

Low-Complexity Soft-Output MIMO Detectors Based on Optimal Channel Puncturing

Mohammad M. Mansour, *Senior Member, IEEE*

Abstract

Channel puncturing transforms a multiple-input multiple-output (MIMO) channel into a sparse lower-triangular form using the so-called WL decomposition scheme in order to reduce tree-based detection complexity. We propose computationally efficient soft-output detectors based on two forms of channel puncturing: *augmented* and *two-sided*. The augmented WL detector (AWLD) employs a punctured channel derived by triangularizing the true channel in augmented form, followed by left-sided Gaussian elimination. The two-sided WL detector (dubbed WLZ) employs right-sided reduction and left-sided elimination to puncture the channel. We prove that augmented channel puncturing is optimal in maximizing the lower-bound on the achievable information rate (AIR) based on a new mismatched detection model. We show that the AWLD decomposes into an MMSE prefilter and channel gain compensation stages, followed by a regular WL detector (WLD) that computes least-squares soft-decision estimates. Similarly, WLZ decomposes into a pre-processing reduction step followed by WLD. AWLD attains the same performance as the existing AIR-based partial marginalization (PM) detector, but with less computational complexity. We empirically show that WLZ attains the best complexity-performance tradeoff among tree-based detectors.

Index Terms

MIMO detectors, MMSE, achievable information rate, partial marginalization, channel puncturing

I. INTRODUCTION

Modern communication systems rely on multiple-input multiple-output (MIMO) antenna configurations with large dimensions to support the aggressive targets set on spectral efficiencies. However, achieving the ideal performance promised by MIMO technology requires detectors

Parts of this work have been presented at IEEE ICC 2020 [1].

This work is supported by the University Research Board (URB) at the American University of Beirut.

M. M. Mansour is with the Department of Electrical and Computer Engineering, American University of Beirut, Beirut 1107 2020, Lebanon (e-mail: mmansour@aub.edu.lb; mmansour@ieee.org).

This work has been submitted to IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS for possible publication. Copyright may be transferred without notice, after which this version may no longer be accessible.

whose complexity grows exponentially in MIMO dimensions. To support low-latency communications while providing high throughput rates, computationally efficient designs of MIMO detectors that do not incur substantial performance loss are essential.

MIMO detection is a classical problem in communications, and the literature is rich with schemes that provide various performance-complexity tradeoffs in the design space (e.g., [2], [3]). The benchmark for performance in the sense of generating ‘good’ soft decisions on the transmitted bits is maximum likelihood (ML) detection, which provides optimal performance but with exponential complexity. Alternatively, the benchmarks for low-complexity detection are the zero-forcing (ZF) and minimum mean-square error (MMSE) schemes, which decouple the transmit layers through linear filtering to generate log-likelihood ratios (LLRs) for each symbol bit in parallel, or sequentially with decision feedback.

Tree-search based detectors such as sphere decoding [4], list decoding [5], and other variants map the detection problem into a search problem for the closest signal vector. They find the closest vector in N -dimensional signal space to the received vector by forming a search-tree and recursively enumerating symbols across all layers from the parent down to the leaves. Such schemes suffer from non-deterministic search-time complexity (see [6]–[8]). To simplify the search process, fixed-complexity schemes such as [9]–[11] limit the search steps to a set of survivor paths. While these schemes are efficient in finding the ML path, they do not necessarily find all the best competing paths that are needed to generate soft decisions for each symbol bit.

An alternative concept is partial marginalization (PM) [12], [13], which exhaustively enumerates only over a small subset of ν carefully chosen parent layers out of N , and approximately marginalizes over the other $N-\nu$ child layers using ZF with decision-feedback (ZF-DF) estimates. While the bit LLRs for parent symbols are easy to compute, computing bit LLRs for child symbols is complicated by three facts: 1) for each bit hypothesis of the child symbols, a separate ZF-DF process is needed, which is compute-intensive for large N ; 2) the LLRs are prone to error propagation for large N due to decision feedback; 3) the quality of the LLRs is very sensitive to the choice of the ν parent layers. In [14], the closely related layered orthogonal lattice detector (LORD) scheme mitigates the first drawback by operating with $\nu=1$ and computing bit LLRs for the parent symbol only; N independent searches using N trees are performed to compute the bit LLRs for all symbols by choosing a new symbol as a parent in each tree.

To overcome the second drawback, the so-called WL detection (WLD) ¹ scheme [15] first applies a (non-unitary) filtering matrix \mathbf{W} to transform the channel into sparse lower-triangular form \mathbf{L} . It then enumerates across one parent layer and detects symbols in all other child layers in parallel via least-squares (LS) estimates without decision feedback. The channel matrix is “punctured” to have a special structure that breaks the connections among child nodes, while retaining connections only to the parents. Essentially, all child nodes become leaves, and hence, marginalization is exact in the LS sense. An immediate consequence is that the LS estimates of the counter hypotheses of each leaf symbol bit can be easily derived from the LS estimate itself [16]. A closely related concept is the achievable information rate (AIR)-PM detector [17], [18], which derives a “shortened” channel similar to the WLD’s punctured structure using information-theoretic optimizations. Other optimal linear detectors are presented in [19].

In this paper, we show that the concepts of channel puncturing of [15] and AIR-PM-based channel shortening of [18] are related. After introducing the system model and reviewing tree-based detection in Sec. II, we present a matrix characterization of one-sided and two-sided channel puncturing based on Gaussian elimination and lattice reduction in Sec. III. In Sec. IV, we present the WLD detection model and derive a lower bound on the achievable rate of the WLD detector, as well as a bound on the quality of its hard decision estimate, and show that these bounds approach capacity and the hard ML decision as the puncturing order increases. In Sec. V, we propose a new *augmented* WLD (AWLD) detection scheme, in which an augmented channel, rather than the true channel, is punctured. We derive a lower bound on the AIR of the AWLD detector and characterize its gap to capacity. In Sec. VI, we propose an alternate mismatched detection model compared to [17], and use it to derive optimal punctured channels that maximize the AIR. We prove that the AWLD detector is optimal under this model, and is in fact equivalent to the AIR-PM detector of [18]. The AWLD detector decomposes into an MMSE prefilter and channel gain compensation stages, followed by a WLD detector. Hence, AIR-optimal channel puncturing can be achieved using simple QL decomposition followed by Gaussian elimination. In Secs. VII-VIII, we present computationally efficient matrix decomposition, puncturing, and MIMO detection algorithms based on the proposed schemes. Empirical simulation results are presented in Sec. IX. Finally, Sec. X concludes the paper. The supplementary material includes

¹The WL decomposition is defined to be a decomposition of the matrix \mathbf{H} as $\mathbf{W}\mathbf{H}=\mathbf{L}$, where \mathbf{W} is a (non-unitary) filtering matrix and \mathbf{L} is a sparse lower-triangular matrix. A detector that applies WL decomposition to the channel matrix \mathbf{H} and detects symbols based on \mathbf{L} is called a WL detector.

proofs, pseudo-codes of all proposed algorithms, and enlarged figures.

Notation: $i = \sqrt{-1}$; $\mathcal{Z}, \mathcal{R}, \mathcal{C}, \mathcal{G} = \mathcal{Z} + i\mathcal{Z}$ are the sets of integers, reals, complex numbers, and Gaussian integers; $\mathbf{a} = [a_k]$ column vector with elements a_k ; $\mathbf{A} = [a_{kj}]$ matrix with elements a_{kj} ; $[\mathbf{A}]_k = [a_{k1}, \dots, a_{kk}]$; $[\mathbf{A}]_{\bar{k}} = [a_{k1}, \dots, a_{k,k-1}]$; $[\mathbf{A}]_{\bar{1}} = \emptyset$; $\mathbf{0}_{M \times N} = M \times N$ zero matrix; $\mathbf{I}_N = N \times N$ identity matrix; $\mathbf{e}_k = k$ th column of \mathbf{I} ; $\mathbb{E}[\cdot]$ = expectation; $\mathcal{CN}(\mathbf{m}, \mathbf{C})$ denotes circularly-symmetric complex Gaussian distribution with mean \mathbf{m} and covariance matrix \mathbf{C} ; $(\cdot)^T$ = transpose; $(\cdot)^\dagger$ = Hermitian transpose; $\Re\{\cdot\}, \Im\{\cdot\}$ = real, imaginary part; $\text{diag}(\cdot)$ = matrix diagonal; $\det(\cdot)$ = determinant; $\|\cdot\| = L_2$ norm; $\|\cdot\|_F$ = Frobenius norm; $\mathbf{A}^{1/2}$ = matrix square-root; $\mathbf{A} \succeq \mathbf{B}$ denotes $(\mathbf{A} - \mathbf{B})$ positive semidefinite; \cong denotes equality up to an additive constant.

II. SYSTEM MODEL AND LAYERED DETECTION

Let $\mathbf{H} \in \mathcal{C}^{M \times N}$ model a MIMO communication channel with N transmit antennas and $M \geq N$ receive antennas. The transmit signal $\mathbf{x} = [x_n] \in \mathcal{X}^{N \times 1}$ is composed of N symbols x_n drawn from constellation \mathcal{X} with average energy $\mathbb{E}[x_n x_n^\dagger] = E_s$ and size $|\mathcal{X}| = Q$. Each symbol x_n is mapped from $B = \log_2 Q$ bits $x_{n,b} \in \{\pm 1\}$ as $x_n = (x_{n,b})_{b=1}^B$. Assuming \mathbf{H} is perfectly known only at the receiver, the receive signal $\mathbf{y} \in \mathcal{C}^{M \times 1}$ is modeled using the input-output relation

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n}, \quad (1)$$

where the noise term $\mathbf{n} \sim \mathcal{CN}(\mathbf{0}_{M \times 1}, N_0 \mathbf{I}_M)$ and N_0 is the noise variance. The conditional probability $p(\mathbf{y}|\mathbf{x})$ and metric $\mu(\mathbf{y}|\mathbf{x})$ according to (1) are

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{(\pi N_0)^M} \exp(\mu(\mathbf{y}|\mathbf{x})), \quad (2)$$

$$\mu(\mathbf{y}|\mathbf{x}) = -\frac{1}{N_0} \|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 \quad (3)$$

$$= -\frac{1}{N_0} (\mathbf{y}^\dagger \mathbf{y} - 2\Re\{\mathbf{y}^\dagger \mathbf{H}\mathbf{x}\} + \mathbf{x}^\dagger \mathbf{H}^\dagger \mathbf{H}\mathbf{x}) \quad (4)$$

$$\cong 2\Re\{\mathbf{y}^\dagger \mathbf{H}\mathbf{x}\} - \mathbf{x}^\dagger \mathbf{H}^\dagger \mathbf{H}\mathbf{x}. \quad (5)$$

Using the observation \mathbf{y} and assuming no prior information on \mathbf{x} (i.e., $P(x_{n,b} = +1) = P(x_{n,b} = -1) = \frac{1}{2}$), the ML detector generates the LLR of the b th bit $x_{n,b}$ of the n th symbol x_n in \mathbf{x} as

$$L(x_{n,b}|\mathbf{y}) = \ln \frac{\sum_{\mathbf{x}: x_{n,b}=+1} \exp(\mu(\mathbf{y}|\mathbf{x}))}{\sum_{\mathbf{x}: x_{n,b}=-1} \exp(\mu(\mathbf{y}|\mathbf{x}))}. \quad (6)$$

To avoid computing sums of exponentials in (6), the ML detector with Max-Log approximation (MLM) recursively applies the Jacobian approximation $\ln(e^c + e^d) \approx \max(c, d)$ [20] to the exponentials in (6), and approximates $L(x_{n,b}|\mathbf{y})$ by $\Lambda(x_{n,b}|\mathbf{y})$ as

$$\Lambda(x_{n,b}|\mathbf{y}) = \max_{\mathbf{x}: x_{n,b}=+1} \mu(\mathbf{y}|\mathbf{x}) - \max_{\mathbf{x}: x_{n,b}=-1} \mu(\mathbf{y}|\mathbf{x}). \quad (7)$$

In the absence of any structure on \mathbf{H} or any further simplifying assumptions, computing the sums in (6) or the max terms in (7) have exponential complexities in N .

A. Tree-based Layered Detection

Detecting symbols and generating bit LLRs can be done efficiently on a tree. By triangularizing \mathbf{H} and associating symbols with edges and partial Euclidean distances with nodes, symbols can be detected by searching the tree for a path from the root to a leaf with minimal weight.

Let $\mathbf{H} = \mathbf{Q}\mathbf{L}$ denote the thin QL decomposition (QLD) [21] of \mathbf{H} , where $\mathbf{Q} \in \mathcal{C}^{M \times N}$ has orthonormal columns ($\mathbf{Q}^\dagger \mathbf{Q} = \mathbf{I}$) and $\mathbf{L} \in \mathcal{C}^{N \times N}$ is a square lower-triangular matrix with real and positive diagonal elements. We write $\mathbf{y} - \mathbf{H}\mathbf{x}$ in terms of \mathbf{Q} , \mathbf{L} as $\mathbf{y} - \mathbf{H}\mathbf{x} = \mathbf{Q}(\mathbf{Q}^\dagger \mathbf{y} - \mathbf{L}\mathbf{x}) + (\mathbf{I} - \mathbf{Q}\mathbf{Q}^\dagger)\mathbf{y}$. Since $\mathbf{Q} \perp (\mathbf{I} - \mathbf{Q}\mathbf{Q}^\dagger)$, i.e., $\mathbf{Q}^\dagger(\mathbf{I} - \mathbf{Q}\mathbf{Q}^\dagger) = \mathbf{0}$, the squared-distance in (3) can be expanded as $\|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2 = \|\mathbf{Q}(\mathbf{Q}^\dagger \mathbf{y} - \mathbf{L}\mathbf{x})\|^2 + \|(\mathbf{I} - \mathbf{Q}\mathbf{Q}^\dagger)\mathbf{y}\|^2$. Since $\mathbf{Q}^\dagger \mathbf{Q} = \mathbf{I}$, then \mathbf{Q} does not scale Euclidean distances. Also the term $\|(\mathbf{I} - \mathbf{Q}\mathbf{Q}^\dagger)\mathbf{y}\|^2$ is independent of \mathbf{x} and hence is irrelevant for detection. Thus, it suffices to work with the quantity $\|\tilde{\mathbf{y}} - \mathbf{L}\mathbf{x}\|^2$ rather than $\|\mathbf{y} - \mathbf{H}\mathbf{x}\|^2$ in (3), with $\tilde{\mathbf{y}} = \mathbf{Q}^\dagger \mathbf{y}$.

With proper layer ordering and partial marginalization, the tree can be searched by enumerating only over a subset of ν parent layers, rather than all the layers. Let $\mathbf{x}_1 = [x_1, \dots, x_\nu]^\top$ and $\mathbf{x}_2 = [x_{\nu+1}, \dots, x_N]^\top$ denote the parent and child symbol vectors, respectively. Let $\tilde{\mathbf{y}}_1, \tilde{\mathbf{y}}_2$ be similarly defined from $\tilde{\mathbf{y}}$. Define the variables w_k and z_k as

$$w_k = \tilde{y}_k - \sum_{j=1}^{\min\{k, \nu\}} l_{kj} x_j, \quad z_k = w_k - \sum_{j=\nu+1}^{k-1} l_{kj} x_j, \quad (8)$$

for $k = 1, \dots, N$. Note that w_k depends only on \mathbf{x}_1 , while z_k depends on both \mathbf{x}_1 and \mathbf{x}_2 . The weight of a parent node ($1 \leq k \leq \nu$) and a child node ($\nu+1 \leq k \leq N$) are given by

$$e_1(w_k) = -\frac{1}{N_0} |w_k|^2, \quad 1 \leq k \leq \nu, \quad (9)$$

$$e_2(z_k, x_k) = -\frac{1}{N_0} |z_k - l_{kk} x_k|^2, \quad \nu+1 \leq k \leq N. \quad (10)$$

The weight of a path associated with symbols $\mathbf{x} = [\mathbf{x}_1; \mathbf{x}_2]$ is

$$\mu(\tilde{\mathbf{y}}|\mathbf{x}) = \sum_{k=1}^{\nu} e_1(w_k) + \sum_{k=\nu+1}^N e_2(z_k, x_k) \triangleq \mu_1(\tilde{\mathbf{y}}_1|\mathbf{x}_1) + \mu_2(\tilde{\mathbf{y}}_2|\mathbf{x}_1, \mathbf{x}_2). \quad (11)$$

Maximizing $\mu(\tilde{\mathbf{y}}|\mathbf{x})$ over all \mathbf{x} such that $x_{n,b}$ is $s = \pm 1$, for $b = 1, \dots, B$, $n = 1, \dots, N$, can be expressed using (11) as

$$\max_{\substack{\mathbf{x}: x_{n,b}=s \\ 1 \leq n \leq \nu}} \mu(\tilde{\mathbf{y}}|\mathbf{x}) = \max_{\substack{\mathbf{x}_1: \\ x_{n,b}=s}} \left\{ \mu_1(\tilde{\mathbf{y}}_1|\mathbf{x}_1) + \max_{\mathbf{x}_2} \mu_2(\tilde{\mathbf{y}}_2|\mathbf{x}_1, \mathbf{x}_2) \right\}, \quad (12)$$

$$\max_{\substack{\mathbf{x}: x_{n,b}=s \\ \nu+1 \leq n \leq N}} \mu(\tilde{\mathbf{y}}|\mathbf{x}) = \max_{\mathbf{x}_1} \left\{ \mu_1(\tilde{\mathbf{y}}_1|\mathbf{x}_1) + \max_{\substack{\mathbf{x}_2: \\ x_{n,b}=s}} \mu_2(\tilde{\mathbf{y}}_2|\mathbf{x}_1, \mathbf{x}_2) \right\}. \quad (13)$$

The inner max operations in (12)-(13) can be approximated by successively solving using ZF-DF for symbols in \mathbf{x}_2 having \mathbf{x}_1 as parents. Let $\lfloor z \rfloor_{\mathcal{X}}$ and $\lfloor z \rfloor_{\mathcal{X}_b^{(s)}}$ denote slicing to the closest symbol to z in \mathcal{X} and $\mathcal{X}_b^{(s)} \triangleq \{x_n \in \mathcal{X} : x_{n,b} = s\}$, respectively. When the hypothesis is for a parent symbol bit ($1 \leq n \leq \nu$), ZF-DF on child symbols proceeds as follows:

$$k = \nu + 1, \dots, N: \hat{z}_k = w_k - \sum_{j=\nu+1}^{k-1} l_{kj} \hat{x}_j, \quad \hat{x}_k = \lfloor \hat{z}_k / l_{kk} \rfloor_{\mathcal{X}}.$$

Set $\hat{\mathbf{x}}_2 = [\hat{x}_{\nu+1}, \dots, \hat{x}_N]^T$ to be the child symbol vector estimate. On the other hand, for a child symbol bit hypothesis ($\nu + 1 \leq n \leq N$), ZF-DF on child symbols $k = \nu + 1, \dots, N$ proceeds as:

$$\begin{aligned} k < n: \quad \hat{z}_k &= w_k - \sum_{j=\nu+1}^{k-1} l_{kj} \hat{x}_j, & \hat{x}_k &= \lfloor \hat{z}_k / l_{kk} \rfloor_{\mathcal{X}}; \\ k = n: \quad \hat{z}_k &= w_k - \sum_{j=\nu+1}^{k-1} l_{kj} \hat{x}_j, & \hat{x}_{k;b}^{(s)} &\triangleq \lfloor \hat{z}_k / l_{kk} \rfloor_{\mathcal{X}_b^{(s)}}; \\ k > n: \quad \hat{z}_k &= w_k - \sum_{\substack{j=\nu+1: \\ j \neq n}}^{k-1} l_{kj} \hat{x}_j - l_{kn} \hat{x}_{n;b}^{(s)}, & \hat{x}_k &= \lfloor \hat{z}_k / l_{kk} \rfloor_{\mathcal{X}}. \end{aligned}$$

Let $\hat{\mathbf{x}}_{2n;b}^{(s)} = [x_{\nu+1}, \dots, \hat{x}_{n;b}^{(s)}, \dots, x_N]^T$ be the resulting child symbol vector estimate. Therefore, the inner max operations in (12)-(13) are approximated as

$$\max_{\mathbf{x}_2} \mu_2(\tilde{\mathbf{y}}_2 | \mathbf{x}_1, \mathbf{x}_2) \geq \sum_{k=\nu+1}^N \max_{x_k} e_2(\hat{z}_k, x_k) = \sum_{k=\nu+1}^N e_2(\hat{z}_k, \hat{x}_k) \triangleq \hat{\mu}_2(\tilde{\mathbf{y}}_2 | \mathbf{x}_1, \hat{\mathbf{x}}_2), \quad (14)$$

$$\begin{aligned} \max_{\substack{\mathbf{x}_2: \\ x_{n,b}=s}} \mu_2(\tilde{\mathbf{y}}_2 | \mathbf{x}_1, \mathbf{x}_2) &\geq \sum_{\substack{k=\nu+1: \\ k \neq n}}^N \max_{x_k} e_2(\hat{z}_k, x_k) + \max_{\substack{x_n: \\ x_{n,b}=s}} e_2(\hat{z}_n, x_n) \\ &= \sum_{\substack{k=\nu+1: \\ k \neq n}}^N e_2(\hat{z}_k, \hat{x}_k) + e_2(\hat{z}_n, \hat{x}_{n;b}^{(s)}) = \hat{\mu}_2(\tilde{\mathbf{y}}_2 | \mathbf{x}_1, \hat{\mathbf{x}}_{2n;b}^{(s)}), \end{aligned} \quad (15)$$

and (12)-(13) are approximated as

$$\max_{\substack{\mathbf{x}: x_{n,b}=s \\ 1 \leq n \leq \nu}} \mu(\tilde{\mathbf{y}} | \mathbf{x}) \geq \max_{\substack{\mathbf{x}_1: \\ x_{n,b}=s}} \{ \mu_1(\tilde{\mathbf{y}}_1 | \mathbf{x}_1) + \hat{\mu}_2(\tilde{\mathbf{y}}_2 | \mathbf{x}_1, \hat{\mathbf{x}}_2) \}, \quad (16)$$

$$\max_{\substack{\mathbf{x}: x_{n,b}=s \\ \nu+1 \leq n \leq N}} \mu(\tilde{\mathbf{y}} | \mathbf{x}) \geq \max_{\mathbf{x}_1} \{ \mu_1(\tilde{\mathbf{y}}_1 | \mathbf{x}_1) + \hat{\mu}_2(\tilde{\mathbf{y}}_2 | \mathbf{x}_1, \hat{\mathbf{x}}_{2n;b}^{(s)}) \}. \quad (17)$$

The above maxima are not optimal because of the ZF-DF operations on the child layers in (14)-(15). However, if the l_{kj} terms are 0 for $k = \nu + 2, \dots, N$ and $j = \nu + 1, \dots, k - 1$ in the z_k summation in (8), then $z_k = w_k$, $e(z_k, x_k) = \frac{-1}{N_0} |w_k - l_{kk} x_k|^2$, and the maximizations in (14)-(15) become exact in this case:

$$\begin{aligned} \max_{\mathbf{x}_2} \sum_{k=\nu+1}^N e_2(w_k, x_k) &= \sum_{k=\nu+1}^N \max_{x_k} e_2(w_k, x_k) = \sum_{k=\nu+1}^N e_2(w_k, \hat{x}_k), \\ \max_{\substack{\mathbf{x}_2: \\ x_{n,b}=s}} \sum_{k=\nu+1}^N e_2(w_k, x_k) &= \sum_{\substack{k=\nu+1: \\ k \neq n}}^N \max_{x_k} e_2(w_k, x_k) + \max_{\substack{x_n: \\ x_{n,b}=s}} e_2(w_n, x_n) = \sum_{\substack{k=\nu+1: \\ k \neq n}}^N e_2(w_k, \hat{x}_k) + e_2(w_n, \hat{x}_{n;b}^{(s)}). \end{aligned}$$

In addition, all intermediate complex products involving the zeroed entries l_{kj} are not needed.

