A Lattice-Reduction-Aided Max-Log List Demapper for Coded MIMO Receivers

Tung-Jung Hsieh and Wern-Ho Sheen, *Member, IEEE*

***Abstract*—The max-log list demapper has been widely employed in the implementations of a coded multiple-input–multiple-output (MIMO) receiver, where only a candidate list of signal vectors is examined in the likelihood-ratio calculation to reduce complexity. Traditionally, the candidate list is generated in the original-lattice domain, which, unfortunately, results in severe degradation in the performance of the demapper if the channel is in ill-condition. In this paper, two new lattice-reduction-aided max-log list demappers are proposed, i.e., one for an iterative receiver and the other for a noniterative receiver. With similar complexity, the proposed demappers provide significant gains over existing demappers, particularly for the cases with a small list size and/or under a spatially correlated channel, due to the new algorithms for the generation of the candidate list. In addition, for the iterative receiver, the prior information coming out of the decoder is exploited to lower the complexity of the demapper.**

***Index Terms*—Coded multiple-input–multiple-output (MIMO) receiver, lattice reduction (LR), max-log list demapper.**

I. Introduction

MULTIPLE-INPUT–MULTIPLE-OUTPUT (MIMO) technology has been widely employed to improve the performance of wireless communications in a rich-scattered fading environment [1], [2]; by using multiple antennas at both the transmitter and the receiver, it is able to provide diversity gain, array gain (power gain), and/or degree-of-freedom gain over the single-input–single-output counterpart. MIMO technology along with channel coding (coded MIMO) has been widely adopted in today's wireless broadband standards, including IEEE 802.16e [3] and 3GPP-LTE [4].

In a coded-MIMO system, the optimal maximum *a posteriori* (MAP) receiver is often too complex to be implemented in practice [5]–[7]. Instead, a suboptimal receiver that consists of a separate demapper and decoder is regarded as a more practical design and has been widely implemented in real systems. In such a receiver, the *a posteriori* probability (APP) demapper that calculates the true log-likelihood ratio (LLR) of coded bits has the best performance, but with complexity exponentially growing with the number of MIMO layers and/or the size of signal constellation.

Manuscript received August 8, 2012; accepted July 21, 2013. Date of publication August 15, 2013; date of current version February 12, 2014. The review of this paper was coordinated by Prof. Y. L. Guan.

T.-J. Hsieh is with the Department of Electrical and Computer Engineering, National Chiao Tung University, Hsinchu 300, Taiwan (e-mail: fishdon.cm94g@nctu.edu.tw).

W.-H. Sheen is with the Department of Communications Engineering, National Chung Cheng University, Chiayi 621, Taiwan (e-mail: whsheen@ccu.edu.tw).

Digital Object Identifier 10.1109/TVT.2013.2278029

One common way to reduce the complexity of the APP demapper is to use the max-log demapper in which a simplified LLR, called max-LLR, is calculated by just including the signal vectors that have the MAP probability in each of the signal set associated with the code bit 0 and 1, respectively [8], [15]. Another popular way to reduce the complexity is to use the list demapper, where only a list of candidate signal vectors (rather than all) is examined in the LLR calculation [8]–[20]. The method of using a candidate list can be applied to both the APP and max-log demappers [8], [10]. In this paper, we are concerned with the design of the max-log list demapper.

In the literature, max-log list demapping can be done in the original-lattice [8]–[18], [35]–[37] or the lattice-reduced domain [19]–[26]. In the original-lattice domain, the demappers can be roughly classified into the depth-first [8]–[10], [35] and the breadth-first [11]–[18], [36], [37] type of methods; since the symbols of different MIMO layers are independent, the demapper can be implemented in a complexity-friendly way, particularly for the breadth-first type of methods [12]–[14], [36], [37]. Nevertheless, in the methods of the original-lattice domain, an increase in the list size is often necessary for a given performance if the channel is ill-conditioned [29], [30], and that significantly increases the complexity of demappers. To overcome this problem, the lattice-reduction-aided (LR-aided) list demappers have been proposed [19]–[26], with [19]–[25] focusing on the noniterative receiver and [26] on the iterative receiver. By exchanging the extrinsic information back and forth between the demapper and the decoder, the iterative receiver is regarded as a practical way to nearly achieve the performance of the optimal MAP receiver [5]–[7].

In [19]–[21], the candidate list of the max-log list demapper is generated after a linear filtering of the received signal in the lattice-reduced domain, whereas in [22], the successive cancelation of multilayer interference was employed instead so as to improve performance. The complexity in [22] was further lowered in [23]–[25]. In [26], a novel LR-aided successive-interference-cancelation-based list demapper was proposed; by exploiting the regularity property of the original-lattice domain constellation, the utilization of prior information was realized in the original-lattice domain.

In this paper, two new LR-aided max-log list demappers are proposed, i.e., one for an iterative receiver and the other for a noniterative receiver. Due to the new methods of producing the candidate list, which is produced in the breadth-first manner after a successive cancelation of multilayer interference in the lattice-reduced domain, the proposed demappers are superior to the existing methods under similar complexity; the gain is particularly prominent for the cases with small list size

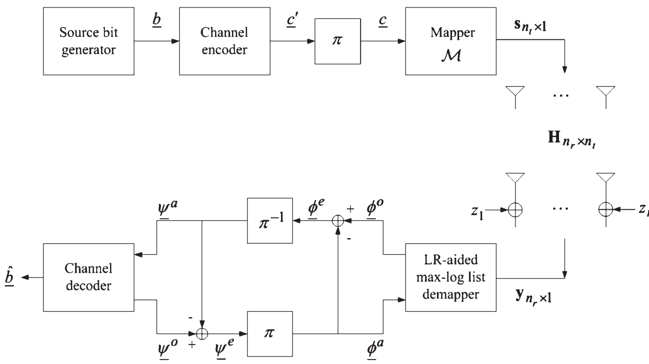

Fig. 1. Considered spatially multiplexed MIMO system.

and/or under a spatially correlated channel. For the iterative receiver, the prior information coming out of the decoder is also exploited to lower the complexity of the demapper.

This paper is organized as follows. Section II describes the system and channel models. Section III reviews the concept of LR and the list-based max-log demapper. In Section IV, the new LR-aided demapper is presented, and simulation results and conclusions are given in Section V and Section VI, respectively.

II. System and Channel Models

Consider the bit-interleaved coded-MIMO system with n_l transmit and $n_r \geq n_t$ receive antennas in Fig. 1. The source bit sequence $\underline{b}$ is fed into the channel encoder to generate the coded bit sequence $\underline{c}'$. Then, after interleaver π, coded bit sequence $\underline{c}$ is grouped per $m \doteq \log_2 M$ bits, which is mapped to a symbol in constellation $\mathcal{S}$ of size M. A total of n_t symbols are collected to form signal vector $\mathbf{s} = [s_1, \ldots, s_{n_t}]^T \in \boldsymbol{\mathcal{S}} \doteq \mathcal{S}^{n_t}$, where s_n is the nth layer symbol to be transmitted from antenna n, $\mathcal{S}^{n_t}$ is the n_t-fold Cartesian product of $\mathcal{S}$, and $[\bullet]^T$ denotes the transpose of a matrix or vector. The symbols $\{s_n\}$ are assumed to be independent and identically distributed random variables with zero mean and variance σ_s^2. That is, $\mathrm{E}[\mathbf{s}\mathbf{s}^H] = \sigma_s^2 \cdot \mathbf{I}_{n_t}$, where $\mathrm{E}[\bullet]$ denotes taking expectation, $\mathbf{I}_{n_t}$ is the $n_t \times n_t$ identity matrix, and $[\bullet]^H$ denotes the Hermitian transpose of a matrix or vector.

For a flat-faded channel, the received signal vector is given by

$$\mathbf{y} \doteq [y_1, \ldots, y_{n_r}]^T = \mathbf{H}\mathbf{s} + \mathbf{z} \tag{1}$$

where $\mathbf{H}$ is an $n_r \times n_t$ channel matrix, and $\mathbf{z} = [z_1, \ldots, z_{n_t}]^T$ is a complex Gaussian noise vector with zero mean and covariance matrix $\mathrm{E}\{\mathbf{z}\mathbf{z}^H\} = \sigma_z^2 \cdot \mathbf{I}_{n_r}$. The widely used correlated channel model $\mathbf{H} = \mathbf{J}_r^{1/2}\mathbf{F}\mathbf{J}_t^{1/2}$ will be adopted in this paper, where $\mathbf{F}$ consists of zero-mean uncorrelated complex Gaussian coefficients with unit variance, and $\mathbf{J}_t$ and $\mathbf{J}_r$ are the spatial correlation matrices due to transmit and receive antennas, respectively [27]–[29].

The demapper calculates the max-LLR $\underline{\phi}^o$, based on the sequence of received signal vector $\underline{\mathbf{y}}$ and prior information $\underline{\phi}^a$ if there is any. The extrinsic information $\underline{\phi}^e$ is obtained by subtracting prior information $\underline{\phi}^a$ from $\underline{\phi}^o$, and after deinterleaver π^{-1}, $\underline{\phi}^e$ becomes prior information $\underline{\psi}^a$ to the decoder. For the iterative receiver, the outer decoder outputs the estimated source bit sequence $\hat{\underline{b}}$ if the maximum number of iterations is reached; otherwise, it outputs $\underline{\psi}^o$, which, after subtracting prior information $\underline{\psi}^a$, gives extrinsic information $\underline{\psi}^e$, and that becomes prior information $\underline{\phi}^a$ to the demapper after interleaver π. A more detailed description on the iterative receiver can be found in [5]–[10].

III. Preliminaries

Here, the concept of LR will be reviewed first, followed by the calculation of max-LLR in the lattice-reduced domain.

