Signal Processing 90 (2010) 2863–2876

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Implementation aspects of list sphere decoder algorithms for MIMO-OFDM systems

Markus Myllylä [a,*], Markku Juntti [b], Joseph R. Cavallaro [c]

[a] Nokia Corporation, P.O. Box 50, FI-90571 Oulu, Finland
[b] Centre for Wireless Communications, P.O. Box 4500, FI-90014 University of Oulu, Finland
[c] Department of Electrical and Computer Engineering, Rice University, Houston, TX, USA

ARTICLE INFO

Article history:
Received 4 November 2009
Received in revised form
1 February 2010
Accepted 15 April 2010
Available online 20 April 2010

Keywords:
MIMO
LSD
Soft-output detector
Implementation

ABSTRACT

A list sphere decoder (LSD) can be used to approximate the optimal maximum *a posteriori* (MAP) detector for the detection of multiple-input multiple-output (MIMO) signals. In this paper, we consider two LSD algorithms with different search methods and study some algorithm design choices which relate to the performance and computational complexity of the algorithm. We show that by limiting the dynamic range of log-likelihood ratio, the required LSD list size can be lowered, and, thus, the complexity of the LSD algorithm is decreased. We compare the real and the complex-valued signal models and their impact on the complexity of the algorithms. We show that the real-valued signal model is clearly the less complex choice and a better alternative for implementation. We also show the complexity of the sequential search LSD algorithm can be reduced by limiting the maximum number of checked nodes without sacrificing the performance of the system. Finally, we study the complexity and performance of an iterative receiver, analyze the tradeoff choices between complexity and performance, and show that the additional computational cost in LSD is justified to get better soft-output approximation.

1. Introduction

Multiple-input multiple-output (MIMO) techniques in combination with orthogonal frequency-division multiplexing (MIMO-OFDM) have been identified as a promising approach for high spectral efficiency wideband systems and has been included to many upcoming wireless standards such as (3GPP) long term evolution (LTE), and IEEE 802.16e WiMAX and IEEE 802.11n.

Practical communication systems apply forward error control (FEC) coding in order to achieve near capacity performance. The optimal way to decode the MIMO-OFDM signal with FEC would be to use a joint detector and decoder for the whole coded data block, which is computationally very complex and not feasible with the current technology. However, the optimal receiver can be approximated by using an iterative receiver with a separate soft-input soft-output (SISO) detector and decoder, which exchange reliability information between the units [1]. The optimal SISO detector would be the maximum *a posteriori* (MAP) detector, which is often too complex for systems with a large number of transmitted spatial streams and high order modulation. Suboptimal linear detectors [2] offer low complexity solutions with adequate performance in low correlated channels, but suffer from rather poor performance in correlated fading channels especially with high coding rate [3,4]. The ordered serial interference cancellation (OSIC) detector [5,6] offers better performance compared to linear detectors, but still suffers a loss in performance in poor channel conditions compared to the MAP detector.

* Corresponding author.
E-mail addresses: markus.myllyla@nokia.com (M. Myllylä), markku.juntti@ee.oulu.fi (M. Juntti), cavallar@rice.edu (J.R. Cavallaro).

0165-1684/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.sigpro.2010.04.014

A sphere decoder (SD) [7,8] calculates the hard output maximum likelihood (ML) solution with reduced complexity compared to the full-complexity ML detector. We focus our interest on the list sphere decoder (LSD) [1], which is a variant of the sphere decoder that can be efficiently used to approximate a soft output MAP detector. The practical feasibility of various SD versions is further supported recently by practical implementations reported in the literature [9–13].

Contribution: In this paper, we identify and study some key implementation challenges encountered while implementing the LSD algorithms in practical wireless systems. We concentrate our study on two LSD algorithm variants with different search strategies and focus on four significant challenges, namely, limiting the dynamic range of the soft output log-likelihood ratio (LLR) values to the decoder, comparing the real and complex-valued signal models in the LSD, limiting the search complexity of the LSD algorithm, and analyzing the complexity and performance tradeoffs of an iterative receiver. The paper is an extension of our earlier conference contributions [14–16] and provides a more systematic and uniform treatment as well as deeper analysis and more numerical results.

The paper is organized as follows. The MIMO signal detection is discussed in Section 2. The SD and the LSD algorithms are presented in Section 3. The LLR clipping is presented in Section 4, and the real and complex-valued signal models are compared and analyzed in Section 5. The effects of limits on the number of nodes search by the LSD algorithms is discussed in Section 6, and the performance and complexity of the iterative receiver is analyzed in Section 7. Conclusions are drawn in Section 8.

2. MIMO signal detection

A coded MIMO-OFDM system is considered with N_T transmit (TX) antennas and N_R receive (RX) antennas. A spatial multiplexing (SM) transmission with N_T spatial streams is used with quadrature amplitude modulation constellation (QAM) and with the assumption $N_R \geq N_T$. The received signal at baseband can be expressed for each OFDM subcarrier in terms of code symbol interval as

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \boldsymbol{\eta}, \tag{1}$$

where $\mathbf{y} \in \mathbb{C}^{N_R \times 1}$ is the received signal vector, $\mathbf{x} \in \mathbb{C}^{N_T \times 1}$ is the transmit symbol vector with symbol power of E_S and $\boldsymbol{\eta} \in \mathbb{C}^{N_R \times 1}$ is the noise vector with independent and complex zero-mean Gaussian elements with power N_0. The channel matrix $\mathbf{H} \in \mathbb{C}^{N_R \times N_T}$ contains complex Gaussian fading coefficients with unit variance. The entries of $\mathbf{x}$ are chosen independently from a complex QAM constellation Ω with sets of Q transmitted coded binary information bits $\mathbf{b}=[\mathbf{b_1},\ldots,\mathbf{b_Q}]$ per symbol, i.e., $|\Omega| = 2^Q$.

The optimal way to decode the coded signal would be to use a joint detector and decoder for the whole coded data block. This, however, is computationally very complex and not feasible with the current technology. A suboptimal way is to have a separate soft-input soft-output (SISO) detector and channel decoder at the receiver. The turbo principle [1] can be applied in the receiver so that the detector and decoder exchange the information in an iterative fashion as illustrated in the block diagram of the system in Fig. 1. The detector generates soft output information L_{D1} from received data $\mathbf{y}$ and *a priori* information L_{A1}, and calculates extrinsic information L_{E1}. This information is fed as *a priori* information L_{A2} to the decoder after interleaving. The decoder output information L_{D2} can then be fed back to detector. The *a posteriori* log-likelihood ratio (LLR) $L_D(b_k)$ of the k th transmitted bit b_k, conditioned on the received signal vector $\mathbf{y}$, is defined as

$$L_D(b_k) = \ln \frac{P(b_k = +1|\mathbf{y})}{P(b_k = -1|\mathbf{y})}. \tag{2}$$

By using the Bayes' theorem, the probability can be written as [1,17]

$$\begin{aligned} L_D(b_k) &= \ln\left(\frac{p(\mathbf{y}|b_k=+1)}{p(\mathbf{y}|b_k=-1)}\frac{P(b_k=+1)}{P(b_k=-1)}\right) \\ &= \ln\frac{P(b_k=+1)}{P(b_k=-1)} + \ln\frac{p(\mathbf{y}|b_k=+1)}{p(\mathbf{y}|b_k=-1)} \\ &= L_A(b_k) + L_E(b_k|\mathbf{y}), \end{aligned} \tag{3}$$

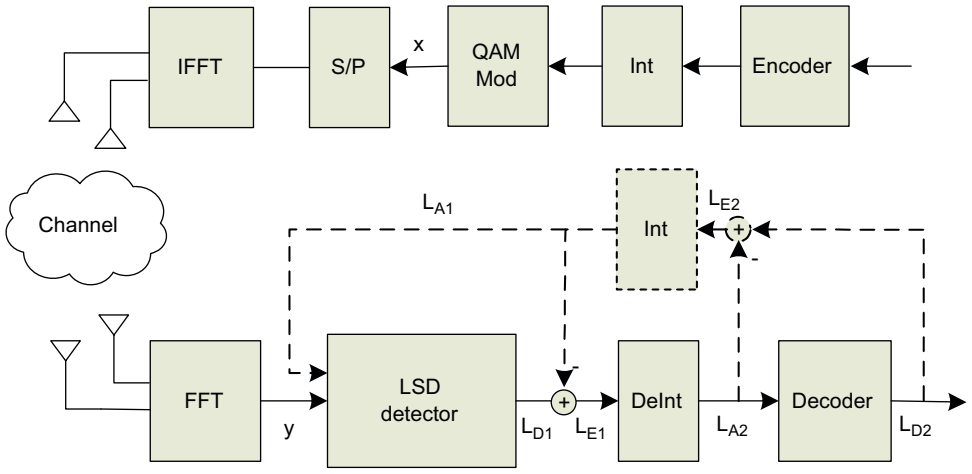

Fig. 1. A MIMO-OFDM system with N_T transmit and N_R receive antennas.

where $L_A(b_k)$ is the *a priori* information and $L_E(b_k)$ is the extrinsic information of the bits provided by the detector or decoder.

