Reduced-Complexity MIMO Detection via a Slicing Breadth-First Tree Search

Sangwook Suh, *Member, IEEE*, and John R. Barry, *Senior Member, IEEE*

***Abstract*— A bottleneck in multiple-input multiple-output communications systems is the complexity of detection at the receiver. The complexity of optimum maximum-likelihood detection is often prohibitive, especially for large numbers of antennas and large alphabets. A suboptimal tree-search-based detector known as the K-best detector is an effective scheme that provides a flexible performance-complexity tradeoff. In this paper, we identify scalar list detection as a key building block of the K-best detector, and we propose an efficient low-complexity implementation of the scalar list detector for M-ary QAM using a slicing operation. Embedding the slicing list detector into the K-best framework leads to our proposed slicing K-best detector. Simulation results show that the proposed detector offers comparable performance to the conventional K-best detector, but with significantly reduced complexity when K is less than the QAM alphabet size M. Since the slicing list detection is performed at each visited node in the detection tree, the complexity reduction is especially significant when the number of antennas and the alphabet size are large, making the proposed detector a competitive option for high spectral-efficiency wireless systems.**

***Index Terms*— Slicing, K-best detection, multiple-input multiple-output, spatial multiplexing, list error probability.**

I. Introduction

THE push for higher spectral efficiency in multiple-input multiple-output (MIMO) communications systems and standards is leading to an increase in both the number of antennas and the alphabet size. The IEEE 802.11ac wireless LAN standard [1] supports up to $N = 8$ spatial streams through a MIMO channel using M-ary QAM with alphabet size as large as $M = 256$. Also, the next-generation (5G) cellular network [2] will use carrier frequencies above 6 GHz, enabling an increase in the number of antennas with half-wavelength spacing in a small form factor. The increased spectral efficiency comes at the price of increased complexity of the MIMO detector at the receiver, which grows rapidly with N and M. For example, an exhaustive search maximum-likelihood (ML) detector that considers all M^N possible combinations of transmitted symbols quickly becomes prohibitive for large N and M.

Manuscript received May 17, 2016; revised October 30, 2016 and January 7, 2017; accepted January 10, 2017. Date of publication January 17, 2017; date of current version March 8, 2017. This work was supported by the National Science Foundation under Grant CNS-1513884. The associate editor coordinating the review of this paper and approving it for publication was L. K. Rasmussen.

The authors are with the Department of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: swsuh@gatech.edu; john.barry@ece.gatech.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TWC.2017.2654344

A variety of suboptimal MIMO detectors with reduced complexity have been proposed [3]. The zero-forcing (ZF) and minimum mean-squared-error (MMSE) linear detectors have low complexity but inferior performance. Ordered successive interference cancellation (OSIC) detectors, also known as vertical Bell Laboratories layered space-time (V-BLAST) detectors [4], [5] perform better than linear detectors, but still fall significantly short of ML performance. The parallel detector (PD) [6] considers all possible candidates for the first symbol, then separately implements an OSIC detector based on each candidate for the first symbol to recover the remaining symbols. Accordingly, the PD scheme provides better performance than the OSIC scheme at the cost of increased complexity. Lattice-reduction aided detectors use the Lenstra-Lenstra-Lovasz algorithm [7] or Seysen's algorithm [8] as a preprocessing step to transform the channel matrix into a more orthogonal channel, but the high complexity of lattice reduction is a burden for fast fading channels. A semidefinite relaxation based scheme rephrases MIMO detection as a convex optimization problem, but the complexity is still high when M is large [9].

Many MIMO detectors are cast as a search through an M-ary tree with N layers. For example, the sphere decoder is a depth-first tree search strategy based on Fincke-Pohst [10], [11] or Schnorr-Euchner [12], [13] enumeration that can approach the performance of the ML detector, but with high complexity that varies depending on the channel and noise. The K-best detector is an effective strategy for searching a tree [14]–[19], not only because it provides a flexible performance-complexity tradeoff, but also because it has a fixed complexity that is independent of the channel and noise, which makes it amenable to implementation. However, as the alphabet size M increases, the complexity of the K-best scheme rapidly increases, due to the fact that the number of candidates in each layer is scaled up by a factor M. The processing delay of the K-best scheme can be reduced by exploiting parallel computations at the cost of larger number of functional units, and also by simplifying the Euclidean-norm metric computations [15], but nether reduces the size of the search space.

Selective spanning with fast enumeration (SSFE) [20]–[22] is a method aimed at reducing the search space of the K-best scheme. With selective spanning, the value of K decreases as the scheme progresses through the layers of the tree. Fast enumeration is realized by first quantizing the complex input to the closest alphabet symbol, then obtaining the set of $K - 1$ neighboring symbols from the quantized symbol. For a 4×4 MIMO system using 64-QAM, it is claimed

in [20] and [21] that with the help of channel coding, the coded BER performance of SSFE with $K = [4, 2, 1, 1]$ in each layer is similar to that of the K-best scheme with $K = 4$, and that SSFE with $K = [8, 4, 2, 1]$ is similar to K-best with $K = 8$. However, the performance of detector itself is degraded due to having less number of candidates in the later layers. Also, using a small value of K at the final layer can lead to unreliable log-likelihood ratio (LLR) estimates for soft-output detection [23].

In this paper, we propose a modification of the K-best detector for the case of QAM alphabets that significantly reduces complexity when $K < M$, without compromising performance. The proposed detector is based on a scalar list detector for M-ary QAM that uses a slicing operation to approximate the minimum-distance list detector, a modification that provides two benefits: it reduces the number of metric computations for each node from M to K, and it reduces the number of candidates to be sorted at each layer from KM to K^2. Simulation results show that the proposed slicing K-best scheme provides almost identical performance to that of the conventional K-best scheme, but with significantly reduced complexity. Since the slicing list detection is performed at each node in every layer of the detection process, the complexity reduction is especially significant for a MIMO system with a large number of antennas N and with a large alphabet size M.

The following section presents the channel model and a description of the tree structure, along with a summary of the conventional K-best MIMO detector. In Sect. III, we examine the problem of scalar list detection. In Sect. IV, we rephrase the K-best detector in terms of the scalar list detector, and propose the slicing K-best MIMO detector. In Sect. V, we provide simulation results, and Sect. VI summarizes the conclusions of our work.

II. Channel Model and The K-Best MIMO Detector

A. Channel Model and the Detection Tree

We consider a linear memoryless MIMO channel with N transmit antennas and N_r ($\geq N$) receive antennas, whose complex baseband model is given by:

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

where $\mathbf{y} = [y_1, \ldots, y_{N_r}]^T$ is the received vector, $\mathbf{x} = [x_1, \ldots, x_N]^T$ is the transmitted vector whose elements are independently chosen from the same QAM alphabet $x_\ell \in \mathcal{A}$ of size $|\mathcal{A}| = M$, $\mathbf{H}$ is the $N_r \times N$ channel matrix with rank N, and $\mathbf{z} = [z_1, \ldots, z_{N_r}]^T$ is the noise vector. We assume throughout the paper a QAM alphabet whose size M is an even power of 2, with real and imaginary parts consisting of the odd integers from $-\sqrt{M}+1$ to $\sqrt{M}-1$, so that the energy of the alphabet is $\mathcal{E} = \frac{2(M-1)}{3}$. Also, we assume a Rayleigh fading channel, so that the entries of $\mathbf{H}$ are i.i.d. zero-mean complex Gaussian random variables with unit variance, and we assume additive white Gaussian noise (AWGN), in which the entries of $\mathbf{z}$ are i.i.d. zero-mean complex Gaussian random variables with variance σ^2. The signal-to-noise ratio (SNR) at the receiver is then given by $\frac{N\mathcal{E}}{\sigma^2}$.

