iteration while in the LC-K-Best decoder the SQR decomposition is only done at the first iteration. Moreover, LC-MMSE-IC presents a lower complexity than LC-K-Best as discussed previously in Figure 14 with more DIV and SQRT operations.

Other aspects must be taken into consideration such as the required memory and the number of access to the memory since these aspects affect the scale of the receiver and the latency of the system. Moreover, the interleaver plays a major role to manage the access of the memory. These aspects can be the subject of future investigations.

From the theoretical complexity results, LC-K-Best decoder seems to be less complex and more suitable for implementation. Furthermore, LC-K-Best decoder performs a breath-first search that can be easily paralyzed and pipelined in hardware architecture as discussed in [14,28].

7 Conclusions

This paper provides an overview of several SISO MIMO detection algorithms proposed to avoid the exponential complexity of the MAP algorithm such as LSD, STS-SD, and K-Best decoders and IC equalizers. A LC-K-Best decoder is presented in order to avoid the full expansion and to simplify the enumeration through the use of two LUTs. Moreover, we analyze the convergence of the iterative process for several detection algorithms. We then compare the performance of these detection algorithms with the original scheduling of number of iterations and the new proposed scheduling. The theoretical complexity of turbo decoder and the family of MIMO detectors is evaluated in terms of number and type of real valued operations. We compare their complexity with different system configurations. Simulation results show that the new schedule gives similar performance to the original schedule while saving a large amount of turbo decoder complexity and latency. Additionally, complexity results show that the LC-K-Best decoder achieves a best trade-off

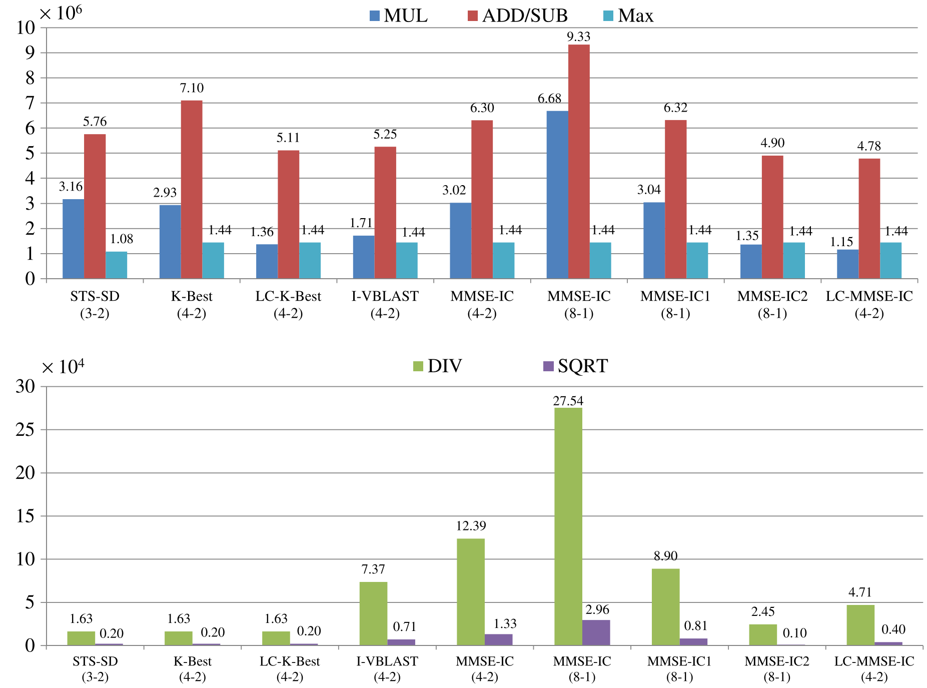


Figure 15 Overall computational complexity of a 4×4 16-QAM system of different detection algorithms. With approximately an equivalent performance of 1×10^{-5} BER at $E_b/N_0 = 2$ dB.

between performance and complexity among the studied detectors. Future work can include other aspects like memory access, memory requirement, fixed point conversion, and implementation in real environments.