3. Sphere decoding

The SD algorithms achieve the ML solution with a reduced number of considered candidate symbol vectors in the search. This is done by limiting the search to points that lie inside a N_R-dimensional hyper-sphere $S(\mathbf{y},\sqrt{C_0})$ centered at $\mathbf{y}$ with a sphere radius $\sqrt{C_0}$. After QR decomposition (QRD) of the channel matrix $\mathbf{H}$, the condition can be written as [8]

$$\|\tilde{\mathbf{y}}-\mathbf{R}\mathbf{x}\|_2^2 \le C_0, \tag{4}$$

where $\mathbf{R}\in\mathbb{C}^{N_R\times N_T}$ is an upper triangular matrix with positive diagonal elements, $\mathbf{Q}\in\mathbb{C}^{N_R\times N_R}$ is an orthogonal matrix, $\tilde{\mathbf{y}}=\mathbf{Q}^H\mathbf{y}$, and C_0 is the squared radius of the sphere. Due to the upper triangular form of $\mathbf{R}$ the values of $\mathbf{x}$ can be solved from (4) level by level using the back-substitution algorithm. Let $\mathbf{x}_i^{N_T}=(x_i,x_{i+1},\ldots,x_{N_T})^T$ denote the last N_T-i+1 components of the vector $\mathbf{x}$. The sphere search can be illustrated with a tree structure, where the algorithm aims at finding the shortest path between the root layer and the leaf layer. First, the last elements of the possible symbol vectors are calculated, i.e., x_{N_T} and then x_{N_T-1} and so on. The squared partial Euclidean distance (PED) of $\mathbf{x}_i^{N_T}$ can be calculated as [9]

$$d(\mathbf{x}_i^{N_T})=d(\mathbf{x}_{i+1}^{N_T})+\left|\tilde{y}_i-\sum_{j=i}^{N_T} r_{i,j}x_j\right|^2, \tag{5}$$

where $d(\mathbf{x}_{N_T}^{N_T})=0$, $r_{i,j}$ is the (i,j)th term of $\mathbf{R}$ and $i=N_T,\ldots,1$. Depending on the search strategy and the channel realization, the SD searches a variable number of nodes in the tree structure, and aims to find the point $\mathbf{x}=\mathbf{x}_1^{N_T}$, also called a leaf node, for which the Euclidean distance (ED) $d(\mathbf{x}_1^{N_T})$ is minimum. It has been shown that the SD algorithm complexity, i.e., the number of visited nodes, can be decreased by applying proper preprocessing of the detection order [8,10], e.g., sorted QRD (SQRD) [18], which we also apply in the simulation results in the paper.

The ML solution that is given as an output by the SD may cause significant performance degradation compared to the optimal soft output MAP detection in a communication system with FEC. The more appropriate detector is the list sphere decoder (LSD) [1] that can be used for obtaining a list of candidate symbol vectors and the corresponding EDs $\mathcal{L}\in\mathbb{Z}^{N_{cand}\times N_T}$ as an output, where N_{cand} is the size of the candidate list so that $1\le N_{cand}\le 2^{QN_T}$. The output candidate list can then be used to approximate the MAP solution $L_D(b_k)$. The list sphere decoder algorithms can often be composed from the sphere decoder algorithms with minor modifications. The LSD algorithm is the most complex part of the detector, and the complexity is very much dependent on the applied search method. The tree search algorithms in the literature are often divided into three categories according to the search strategy, the breadth-first, the depth-first, and the metric-first [19,20]. In this paper, we consider two different LSD algorithms based on different search strategies, the breadth-first search based K-best-LSD [12,21] and the metric-first search based increasing radius (IR)-LSD [22,23].

3.1. K-best-LSD algorithm

The K-best-LSD algorithm listed as Algorithm 1 is a modification from the K-best-SD algorithm [12,21] to the LSD algorithm. The algorithm is based on the breadth-first strategy, i.e., the search proceeds one layer at a time in the search tree by extending the partial candidates $\mathbf{s}$ with admissible nodes and calculating the PEDs $d(\mathbf{s})$. The K-best-LSD algorithm search goes through a fixed number of nodes in the tree structure if no enumeration method and sphere radius C_0 is introduced, which makes the algorithm very suitable for implementation. However, it should be noted that the output candidate list $\mathcal{L}$ of the algorithm does not necessarily contain the candidates with the lowest EDs, which may result in inaccurate likelihood information approximation and performance loss.

Algorithm 1 $[\mathcal{L}]=$ K-best-LSD$(\tilde{\mathbf{y}},\mathbf{R},C_0,K,\Omega,N_T)$

Initialize set $\{\mathcal{L}\}_0$ with $\mathcal{N}(\mathbf{s}_0=\mathbf{x}_{N_T}^{N_T},d(\mathbf{s})=0)$ and empty set $\mathcal{S}$
for Layer $i=N_T-1$ to 0 **do**
  for $k=0$ to $|\mathcal{L}|-1$ **do**
    Remove $\mathcal{N}(\mathbf{s}=\mathbf{x}_{i+1}^{N_T},d(\mathbf{s}))$ from $\{\mathcal{L}\}_k$
    for $j=1$ to $|\Omega|$ **do**
      Determine $\mathbf{s}_c=(x_i,\mathbf{s})^T$, where $x_i=\{\Omega\}_j$ and calculate $d(\mathbf{s}_c)$
      if $d(\mathbf{s}_c)<C_0$ **then**
        Store $\mathcal{N}_c(\mathbf{s}_c,d(\mathbf{s}_c))$ to $\mathcal{S}$
      end if
    end for
  end for
  Sort $\mathcal{S}$ according to the PED if $|\mathcal{S}|>K$
  Move K candidates with smallest PED from $\mathcal{S}$ to $\mathcal{L}$ and empty $\mathcal{S}$
end for

3.2. IR-LSD algorithm

The increasing radius (IR)-LSD is listed as Algorithm 2. The increasing radius (IR)-SD algorithm [23] is a modification of Dijkstra's algorithm [22], which uses the metric-first search strategy [20,24,25]. The IR-LSD algorithm is optimal in the sense of visited number of nodes in the tree structure [23,25] and always extends the partial candidate with the lowest PED in one extend loop, but requires that the visited nodes are maintained in metric order to ensure the optimality, which requires the usage of memory and sorting [19]. The output candidate list $\mathcal{L}$ includes the candidates with lowest EDs.

Algorithm 2 $[\mathcal{L}]=$ IR-LSD$(\tilde{\mathbf{y}},\mathbf{R},N_{cand},\Omega,N_T)$

Initialize sets $\mathcal{S}$ and $\mathcal{L}$, and set $C_0=\infty$, $m=0$, $n_1=1$
Initialize $\mathcal{N}(\mathbf{s}=\mathbf{x}_{N_T}^{N_T},d(\mathbf{s})=0,n_2=2,i=N_T)-1$
while $C_0<d(\mathbf{s})$ **do**
  Determine the n_1th best node x_i for $\mathbf{s}_c=(x_i,\mathbf{x}_{i+1}^{N_T})^T$ and calculate $d(\mathbf{s}_c)$
  Determine the n_2th best node x_{i+1} for father candidate $\mathbf{s}_f=(x_{i+1},\mathbf{x}_{i+2}^{N_T})^T$ and calculate $d(\mathbf{s}_f)$ if $n_2\le|\Omega|$
  if $d(\mathbf{s}_c)<C_0$ **then**
    if $\mathbf{s}_c$ is a leaf node, i.e., $i=0$ **then**
      Store $\mathcal{N}_F(\mathbf{s}_c,d(\mathbf{s}_c))$ in $\{\mathcal{L}\}_m$

```
9:        Set m=m+1 or, if L is full, set m according to {L}_m with
          max ED and C_0=d(s)_m
10:       Continue with N(s=x_{i+1}^{N_T}, d(s), n_1=n_1+1, 1) if n_1+1 ≤ |Ω|
11:     else
12:       Store N_c(s_c, d(s_c), n_2=2, i=i−1) in S
13:     end if
14:   end if
15:   if N_f calculated and d(s_f) < C_0 then
16:     Store N_f(s_f, d(s)_f, n_2=n_2+1, i) in S
17:   end if
18:   Continue with N with min PED from S and set n_1=1
19: end while
```

4. Limiting the dynamic range of LLR

The soft output information $L_D(b_k)$ is summed from the extrinsic information $L_E(b_k|\mathbf{y})$, which is generated by the LSD, and the *a priori* information $L_A(b_k)$ as in (3). The extrinsic information $L_E(b_k|\mathbf{y})$ can be calculated for a system containing additive white Gaussian noise (AWGN) directly from the cost information known about the candidates as

$$L_E(b_k|\mathbf{y}) = \ln \frac{\sum_{\mathbf{x}\in\chi_{k,+1}} e^{-d(\mathbf{x})/N_0}}{\sum_{\mathbf{x}\in\chi_{k,-1}} e^{-d(\mathbf{x})/N_0}} = \ln \sum_{\mathbf{x}\in\chi_{k,+1}} e^{-d(\mathbf{x})/N_0} - \ln \sum_{\mathbf{x}\in\chi_{k,-1}} e^{-d(\mathbf{x})/N_0}, \tag{6}$$

where $\chi_{k,+1} = \{\mathbf{x}|b_k = +1\}$ is the set of $\Omega^{N_{T-1}}$ bit vectors $\mathbf{x}$ having b_k=+1, and $d(\mathbf{x})$ is the squared Euclidean distance between received vector $\mathbf{y}$ and lattice points $\mathbf{Hx}$. Eq. (6) can then be computed using the well-known Jacobian logarithm and a small look-up table [26].