A. Lattice Reduction

Let $\mathbf{h}_n \in \mathbb{C}^{n_r}$ denote the nth column of channel matrix $\mathbf{H}$, where $\mathbb{C}$ is the set of complex numbers. As in [31]–[34], the lattice spanned by $\mathbf{H}$ is defined as the set of points

$$\mathcal{L} \doteq \left\{ \sum_{n=1}^{n_t} \mathbf{h}_n z_n, z_n \in \mathcal{Z} \right\} = \{\mathbf{H}\mathbf{z}, \mathbf{z} \in \mathcal{Z}^{n_t}\} \tag{2}$$

where $\mathcal{Z}$ is the set of Gaussian integers, and $\mathbf{H}$ is called a basis of lattice $\mathcal{L}$. Note that there are infinite number of bases of $\mathcal{L}$, and two bases $\tilde{\mathbf{H}}$ and $\mathbf{H}$ are said to span the same lattice $\mathcal{L}$ if and only if $\tilde{\mathbf{H}} = \mathbf{H}\mathbf{T}$ for a unimodular matrix $\mathbf{T}$ [31]–[34]. A Gaussian integer matrix $\mathbf{T}$ is said to be unimodular if $|\det(\mathbf{T})| = 1$, where $\det(\mathbf{T})$ denotes the determinant of $\mathbf{T}$.

Lattice reduction is a procedure to find reduced basis $\tilde{\mathbf{H}}$ from $\mathbf{H}$ with a set of more orthogonal (shorter) column vectors than those of $\mathbf{H}$. LR has been widely applied to improve the performance of the MIMO systems [20]–[22], [28], [31], [32]; with a set of more orthogonal column vectors, the effect of noise enhancement can be reduced in detection/decoding in the lattice-reduced domain. The Lenstra–Lenstra–Lovász (LLL) algorithm (real-valued LLL [33] or complex-valued LLL [34]) has been known as one of the effective LR algorithms to find reduced basis $\tilde{\mathbf{H}}$ in a polynomial time [33].

In terms of reduced basis $\tilde{\mathbf{H}} = \mathbf{H}\mathbf{T}$, the signal model in (1) can be rewritten as

$$\mathbf{y} = \tilde{\mathbf{H}}\tilde{\mathbf{s}} + \mathbf{z} \tag{3}$$

where $\tilde{\mathbf{s}} \doteq [\tilde{s}, \ldots, \tilde{s}_{n_t}]^T = \mathbf{T}^{-1}\mathbf{s}$.[1] Note that in the lattice-reduced domain, the signal constellation is not as regular as in the original-lattice domain, and transformation $\mathbf{T}^{-1}$ introduces dependence between layers in $\tilde{\mathbf{s}}$. Constellation irregularity and dependence between layers in the lattice-reduced domain increase the complexity of the demapper, as will be detailed in Section IV.

B. Calculation of Max-LLR in Lattice-Reduced Domain

Using reduced basis $\tilde{\mathbf{H}}$, the max-LLR for the ith bit of the nth layer symbol is calculated by

$$\begin{aligned}\tilde{\phi}_{n,i}^o = &\max_{\tilde{\mathbf{s}} \in \tilde{\boldsymbol{\mathcal{S}}}_{n,i}^1} \left(\frac{-1}{\sigma_z^2} \left(\|\mathbf{y} - \tilde{\mathbf{H}}\tilde{\mathbf{s}}\|_F \right)^2 + \log P_r(\tilde{\mathbf{s}}) \right) \\ &- \max_{\tilde{\mathbf{s}} \in \tilde{\boldsymbol{\mathcal{S}}}_{n,i}^0} \left(\frac{-1}{\sigma_z^2} \left(\|\mathbf{y} - \tilde{\mathbf{H}}\tilde{\mathbf{s}}\|_F \right)^2 + \log P_r(\tilde{\mathbf{s}}) \right)\end{aligned} \tag{4}$$

[1] Throughout this paper, the signal constellation has been shifted and scaled such that the constellation points are contiguous integers in both original-lattice and lattice-reduced domains [22], [31].

where $\tilde{\boldsymbol{\mathcal{S}}}_{n,i}^{b}$ is the set of $\tilde{\mathbf{s}}$ with the ith bit of the nth layer symbol s_n equal to b, $b \in \{1, 0\}$, $P_r(\tilde{\mathbf{s}})$ is the *a priori* probability of $\tilde{\mathbf{s}}$, and $\|\mathbf{x}\|_F$ denotes the Frobenius norm of $\mathbf{x}$. Using QR factorization on $\tilde{\mathbf{H}}$, we have

$$\tilde{\mathbf{H}} = \tilde{\mathbf{Q}}\tilde{\mathbf{R}} \tag{5}$$

where $\tilde{\mathbf{Q}}$ is an $n_r \times n_t$ matrix with orthonormal columns, and $\tilde{\mathbf{R}}$ is an $n_t \times n_t$ upper triangular matrix. Premultiplying $\mathbf{y}$ by $\tilde{\mathbf{Q}}^H$, one has

$$\tilde{\mathbf{v}} \doteq \tilde{\mathbf{Q}}^H\mathbf{y} = \tilde{\mathbf{R}}\tilde{\mathbf{s}} + \tilde{\mathbf{z}} \tag{6}$$

where $\tilde{\mathbf{z}} \doteq \tilde{\mathbf{Q}}^H\mathbf{z}$, and $\phi_{n,i}^{o}$ can be rewritten as

$$\tilde{\phi}_{n,i}^{o} = \max_{\tilde{\mathbf{s}}\in\tilde{\boldsymbol{\mathcal{S}}}_{n,i}^{1}} \left(\frac{-1}{\sigma_z^2}\left(\|\tilde{\mathbf{v}} - \tilde{\mathbf{R}}\tilde{\mathbf{s}}\|_F\right)^2 + \log P_r(\tilde{\mathbf{s}})\right) - \max_{\tilde{\mathbf{s}}\in\tilde{\boldsymbol{\mathcal{S}}}_{n,i}^{0}} \left(\frac{-1}{\sigma_z^2}\left(\|\tilde{\mathbf{v}} - \tilde{\mathbf{R}}\tilde{\mathbf{s}}\|_F\right)^2 + \log P_r(\tilde{\mathbf{s}})\right). \tag{7}$$

As is clear in (7), since all signal vectors in $\tilde{\boldsymbol{\mathcal{S}}} \doteq \tilde{\boldsymbol{\mathcal{S}}}_{n,i}^{1} \cup \tilde{\boldsymbol{\mathcal{S}}}_{n,i}^{0}$ have to be examined, the calculation of max-LLR may still be too complex in practical applications. To reduce the complexity, one practice is just to examine a candidate list $\boldsymbol{\Psi}$ with $|\boldsymbol{\Psi}| < |\tilde{\boldsymbol{\mathcal{S}}}|$ in the calculation [22]–[25], where $|\mathbf{X}|$ denotes the cardinality of set $\mathbf{X}$. In this way, (7) can be simplified with $\tilde{\boldsymbol{\mathcal{S}}}_{n,i}^{1}$ and $\tilde{\boldsymbol{\mathcal{S}}}_{n,i}^{0}$ replaced by $\boldsymbol{\Psi}_{n,i}^{1}$ and $\boldsymbol{\Psi}_{n,i}^{0}$, where $\boldsymbol{\Psi}_{n,i}^{b} \subset \boldsymbol{\Psi}$ is the set of $\tilde{\mathbf{s}}$ in the list with the ith bit of the nth layer symbol s_n equal to b. Since only a part of signal vectors (the candidate list $\boldsymbol{\Psi}$) are involved in the calculation of max-LLR in the list demapper, those $\tilde{\mathbf{s}}$'s with small $D(\tilde{\mathbf{s}}) \doteq (\|\tilde{\mathbf{v}} - \tilde{\mathbf{R}}\tilde{\mathbf{s}}\|_F)^2 - \sigma_z^2 \log P_r(\tilde{\mathbf{s}})$ should be selected and put into $\boldsymbol{\Psi}$ for better performance.

It is worth noting that if all signal vectors, that is, $\boldsymbol{\mathcal{S}}$ (or $\tilde{\boldsymbol{\mathcal{S}}}$), are examined in the calculation of max-LLR, then there is no difference whether the calculation is done in the original-lattice or lattice-reduced domain. If only a subset of signal vectors (a candidate list) is to be examined, on the other hand, it would be beneficial to do the calculation in the lattice-reduced domain because of less noise enhancement.

IV. Proposed Lattice-Reduction-Aided Max-Log List Demappers

Two new LR-aided max-log list demappers are proposed here, i.e., one for an iterative receiver and the other for a noniterative receiver. After obtaining a reduced basis, a candidate list $\boldsymbol{\Psi}$ is generated in the lattice-reduced domain, followed by the calculation of max-LLR using the candidate list. The main difference between the demappers for the iterative receiver and the noniterative receiver is the generation of the candidate list $\boldsymbol{\Psi}$.

A. Generation of $\boldsymbol{\Psi}$ With Prior Information (Iterative Receiver)

The generation of the candidate list $\boldsymbol{\Psi}$ directly from $\tilde{\boldsymbol{\mathcal{S}}}$ is usually too complex to be practical for the case of large M^{n_t} because of constellation irregularity and dependence between layers in the lattice-reduced domain. Here, for the case with prior information (after the first demapping in the iterative receiver), a two-step algorithm is proposed to reduce the complexity. Initially, a set $\boldsymbol{\mathcal{T}}$ that contains the $K_{\mathcal{T}}$ most probable signal vectors in $\tilde{\boldsymbol{\mathcal{S}}}$ is constructed based on the prior information coming out of the decoder. Then, the candidate list $\boldsymbol{\Psi}$ is generated from $\boldsymbol{\mathcal{T}}$ in a breadth-first search (BFS) manner. Since $\tilde{\mathbf{s}} = \mathbf{T}^{-1}\mathbf{s}$, $\boldsymbol{\mathcal{T}}$ can be generated by searching over $\mathbf{s} \in \boldsymbol{\mathcal{S}}$.