The ML detector chooses its decision $\mathbf{x} \in \mathcal{A}^N$ to minimize

$$J_N(\mathbf{x}) = \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 \tag{2}$$

$$= \left\|\hat{\mathbf{y}} - \mathbf{R}\mathbf{x}\right\|^2 = \sum_{\ell=1}^{N} |\hat{y}_\ell - \sum_{j=1}^{\ell} R_{\ell j} x_j|^2, \tag{3}$$

where we have introduced $\hat{\mathbf{y}} = \mathbf{Q}^H\mathbf{y}$, where $\mathbf{Q}$ is the unitary matrix in the QR decomposition $\mathbf{H} = \mathbf{QR}$ of the channel matrix, and where $\mathbf{R}$ is a lower-triangular matrix with positive real diagonal elements. We choose $\mathbf{R}$ to be lower triangular for the sake of simpler presentation, since an upper-triangular choice would require that the entries of $\hat{\mathbf{y}}$ and $\mathbf{x}$ be processed in a reverse order.

We can interpret (3) as the sum of N *branch metrics* in a path through an M-ary tree with N layers, whose M^N leaf nodes represent all possibilities for $\mathbf{x} \in \mathcal{A}^N$. The ℓ-th layer of the tree has M^ℓ nodes, one for each possible combination of $\{x_1, \ldots, x_\ell\}$. To the branch from a node at the $(\ell-1)$-th layer to a child at the ℓ-th layer we assign a metric according to (3), which can be rewritten as:

$$BM_\ell = |\tilde{y}_\ell - R_{\ell\ell}x_\ell|^2, \tag{4}$$

where $\tilde{y}_\ell = \hat{y}_\ell - \sum_{j=1}^{\ell-1} R_{\ell j}x_j$ represents the observation for the ℓ-th layer after the contributions from the previous layer symbols $\{x_1, \ldots, x_{\ell-1}\}$ have been subtracted.

B. The K-Best MIMO Detector

The K-best algorithm [14]–[16], also known as the QRD-M algorithm [17]–[19], is a breadth-first strategy for searching the tree. It proceeds through the tree one layer at a time, starting at the root, and at each layer it only hands over the surviving K nodes with the smallest cost to the next layer.

The pseudocode for the K-best detector is given in Algorithm 1. Here an ordered vector $c = [x_1, \ldots, x_\ell] \in \mathcal{A}^\ell$ is used to represent a candidate node at the ℓ-th layer of the tree, with the null set ϕ representing the root node at layer zero, and the set of all transmit vectors $\mathcal{A}^N$ representing the M^N leaf nodes at layer N. The cost of a candidate node c at the ℓ-th layer is denoted $J_\ell(c)$. The set of all candidates (before pruning) at a given layer is denoted by $\mathscr{C}$, while the set of all survivor nodes (after pruning) is denoted by $\mathcal{S}$.

Lines 1 and 2 of the pseudocode initialize the survivor as the root node with zero cost. The algorithm then steps through the N layers of the tree, where for each layer it forms a set $\mathscr{C}$ of candidate nodes comprising the M children of each of the survivor nodes from the previous layer, and further computes their associated costs. At the first layer of the tree there are $|\mathscr{C}| = M$ candidates, while at all remaining layers there are $|\mathscr{C}| = KM$ candidates. The K best candidates from $\mathscr{C}$ are declared to be the survivors in line 14. After reaching the last layer of the tree, the leaf node with the smallest metric is declared to be the detected symbol set in line 16.

In lines 7-11, the algorithm adds all M children of each surviving node to the candidate set $\mathscr{C}$. In fact, however, there is no need to extend all M children of a node, since at most K children from each node will ever survive the pruning

Algorithm 1 The K-Best MIMO Detector

$\mathcal{S} = \phi$
$J_0(\phi) = 0$
for $\ell = 1$ **to** N **do**
  $\mathscr{C} = \phi$
  for $s \in \mathcal{S}$ **do**
    $\tilde{y}_\ell = \hat{y}_\ell - \sum_{j=1}^{\ell-1} R_{\ell j} s_j$
    for $x_\ell \in \mathcal{A}$ **do**
      $c = [\ s \mid x_\ell\ ]$
      $J_\ell(c) = J_{\ell-1}(s) + |\tilde{y}_\ell - R_{\ell\ell} x_\ell|^2$
      $\mathscr{C} \leftarrow \mathscr{C} \cup c$
    end for
  end for
  if $\ell < N$ **then**
    $\mathcal{S} =$ **prune**$(\mathscr{C})$, keeping only the K best
  else
    $\hat{\mathbf{x}} =$ **Best**$(\mathcal{S})$
  end if
end for

process to the next layer. This suggests a *prune-as-you-go* modification of the K-best detector, in which the M branches from each survivor node are immediately pruned to the K best branches, without consideration of any other survivor nodes. This modification reduces the number of branches from each survivor node to K, and thus reduces the total number of metrics to be computed and sorted in each layer from KM to K^2. While it might seem that this reduction in complexity is partially offset by the additional complexity introduced by the intermediate pruning steps, we will see in the following section that these pruning steps can be accomplished with negligible complexity using our proposed slicing list detector.

III. Scalar List Detection

The prune-as-you-go approach in the previous section leads to an alternative implementation of the K-best detector built upon a scalar *list detector*. In particular, the combination of (i) extending a survivor node to its M children, and (ii) pruning to the K best, can be viewed as a single instance of a scalar list detector. Specifically, we define a list detector of length K for an M-ary alphabet $\mathcal{A}$ as a mapping from $\mathbb{C}$ to a subset $\mathscr{L} \subset \mathcal{A}$ of size $|\mathscr{L}| = K$. The list detector is thus like a conventional hard-output detector, except that it produces a list of multiple alphabet symbols as its decision instead of just one. The remainder of this section examines the design and performance of the scalar list detector.

A. The Minimum-Distance List Detector

Given a complex input p, the *minimum-distance list* detector produces the list of K elements of the alphabet that are closest to p, based on the squared Euclidean distances $\{|p - x|^2 : x \in \mathcal{A}\}$.

As an example of minimum-distance list detection, consider the case of $M = 16$ QAM. Fig. 1 shows the decision regions of the minimum-distance list detector for six different values of K. In each case, the complex plane is partitioned into a finite number of disjoint *decision regions*, so-named because every point in a given decision region get mapped to the same decision list. With the decision regions and decision lists defined, the problem of mapping a given point p to its minimum-distance list can be transformed to determining in which of the decision regions the given point p is located. However, the irregular shapes of the decision regions make the task of identifying the correct decision region to be computationally intensive, with no obvious alternative to a brute-force approach for each point p of first computing then sorting the M metrics $\{|p - x|^2 : x \in \mathcal{A}\}$.