The list sphere decoder uses a limited number of elements in the considered sets $\chi_{k,+1} \cap \mathcal{L}$ and $\chi_{k,-1} \cap \mathcal{L}$ by using the LSD output candidate list $\mathcal{L}$ to approximate the likelihood information $L_D(b_k)$ in (6), which decreases the complexity of the calculation of (6) significantly compared to the full set of candidates. The accuracy of the approximation depends on the quality of the candidate list $\mathcal{L}$ and the list size N_{cand}. The IR-LSD algorithm provides a list $\mathcal{L}$ including the most probable candidates, while the K-best-LSD algorithm does not guarantee that. This typically leads to a better approximation with the IR-LSD compared to the K-best-LSD with the same list size. If the size N_{cand} of the candidate list $\mathcal{L}$ is large enough so that both sets $\chi_{k,+1} \cap \mathcal{L}$ and $\chi_{k,-1} \cap \mathcal{L}$ include candidates for the bit b_k, the approximation of the $L_D(b_k)$ is typically accurate enough for adequate performance. However, the performance of the LSD may suffer due to too small a list size, and, thus, inaccurate $L_D(b_k)$ values. The error in the approximation of the $L_D(b_k)$ is especially large in the case where all the candidates in $\mathcal{L}$ for the bit b_k belong to either $\chi_{k,+1} \cap \mathcal{L}$ and $\chi_{k,-1} \cap \mathcal{L}$. In that case the approximation of one of the conditional probabilities $p(\mathbf{y}|b_k = \pm 1)$ goes to zero, which leads to an infinite value in (6). As the information is used as *a priori* information in the decoder, the decoder is most likely not able to correct the falsely detected signals.

4.1. Clipping methods

The effect of unreliable $L_D(b_k)$ may be reduced by limiting the $L_D(b_k)$ range, which is often called LLR clipping. Several LLR clipping methods for different algorithms have been proposed, e.g., in [10,27–29]. All the methods aim at reducing the detector algorithm complexity required to achieve certain performance. The LLR clipping in [10,27] is used to control the search effort used in the bit counterpart search. The method cannot be applied as such for LSD algorithms due to the differences in the algorithms. The LLR clipping in [28,29] is based on the reliability information and the channel state information (CSI) to determine close to optimal LLR clipping values with additional complexity. We introduce and study two simple methods to process $L_D(b_k)$ information and the impact of the methods on the performance of a coded system. Importantly, both presented methods are simple to implement and are suitable for VLSI implementation. The $L_{D1}(b_k)$ calculated in the detector is given as $L_{A2}(b_k)$ input to the decoder as illustrated in Fig. 1. By limiting the dynamic range of the variable, the decoder can overcome the wrong information given as $L_{A2}(b_k)$ in (3).

Method 1: A very simple way to prevent very large $L_D(b_k)$ values is to limit the dynamic range of $L_D(b_k)$ value as [1]

$$L_{D_{clip}}(b_k) = \begin{cases} L_D(b_k) & \text{if } |L_D(b_k)| \le L_{max}, \\ \mathrm{sgn}(L_D(b_k))L_{max} & \text{if } |L_D(b_k)| > L_{max}, \end{cases} \tag{7}$$

where $L_{D_{clip}}(b_k)$ is the clipped likelihood information and L_{max} is the selected maximum value for $|L(b_k)|$.

Method 2: The other method is slightly different compared to the first one. The $L_D(b_k)$ values are clipped to L_{max} if a threshold value of $L_{limit} > L_{max}$ is exceeded as

$$L_{D_{clip}}(b_k) = \begin{cases} L_D(b_k) & \text{if } |L_D(b_k)| \le L_{limit}, \\ \mathrm{sgn}(L_D(b_k))L_{max} & \text{if } |L_D(b_k)| > L_{limit}. \end{cases} \tag{8}$$

The main idea of Method 2 is to clip only the very large $L_D(b_k)$ values, which are due to small LSD list size, and bypass the $L_D(b_k)$ values where both bit values are present in (6). This is achieved by setting the L_{limit} value large enough. Graphical illustrations of both methods are shown in Figs. 2(a) and (b).

4.2. Numerical examples

We studied the impact of the two different LLR clipping methods on the performance of the system via Monte Carlo simulations. A MIMO-OFDM system model was assumed with 512 subcarriers (N_{sub}=300 used)

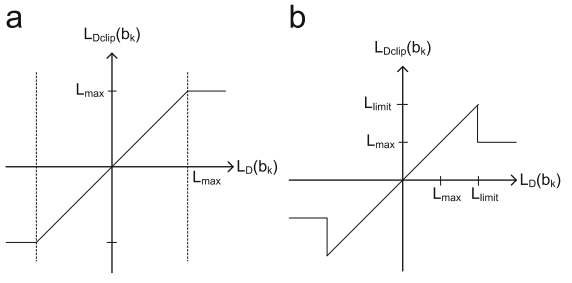

Fig. 2. A graphical illustration of the LLR clipping methods. (a) Method 1 and (b) Method 2.

according to the 3G long term evolution (LTE) parameters [30]. A bit-interleaved coded modulation (BICM) [4,31] with 1/2 rate $[13_o,15_o]$ turbo code was applied in a high correlated (CORR) typical urban (TU) 6 tap channel with a user velocity of 120 kmph. The system was operating with 5 MHz bandwidth at a carrier frequency of 2.4 GHz. The K-best-LSD and the IR-LSD with the SQRD [18] preprocessing were considered for detection and a max-log-MAP turbo decoder with 8 decoder iterations (DI) was used for decoding. The K-best-LSD was applied with $C_0=\infty$.

We illustrate the distribution behavior of the calculated LLR values with SNR and a histogram of the IR-LSD soft-output LLR with $N_{\text{cand}}=1024$ and 8 and with different SNR values are shown in Fig. 3. The studied $E_S/N_0=18$ and 26 dB cases reflect to the low and high throughput operating points of the system, respectively. The LLR values were limited to $L_{\max}=100$ to show the very large and infinite values. It can be seen that with increasing SNR the deviation of LLR widens and we can see more high LLR values, but the probability of $|L_D|>40$ and $|L_D|>20$ is fairly low with the higher and lower SNR values, respectively. The effect of the small list size results as a large amount of $L_{\max}$ values in the histogram as the probability of empty set $\chi_{k,+1}\cap\mathcal{L}$ and $\chi_{k,-1}\cap\mathcal{L}$ in the calculation of $L(b_k)$ is significantly higher, which can be seen as peaks in ± 100 values in 3D plot.

We also studied the impact of different clipping methods and $L_{\max}$ values on the performance of the system to determine the optimal clipping method and the threshold value to be used for clipping. It should be noted that while too low a value for $L_{\max}$ prevents the detector to give any significant *a priori* information to the decoder, too high a value for $L_{\max}$ enables the possibility for too high *a priori* information caused by unreliable or too small $\mathcal{L}$ which the decoder cannot correct in the case of a wrong decision. Performance of the IR-LSD with $N_{\text{cand}}=8$ and real K-best-LSD with $N_{\text{cand}}=64$ with both clipping methods applied are shown for a 4×4 16-QAM system in Fig. 4. The horizontal axis is the SNR as E_S/N_0 and the vertical axis is the throughput in bits per second, i.e., equal to $1-FER$ multiplied by the number of bits per channel use. Method 2 is applied with $L_{\text{limit}}=100$ to clip only the very large $L_D(b_k)$ values. The results show that the performance of a system is clearly improved by applying LLR clipping to limit the effect of the moderate LLR approximation compared to the system without clipping. We also notice that there is no significant performance difference between the two clipping methods with the IR-LSD. However, we noticed from the results that Method 1 is clearly better with the K-best-LSD. The reason for this is the different outputs from IR-LSD and K-best-LSD. The IR-LSD gives the most probable candidates as an output, and, thus, the LLR approximation is rather good and reliable in the cases where both bits are present in (6). The K-best-LSD output, however, may result in a poor LLR approximation also when candidates for both bits are present in (6). Thus, we conclude that Method 1 is a good choice to be applied as it requires less dynamic range for the $L(b_k)$ before clipping. The simulation results show that $L_{\max}=8$ gives the best performance, which means that the dynamic range of probability $P(b_k=\pm 1|\mathbf{y})$ is limited between [0.0003,0.9997].

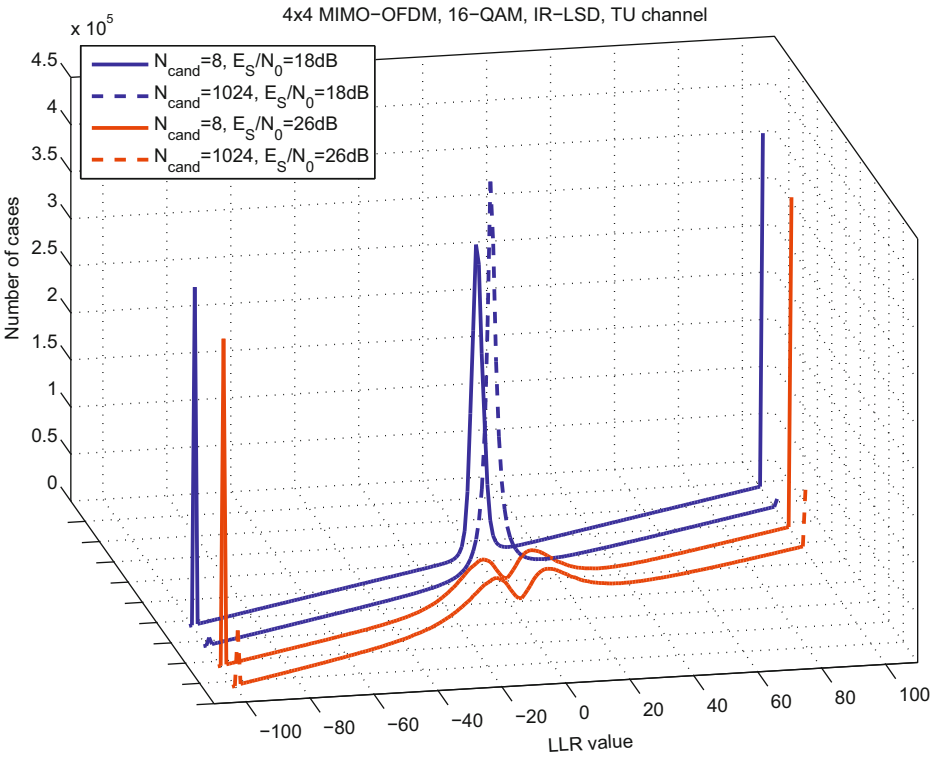

Fig. 3. A histogram of the resulted LLR values with IR-LSD with different list sizes and SNR values in 4×4 antenna system with 16-QAM.