Competing interests

The authors declare that they have no competing interests.

Received: 21 July 2014 Accepted: 20 February 2015
Published online: 12 March 2015

References

1. E Telatar, Capacity of multi-antenna Gaussian channels. Eur. Trans. Telecommun. **10**, 585–595 (1999)
2. C Berrou, A Glavieux, P Thitimajshima, Near Shannon limit error-correcting coding and decoding Turbo-codes. IEEE Int. Conf. Commun. **2**, 1064–1070 (1993)
3. C Douillard, M Jézéquel, C Berrou, A Picart, P Didier, A Glavieux, Iterative correction of intersymbol interference: Turbo-equalization. Eur. Trans. Telecommun. **6**(5), 507–511 (1995)
4. M Tüchler, AC Singer, R Kotter, Minimum mean squared error equalization using a priori information. IEEE Trans. Signal Process. **50**(3), 673–683 (2002)
5. X Wang, HV Poor, Iterative (turbo) soft interference cancellation and decoding for coded CDMA. IEEE Trans. Commun. **47**(7), 1046–1061 (1999)
6. M Witzke, S Baro, F Schreckenbach, J Hagenauer, Iterative detection of MIMO signals with linear detectors. Asilomar Conf. Signals Syst. Comput. **1**, 289–293 (2002)
7. GJ Foschini, Layered space-time architecture for wireless communication in fading environment when using multiple antennas. Bell Labs Tech. J. **1**(2), 41–59 (1996)
8. PW Wolniansky, GJ Foschini, GD Golden, RA Valenzuela, V-BLAST : an architecture for realizing very high data rates over the rich-scattering wireless channel. Int. Symp. Signals Syst. Electron, 295–300 (1998)
9. U Fincke, M Pohst, Improved methods for calculating vectors of short length in a lattice, including a complexity analysis. Math. Comput. **44**(170), 463–471 (1985)
10. CP Schnorr, M Euchner, Lattice Basis Reduction: Improved Practical Algorithms and Solving Subset Sum Problems. Math. Program. **66**, 181–191 (1994)
11. E Viterbo, J Boutros, A Universal lattice code decoder for fading channels. IEEE Trans. Inform. Theory. **45**(5), 1639–1642 (1999)
12. E Agrell, T Eriksson, A Vardy, K Zeger, Closest point search in lattices. IEEE Trans. Inform. Theory. **48**(8), 2201–2214 (2002)
13. MO Damen, HE Gamal, G Caire, On maximum-likelihood detection and the search for the closest lattice point. IEEE Trans. Inform. Theory. **49**(10), 2389–2402 (2003)
14. K-w Wong, C-y Tsui, RS-k Cheng, W-h Mow, A VLSI Architecture of a K-Best lattice decoding algorithm for MIMO channels. IEEE Int. Symp. Circuits Syst. **3**, 273–276 (2002)
15. LG Barbero, JS Thompson, Rapid prototyping of a fixed-throughput sphere decoder for MIMO systems. IEEE Int. Conf. Commun. **7**, 3082–3087 (2006)
16. B Hassibi, H Vikalo, On the sphere-decoding Algorithm I. Expected complexity. IEEE Trans. Signal Process. **53**(8), 2806–2818 (2005)
17. J Jaldén, B Ottersten, On the complexity of sphere decoding in digital communications. IEEE Trans. Signal Process. **53**(4), 1474–1484 (2005)
18. H Lee, B Lee, I Lee, Iterative detection and decoding with an improved V-BLAST for MIMO-OFDM systems. IEEE J. Selected Areas Commun. **24**(3), 504–513 (2006)
19. JW Choi, AC Singer, J Lee, NI Cho, Improved linear soft-input soft-output detection via soft feedback successive interference cancellation. IEEE Trans. Commun. **58**(3), 986–996 (2010)
20. L Boher, R Rabineau, M Hélard, FPGA implementation of an iterative receiver for MIMO-OFDM systems. IEEE J. Selected Areas Commun. **26**(6), 857–866 (2008)
21. C Studer, S Fateh, D Seethaler, ASIC Implementation of soft-input soft-output MIMO detection using MMSE parallel interference cancellation. IEEE J. Solid-State Circuits. **46**(7), 1754–1765 (2011)
22. BM Hochwald, S ten Brink, Achieving near-capacity on a multiple-antenna channel. IEEE Trans. Commun. **51**(3), 389–399 (2003)
23. S Baro, J Hagenauer, M Witzke, Iterative detection of MIMO transmission using a list-sequential (LISS) detector. IEEE Int. Conf. Commun. **4**, 2653–2657 (2003)
24. C Studer, A Burg, H Bolcskei, Soft-output sphere decoding: algorithms and VLSI implementation. IEEE J. Selected Areas Commun. **26**(2), 290–300 (2008)