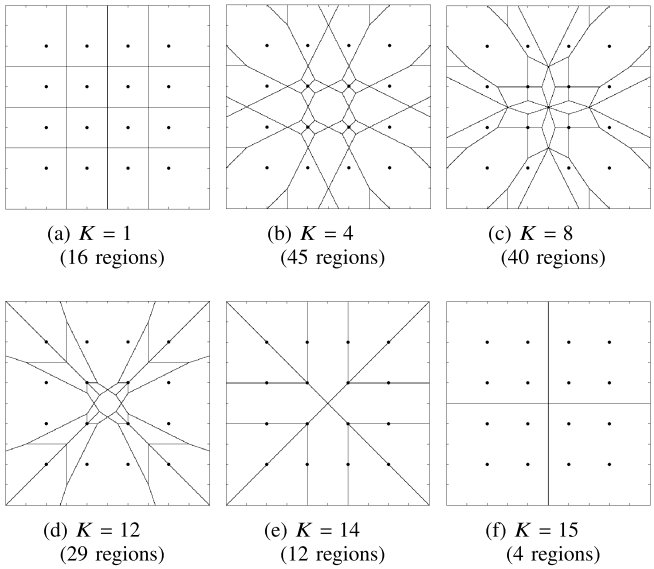


Fig. 1. The decision regions of the minimum-distance list detector for 16-QAM with $K \in \{1, 4, 8, 12, 14, 15\}$.

B. The Slicing List Detector

We propose a suboptimal list detector with significantly reduced complexity called the *slicing list* detector. Like the minimum-distance list detector, the slicing list detector has a finite number of decision regions, and every point in a given region gets mapped to the same decision list. Unlike the minimum-distance detector, however, we impose the constraint that the thresholds that divide adjacent decision regions be either *horizontal* or *vertical* lines. This simple constraint enables us to determine in which decision region a given complex point lies by a simple operation commonly known as *slicing*: by separately comparing the real and imaginary parts of the given point with a set of fixed threshold values. By constructing a lookup table that maps each decision region to the corresponding decision list, the slicing list detector can be implemented with extremely low complexity.

There are two primary degrees of freedom in the design of a slicing list detector: (i) choosing the slicing thresholds which defines the decision regions, and (ii) choosing the decision list that corresponds to each decision region. These two choices are examined in the following two subsections.

1) Slicing Thresholds: Given that the real and imaginary parts of the M-QAM symbols consist of the odd integers

from $-\sqrt{M}+1$ to $\sqrt{M}-1$, we propose that the threshold values for both the real and imaginary parts be placed at all integers from $-\sqrt{M}$ to $\sqrt{M}$. This choice leads to dividing the complex plane into a total of $4(\sqrt{M}+1)^2$ sub-regions, $4M$ of which in the middle are square in shape, while $4(1+2\sqrt{M})$ at the edges and the corners are semi-infinite rectangular in shape.

2) Decision Lists: Once the thresholds are defined, it remains to identify the decision list of K alphabet symbols with each corresponding decision region. We propose that the list $\mathscr{L}_i$ associated with the i-th sub-region $\mathcal{R}_i$ be chosen so as to minimize the following mean-squared-error (MSE) metric of the difference in Euclidean distances:

$$\mathscr{L}_i = \arg\min_{\mathscr{L}} \mathbf{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\{\hat{d}_k(p) - d_k(p)\right\}^2 \;\middle|\; p \in \mathcal{R}_i\right], \quad i = 1, \cdots, 4(\sqrt{M}+1)^2, \qquad (5)$$

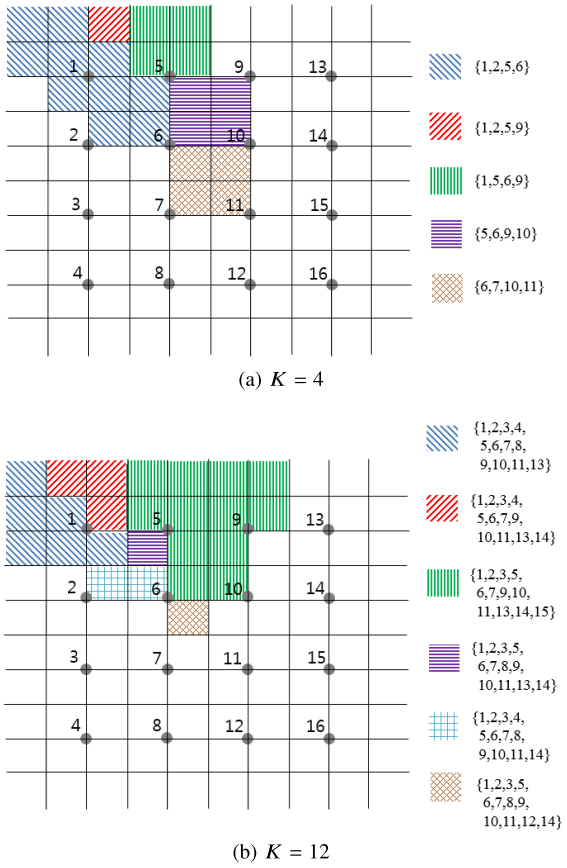
where $\hat{d}_1(p) \le \hat{d}_2(p) \le \cdots \le \hat{d}_K(p)$ are the sorted distances from the point $p \in \mathbb{C}$ to each of the K symbols in the list $\mathscr{L}$, and $d_1(p) \le d_2(p) \le \cdots \le d_K(p)$ are the sorted distances from the point p to each of the closest K symbols in the minimum-distance list. Here, the point p is a complex random variable that is defined by $p = x + n$, where $x \in \mathcal{A}$ is a uniformly distributed complex symbol, and n is zero-mean circularly-symmetric complex Gaussian noise.

Strictly speaking, the decision lists found through (5) are a function of SNR. Nevertheless, in practice we have achieved good performance by obtaining a single list at a single nominal SNR value, then using that list for all encountered SNR values. This is possible because when a change in SNR causes the decision list for a complex point p to change, the change is at the end of the list (corresponding to the symbol furthest from p), where one distant symbol with a low probability of being transmitted is replaced by another distant symbol with a similarly low probability of being transmitted.

The decision lists of the slicing list detector for 16-QAM with $K = 4$ and $K = 12$ are illustrated in Fig. 2, where the thresholds are shown as solid black lines. In the figures, the decision lists associated with the decision regions are obtained by the criterion given in (5) at SNR $= 0$ dB. The decision lists for the remaining decision regions follow from symmetry considerations.

Fig. 3 shows the decision regions of the slicing list detector for 16-QAM with different values of K (assuming SNR $= 0$ dB). Comparing Fig. 3 to Fig. 1, we can see how the slicing list detector approximates the decision regions of the minimum-distance list detector, subject to its constraint that the boundaries be horizontal and vertical lines.