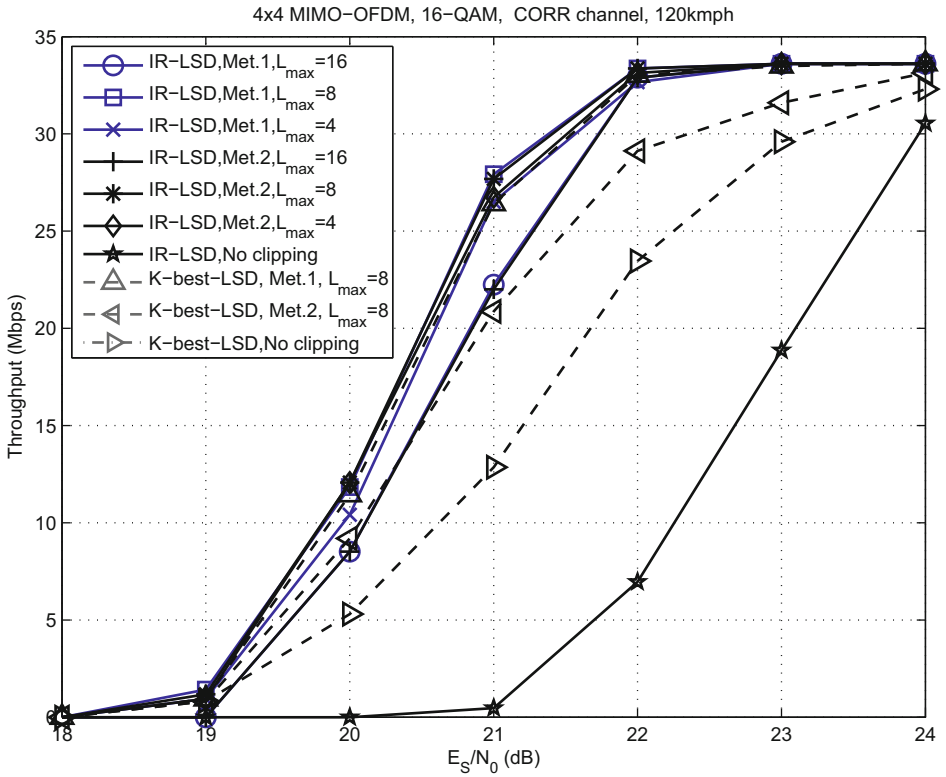

Fig. 4. Throughput vs SNR: performance of the IR-LSD and the real K-best-LSD with different LLR clipping methods and values in 4×4 antenna system with 16-QAM.

We also studied if and how the code rate possibly inter-plays with the optimal $L_{\max}$ value at the detector. The performance of a IR-LSD based system with code rates 1/3 and 4/5 and with different Method 1 clipping values is shown in Fig. 5. We can see that with a lower code rate 1/3 the value $L_{\max}=6$ gives the best performance with a difference of 0.1 dB compared to $L_{\max}=8$. The performance of the system with a higher code rate 4/5 is maximized with the value $L_{\max}=10$, but the difference from the value $L_{\max}=8$ is approximately 0.05 dB. The results indicate that as the decoder has more parity bits to be used in the decoding, the decoder should rely less on the *a priori* information $L_{A2}(b_k)$ from the detector and the $L_{\max}$ can be set to be a lower value. However, the differences are rather small, and in practice, $L_{\max}=8$ gives good results.

The LLR clipping enhances the performance of LSD based systems with low list size and, thus, impacts on the required list size, which leads to algorithm complexity reduction as observed also in [1,10]. We applied Method 1 with $L_{\max}=8$ for LLR clipping and the performance was studied in two channel models, a highly correlated and an uncorrelated (UNC) TU channel, in order to study the effect of the channel correlation and also compared to a soft output LMMSE detector [2]. Performance examples of a 4×4 16-QAM system with the IR-LSD is shown in Fig. 6. It can be seen that the required list size with the IR-LSD decreases significantly with LLR clipping applied and, e.g., the required list size of IR-LSD decreases from 64 to 8 in a 4×4 16-QAM system. Also the performance of the soft output LMMSE detector suffers significantly compared to the IR-LSD in the highly correlated channel. We also noticed that the required list size with the K-best-LSD does not decrease as significantly as with IR-LSD. The benefit of the LLR clipping with the real and complex[1] K-best-LSD is smaller than that with the IR-LSD, because of the breadth-first search strategy which usually leads to having both $b_k=+1$ or $b_k=-1$ candidates in the LLR calculation, but does not provide the most probable candidates. Thus, K-best-LSD requires a larger list size compared to the IR-LSD to obtain as accurate LLR approximation and performance. However, it can be noted that the quality of the obtained list increases as the channel is uncorrelated, i.e., the tree search is easier. The required list sizes were determined for 2×2 and 4×4 antenna cases with 4-QAM, 16-QAM, and 64-QAM, and the results are concluded in Table 1.

5. Comparison of real and complex-valued signal model

The SD and LSD algorithms are often assumed to apply a real equivalent system model [8,12,32,33] especially in the implementation of the algorithms. However, complex-valued signal models are also applied in the literature [1,9,10]. The definition of the signal model does not affect

[1] The real and complex versions are rigorously defined and considered in more detail in Section 5.

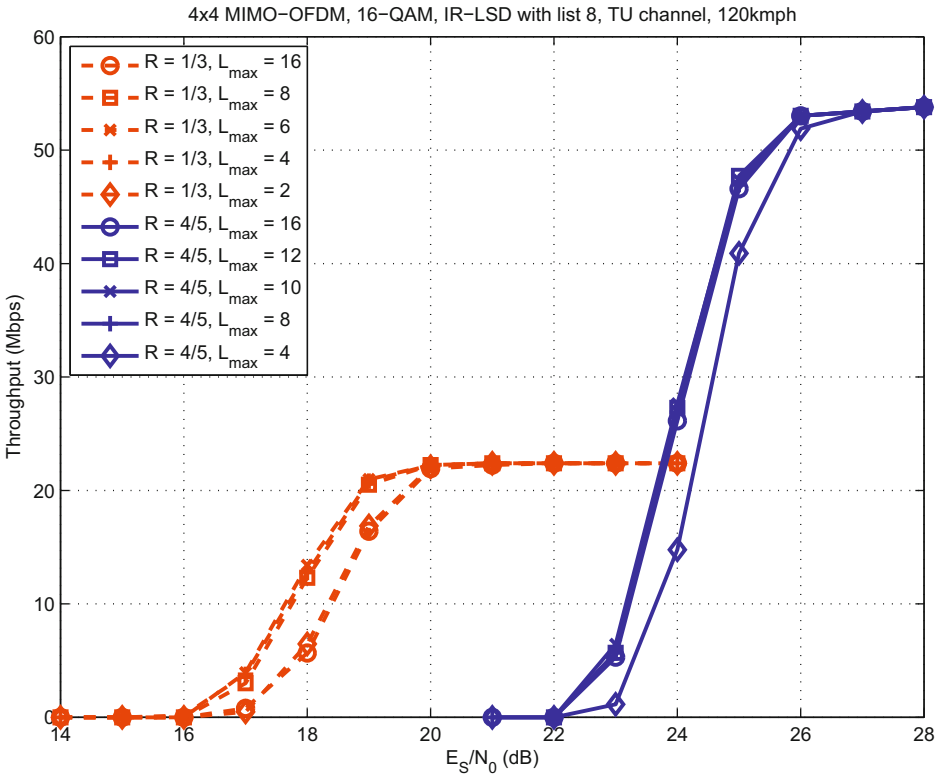

Fig. 5. Throughput vs SNR: performance of the IR-LSD with different code rates and LLR clipping values in 4×4 antenna system with 16-QAM.