Let $c_{n,i}$ denote the ith bit of the nth layer symbol s_n and $P_r(c_{n,i})$ the *a priori* probability of $c_{n,i}$ coming out of the decoder.[2] Under the assumption of ideal interleaving, the probability of signal vector $\mathbf{s}$ can be expressed as

$$P_r(\mathbf{s}) = \prod_{n=1}^{n_t} P_r(s_n) = \prod_{n=1}^{n_t}\prod_{i=1}^{m} P_r(c_{n,i}) \tag{8}$$

where

$$P_r(c_{n,i}) = \frac{\exp\left(c_{n,i}\psi_{n,i}^{o}\right)}{1 + \exp\left(\psi_{n,i}^{o}\right)} \tag{9}$$

and $\psi_{n,i}^{o}$ is the max-LLR of $c_{n,i}$ coming out of the decoder. By disregarding the constant term $\Pi_{n=1}^{n_t}\Pi_{i=1}^{m}(1 + \exp(\psi_{n,i}^{o}))^{-1}$ in (8), the metric

$$D_{\mathcal{T}}(\mathbf{s}) = \sum_{n=1}^{n_t}\sum_{i=1}^{m} c_{n,i}\psi_{n,i}^{o} \tag{10}$$

is used for the generation of $\boldsymbol{\mathcal{T}}$, as is detailed below.

First, a binary tree of depth $n_t m$ is constructed with the left and right nodes at level $l = (n-1)m + i$ indicating $c_{n,i} = 0$ and $c_{n,i} = 1$, respectively, where $1 \le n \le n_t$,$1 \le i \le m$. The root is indexed as level 0, and the other levels are indexed starting from $n = 1, i = 1, \ldots, m, n = 2, i = 1, \ldots, m$, and so forth. Furthermore, a path from level 0 to level l is denoted by the l-tuple $\mathbf{s}^{(l)} \doteq [c_{11}, c_{12}, \ldots, c_{n,i}]$ and is associated with the path metric

$$D_{\mathcal{T}}\left(\mathbf{s}^{(l)}\right) = \sum_{j=1}^{n}\sum_{k=1}^{i} c_{j,k} \cdot \psi_{j,k}^{o} \tag{11}$$

where $\mathbf{s}^{(n_t m)}$ is an $n_t m$-tuple that corresponds to signal vector $\mathbf{s}$ (or $\tilde{\mathbf{s}}$). Second, starting from the root node, the tree is searched in a breadth-first manner, where the most $K_{\mathcal{T}}$ probable paths at each level are retained according to (11). If the number of nodes is less than $K_{\mathcal{T}}$, all paths are retained. Finally, at level $n_t m$, the most probable $K_{\mathcal{T}}$ tuples of length $n_t m$ bit are used to construct $\boldsymbol{\mathcal{T}}$ after being mapped to the lattice-reduced domain.

To generate the candidate list $\boldsymbol{\Psi} \subset \boldsymbol{\mathcal{T}}$, we first define the following metric $D_{\Psi}(\tilde{\mathbf{s}}^{(k)})$ for the partial signal vector $\tilde{\mathbf{s}}^{(k)} \doteq [\tilde{s}_{n_t-k+1}, \tilde{s}_{n_t-k+2}, \ldots, \tilde{s}_{n_t}]^T$, $k = 1, \ldots, n_t$ [see (7)]:

$$D_{\Psi}\left(\tilde{\mathbf{s}}^{(k)}\right) \doteq \left(\left\|\tilde{\mathbf{v}}^{(k)} - \tilde{\mathbf{R}}^{(k)}\tilde{\mathbf{s}}^{(k)}\right\|_F\right)^2 - \sigma_z^2 \log P_r\left(\tilde{\mathbf{s}}^{(k)}\right) = \sum_{n=n_t-k+1}^{n_t}\left|\tilde{v}_n - \tilde{r}_{n,n}\tilde{s}_n - \sum_{j=n+1}^{n_t}\tilde{r}_{n,j}\tilde{s}_j\right|^2 - \sigma_z^2 \log P_r\left(\tilde{\mathbf{s}}^{(k)}\right) \tag{12}$$

[2]For notation simplicity, $P_r(x)$ is used to denote the probability that random variable X has assumed the value x.

where $\tilde{\mathbf{v}}^{(k)} = [\tilde{\mathbf{v}}_{n_t-k+1}, \ldots, \tilde{\mathbf{v}}_{n_t}]^T$, $\tilde{r}_{n,j} = [\tilde{\mathbf{R}}]_{n,j}$, and $\tilde{\mathbf{R}}^{(k)}$ is a $k \times k$ matrix consisting of $\tilde{r}_{n,j}$, $n = n_t - k + 1, \ldots, n_t$, $j = n_t - k + 1, \ldots, n_t$, that is, $\tilde{\mathbf{R}}^{(k)}$ is the lower right square submatrix of $\tilde{\mathbf{R}}$. Using the chain rule, i.e., $P_r(\tilde{\mathbf{s}}^{(k)}) = \Pi_{n=n_t-k+1}^{n_t} P_r(\tilde{s}_n | \tilde{s}_{n_t}, \ldots, \tilde{s}_{n+1})$, (12) becomes

$$
\begin{aligned}
D_\Psi\left(\tilde{\mathbf{s}}^{(k)}\right) &= \sum_{n=n_t-k+1}^{n_t} \left(\left| \tilde{v}_n - \tilde{r}_{n,n}\tilde{s}_n - \sum_{j=n+1}^{n_t} \tilde{r}_{n,j}\tilde{s}_j \right|^2 \right. \\
&\qquad \left. - \sigma_z^2 \log P_r\left(\tilde{s}_n \left| \tilde{s}_{n_t}, \ldots, \tilde{s}_{n+1}\right.\right) \right) \\
&= \left| \tilde{v}_{n_t-k+1} - \tilde{r}_{n_t-k+1,n_t-k+1}\tilde{s}_{n_t-k+1} \right. \\
&\qquad \left. - \sum_{j=n_t-k+2}^{n_t} \tilde{r}_{n_t-k+1,j}\tilde{s}_j \right|^2 \\
&\quad - \sigma_z^2 \log P_r\left(\tilde{s}_{n_t-k+1} \left| \tilde{s}_{n_t}, \ldots, \tilde{s}_{n_t-k+2}\right.\right) \\
&\quad + D_\Psi\left(\tilde{\mathbf{s}}^{(k-1)}\right)
\end{aligned} \tag{13}
$$

where $P_r(\tilde{s}_{n_t} | \tilde{s}_{n_t}, \ldots, \tilde{s}_{n_t+1}) \doteq P_r(\tilde{s}_{n_t})$, and $D_\Psi(\tilde{\mathbf{s}}^{(0)}) \doteq 0$. In addition, for a specific $\boldsymbol{\alpha}^{(k)} = [\alpha_{n_t-k+1}, \alpha_{n_t-k+2}, \ldots, \alpha_{n_t}]^T$, the term $P_r(\tilde{s}_{n_t-k+1} | \tilde{s}_{n_t}, \ldots, \tilde{s}_{n_t-k+2})$ is calculated as

$$
\begin{aligned}
P_r&\left(\tilde{s}_{n_t-k+1} = \alpha_{n_t-k+1} \left| \tilde{s}_{n_t} = \alpha_{n_t}, \ldots, \tilde{s}_{n_t-k+2} = \alpha_{n_t-k+2}\right.\right) \\
&= \sum_{\substack{\tilde{\mathbf{s}} \in \mathcal{T} \\ \tilde{\mathbf{s}}^{(k)} = \boldsymbol{\alpha}^{(k)}}} P_r(\tilde{\mathbf{s}}) \Bigg/ \sum_{\substack{\tilde{\mathbf{s}} \in \mathcal{T} \\ \tilde{\mathbf{s}}^{(k-1)} = \boldsymbol{\alpha}^{(k-1)}}} P_r(\tilde{\mathbf{s}})
\end{aligned} \tag{14}
$$

where $k = 2, \ldots, n_t$.

In the following, a BFS algorithm is proposed for the generation of the candidate list $\boldsymbol{\Psi}$. First, before carrying out the search, an n_t-level tree, i.e., $G_{\mathcal{T}}$, is constructed from all $\tilde{\mathbf{s}} \in \mathcal{T}$, where a node at level k is associated with a partial signal vector $\boldsymbol{\alpha}^{(k)}$. Since the constellation in the lattice-reduced domain is irregular, the number of nodes per level is not the same, and that leads to different search complexity at different levels. Second, the algorithm given in Table I is applied on the tree $G_{\mathcal{T}}$ using the metric in (13). Starting from the root, the best K_Ψ nodes are retained at each level, and if the number of nodes is less than K_Ψ, all the nodes are retained. At the end, the retained nodes at level n_t are output as $\boldsymbol{\Psi}$, which will be used for the LLR calculation. In the algorithm, $\mathcal{T}(\boldsymbol{\alpha}^{(k)})$ is given by

$$
\mathcal{T}\left(\boldsymbol{\alpha}^{(k)}\right) = \left\{ \tilde{\mathbf{s}} \,|\, \tilde{\mathbf{s}} \in \mathcal{T}, \tilde{\mathbf{s}}^{(k)} = \boldsymbol{\alpha}^{(k)} \right\} \tag{15}
$$

which is employed for an easy calculation of $P_r(\tilde{\mathbf{s}}^{(k)})$. As is shown in (14), the calculation of $P_r(\tilde{\mathbf{s}}^{(k)})$ is more complicated than that in the original-lattice domain because of layer dependence in the lattice-reduced domain.