B. Single-Tree and Multi-Tree Approaches

Soft-output detection is essentially a multi-point search problem for the ML point and all its counter-ML hypotheses. Tree-search algorithms used to generate bit LLRs for channels partitioned into parent and child symbol layers follow either a single-tree or a multi-tree approach to find these points. For single-tree, a pre-processing step chooses ν ordered layers as parents and $N - \nu$ ordered layers as children; one tree is used to solve for both parent and child bit LLRs. For multi-tree, N/ν trees are used to solve only for parent bit LLRs, such that a different combination of layers is chosen as parents for each tree.

Both approaches use enumeration over the parent layers, and marginalization over the child layers. Marginalization complicates LLR generation of child bits for single-tree because it has to be repeated for every child bit hypothesis and for every candidate parent symbol vector. Also, the quality of the LLRs under the single-tree approach is very sensitive to the choice of parent layers and overall ordering of layers. On the other hand, in the multi-tree approach the distinct layer orderings of each tree constitute an added diversity that can be leveraged to globally optimize the closest points locally searched by each tree and their metrics across all the trees.

III. CHANNEL PUNCTURING

Motivated by the observation from the last section to improve the efficiency and reduce the computational complexity of the detection process by nulling entries below the main diagonal of \mathbf{L} , we next investigate possible puncturing schemes that are applicable to integer LS problems.

Consider the lower-triangular matrix shown in Fig. 1. To null all entries below the diagonal and to the right of the ν th column of $\mathbf{L} = [l_{kj}]$ ($l_{kj} \leftarrow 0$ for $\nu+1 < k \leq N$ and $\nu < j < k$) for some ν , $1 \leq \nu \leq N-1$, we partition \mathbf{L} conformally as

$$\mathbf{L}_{N \times N} = \begin{bmatrix} \mathbf{P}_{\nu \times \nu} & \mathbf{0}_{\nu \times (N-\nu)} \\ \mathbf{R}_{(N-\nu) \times \nu} & \mathbf{S}_{(N-\nu) \times (N-\nu)} \end{bmatrix}, \quad (18)$$

where $\mathbf{P} \in \mathcal{C}^{\nu \times \nu}$ and $\mathbf{S} \in \mathcal{C}^{(N-\nu) \times (N-\nu)}$ are complex square lower-triangular matrices of sizes ν and $N-\nu$, respectively, having real diagonal elements, and $\mathbf{R} \in \mathcal{C}^{(N-\nu) \times \nu}$ is a complex rectangular

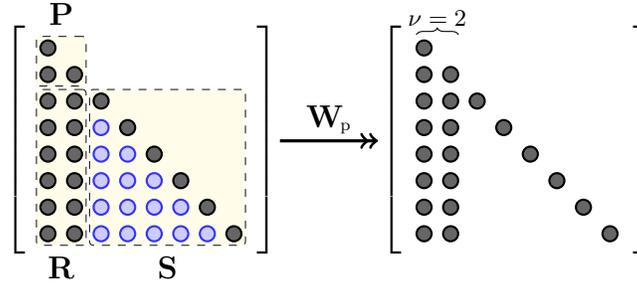


Fig. 1. Puncturing an 8×8 matrix \mathbf{L} into \mathbf{L}_p using \mathbf{W}_p for $\nu = 2$.

matrix. The target of puncturing is to diagonalize \mathbf{S} . Hence, without loss of generality, we focus on techniques to diagonalize \mathbf{S} that do not alter Euclidean distances of the form $\|\mathbf{y} - \mathbf{L}\mathbf{x}\|^2$. Henceforth, \mathbf{L} is assumed to be non-singular.

Using two-sided unitary transformations \mathbf{W}_p and \mathbf{Z} of size N , it is well-known that \mathbf{S} or all of \mathbf{L} can be reduced to diagonal form $\mathbf{D} = \mathbf{W}_p \mathbf{L} \mathbf{Z}$ via an SVD-like decomposition [21]. The left transformation \mathbf{W}_p must be unitary in order to preserve L_2 -norms and not alter noise statistics:

$$\|\mathbf{W}_p \mathbf{x}\|^2 = \mathbf{x}^\dagger \mathbf{W}_p^\dagger \mathbf{W}_p \mathbf{x} = \|\mathbf{x}\|^2 \Rightarrow \mathbf{W}_p^\dagger \mathbf{W}_p = \mathbf{I}, \quad (19)$$

$$\mathbb{E}[\mathbf{W}_p \mathbf{n} \mathbf{n}^\dagger \mathbf{W}_p^\dagger] = N_0 \mathbf{W}_p \mathbf{W}_p^\dagger = N_0 \mathbf{I} \Rightarrow \mathbf{W}_p \mathbf{W}_p^\dagger = \mathbf{I}. \quad (20)$$

The right transformation \mathbf{Z} must preserve the (Gaussian) integer nature of the unknown \mathbf{x} ; that is, if for some $\mathbf{y} \in \mathcal{C}^N$

$$\mathbf{z}^* = \underset{\mathbf{z} \in \mathcal{Z}^N}{\operatorname{argmin}} \|\mathbf{y} - \mathbf{L}\mathbf{z}\|^2, \quad \text{then} \quad \mathbf{Z}^{-1} \mathbf{z}^* = \underset{\mathbf{x} \in \mathcal{X}^N}{\operatorname{argmin}} \|\mathbf{y} - \mathbf{L}\mathbf{Z}\mathbf{x}\|^2 \in \mathcal{X}^N. \quad (21)$$

Hence \mathbf{Z} has to be *unimodular*, i.e., an integer matrix in $\mathcal{G}^{N \times N}$ with integer inverse having $|\det \mathbf{Z}| = 1$. Also, \mathbf{Z} must be lower-triangular in order to induce a parent-child tree structure using forward substitution, and hence cannot be unitary (\mathbf{Z}^\dagger is upper triangular, while \mathbf{Z}^{-1} is lower triangular; hence they cannot be equal). However, if \mathbf{Z} is not unitary, \mathbf{W}_p being unitary and applied from the left cannot alone null an element below the main diagonal of \mathbf{S} without creating a non-zero entry in its upper-triangular counterpart, hence altering the lower-triangular structure of \mathbf{S} . Therefore, both \mathbf{W}_p and \mathbf{Z} cannot be unitary, and (19)-(20) cannot be satisfied.

A. One-Sided Puncturing Transformations

The matrix \mathbf{L} in (18) can be punctured into $\mathbf{L}_p \in \mathcal{C}^{N \times N}$ using a left puncturing matrix $\mathbf{W}_p \in \mathcal{C}^{N \times N}$ only ($\mathbf{Z} = \mathbf{I}$) as follows:

$$\begin{bmatrix} \blacksquare & & & \\ \times & \blacksquare & & \\ \times & \mathbf{0} & \blacksquare & \\ \times & \times & \times & \blacksquare \\ \times & \times & \times & \times \end{bmatrix} = \begin{bmatrix} 1 & & & \\ \bullet & 1 & & \\ \omega & 1 & & \\ \bullet & \bullet & 1 & \\ \bullet & \bullet & \bullet & 1 \end{bmatrix} \begin{bmatrix} \blacksquare & & & \\ \times & \blacksquare & & \\ \times & \times & \times & \blacksquare \\ \times & \times & \times & \times \end{bmatrix} \begin{bmatrix} 1 & & & \\ \bullet & 1 & & \\ \zeta & 1 & & \\ \bullet & \bullet & 1 & \\ \bullet & \bullet & \bullet & 1 \end{bmatrix}$$

$\mathbf{L}_p \qquad \qquad \mathbf{W}_p \qquad \qquad \mathbf{L} \qquad \qquad \mathbf{Z}$

Fig. 2. Puncturing entry l_{32} of \mathbf{L} using two-sided transformations.

$$\mathbf{W}_p = \mathbf{D}_p \text{diag}(\mathbf{L}) \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}^{-1} \end{bmatrix}, \quad (22)$$

$$\mathbf{L}_p = \mathbf{W}_p \mathbf{L} = \mathbf{D}_p \text{diag}(\mathbf{L}) \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{P} & \mathbf{0} \\ \mathbf{R} & \mathbf{S} \end{bmatrix} = \mathbf{D}_p \text{diag}(\mathbf{L}) \begin{bmatrix} \mathbf{P} & \mathbf{0} \\ \mathbf{S}^{-1} \mathbf{R} & \mathbf{I} \end{bmatrix}, \quad (23)$$

where $\mathbf{D}_p \in \mathcal{R}^{N \times N}$ is a (normalizing) diagonal matrix. Since \mathbf{W}_p is not unitary, both conditions (19)-(20) are not met. We can relax (20) by choosing \mathbf{D}_p so that \mathbf{W}_p satisfies $\text{diag}(\mathbf{W}_p \mathbf{W}_p^\dagger) = \mathbf{I}_N$ instead. Hence

$$\mathbf{D}_p = \text{diag}(\mathbf{L})^{-1} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Omega} \end{bmatrix}, \quad \mathbf{W}_p = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Omega} \mathbf{S}^{-1} \end{bmatrix}, \quad \mathbf{L}_p = \begin{bmatrix} \mathbf{P} & \mathbf{0} \\ \mathbf{\Omega} \mathbf{S}^{-1} \mathbf{R} & \mathbf{\Omega} \end{bmatrix}, \quad (24)$$

where

$$\mathbf{\Omega} = \text{diag}(\mathbf{S}^{-1} \mathbf{S}^{-\dagger})^{-1/2}. \quad (25)$$

Note that \mathbf{W}_p is a non-singular lower-triangular matrix with ν ones and $N - \nu$ positive real numbers on the diagonal. Also, since $\mathbf{\Omega}$ normalizes \mathbf{S}^{-1} so that $\text{diag}(\mathbf{W}_p \mathbf{W}_p^\dagger) = \mathbf{I}_N$, then $\|\mathbf{W}_p\|_F = \sqrt{N}$ and the remaining $N - \nu$ eigenvalues λ of \mathbf{W}_p (i.e., diagonal elements of $\mathbf{\Omega} \mathbf{S}^{-1}$) satisfy $0 < \lambda \leq 1$. It follows that $\sqrt{N} \geq \sigma_{\max} \geq \lambda_{\max} = 1$ and $0 < \sigma_{\min} \leq \lambda_{\min} \leq 1$, where σ_{\max} (σ_{\min}) and λ_{\max} (λ_{\min}) are the maximum (minimum) singular values and eigenvalues of \mathbf{W}_p , respectively.

B. Two-Sided Puncturing Transformations

Note that the lower-triangular matrix \mathbf{S}^{-1} in the left non-unitary transformation \mathbf{W}_p in (22) is equivalent to a Gaussian elimination matrix. Using an integer Gauss *reduction* matrix \mathbf{Z} as a right transformation can help approximate (19) better by first reducing the lower-triangular entries of \mathbf{L} in \mathbf{S} using integer multiples of the diagonal elements and then completely eliminating the remainder using \mathbf{W}_p from the left. The reduction step by \mathbf{Z} from the right to reduce l_{kj} by an integer multiple of l_{kk} into \tilde{l}_{kj} , followed by an elimination step by \mathbf{W}_p from the left to null \tilde{l}_{kj} using l_{kk} are expressed as (see Fig. 2)

$$\text{reduction } \mathbf{Z}_{kj} : \zeta_{kj} = \lfloor \frac{l_{kj}}{l_{kk}} \rfloor, \quad \tilde{l}_{kj} = l_{kj} - \zeta_{kj} l_{kk}, \quad (26)$$

$$\text{elimination } \mathbf{W}_{pkj} : \omega_{kj} = \frac{\tilde{l}_{kj}}{l_{kk}}, \quad l_{kj} = \tilde{l}_{kj} - \omega_{kj} l_{kk}, \quad (27)$$

for $k = \nu + 2, \dots, N$ and $j = \nu + 1, \dots, k - 1$, where $[z] = [\Re\{z\}] + i[\Im\{z\}]$ and $[a] = [a + 1/2]$ for $a \in \mathcal{R}$. In particular, since $|a - [\frac{a}{b} + \frac{1}{2}]b| \leq \frac{|b|}{2}$ for $a, b \in \mathcal{R}$, then (26) results in

$$\left| \frac{\tilde{l}_{kj}}{l_{kk}} \right| = \left| (l_{kj} - [\frac{l_{kj}}{l_{kk}}]l_{kk}) / l_{kk} \right| \leq \left| \frac{1}{2} + i\frac{1}{2} \right| = \frac{1}{\sqrt{2}}. \quad (28)$$

In matrix form, operations (26)-(27) become

$$\mathbf{Z}_{kj} = \mathbf{I}_N - \zeta_{kj} \mathbf{e}_k \mathbf{e}_j^T \in \mathcal{G}^{N \times N}, \quad k \neq j, \quad \zeta_{kj} \in \mathcal{G}, \quad (29)$$

$$\mathbf{W}_{pkj} = \mathbf{I}_N - \omega_{kj} \mathbf{e}_k \mathbf{e}_j^T \in \mathcal{C}^{N \times N}, \quad k \neq j, \quad \omega_{kj} \in \mathcal{C}. \quad (30)$$

Note that $\mathbf{Z}_{kj}^{-1} = \mathbf{I}_N + \zeta_{kj} \mathbf{e}_k \mathbf{e}_j^T \in \mathcal{G}^{N \times N}$. The matrices \mathbf{Z} and \mathbf{W}_p are formed from the products of the $(N - \nu - 1)(N - \nu)/2$ matrices in (29) and (30), respectively. \mathbf{W}_p is then normalized using a diagonal matrix \mathbf{D}_p to satisfy $\text{diag}(\mathbf{W}_p \mathbf{W}_p^\dagger) = \mathbf{I}_N$:

$$\mathbf{Z} = \prod_{k=\nu+2}^N \prod_{j=\nu+1}^{k-1} \mathbf{Z}_{kj}, \quad \mathbf{W}_p = \mathbf{D}_p \prod_{k=\nu+2}^N \prod_{j=\nu+1}^{k-1} \mathbf{W}_{pkj}. \quad (31)$$

Lemma 1 ([22]): If $\mathbf{T} = [t_{kj}] \in \mathcal{C}^{N \times N}$ is a nonsingular lower-triangular matrix, then

$$\|\mathbf{T}^{-1}\|_{2,F} \leq \frac{1}{(\rho+2)\delta} \sqrt{(\rho+1)^{2N} + 2N(\rho+2) - 1}, \quad (32)$$

where $\rho = \max_{k < j} |t_{kj}| / |t_{kk}|$ and $\delta = \min_k |t_{kk}|$.

Proof: See [22]. ■

We use Lemma 1 to show that the norm of \mathbf{S}^{-1} (and hence \mathbf{W}_p in (24)) tends to decrease by applying \mathbf{Z} . Let $\check{\mathbf{Z}}$ be the principal submatrix obtained by deleting the first ν rows and columns of \mathbf{Z} , and let $\check{\mathbf{S}} = \check{\mathbf{S}}\check{\mathbf{Z}}$. The reduction step in (26) ensures that the magnitudes of the lower-diagonal elements of $\check{\mathbf{S}}$ are $\leq \frac{l_{kk}}{\sqrt{2}}$, while not altering the diagonal elements. The quantity $\|\check{\mathbf{S}}^{-1} - \text{diag}(\check{\mathbf{S}}^{-1})\|_F$ measures the ‘weight’ of the lower-triangular portion of $\check{\mathbf{S}}^{-1}$. Applying (32) for $\mathbf{T} = \check{\mathbf{S}}$, we obtain

$$\|\check{\mathbf{S}}^{-1} - \text{diag}(\check{\mathbf{S}}^{-1})\|_F^2 \leq \frac{(\rho+1)^{2(N-\nu)} - \rho(\rho+2)(N-\nu) - 1}{(\rho+2)^2 \delta^2}, \quad (33)$$

with $\rho = 1/\sqrt{2}$ and $\delta = \min_{k > \nu} l_{kk}$. As ρ decreases, this upper bound decreases, and hence $\check{\mathbf{S}}^{-1}$ becomes more diagonal. Therefore, with $\check{\mathbf{\Omega}} = \text{diag}(\check{\mathbf{S}}^{-1} \check{\mathbf{S}}^{-\dagger})^{-1/2}$, $\check{\mathbf{\Omega}} \check{\mathbf{S}}^{-1}$ becomes closer to the identity, and when used in lieu of $\mathbf{\Omega} \mathbf{S}^{-1}$ in (24) makes \mathbf{W}_p closer to \mathbf{I}_N .

To reduce ρ below $1/\sqrt{2}$, the reduction step in (26) can be changed by scaling the ratio l_{kj}/l_{kk} by a power-of-2 so that

$$\zeta_{kj} = 2^{-c} \left[2^c \frac{l_{kj}}{l_{kk}} \right], \quad (34)$$

for some integer $c \geq 0$. In this case, (28) becomes

$$\left| \frac{\tilde{l}_{kj}}{l_{kk}} \right| = \left| (l_{kj} - [\frac{l_{kj}}{l_{kk}/2^c}] \frac{l_{kk}}{2^c}) / l_{kk} \right| \leq \left| \frac{1}{2^{c+1}} + \frac{i}{2^{c+1}} \right| = \frac{1}{2^c \sqrt{2}}, \quad (35)$$

which gives $\rho = 1/2^{c+1/2}$. Since $2^c \zeta_{kj}$ is an integer, it follows that $2^c \mathbf{Z}_{kj} \in \mathcal{G}^{N \times N}$ and the integer condition (21) still holds.

Note that (26) is similar to the first lattice reduction condition of [23]–[25]. However, the Lovász condition [23]

$$l_{kk}^2 + |l_{k+1,k}|^2 \geq \gamma \cdot l_{k+1,k+1}^2, \quad \frac{1}{4} < \gamma < 1, \quad (36)$$

cannot be enforced for $k = \nu + 1, \dots, N - 1$ because it requires permuting the columns of \mathbf{L} , which destroys the lower-triangular structure of \mathbf{Z} .

IV. WLD MIMO DETECTION

In this section, we develop the detection model of the WLD detector and characterize its AIR using single-sided puncturing. The analysis for two-sided puncturing is similar.