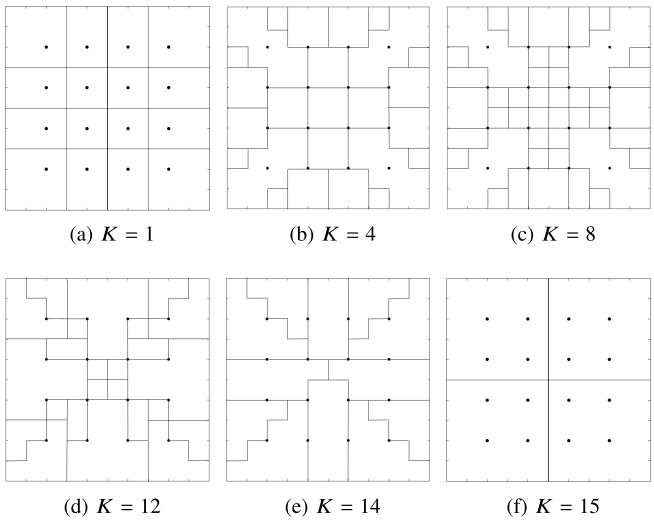
The performance of a scalar list detector is quantified by its *list error probability* (LEP), which is defined as the probability that the transmitted symbol is not included in the decision list [24]. Fig. 4 compares the LEP of the slicing list detector and the minimum-distance list detector in a Rayleigh fading channel for the case of 16-QAM. It can be observed that the LEP of the slicing list detector closely matches that of the minimum-distance list detector.



(a) $K = 4$

(b) $K = 12$

Fig. 2. The decision lists of the slicing list detector for 16-QAM with (a) $K = 4$ and (b) $K = 12$. The decision lists associated with each shaded region were found using (5).



(a) $K = 1$ (b) $K = 4$ (c) $K = 8$

(d) $K = 12$ (e) $K = 14$ (f) $K = 15$

Fig. 3. The decision regions of the slicing list detector for 16-QAM with $K \in \{1, 4, 8, 12, 14, 15\}$.

Also shown in Fig. 4 is the performance of an alternative design strategy for a list detector, in which the set of K candidates are determined by first quantizing the point p to the closest symbol, then obtaining the list of neighboring

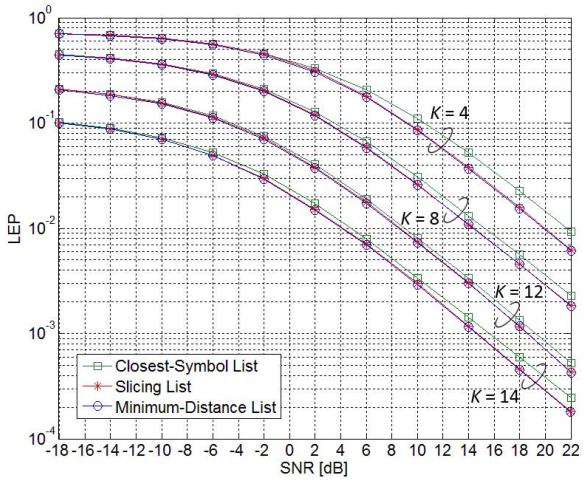


Fig. 4. The list error probability of the proposed slicing list detector (with thresholds and lists given in Fig. 2) closely matches with that of the minimum-distance list detector in a Rayleigh fading channel for 16-QAM with $K \in \{4, 8, 12, 14\}$.

$K-1$ symbols from the closest symbol. Since the distance between adjacent thresholds is fixed to the minimum distance between symbols, it suffers from performance degradation. Furthermore, the discrete Gaussian distribution of the point p is not accounted for in mapping the decision lists. Accordingly, Fig. 4 shows that the closest-symbol list detector suffers an SNR loss of between 0.5 dB and 1.8 dB, depending on the value of K.

IV. The Slicing K-Best MIMO Detector

The prune-as-you-go strategy from Sect. II can be built on the concept of scalar list detection from Sect. III, since the process of determining the K best of the M children from a given survivor node is equivalent to applying the interference-reduced channel observation for the node to the minimum-distance list detector of Sect. III-A. This leads to our proposed slicing K-best detector, presented in the pseudocode of Algorithm 2. Unlike Algorithm 1, which extends each survivor to all M children, Algorithm 2 uses a list detector to extend each survivor to only K of its children. If we were to use the minimum-distance list detector of Sect. III-A in line 7, then Algorithm 2 would produce exactly the same decisions as Algorithm 1. Instead, we propose to use the slicing list detector, such that **SlicingList**() in line 7 maps its input argument $\tilde{y}_\ell / R_{\ell\ell} \in \mathbb{C}$ to the decision list $\mathscr{L} \subset \mathcal{A}$ determined in Sect. III-B, where $|\mathscr{L}| = K$. At the first layer of the tree there are $|\mathscr{C}| = K$ candidates, while at all remaining layers there are $|\mathscr{C}| = K^2$ candidates. The previous path metric $J_{\ell-1}(s)$ in line 10 remains constant for all children from the same node, so adding the term does not alter the choice of K best children from each node. The K best survivors from K^2 candidates are determined in line 15 while taking into account K different values of the previous path metrics, so the ordering within the K children from each node is not required. Thus, a simple slicing operation that yields an *unsorted* list of K best children from each node suffices in line 7.

There are two immediate benefits from the fact that the slicing list detector in Algorithm 2 extends a survivor node only to K of its children, as opposed to all M children in Algorithm 1:

- The number of metrics computations in each layer is reduced by K/M.
- The number of metrics to be sorted to prune to K survivors in each layer is reduced by K/M.

It is known that sorting techniques have an average complexity of $O(n \log n)$ or larger, where n is the length of a sorting list. So, by reducing the size of a sorting list from KM to K^2, the sorting complexity is reduced by a factor of less than K/M. It is also noteworthy that there is no sorting and selection process required at the first layer in the proposed slicing K-best detector, since the K survivors from the first layer are solely determined by a simple slicing operation, and the ordering within the K survivors is again not required. The overall effect is a net reduction in complexity by a factor of less than K/M.

Algorithm 2 The Proposed Slicing K-Best MIMO Detector

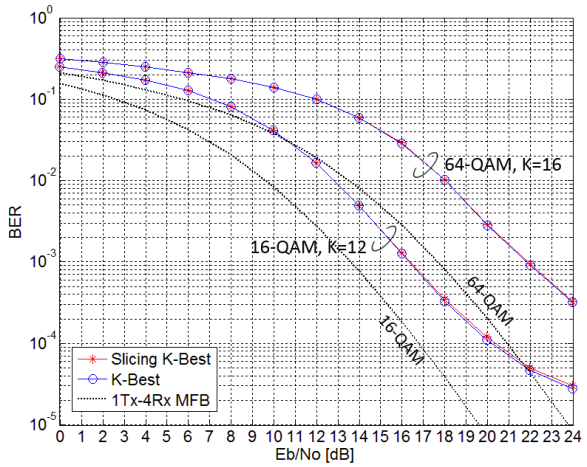
1: $\mathcal{S} = \phi$
2: $J_0(\phi) = 0$
3: **for** $\ell = 1$ **to** N **do**
4: $\quad \mathscr{C} = \phi$
5: $\quad$ **for** $s \in \mathcal{S}$ **do**
6: $\quad\quad \tilde{y}_\ell = \hat{y}_\ell - \sum_{j=1}^{\ell-1} R_{\ell j} s_j$
7: $\quad\quad \mathscr{L} =$ **SlicingList**$(\tilde{y}_\ell / R_{\ell\ell})$
8: $\quad\quad$ **for** $x_\ell \in \mathscr{L}$ **do**
9: $\quad\quad\quad c = [\ s \mid x_\ell \ ]$
10: $\quad\quad\quad J_\ell(c) = J_{\ell-1}(s) + |\tilde{y}_\ell - R_{\ell\ell} x_\ell|^2$
11: $\quad\quad\quad \mathscr{C} \leftarrow \mathscr{C} \cup c$
12: $\quad\quad$ **end for**
13: $\quad$ **end for**
14: $\quad$ **if** $\ell < N$ **then**
15: $\quad\quad \mathcal{S} =$ **prune**$(\mathscr{C})$, keeping only the K best
16: $\quad$ **else**
17: $\quad\quad \hat{\mathbf{x}} =$ **Best**$(\mathcal{S})$
18: $\quad$ **end if**
19: **end for**