25. C Studer, H Bölcskei, Soft-input soft-output single tree-search sphere decoding. IEEE Trans. Inform. Theory. **56**(10), 4827–4842 (2010)
26. EP Adeva, T Seifert, G Fettweis, VLSI architecture for MIMO soft-input soft-output sphere detection. J. Signal Process. Syst. **70**(2), 125–143 (2013)
27. Y de Jong, T Willink, Iterative tree search detection for MIMO wireless systems. IEEE Trans. Commun. **53**(6), 930–935 (2005)
28. Z Guo, P Nilsson, Algorithm and implementation of the K-Best sphere decoding for MIMO detection. IEEE J. Selected Areas Commun. **24**(3), 491–503 (2006)
29. M Myllyla, M Juntti, JR Cavallaro, Implementation aspects of list sphere decoder algorithms for MIMO-OFDM systems. Signal Process. **90**(10), 2863–2876 (2010)
30. L Barbero, J Thompson, Extending a fixed-complexity sphere decoder to obtain likelihood information for turbo-MIMO systems. IEEE Trans. Vehic. Technol. **57**(5), 2804–2814 (2008)
31. EM Witte, F Borlenghi, G Ascheid, R Leupers, H Meyr, A scalable VLSI architecture for soft-input soft-output single tree-search sphere decoding. IEEE Trans. Circuits Syst. II. **57**(9), 706–710 (2010)
32. F Borlenghi, EM Witte, G Ascheid, H Meyr, A Burg, A 772Mbit/s 8.81bit/nJ 90nm CMOS soft-input soft-output sphere decoder. IEEE Asian Solid State Circuits Conf., 297–300 (2011)
33. Y Sun, JR Cavallaro, Trellis-search based soft-input soft-output MIMO detector: Algorithm and VLSI Architecture. IEEE Trans. Signal Process. **60**(5), 2617–2627 (2012)
34. B Wu, G Masera, Efficient VLSI implementation of soft-input soft-output fixed-complexity sphere decoder. IET Commun. **6**(9), 1111–1118 (2012)
35. L Liu, High-throughput hardware-efficient soft-input soft-output MIMO detector for iterative receivers. IEEE Int. Symp. Circuits Syst, 2151–2154 (2013)
36. X Chen, G He, J Ma, VLSI implementation of a high-throughput iterative fixed-complexity sphere decoder. IEEE Trans. Circuits Syst. II: Express Briefs. **60**(5), 272–276 (2013)
37. D Patel, V Smolyakov, M Shabany, PG Gulak, VLSI implementation of a WiMAX/LTE compliant low-complexity high-throughput soft-output K-Best MIMO detector. IEEE Int. Symp. Circuits Syst, 593–596 (2010)
38. P-Y Tsai, W-T Chen, X-C Lin, M-Y Huang, A 4×4 64-QAM reduced-complexity K-Best MIMO detector up to 1.5Gbps. IEEE Int. Symp. Circuits Syst, 3953–3956 (2010)
39. M Mahdavi, M Shabany, Novel MIMO detection algorithm for high-order constellations in the complex domain. IEEE Trans. Very Large Scale Integration (VLSI) Syst. **21**(5), 834–847 (2013)
40. RE Chall, F Nouvel, M Hélard, M Liu, Low complexity K-Best based iterative receiver for MIMO systems. Int. Congress Ultra Modern Telecommun. and Control Syst, 451–455 (2014). doi:10.1109/ICUMT.2014.7002143
41. S ten Brink, Convergence behavior of iteratively decoded parallel concatenated codes. IEEE Trans. Commun. **49**(10), 1727–1737 (2001)
42. JJ Boutros, F Boixadera, C Lamy, Bit-interleaved coded modulations for multiple-input multiple-output channels. IEEE Sixth Int. Symp. Spread Spectrum Tech. Appl. **1**, 123–126 (2000)
43. A Burg, N Felber, W Fichtner, A 50 Mbps 4×4 maximum likelihood decoder for multiple-input multiple-output systems with QPSK modulation. IEEE Int. Conf. Electron. Circuits Syst. **1**, 322–335 (2003)
44. D Wübben, R Bohnke, J Rinas, V Kuhn, KD Kammeyer, Efficient algorithm for decoding layered space-time codes. Electron. Lett. **37**(22), 1348–1350 (2001)
45. D Wübben, R Bohnke, V Kuhn, K-D Kammeyer, MMSE extension of V-BLAST based on sorted QR decomposition. IEEE 58th Vehic. Technol. Conf. (VTC). **1**, 508–512 (2003)
46. E Zimmermann, G Fettweis, in *Int. Symp. Wireless Personel, Mutimedia Commun*. Unbiased MMSE tree search detection for multiple antenna systems (San Diego, USA, 2006), pp. 1–5
47. R Wang, GB Giannakis, Approaching MIMO channel capacity with reduced-complexity soft sphere decoding. IEEE Wireless Commun. Netw. Conf. **3**, 1620–1625 (2004)