Starting with $\|\mathbf{Q}^\dagger \mathbf{y} - \mathbf{L}\mathbf{x}\|^2$ and applying \mathbf{W}_p in (24), the equivalent metric to (3) computed by the WLD detector is

$$-\frac{1}{N_0} \|\mathbf{Q}^\dagger \mathbf{y} - \mathbf{L}\mathbf{x}\|^2 \xrightarrow{\mathbf{W}_p} \mu_p(\mathbf{y}|\mathbf{x}) = -\frac{1}{N_0} \|\mathbf{W}_p(\mathbf{Q}^\dagger \mathbf{y} - \mathbf{L}\mathbf{x})\|^2.$$

By expanding $\mu_p(\mathbf{y}|\mathbf{x})$ and dropping the irrelevant term $-\frac{1}{N_0} \|\mathbf{W}_p \mathbf{Q}^\dagger \mathbf{y}\|^2$, we obtain

$$\mu_p(\mathbf{y}|\mathbf{x}) = -\frac{1}{N_0} \|\mathbf{y}_p - \mathbf{L}_p \mathbf{x}\|^2 \cong 2\Re\{\mathbf{y}^\dagger \mathbf{F}_p \mathbf{x}\} - \mathbf{x}^\dagger \mathbf{G}_p \mathbf{x}, \quad (37)$$

where $\mathbf{y}_p = \mathbf{W}_p \mathbf{Q}^\dagger \mathbf{y}$, $\mathbf{L}_p = \mathbf{W}_p \mathbf{L}$,

$$\mathbf{F}_p = \frac{1}{N_0} \mathbf{Q} \mathbf{W}_p^\dagger \mathbf{L}_p, \quad \text{and} \quad \mathbf{G}_p = \frac{1}{N_0} \mathbf{L}_p^\dagger \mathbf{L}_p = \mathbf{H}^\dagger \mathbf{F}_p. \quad (38)$$

The corresponding detection model becomes

$$p_p(\mathbf{y}|\mathbf{x}) = \exp(2\Re\{\mathbf{y}^\dagger \mathbf{F}_p \mathbf{x}\} - \mathbf{x}^\dagger \mathbf{G}_p \mathbf{x}), \quad (39)$$

instead of the true conditional probability in (2). Based on (39), the achievable information rate of the WLD detector is lower-bounded by [26]

$$I_{\text{LB}}^{\text{WLD}} = \mathbb{E}_{\mathbf{Y}, \mathbf{X}}[\ln(p_p(\mathbf{y}|\mathbf{x}))] - \mathbb{E}_{\mathbf{Y}}[\ln(p_p(\mathbf{y}))], \quad (40)$$

where the expectations are taken over the true channel statistics, and $p_p(\mathbf{y}) = \int p_p(\mathbf{y}|\mathbf{x})p(\mathbf{x}) \, d\mathbf{x}$ with $p(\mathbf{x})$ being the prior distribution of \mathbf{x} .

Theorem 1: Assuming Gaussian inputs $\mathbf{x} \sim \mathcal{CN}(\mathbf{0}, E_s \mathbf{I}_N)$, and let $\beta = \frac{E_s}{N_0}$ be the SNR, the lower-bound on the AIR in (40) attained by the WLD detector is given by

$$I_{\text{LB}}^{\text{WLD}} = \ln \det(\mathbf{I} + \beta \mathbf{L}_p^\dagger \mathbf{L}_p) - \text{Tr}((\mathbf{I} - \mathbf{W}_p \mathbf{W}_p^\dagger)(\mathbf{I} + \beta \mathbf{L}_p \mathbf{L}_p^\dagger)^{-1}). \quad (41)$$

Proof: Following the approach in [17], we first compute the probability $p_p(\mathbf{y}) = \int p_p(\mathbf{y}|\mathbf{x})p(\mathbf{x}) \, d\mathbf{x}$ for $p_p(\mathbf{y}|\mathbf{x})$ in (39) and $p(\mathbf{x}) = \frac{1}{\pi^N E_s^N} \exp(-\frac{\|\mathbf{x}\|^2}{E_s})$. We then compute the expectations in (40) over the true channel statistics as

$$\begin{aligned} \mathbb{E}_{\mathbf{Y}, \mathbf{X}}[\ln(p_p(\mathbf{y}|\mathbf{x}))] &= 2E_s \Re\{\text{Tr}(\mathbf{F}_p^\dagger \mathbf{H})\} - E_s \text{Tr}(\mathbf{G}_p) = E_s \text{Tr}(\mathbf{G}_p), \\ -\mathbb{E}_{\mathbf{Y}}[\ln(p_p(\mathbf{y}))] &= N \ln E_s + \ln \det(\mathbf{G}_p + \frac{1}{E_s} \mathbf{I}) - \text{Tr}(\mathbf{F}_p^\dagger [E_s \mathbf{H} \mathbf{H}^\dagger + N_0 \mathbf{I}] \mathbf{F}_p [\mathbf{G}_p + \frac{1}{E_s} \mathbf{I}]^{-1}). \end{aligned}$$

Substituting for $\mathbf{F}_p = \frac{1}{N_0} \mathbf{Q} \mathbf{W}_p^\dagger \mathbf{L}_p$ and $\mathbf{G}_p = \frac{1}{N_0} \mathbf{L}_p^\dagger \mathbf{L}_p$, and applying the matrix inversion lemma [27], followed by some standard simplification steps, the result follows. \blacksquare

For $\nu = 1$, the sorted singular values $\{\sigma_k\}_{k=1}^N$ of \mathbf{L}_p satisfy an interlacing property with respect to the diagonal elements of $\mathbf{\Omega}$ in (24). Let $\omega_1 < \omega_2 < \dots < \omega_{N-\nu}$ be the sorted diagonal elements of $\mathbf{\Omega}$, and let \mathbf{v} be the first column of \mathbf{L}_p , then [28]

$$0 < \sigma_1 < \omega_1 < \dots < \sigma_{N-1} < \omega_{N-1} < \sigma_N < \omega_{N-1} + \|\mathbf{v}\|. \quad (42)$$

Property (42) can be used to bound $I_{\text{LB}}^{\text{WLD}}$ in Theorem 1 for $\nu = 1$ since $\ln \det(\mathbf{I} + \beta \mathbf{L}_p^\dagger \mathbf{L}_p) = \sum_{k=1}^N \ln(1 + \beta \sigma_k^2)$, $\text{Tr}((\mathbf{I} + \beta \mathbf{L}_p \mathbf{L}_p^\dagger)^{-1}) = \sum_{k=1}^N 1/(1 + \beta \sigma_k^2)$, and $\text{Tr}(\mathbf{W}_p \mathbf{W}_p^\dagger) = N$. The details are omitted due to lack of space.

Note that for $\nu = N - 1$, we have $\mathbf{W}_p = \mathbf{I}$ and $\mathbf{L}_p = \mathbf{L}$, and hence $I_{\text{LB}}^{\text{WLD}} = \ln \det(\mathbf{I} + \beta \mathbf{L}^\dagger \mathbf{L})$, which is the capacity of the channel. In fact, as ν increases from 1, the metrics computed by the WLD detector approach the hard-decision ML metrics as shown by the following lemma.

Lemma 2: If $\mathbf{x}_{\text{ML}} = \arg \min_{\mathbf{x} \in \mathcal{X}^N} \|\tilde{\mathbf{y}} - \mathbf{L}\mathbf{x}\|$ and $\mathbf{x}_{\text{WLD}} = \arg \min_{\mathbf{x} \in \mathcal{X}^N} \|\mathbf{W}_p(\tilde{\mathbf{y}} - \mathbf{L}\mathbf{x})\|$ where $\mathbf{H} = \mathbf{Q}\mathbf{L}$ and $\tilde{\mathbf{y}} = \mathbf{Q}^\dagger \mathbf{y}$, then

$$\|\tilde{\mathbf{y}} - \mathbf{L}\mathbf{x}_{\text{ML}}\| \leq \|\tilde{\mathbf{y}} - \mathbf{L}\mathbf{x}_{\text{WLD}}\| \leq \kappa(\mathbf{W}_p) \|\tilde{\mathbf{y}} - \mathbf{L}\mathbf{x}_{\text{ML}}\|, \quad (43)$$

$$\|\mathbf{W}_p(\tilde{\mathbf{y}} - \mathbf{L}\mathbf{x}_{\text{WLD}})\| \leq \sigma_{\max}(\mathbf{W}_p) \|\tilde{\mathbf{y}} - \mathbf{L}\mathbf{x}_{\text{ML}}\|, \quad (44)$$

where $\kappa(\mathbf{W}_p) = \sigma_{\max}(\mathbf{W}_p)/\sigma_{\min}(\mathbf{W}_p)$ is the condition number of \mathbf{W}_p , and $\sigma_{\max}(\mathbf{W}_p), \sigma_{\min}(\mathbf{W}_p)$ are the largest and smallest singular values of \mathbf{W}_p , respectively.

Proof: The first inequality in (43) follows from the definition of the ML solution. For the second, we have

$$\begin{aligned} \|\tilde{\mathbf{y}} - \mathbf{L}\mathbf{x}_{\text{WLD}}\| &= \|\mathbf{W}_p^{-1} \mathbf{W}_p(\tilde{\mathbf{y}} - \mathbf{L}\mathbf{x}_{\text{WLD}})\| \leq \sigma_{\max}(\mathbf{W}_p^{-1}) \|\mathbf{W}_p(\tilde{\mathbf{y}} - \mathbf{L}\mathbf{x}_{\text{WLD}})\| \\ &\leq \sigma_{\max}(\mathbf{W}_p^{-1}) \|\mathbf{W}_p(\tilde{\mathbf{y}} - \mathbf{L}\mathbf{x}_{\text{ML}})\| \\ &\leq \sigma_{\max}(\mathbf{W}_p^{-1}) \sigma_{\max}(\mathbf{W}_p) \|\tilde{\mathbf{y}} - \mathbf{L}\mathbf{x}_{\text{ML}}\|, \end{aligned} \quad (45)$$

from which (43) follows. Note that both (44) and (45) follow because $\|\mathbf{W}_p(\tilde{\mathbf{y}} - \mathbf{L}\mathbf{x}_{\text{WLD}})\| \leq \|\mathbf{W}_p(\tilde{\mathbf{y}} - \mathbf{L}\mathbf{x})\|$ for any \mathbf{x} . \blacksquare

Note that the layer orders within the ν parent layers and within the $N - \nu$ child layers are irrelevant. What matters is which layers are selected to form the parent set. This is formalized using the following lemma.

Lemma 3: Let \mathbf{J}_1 and \mathbf{J}_2 be permutation matrices of sizes ν and $N - \nu$, respectively. If the columns of \mathbf{H} are permuted by $\mathbf{J} = \begin{bmatrix} \mathbf{J}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{J}_2 \end{bmatrix}$, then the distance metric computed by the WLD

detector in (37) does not change, i.e.,

$$\|\mathbf{W}_p(\mathbf{Q}^\dagger \mathbf{y} - \mathbf{L}\mathbf{x})\|^2 = \|\tilde{\mathbf{W}}_p(\tilde{\mathbf{Q}}^\dagger \mathbf{y} - \tilde{\mathbf{L}}\mathbf{J}^{-1}\mathbf{x})\|^2, \quad (46)$$

where $\mathbf{H} = \mathbf{Q}\mathbf{L}$, \mathbf{W}_p ($\tilde{\mathbf{W}}_p$) is the puncturing matrix of \mathbf{L} ($\tilde{\mathbf{L}}$), and $\mathbf{H}\mathbf{J} = \tilde{\mathbf{Q}}\tilde{\mathbf{L}}$.

Proof: Let $\mathbf{x} = [\mathbf{x}_1; \mathbf{x}_2]$, $\mathbf{Q} = [\mathbf{Q}_1 \ \mathbf{Q}_2]$, $\tilde{\mathbf{Q}} = [\tilde{\mathbf{Q}}_1 \ \tilde{\mathbf{Q}}_2]$, $\mathbf{L} = \begin{bmatrix} \mathbf{P} \\ \mathbf{R} \ \mathbf{S} \end{bmatrix}$, and $\tilde{\mathbf{L}} = \begin{bmatrix} \tilde{\mathbf{P}} \\ \tilde{\mathbf{R}} \ \tilde{\mathbf{S}} \end{bmatrix}$ be partitioned corresponding to ν parent layers and $N-\nu$ child layers. Then $\mathbf{W}_p = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \Omega \mathbf{S}^{-1} \end{bmatrix}$ and $\tilde{\mathbf{W}}_p = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \tilde{\Omega} \tilde{\mathbf{S}}^{-1} \end{bmatrix}$, where $\Omega = \text{diag}(\mathbf{S}^{-1} \mathbf{S}^{-\dagger})^{-1/2}$ and $\tilde{\Omega} = \text{diag}(\tilde{\mathbf{S}}^{-1} \tilde{\mathbf{S}}^{-\dagger})^{-1/2}$. The partitions of $\tilde{\mathbf{L}}$ are related to those of \mathbf{L} since $\mathbf{Q}\mathbf{L}\mathbf{J} = \tilde{\mathbf{Q}}\tilde{\mathbf{L}}$. Furthermore, $\tilde{\Omega} = \mathbf{J}_2^\dagger \Omega \mathbf{J}_2$ since $\mathbf{Q}_2^\dagger \tilde{\mathbf{Q}}_2 \tilde{\mathbf{Q}}_2^\dagger \mathbf{Q}_2 = \mathbf{I}$. Substituting back in both squared-norms in (46), and performing simplifications, it follows that both sides are equal to $\|\mathbf{Q}_1^\dagger \mathbf{y} - \mathbf{P}\mathbf{x}_1\|^2 + \|\Omega(\mathbf{S}^{-1} \mathbf{Q}_2^\dagger \mathbf{y} - \mathbf{S}^{-1} \mathbf{R}\mathbf{x}_1 - \mathbf{x}_2)\|^2$. ■

Corollary 1: Let \mathbf{J} be any permutation matrix, $\mathbf{H}\mathbf{J} = \tilde{\mathbf{Q}}\tilde{\mathbf{L}}$, and $\tilde{\mathbf{W}}_p$ the puncturing matrix of $\tilde{\mathbf{L}}$. Then, the number of distinct solutions of $\mathbf{x}_{\text{WLD}} = \text{argmin}_{\mathbf{x}} \|\tilde{\mathbf{W}}_p(\tilde{\mathbf{Q}}^\dagger \mathbf{y} - \tilde{\mathbf{L}}\mathbf{J}^{-1}\mathbf{x})\|$ for all possible values of \mathbf{J} depends only on the number of parent layer combinations, and is at most $\binom{N}{\nu}$. ■

Finally, the bound $I_{\text{LB}}^{\text{WLD}}$ for Gaussian inputs can be used as a criterion for parent layer selection, but the complexity of possible combinations grows as $\binom{N}{\nu}$. Alternatively, a less sensitive approach to parent layer selection is to do multiple detection rounds, each time choosing ν new layers as parents and generating bit LLRs for these parent symbols only.

V. AUGMENTED WLD (AWLD) MIMO DETECTION

The lower bound on the AIR in (41) attained by the WLD is not optimal. Motivated by the result for the optimal receiver filter derived in [17] in the context of channel shortening for ISI channels, which involves an MMSE filter compensated by receiver tree processing, we introduce in this section an alternate form of puncturing using augmented channels. Instead of basing the detection metric in (3) on \mathbf{H} , we form the augmented vector $\mathbf{y}_a = \frac{1}{\sqrt{N_0}}[\mathbf{y}; \mathbf{0}_{N \times 1}]$ and matrix

$$\mathbf{H}_a = \begin{bmatrix} \frac{1}{\sqrt{N_0}} \mathbf{H}_{M \times N} \\ \frac{1}{\sqrt{E_s}} \mathbf{I}_N \end{bmatrix} \quad (\text{size } (M+N) \times N), \quad (47)$$

in a manner analogous to the square-root MMSE of [29], and reformulate $\mu(\mathbf{y}|\mathbf{x})$ in (3) using $\mathbf{H}_a, \mathbf{y}_a$ rather than \mathbf{H}, \mathbf{y} as

$$\begin{aligned} \mu(\mathbf{y}|\mathbf{x}) &= \frac{1}{E_s} \|\mathbf{x}\|^2 + 2\Re \left\{ \frac{1}{\sqrt{N_0}} [\mathbf{y}^\dagger \ \mathbf{0}] \begin{bmatrix} \frac{1}{\sqrt{N_0}} \mathbf{H} \\ \frac{1}{\sqrt{E_s}} \mathbf{I}_N \end{bmatrix} \mathbf{x} \right\} - \mathbf{x}^\dagger \left(\frac{1}{N_0} \mathbf{H}^\dagger \mathbf{H} + \frac{1}{E_s} \right) \mathbf{x} - \frac{1}{N_0} \|\mathbf{y}\|^2 \\ &= \frac{1}{E_s} \|\mathbf{x}\|^2 + 2\Re \{ \mathbf{y}_a^\dagger \mathbf{H}_a \mathbf{x} \} - \mathbf{x}^\dagger \mathbf{H}_a^\dagger \mathbf{H}_a \mathbf{x} - \|\mathbf{y}_a\|^2 \\ &= \frac{1}{E_s} \|\mathbf{x}\|^2 - \|\mathbf{y}_a - \mathbf{H}_a \mathbf{x}\|^2. \end{aligned} \quad (48)$$

We next expand the squared-distance in (48) in terms of the projection matrix $\mathbf{P}_{\mathbf{H}_a} = \mathbf{H}_a (\mathbf{H}_a^\dagger \mathbf{H}_a)^{-1} \mathbf{H}_a^\dagger$ onto the column space of \mathbf{H}_a and its orthogonal complement $\mathbf{P}_{\mathbf{H}_a}^\perp = \mathbf{I}_{M+N} - \mathbf{H}_a (\mathbf{H}_a^\dagger \mathbf{H}_a)^{-1} \mathbf{H}_a^\dagger$ as

$$\|\mathbf{y}_a - \mathbf{H}_a \mathbf{x}\|^2 = \|\mathbf{P}_{\mathbf{H}_a}(\mathbf{y}_a - \mathbf{H}_a \mathbf{x})\|^2 + \|\mathbf{P}_{\mathbf{H}_a}^\perp \mathbf{y}_a\|^2. \quad (49)$$

Let $\mathbf{Q}_a \mathbf{L}_a$ be the thin QL decomposition of \mathbf{H}_a partitioned as

$$\mathbf{H}_a = \begin{bmatrix} \frac{1}{\sqrt{N_0}} \mathbf{H} \\ \frac{1}{\sqrt{E_s}} \mathbf{I}_N \end{bmatrix} = \mathbf{Q}_a \mathbf{L}_a = \begin{bmatrix} \mathbf{Q}_{a1} \\ \mathbf{Q}_{a2} \end{bmatrix} \mathbf{L}_a = \begin{bmatrix} \mathbf{Q}_{a1} \mathbf{L}_a \\ \mathbf{Q}_{a2} \mathbf{L}_a \end{bmatrix}, \quad (50)$$

where \mathbf{Q}_a is an $(M+N) \times N$ matrix with orthonormal columns (i.e., $\mathbf{Q}_a^\dagger \mathbf{Q}_a = \mathbf{I}_N$ but not unitary since $\mathbf{Q}_a \mathbf{Q}_a^\dagger \neq \mathbf{I}_{M+N}$), \mathbf{L}_a is $N \times N$ lower-triangular, and \mathbf{Q}_{a1} , \mathbf{Q}_{a2} are respectively the upper $M \times N$ and lower $N \times N$ block matrices of \mathbf{Q}_a . Note that neither the rows nor the columns of \mathbf{Q}_{a1} and \mathbf{Q}_{a2} are orthonormal. From the partitions in (50), it follows that

$$\mathbf{H} = \sqrt{N_0} \mathbf{Q}_{a1} \mathbf{L}_a, \quad (51)$$

$$\frac{1}{\sqrt{E_s}} \mathbf{I}_N = \mathbf{Q}_{a2} \mathbf{L}_a = \mathbf{L}_a \mathbf{Q}_{a2}. \quad (52)$$

However, (51) is *not* the QL decomposition of \mathbf{H} . (52) implies that \mathbf{Q}_{a2} is a lower-triangular matrix proportional to the inverse of \mathbf{L}_a , i.e., $\mathbf{L}_a^{-1} = \sqrt{E_s} \mathbf{Q}_{a2}$. Then, from (50) we have

$$\frac{1}{N_0} \mathbf{H}^\dagger \mathbf{H} + \frac{1}{E_s} \mathbf{I}_N = \mathbf{H}_a^\dagger \mathbf{H}_a = \mathbf{L}_a^\dagger \mathbf{L}_a,$$

from which it follows that

$$\|\mathbf{y}_a - \mathbf{H}_a \mathbf{x}\|^2 = \|\mathbf{L}_a (\mathbf{M} \mathbf{y} - \mathbf{x})\|^2 + \|(\mathbf{I} - \mathbf{Q}_a \mathbf{Q}_a^\dagger) \mathbf{y}_a\|^2, \quad (53)$$

where \mathbf{M} is the standard $N \times M$ MMSE filter matrix,

$$\mathbf{M} = \mathbf{H}^\dagger [\mathbf{H} \mathbf{H}^\dagger + \alpha \mathbf{I}_M]^{-1} = [\mathbf{H}^\dagger \mathbf{H} + \alpha \mathbf{I}_N]^{-1} \mathbf{H}^\dagger \quad (54)$$

$$= \frac{1}{N_0} (\mathbf{H}_a^\dagger \mathbf{H}_a)^{-1} \mathbf{H}^\dagger = \frac{1}{N_0} (\mathbf{L}_a^\dagger \mathbf{L}_a)^{-1} \mathbf{H}^\dagger \quad (55)$$

$$= \sqrt{\beta} \mathbf{Q}_{a2} \mathbf{Q}_{a1}^\dagger, \quad (56)$$

with $\alpha = \frac{1}{\beta} = \frac{N_0}{E_s}$. Substituting (53) back in (48), we obtain

$$\mu(\mathbf{y}|\mathbf{x}) = \frac{1}{E_s} \|\mathbf{x}\|^2 - \|\mathbf{L}_a (\mathbf{M} \mathbf{y} - \mathbf{x})\|^2 - \|(\mathbf{I} - \mathbf{Q}_a \mathbf{Q}_a^\dagger) \mathbf{y}_a\|^2. \quad (57)$$

Note that in (57), the term $\|\mathbf{x}\|^2$ appears explicitly, while tree processing is solely based on \mathbf{L}_a in $\|\mathbf{L}_a (\mathbf{M} \mathbf{y} - \mathbf{x})\|^2$. We therefore puncture \mathbf{L}_a using an appropriate puncturing matrix \mathbf{W}_{ap} similar to puncturing \mathbf{L} in (23) using \mathbf{W}_p . For a given puncturing order ν , we conformally partition \mathbf{L}_a similar to (18) and obtain the partition blocks \mathbf{P}_a of size $\nu \times \nu$, \mathbf{R}_a of size $(N-\nu) \times \nu$, and \mathbf{S}_a of size $(N-\nu) \times (N-\nu)$. The resulting punctured augmented matrix, denoted as \mathbf{L}_{ap} , is given by

$$\mathbf{L}_{\text{ap}} = \mathbf{W}_{\text{ap}} \mathbf{L}_a = \mathbf{W}_{\text{ap}} \begin{bmatrix} \mathbf{P}_a & \mathbf{0} \\ \mathbf{R}_a & \mathbf{S}_a \end{bmatrix} = \begin{bmatrix} \mathbf{P}_a & \mathbf{0} \\ \boldsymbol{\Omega}_a \mathbf{S}_a^{-1} \mathbf{R}_a & \boldsymbol{\Omega}_a \end{bmatrix}, \quad (58)$$

where

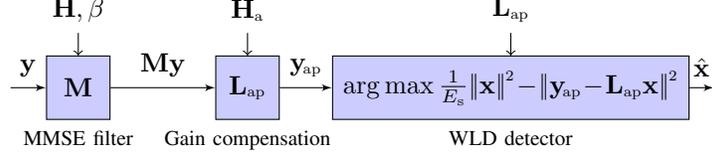


Fig. 3. Block diagram of the AWLD detector, where $\mathbf{y}_{\text{ap}} = \mathbf{L}_{\text{ap}}\mathbf{M}\mathbf{y}$.

$$\mathbf{W}_{\text{ap}} = \mathbf{D}_{\text{ap}}\text{diag}(\mathbf{L}_{\text{a}}) \begin{bmatrix} \mathbf{I}_{\nu} & \mathbf{0} \\ \mathbf{0} & \mathbf{S}_{\text{a}}^{-1} \end{bmatrix} = \begin{bmatrix} \mathbf{I}_{\nu} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Omega}_{\text{a}}\mathbf{S}_{\text{a}}^{-1} \end{bmatrix}, \quad (59)$$

$$\mathbf{D}_{\text{ap}} = \text{diag}(\mathbf{L}_{\text{a}})^{-1} \begin{bmatrix} \mathbf{I}_{\nu} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Omega}_{\text{a}} \end{bmatrix}, \quad (60)$$

$$\boldsymbol{\Omega}_{\text{a}} = \text{diag}(\mathbf{S}_{\text{a}}^{-1}\mathbf{S}_{\text{a}}^{-\dagger})^{-1/2}, \quad (61)$$

and \mathbf{D}_{ap} in (60) is chosen so that $\text{diag}(\mathbf{W}_{\text{ap}}\mathbf{W}_{\text{ap}}^{\dagger}) = \mathbf{I}_N$.