TABLE I
GENERATION OF Ψ WITH PRIOR INFORMATION

Input: $\tilde{\mathbf{v}}$, $\tilde{\mathbf{R}}$, σ_z^2, $\mathcal{T}$, $\{P_r(\tilde{\mathbf{s}})\}$, K_Ψ, $G_{\mathcal{T}}$
Output: $\boldsymbol{\Psi}$
Initialize: $\boldsymbol{\Psi}^{(0)} = \{\boldsymbol{\alpha}^{(0)} : \text{the root}\}$, $D_\Psi(\boldsymbol{\alpha}^{(0)}) = 0$, $\sum_{\tilde{\mathbf{s}} \in \mathcal{T}(\boldsymbol{\alpha}^{(0)})} P_r(\tilde{\mathbf{s}}) = 1$
for $k = 1$ to n_t do
$\tilde{\mathbf{r}}^{(k)} = [\tilde{r}_{n_t-k+1,n_t-k+1}, \cdots, \tilde{r}_{n_t-k+1,n_t}]^T$
$\mathcal{V}^{(k)} \leftarrow \{$legitimate $\boldsymbol{\alpha}^{(k)}$ extended from $\boldsymbol{\alpha}^{(k-1)} \in \boldsymbol{\Psi}^{(k-1)}$, using the tree $G_{\mathcal{T}}\}$
for $\boldsymbol{\alpha}^{(k)} \in \mathcal{V}^{(k)}$ do
$\mathcal{T}(\boldsymbol{\alpha}^{(k)}) = \{\tilde{\mathbf{s}} \,\|\, \tilde{\mathbf{s}} \in \mathcal{T}, \tilde{\mathbf{s}}^{(k)} = \boldsymbol{\alpha}^{(k)}\}$
$P_r(\alpha_{n_t-k+1} \,\|\, \boldsymbol{\alpha}^{(k-1)}) = \sum_{\tilde{\mathbf{s}} \in \mathcal{T}(\boldsymbol{\alpha}^{(k)})} P_r(\tilde{\mathbf{s}}) \Big/ \sum_{\tilde{\mathbf{s}} \in \mathcal{T}(\boldsymbol{\alpha}^{(k-1)})} P_r(\tilde{\mathbf{s}})$
$D_\Psi(\boldsymbol{\alpha}^{(k)}) = \left\| \tilde{v}_{n_t-k+1} - (\tilde{\mathbf{r}}^{(k)})^T \boldsymbol{\alpha}^{(k)} \right\|^2 - \sigma_z^2 \log P_r(\alpha_{n_t-k+1} \,\|\, \boldsymbol{\alpha}^{(k-1)}) + D_\Psi(\boldsymbol{\alpha}^{(k-1)})$
end for
$\boldsymbol{\Psi}^{(k)} \leftarrow \{\boldsymbol{\alpha}^{(k)} \,\|\,$ those with the smallest K_Ψ $D_\Psi(\boldsymbol{\alpha}^{(k)})$'s$\}$
store $D_\Psi(\boldsymbol{\alpha}^{(k)})$ and $\mathcal{T}(\boldsymbol{\alpha}^{(k)})$ $\forall \boldsymbol{\alpha}^{(k)} \in \boldsymbol{\Psi}^{(k)}$
end for
$\boldsymbol{\Psi} \leftarrow \boldsymbol{\Psi}^{(n_t)}$

B. Generation of $\boldsymbol{\Psi}$ Without Prior Information

For the case of no prior information (noniterative receiver or the first demapping in the iterative receiver), since $P_r(\tilde{\mathbf{s}})$ is the same for all $\tilde{\mathbf{s}} \in \tilde{\mathcal{S}}$, the max-LLR $\phi_{n,i}^o$ is calculated by

$$
\begin{aligned}
\phi_{n,i}^o = &\max_{\tilde{\mathbf{s}} \in \boldsymbol{\Psi}_{n,i}^1} \left(\frac{-1}{\sigma_z^2} \left(\|\tilde{\mathbf{v}} - \tilde{\mathbf{R}}\tilde{\mathbf{s}}\|_F \right)^2 \right) \\
&- \max_{\tilde{\mathbf{s}} \in \boldsymbol{\Psi}_{n,i}^0} \left(\frac{-1}{\sigma_z^2} \left(\|\tilde{\mathbf{v}} - \tilde{\mathbf{R}}\tilde{\mathbf{s}}\|_F \right)^2 \right).
\end{aligned} \tag{16}
$$

In addition, due to lack of prior information, there is no way to use $\mathcal{T}$ to reduce complexity as in the previous section. The generation of $\boldsymbol{\Psi}$ consists of two steps, i.e., candidate signal-vector search followed by legitimacy check. In the step of candidate signal-vector search, we adopt the idea of treating neighboring complex integers around an estimate of a layer symbol as potential legitimate candidates [19]–[23]. In particular, two popular BFSs are employed, i.e., one is based on a predetermined expansion set [19]–[22], and the other is on the on-demand expansion along with distributed sorting [23]. Since the search method in [23] was developed with a real-signal model, for notational consistency of text, only the method using a predetermined expansion set (denoted as Proposed$_{PDES}$) is discussed in detail here; the complexity and performance of the method using on-demand expansion along with distributed sorting (denoted as Proposed$_{ODEDS}$) are evaluated in Sections IV-D and V, respectively.

To begin with, we define a metric for the partial vector $\tilde{\mathbf{t}}^{(k)} \doteq [\tilde{t}_{n_t-k+1}, \ldots, \tilde{t}_{n_t}]^T$ at level k as in (17), shown at the bottom of the next page, where $e(\tilde{\mathbf{t}}^{(k-1)}) \doteq (\tilde{v}_{n_t-k+1} - \sum_{j=n_t-k+2}^{n_t} \tilde{r}_{n_t k+1,j}\tilde{t}_j)/\tilde{r}_{n_t-k+1,n_t-k+1}$, and $D_\Psi(\tilde{\mathbf{t}}^{(0)}) = 0$.

TABLE II
GENERATION OF Ψ WITH NO PRIOR INFORMATION

Input: $\tilde{\mathbf{v}}$, $\tilde{\mathbf{R}}$, σ_z^2, K_Ψ, $\mathbf{T}$, $\boldsymbol{\mathcal{B}}$

Output: $\boldsymbol{\Psi}$

Initialize: $\boldsymbol{\Psi}^{(0)} = \left\{\tilde{\mathbf{t}}^{(0)} = 0;\ \text{the root}\right\}$, $D_\Psi\left(\tilde{\mathbf{t}}^{(0)}\right) = 0$
for $k = 1$ to n_t do
  if $k = n_t$
    $K_\Psi \leftarrow K_\Psi \cdot |\boldsymbol{\mathcal{B}}|$
  end if
  $\boldsymbol{\mathcal{V}}^{(k)} \leftarrow \varnothing$, $\boldsymbol{\Psi}^{(k)} \leftarrow \varnothing$, and $\tilde{\mathbf{r}}^{(k)} = \left[\tilde{r}_{n_t-k+1,n_t-k+1}, \cdots, \tilde{r}_{n_t-k+1,n_t}\right]^T$
  for $\tilde{\mathbf{t}}^{(k-1)} \in \boldsymbol{\Psi}^{(k-1)}$ do
    $e\left(\tilde{\mathbf{t}}^{(k-1)}\right) = \left(\tilde{v}_{n_t-k+1} - \left[\tilde{r}_{n_t-k+1,n_t-k+2}, \cdots, \tilde{r}_{n_t-k+1,n_t}\right]^T \cdot \tilde{\mathbf{t}}^{(k-1)}\right) \Big/ \tilde{r}_{n_t-k+1,n_t-k+1}$
    $\boldsymbol{\mathcal{V}}\left(\tilde{\mathbf{t}}^{(k-1)}\right) \leftarrow \left\{\tilde{\mathbf{t}}^{(k)} \,\middle|\, \begin{array}{l}\text{extended from } \tilde{\mathbf{t}}^{(k-1)} \text{ with} \\ \tilde{t}_{n_t-k+1} = round\left(e\left(\tilde{\mathbf{t}}^{(k-1)}\right)\right) + \beta,\ \beta \in \boldsymbol{\mathcal{B}}\end{array}\right\}$
    $\boldsymbol{\mathcal{V}}^{(k)} \leftarrow \boldsymbol{\mathcal{V}}^{(k)} \cup \boldsymbol{\mathcal{V}}\left(\tilde{\mathbf{t}}^{(k-1)}\right)$
  end for
  for each $\tilde{\mathbf{t}}^{(k)} \in \boldsymbol{\mathcal{V}}^{(k)}$ do
    $D_\Psi\left(\tilde{\mathbf{t}}^{(k)}\right) = \left|\tilde{v}_{n_t-k+1} - \left(\tilde{\mathbf{r}}^{(k)}\right)^T \tilde{\mathbf{t}}^{(k)}\right|^2 + D_\Psi\left(\tilde{\mathbf{t}}^{(k-1)}\right)$
  end for
  $\boldsymbol{\Psi}^{(k)} \leftarrow \left\{\tilde{\mathbf{t}}^{(k)} \,\middle|\, \text{those with smallest } K_\Psi\ D_\Psi\left(\tilde{\mathbf{t}}^{(k)}\right)\text{'s}\right\}$
end for
remove those $\tilde{\mathbf{t}}^{(n_t)} \in \boldsymbol{\mathcal{V}}^{(n_t)}$ having illegitimate $\mathbf{t} = \mathbf{T} \cdot \tilde{\mathbf{t}}^{(n_t)}$
$\boldsymbol{\Psi} = \left\{\tilde{\mathbf{t}}^{(n_t)} \,\middle|\, \text{the best } K_\Psi \text{ legitimate } \tilde{\mathbf{t}}^{(n_t)} \in \boldsymbol{\Psi}^{(n_t)}\right\}$

With the metric in (17), $\boldsymbol{\Psi}$ can be generated in a breadth-first manner, as summarized in Table II. In particular, at level k, sets $\boldsymbol{\mathcal{V}}(\tilde{\mathbf{t}}^{(k-1)}) \doteq \{\tilde{\mathbf{t}}^{(k)} | \tilde{t}_{n_t-k+1} = round(e(\tilde{\mathbf{t}}^{(k-1)})) + \beta, \beta \in \mathcal{B}\}$ and $\boldsymbol{\mathcal{V}}^{(k)} \doteq \cup_{\tilde{\mathbf{t}}^{(k-1)}} \boldsymbol{\mathcal{V}}(\tilde{\mathbf{t}}(k-1))$ are formed, where $\mathcal{B}$ is a set of complex integers closest to the origin with size $|\mathcal{B}| = J$ selected to extend the search space [22]. For example, $\mathcal{B} = \{0, \pm 1 \pm j\}$ for $J = 5$. The best K_Ψ $\tilde{\mathbf{t}}^{(k)}$'s in $\boldsymbol{\mathcal{V}}^{(k)}$ are then searched and retained based on the metric in (17). It is worth noting that in [22], the term $|\tilde{r}_{n_t-k+1,n_t k+1}|^2$ in the calculation of (17) is omitted, leading to large performance degradation, as will be shown in Section V.