V. Numerical Results

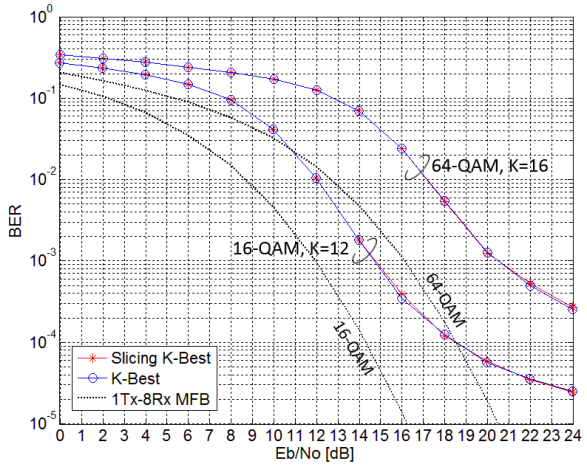
In this section we present simulation results for the performance and complexity of several MIMO detection schemes, including the conventional K-best and the proposed slicing K-best schemes. The performance is compared in terms of BER, and the complexity is compared in terms of the number of operations. In order to make fair comparisons of performance and complexity between various detection schemes, the simulations for the schemes that involve sequential detection process, such as OSIC, PD, K-best, and slicing K-best, all employ the optimal ordering for the sequential detection.

A. Uncoded BER Performance

Fig. 5 depicts the uncoded BER versus E_b/N_0, or SNR per bit, of the K-best detector and the slicing K-best detector for 16-QAM with $K = 12$ and 64-QAM with $K = 16$, respectively. Before the QR decomposition, the columns of



(a) 4×4 MIMO



(b) 8×8 MIMO

Fig. 5. Uncoded BER versus E_b/N_0 for (a) 4×4 MIMO and (b) 8×8 MIMO i.i.d. Rayleigh fading channel with Gray mapped 16-QAM and 64-QAM.

the channel matrix H are permuted based on the decreasing order of signal-to-interference-plus-noise ratio (SINR). We see from Fig. 5 that the BER curve for the proposed slicing K-best detector is nearly indistinguishable from the BER curve for the conventional K-best detector.

The black dashed curves indicate the theoretical BER curves of the matched-filter bound (MFB) [25] based on a single-input multiple-output (SIMO) channel. Comparing the BER curves of K-best and slicing K-best schemes to the MFB suggests that both schemes achieve almost full receiver diversity gain up to a certain SNR, even with $K < M$. This indicates that the slicing K-best schemes can be a good match for a coded system, as it performs relatively well with low complexity at lower SNR. At higher SNR, the diversity gain gets lower for both schemes.

B. Coded BER Performance

Soft outputs in the form of LLR estimates required for soft-decision channel decoding can be computed by considering the set $\mathcal{S}$ of all K survivor vectors at the final layer, each vector with N symbols. The LLR estimates can be obtained from the following, based on the max-log approximation [26], [27]:

$$\begin{aligned} &LLR(b_{i,\ell}|\mathbf{y}) \\ &= \ln \frac{P(b_{i,\ell}=1|\mathbf{y})}{P(b_{i,\ell}=0|\mathbf{y})} \\ &= \ln \frac{\sum_{\mathbf{x}\in S^1_{i,\ell}\subset\mathcal{S}} \exp\left(-\frac{1}{2\sigma^2}||\mathbf{y}-\mathbf{Hx}||^2\right)}{\sum_{\mathbf{x}\in S^0_{i,\ell}\subset\mathcal{S}} \exp\left(-\frac{1}{2\sigma^2}||\mathbf{y}-\mathbf{Hx}||^2\right)} \\ &\approx \max_{\mathbf{x}\in S^1_{i,\ell}\subset\mathcal{S}} \left(-\frac{1}{2\sigma^2}||\mathbf{y}-\mathbf{Hx}||^2\right) \\ &\quad - \max_{\mathbf{x}\in S^0_{i,\ell}\subset\mathcal{S}} \left(-\frac{1}{2\sigma^2}||\mathbf{y}-\mathbf{Hx}||^2\right) \\ &= \frac{1}{2\sigma^2}\left(\min_{\mathbf{x}\in S^0_{i,\ell}\subset\mathcal{S}} ||\mathbf{y}-\mathbf{Hx}||^2 - \min_{\mathbf{x}\in S^1_{i,\ell}\subset\mathcal{S}} ||\mathbf{y}-\mathbf{Hx}||^2\right), \end{aligned} \tag{6}$$

where $b_{i,\ell}$ denotes the i-th bit of the symbol in the ℓ-th layer, and $S^1_{i,\ell}$, $S^0_{i,\ell} \subset \mathcal{S}$ denote the set of symbol vectors in which the ℓ-th symbol of each vector has either 1 or 0 at the i-th bit. The computed LLRs are then fed into the subsequent soft-decision channel decoder, such as a low-density parity-check (LDPC) decoder.

Fig. 6 shows the (64800, 32400)-LDPC coded BER versus E_b/N_0 of the K-best detection and the slicing K-best detection schemes for 16-QAM with $K = 12$ and 64-QAM with $K = 16$, respectively. In both cases, we again observe a close match between the BER curves of the K-best detector and the slicing K-best detector. In our simulations the list length K was large enough to prevent an empty set $S^1_{i,\ell} = \phi$ or $S^0_{i,\ell} = \phi$ in the numerator or denominator of (6). For a smaller K, however, it has been reported in [23] that clipping the LLR value to a certain range can mitigate the performance degradation resulting from unbounded LLR due to an insufficient list length.

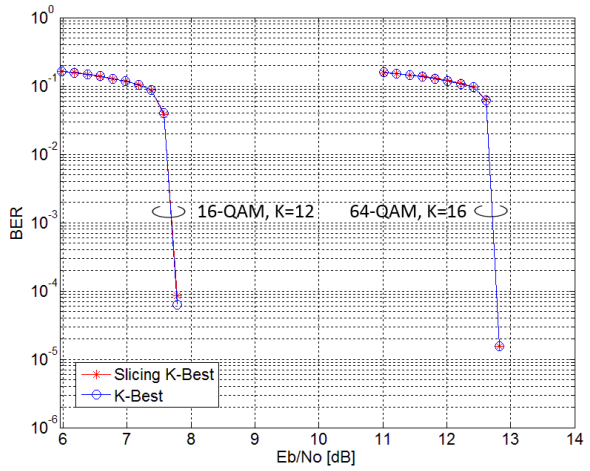
To accommodate iterative detection and decoding schemes [27]– [29], soft inputs to the detector as well as soft outputs are required. This can be realized by adding the extrinsic information from the soft-decision channel decoder to the branch metric in (4) as an updated *a priori* probability, which converts the ML criterion to the maximum a posteriori (MAP) criterion [30]– [32].