Next, applying \mathbf{W}_{ap} to filter $\mathbf{L}_{\text{a}}(\mathbf{M}\mathbf{y} - \mathbf{x})$ in (57) as

$$\|\mathbf{L}_{\text{a}}(\mathbf{M}\mathbf{y} - \mathbf{x})\|^2 \xrightarrow{\mathbf{W}_{\text{ap}}} \|\mathbf{W}_{\text{ap}}(\mathbf{L}_{\text{a}}\mathbf{M}\mathbf{y} - \mathbf{L}_{\text{a}}\mathbf{x})\|^2, \quad (62)$$

and dropping the irrelevant term $\|(\mathbf{I} - \mathbf{Q}_{\text{a}}\mathbf{Q}_{\text{a}}^{\dagger})\mathbf{y}_{\text{a}}\|^2$ in (57), the metric computed by the *augmented* WLD (AWLD) detector corresponding to (57) takes the form

$$\mu_{\text{ap}}(\mathbf{y}|\mathbf{x}) = \frac{1}{E_s}\|\mathbf{x}\|^2 - \|\mathbf{W}_{\text{ap}}\mathbf{L}_{\text{a}}(\mathbf{M}\mathbf{y} - \mathbf{x})\|^2 = \frac{1}{E_s}\|\mathbf{x}\|^2 - \|\mathbf{y}_{\text{ap}} - \mathbf{L}_{\text{ap}}\mathbf{x}\|^2 \quad (63)$$

$$\cong 2\Re\{\mathbf{y}^{\dagger}\mathbf{F}_{\text{ap}}\mathbf{x}\} - \mathbf{x}^{\dagger}\mathbf{G}_{\text{ap}}\mathbf{x} + \frac{1}{E_s}\mathbf{x}^{\dagger}\mathbf{x}, \quad (64)$$

where $\mathbf{y}_{\text{ap}} = \mathbf{W}_{\text{ap}}\mathbf{L}_{\text{a}}\mathbf{M}\mathbf{y} = \mathbf{L}_{\text{ap}}\mathbf{M}\mathbf{y}$,

$$\mathbf{F}_{\text{ap}} = \mathbf{M}^{\dagger}\mathbf{G}_{\text{ap}}, \quad \text{and} \quad \mathbf{G}_{\text{ap}} = \mathbf{L}_{\text{ap}}^{\dagger}\mathbf{L}_{\text{ap}}. \quad (65)$$

The corresponding AWLD detection model (Fig. 3) becomes

$$p_{\text{ap}}(\mathbf{y}|\mathbf{x}) = \exp(2\Re\{\mathbf{y}^{\dagger}\mathbf{F}_{\text{ap}}\mathbf{x}\} - \mathbf{x}^{\dagger}\mathbf{G}_{\text{ap}}\mathbf{x} + \frac{1}{E_s}\mathbf{x}^{\dagger}\mathbf{x}). \quad (66)$$

Theorem 2: Under the same assumptions as Theorem 1, the AIR of the augmented WLD detector based on (66), with $\mathbf{G}_{\text{ap}}, \mathbf{F}_{\text{ap}}$ as given in (65), is lower-bounded by

$$I_{\text{LB}}^{\text{AWLD}} = N \ln E_s + \ln \det(\mathbf{L}_{\text{ap}}^{\dagger}\mathbf{L}_{\text{ap}}). \quad (67)$$

Proof: The lower bound on the AIR of the AWLD detector based on (66) is defined as

$$I_{\text{LB}}^{\text{AWLD}} = \mathbb{E}_{\mathbf{Y}, \mathbf{X}}[\ln(p_{\text{ap}}(\mathbf{y}|\mathbf{x}))] - \mathbb{E}_{\mathbf{Y}}[\ln(p_{\text{ap}}(\mathbf{y}))], \quad (68)$$

where $p_{\text{ap}}(\mathbf{y})$ is given by

$$p_{\text{ap}}(\mathbf{y}) = \int_{\mathbf{x} \in \mathcal{C}^N} p_{\text{ap}}(\mathbf{y}|\mathbf{x})p(\mathbf{x}) \, d\mathbf{x}, \quad (69)$$

assuming $\mathbf{x} \sim \mathcal{CN}(\mathbf{0}, E_s\mathbf{I}_N)$. The main difference compared to the proof of Theorem 1 is the effect of the term $\frac{1}{E_s}\mathbf{x}^{\dagger}\mathbf{x}$ in (66) when evaluating (69) under Gaussian densities, which annihilates the effect of the prior density $p(\mathbf{x})$ to give

$$p_{\text{ap}}(\mathbf{y}) = \frac{1}{\pi^N E_s^N} \int \exp(2\Re\{\mathbf{y}^{\dagger}\mathbf{F}_{\text{ap}}\mathbf{x}\} - \mathbf{x}^{\dagger}\mathbf{G}_{\text{ap}}\mathbf{x}) \, d\mathbf{x}. \quad (70)$$

With standard manipulations, the expectations in (68) become

$$\begin{aligned} E_{\mathbf{Y}, \mathbf{X}}[\ln(p_{\text{ap}}(\mathbf{y}|\mathbf{x}))] &= N - E_s \text{Tr}(\mathbf{G}_{\text{ap}}) + 2E_s \Re\{\text{Tr}(\mathbf{F}_{\text{ap}}^\dagger \mathbf{H})\}, \\ -E_{\mathbf{Y}}[\ln(p_{\text{ap}}(\mathbf{y}))] &= N \ln E_s + \ln \det(\mathbf{G}_{\text{ap}}) - \text{Tr}(\mathbf{F}_{\text{ap}}^\dagger [E_s \mathbf{H} \mathbf{H}^\dagger + N_0 \mathbf{I}] \mathbf{F}_{\text{ap}} \mathbf{G}_{\text{ap}}^{-1}). \end{aligned}$$

Substituting (65) for \mathbf{G}_{ap} and \mathbf{F}_{ap} , and applying (54) for \mathbf{M} , then $\mathbf{F}_{\text{ap}}^\dagger [E_s \mathbf{H} \mathbf{H}^\dagger + N_0 \mathbf{I}] \mathbf{F}_{\text{ap}} \mathbf{G}_{\text{ap}}^{-1} = E_s \mathbf{F}_{\text{ap}}^\dagger \mathbf{H} = E_s \mathbf{G}_{\text{ap}} \mathbf{M} \mathbf{H}$. Also, it is easy to show that

$$\mathbf{M} \mathbf{H} = [\mathbf{H}^\dagger \mathbf{H} + \alpha \mathbf{I}_N]^{-1} \mathbf{H}^\dagger \mathbf{H} = \mathbf{I} - \alpha [\alpha \mathbf{I}_N + \mathbf{H}^\dagger \mathbf{H}]^{-1}, \quad (71)$$

which implies that $\mathbf{M} \mathbf{H}$ is Hermitian. Hence, $\text{Tr}(\mathbf{G}_{\text{ap}} \mathbf{M} \mathbf{H}) = \text{Tr}(\mathbf{G}_{\text{ap}} [\mathbf{I} - \alpha (\alpha \mathbf{I} + \mathbf{H}^\dagger \mathbf{H})^{-1}])$ is real.

Adding the two expectations above results in

$$\begin{aligned} I_{\text{LB}}^{\text{AWLD}} &= N \ln E_s + \ln \det(\mathbf{G}_{\text{ap}}) - \text{Tr}(\mathbf{G}_{\text{ap}} [\frac{1}{E_s} \mathbf{I} + \frac{1}{N_0} \mathbf{H}^\dagger \mathbf{H}]^{-1}) + N \\ &= N \ln E_s + \ln \det(\mathbf{G}_{\text{ap}}) - \text{Tr}(\mathbf{G}_{\text{ap}} (\mathbf{L}_a^\dagger \mathbf{L}_a)^{-1}) + N \\ &= N \ln E_s + \ln \det(\mathbf{G}_{\text{ap}}) - \text{Tr}(\mathbf{W}_{\text{ap}}^\dagger \mathbf{W}_{\text{ap}}) + N, \end{aligned}$$

from which (67) follows since $\text{Tr}(\mathbf{W}_{\text{ap}}^\dagger \mathbf{W}_{\text{ap}}) = N$. ■

With the punctured structure of the channel matrix \mathbf{L}_{ap} as given in (58), the gap of $I_{\text{LB}}^{\text{AWLD}}$ to AWGN capacity can be determined using the following corollary.

Corollary 2: The gap of the AIR of the AWLD detector to AWGN capacity is

$$C^{\text{AWGN}} - I_{\text{LB}}^{\text{AWLD}} = \sum_{k=1}^{N-\nu} \ln (s_{\text{akk}}^2 \|[\mathbf{S}_a^{-1}]_{\bar{k}}\|^2 + 1), \quad (72)$$

where s_{akk} is the k th diagonal element of \mathbf{S}_a in (58), and $[\mathbf{S}_a^{-1}]_{\bar{k}}$ is the row vector consisting of the first $k-1$ elements in row k of \mathbf{S}_a^{-1} in (59), excluding the diagonal element.

Proof: Applying (58)-(61) in (67), the $\ln \det$ term splits and the $C^{\text{AWGN}} = \ln \det(\frac{E_s}{N_0} \mathbf{H}^\dagger \mathbf{H} + \mathbf{I}_N)$ term emerges. ■

Similar to the WLD case, the gap to capacity vanishes for $\nu = N - 1$. Also, the metrics computed by the AWLD detector approach the hard-decision ML metrics as ν increases from 1.

Lemma 4: Let $\mu(\mathbf{x}) = \frac{1}{E_s} \|\mathbf{x}\|^2 - \|\mathbf{L}_a(\mathbf{M}\mathbf{y} - \mathbf{x})\|^2$, $\mathbf{x}_{\text{ML}} = \arg \max_{\mathbf{x}} \mu(\mathbf{x})$, $\omega(\mathbf{x}) = \frac{1}{E_s} \|\mathbf{x}\|^2 - \|\mathbf{W}_{\text{ap}} \mathbf{L}_a(\mathbf{M}\mathbf{y} - \mathbf{x})\|^2$, and $\mathbf{x}_{\text{AWLD}} = \arg \max_{\mathbf{x}} \omega(\mathbf{x})$. Then,

$$\kappa^2 \mu(\mathbf{x}_{\text{ML}}) - \eta(\kappa^2 - 1) \leq \mu(\mathbf{x}_{\text{AWLD}}) \leq \mu(\mathbf{x}_{\text{ML}}) \quad (73)$$

$$\omega(\mathbf{x}_{\text{AWLD}}) \geq \eta(1 - \sigma_{\max}(\mathbf{W}_{\text{ap}})) + \sigma_{\max}(\mathbf{W}_{\text{ap}}) \mu(\mathbf{x}_{\text{ML}}), \quad (74)$$

where $\kappa = \sigma_{\max}(\mathbf{W}_{\text{ap}}) / \sigma_{\min}(\mathbf{W}_{\text{ap}})$, and $\sigma_{\max}(\mathbf{W}_{\text{ap}})$, $\sigma_{\min}(\mathbf{W}_{\text{ap}})$ are the largest and smallest singular values of \mathbf{W}_{ap} , respectively, $\eta = \frac{N E_{\text{max}}}{E_s}$, and $E_{\text{max}} = \max_{x \in \mathcal{X}} |x|^2$.

Proof: The proof is similar to Lemma 2, and uses the fact that $\omega(\mathbf{x}_{\text{AWLD}}) \geq \omega(\mathbf{x}_{\text{ML}})$. As $\kappa \rightarrow 1$, $\mu(\mathbf{x}_{\text{AWLD}}) \rightarrow \mu(\mathbf{x}_{\text{ML}})$. ■

As illustrated in Fig. 3, the AWLD detector includes the WLD detector as a sub-block; the processing steps of MMSE filtering and gain are done prior to WLD detection. Also, it is worth

noting that computing the augmented channel requires simple processing comparable to QL decomposition. In particular, matrix inversion is not needed to compute \mathbf{M} in (55) because the inverse of \mathbf{L}_a is available from (52). In addition, using the modular approach of [30], an efficient hardware architecture for an AWLD MIMO detector can be constructed from optimized 2×2 MIMO detector cores. Extensions to include soft-input information, imperfect channel estimation effects, and correlated channels are directly applicable based on [18]. Finally, a scheme similar to [31] can be used for analyzing the diversity gain.

VI. AIR-OPTIMALITY OF THE AWLD DETECTOR

Instead of working with Euclidean-distance based metrics as in (3), the authors in [17] propose replacing N_0 , \mathbf{H} , $\mathbf{H}^\dagger\mathbf{H}$ in (4) with mismatched parameters N_r , \mathbf{F}_r , \mathbf{G}_r that are subject to AIR optimization. Hence, instead of the true metric in (5) and true probability in (2), the mismatched model of [17] is

$$\mu_r(\mathbf{y}|\mathbf{x}) = 2\Re\{\mathbf{y}^\dagger\mathbf{F}_r\mathbf{x}\} - \mathbf{x}^\dagger\mathbf{G}_r\mathbf{x}, \quad (75)$$

$$p_r(\mathbf{y}|\mathbf{x}) = \exp(2\Re\{\mathbf{y}^\dagger\mathbf{F}_r\mathbf{x}\} - \mathbf{x}^\dagger\mathbf{G}_r\mathbf{x}), \quad (76)$$

where N_r is absorbed into \mathbf{F}_r and \mathbf{G}_r . It is shown in [17] that detectors limited to the Euclidean-based model in (5), where \mathbf{G}_r admits a Cholesky factorization proportional to $\mathbf{H}^\dagger\mathbf{H}$, are not optimal from a mutual information perspective because the resulting optimal matrix \mathbf{G}_r to use in (76) may not be positive semidefinite, and hence no such factorization exists. The optimal \mathbf{F}_r and \mathbf{G}_r are derived by maximizing the lower bound on the AIR in two steps, assuming \mathbf{G}_r is Hermitian (and hence has real eigenvalues). First, an explicit expression for \mathbf{F}_r is derived, having the form $\mathbf{F}_r^{\text{opt}} = (\mathbf{H}\mathbf{H}^\dagger + \alpha\mathbf{I})^{-1}\mathbf{H}(\mathbf{G}_r + \mathbf{I})$; this is the MMSE filter compensated by the receiver tree processing through $\mathbf{G}_r + \mathbf{I}$ (rather than \mathbf{G}_r). Next, the corresponding AIR bound with $\mathbf{F}_r^{\text{opt}}$ substituted, depends on \mathbf{G}_r through the factor $(\mathbf{G}_r + \mathbf{I})$. An assumption on the matrix \mathbf{G}_r is imposed to have all its eigenvalues strictly larger than -1 , so that $\mathbf{G}_r + \mathbf{I}$ becomes positive semidefinite and hence admits a Cholesky factorization of the form $\mathbf{L}_r^\dagger\mathbf{L}_r$. Accordingly, the AIR bound depends solely on the lower-triangular matrix \mathbf{L}_r . By maximizing this bound, the optimal \mathbf{L}_r is derived, having a shortened (punctured) structure analogous to that of the WLD scheme [15].

In this work, we propose the following modified model

$$\mu_m(\mathbf{y}|\mathbf{x}) = 2\Re\{\mathbf{y}^\dagger\mathbf{F}\mathbf{x}\} - \mathbf{x}^\dagger\mathbf{G}\mathbf{x} + \frac{1}{E_s}\mathbf{x}^\dagger\mathbf{x}, \quad (77)$$

and $p_m(\mathbf{y}|\mathbf{x}) = \exp(\mu_m(\mathbf{y}|\mathbf{x}))$, where tree processing is split into an explicit term $\frac{1}{E_s}\mathbf{x}^\dagger\mathbf{x}$ separate from $\mathbf{x}^\dagger\mathbf{G}\mathbf{x}$ for which \mathbf{G} is subject to optimization. The reason is that the optimal \mathbf{F} in this

case, as will be shown, takes the form $\mathbf{F}^{\text{opt}} = [\mathbf{H}\mathbf{H}^\dagger + \alpha\mathbf{I}]^{-1}\mathbf{H}\mathbf{G}$, and the resulting AIR lower bound depends on \mathbf{G} directly and not through the term $\mathbf{G} + \mathbf{I}$, as is the case with [17]. Hence, the assumption on \mathbf{G} to have all its eigenvalues strictly larger than -1 is dropped. We directly require that \mathbf{G} be positive semidefinite having a Cholesky factorization $\mathbf{J}^\dagger\mathbf{J}$, where \mathbf{J} has the desired punctured lower-triangular form. Under such formulation, we show that the optimal \mathbf{F} and \mathbf{G} coincide exactly with those of the AWLD detector in (65).

Theorem 3: Under the same assumptions as Theorem 1, the optimal \mathbf{F} and \mathbf{G} that maximize $I_{\text{LB}} = \mathbb{E}_{\mathbf{Y},\mathbf{X}}[\ln(p_m(\mathbf{y}|\mathbf{x}))] - \mathbb{E}_{\mathbf{Y}}[\ln(p_m(\mathbf{y}))]$, such that \mathbf{G} is positive semidefinite with factor matrices having a punctured structure of order ν , are

$$\mathbf{F}^{\text{opt}} = \mathbf{M}^\dagger\mathbf{G}^{\text{opt}} \quad \text{and} \quad \mathbf{G}^{\text{opt}} = \mathbf{J}^{\text{opt}\dagger}\mathbf{J}^{\text{opt}}, \quad (78)$$

where \mathbf{M} is the standard $N \times M$ MMSE filter matrix in (54), and \mathbf{J}^{opt} is the punctured augmented WLD matrix \mathbf{L}_{ap} given in (58). Accordingly, the lower bound attained by the AWLD detector in (67) is optimal.