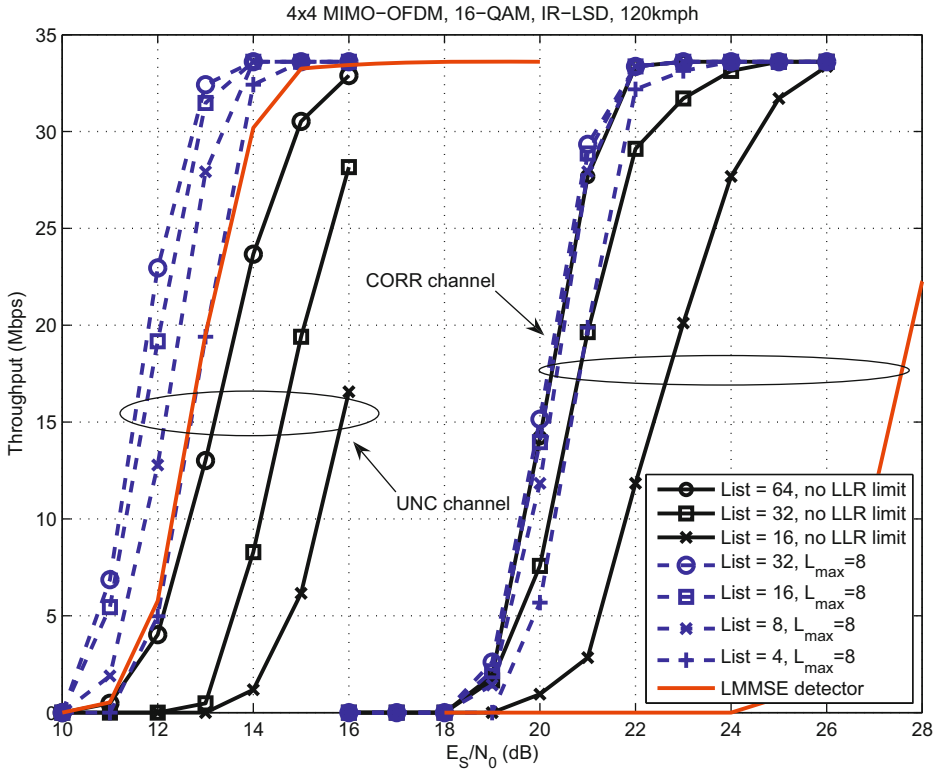

Fig. 6. Throughput vs SNR: performance of the IR-LSD with different list sizes in 4×4 antenna system with and without LLR clipping.

Table 1
List sizes for IR-LSD[a], real K-best-LSD[b] and complex K-best-LSD[c] with LLR clipping.

	2×2	4×4
4-QAM	Not studied	$N_{cand}=8^a/32^b/32^c$
16-QAM	$N_{cand}=8^a/16^b/16^c$	$N_{cand}=8^a/64^b/128^c$
64-QAM	$N_{cand}=16^a/64^b/64^c$	$N_{cand}=16^a/128^b/256^c$

the mathematical equivalence of the expressions, but it affects on the lattice definition where the LSD algorithm search is executed. Thus, there is a need to justify the selection of the signal model by studying the effect of it on the LSD search and on the total complexity of the search. The complex MIMO system model in (1) can be presented as an equivalent real model as follows:

$$\begin{bmatrix}\mathrm{Re}(\mathbf{r})\\ \mathrm{Im}(\mathbf{r})\end{bmatrix}=\begin{bmatrix}\mathrm{Re}(\mathbf{H}) & -\mathrm{Im}(\mathbf{H})\\ \mathrm{Im}(\mathbf{H}) & \mathrm{Re}(\mathbf{H})\end{bmatrix}\begin{bmatrix}\mathrm{Re}(\mathbf{x})\\ \mathrm{Im}(\mathbf{x})\end{bmatrix}+\begin{bmatrix}\mathrm{Re}(\boldsymbol{\eta})\\ \mathrm{Im}(\boldsymbol{\eta})\end{bmatrix}. \quad (9)$$

Then the new real dimensions are defined as $M_T=2N_T$, $M_R=2N_R$, and the real symbol alphabet is now $\Omega_R\subset\mathbb{Z}$, e.g., $\Omega_R\in\{-3,-1,1,3\}$ in the case of 16-QAM.

The complexity of the LSD algorithms is relative to the number of visited nodes in the search tree and the size of the search tree, and, as already mentioned, the size of the search tree depends on the applied signal model. The use of a real system model doubles the depth of the search tree, i.e., $M_T=2N_T$, but decreases the number of branches at each level compared to the complex-valued signal model, i.e., $|\Omega_R|=|\Omega|^{1/2}$. Thus, the total number of branches in the search tree with real-valued signal model is given as $B_R=\sum_{i=1}^{M_T}|\Omega_R|^i$, and with complex-valued signal model as $B_C=\sum_{i=1}^{N_T}|\Omega|^i$. The sizes of the real and complex search trees approach each other as the number of transmit antennas N_T and the constellation size $|\Omega|$ increase, but the search tree size is larger with real-valued signal model with moderate N_T and $|\Omega|$, e.g., in a 4×4 system with 16-QAM B_R=87 380 and B_C=69 904. Thus, it is likely that the number of visited nodes by the LSD algorithm increases somewhat with the real-valued signal model.

The LSD algorithms considered were described in detail in Section 3. The main complexity of the LSD algorithm comes from the PED calculation in (5), which is executed for each studied node. When considering the choice of the signal model, we should notice that the complexity of the PED calculation in (5) includes different operations with real and complex valued signals. The numbers of operations required to calculate (5) depend on the number of transmit antennas N_T and the current layer i in the search tree and they are listed as real operations in Table 2 given that one complex multiplication (MUL) is equal to three real MULs and five real additions (ADD) and one complex ADD is equal to two real ADDs. As a numerical example, we assume that N_T=4 and the average studied node in both the real and the complex tree is in the middle of the tree depth, i.e., $E[i_R]=M_T/2=4$ and $E[i_C]=N_T/2=2$. Then the number of required real operations on average for the PED calculation is 9 MULs and 21 ADDs for the complex-valued signal model, and 5 MULs and 5 ADDs for the real-valued signal model. Thus, we can say that on average the complexity of one LSD algorithm node check in a system with N_T=4 is approximately double with the complex-valued signal model compared to the real-valued signal model.

Table 2
The number of real operations used for PED calculation in (5).

	Real-valued signal model	Complex-valued signal model
MUL	$2N_T-i+1$	$3(N_T-i+1)$
ADD	$2N_T-i+1$	$7(N_T-i+1)$

Table 3
Number of visited nodes with the real[a] K-best-LSD and the complex[b] K-best-LSD with the list sizes given in Table 1.

	2×2	4×4
4-QAM	not studied	$254^a/212^b$
16-QAM	$148^a/272^b$	$1364^a/4368^b$
64-QAM	$1096^a/4160^b$	$5704^a/36928^b$

5.1. Numerical examples

We studied the impact of the real and the complex-valued signal models on the number of visited nodes by considered LSD algorithms via Monte Carlo simulations. The simulations were executed with the same parameters as those in Section 4.2.

The number of visited nodes by the K-best-LSD algorithm depends on the signal model and the applied list size $K=N_{cand}$. The K-best-LSD algorithm visits a fixed number of nodes given the list size and the fact that no sphere radius is introduced, i.e., $C_0=\infty$. The number of visited nodes by the real K-best-LSD and the complex K-best-LSD algorithms are determined as $V_R=\sum_{i=1}^{M_T}|\mathcal{S}|\,|\Omega_R|$ and $V_C=\sum_{i=1}^{N_T}|\mathcal{S}|\,|\Omega|$, where $|\mathcal{S}|$ is the number of stored candidates at each layer as in Algorithm 1. The numbers of visited nodes in different antenna and constellation cases with list sizes determined in Table 1 are listed in Table 3. It can be seen that the K-best-LSD with the real-valued signal model visits less nodes in all the cases except the 4×4 with QPSK case compared to the complex-valued signal model with the same performance. The reason for this is the difference in possible signal points in one layer between the real and the complex-valued signal model, i.e., $|\Omega_R|=|\Omega|^{1/2}$. As the algorithm visits all the possible child nodes of the stored partial candidates at each layer, the search with the real-valued signal model is done with less visited nodes in total even though visiting double the number of layers.

The number of visited nodes by the metric-first search LSD algorithm varies with channel realization and SNR. We studied the number of visited nodes by the IR-LSD algorithm with both the real and the complex-valued signal model by collecting data from the Monte Carlo

simulations. The data are then plotted as a histogram to illustrate the distribution of the number of visited nodes by the LSD algorithm. The number of visited nodes was studied in a high correlated and uncorrelated TU channels to also examine the effect of the channel realization. The number of visited nodes by the IR-LSD algorithm with SQRD preprocessing in a 4×4 system with 16-QAM is shown in Fig. 7. The results show that the algorithm with real and complex-valued signal model visits approximately the same amount of nodes. It can also be noted that the channel realization and correlation properties of the channel have an effect on the distribution of the visited nodes with the IR-LSD algorithm. The Monte Carlo results also confirmed that the distribution of the number of visited nodes of the real and complex-valued signal model is similar also as the N_T and Ω increase.

The results clearly show that there is a difference in terms of complexity whether the real valued or the complex valued signal model is used and that the channel scenario affects the distribution of the number of visited nodes by the metric first search IR-LSD algorithm. The real-valued signal model is clearly the better choice to be applied with the LSD algorithms given the similar distribution in the number of visited nodes with both the real and the complex-valued signal model and that the complexity of the real-valued signal model in required operations is much less complex compared to the complex-valued signal model. Also it should be noted that if the LSD algorithm search is highly limited to lower the latency, the use of complex valued signal model might lead to higher detector throughput as in [9]. However, in order to achieve close to max-log-MAP performance, the LSD algorithm search cannot be limited too much as we will show in Section 6.