48. S Chen, T Zhang, Y Xin, Relaxed K-Best MIMO signal detector design and VLSI implementation. IEEE Trans. Very Large Scale Integration Syst. **15**(3), 328–337 (2007)
49. M Wenk, M Zellweger, A Burg, N Felber, W Fichtner, K-Best MIMO detection VLSI architectures achieving up to 424 Mbps. IEEE Int. Symp. Circuits and Syst, 4 pp–1154 (2006)
50. M Shabany, PG Gulak, Scalable VLSI architecture for K-Best lattice decoders. IEEE Int. Symp. Circuits and Syst, 940–943 (2008)
51. DL Milliner, E Zimmermann, JR Barry, G Fettweis, A fixed-complexity smart candidate adding algorithm for soft-output MIMO detection. IEEE J. Selected Topics Signal Process. **3**(6), 1016–1025 (2009)
52. JW Choi, B Shim, JK Nelson, AC Singer, Efficient soft-input soft-output MIMO detection via improved M-algorithm. IEEE Int. Conf. Commun, 1–5 (2010)
53. IB Collings, MRG Butler, MR McKay. Low complexity receiver design for MIMO bit-interleaved coded modulation. *IEEE Int. Symp. Spread Spectrum Techniques and Applications* (IEEE Piscataway, 2004), pp. 12–16
54. E Zimmermann, G Fettweis. Adaptive vs. hybrid iterative MIMO receivers based on MMSE linear and soft-SIC detection. *IEEE 17th Int. Symp. Personal, Indoor and, Mobile Radio Commun* (IEEE Helsinki, Finland, 2006), pp. 1–5
55. C-H Liao, I-W Lai, K Nikitopoulos, F Borlenghi, D Kammler, M Witte, D Zhang, T-D Chiueh, G Ascheid, H Meyr. Combining orthogonalized partial metrics: efficient enumeration for soft-input sphere decoder. *IEEE 20th Int. Symp. Personal, Indoor and, Mobile Radio Commun* (IEEE Tokyo, Japan, 2009), pp. 1287–1291
56. P Robertson, E Villebrun, P Hoeher, A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain. IEEE Int. Conf. Commun. **2**, 1009–1013 (1995)