At level n_t, the legitimacy of $\tilde{\mathbf{t}}^{(n_t)}$'s is checked before being considered to be retained in $\boldsymbol{\Psi}$. This is a very important step for the LR-aided demapper with no prior information since $\tilde{t}_k$ in $\tilde{\mathbf{t}}^{(k)}$ may not be a legitimate layer symbol. In our method, all the searched $K_\Psi J$ $\tilde{\mathbf{t}}^{(n_t)}$'s at level n_t are transformed back to the original-lattice domain for legitimacy check; this way, the size of $\boldsymbol{\Psi}$ can be kept as large as possible, and that improves the performance of the demapper; the improvement is around 0.3 dB from our experience of extensive simulations. The extra complexity incurred by checking $K_\Psi J$ $\tilde{\mathbf{t}}^{(n_t)}$'s is quite marginal, as compared with checking $K_\Psi J$ $\tilde{\mathbf{t}}^{(n_t)}$'s as in [22], because only addition and comparison operations are involved in legitimacy checking [20], [26], [32]. In addition, checking legitimacy in the original-lattice domain can take advantage of the regular signal constellation to reduce complexity.[3]

C. Calculation of Max-Log Likelihood Ratio

After the generation of $\boldsymbol{\Psi}$, the list-based max-LLRs are calculated for the cases with and without prior information [using (7) for the former with $\tilde{\boldsymbol{\mathcal{S}}}^1_{n,i}$ and $\tilde{\boldsymbol{\mathcal{S}}}^0_{n,i}$ replaced by $\boldsymbol{\Psi}^1_{n,i}$ and $\boldsymbol{\Psi}^0_{n,i}$, respectively, and (16) for the latter]. Note that it may happen that $\boldsymbol{\Psi}^b_{n,i} = \varnothing$, $b \in \{1, 0\}$ for some n and i. In such a case, the LLR needs to be clipped to a fixed value [8], [15]. In particular, as proposed in [8], it is plausible to set the clipped LLR as sign$(0.5 - b) \cdot 8$ if $\boldsymbol{\Psi}^b_{n,i} = \varnothing$, where sign$(x)$ denotes the operation taking sign of x.

D. Complexity Analysis

The number of real multiplications (NRMs) will be used to evaluate the complexity of the proposed demapper because its hardware complexity is much higher than other operations, such as addition/subtraction and taking the maximum of operands. Since the complexity of a demapper is different for the case with or without prior information, the two cases are to be discussed separately.

For the case of no prior information, the demapping operations are divided into three parts: initialization, metric

[3]In [25], legitimacy is checked in the lattice-reduced domain such that additional complexity is required to determine the constellation boundary.

$$
\begin{aligned}
D_\Psi\left(\tilde{\mathbf{t}}^{(k)}\right) &= \left(\left\|\tilde{\mathbf{v}}^{(k)} - \tilde{\mathbf{R}}^{(k)}\tilde{\mathbf{t}}^{(k)}\right\|_F\right)^2 \\
&= \sum_{n=n_t+k-1}^{n_t} \left|\tilde{v}_n - \tilde{r}_{n,n}\tilde{t}_n - \sum_{j=n+1}^{n_t} \tilde{r}_{n,j}\tilde{t}_j\right|^2 \\
&= |\tilde{r}_{n_t-k+1,n_t-k+1}|^2 \cdot \left|\frac{1}{\tilde{r}_{n_t-k+1,n_t-k+1}}\left(\tilde{v}_{n_t-k+1} - \sum_{j=n_t-k+2}^{n_t} \tilde{r}_{n_t-k+1,j}\tilde{t}_j\right) - \tilde{t}_{n_t-k+1}\right|^2 + D_\Psi\left(\tilde{\mathbf{t}}^{(k-1)}\right) \\
&= |\tilde{r}_{n_t-k+1,n_t-k+1}|^2 \cdot \left|e\left(\tilde{\mathbf{t}}^{(k-1)}\right) - \tilde{t}_{n_t-k+1}\right|^2 + D_\Psi\left(\tilde{\mathbf{t}}^{(k-1)}\right)
\end{aligned} \tag{17}
$$

TABLE III
COMPLEXITY OF DIFFERENT DEMAPPERS WITH NO PRIOR INFORMATION

Demapper	Initialization	Metric calculation in list generation	LLR calculation
[12], [36]	$4n_t n_r$	$(4M+2)+K_\Psi \cdot \sum_{k=2}^{n_t}(4k+4M-2)$	$K_\Psi \cdot (4n_t^2+2n_t+1)$
[16]	$4n_t n_r$	$(4J+2)+K_\Psi \cdot \sum_{k=2}^{n_t}(4k+4J-2)$	$K_\Psi \cdot (4n_t^2+2n_t+1)$
[22]	$4n_t n_r$	$(2J+2)+K_\Psi \cdot \sum_{k=2}^{n_t}(4k+2J-2)$	0
[26]	$4n_t n_r + 4n_t^2$	$(2J+2)+(K_\Psi-1) \cdot \sum_{k=2}^{n_t}(4k+2J-2)$	$K_\Psi \cdot (4n_t^2+2n_t+1)$
[37]	$4n_t n_r$	$(4J_1+2)+K_\Psi \cdot \sum_{k=2}^{n_t}(4k+4(J_k+m)-2)$ (J_k is the number of node-expansions of the k-th level)	$K_\Psi \cdot (4n_t^2+2n_t+1)$
Proposed$_{PDES}$	$4n_t n_r$	$(4J+2)+K_\Psi \cdot \sum_{k=2}^{n_t}(4k+4J-2)$	$K_\Psi \cdot (4n_t^2+2n_t+1)$
Proposed$_{ODEDS}$	$4n_t n_r$	$3K_\Psi+2n_t K_\Psi J+6K_\Psi-3+\sum_{k=2}^{2n_t-1}(K_\Psi \cdot k+6K_\Psi-3)$	$K_\Psi \cdot (4n_t^2+2n_t+1)$

TABLE IV
COMPLEXITY OF DIFFERENT DEMAPPERS WITH NO PRIOR INFORMATION ($n_t = n_r = 4, M = 16$)

Parameters	NRMs							
	[12], [36]	[16]	[22]	[26]	[37] with node expansions [16, 9, 9, 9]	Proposed$_{PDES}$	Proposed$_{ODEDS}$	Max-log
$K_\Psi=3$, $J=9$	1015	735	336	535	907	735	694	4784128
$K_\Psi=5$, $J=9$	1605	1157	504	849	1425	1157	1128	
$K_\Psi=8$, $J=9$	2490	1790	756	1320	2202	1790	1779	
$K_\Psi=13$, $J=9$	3965	2845	1176	2105	3497	2845	2864	

calculation in the list generation, and max-LLR calculation. Because the detailed calculation is quite tedious, only the final results are summarized here. Table III shows the complexity of the proposed demapper along with those of the other methods considered in this paper. Note that the complexity of doing LR and (sorted) QR factorizations is not included in Table III because they are only performed once at the beginning of a packet. It is worth noting that, in the initialization stage, the demapper proposed in [26], i.e., the fixed-candidates algorithm (FCA) demapper, has additional complexity since it has to do LR-aided zero forcing over the received signal. Furthermore, the method in [36] is the same as that in [12] in terms of NRMs.

For easy comparison, the complexity for the case of $n_t = 4$, $n_r = 4$, $M = 16$ [16-quadrature amplitude modulation (QAM)] is given in Table IV. As can be seen, the complexity of all list demappers is much lower than that of the max-log demapper. For the proposed demapper, there is a bit difference in complexity between Proposed$_{PDES}$ and Proposed$_{ODEDS}$. Moreover, with $K_\Psi = 3$ ($K_\Psi = 5$), the complexity of the proposed demapper is similar to that in [22] with $K_\Psi = 8$ ($K_\Psi = 13$) and in [26] with $K_\Psi = 5$ ($K_\Psi = 8$), respectively. As will be shown in Section V, however, the proposed demapper has a much superior performance, particularly with Proposed$_{PDES}$.

For the case with prior information, the complexity of the proposed method is variable because the number of possible $\tilde{s}^{(k)}$'s in each of the search levels is different for different $\mathcal{T}$'s. As a result, the cumulative distribution function (CDF) of the required NRMs is employed for evaluating the complexity. Fig. 2 shows the complexity of the proposed demapper for the case of $n_t = 4$, $n_r = 4$, and $M = 16$. As can be seen, the complexity increases with $K_\mathcal{T}$ due to that for larger $K_\mathcal{T}$, more possible $\tilde{s}^{(k)}$'s per search level and all of them will be visited. As will be shown in the following section, $K_\mathcal{T} = 256$ is generally sufficient to achieve desirable performance. Table V shows the complexity comparison of the proposed demapper with $K_\mathcal{T} = 256$ and those in [16], [26], and [36] under one iteration (including one list search and LLR calculation). If the receiver performs n iterations, the complexity is n times that listed in Table V. The proposed demapper has lower average NRMs, but the maximum NRMs is higher than the others.