6. Limited search

In the hardware implementation of an algorithm for a practical system, there is usually a predetermined time to execute the process that the algorithm carries out. In order to reserve the hardware resources for the algorithm to meet the given timing constraints, we need to determine the so-called worst case scenario and determine the complexity accordingly. From the LSD algorithms considered, the K-best-LSD checks a fixed number of nodes when $C_0=\infty$ and, thus, the complexity of the algorithm is fixed. The IR-LSD, however, visits a variable number of nodes depending on the channel realization, and the hardware implementation of algorithms as such is not feasible in a system with a fixed latency requirement.

In order to fix the complexity of the IR-LSD algorithms, we propose a simple way to modify the algorithms to limit the maximum number of visited nodes by the LSD algorithm, which we call limited search (LS). The *while* loop in the algorithm description of the IR-LSD in Algorithm 2 can be replaced with a *for* loop, which means that a predefined maximum number of loop runs L_{node} is set. If the sphere search is not completed within the defined maximum limit L_{node}, the algorithm is stopped

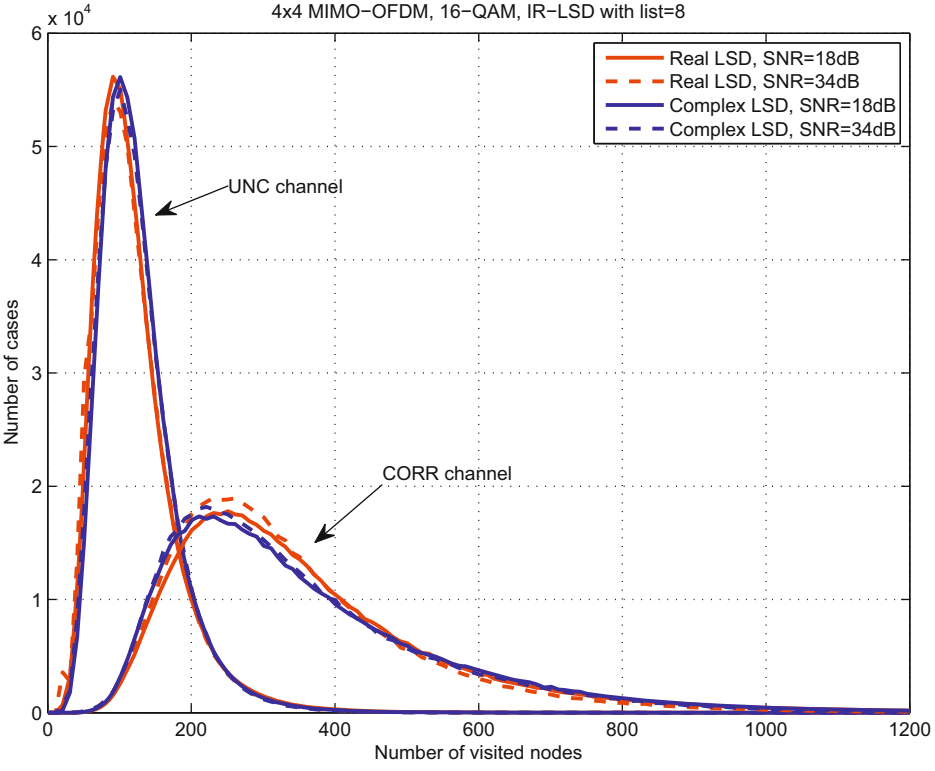

Fig. 7. Histogram of the number of visited nodes per symbol vector with real and complex IR-LSD in TU channel with 4×4 16-QAM system.

and the current final candidate list $\mathcal{L}$ is given as an output. Another more sophisticated alternative is to use a scheduling algorithm as, e.g., in [10,34]. We modified the scheduling algorithm in [10] to be more suitable for the IR-LSD and call it as scheduled search (SS). We use the algorithm to determine the search limit $L_{\mathrm{node}}(n)$ for n th subcarrier in OFDM symbol as

$$L_{\mathrm{node}}(n) = N_{\mathrm{sub}}L_{\mathrm{avg}} - \sum_{i=1}^{n-1} L_{\mathrm{node}}(i) - (N_{\mathrm{sub}} - n)L_{\mathrm{min}}, \tag{10}$$

where $N_{\mathrm{sub}}L_{\mathrm{avg}}$ is the total node run-time constraint for the whole OFDM symbol and L_{min} is minimum number of studied nodes reserved for each subcarrier. The $L_{\mathrm{node}}(n)$ is also upper limited to maximum of DL_{avg} nodes, where D is a node coefficient, to prevent too much resources to be used in one subcarrier. The idea behind the scheduled search is that the algorithm is able to allocate higher maximum limits $L_{\mathrm{node}}(n)$ for subcarriers that have a channel realization resulting in low SNR while subcarriers with easier channel realization can be allocated with lower limits $L_{\mathrm{node}}(n)$.

6.1. Numerical examples

We studied the effect of limited search and scheduling algorithm via Monte Carlo simulations to determine the performance of the method. The numerical examples were executed with the same parameters as in Sections 4.2 and 5.1, and including LLR clipping with $L_{\mathrm{max}}=8$ and the real-valued signal model. First we studied the number of visited nodes in the search tree by the IR-LSD algorithm to determine the initial values for limit parameters. The histograms of the number of visited nodes in a 4×4 system with 16-QAM and the IR-LSD with list size $N_{\mathrm{cand}}=8$ in TU channels are shown in Fig. 7. It can be noted that the channel realization and especially the correlation properties of the channel affect to the distribution of the number of visited nodes by the LSD algorithms and it should be taken into account when determining the proper limit parameter values for the search.

Then the impact of limiting the IR-LSD algorithm search with both LS and SS methods was studied on the performance of the system. Numerical examples of the performance with different limited search parameter values in a 4×4 system with 16-QAM and in TU channels are shown in Fig. 8. It can be seen that there is no performance loss with the real-valued IR-LSD algorithm as the LS method limit L_{node} is set high enough, i.e., $L_{\mathrm{node}}=200/500$ in UNC/CORR channel. The performance degradation of the system with the LSD search limit set to $L_{\mathrm{node}}=500$ nodes, which is a rather low limit compared to the determined distribution of visited nodes $X_{IR}^{(CORR,18\,\mathrm{dB})}$, is about 0.2–0.3 dB at maximum compared to the LSD with unlimited search. The CDF with determined L_{node} equals to 80.6%, i.e., $\Pr[X_{IR}^{(CORR,18\,\mathrm{dB})} \le L_{\mathrm{node}} = 500] = 0.806$. The corresponding value in uncorrelated channel, $L_{\mathrm{node}}=160$, equals to 79.5% of the CDF. As a comparison, the common

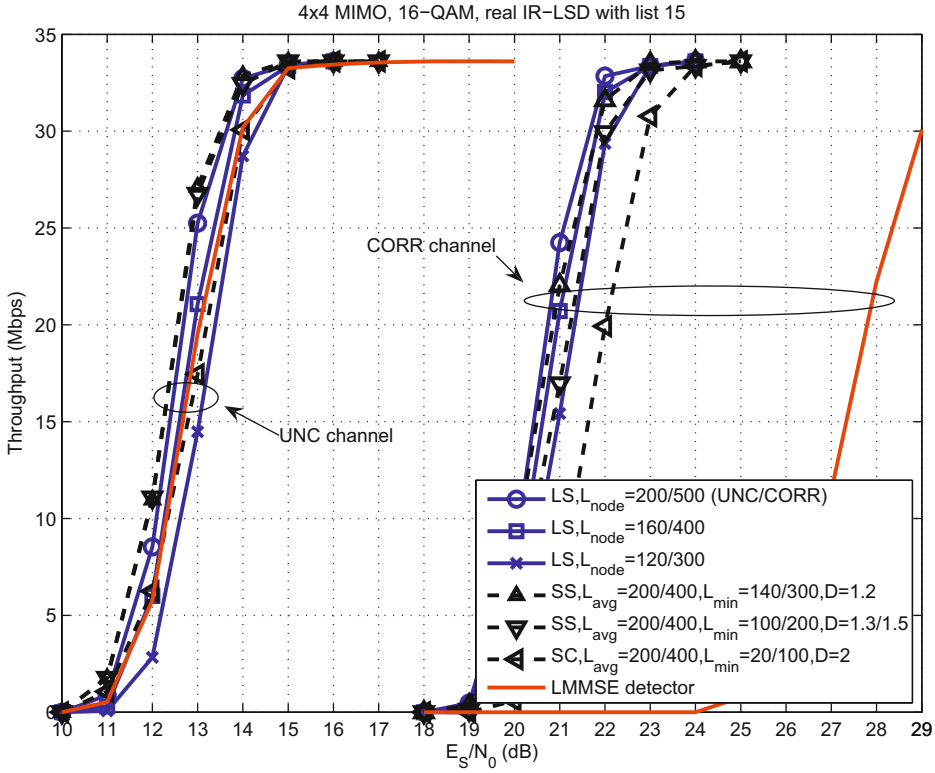

Fig. 8. Throughput vs SNR: performance of the real-valued IR-LSD with limited maximum number of nodes in 4×4 16-QAM system.

depth-first search LSD [16] requires the L_{node} to be set to 98% in CDF according to our studies. However, the depth-first search LSD is a less complex compared to the IR-LSD, and does not require any sorting operations. The good tolerance of the IR-LSD algorithm to the LS method is due to the metric-first search strategy, where the algorithm proceeds uniformly in the search tree. A more sophisticated SS method was applied with L_{avg}=200/400 and with different L_{min} and D values for UNC/CORR channel. The SS method can outperform the simple LS method slightly, i.e., by ≈ 0.1 dB, with proper parameter configuration when the same resources are applied L_{node}=L_{avg}. In practice, L_{min} has to be set high enough and D not too large to guarantee enough nodes for each subcarrier detection.