C. Complexity-Performance Tradeoffs

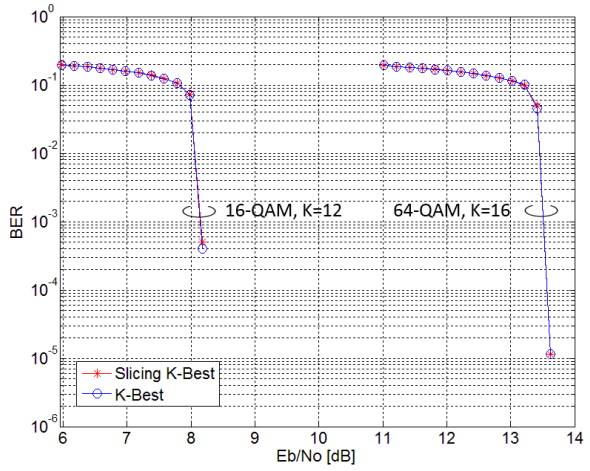
There are two factors that affect the complexity involved in the metric calculations: the computational complexity involved in each calculation itself, and the number of calculations required for the detection process, which is determined by the size of the search space. An example for reducing the computational complexity of calculation is approximating the Euclidean-norm by

$$\sqrt{x^2+y^2} \approx \frac{3}{8}\left(|x|+|y|\right) + \frac{5}{8}\max\left(|x|,|y|\right). \tag{7}$$

Fig. 7 depicts the complexity-performance tradeoffs for different MIMO detection schemes. The complexity is quantified

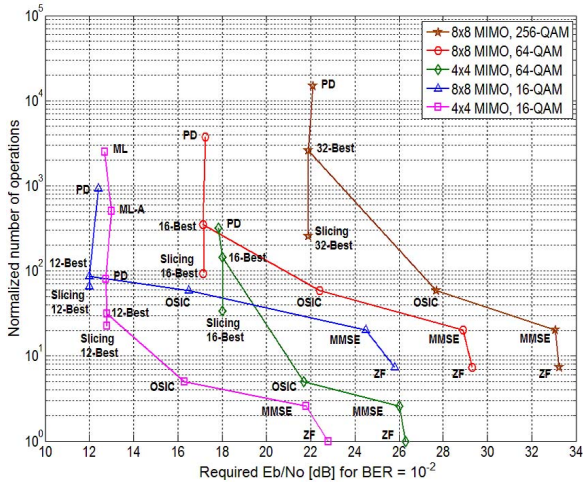


(a) 4×4 MIMO

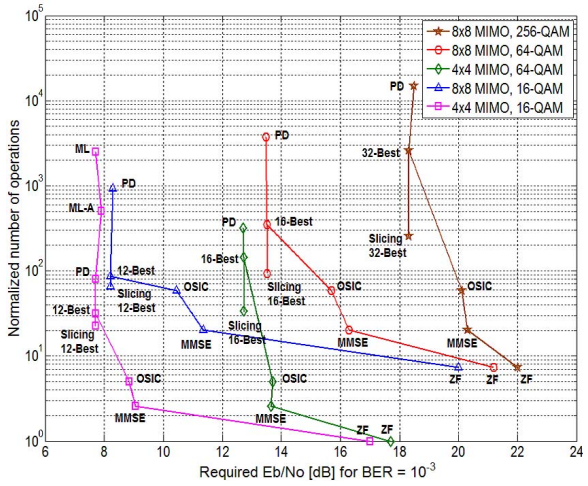


(b) 8×8 MIMO

Fig. 6. (64800, 32400)-LDPC coded BER versus E_b/N_0 for (a) 4×4 MIMO and (b) 8×8 MIMO i.i.d. Rayleigh fading channel with Gray mapped 16-QAM and 64-QAM.



(a) Uncoded



(b) Coded

Fig. 7. Tradeoffs between number of operations and required E_b/N_0 for (a) uncoded BER $= 10^{-2}$ and (b) coded BER $= 10^{-3}$ in Rayleigh fading channels.

by the relative number of operations normalized by the number of operations for ZF detection[1] in the 4×4 MIMO case, and the performance is quantified by the required E_b/N_0 for either an uncoded or coded target BER of 10^{-2} and 10^{-3}, respectively. From the figure, we observe that the slicing K-best scheme can significantly reduce the number of operations, while keeping the required E_b/N_0 almost identical to that of the K-best scheme. When compared to all of the other detectors, which all trace out a roughly inverse-proportional tradeoff curve between performance and complexity, the proposed slicing K-best detector stands out as an exception, achieving near-best performance with a substantially low complexity.

For the K-best, slicing K-best, and PD schemes, comparing the required E_b/N_0 for the uncoded target BER of 10^{-2} while increasing the MIMO channel rank for a fixed modulation order (□ vs. Δ markers, and ♢ vs. ○ markers in Fig. 7(a)), it can be observed that the required E_b/N_0 for the 8×8 MIMO case is even smaller than that for the 4×4 MIMO case. This is attributed to the higher diversity gain (exhibited as steeper slope of BER curves in Fig. 5(b) compared to that in Fig. 5(a)) provided by the larger number of receive antennas. However, for the ZF and MMSE linear detectors, the achievable diversity gain is $N_r - N + 1 = 1$ for both 4×4 MIMO and 8×8 MIMO cases. So, the required E_b/N_0 increases as the MIMO channel rank increases for a given modulation order.

The ML-A scheme in Fig. 7 uses the Euclidean-norm approximation of (7) in calculating path metrics, so as to reduce the computational complexity. However, the size of the search space is often more critical, as it also entails a requirement for a buffer size to store and fetch calculated metrics and to compare them. Thus, reducing the calculation complexity alone has limited effect on lowering the overall complexity, especially for a higher rank MIMO system with a higher modulation order.[2] From the figure, we can observe that the K-best, slicing K-best, and PD schemes outperform

[1]Since the channel $\mathbf{H}$ is a square matrix, the inverse $\mathbf{H}^{-1}$ is used instead of the pseudoinverse $\left(\mathbf{H}^H\mathbf{H}\right)^{-1}\mathbf{H}^H$ in counting the number of operations for ZF detection.

[2]It is worth mentioning that the Euclidean-norm approximation of (7) can be incorporated with K-best or slicing K-best scheme to further reduce the complexity involved in path metric calculations.

the ML-A scheme even with less complexity for 4×4 MIMO with 16-QAM case. Beyond the 4×4 MIMO with 16-QAM, an exhaustive search-based detection scheme, such as ML or ML-A, becomes impractical as the number of candidate sets gets out of a practical scope. Therefore, the simulations for ML and ML-A schemes are performed only for the 4×4 MIMO with 16-QAM case.

The layer ordering for the OSIC scheme is based on the decreasing order of post-detection SINR. The PD scheme is an extension of the OSIC scheme, while performing parallel sub-stream detection by considering all possible M candidates at the first layer. In this case, detecting the symbol with the smallest SINR at the first layer, then perform sub-stream detection based on the decreasing order of post-detection SINR is shown to be optimal [33]. As stated earlier, the layer ordering for the K-best and slicing K-best detection schemes is based on the decreasing order of SINR.