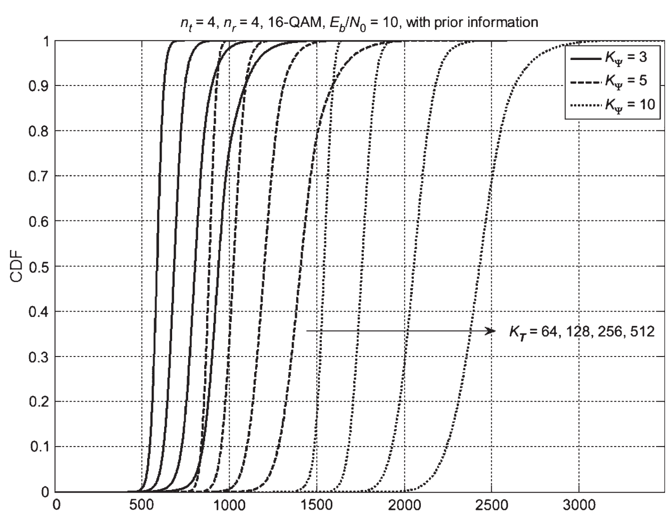

Fig. 2. Complexity of the proposed demapper with prior information under different $K_\mathcal{T}$'s.

In the given complexity analysis, the complexity of finding $\mathbf{T}$ is not taken into consideration. Finding matrix $\mathbf{T}$ is often

TABLE V
COMPLEXITY OF DIFFERENT DEMAPPERS WITH PRIOR INFORMATION ($n_t = n_r = 4, M = 16$)

Parameters	NRMs			
	[16], [36]	[26]	Proposed	
			Average	Maximum
$K_\Psi = 3$	1094	795	803	1380
$K_\Psi = 5$	1770	1441	1199	1882

TABLE VI
SIMULATION PARAMETERS

Number of antennas	$n_t = 4$, $n_r = 4$
Modulation of each layer	QPSK or 16-QAM
δ used in complex LLL	0.99
Number of source bits per packet	9216
Interleaver between encoder and mapper	S-random with lengths 18480 (QPSK) or 18528 (16-QAM)
Internal interleaver for turbo code	S-random with length 9216
Generator polynomial	$(7, 5)_8$ for convolutional code; two $(7_R, 5)_8$'s for turbo code
Code rate	1/2 for convolutional code; 1/3 for turbo code
Constraint length	3
Channel decoder	Max-log MAP
Number of internal iterations for turbo decoder	8

treated as part of preprocessing, and its complexity is shared by symbols within the coherence time of the channel [34]. In slowly time-varying channels, the complexity of finding $\mathbf{T}$ is usually negligible since it can be used for a long time [19], [22]. However, if the channel is rapidly time varying, the complexity of finding $\mathbf{T}$ becomes important. This problem was addressed in [34] where an efficient LR algorithm, which is called complex LLL (CLLL), was proposed to achieve a reduction of as large as 50% in the complexity of the traditional algorithm without sacrificing any performance. Therefore, with CLLL, it is also feasible to use the LR-aided schemes even in rapidly time-varying channels [34]. CLLL is adopted in this paper.

V. SIMULATION RESULTS

Here, the proposed LR-aided list demappers are simulated and compared with the existing demappers in the literature. Table VI summarizes the system parameters adopted in the simulations. The extended channel model in [31] is employed for all simulations, that is, we use the model $\mathbf{y}_{\text{ext}} \doteq [\mathbf{y}^T \; 0_{n_t}^T]^T$, and $\mathbf{H}_{\text{ext}} \doteq [\mathbf{H}^T, \sigma_z \mathbf{I}_{n_t}]^T$, where 0_{n_t} is the length-n_t zero-entry column vector. The CLLL [34] is then carried out over $\mathbf{H}_{\text{ext}}$ to obtain reduced basis $\tilde{\mathbf{H}}_{\text{ext}} = \mathbf{H}_{\text{ext}}\mathbf{T}$. Carrying out the LR on the extended channel $\mathbf{H}_{\text{ext}}$ generally results in a better performance [22], [31], [35]. For other non-LR-aided schemes, the sorted QR factorization is applied on $\mathbf{H}_{\text{ext}}$ [12], [17], [37]. For the spatially correlated channels, $\mathbf{J}_t$ and $\mathbf{J}_r$ are set to be (18), shown at the bottom of the page, which are the same as those in [29].

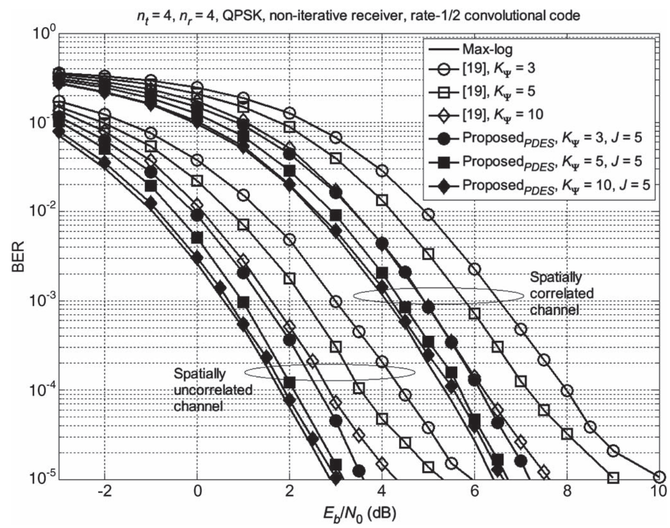

Fig. 3. BER comparison of the proposed demapper with that in [19].

Fig. 3 compares the Proposed$_{PDES}$ demapper with that in [19] in a noniterative receiver for both the spatially correlated and uncorrelated channels under different list sizes. Simple quaternary phase-shift keying (QPSK) is employed because the method in [19] would be too complex to be practical for higher order modulations. In addition, $J = 5$ is used for Proposed$_{PDES}$. As is shown, a gain of more than 2 dB is provided by Proposed$_{PDES}$ for the cases of $K_\Psi = 3$ and $K_\Psi = 5$ and about 1 dB for the case of $K_\Psi = 10$ at the BER $= 10^{-5}$, and by using $K_\Psi = 10$, the proposed demapper performs nearly as the max-log demapper.

Figs. 4–6 compare different demappers in the noniterative receiver with 16-QAM and rate-1/2 convolutional code. In all methods, $J = 9$ is used except for [37], where J's are set to 16, 9, 9, and 9 for search levels 1–4, respectively. The parameters (K_Ψ and J) of the considered demappers are selected for similar complexity (see Table IV). In addition to Proposed$_{PDES}$, Proposed$_{ODEDS}$ is also simulated, and results are given in Figs. 4 and 5 for comparison purposes. As shown in Fig. 4, for the spatially uncorrelated channels, Proposed$_{PDES}$ significantly outperforms the existing demappers with $K_\Psi = 3$; a gain of more than 2 dB is observed at BER $= 10^{-5}$. For the case of $K_\Psi = 5$, Proposed$_{PDES}$ performs similarly to those in [12] and [16] and has a gain of more than 2 dB over the methods in [22] and [26]. As compared with Proposed$_{PDES}$, Proposed$_{ODEDS}$ suffers from a loss of 0.3–0.5 dB in this case. For the spatially correlated channels, as is shown in Fig. 5,

$$\mathbf{J}_t = \mathbf{J}_r = \begin{bmatrix} 1 & 0.01 + 0.7i & -0.47 - 0.08i & 0.19 - 0.26i \\ 0.01 - 0.7i & 1 & 0.01 + 0.7i & -0.47 - 0.08i \\ -0.47 + 0.08i & 0.01 - 0.7i & 1 & 0.01 + 0.7i \\ 0.19 + 0.26i & -0.47 + 0.08i & 0.01 - 0.7i & 1 \end{bmatrix} \tag{18}$$

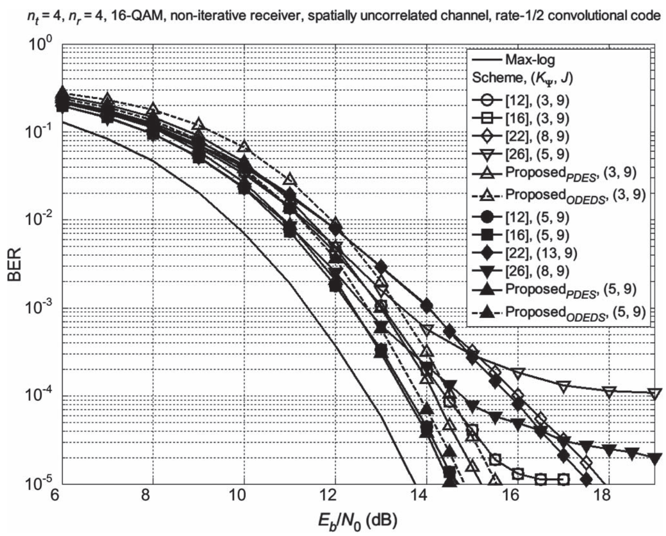

Fig. 4. BER comparison of different demappers for spatially uncorrelated channels.

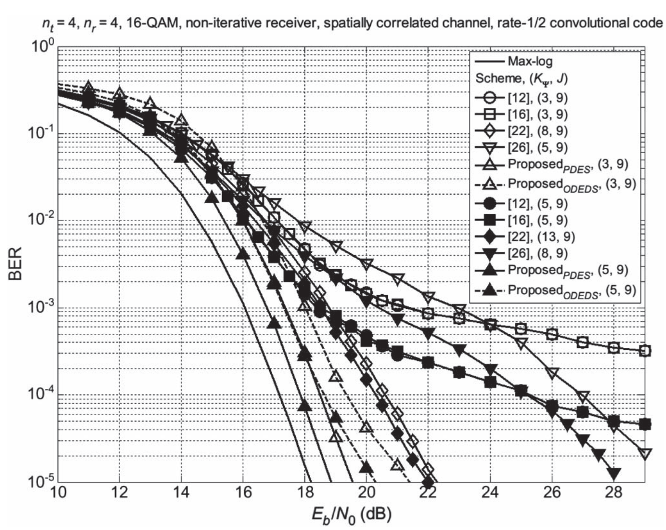

Fig. 5. BER comparison of different demappers for spatially correlated channels.