The simulation results show clearly that the metric search IR-LSD algorithm works fine with limited search with only minor performance loss. A minor performance gain can be achieved with more complex SS method compared to less complex LS method, but both methods are feasible for implementation. The results indicate that the CDF of the visited nodes can be used as a guideline for determining the proper L_{node} value. We determined the proper search limits L_{node} for the algorithms for high correlated TU channels and they are listed in Table 4. It should be noted that the highly correlated channel scenario is difficult for the sphere search, and the listed results can be thought as the so-called worst case scenario limits. The results show that the channel correlation properties affect the distribution of nodes in the algorithm, and a feasible search limit L_{node} can be set lower in an uncorrelated channel.

Table 4
Determined maximum node limits for the real IR-LSD with LLR clipping in highly correlated TU channel.

	2×2	4×4
16-QAM	80	500
64-QAM	200	1000

7. Complexity and performance of an iterative receiver

The optimal joint receiver can be approximated by using an iterative receiver and soft-input soft-output (SISO) detector and decoder [1]. The performance of the system can be increased to a certain extent by executing global iterations (GIs), where the soft reliability information is fed back to detector from the decoder. However, the iterative receiver structure also increases the computational complexity of the receiver with each GI as more signal processing is done. The effect of the receiver convergence properties depends on multiple variables such as decoder iterations, channel code properties and channel realization [35]. Thus, it is not straightforward to determine the optimal receiver configuration and how much computing effort should be used in detector, decoder and global iterations in total.

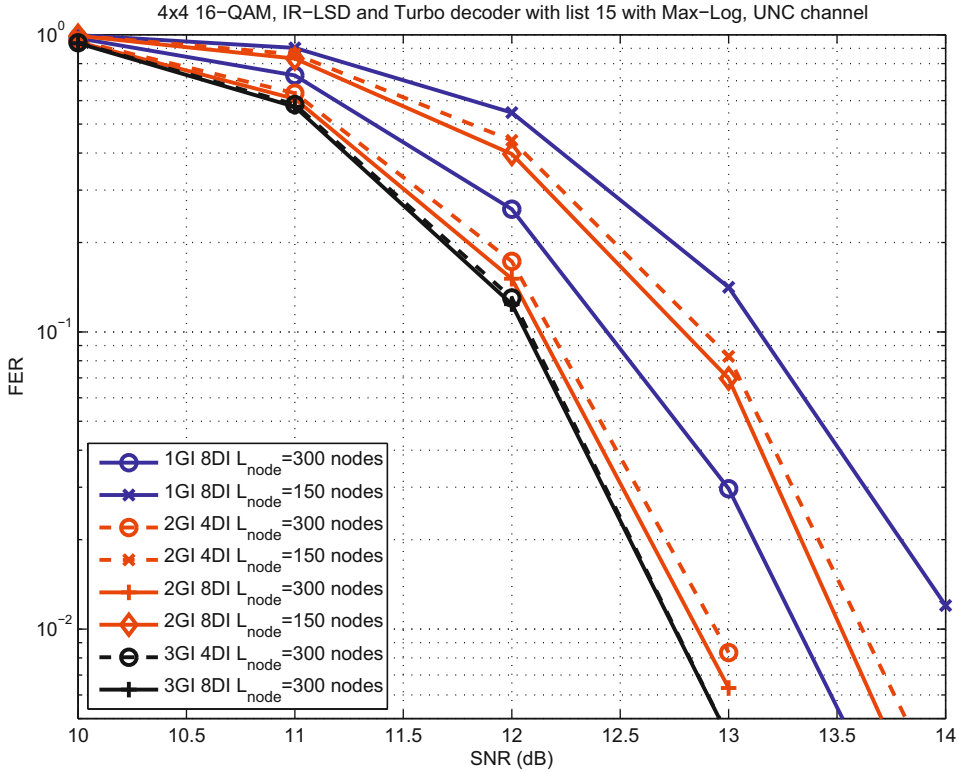

Fig. 9. FER vs SNR: performance of the real IR-LSD based receiver in 4×4 system with 16-QAM.

We studied the performance of the iterative receiver with variable GIs and their effect on the computational complexity by numerical examples. The numerical examples were executed for a 4×4 MIMO-OFDM system with 16-QAM and with the same typical system model parameters as in earlier sections with the following details. The LLR clipping was applied with $L_{\max}$=8, the IR-LSD was operating with variable L_{node}, the max-log turbo decoder was operating with 4 or 8 decoder iterations (DI) and iterative receiver was operating with 1–3 GIs. Performance examples of a real IR-LSD based receiver are shown in Fig. 9. We can see that the 2nd GI improves the performance approximately 0.5 dB at FER 10^{-2} and there is not much performance gain with 3rd GI. It can be also noted that 4 DIs are sufficient with 2 or more GIs and the increase in the IR-LSD L_{node} from 150 nodes to 300 nodes improves performance by approximately 0.7 dB. Similar performance behavior could also be seen from the numerical examples with the K-best-LSD based receiver.

Typically, the performance and the complexity of the receiver are both important measures in system design. Therefore, we also calculated the required computational complexity for different receiver configurations to compare and determine the most efficient configuration. The complexity calculation includes the required arithmetic operations of LSDs and max-log turbo decoder, which are MUL, ADD, comparison (COMP) and division (DIV), with given number of iterations for reception of one OFDM frame, i.e., 2400 transmitted bits. The memory requirements and word length requirements were omitted from the calculations. In order to get an explicit complexity description, we approximated the computational complexity of the required arithmetic operations according to their relative complexity units (CU) to NAND2 operation, i.e., according to how many 1-bit NAND2 gates are required for corresponding 1-bit operation. The approximations used are listed in Table 5 and are based on authors' experience and on [36].

Table 5
The number of 1-bit NAND2 gates used for corresponding 1-bit arithmetic operation.

Operation	MUL	ADD	COMP	DIV
Required NAND2's	12	5	4	30

The performance and complexity of K-best-LSD based receiver with different configurations is shown in Fig. 10. The curves describe the required SNR for 4% target FER and the corresponding complexity of the receiver with variable LSD L_{node} or list size N_{cand} in calculated complexity units. We noted with both K-best-LSD and IR-LSD based receivers that the complexity from the decoder iterations dominates the total complexity and that there is only a minor performance gain with 8 DI compared to 4 DI when 2 or more GIs are executed. Also the performance gain of the 3rd GI with the given system configuration is minor compared to the required additional computational complexity. The cost of the increased number of visited nodes by the LSD is minor compared to the additional performance gain with both

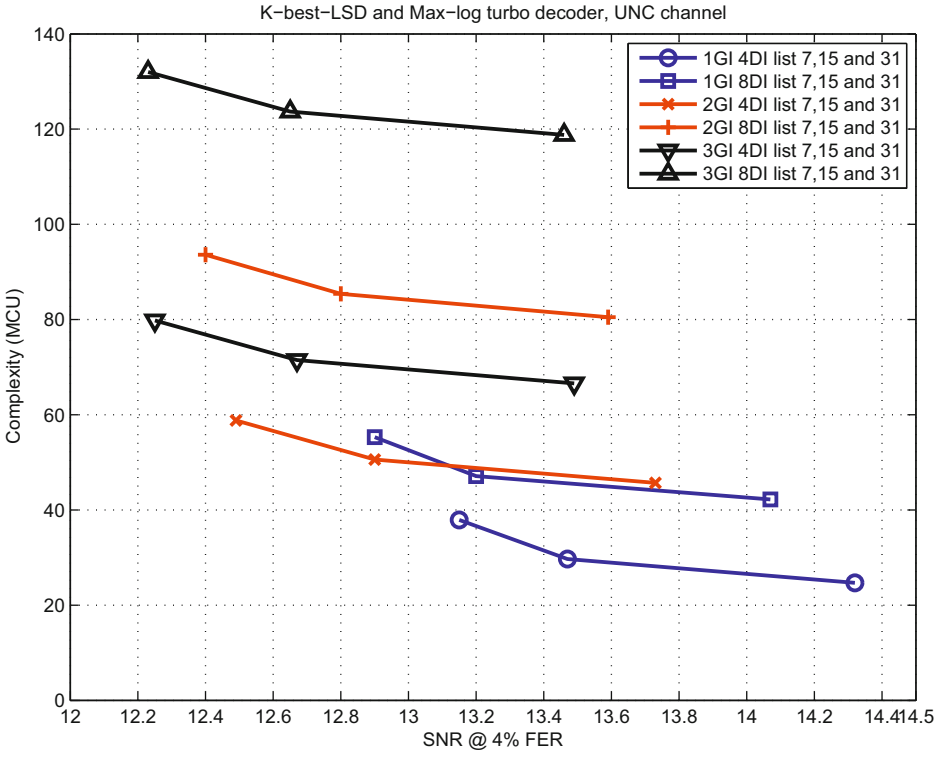

Fig. 10. Required SNR for 4% target FER vs. required operation complexity with different K-best-LSD based configurations.

IR-LSD and K-best-LSD. Thus, we can say that the additional GIs, especially the 2nd GI, are justified in terms of complexity and performance tradeoff and they should be considered in receiver design. Also it is justified to use lower amount of decoder resources (4DI) per GI when additional GIs are executed. The numerical examples should be taken as design guidelines and the exact setup depends on system configuration [35].