Proof: Let $I_{\text{LB}}^{\text{opt}} = \max_{\mathbf{F},\mathbf{G}:\mathbf{G} \succeq 0} I_{\text{LB}}$ and $(\mathbf{F}^{\text{opt}}, \mathbf{G}^{\text{opt}}) = \text{argmax}_{\mathbf{F},\mathbf{G}:\mathbf{G} \succeq 0} I_{\text{LB}}$. The expectations in the I_{LB} expression with $p_m(\mathbf{y}|\mathbf{x}) = \exp(\mu_m(\mathbf{y}|\mathbf{x}))$ and $p_m(\mathbf{y}) = \int p_m(\mathbf{y}|\mathbf{x})p(\mathbf{x})d\mathbf{x}$ are

$$\begin{aligned} \mathbb{E}_{\mathbf{Y},\mathbf{X}}[\ln(p_m(\mathbf{y}|\mathbf{x}))] &= N - E_s \text{Tr}(\mathbf{G}) + 2E_s \Re\{\text{Tr}(\mathbf{F}^\dagger\mathbf{H})\}, \\ -\mathbb{E}_{\mathbf{Y}}[\ln(p_m(\mathbf{y}))] &= N \ln E_s + \ln \det(\mathbf{G}) - \text{Tr}(\mathbf{F}^\dagger[E_s\mathbf{H}\mathbf{H}^\dagger + N_0\mathbf{I}]\mathbf{F}\mathbf{G}^{-1}). \end{aligned}$$

To determine \mathbf{F} that maximizes I_{LB} , we set the derivative of the terms in the sum of the two expectations involving \mathbf{F} to 0,

$$\frac{\partial}{\partial \mathbf{F}}(2E_s \Re\{\text{Tr}(\mathbf{F}^\dagger\mathbf{H})\} - \text{Tr}(\mathbf{F}^\dagger[E_s\mathbf{H}\mathbf{H}^\dagger + N_0\mathbf{I}]\mathbf{F}\mathbf{G}^{-1})) = 0,$$

from which it follows, after some tedious steps, that

$$\mathbf{F}^{\text{opt}} = [\mathbf{H}\mathbf{H}^\dagger + \alpha\mathbf{I}]^{-1}\mathbf{H}\mathbf{G} = \mathbf{M}^\dagger\mathbf{G}, \quad \alpha = \frac{N_0}{E_s}.$$

Substituting \mathbf{F}^{opt} back in I_{LB} , and noting that $\mathbf{F}^{\text{opt}\dagger}\mathbf{H} = \mathbf{G}(\mathbf{M}\mathbf{H})$ is the product of two Hermitian matrices and hence has real trace, we obtain, after further simplifications

$$\tilde{I}_{\text{LB}}^{\text{opt}} = N \ln E_s + \ln \det(\mathbf{G}) - E_s \text{Tr}((\mathbf{I} - \mathbf{M}\mathbf{H})\mathbf{G}) + N. \quad (79)$$

Using (71), it follows that $E_s(\mathbf{I} - \mathbf{M}\mathbf{H}) = E_s\alpha[\alpha\mathbf{I}_N + \mathbf{H}^\dagger\mathbf{H}]^{-1} = (\mathbf{H}_a^\dagger\mathbf{H}_a)^{-1}$, where \mathbf{H}_a is defined in (47). Then

$$\tilde{I}_{\text{LB}}^{\text{opt}} = N \ln E_s + \ln \det(\mathbf{J}^\dagger\mathbf{J}) - \text{Tr}((\mathbf{L}_a^\dagger\mathbf{L}_a)^{-1}\mathbf{J}^\dagger\mathbf{J}) + N, \quad (80)$$

where $\mathbf{H}_a = \mathbf{Q}_a\mathbf{L}_a$ is the QL decomposition of \mathbf{H}_a , and $\mathbf{G} = \mathbf{J}^\dagger\mathbf{J}$ such that \mathbf{J} is a punctured lower triangular matrix of order ν . We next determine \mathbf{J} that maximizes $\tilde{I}_{\text{LB}}^{\text{opt}}$:

$$\mathbf{J}^{\text{opt}} = \text{argmax}_{\mathbf{J}} \tilde{I}_{\text{LB}}^{\text{opt}}. \quad (81)$$

Assume \mathbf{J} and \mathbf{L}_a are conformally partitioned as

$$\mathbf{J} = \begin{bmatrix} \mathbf{J}_1 & \\ & \mathbf{J}_2 \quad \mathbf{J}_3 \end{bmatrix} \quad \text{and} \quad \mathbf{L}_a = \begin{bmatrix} \mathbf{P}_a & \\ & \mathbf{R}_a \quad \mathbf{S}_a \end{bmatrix}, \quad (82)$$

where $\mathbf{J}_1, \mathbf{P}_a$ are $\nu \times \nu$ lower triangular, \mathbf{J}_3 is $(N-\nu) \times (N-\nu)$ real diagonal, \mathbf{S}_a is $(N-\nu) \times (N-\nu)$ lower triangular, and $\mathbf{J}_2, \mathbf{R}_a$ are $(N-\nu) \times \nu$ matrices. Note that \mathbf{J}_3 is constrained to be a diagonal matrix, not just lower-triangular². Then the trace $\text{Tr}((\mathbf{L}_a^\dagger \mathbf{L}_a)^{-1} \mathbf{J}^\dagger \mathbf{J}) = \text{Tr}((\mathbf{J} \mathbf{L}_a^{-1})(\mathbf{J} \mathbf{L}_a^{-1})^\dagger) = \|\mathbf{J} \mathbf{L}_a^{-1}\|_F^2$ in (80) can be computed using $\mathbf{J} \mathbf{L}_a^{-1}$ as follows:

$$\mathbf{J} \mathbf{L}_a^{-1} = \begin{bmatrix} \mathbf{J}_1 & \\ & \mathbf{J}_2 \quad \mathbf{J}_3 \end{bmatrix} \begin{bmatrix} \mathbf{P}_a^{-1} & \\ -\mathbf{S}_a^{-1} \mathbf{R}_a \mathbf{P}_a^{-1} & \mathbf{S}_a^{-1} \end{bmatrix}$$

$$\|\mathbf{J} \mathbf{L}_a^{-1}\|_F^2 = \|\mathbf{J}_1 \mathbf{P}_a^{-1}\|_F^2 + \|(\mathbf{J}_2 - \mathbf{J}_3 \mathbf{S}_a^{-1} \mathbf{R}_a) \mathbf{P}_a^{-1}\|_F^2 + \|\mathbf{J}_3 \mathbf{S}_a^{-1}\|_F^2.$$

Since the $\ln \det(\mathbf{J}^\dagger \mathbf{J})$ term in (80) involves the diagonal terms of \mathbf{J}_1 and \mathbf{J}_3 only, then $\tilde{I}_{\text{LB}}^{\text{opt}}$ can be optimized for \mathbf{J}_2 and $(\mathbf{J}_1, \mathbf{J}_3)$ independently.

Starting with \mathbf{J}_2 , we set $\frac{\partial}{\partial \mathbf{J}_2} \tilde{I}_{\text{LB}}^{\text{opt}} = \frac{\partial}{\partial \mathbf{J}_2} \text{Tr}((\mathbf{L}_a^\dagger \mathbf{L}_a)^{-1} \mathbf{J}^\dagger \mathbf{J}) = 0$, to obtain $\mathbf{J}_2^{\text{opt}} = \mathbf{J}_3 \mathbf{S}_a^{-1} \mathbf{R}_a$. Substituting back in (80), we get

$$\tilde{I}_{\text{LB}}^{\text{opt}} = N \ln E_s + \ln \det(\mathbf{J}_1^\dagger \mathbf{J}_1) + \ln \det(\mathbf{J}_3^\dagger \mathbf{J}_3) + N - \|\mathbf{J}_1 \mathbf{P}_a^{-1}\|_F^2 - \|\mathbf{J}_3 \mathbf{S}_a^{-1}\|_F^2. \quad (83)$$

Moving to \mathbf{J}_3 , we set $\frac{\partial}{\partial \mathbf{J}_3} \tilde{I}_{\text{LB}}^{\text{opt}} = 0$. Noting that \mathbf{J}_3 is real and diagonal, we obtain $2\mathbf{J}_3^{-1} - 2\mathbf{J}_3 \text{diag}(\mathbf{S}_a^{-1} \mathbf{S}_a^{-\dagger}) = 0$, from which it follows that $\mathbf{J}_3^{\text{opt}} = \text{diag}(\mathbf{S}_a^{-1} \mathbf{S}_a^{-\dagger})^{-1/2} = \mathbf{\Omega}_a$. Substituting back in (83), we get

$$\tilde{I}_{\text{LB}}^{\text{opt}} = N \ln E_s + \ln \det(\mathbf{\Omega}_a^2) - \|\mathbf{\Omega}_a \mathbf{S}_a^{-1}\|_F^2 + N + \ln \det(\mathbf{J}_1^\dagger \mathbf{J}_1) - \text{Tr}((\mathbf{J}_1 \mathbf{P}_a^{-1})(\mathbf{J}_1 \mathbf{P}_a^{-1})^\dagger). \quad (84)$$

Finally, using Lemma 5 below, the optimal \mathbf{J}_1 that maximizes $\tilde{I}_{\text{LB}}^{\text{opt}}$ with \mathbf{P}_a being lower triangular is $\mathbf{J}_1^{\text{opt}} = \mathbf{P}_a$. The resulting \mathbf{J}^{opt} , with $\mathbf{J}_1^{\text{opt}}, \mathbf{J}_2^{\text{opt}}, \mathbf{J}_3^{\text{opt}}$ in place, is

$$\mathbf{J}^{\text{opt}} = \begin{bmatrix} \mathbf{P}_a & \\ & \mathbf{\Omega}_a \mathbf{S}_a^{-1} \mathbf{R}_a \quad \mathbf{\Omega}_a \end{bmatrix}, \quad (85)$$

which coincides with \mathbf{L}_{ap} as given in (58). The optimal lower bound $I_{\text{LB}}^{\text{opt}}$ attained in (84) is

$$I_{\text{LB}}^{\text{opt}} = N \ln E_s + \ln \det(\mathbf{\Omega}_a^2) - \|\mathbf{\Omega}_a \mathbf{S}_a^{-1}\|_F^2 + N + \ln \det(\mathbf{P}_a^\dagger \mathbf{P}_a) - \nu = N \ln E_s + \ln \det(\mathbf{J}^{\text{opt}^\dagger} \mathbf{J}^{\text{opt}}), \quad (86)$$

since $\|\mathbf{\Omega}_a \mathbf{S}_a^{-1}\|_F^2 = N - \nu$, which equals $I_{\text{LB}}^{\text{AWLD}}$ in (67). \blacksquare

Lemma 5: Let \mathbf{U} and \mathbf{V} be two non-singular square matrices in $\mathcal{C}^{N \times N}$. Let $f(\mathbf{U}, \mathbf{V}) = \ln \det(\mathbf{U} \mathbf{U}^\dagger) - \text{Tr}((\mathbf{U} \mathbf{V})(\mathbf{U} \mathbf{V})^\dagger)$ be a real-valued function of complex-valued matrices. Then the optimal \mathbf{U} that maximizes f for a given \mathbf{V} is

$$\mathbf{U}^{\text{opt}} = \underset{\mathbf{U}}{\text{argmax}} f(\mathbf{U}, \mathbf{V}) = \mathbf{V}^{-1},$$

²Hence Lemma (5) is not directly applicable to derive the optimal \mathbf{J} that minimizes (80) at this point.

and $f(\mathbf{U}^{\text{opt}}, \mathbf{V}) = -\sum_{k=1}^N \ln \tilde{v}_{kk}^2 - N$, where \tilde{v}_{kk} is the k th diagonal element of the Cholesky factor of $\mathbf{V}\mathbf{V}^\dagger$.

Proof: See Supplement 1. ■

Discussion: We conclude that punctured augmented channel matrices processed by the AWLD detector are optimal in maximizing the lower bound on the achievable information rate. Their structure matches exactly that of the AIR-PM detector, but most importantly, they can be computed using simple QL decomposition followed by Gaussian elimination, resulting in a significant complexity reduction compared to [18].

VII. EFFICIENT MATRIX DECOMPOSITION ALGORITHMS

A. Matrix-Inverse-Free Puncturing via Gaussian Elimination

Directly inverting \mathbf{S} in (24) can be avoided if we apply Gaussian elimination to null the elements below the main diagonal of $\mathbf{S} = [s_{kj}]$ in (18). Let

$$\mathbf{E}_j = \mathbf{I}_{N-\nu} - \boldsymbol{\tau}_j \mathbf{e}_j^\top \in \mathcal{C}^{(N-\nu) \times (N-\nu)}, \quad (87)$$

be a Gauss transformation [21], where \mathbf{e}_j is the j th column of $\mathbf{I}_{N-\nu}$, and $\boldsymbol{\tau}_j$ is the Gauss vector

$$\boldsymbol{\tau}_j^\top = \underbrace{[0, \dots, 0]}_j, \tau_{j+1}, \dots, \tau_{N-\nu}, \quad \tau_i = \frac{s_{kj}}{s_{jj}}, \quad k = j+1 : N-\nu.$$

Then the operation $\mathbf{E}_j \mathbf{S}$ nulls all the entries below the j th diagonal element in \mathbf{S} . Applying this operation repeatedly for $j=1, \dots, N-\nu-1$ would null all entries in \mathbf{S} below the main diagonal.

Grouping these row operations into

$$\mathbf{E} = \mathbf{E}_{N-\nu-1} \cdots \mathbf{E}_2 \mathbf{E}_1 = \prod_{j=1}^{N-\nu-1} \mathbf{E}_j, \quad (88)$$

results in $\mathbf{E}\mathbf{S} = \text{diag}(\mathbf{S})$, or $\mathbf{S}^{-1} = \text{diag}(\mathbf{S})^{-1} \mathbf{E}$ (note that \mathbf{E} is non-unitary). Setting

$$\boldsymbol{\Omega}_{\mathbf{E}} = \text{diag}(\mathbf{E}\mathbf{E}^\dagger)^{-1/2}, \quad (89)$$

gives the required product $\boldsymbol{\Omega}\mathbf{S}^{-1}$ in (24) inverse-free as

$$\boldsymbol{\Omega}\mathbf{S}^{-1} = \boldsymbol{\Omega}_{\mathbf{E}} \mathbf{E}. \quad (90)$$

B. Eliminating Square-Roots via QDL Decomposition

In forming the QL decomposition $\mathbf{H} = \mathbf{Q}\mathbf{L}$, the j th column \mathbf{q}_j of $\mathbf{Q} = [q_{kj}]$ is obtained by subtracting from the j th column \mathbf{h}_j of \mathbf{H} the orthogonal projection of all other columns of \mathbf{H} (denoted as \mathbf{H}_j) onto \mathbf{h}_j , i.e., $\mathbf{q}_j = \mathbf{h}_j - (\mathbf{H}_j^\dagger \mathbf{h}_j) \mathbf{H}_j$. The j th diagonal element l_{jj} of \mathbf{L} is set to the norm of \mathbf{q}_j , $l_{jj} = \|\mathbf{q}_j\|$. Finally, \mathbf{q}_j is normalized as $\mathbf{q}_j = \mathbf{q}_j / \|\mathbf{q}_j\|$.

The square-root operation required to compute $\|\mathbf{q}_j\| = \sqrt{\sum_{k=1}^N |q_{kj}|^2}$ can be eliminated by working with squared-norms $d_{jj} = \|\mathbf{q}_j\|^2$ instead, and storing them in a diagonal normalizer matrix $\mathbf{D} = [d_{jj}] \in \mathcal{C}^{N \times N}$, apart from the factors \mathbf{Q}, \mathbf{L} . The modified ‘QDL’ decomposition becomes

$$\mathbf{H} = \mathbf{Q}\mathbf{L} = (\mathbf{Q}\mathbf{D}^{-1/2})\mathbf{D}(\mathbf{D}^{-1/2}\mathbf{L}) = \tilde{\mathbf{Q}}\mathbf{D}\tilde{\mathbf{L}}, \quad (91)$$

where $\tilde{\mathbf{Q}} = \mathbf{Q}\mathbf{D}^{-1/2} \in \mathcal{C}^{M \times N}$ is an unnormalized matrix with orthogonal columns $\tilde{\mathbf{Q}}^\dagger \tilde{\mathbf{Q}} = \mathbf{D}^{-1} \neq \mathbf{I}_N$, $\mathbf{D} = \text{diag}(\mathbf{L})^2$, and $\tilde{\mathbf{L}} = \mathbf{D}^{-1/2}\mathbf{L} \in \mathcal{C}^{N \times N}$ is an unnormalized unit lower-triangular matrix. Observe now that the column vectors $\tilde{\mathbf{q}}_j$ of $\tilde{\mathbf{Q}}$ and the diagonal entries \tilde{l}_{jj} of $\tilde{\mathbf{L}}$ both do not involve square-roots also because $\tilde{\mathbf{q}}_j = \mathbf{q}_j / \|\mathbf{q}_j\|^2$ and $\tilde{l}_{jj} = \|\mathbf{q}_j\| / \|\mathbf{q}_j\| = 1$.

The pseudo-codes of the standard (unnormalized) QL algorithm and QDL algorithm are shown in Algs. 1 and 2, respectively. The codes are optimized to produce $\tilde{\mathbf{y}} = \mathbf{Q}^\dagger \mathbf{y}$ and $\tilde{\tilde{\mathbf{y}}} = \tilde{\mathbf{Q}}^\dagger \mathbf{y}$ indirectly as well by augmenting \mathbf{y} to \mathbf{H} and performing modified Gram-Schmidt operations on $[\mathbf{y} \ \mathbf{H}]$.

C. Combined Inverse-Free and Square-Root-Free WLD

The puncturing matrix \mathbf{W}_p and the punctured lower-triangular matrix \mathbf{L}_p can now be expressed in terms of the QDL factors of $\mathbf{H} = \tilde{\mathbf{Q}}\mathbf{D}\tilde{\mathbf{L}}$ as follows. Starting with

$$\tilde{\mathbf{Q}} = \mathbf{Q}\mathbf{D}^{-1/2}, \quad \mathbf{D} = \text{diag}(\mathbf{L})^2, \quad \tilde{\mathbf{L}} = \mathbf{D}^{-1/2}\mathbf{L}, \quad (92)$$

and forming the conformal partitions as in (18),

$$\mathbf{L} = \begin{bmatrix} \mathbf{P} & \mathbf{0} \\ \mathbf{R} & \mathbf{S} \end{bmatrix}, \quad \mathbf{D}^{1/2} = \begin{bmatrix} \mathbf{D}_1^{1/2} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_2^{1/2} \end{bmatrix}, \quad \tilde{\mathbf{L}} = \begin{bmatrix} \tilde{\mathbf{P}} & \mathbf{0} \\ \tilde{\mathbf{R}} & \tilde{\mathbf{S}} \end{bmatrix}, \quad (93)$$

we have $\tilde{\mathbf{P}} = \mathbf{D}_1^{-1/2}\mathbf{P}$ with $\text{diag}(\tilde{\mathbf{P}}) = \mathbf{I}$, $\tilde{\mathbf{R}} = \mathbf{D}_2^{-1/2}\mathbf{R}$, and $\tilde{\mathbf{S}} = \mathbf{D}_2^{-1/2}\mathbf{S}$ with $\text{diag}(\tilde{\mathbf{S}}) = \mathbf{I}$. However, the true \mathbf{W}_p , $\mathbf{\Omega}$, and \mathbf{L}_p ,

$$\mathbf{W}_p = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Omega}\mathbf{S}^{-1} \end{bmatrix}, \quad \mathbf{\Omega} = \text{diag}(\mathbf{S}^{-1}\mathbf{S}^{-\dagger})^{-1/2}, \quad \mathbf{L}_p = \begin{bmatrix} \mathbf{P} & \mathbf{0} \\ \mathbf{\Omega}\mathbf{S}^{-1}\mathbf{R} & \mathbf{\Omega} \end{bmatrix},$$

in (24)-(25) require the inverse of \mathbf{S} , when only the submatrix $\tilde{\mathbf{S}}$ is computed in (93) via the QDL scheme. In addition, $\mathbf{\Omega}$ involves square-root operations. We first expand \mathbf{L}_p as follows

$$\mathbf{L}_p = \mathbf{W}_p\mathbf{L} = \mathbf{W}_p\mathbf{D}^{1/2}\tilde{\mathbf{L}} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Omega}\mathbf{S}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{D}_1^{1/2} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_2^{1/2} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{P}} & \mathbf{0} \\ \tilde{\mathbf{R}} & \tilde{\mathbf{S}} \end{bmatrix}.$$

Substituting $\mathbf{S}^{-1} = \tilde{\mathbf{S}}^{-1}\mathbf{D}_2^{-1/2}$, we obtain

$$\tilde{\mathbf{W}}_p = \mathbf{W}_p\mathbf{D}^{-1/2} = \begin{bmatrix} \mathbf{D}_1^{-1/2} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Omega}\tilde{\mathbf{S}}^{-1}\mathbf{D}_2^{-1} \end{bmatrix} \quad (94)$$

$$\mathbf{L}_p = \mathbf{W}_p\mathbf{L} = \tilde{\mathbf{W}}_p\mathbf{D}\tilde{\mathbf{L}} = \begin{bmatrix} \mathbf{D}_1^{1/2}\tilde{\mathbf{P}} & \mathbf{0} \\ \mathbf{\Omega}\tilde{\mathbf{S}}^{-1}\tilde{\mathbf{R}} & \mathbf{\Omega} \end{bmatrix}. \quad (95)$$

Similarly, we can express $\mathbf{\Omega}$ in terms of $\tilde{\mathbf{S}}$ as follows

$$\mathbf{\Omega} = \text{diag}(\tilde{\mathbf{S}}^{-1}\mathbf{D}_2^{-1}\tilde{\mathbf{S}}^{-\dagger})^{-1/2}. \quad (96)$$

We next eliminate computing the inverse $\tilde{\mathbf{S}}^{-1}$ in the above equations. Using Gaussian elimination, we apply a sequence of Gauss transformations $\tilde{\mathbf{E}}$ to invert $\tilde{\mathbf{S}}$ similar to (87). Since $\tilde{\mathbf{S}}$ has unit diagonal, we obtain $\tilde{\mathbf{E}}\tilde{\mathbf{S}} = \text{diag}(\tilde{\mathbf{S}}) = \mathbf{I}_{N-\nu}$, from which it follows that the inverse of $\tilde{\mathbf{S}}$ is simply

$$\tilde{\mathbf{S}}^{-1} = \tilde{\mathbf{E}}. \quad (97)$$

Substituting $\tilde{\mathbf{E}}$ for $\tilde{\mathbf{S}}^{-1}$ in the equations of $\tilde{\mathbf{W}}_p$ (94), Ω (96), and \mathbf{L}_p (95), we get:

$$\tilde{\mathbf{W}}_p = \begin{bmatrix} \mathbf{D}_1^{-1/2} & \mathbf{0} \\ \mathbf{0} & \Omega \tilde{\mathbf{E}} \mathbf{D}_2^{-1} \end{bmatrix}, \quad \Omega = \text{diag}(\tilde{\mathbf{E}} \mathbf{D}_2^{-1} \tilde{\mathbf{E}}^\dagger)^{-1/2}, \quad \mathbf{L}_p = \begin{bmatrix} \mathbf{D}_1^{1/2} \tilde{\mathbf{P}} & \mathbf{0} \\ \Omega \tilde{\mathbf{E}} \tilde{\mathbf{R}} & \Omega \end{bmatrix}. \quad (98)$$

Note now that the above equations do not involve matrix inversion ($\mathbf{D}_1, \mathbf{D}_2$ are diagonal matrices).