Comparing complexity while increasing the modulation order (□ Δ markers for 16-QAM, ◊ ○ markers for 64-QAM, and ☆ markers for 256-QAM), one can notice that the complexity gap between the K-best scheme and the slicing K-best scheme increases rapidly as the modulation order M increases. The complexity in the K-best scheme is affected by both M and K values, since the number of branches from each node is governed by M, and the number of nodes in each layer is governed by K. From the figure, it can be observed that the number of operations for the K-best scheme increases more than a linear factor of the increase in M, as K value also has been increased to avoid increase in the BER. On the other hand, the number of operations for the slicing K-best scheme is only affected by the increase in K, as the number of branches from each node and the number of nodes in each layer are both governed by K. The complexity of PD scheme increases roughly linear to M, as the number of parallel sub-streams from the first layer increases proportional to the alphabet size.

The size of the exhaustive search space for the 16-QAM, 8×8 MIMO case is 16^8, which is 256 times larger than that for the 64-QAM, 4×4 MIMO case, which is 64^4. However, Fig. 7 shows that the number of operations for 64-QAM, 4×4 MIMO case is higher than for the 16-QAM, 8×8 MIMO case, which indicates the complexity of the K-best scheme does not scale accordingly with the complexity of the problem to be solved, since it is is dominantly affected by the increase in M. On the other hand, the breadth of the tree search in the slicing K-best scheme is controlled by K, so the complexity does not scale by a factor of M. This becomes more crucial for higher-order modulation cases. In [16], it is claimed that as the symbol constellations quadruples from 16-QAM to 64-QAM, the K value only doubles for the similar BER performance. If the K value increases proportionally as the modulation order M increases, still the difference between M and K increases, so the complexity gap between the conventional K-best and the slicing K-best increases even for the same K/M ratio.

In each layer except for the first layer, the number of metrics to be computed and sorted to select K best candidates in the conventional K-best scheme is KM, whereas that in the slicing K-best scheme is K^2, which is reduced by a factor of K/M. In the first layer, M number of metrics need to be computed and sorted to select K best candidates from them for the conventional K-best scheme, whereas there is no sorting and selection process required for the slicing K-best scheme.

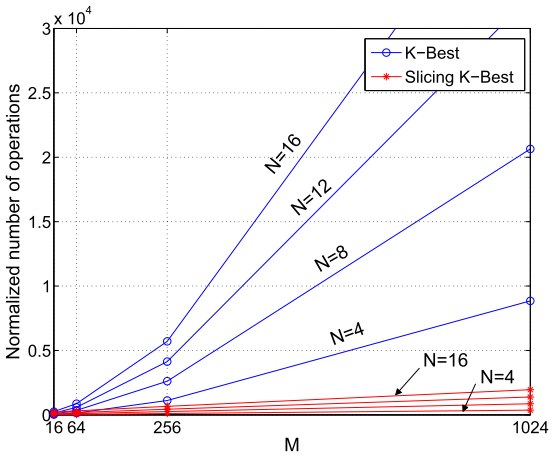


Fig. 8. Detection complexity of the K-best scheme and the slicing K-best scheme for $N \in \{4, 8, 12, 16\}$, and $M \in \{16, 64, 256, 1024\}$ with corresponding $K = 2\sqrt{M} \in \{8, 16, 32, 64\}$.

Fig. 8 compares the detection complexity of the K-best scheme and the slicing K-best scheme for different number of antennas N and alphabet size M. The value of K is fixed at $K = 2\sqrt{M}$, so as to make it double as M quadruples. In this case, the complexity of the K-best scheme rapidly increases as M increases in the order of $O(KM) = O(M^{3/2})$, whereas the complexity of slicing K-best scheme increases linearly with M as in $O(K^2) = O(M)$. The complexity of K-best and slicing K-best schemes both increase linearly as N increases, but the increasing rate in the K-best scheme is faster than that in the slicing K-best scheme by a factor of $O(M/K) = O(\sqrt{M})$. Hence, the proposed slicing K-best detection scheme can substantially reduce the complexity of conventional K-best detection scheme, especially for MIMO systems with large M and N.

VI. Conclusions

In this work, we proposed a prune-as-you-go approach for the K-best MIMO detector that extends only the K best candidates from each node using a simple slicing operation. This causes the breadth of the tree search to be governed by K, not by the alphabet size M, and it reduces the number of metric calculations by a factor of K/M. The subsequent sorting process also benefits from having fewer entries to be sorted. The slicing simplification is justified by comparing the list error probabilities of the slicing list detector and the minimum-distance list detector. Simulation results show that the proposed slicing K-best detection scheme provides comparable performance to that of a conventional K-best detection with substantially reduced complexity. The complexity reduction is especially significant for a MIMO system with a large number of antennas N and with a large alphabet size M. Comparing the performance-complexity tradeoffs for different MIMO detection schemes, the proposed slicing K-best detector is shown to be a promis-

ing choice for practical implementation of next-generation MIMO systems.

REFERENCES

[1] (2013). *IEEE Standard 802.11ac-2103*. [Online]. Available: https://standards.ieee.org/findstds/standard/802.11ac-2013.html

[2] (Mar. 2015). *Tentative 3GPP Timeline for 5G*. [Online]. Available: http://www.3gpp.org/news-events/3gpp-news/1674-timeline_5g

[3] S. Yang and L. Hanzo, "Fifty years of MIMO detection: The road to large-scale MIMOs," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 1941–1988, 4th Quart., 2015.

[4] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs Tech. J.*, vol. 1, no. 2, pp. 41–59, 1996.

[5] P. W. Wolniansky *et al.*, "V-BLAST: An architecture for realizing very high data rates over the rich-scattering wireless channel," in *Proc. URSI Int. Symp. Signals, Syst., Electron.*, Oct. 1998, pp. 295–300.

[6] Y. Li and Z.-Q. Luo, "Parallel detection for V-BLAST system," in *Proc. IEEE Int. Conf. Commun.*, vol. 1. Apr. 2002, pp. 340–344.

[7] A. K. Lenstra, H. W. Lenstra, and L. Lovász, "Factoring polynomials with rational coefficients," *Math. Annalen*, vol. 261, no. 4, pp. 515–534, 1982.

[8] M. Seysen, "Simultaneous reduction of a lattice basis and its reciprocal basis," *Combinatorica*, vol. 13, no. 3, pp. 363–376, 1993.

[9] N. D. Sidiropoulos and Z.-Q. Luo, "A semidefinite relaxation approach to MIMO detection for high-order QAM constellations," *IEEE Signal Process. Lett.*, vol. 13, no. 9, pp. 525–528, Sep. 2006.

[10] U. Fincke and M. Pohst, "Improved methods for calculating vectors of short length in a lattice, including a complexity analysis," *Math. Comput.*, vol. 44, no. 170, pp. 463–471, Apr. 1985.

[11] E. Viterbo and J. Boutros, "A universal lattice code decoder for fading channels," *IEEE Trans. Inf. Theory*, vol. 45, no. 5, pp. 1639–1642, Jul. 1999.

[12] C. P. Schnorr and M. Euchner, "Lattice basis reduction: Improved practical algorithms and solving subset sum problems," *Math. Program.*, vol. 66, no. 1, pp. 181–199, Aug. 1994.