8. Conclusions

We considered two different LSD algorithms and the algorithm design aspects related to the implementation. We showed that the use of LLR clipping enables the use of lower list sizes with LSD, and, thus, decreases the required algorithm complexity. The simple clipping introduced as Method 1 with $L_{\max}=8$ value is a good choice to be applied in practice. We compared the use of real and complex-valued signal models in the LSD algorithm, and showed the real valued signal model to be less complex and a better choice for LSD algorithms. The complexity of the metric first search IR-LSD algorithm can be constrained by limiting the maximum number of visited nodes in the sphere search without sacrificing the performance of the system. It was noted that the channel realization affects the CDF of the visited nodes, and the IR-LSD is much less sensitive to limited search compared to the common depth first algorithm. We also studied the complexity and performance of an iterative receiver, analyzed the tradeoff choices between complexity and performance, and showed that the additional computational cost in LSD is justified to get better soft-output approximation.

The results presented clarify some LSD algorithm design choices in implementation and give general guidelines to the implementation aspects of the tree search algorithm design. Future research will address the architecture design and the implementation of the LSD algorithms in hardware and verification of the performance in an FPGA based test bed to get a fair comparison of LSD algorithms' true costs.

Acknowledgements

The authors would like to thank Dr. B. Hochwald, Mr. P. Silvola, Mr. J. Antikainen, Mr. J. Ylioinas, Mr. E. Kunnari, and Mr. Y. Sun for helpful comments and fruitful discussions. This work was done in MITSE project which was supported in part by Tekes, the Finnish Funding Agency for Technology and Innovation, Nokia, Nokia Siemens Networks, Elektrobit, and Uninord.

References

[1] B. Hochwald, S. ten Brink, Achieving near-capacity on a multiple-antenna channel, IEEE Trans. Commun. 51 (3) (2003).
[2] G. Foschini, Layered space–time architecture for wireless communication in a fading environment when using multi-element antennas, Bell Labs Tech. J. 1 (2) (1996) 41–59.
[3] H. Artes, D. Seethaler, F. Hlawarsch, Efficient detection algorithms for MIMO channels: a geometrical approach to approximate ML detection, IEEE Trans. Signal Process. 51 (11) (2003) 2808–2820.
[4] P. Fertl, J. Jalden, G. Matz, Capacity-based performance comparison of MIMO-BICM demodulators, in: IEEE 9th Workshop on Signal Processing Advances in Wireless Communications, 2008, SPAWC 2008, 2008, pp. 166–170.
[5] G.J. Foschini, M.J. Gans, On limits of wireless communications in a fading environment when using multiple antennas, Wireless Pers. Commun. 6 (1998) 311–335.
[6] P.W. Wolniansky, G.J. Foschini, G.D. Golden, R.A. Valenzuela, V-BLAST: an architecture for realizing very high data rates over the rich-scattering wireless channel, in: International Symposium on Signals, Systems, and Electronics (ISSSE), Pisa, Italy, 1998, pp. 295–300.
[7] U. Fincke, M. Pohst, Improved methods for calculating vectors of short length in a lattice, including a complexity analysis, Math. Comput. 44 (5) (1985) 463–471.
[8] M.O. Damen, H.E. Gamal, G. Caire, On maximum-likelihood detection and the search for the closest lattice point, IEEE Trans. Inform. Theory 49 (10) (2003) 2389–2402.
[9] A. Burg, M. Borgmann, M. Wenk, M. Zellweger, W. Fichtner, H. Bölcskei, VLSI implementation of MIMO detection using the sphere decoding algorithm, IEEE J. Solid-State Circuits 40 (7) (2005) 1566–1577.
[10] C. Studer, A. Burg, H. Bölcskei, Soft-output sphere decoding: algorithms and VLSI implementation, IEEE J. Select. Areas Commun. 26 (2) (2008) 290–300.
[11] D. Garrett, L. Davis, S. ten Brink, B. Hochwald, G. Knagge, Silicon complexity for maximum likelihood MIMO detection using spherical decoding, IEEE J. Solid-State Circuits 39 (9) (2004) 1544–1552.
[12] Z. Guo, P. Nilsson, Algorithm and implementation of the K-best sphere decoding for MIMO detection, IEEE J. Select. Areas Commun. 24 (3) (2006) 491–503.
[13] M. Wenk, A. Burg, M. Zellweger, C. Studer, W. Fichtner, VLSI implementation of the list sphere algorithm, in: 24th Norchip Conference, 2006, 2006, pp. 107–110.
[14] M. Myllylä, J. Cavallaro, M. Juntti, A list sphere detector based on Dijkstra's algorithm for MIMO-OFDM systems, in: Proceedings of the IEEE International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Athens, Greece, September 12–19, 2007, pp. 1–5.
[15] M. Myllylä, J. Antikainen, J. Cavallaro, M. Juntti, The effect of LLR clipping to the complexity of list sphere detector algorithms, in: Asilomar Conference on Signals, Systems and Computers, Monterey, USA, November 4–7, 2007, pp. 1559–1563.
[16] M. Myllylä, M. Juntti, J. Cavallaro, Implementation aspects of list sphere detector algorithms, in: Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM), Washington, DC, USA, November 26–30, 2007, pp. 3915–3920.
[17] J. Hagunauer, E. Offer, L. Papke, Iterative decoding of binary block and convolutional codes, IEEE Trans. Inform. Theory 42 (2) (1996) 429–445.
[18] D. Wübben, R. Böhnke, V. Kühn, K. Kammeyer, MMSE extension of V-BLAST based on sorted QR decomposition, in: Proceedings of the IEEE Vehicular Technology Conference (VTC), vol. 1, Orlando, FL, 2003, pp. 508–512.
[19] J. Anderson, S. Mohan, Sequential coding algorithms: a survey and cost analysis, IEEE Trans. Commun. 32 (2) (1984) 169–176.
[20] A. Murugan, H. El Gamal, M. Damen, G. Caire, A unified framework for tree search decoding: rediscovering the sequential decoder, IEEE Trans. Inform. Theory 52 (3) (2006) 933–953.
[21] K. Wong, C. Tsui, R.-K. Cheng, W. Mow, A VLSI architecture of a K-best lattice decoding algorithm for MIMO channels, in: Proceedings of the IEEE ISCAS'02, vol. 3, Helsinki, Finland, 2002, pp. 273–276.
[22] E.W. Dijkstra, A note on two problems in connexion with graphs, in: Numerische Mathematik, vol. 1, Mathematisch Centrum, Amsterdam, Netherlands, 1959, pp. 269–271.
[23] W. Xu, Y. Wang, Z. Zhou, J. Wang, A computationally efficient exact ML sphere decoder, in: Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM), vol. 4, 2004, pp. 2594–2598.
[24] F. Jelinek, J.B. Anderson, Instrumental tree encoding of information sources, IEEE Trans. Inform. Theory 17 (1) (1971) 118–119.
[25] S. Mohan, J.B. Anderson, Computationally optimal metric-first code tree search algorithms, IEEE Trans. Commun. 32 (6) (1984) 710–717.
[26] P. Robertson, E. Villebrun, P. Hoeher, A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain, in: Proceedings of the IEEE International Conference on Communications (ICC), 1995, pp. 1009–1013.

[27] M.S. Yee, Max-log-MAP sphere decoder, in: IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005, Proceedings (ICASSP '05), vol. 3, 2005, pp. iii/1013–iii/1016.
[28] E. Zimmermann, D. Milliner, J. Barry, G. Fettweis, Optimal LLR clipping levels for mixed hard/soft output detection, in: Global Telecommunications Conference, 2008, IEEE GLOBECOM 2008, IEEE, New Orleans, USA, 2008, pp. 1–5.
[29] D. Milliner, E. Zimmermann, J. Barry, G. Fettweis, Channel state information based LLR clipping in list MIMO detection, in: IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications, 2008, PIMRC 2008, 2008, pp. 1–5.
[30] 3rd Generation Partnership Project (3GPP), TSGR1#41 R1-050-520, EUTRA downlink numerology, Technical Report, 3rd Generation Partnership Project (3GPP), 2005.
[31] G. Caire, G. Taricco, E. Biglieri, Bit-interleaved coded modulation, IEEE Trans. Inform. Theory 44 (3) (1998) 927–946.
[32] C. Dick, K. Amiri, J. Cavallaro, R. Rao, Design and architecture of spatial multiplexing MIMO decoders for FPGAs, in: 2008 42nd Asilomar Conference on Signals, Systems and Computers, 2008, pp. 160–164.
[33] S. Seo, S. Park, Efficient VLSI implementation of the list sphere decoder with real-value based tree searching method, in: Advanced Communication Technology, 2006, ICACT 2006, The 8th International Conference, vol. 3, 2006, pp. 1694–1697.
[34] A. Burg, M. Borgmanr, M. Wenk, C. Studer, H. Bolcskei, Advanced receiver algorithms for MIMO wireless communications, in: Design, Automation and Test in Europe, 2006. DATE '06. Proceedings, vol. 1, 2006, pp. 593–598.
[35] S. ten Brink, Convergence behavior of iteratively decoded parallel concatenated codes, IEEE Trans. Commun. 49 (6) (2001) 1727–1737.
[36] M.J.S. Smith, Application-Specific Integrated Circuits (VLSI Systems Series), Addison-Wesley Longman Inc, Boston, Massachusetts, 1997.