Moving to the square-roots in (98), we show that these operations also are not needed by the detector when computing squared-distances. Since $\mathbf{W}_p \mathbf{Q}^\dagger = \tilde{\mathbf{W}}_p \mathbf{D} \tilde{\mathbf{Q}}^\dagger$ and $\mathbf{W}_p \mathbf{L} = \tilde{\mathbf{W}}_p \mathbf{D} \tilde{\mathbf{L}}$, then

$$\|\mathbf{W}_p(\mathbf{Q}^\dagger \mathbf{y} - \mathbf{L}\mathbf{x})\|^2 = \|\tilde{\mathbf{W}}_p \mathbf{D}(\tilde{\mathbf{Q}}^\dagger \mathbf{y} - \tilde{\mathbf{L}}\mathbf{x})\|^2 = (\tilde{\mathbf{Q}}^\dagger \mathbf{y} - \tilde{\mathbf{L}}\mathbf{x})^\dagger \mathbf{D}^\dagger \tilde{\mathbf{W}}_p^\dagger \tilde{\mathbf{W}}_p \mathbf{D}(\tilde{\mathbf{Q}}^\dagger \mathbf{y} - \tilde{\mathbf{L}}\mathbf{x}). \quad (99)$$

The quantities $\tilde{\mathbf{Q}}^\dagger, \tilde{\mathbf{L}}$ are square-root free, and so is the product

$$\mathbf{D}^\dagger \tilde{\mathbf{W}}_p^\dagger \tilde{\mathbf{W}}_p \mathbf{D} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \tilde{\mathbf{E}}^\dagger \Omega^2 \tilde{\mathbf{E}} \end{bmatrix}, \quad (100)$$

since Ω^2 and $\tilde{\mathbf{E}}$ do not involve square-roots.

The pseudo-code of the optimized WDL decomposition algorithm is shown in Alg. 4. It first performs QDL decomposition on \mathbf{H} , followed by Gaussian elimination. The code is further optimized to eliminate computing the matrix products $\tilde{\mathbf{W}}_p \mathbf{D} \tilde{\mathbf{Q}}^\dagger \mathbf{y}$ and $\tilde{\mathbf{W}}_p \mathbf{D} \tilde{\mathbf{L}} = \mathbf{L}_p$ in (99) explicitly. The QDL procedure first generates $\tilde{\mathbf{y}} = \tilde{\mathbf{Q}}^\dagger \mathbf{y}$ as a byproduct to computing $\tilde{\mathbf{L}}, \mathbf{D}$, and $\tilde{\mathbf{Q}}$. Next, starting with $\tilde{\mathbf{W}} = \tilde{\mathbf{Q}}$, the Gaussian elimination loop then immediately applies the same operations to null the entries in $\tilde{\mathbf{L}}$ on $\tilde{\mathbf{y}}$, as well as on the corresponding columns of $\tilde{\mathbf{W}}$, and updates their resulting squared-norms in \mathbf{D} . The generated $\tilde{\mathbf{W}}^\dagger$ equals $\mathbf{D}^{-1/2} \tilde{\mathbf{W}}_p \mathbf{D} \tilde{\mathbf{Q}}^\dagger$, and the required outputs are formed as $\tilde{\mathbf{W}}^\dagger \mathbf{H} = \tilde{\mathbf{L}}_p$ and $\tilde{\mathbf{W}}^\dagger \mathbf{y} = \tilde{\mathbf{y}}_p$ with an extra scaling factor $\mathbf{D}^{-1/2}$. Note that $\tilde{\mathbf{W}}^\dagger$ operates on \mathbf{H} directly, rather than on $\mathbf{D} \tilde{\mathbf{L}}$ like $\tilde{\mathbf{W}}_p$ in (95) to form $\mathbf{L}_p = \tilde{\mathbf{W}}_p \mathbf{D} \tilde{\mathbf{L}}$. The output quantities $\tilde{\mathbf{L}}_p, \tilde{\mathbf{y}}_p, \mathbf{D}, \tilde{\mathbf{W}}$ from the algorithm are then used to compute the metrics in (99) and (37) as follows

$$\begin{aligned} \|\mathbf{W}_p(\mathbf{Q}^\dagger \mathbf{y} - \mathbf{L}\mathbf{x})\|^2 &= \|\mathbf{W}_p \mathbf{Q}^\dagger (\mathbf{y} - \mathbf{H}\mathbf{x})\|^2 = \|\mathbf{D}^{1/2} \tilde{\mathbf{W}}^\dagger (\mathbf{y} - \mathbf{H}\mathbf{x})\|^2 \\ &= \|\mathbf{D}^{1/2} (\tilde{\mathbf{y}}_p - \tilde{\mathbf{L}}_p \mathbf{x})\|^2 = (\tilde{\mathbf{y}}_p - \tilde{\mathbf{L}}_p \mathbf{x})^\dagger \mathbf{D} (\tilde{\mathbf{y}}_p - \tilde{\mathbf{L}}_p \mathbf{x}), \\ \mu_p(\mathbf{y}|\mathbf{x}) &= -\frac{1}{N_0} (\tilde{\mathbf{y}}_p - \tilde{\mathbf{L}}_p \mathbf{x})^\dagger \mathbf{D} (\tilde{\mathbf{y}}_p - \tilde{\mathbf{L}}_p \mathbf{x}). \end{aligned} \quad (101)$$

For reference, an optimized version of the standard (unnormalized) WL algorithm of [32] is listed in Alg. 3. The outputs $\mathbf{L}_p, \mathbf{y}_p, \mathbf{W}$ from this algorithm compute (99) as follows

$$\|\mathbf{W}_p(\mathbf{Q}^\dagger \mathbf{y} - \mathbf{L}\mathbf{x})\|^2 = \|\mathbf{W}_p \mathbf{Q}^\dagger (\mathbf{y} - \mathbf{H}\mathbf{x})\|^2 = \|\mathbf{W}(\mathbf{y} - \mathbf{H}\mathbf{x})\|^2.$$

D. Eliminating Explicit Computation of MMSE Filter Matrix

For the AWLD detector, the MMSE filter matrix \mathbf{M} in (54) is needed to compute the metrics in (63)-(64). This \mathbf{M} is to be pre-multiplied with $\mathbf{W}_{ap}\mathbf{L}_a = \mathbf{L}_{ap}$ and applied to \mathbf{y} in (63), or pre-multiplied with $\mathbf{L}_{ap}^\dagger \mathbf{L}_{ap}$ and then applied to \mathbf{y} in (64). In either case, working with the quantity $\mathbf{W}_{ap}\mathbf{L}_a\mathbf{M}$ suffices. However, (56) shows that \mathbf{M} can be obtained from the QL decomposition of \mathbf{H}_a in (47) as $\sqrt{\beta}\mathbf{Q}_{a2}\mathbf{Q}_{a1}^\dagger$ without explicitly inverting \mathbf{H}_a . But $\mathbf{L}_a\mathbf{Q}_{a2} = \frac{1}{\sqrt{E_s}}\mathbf{I}_N$ from (52), so that $\mathbf{W}_{ap}\mathbf{L}_a\mathbf{M}\mathbf{y}$ actually reduces to $\frac{1}{\sqrt{N_0}}\mathbf{W}_{ap}\mathbf{Q}_{a1}^\dagger\mathbf{y}$. The product $\frac{1}{\sqrt{N_0}}\mathbf{Q}_{a1}^\dagger\mathbf{y}$ can be obtained indirectly from the QL decomposition procedure (Alg. 1) when applied to \mathbf{H}_a and $\mathbf{y}_a = \frac{1}{\sqrt{N_0}}[\mathbf{y}; \mathbf{0}_{N \times 1}]$ as $\mathbf{Q}_{a1}^\dagger\mathbf{y}_a = \frac{1}{\sqrt{N_0}}[\mathbf{Q}_{a1}^\dagger \ \mathbf{Q}_{a2}^\dagger][\mathbf{y}; \mathbf{0}] = \frac{1}{\sqrt{N_0}}\mathbf{Q}_{a1}^\dagger\mathbf{y}$, in addition to generating \mathbf{L}_a . Finally, applying \mathbf{W}_{ap} to puncture \mathbf{L}_a can be done using Gaussian elimination as before, with the elimination operations simultaneously applied to $\tilde{\mathbf{y}}_a = \frac{1}{\sqrt{N_0}}\mathbf{Q}_{a1}^\dagger\mathbf{y}$ to generate the product $\mathbf{y}_{ap} = \frac{1}{\sqrt{N_0}}\mathbf{W}_{ap}\mathbf{Q}_{a1}^\dagger\mathbf{y}$. Therefore, the WL algorithm in Alg. 3, when applied to \mathbf{H}_a and \mathbf{y}_a , produces the necessary quantities to compute the metrics in (62)-(63), without any matrix inversion, as

$$\|\mathbf{L}_a(\mathbf{M}\mathbf{y} - \mathbf{x})\|^2 = \|\frac{1}{\sqrt{N_0}}\mathbf{Q}_{a1}^\dagger\mathbf{y} - \mathbf{L}_a\mathbf{x}\|^2 = \|\tilde{\mathbf{y}}_a - \mathbf{L}_a\mathbf{x}\|^2 \quad (102)$$

$$\mu_{ap}(\mathbf{y}|\mathbf{x}) = \frac{1}{E_s}\|\mathbf{x}\|^2 - \|\mathbf{W}_{ap}\mathbf{L}_a(\mathbf{M}\mathbf{y} - \mathbf{x})\|^2 \quad (103)$$

$$= \frac{1}{E_s}\|\mathbf{x}\|^2 - \|\mathbf{W}_{ap}(\tilde{\mathbf{y}}_a - \mathbf{L}_a\mathbf{x})\|^2 \quad (104)$$

$$= \frac{1}{E_s}\|\mathbf{x}\|^2 - \|\mathbf{y}_{ap} - \mathbf{L}_{ap}\mathbf{x}\|^2. \quad (105)$$

Similarly, the WDL procedure in Alg. 4 generates these quantities without any square-root operations (assuming $\sqrt{E_s}, \sqrt{N_0}$ are available at the input to form $\mathbf{H}_a, \mathbf{y}_a$). The output quantities from the algorithm, now labeled as $\tilde{\mathbf{L}}_{ap}, \mathbf{D}, \tilde{\mathbf{y}}_{ap}, \tilde{\mathbf{W}}$, are used to compute the above metrics as

$$\begin{aligned} \|\mathbf{W}_{ap}(\frac{\mathbf{Q}_{a1}^\dagger\mathbf{y}}{\sqrt{N_0}} - \mathbf{L}_a\mathbf{x})\|^2 &= \|\mathbf{D}^{1/2}\tilde{\mathbf{W}}^\dagger(\frac{\mathbf{y}}{\sqrt{N_0}} - \mathbf{H}\mathbf{x})\|^2 = \|\mathbf{D}^{1/2}(\tilde{\mathbf{y}}_{ap} - \tilde{\mathbf{L}}_{ap}\mathbf{x})\|^2 \\ &= (\tilde{\mathbf{y}}_{ap} - \tilde{\mathbf{L}}_{ap}\mathbf{x})^\dagger \mathbf{D} (\tilde{\mathbf{y}}_{ap} - \tilde{\mathbf{L}}_{ap}\mathbf{x}), \end{aligned} \quad (106)$$

$$\mu_{ap}(\mathbf{y}|\mathbf{x}) = \frac{1}{E_s}\|\mathbf{x}\|^2 - \|\mathbf{W}_{ap}(\frac{1}{\sqrt{N_0}}\mathbf{Q}_{a1}^\dagger\mathbf{y} - \mathbf{L}_a\mathbf{x})\|^2 = \frac{1}{E_s}\|\mathbf{x}\|^2 - (\tilde{\mathbf{y}}_{ap} - \tilde{\mathbf{L}}_{ap}\mathbf{x})^\dagger \mathbf{D} (\tilde{\mathbf{y}}_{ap} - \tilde{\mathbf{L}}_{ap}\mathbf{x}). \quad (107)$$

E. Combined Two-Sided QLZ and WLZ Decompositions

The reduction and elimination operations for two-sided decompositions of Sec. III-B can be combined efficiently as shown in Alg. 6. The code starts with QL decomposition, and then performs right reduction followed immediately by left elimination operations, analogous to (26)-(27). The matrix \mathbf{Z} and its inverse are updated with every right operation, the matrix \mathbf{W} and output vector \mathbf{y}_p are updated with every left operation, while \mathbf{L}_p is updated after each of these operations. For reference, Alg. 5 shows the code for QLZ decomposition with right reduction but without elimination. Note again that the generated \mathbf{W} is related to \mathbf{W}_p in (31) as $\mathbf{W}^\dagger = \mathbf{W}_p\mathbf{Q}^\dagger$.

Table [II](#) in the supplement summarizes all algorithms presented in this section, and highlights their main features.

VIII. (A)WLD-BASED MIMO DETECTION ALGORITHMS

In this section, we present computationally-efficient soft-output MIMO detection algorithms based on the AWLD, WLD, and WLZ puncturing schemes, and compare them with the LORD algorithm [[14](#)]. Supplement Table [III](#) summarizes all the algorithms discussed and their features in terms of decomposition and puncturing schemes, metric used for LLR computation, marginalization on child layers, single-tree versus multi-tree, as well as local versus global metric update. The pseudo-codes of all algorithms are available in the supplement.

In general, MLM bit LLRs are computed using [\(7\)](#) as Max-Log approximations of the exact ML LLRs in [\(6\)](#), with the true metric $\mu(\mathbf{y}|\mathbf{x}) = \mu(\tilde{\mathbf{y}}|\mathbf{x}) = -\frac{1}{N_0}\|\tilde{\mathbf{y}} - \mathbf{L}\mathbf{x}\|^2$ as defined in [\(3\)](#). For an arbitrary \mathbf{L} partitioned into ν parent layers and $N-\nu$ child layers, the exact maximizations in [\(7\)](#) are expressed as in [\(12\)](#)-[\(13\)](#), and approximated using ZF-DF on child symbols using [\(16\)](#)-[\(17\)](#). For a punctured \mathbf{L} , alternative metrics $\mu_p(\tilde{\mathbf{y}}|\mathbf{x})$ and $\mu_{ap}(\tilde{\mathbf{y}}|\mathbf{x})$ to $\mu(\tilde{\mathbf{y}}|\mathbf{x})$ are derived in [\(101\)](#) and [\(107\)](#) under the optimized WLD and AWLD models. When $\mu_p(\tilde{\mathbf{y}}|\mathbf{x})$ and $\mu_{ap}(\tilde{\mathbf{y}}|\mathbf{x})$ are used in [\(7\)](#) instead of $\mu(\tilde{\mathbf{y}}|\mathbf{x})$, and with \mathbf{L} being punctured, then all child symbols become leaves, decision feedback disappears, and the ZF-DF approximations in [\(14\)](#)-[\(15\)](#) turn into exact LS estimates as required to satisfy [\(12\)](#)-[\(13\)](#).

A. AWDL MIMO Detection Algorithm

An augmented channel is first formed as in [\(47\)](#), and then punctured using Alg. [4](#) for a given ν , to yield $\tilde{\mathbf{L}}_{ap}, \tilde{\mathbf{y}}_{ap}, \mathbf{D}$, with the following structure: $\tilde{\mathbf{L}}_{ap} = \begin{bmatrix} \tilde{\mathbf{P}} & \mathbf{0} \\ \tilde{\mathbf{R}} & \mathbf{I} \end{bmatrix}$, $\tilde{\mathbf{y}}_{ap} = [\tilde{\mathbf{y}}_1; \tilde{\mathbf{y}}_2]$, and $\mathbf{D} = \begin{bmatrix} \mathbf{D}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_2 \end{bmatrix}$. With $\mathbf{x} = [\mathbf{x}_1; \mathbf{x}_2]$, the bit LLRs are computed as

$$\Lambda_{ap}(x_{n,b}|\mathbf{y}) = \max_{\mathbf{x}:x_{n,b}=+1} \mu_{ap}(\mathbf{y}|\mathbf{x}) - \max_{\mathbf{x}:x_{n,b}=-1} \mu_{ap}(\mathbf{y}|\mathbf{x}),$$

$$\mu_{ap}(\mathbf{y}|\mathbf{x}) = \frac{1}{E_s} \|\mathbf{x}\|^2 - (\tilde{\mathbf{y}}_{ap} - \tilde{\mathbf{L}}_{ap}\mathbf{x})^\dagger \mathbf{D} (\tilde{\mathbf{y}}_{ap} - \tilde{\mathbf{L}}_{ap}\mathbf{x}) \triangleq \mu_1(\tilde{\mathbf{y}}_1|\mathbf{x}_1) + \mu_2(\tilde{\mathbf{y}}_2|\mathbf{x}_1, \mathbf{x}_2),$$

where $\mu_1(\tilde{\mathbf{y}}_1|\mathbf{x}_1) = \frac{1}{E_s} \|\mathbf{x}_1\|^2 - \|\tilde{\mathbf{y}}_1 - \tilde{\mathbf{P}}\mathbf{x}_1\|_{\mathbf{D}_1}^2$ and $\mu_2(\tilde{\mathbf{y}}_2|\mathbf{x}_1, \mathbf{x}_2) = \frac{1}{E_s} \|\mathbf{x}_2\|^2 - \|\tilde{\mathbf{y}}_2 - \tilde{\mathbf{R}}\mathbf{x}_1 - \mathbf{x}_2\|_{\mathbf{D}_2}^2$. Next, for any \mathbf{x}_1 , the leaf symbols \mathbf{x}_2 that maximize μ_2 are obtained through LS by setting the derivative of μ_2 with respect \mathbf{x}_2 to 0. We obtain $\frac{1}{E_s} \mathbf{x}_2^\dagger + (\tilde{\mathbf{y}}_2 - \tilde{\mathbf{R}}\mathbf{x}_1 - \mathbf{x}_2)^\dagger \mathbf{D}_2 = 0$, from which it follows that

$$\begin{aligned} \max_{\mathbf{x}:x_{n,b}=s} \mu_{\text{ap}}(\tilde{\mathbf{y}}|\mathbf{x}) &= \max_{\mathbf{x}_1:x_{n,b}=s} \left\{ \mu_1(\tilde{\mathbf{y}}_1|\mathbf{x}_1) + \max_{\mathbf{x}_2} \mu_2(\tilde{\mathbf{y}}_2|\mathbf{x}_1, \mathbf{x}_2) \right\} \\ &= \max_{\mathbf{x}_1:x_{n,b}=s} \left\{ \mu_1(\tilde{\mathbf{y}}_1|\mathbf{x}_1) + \hat{\mu}_2(\tilde{\mathbf{y}}_2|\mathbf{x}_1, \hat{\mathbf{x}}_2) \right\}, \end{aligned} \quad (108)$$

$$\begin{aligned} \hat{\mathbf{x}}_2 &= \left[(\mathbf{D}_2 - \frac{1}{E_s} \mathbf{I}_{N-\nu})^{-1} \mathbf{D}_2 (\tilde{\mathbf{y}}_2 - \tilde{\mathbf{R}} \mathbf{x}_1) \right]_{\mathcal{X}^{N-\nu}}, \\ \max_{\mathbf{x}:x_{n,b}=s} \mu_{\text{ap}}(\tilde{\mathbf{y}}|\mathbf{x}) &= \max_{\mathbf{x}_1} \left\{ \mu_1(\tilde{\mathbf{y}}_1|\mathbf{x}_1) + \max_{\mathbf{x}_2:x_{n,b}=s} \mu_2(\tilde{\mathbf{y}}_2|\mathbf{x}_1, \mathbf{x}_2) \right\} \\ &= \max_{\mathbf{x}_1} \left\{ \mu_1(\tilde{\mathbf{y}}_1|\mathbf{x}_1) + \hat{\mu}_2(\tilde{\mathbf{y}}_2|\mathbf{x}_1, \hat{\mathbf{x}}_{2n;b}^{(s)}) \right\}, \end{aligned} \quad (109)$$

$$\hat{\mathbf{x}}_{2n;b}^{(s)} \triangleq \left[(\mathbf{D}_2 - \frac{1}{E_s} \mathbf{I}_{N-\nu})^{-1} \mathbf{D}_2 (\tilde{\mathbf{y}}_2 - \tilde{\mathbf{R}} \mathbf{x}_1) \right]_{\mathcal{A}_{n,b}^{(s)}},$$

for $s = \pm 1$, where $\mathcal{A}_{n,b}^{(s)} = \{[x_{\nu+1}, \dots, x_n, \dots, x_N]^T \in \mathcal{X}^{N-\nu} : x_{n,b} = s\}$.

The pseudo-code of the multi-tree version of the AWDL algorithm is shown in Alg. 9. It performs multiple runs, each time grouping a new set of ν layers as parents to generate bit LLRs using (108) only. The multi-tree WLD algorithm that implements (108) but using the metric μ_p in (101) is shown in Alg. 7. For reference, the pseudo-code of the multi-tree LORD algorithm that implements (16) with the true metric (3) is shown in Alg. 11. Because its channel \mathbf{L} is full lower-triangular, LORD applies ZF-DF rather than LS to estimate the child symbols, resulting in a significant increase in computational complexity compared to WLD/AWDL. Note that for all three algorithms, the multiple runs are independent and the metrics computed are used to update just the tracked maxima of the parent layer bits only, and are not globally shared to update the maxima for other bits.