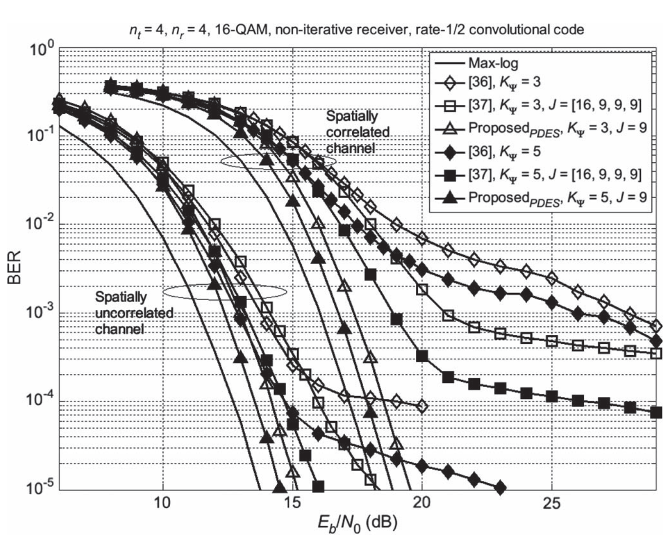

Fig. 6. BER comparison of the proposed method with that in [36] and [37].

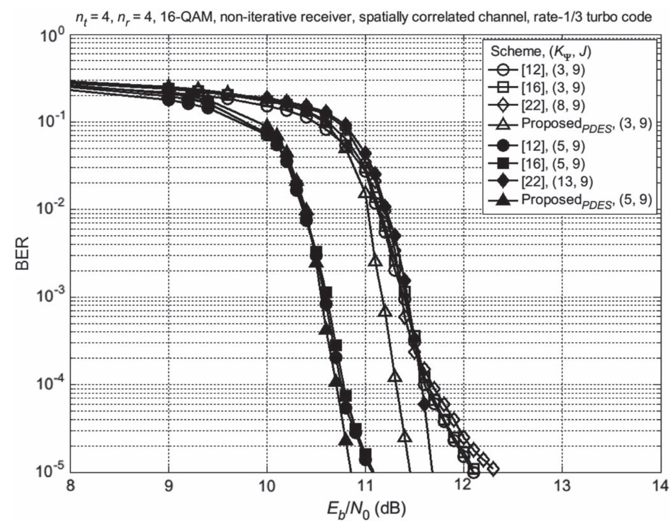

Fig. 7. BER comparison of different demappers for spatially correlated channels with turbo code.

Proposed$_{PDES}$ has a gain of at least 2 dB over other demappers for both cases of $K_\Psi = 3$ and $K_\Psi = 5$. As compared with Proposed$_{PDES}$, the performance loss with Proposed$_{ODEDS}$ becomes larger; losses of 1.8 and 1.5 dB are observed for $K_\Psi = 3$ and $K_\Psi = 5$, respectively. In Fig. 6, Proposed$_{PDES}$ is compared with methods in [36] and [37]. As is seen, for the spatially uncorrelated channels, Proposed$_{PDES}$ outperforms [36] ([37]) by a margin of 5 (1.5) and 6 (3.5) dB for $K_\Psi = 5$ and $K_\Psi = 3$, respectively. For the spatially correlated channels, the gains obtained with the proposed method are more than 10 dB.

Fig. 7 shows the simulation results for the noniterative receiver with a turbo code given in Table VI. Eight iterations are performed in the turbo decoder. As can be seen, the proposed method with $K_\Psi = 3$ ($K_\Psi = 5$) has a gain of 0.6 (0.3) dB over [12] and [16] and about 0.8 (0.9) dB over [22] with $K_\Psi = 8$ ($K_\Psi = 13$). As compared with the results in Fig. 5, the gains obtained with the proposed method become smaller under a powerful channel code. This is something that one might expect because, in this case, the adverse effect of noise enhancement (the effect that the proposed LR-aided demapper intends to remove) can be counteracted by the powerful outer code.

For the iterative receiver, Fig. 8 shows the BER performance of the proposed demapper with different values of $K_{\mathcal{T}}$. Recall that $K_{\mathcal{T}}$ is the size of $\boldsymbol{\mathcal{T}}$ from which the candidate list $\boldsymbol{\Psi}$ is constructed so as to reduce the complexity of the demapper. As is seen, there is almost no performance loss with $K_{\mathcal{T}} \geq 256$. In addition, the iterative receiver indeed provides significant gains over the noniterative receiver; a gain of more than 3 dB is observed with only one iteration, and an additional 1-dB gain is obtained with two iterations at BER $= 10^{-5}$. The gain, however, diminishes with more than two iterations. In the following, $K_{\mathcal{T}} = 256$ is used for the proposed demapper, and two iterations are performed for all the considered methods.

Fig. 9 compares the proposed demapper in the iterative receiver with that in [16], the FCA demapper in [26], and modified K-best demapper in [36] under the spatially correlated and uncorrelated channels. For the spatially uncorrelated channels, both the proposed demapper and that in [16] have a

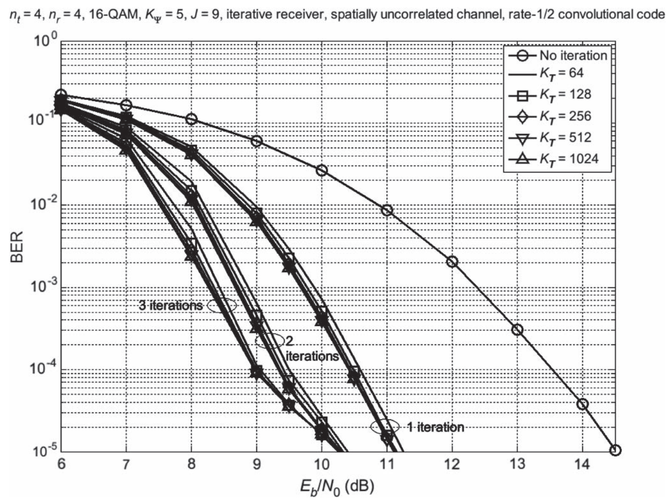

Fig. 8. Effect of $K_{\mathcal{T}}$ on the BER performance.

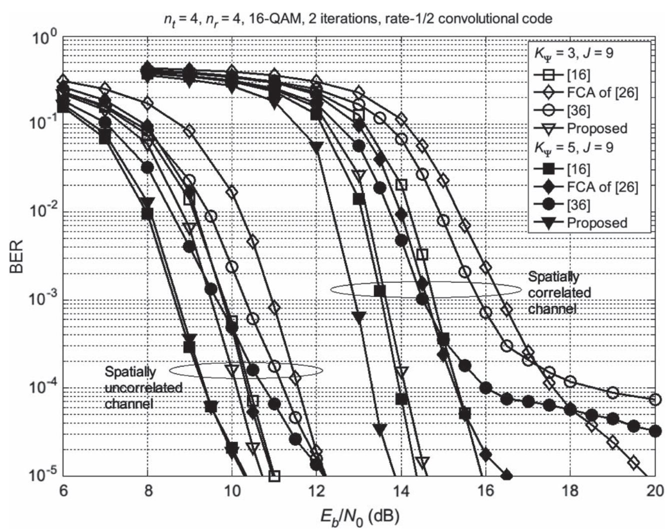

Fig. 9. BER comparison of the proposed demapper with that in [16], [26], and [36].

similar performance and are about 0.8 and 1.9 dB better than FCA and [36] at BER $= 10^{-5}$ for $K_\Psi = 5$, respectively. For the case of $K_\Psi = 3$, the proposed demapper is about 0.2 dB over that in [16] and is about 1.4 dB better than FCA and [36]. For the spatially correlated channels, the gains provided by the proposed demapper become more prominent, i.e., for $K_\Psi = 5$, 0.4 dB over [16], 2.1 dB over FCA, and 6.5 dB over [36] and for $K_\Psi = 3$, 1.5 dB over [16], 5 dB over FCA, and more than 6 dB over [36]. Complexity-wise, as compared in Table V, the average NRMs of the proposed demapper are lower than that of [16], FCA, and [36], whereas the maximum NRM is higher with the proposed demapper. In Fig. 9, for a fair comparison, the number of iterations is set to two for all methods. For some other methods, however, more than two iterations may be needed to have the best performance.

In Fig. 10, the number of iterations is set large enough to investigate the achievable performance of the considered methods in this paper for the spatially correlated channels with $K_\Psi = 5$. As can be seen, the performance improvement beyond seven iterations is diminishing for all methods. In particular,

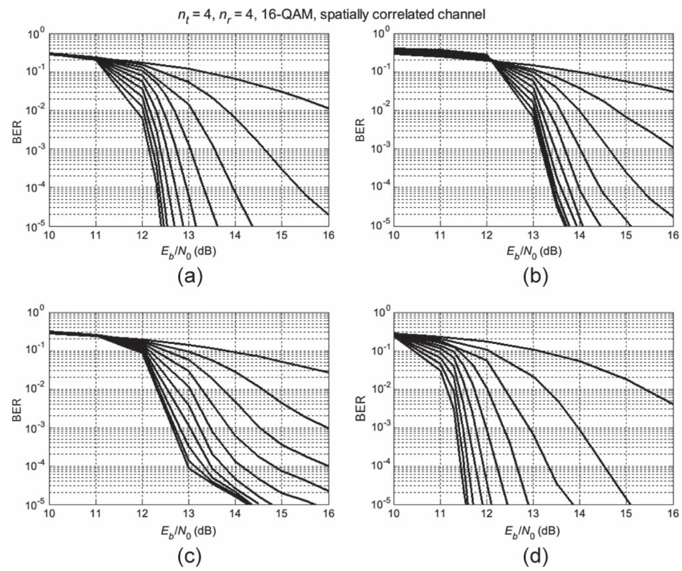

Fig. 10. BER performance under spatially correlated channels. In each subfigure, curves from right to left represent the performance of no iteration to the ninth iteration, respectively.