[13] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattices," *IEEE Trans. Inf. Theory*, vol. 48, no. 8, pp. 2201–2214, Aug. 2002.

[14] Z. Guo and P. Nilsson, "Algorithm and implementation of the K-best sphere decoding for MIMO detection," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 3, pp. 491–503, Mar. 2006.

[15] M. Wenk, M. Zellweger, A. Burg, N. Felber, and W. Fichtner, "K-best MIMO detection VLSI architectures achieving up to 424 Mbps," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2006, p. 1154.

[16] M. Shabany and P. G. Gulak, "A 675 Mbps, 4 4 64-QAM K-best MIMO detector in 0.13 CMOS," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 20, no. 1, pp. 135–147, Jan. 2012.

[17] K. J. Kim, J. Yue, R. A. Iltis, and J. D. Gibson, "A QRD-M/Kalman filter-based detection and channel estimation algorithm for MIMO-OFDM systems," *IEEE Trans. Wireless Commun.*, vol. 4, no. 2, pp. 710–721, Mar. 2005.

[18] Y. Dai, S. Sun, and Z. Lei, "A comparative study of QRD-M detection and sphere decoding for MIMO-OFDM systems," in *Proc. IEEE 16th Int. Symp. Personal, Indoor Mobile Radio Commun. (PIMRC)*, vol. 1. Sep. 2005, pp. 186–190.

[19] W. H. Chin, "QRD based tree search data detection for MIMO communication systems," in *Proc. IEEE 61st Veh. Technol. Conf. (Spring)*, vol. 3. May 2005, pp. 1624–1627.

[20] M. Li *et al.*, "Selective spanning with fast enumeration: A near maximum-likelihood MIMO detector designed for parallel programmable baseband architectures," in *Proc. IEEE Int. Conf. Commun.*, May 2008, pp. 737–741.

[21] R. Fasthuber, M. Li, D. Novo, P. Raghavan, L. Van Der Perre, and F. Catthoor, "Novel energy-efficient scalable soft-output SSFE MIMO detector architectures," in *Proc. Int. Symp. Syst., Archit., Model., Simulation (SAMOS)*, Jul. 2009, pp. 165–171.

[22] Z. H. Cai, P. H. Tan, J. Z. Hao, C. M. Pang, S. M. Sun, and P. S. Chin, "Low complexity near-ML detection for MIMO-OFDM system," in *Proc. IEEE 72nd Veh. Technol. Conf. Fall (VTC-Fall)*, Sep. 2010, pp. 1–5.

[23] C. Studer, A. Burg, and H. Bolcskei, "Soft-output sphere decoding: Algorithms and VLSI implementation," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 2, pp. 290–300, Feb. 2008.

[24] I. E. Bocharova, R. Johannesson, B. D. Kudryashov, and M. Loncar, "An improved bound on the list error probability and list distance properties," *IEEE Trans. Inf. Theory*, vol. 54, no. 1, pp. 13–32, Jan. 2008.

[25] J. R. Barry, E. A. Lee, and D. G. Messerschmitt, *Digital Communication*. Springer, 2004.

[26] A. J. Viterbi, "An intuitive justification and a simplified implementation of the MAP decoder for convolutional codes," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 2, pp. 260–264, Feb. 1998.

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

[28] X. Li, H. Huang, G. J. Foschini, and R. A. Valenzuela, "Effects of iterative detection and decoding on the performance of BLAST," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, vol. 2. Nov./Dec. 2000, pp. 1061–1066.

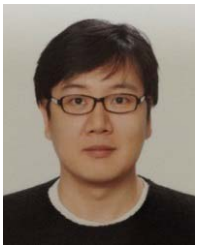
[29] H. Vikalo, B. Hassibi, and T. Kailath, "Iterative decoding for MIMO channels via modified sphere decoding," *IEEE Trans. Wireless Commun.*, vol. 3, no. 6, pp. 2299–2311, Nov. 2004.

[30] E. Zimmermann, G. Fettweis, D. L. Milliner, and J. R. Barry, "A parallel smart candidate adding algorithm for soft-output MIMO detection," in *Proc. 7th Int. ITG Conf. Sour. Channel Coding (SCC)*, Jan. 2008, pp. 1–6.

[31] J. Lee, B. Shim, and I. Kang, "Soft-input soft-output list sphere detection with a probabilistic radius tightening," *IEEE Trans. Wireless Commun.*, vol. 11, no. 8, pp. 2848–2857, Aug. 2012.

[32] G. Papa, D. Ciuonzo, G. Romano, and P. S. Rossi, "A dominance-based soft-input soft-output MIMO detector with near-optimal performance," *IEEE Trans. Commun.*, vol. 62, no. 12, pp. 4320–4335, Dec. 2014.

[33] D. L. Milliner, E. Zimmermann, J. R. Barry, and G. Fettweis, "A fixed-complexity smart candidate adding algorithm for soft-output MIMO detection," *IEEE J. Sel. Topics Signal Process.*, vol. 3, no. 6, pp. 1016–1025, Dec. 2009.



Sangwook Suh (S'06–M'15) received the B.S. degree from Seoul National University, Seoul, South Korea, in 1998, the M.S. degree from New York University, New York, NY, USA, in 2005, and the Ph.D. degree from the Georgia Institute of Technology, Atlanta, GA, USA, in 2011, all in electrical engineering. Since 2015, he has been a Post-Doctoral Research Scholar with the Georgia Institute of Technology. In lieu of military service, he was with Sein Electronics and Corecess, South Korea, from 1998 to 2002. In 2002, he was a Software Engineering Intern with Microsoft Korea. From 2005 to 2006 and from 2012 to 2015, he was with Samsung Electronics, South Korea. His research interests span a wide range of communication theory, such as MIMO detection algorithm, space-time coding, low PAPR OFDM, interference alignment, non-orthogonal multiple access, and capacity region for broadcast channels.

John R. Barry (SM'04) received the B.S. (*summa cum laude*) degree from the University at Buffalo, The State University of New York, Buffalo, NY, USA, in 1986, and the M.S. and Ph.D. degrees from the University of California at Berkeley, Berkeley, CA, USA, in 1987 and 1992, respectively, all in electrical engineering. His Ph.D. research explored the feasibility of broadband wireless communications using diffuse infrared radiation. He is currently a Professor with the School of Electrical and Computer Engineering, Georgia Institute of Technology. He has held engineering positions in communications and radar systems with Bell Communications Research, Murray Hill, NJ, USA, the IBM Thomas J. Watson Research Center, Yorktown Heights, NY, USA, Hughes Aircraft Company, Glendale, CA, USA, and General Dynamics Company, Falls Church, VA, USA, since 1985. He has co-authored a book entitled *Digital Communication-Third Edition* (Kluwer, 2004), co-edited a book entitled *Advanced Optical Wireless Communication Systems* (Cambridge University Press, 2012), and authored a book entitled *Wireless Infrared Communications* (Kluwer, 1994). He currently serves as a Guest Editor of the special issue of the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS. He was a recipient of the David J. Griep Memorial Prize in 1992, the Eliahu Jury Award from the University of California at Berkeley, the Research Initiation Award from National Science Foundation, and the IBM Faculty Development Award in 1993.