The search space of $|\mathcal{X}|^\nu$ parent symbol vectors of the AWDL algorithm can be reduced to $\nu \cdot |\mathcal{X}|$ by enumerating only over the root and applying ZF-DF on the other $\nu-1$ parents. Using Lemma 3, for a given choice of ν layers as parents and $N-\nu$ layers as leaves, the metric of a given symbol vector does not change if the parent layers are permuted and the leaf layers are permuted. Hence, doing ν runs over the parent layers, each time with a different layer as root, would improve the estimates by updating the maxima being tracked for the bits on *all* ν parents in each run, and not just those bits of the current root symbol. For the case $\nu=2$, the search on layer 2 can be limited to a small window of η symbols around the ZF solution. Empirical simulations demonstrate that $\eta=4$ is sufficient to achieve the accuracy of enumerating all $|\mathcal{X}|^2$ parent symbol vectors. The pseudo-code of the windowed-AWDL algorithm is shown in Alg. 10.

Finally, the LORD algorithm can be similarly optimized to update the tracked maxima for each bit hypothesis on *all* layers in each run as shown on Alg. 12. This is possible in this case because Euclidean distances do not change under column permutation of \mathbf{H} : $\|\mathbf{y} - \mathbf{H}\mathbf{x}\| = \|\mathbf{y} - \mathbf{H}\mathbf{J}\mathbf{J}^T\mathbf{x}\| = \|\tilde{\mathbf{y}} - \mathbf{L}\mathbf{J}^T\mathbf{x}\|$ for any permutation \mathbf{J} , where $\mathbf{H}\mathbf{J} = \mathbf{Q}\mathbf{L}$ and $\tilde{\mathbf{y}} = \mathbf{Q}^T\mathbf{y}$.

B. WLZ-Based MIMO Detection Algorithm

While the metrics of the WLD and AWDL algorithms are not preserved under arbitrary layer permutation as they are for LORD, they are *approximately* preserved under 2-sided WLZ decomposition (Section III-B, Alg. 6). The pseudo-code of the multi-tree version of the WLZ algorithm shown in Alg. 8 implements (108) similar to the WLD detection algorithm of Alg. 7, but with μ_p in (101) being based on the 2-sided WLZ rather than the 1-sided WL decomposition.

IX. SIMULATION RESULTS

A library of MIMO detection algorithms have been implemented and characterized for both algorithmic performance and computational complexity. Fast-fading Rayleigh complex MIMO channels are assumed. In Fig. 4a, we compare the achievable rates of the proposed WLZ and AWLD detectors against the AIR-PM detector [18], as well as the ZF, MMSE, and WLD [15] for 8×8 MIMO channels, assuming Gaussian inputs and with parent layers selected so as to maximize I_{LB}^{WLD} in (41). The AWLD and WLD are simulated for both $\nu=1$ and $\nu=2$ configurations, while WLZ is simulated for ν and $c=1,2$. WLZ attains the closest rate to capacity with reduction parameter $c=2$, followed by AWLD/AIR-PM (which attain the same rate), followed by WLD. This is because as $\rho = 1/2^{c+1/2}$ decreases by increasing c , then from Lemma 1 and (33), \mathbf{W}_p gets closer to \mathbf{I} and \mathbf{L}_p approaches the true unpunctured \mathbf{L} . Hence from Theorem 1, the lower bound on the achievable rate I_{LB}^{WLD} approaches the capacity of the channel.

On the other hand, Fig. 4b plots the AIR of AWLD and WLD with $\nu=1$ for finite constellations. The AWLD achieves higher rates than WLD, especially for 64QAM. Parent layers are selected to maximize I_{LB}^{WLD} in (41) if Gaussian inputs were used. For the AWLD scheme at very low SNR regimes, it attains higher rates for low-order constellations compared to denser constellations. For SNRs beyond ~ 10 dB, the trend gets reversed, with the AWLD scheme attaining higher rates for denser constellations. Similarly for the WLD scheme. However, the SNRs for which denser constellations start to outperform low-order constellations are much higher than those of the AWLD case; the rate for which 16QAM becomes better than that for QPSK is roughly 20 dB for the WLD case, while for the AWLD case, it is roughly 6 dB. The same applies between 64QAM and 16QAM, but at an impractically very high SNR value (range not shown in the figure). The reason is that the WLD scheme is not optimal, and misses the ML decision for denser constellations at low SNRs more often compared to the AWLD scheme.

In Figs. 5-6, we compare the frame error rate (FER) of the proposed WLZ and AWLD detectors against the Max-Log ML (MLM) sphere decoder with optimized pruning [6], ZF, K-

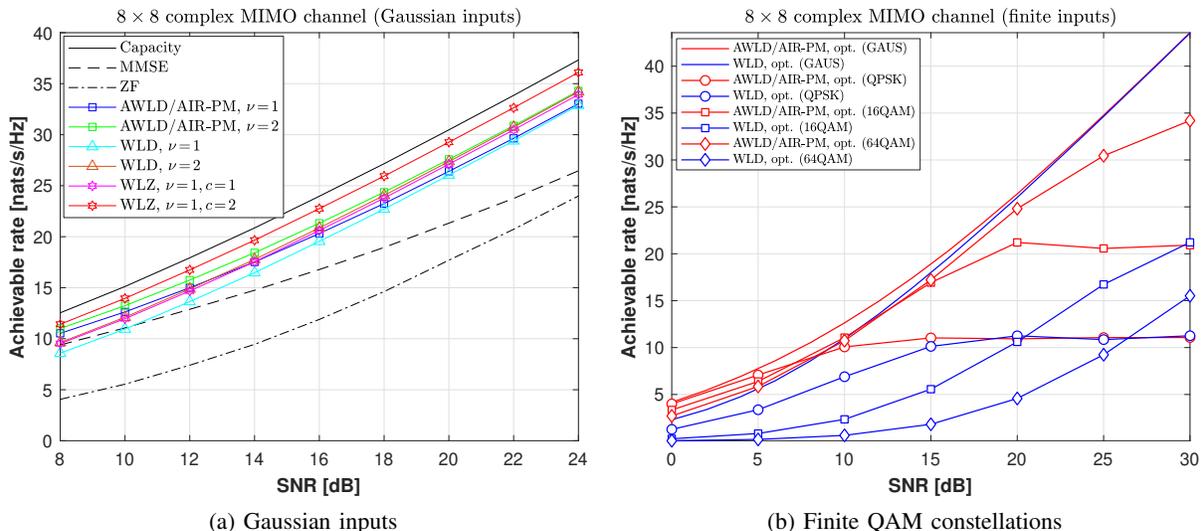


Fig. 4. Comparison of AIRs for 8×8 MIMO channels with (a) Gaussian inputs, and (b) finite QAM constellations.

best [10], LORD [14], WLD [15], and AIR-PM [18] detectors for various MIMO dimensions, QAM constellations, and puncturing orders. Max-log approximations for exponential sums are used. An LTE rate-1/2 punctured turbo code of length 1024 is used, and 8 turbo decoder iterations are performed. For K-best, sorted-QRD [33] is used, and the K best competing paths are retained. Counter hypotheses are formed relative to the best survivor path. Counter hypotheses of all leaf bits are updated using the optimization in [6]. Un-updated LLR values are replaced with the minimum LLR in the corresponding symbol.

For LORD, both $\nu=1, 2$ are simulated using the multi-tree approach; N/ν rounds of ν -layer parent selections, QLDs, and ZF-DF steps on the $N-\nu$ child layers are performed. LORD- $L\nu 1$ and LORD- $L\nu 2$ perform local (within-tree) metric updates only (Alg. 11), while LORD- $G\nu 1$ and LORD- $G\nu 2$ perform global (across all trees) metric updates (Alg. 12). For $\nu=2$, consecutive layer pairing is done.

For WLD, WLD- $L\nu 1$ and WLD- $L\nu 2$ perform local metric updates only (Alg. 7). WLD- $X\nu 2\eta 1$ enumerates on parent 1, does ZF on parent 2, and ZF-DF on child nodes. WLD- $X\nu 2\eta 4$ enumerates on parent 1, then enumerates over a window of 4 symbols around the ZF solution (ZF-W) on parent 2, and does ZF-DF on child nodes. Both WLD-X algorithms update metrics across tree pairs. Similarly for AWLD; AWLD- $L\nu 1$ and AWLD- $L\nu 2$ apply Alg. 9 using augmented channel puncturing with local metric updates, while AWLD- $X\nu 1\eta 1$ and AWLD- $X\nu 2\eta 4$ are similar to their WLD counter parts but apply augmented puncturing (Alg. 10).

For AIR-PM, the single-tree approach is used. AIR- $r-S\nu 1$ randomly selects a parent and

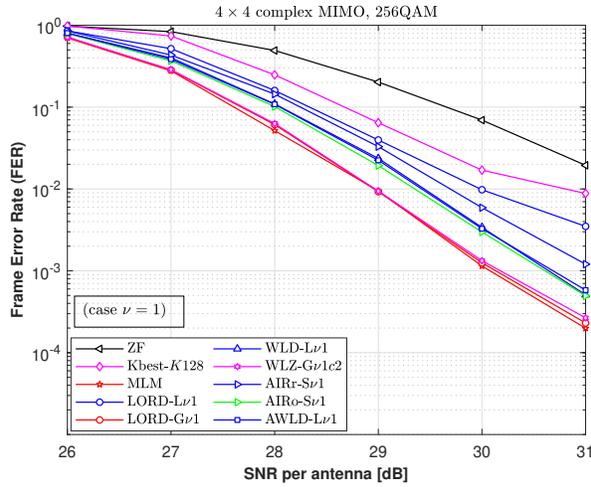
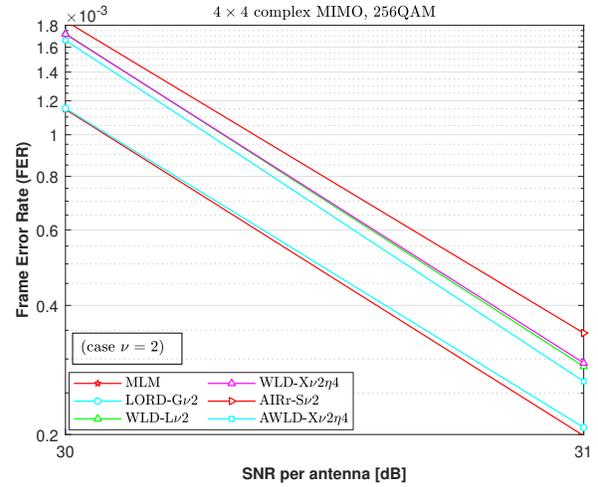
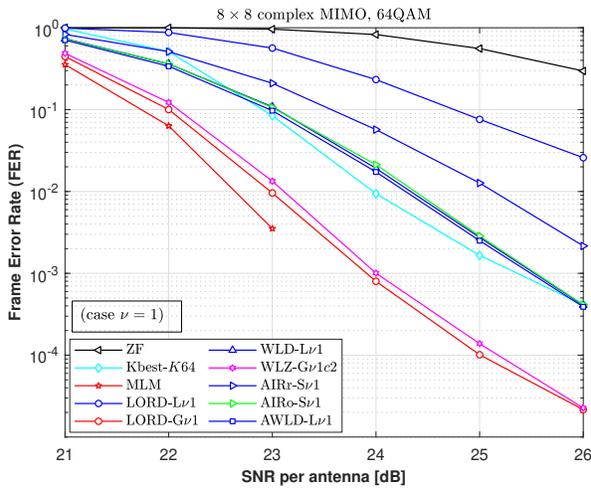
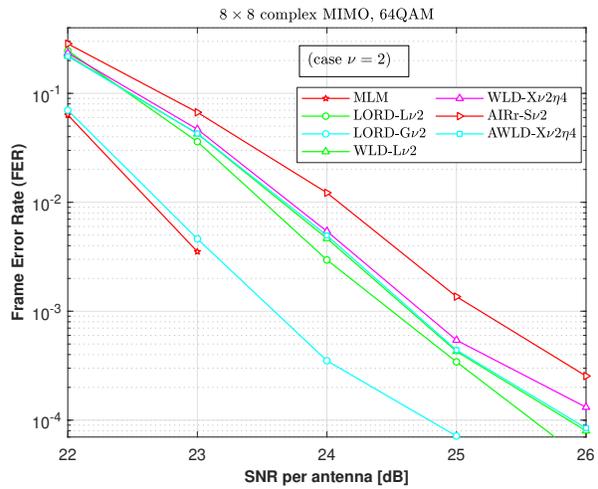
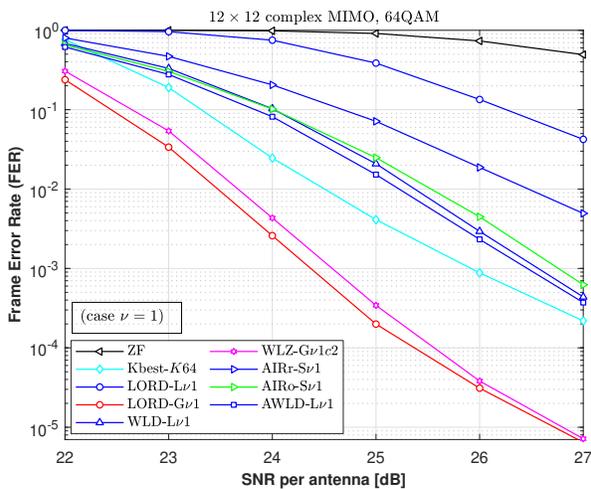
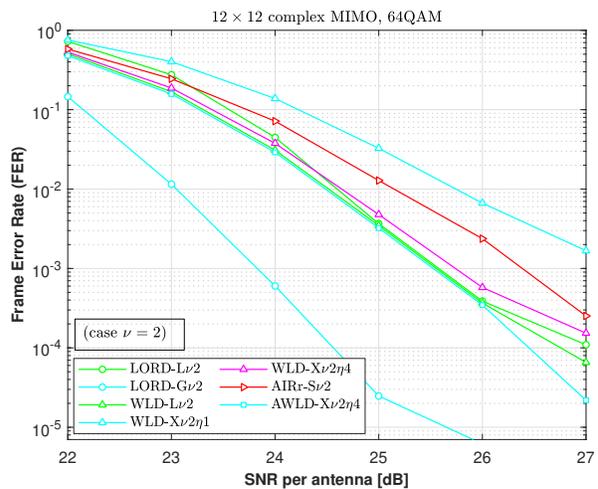
(a) 4 × 4, 256QAM, $\nu = 1$ (b) 4 × 4, 256QAM, $\nu = 2$ (c) 8 × 8, 64QAM, $\nu = 1$ (d) 8 × 8, 64QAM, $\nu = 2$ (e) 12 × 12, 64QAM, $\nu = 1$ (f) 12 × 12, 64QAM, $\nu = 2$

Fig. 5. Comparisons of FERs vs. SNR for various QAM constellations, puncturing orders, and MIMO dimensions 4, 8, and, 12.

orders the other child layers, while AIRo-S ν 1 does optimal layer ordering to maximize the AIR assuming Gaussian inputs. AIRr-S ν 2 uses two parents with random layer ordering.

For WLZ, 2-sided puncturing and reduction are applied using Alg. 8. WLZ-L ν 1 c 2 does local metric updates with one parent and $c = 2$. Similarly, WLZ-G ν 1 c 2 and WLZ-G ν 1 c 3 perform global metric updates with $c=2, 3$, respectively.

Several observations can be made: 1) Multi-tree approaches are superior to single-tree approaches, and are less sensitive to layer ordering. 2) Global metric updates across trees significantly improves performance compared to local within-tree only updates. 3) For trees with more than one parent, there is no need to enumerate across all $|\mathcal{X}|^\nu$ parent combinations. Running ν trees instead, each time enumerating on one parent and doing ZF-W only on parent 2 is as good. 4) Augmented-WLD based algorithms consistently perform better than their WLD counterparts. 5) Two-sided WLZ based algorithms perform better than AWLD and WLD, and almost match the performance of LORD with global metric updates (LORD has dense L, while WLZ has punctured L). 6) Puncturing remains very effective even for large MIMO dimensions.

Figure 7 plots the LLR distributions of bits 1 and 3 of one symbol in a 4×4 , 16QAM MIMO system at SNR=20 dB. As shown, AWLD and WLZ track the optimal LLRs very closely.

The complexity of various algorithms is benchmarked and compared in Fig. 8 for an 8×8 MIMO system and 64QAM. The figure plots the SNR required to achieve a target FER of 0.1% versus normalized complexity. All algorithms (matrix decomposition, filtering, MMSE, MIMO detection) are first implemented using fixed-point arithmetic, and then profiled in terms of memory storage requirements and kernel mathematical operations. These operations include (both for real and complex quantities, where applicable): multiplication, division, multiply-accumulate, squaring, addition/subtraction, inversion, (inverse) square-root, slicing, look-up table (LUT) operations, comparison operations, vector norm and norm-square, multiplexing, sorting, and permutation. The gate-count complexity of these operations is evaluated by mapping them to a library of pre-characterized logic gates that includes basic adders/subtractors, multipliers, squarers, dividers, multiplexers, memory elements, comparators, slicers, and (inverse) square-roots. As a result, each operation is characterized with a gate complexity value.

Parallel architectures for all algorithms are developed, and their gate-count complexity is plotted in Fig. 8. For the MLM algorithm, a serial depth-first tree traversal architecture is developed, and its complexity is reported as gate-count per tree node, multiplied by the number of nodes visited. Since the latter is non-deterministic, the value reported is averaged over 1000

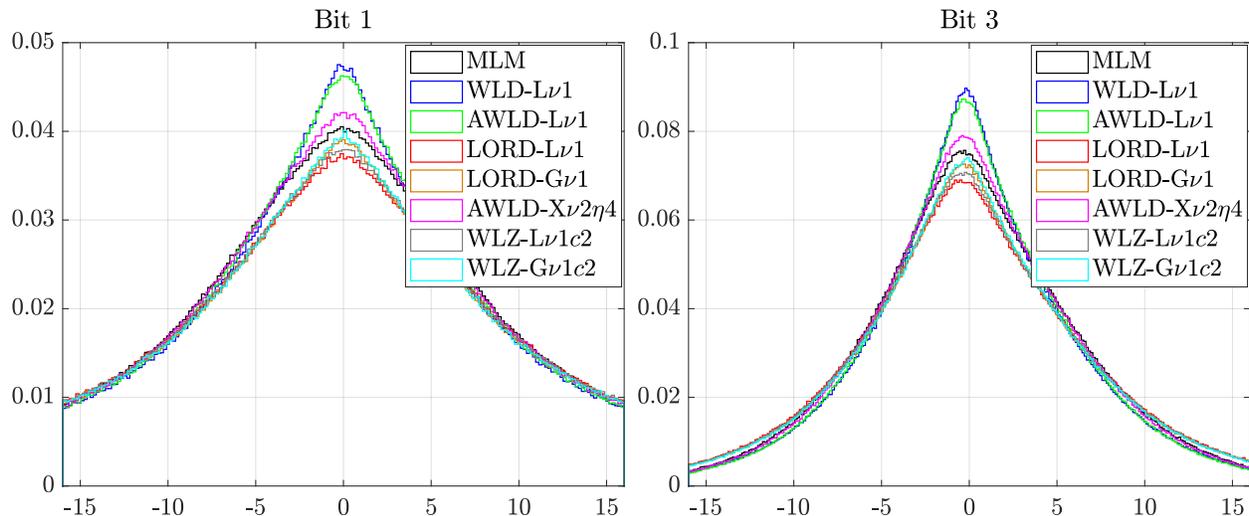


Fig. 7. Distribution of LLRs for bits 1 and 3 of one symbol: 4×4 complex MIMO channel, 16QAM, SNR = 20 dB.

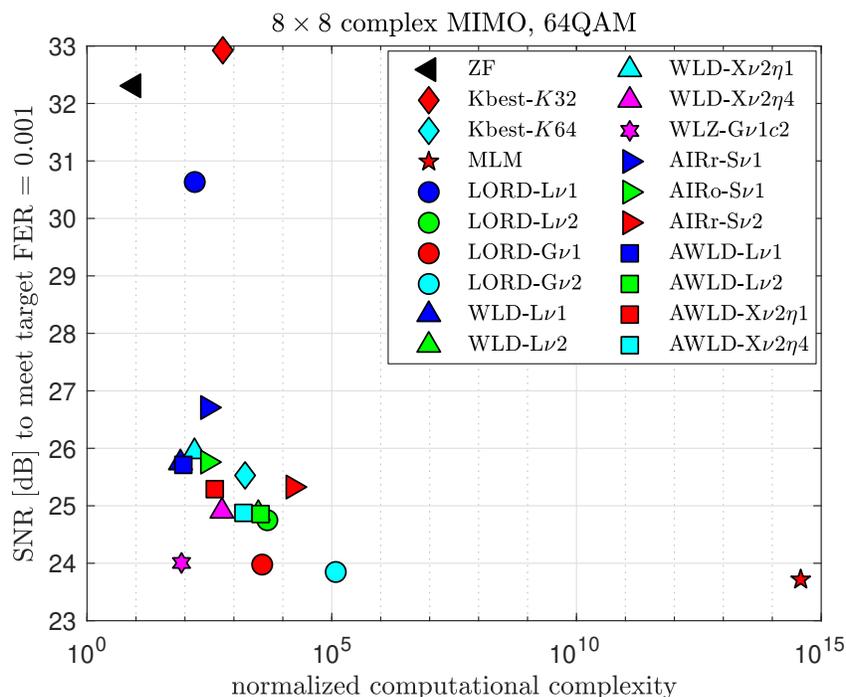


Fig. 8. SNR to meet a target FER of 0.1% versus complexity.

detection trials. For the K-best algorithm, a K -wide parallel architecture is developed.