12.4, 13.7, 14.2, and 11.5 dB are needed for [16], [26], [36], and the proposed method to achieve BER $= 10^{-5}$, respectively. In other words, the proposed method is at least 0.9 dB better than others. The performance improvement offered by the proposed method is expected to be larger for a smaller K_Ψ, as was already shown in Fig. 9.

VI. Conclusion

In this paper, a new LR-aided list demapper for coded MIMO receivers has been proposed, where the candidate list is generated after cancelation of multilayer interference in the lattice-reduced domain. The newly designed metric, legitimacy check, and the use of prior information in the candidate-list generation entail the proposed demapper a superior performance for both the iterative and noniterative receivers, as compared with the existing methods. The performance improvement is particularly prominent for the cases with a small list size and/or under a spatially correlated channel. A two-step algorithm is also devised to reduce the complexity of the demapper for application in the iterative receiver, where the prior information from the decoder is exploited to improve performance.

References

[1] J. Foschini, Jr., "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, Summer 1996.

[2] E. Telatar, "Capacity of multi-antenna Gaussian channels," *Eur. Trans. Telecommun.*, vol. 10, no. 6, pp. 585–595, Nov./Dec. 1999.

[3] *IEEE Standard for Local and Metropolitan Area Networks, Part 16: Air Interface for Broadband Wireless Access Systems*, IEEE Std. 802.16e-2009, 2009.

[4] *3GPP Technical Specification Group Radio Access Network, Evolved Universal Terrestrial Radio Access (E-UTRA), Multiplexing and channel coding (release 9)*, 3GPP TS 36.212, Sep. 2011.

[5] J. Hagenauer, "The turbo principle: Tutorial introduction and state of the art," in *Proc. Int. Symp. Turbo Codes Related Topics*, Brest, France, Sep. 1997, pp. 1–11.

[6] M. Tüchler, R. Koetter, and C. Singer, "Turbo equalization: Principles and new results," *IEEE Trans. Commun.*, vol. 50, no. 5, pp. 754–767, May 2002.

[7] M. Sellathurai and S. Haykin, "Turbo-BLAST for wireless communications: Theory and experiments," *IEEE Trans. Signal Process.*, vol. 50, no. 10, pp. 2538–2546, Oct. 2002.

[8] B. 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.

[9] 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.

[10] S. Bäro, J. Hagenauer, and M. Witzke, "Iterative detection of MIMO transmission using a list sequential (LISS) detector," in *Proc. IEEE Int. Conf. Commun.*, Anchorage, AK, USA, May 2003, pp. 2653–2657.

[11] K.-W. Wong, C.-Y. Tsui, and S.-K. Cheng, "A VLSI architecture of a K-best lattice decoding algorithm for MIMO channels," in *Proc. IEEE Int. Symp. Circuits. Syst.*, May 2002, pp. III-273–III-276.

[12] 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.

[13] S. Mondal, M. Eltawil, and N. Salama, "Architectural optimizations for low-power K-best MIMO decoders," *IEEE Trans. Veh. Technol.*, vol. 58, no. 7, pp. 3145–3153, Sep. 2009.

[14] C.-H. Liao, T.-P. Wang, and D. Chiueh, "A 74.8 mW soft-output detector IC for 8×8 spatial-multiplexing MIMO communications," *IEEE J. Solid-State Circuits*, vol. 45, no. 2, pp. 411–421, Feb. 2010.

[15] C. Mehlführer, D. Seethaler, G. Matz, and M. Rupp, "An iterative MIMO-HSDPA receiver based on a K-Best-MAP algorithm," in *Proc. IEEE Global Telecommun. Conf.*, Nov. 2006, pp. 1–5.

[16] L. Ruyet, T. Bertozzi, and B. Özbek, "Breadth first algorithms for APP detectors over MIMO channels," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2004, pp. 926–930.

[17] L. Milliner, E. Zimmermann, R. Barry, and P. Fettweis, "A framework for fixed complexity breadth-first MIMO detection," in *Proc. IEEE Int. Symp. Spread Spectr. Tech. Appl.*, Bologna, Italy, Aug. 2008, pp. 129–132.

[18] B. Reid, J. Grant, and P. Kind, "Low complexity list detection for high-rate multiple-antenna channels," in *Proc. IEEE Int. Symp. Inf. Theory*, Yokohama, Japan, Jun. 2003, p. 273.

[19] P. Silvola, K. Hooli, and M. Juntti, "Suboptimal soft-output MAP detector with lattice reduction," *IEEE Signal Process. Lett.*, vol. 13, no. 6, pp. 321–324, Jun. 2006.

[20] L. Milliner and R. Barry, "CTH09-4: A lattice-reduction-aided soft detector for multiple-input multiple-output channels," in *Proc. IEEE Global Telecommun. Conf.*, San Francisco, CA, USA, Nov. 2006, pp. 1–5.

[21] V. Ponnampalam, D. McNamara, A. Lillie, and M. Sandell, "On generating soft outputs for lattice-reduction-aided MIMO detection," in *Proc. IEEE Int. Conf. Commun.*, Glasgow, U.K., Jun. 2007, pp. 4144–4149.

[22] X.-F. Qi and K. Holt, "A lattice-reduction-aided soft demapper for high-rate coded MIMO-OFDM systems," *IEEE Signal Process. Lett.*, vol. 14, no. 5, pp. 305–308, May 2007.

[23] M. Shabany and G. Gulak, "The application of lattice-reduction to the K-best algorithm for near-optimal MIMO detection," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2008, pp. 316–319.

[24] M. Shabany, K. Su, and G. Gulak, "A pipelined scalable high-throughput implementation of a near-ML K-best complex lattice decoder," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Mar. 2008, pp. 3173–3176.

[25] S. Roger, A. Gonzalez, V. Almenar, and M. Vidal, "On decreasing the complexity of lattice-reduction-aided K-best MIMO detectors," in *Proc. Eur. Signal Process. Conf.*, Glasgow, U.K., Aug. 2009, pp. 2411–2415.

[26] W. Zhang and X. Ma, "Low-complexity soft-output decoding with lattice-reduction-aided detectors," *IEEE Trans. Commun.*, vol. 58, no. 9, pp. 2621–2629, Sep. 2010.

[27] P. Kermoal, L. Schumacher, I. Pedersen, E. Mogensen, and F. Frederiksen, "A stochastic MIMO radio channel model with experimental validation," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 6, pp. 1211–1226, Aug. 2002.

[28] D. Wübben, V. Küne, and K.-D. Kammeyer, "On the robustness of lattice-reduction aided detectors in correlated MIMO systems," in *Proc. IEEE Veh. Technol. Conf.*, Los Angeles, CA, USA, Sep. 2004, vol. 5, pp. 3639–3643.

[29] G. Barbero and S. Thompson, "Performance of the complex sphere decoder in spatially correlated MIMO channels," *IET Commun.*, vol. 1, no. 1, pp. 122–130, Feb. 2007.

[30] G. Barbero and S. Thompson, "Fixing the complexity of the sphere decoder for MIMO detection," *IEEE Trans. Wireless Commun.*, vol. 7, no. 6, pp. 2131–2142, Jun. 2008.

[31] D. Wübben, R. Böhnke, V. Kühn, and K. Kammeyer, "Near-maximum-likelihood detection of MIMO systems using MMSE-based lattice reduction," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2004, pp. 798–802.

[32] X. Ma and W. Zhang, "Performance analysis for MIMO systems with lattice-reduction aided linear equalization," *IEEE Trans. Commun.*, vol. 56, no. 2, pp. 309–318, Feb. 2008.

[33] K. Lenstra, W. Lenstra, Jr., and L. Lovász, "Factoring polynomials with rational coefficients," *Math. Ann.*, vol. 261, no. 4, pp. 515–534, Dec. 1982.

[34] H. Gan and H. Mow, "Complex lattice reduction algorithm for low-complexity full-diversity MIMO detection," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2701–2710, Jul. 2009.

[35] 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.

[36] J. Ketonen, M. Juntti, and R. Cavallaro, "Performance–Complexity comparison of receivers for a LTE MIMO-OFDM system," *IEEE Trans. Signal Process.*, vol. 58, no. 6, pp. 3360–3372, Jun. 2010.

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

Tung-Jung Hsieh was born in Kaohsiung, Taiwan, in 1978. He received the B.S. degree from National Chiao Tung University (NCTU), Hsinchu, Taiwan, in 2000 and the M.S. degree from National Sun Yat-Sen University, Kaohsiung, Taiwan, in 2004, both in electrical engineering. He is currently working toward the Ph.D. degree in electrical engineering with NCTU.

His research interests include baseband signal processing of communication systems, lattice-reduction-aided techniques for multiple-input–multiple-output systems, and soft-input–soft-output demapping algorithms.

Wern-Ho Sheen (M'91) received the Ph.D. degree in electrical engineering from Georgia Institute of Technology, Atlanta, GA, USA, in 1991.

From 1991 to 1993, he was an Associate Researcher with Chunghwa Telecom Laboratories, Taoyuan, Taiwan. From 1993 to 2001, he was a Professor with the Department of Electrical Engineering, National Chung Cheng University, Chiayi, Taiwan. From 2001 to 2009, he was a Professor with the Department of Communications Engineering, National Chiao Tung University, Hsinchu, Taiwan. From 2009 to 2013, he was a Professor with the Department of Information and Communication Engineering, Chaoyang University of Technology, Taichung, Taiwan. Since 2013, he has been a Professor with the Department of Communications Engineering, National Chung Cheng University. His research interests include communication theory, wireless communication systems, and signal processing for communications.