As expected, the ZF and MLM algorithms lie at opposite extremes in the performance-complexity space. The proposed WLZ algorithm offers the best performance-complexity tradeoff among all algorithms. It matches the performance of LORD at roughly 20x less complexity. The savings are primarily due to the eliminated complex multiplications in \mathbf{L} as a result of the puncturing and reduction operations.

X. CONCLUSIONS

Channel puncturing in augmented and two-sided forms has been investigated in this work as an effective means to reduce computational complexity of tree-based soft-output MIMO detectors. It has been shown that punctured augmented channel matrices processed by the AWLD detector are optimal in maximizing the lower bound on the achievable information rate. Their structure matches exactly that of AIR-PM, but most importantly, they can be derived using simple QL decomposition followed by Gaussian elimination. When used in multi-tree mode with local metric updates, AWLD beats LORD both performance-wise and complexity-wise. However, LORD, when optimized to operate with global across-tree metric updates, attains a significant performance gain that AWLD cannot match because its puncturing matrix is non-unitary, and hence Euclidean-distance based metrics are not preserved under column permutations in multiple trees. This shortcoming is mitigated by employing two-sided puncturing based on right-sided integer reduction and left-sided elimination. The resulting puncturing matrices processed by WLZ are almost unitary, and hence the global across-tree metric update property of LORD is retained. The result is that the proposed WLZ scheme offers the best performance-complexity tradeoff among tree-based detectors. Finally, extensions to include soft-input information, imperfect channel estimation effects, and correlated channels are directly applicable based on [18].

REFERENCES

- [1] M. M. Mansour, "Optimal augmented-channel puncturing for low-complexity soft-output MIMO detectors," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Dublin, Ireland, Jun. 7–11, 2020.
- [2] C. Xu, S. Sugiura, S. X. Ng, P. Zhang, L. Wang, and L. Hanzo, "Two decades of MIMO design tradeoffs and reduced-complexity MIMO detection in near-capacity systems," *IEEE Access*, vol. 5, pp. 18 564–18 632, May 2017.
- [3] E. Larsson, "MIMO detection methods: How they work," *IEEE Signal Process. Mag.*, vol. 26, no. 3, pp. 91–95, May 2009.
- [4] M. Damen, H. El Gamal, and G. Caire, "On maximum-likelihood detection and the search for the closest lattice point," *IEEE Trans. Inf. Theory*, vol. 49, no. 10, pp. 2389–2402, Oct. 2003.
- [5] 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.
- [6] M. M. Mansour, S. Alex, and L. Jalloul, "Reduced complexity soft-output MIMO sphere detectors – Part II: Architectural optimizations," *IEEE Trans. Signal Process.*, vol. 62, no. 21, pp. 5521–5535, Nov. 2014.
- [7] K. Kato, K. Fukawa, R. Yamada, H. Suzuki, and S. Suyama, "Low-complexity MIMO signal detection employing multistream constrained search," *IEEE Trans. Veh. Technol.*, vol. 67, no. 2, pp. 1217–1230, Feb. 2018.
- [8] G. He, X. Zhang, and Z. Liang, "Algorithm and architecture of an efficient MIMO detector with cross-level parallel tree-search," *IEEE Trans. VLSI Syst.*, vol. 28, no. 2, pp. 467–479, Feb. 2020.
- [9] L. G. Barbero and J. 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.
- [10] 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. on Circuits and Systems (ISCAS)*, Island of Kos, Greece, May 2006, pp. 1151–1154.
- [11] C. Studer and H. Bölcskei, "Soft-input soft-output single tree-search sphere decoding," *IEEE Trans. Inf. Theory*, vol. 56, no. 10, pp. 4827–4842, Oct. 2010.
- [12] E. Larsson and J. Jaldén, "Fixed-complexity soft MIMO detection via partial marginalization," *IEEE Trans. Signal Process.*, vol. 56, no. 8, pp. 3397–3407, Aug. 2008.

- [13] D. Persson and E. Larsson, "Partial marginalization soft MIMO detection with higher order constellations," *IEEE Trans. Signal Process.*, vol. 59, no. 1, pp. 453–458, Jan. 2011.
- [14] M. Siti and M. P. Fitz, "A novel soft-output layered orthogonal lattice detector for multiple antenna communications," in *Proc. IEEE Int. Conf. Commun. (ICC)*, vol. 4, Istanbul, Turkey, Jun. 2006, pp. 1686–1691.
- [15] M. M. Mansour, "A near-ML MIMO subspace detection algorithm," *IEEE Signal Process. Lett.*, vol. 22, no. 4, pp. 408–412, Apr. 2015.
- [16] M. M. Mansour, S. Alex, and L. Jalloul, "Reduced complexity soft-output MIMO sphere detectors – Part I: Algorithmic optimizations," *IEEE Trans. Signal Process.*, vol. 62, no. 21, pp. 5505–5520, Nov. 2014.
- [17] F. Rusek and A. Prlja, "Optimal channel shortening for MIMO and ISI channels," *IEEE Trans. Wireless Commun.*, vol. 11, no. 2, pp. 810–818, Feb. 2012.
- [18] S. Hu and F. Rusek, "A soft-output MIMO detector with achievable information rate based partial marginalization," *IEEE Trans. Signal Process.*, vol. 65, no. 6, pp. 1622–1637, Mar. 2017.
- [19] P. Yang and H. Yang, "Optimal linear detection for MIMO systems with finite constellation inputs," *IEEE Signal Process. Lett.*, vol. 26, no. 4, pp. 612–616, Apr. 2019.
- [20] T. K. Moon, *Error Correction Coding: Mathematical Methods and Algorithms*. New York: Wiley, 2005.
- [21] G.-H. Golub and C.-F. Van Loan, *Matrix Computations*, 4th ed. Baltimore, MD: Johns Hopkins Univ. Press, 2013.
- [22] F. Lemeire, "Bounds for condition numbers of triangular and trapezoid matrices," *BIT Numerical Mathematics*, vol. 15, pp. 58–64, Mar. 1975.
- [23] A. K. Lenstra, H. Lenstra, Jr., and L. Lovász, "Factoring polynomials with rational coefficients," *Mathematische Annalen*, vol. 261, no. 4, pp. 515–534, Dec. 1982.
- [24] 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.
- [25] S. Lyu, J. Wen, J. Weng, and C. Ling, "On low-complexity lattice reduction algorithms for large-scale MIMO detection: The blessing of sequential reduction," *IEEE Trans. Signal Process.*, vol. 68, pp. 257–269, 2020.
- [26] D. Arnold, H.-A. Loeliger, P. Vontobel, W. Zeng, and A. Kavčić, "Simulation-based computation of information rates for channels with memory," *IEEE Trans. Inf. Theory*, vol. 52, no. 8, pp. 3498–3508, Aug. 2006.
- [27] F. Zhang, *Matrix Theory: Basic Results and Techniques*, 2nd ed. New York: Springer-Verlag, 2011.
- [28] E. Jessup and D. Sorensen, "A parallel algorithm for computing the singular value decomposition of a matrix," *SIAM Journal on Matrix Analysis and Applications*, vol. 15, no. 2, pp. 530–548, Mar. 1994.
- [29] B. Hassibi, "An efficient square-root algorithm for BLAST," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Process. (ICASSP)*, vol. 2, Istanbul, Turkey, Jun. 2000, pp. 737–740.
- [30] M. M. Mansour and L. Jalloul, "Optimized configurable architectures for scalable soft-input soft-output MIMO detectors with 256-QAM," *IEEE Trans. Signal Process.*, vol. 63, no. 18, pp. 4969–4984, Sep. 2015.
- [31] H. Srieddeen, M. M. Mansour, and A. Chehab, "Large MIMO detection schemes based on channel puncturing: Performance and complexity analysis," *IEEE Trans. Commun.*, vol. 66, no. 6, pp. 2421–2436, Jun. 2018.
- [32] M. M. Mansour, "A low-complexity MIMO subspace detection algorithm," *EURASIP Journal on Wireless Communications and Networking*, vol. 2015, no. 1, pp. 1–11, 2015.
- [33] D. Wübben, R. Böhnke, J. Rinas, V. Kühn, and K. D. Kammeyer, "Efficient algorithm for decoding layered space-time codes," *Electron. Lett.*, vol. 37, no. 22, pp. 1348–1350, Oct. 2001.



Mohammad M. Mansour (S'97-M'03-SM'08) received the B.E. (Hons.) and the M.E. degrees in computer and communications engineering from the American University of Beirut (AUB), Beirut, Lebanon, in 1996 and 1998, respectively, and the M.S. degree in mathematics and the Ph.D. degree in electrical engineering from the University of Illinois at Urbana–Champaign (UIUC), Champaign, IL, USA, in 2002 and 2003, respectively.

He was a Visiting Researcher at Qualcomm, San Jose, CA, USA, in summer of 2016, where he worked on baseband receiver architectures for the IEEE 802.11ax standard. He was a Visiting Researcher at Broadcom, Sunnyvale, CA, USA, from 2012 to 2014, where he worked on the physical layer SoC architecture and algorithm development for LTE-Advanced baseband receivers. He was on research leave with Qualcomm Flarion Technologies in Bridgewater, NJ, USA, from 2006 to 2008, where he worked on modem design and implementation for 3GPP-LTE, 3GPP2-UMB, and peer-to-peer wireless networking physical layer SoC architecture and algorithm development. He was a Research Assistant at the Coordinated Science Laboratory (CSL), UIUC, from 1998 to 2003. He worked at National Semiconductor Corporation, San Francisco, CA, with the Wireless Research group in 2000. He was a Research Assistant with the Department of Electrical and Computer Engineering, AUB, in 1997, and a Teaching Assistant in 1996. He joined as a faculty member with the Department of Electrical and Computer Engineering, AUB, in 2003, where he is currently a tenured full-Professor and Chairperson. He leads the COMNETIC research group whose focus is on fundamental research spanning inter-related core areas in information processing and machine learning, wireless communications, and VLSI systems. His research interests are in the area of energy-efficient and high-performance VLSI circuits, architectures, algorithms, and systems for communications, signal processing, and computing applications.

Prof. Mansour is a member of the Design and Implementation of Signal Processing Systems (DISPS) Technical Committee Advisory Board of the IEEE Signal Processing Society. He served as a member of the DISPS Technical Committee from 2006 to 2013. He served as an Associate Editor for IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II (TCAS-II) from 2008 to 2013, as an Associate Editor for the IEEE SIGNAL PROCESSING LETTERS from 2012 to 2016, and as an Associate Editor of the IEEE TRANSACTIONS ON VLSI SYSTEMS from 2011 to 2016. He served as the Technical Co-Chair of the IEEE Workshop on Signal Processing Systems in 2011, and as a member of the Technical Program Committee of various international conferences and workshops. He was the recipient of the PHI Kappa PHI Honor Society Award twice in 2000 and 2001, and the recipient of the Hewlett Foundation Fellowship Award in 2006. He has seven issued U.S. patents.

SUPPLEMENT S1

PROOF OF LEMMA 5

First, we can assume without any loss of generality that both \mathbf{U} and \mathbf{V} are lower-triangular matrices with real and positive diagonal entries. Otherwise, let $\mathbf{U} = \mathbf{Q}_u \mathbf{L}_u$ be the QL decomposition of \mathbf{U} and $\mathbf{V}\mathbf{V}^\dagger = \mathbf{L}_v \mathbf{L}_v^\dagger$ be the Cholesky factorization of $\mathbf{V}\mathbf{V}^\dagger$, where \mathbf{L}_u and \mathbf{L}_v are lower-triangular matrices with real and positive diagonal entries. Then

$$\begin{aligned} f(\mathbf{U}, \mathbf{V}) &= \ln \det(\mathbf{U}\mathbf{U}^\dagger) - \text{Tr}((\mathbf{U}\mathbf{V})(\mathbf{U}\mathbf{V})^\dagger) \\ &= \ln \det(\mathbf{Q}_u \mathbf{L}_u \mathbf{L}_u^\dagger \mathbf{Q}_u^\dagger) - \text{Tr}((\mathbf{Q}_u \mathbf{L}_u)(\mathbf{L}_v \mathbf{L}_v^\dagger)(\mathbf{L}_u^\dagger \mathbf{Q}_u^\dagger)) \\ &= \ln \det(\mathbf{L}_u \mathbf{L}_u^\dagger) - \text{Tr}((\mathbf{L}_u \mathbf{L}_v)(\mathbf{L}_u \mathbf{L}_v)^\dagger) \\ &= f(\mathbf{L}_u, \mathbf{L}_v). \end{aligned}$$

Henceforth, we assume that both $\mathbf{U} = [u_{kj}]$ and $\mathbf{V} = [v_{kj}]$ are lower-triangular matrices with real and positive diagonal entries. Let $\tilde{\mathbf{u}}_k = [u_{k1} \ u_{k2} \ \cdots \ u_{k,k-1}]$ denote the row vector consisting of the first $k-1$ elements of the k th row of \mathbf{U} , and $\mathbf{u}_k = [\tilde{\mathbf{u}}_k \ u_{kk}]$. Let \mathbf{U}_k denote the leading principal matrix of \mathbf{U} of order k , and let $\tilde{\mathbf{U}}_k = \mathbf{U}_{k-1}$. The vectors $\tilde{\mathbf{v}}_k$, \mathbf{v}_k , and matrices \mathbf{V}_k , $\tilde{\mathbf{V}}_k$ are similarly defined for \mathbf{V} . Let $g(\mathbf{U}) \triangleq \ln \det(\mathbf{U}\mathbf{U}^\dagger)$ and $h(\mathbf{U}, \mathbf{V}) \triangleq \text{Tr}((\mathbf{U}\mathbf{V})(\mathbf{U}\mathbf{V})^\dagger)$.

To determine \mathbf{U}^{opt} , we compute $\frac{\partial}{\partial \mathbf{U}} f(\mathbf{U}, \mathbf{V})$ and set it to 0. We start by computing the trace $\text{Tr}((\mathbf{U}\mathbf{V})(\mathbf{U}\mathbf{V})^\dagger)$ first,

$$\begin{aligned} h &= \sum_{k=1}^N \begin{bmatrix} \tilde{\mathbf{u}}_k & u_{kk} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{V}}_k & \\ & v_{kk} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{V}}_k^\dagger & \tilde{\mathbf{v}}_k^\dagger \\ & v_{kk} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{u}}_k^\dagger \\ u_{kk} \end{bmatrix} \\ &= \sum_{k=1}^N \left\{ \tilde{\mathbf{u}}_k \tilde{\mathbf{V}}_k \tilde{\mathbf{V}}_k^\dagger \tilde{\mathbf{u}}_k^\dagger + 2u_{kk} \Re \left\{ \tilde{\mathbf{v}}_k \tilde{\mathbf{V}}_k^\dagger \tilde{\mathbf{u}}_k^\dagger \right\} + u_{kk}^2 (v_{kk}^2 + \tilde{\mathbf{v}}_k \tilde{\mathbf{v}}_k^\dagger) \right\} \end{aligned}$$

The problem then boils down to determining the unknowns $\tilde{\mathbf{u}}_k$ and u_{kk} that satisfy the required derivative condition. Since $\ln \det(\mathbf{U}\mathbf{U}^\dagger) = \sum_{k=1}^N \ln u_{kk}^2$ involves the diagonal elements u_{kk} only, we can start by determining $\tilde{\mathbf{u}}_k$ by setting the derivative of the trace term only $\frac{\partial h}{\partial \tilde{\mathbf{u}}_k}$ to 0. We obtain

$$\tilde{\mathbf{V}}_k \tilde{\mathbf{V}}_k^\dagger \tilde{\mathbf{u}}_k^\dagger + u_{kk} \tilde{\mathbf{V}}_k \tilde{\mathbf{v}}_k^\dagger = 0,$$

and hence

$$\tilde{\mathbf{u}}_k^{\text{opt}} = -u_{kk} \tilde{\mathbf{v}}_k \tilde{\mathbf{V}}_k^{-1}.$$

We next determine u_{kk} . Substituting back in the trace equation, we get

$$h|_{\tilde{\mathbf{u}}_k = \tilde{\mathbf{u}}_k^{\text{opt}}} = \sum_{k=1}^N u_{kk}^2 v_{kk}^2.$$

Now taking derivative with respect to u_{kk} , including the $\ln \det$ term, we have

$$\frac{\partial f}{\partial u_{kk}} = \frac{\partial}{\partial u_{kk}} \{ \ln(u_{kk}^2) - u_{kk}^2 v_{kk}^2 \} = \frac{2}{u_{kk}} - 2u_{kk} v_{kk}^2 = 0,$$

implying that $u_{kk}^{\text{opt}} = \frac{1}{v_{kk}}$. Therefore,

$$\mathbf{u}_k^{\text{opt}} = [\tilde{\mathbf{u}}_k^{\text{opt}} \ u_{kk}^{\text{opt}}] = [-v_{kk}^{-1} \tilde{\mathbf{v}}_k \tilde{\mathbf{V}}_k^{-1} \ v_{kk}^{-1}].$$

Next note that if we multiply $\mathbf{u}_k^{\text{opt}}$ by \mathbf{V}_k for any k we obtain

$$\mathbf{u}_k^{\text{opt}} \mathbf{V}_k = \begin{bmatrix} -v_{kk}^{-1} \tilde{\mathbf{v}}_k \tilde{\mathbf{V}}_k^{-1} & v_{kk}^{-1} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{V}}_k \\ \tilde{\mathbf{v}}_k \ v_{kk} \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{1 \times (k-1)} & 1 \end{bmatrix},$$

which implies that $\mathbf{U}^{\text{opt}} \mathbf{V} = \mathbf{I}$. Hence the optimal \mathbf{U} is the inverse of \mathbf{V} , $\mathbf{U}^{\text{opt}} = \mathbf{V}^{-1}$, and $f(\mathbf{U}^{\text{opt}}, \mathbf{V}) = -\sum_{k=1}^N \ln v_{kk}^2 - N$. ■

SUPPLEMENT TABLE TI
SUMMARY OF DECOMPOSITION AND PUNCTURING ALGORITHMS

Algorithm Scheme		Functionality	Properties
Alg. 1	QL	Decompose \mathbf{H} as $\mathbf{H} = \mathbf{Q}\mathbf{L}$; generate $\tilde{\mathbf{y}} = \mathbf{Q}^\dagger \mathbf{y}$	$\mathbf{Q}^\dagger \mathbf{Q} = \mathbf{I}$; \mathbf{L} lower-triangular, $\mathbf{L}(k, k) \in \mathcal{R}^+$
Alg. 2	QDL	Decompose \mathbf{H} as $\mathbf{H} = \tilde{\mathbf{Q}}\mathbf{D}\tilde{\mathbf{L}}$; generate $\tilde{\tilde{\mathbf{y}}} = \tilde{\mathbf{Q}}^\dagger \mathbf{y}$	Square-root free QL decomposition $\tilde{\mathbf{Q}}^\dagger \tilde{\mathbf{Q}} = \mathbf{D}^{-1}$; $\tilde{\mathbf{L}}$ unit lower-triangular
Alg. 3	WL	Puncture \mathbf{H} using \mathbf{W} as $\mathbf{W}^\dagger \mathbf{H} = \mathbf{L}_p$; generate $\mathbf{y}_p = \mathbf{W}^\dagger \mathbf{y}$	One-sided puncturing algorithm \mathbf{W} non-unitary but $\text{diag}(\mathbf{W}^\dagger \mathbf{W}) = \mathbf{I}$ \mathbf{L}_p punctured lower-triangular, $\mathbf{L}_p(k, k) \in \mathcal{R}^+$
Alg. 4	WDL	Puncture \mathbf{H} using $\tilde{\mathbf{W}}$ as $\tilde{\mathbf{W}}^\dagger \mathbf{H} = \tilde{\mathbf{L}}_p$; generate $\tilde{\mathbf{y}}_p = \tilde{\mathbf{W}}^\dagger \mathbf{y}$	One-sided square-root free puncturing algorithm $\tilde{\mathbf{W}}$ non-unitary but $\text{diag}(\tilde{\mathbf{W}}^\dagger \tilde{\mathbf{W}}) = \mathbf{D}^{-1}$ $\tilde{\mathbf{L}}_p$ punctured unit lower-triangular
Alg. 5	QLZy	Decompose \mathbf{H} as $\mathbf{H} = \mathbf{Q}\mathbf{L}\mathbf{Z}^{-1}$; generate $\tilde{\mathbf{y}} = \mathbf{Q}^\dagger \mathbf{y}$	QL decomposition with right reduction $\mathbf{Q}^\dagger \mathbf{Q} = \mathbf{I}$; \mathbf{L} lower-triangular, $\mathbf{L}(k, k) \in \mathcal{R}^+$ \mathbf{Z} unimodular with $\det \mathbf{Z} = 1$
Alg. 6	WLZ	Puncture \mathbf{H} using \mathbf{W}, \mathbf{Z} as $\mathbf{W}^\dagger \mathbf{H}\mathbf{Z} = \mathbf{L}_p$; generate $\mathbf{y}_p = \mathbf{W}^\dagger \mathbf{y}$	Two-sided puncturing algorithm \mathbf{W} non-unitary but $\text{diag}(\mathbf{W}^\dagger \mathbf{W}) = \mathbf{I}$ \mathbf{L}_p punctured lower-triangular, $\mathbf{L}_p(k, k) \in \mathcal{R}^+$ \mathbf{Z} unimodular with $\det \mathbf{Z} = 1$

SUPPLEMENT S2

QL DECOMPOSITION ALGORITHM

Alg. 1 Optimized thin QL decomposition algorithm

\triangleright Decompose \mathbf{H} as $\mathbf{H} = \mathbf{Q}\mathbf{L}$ and generate $\tilde{\mathbf{y}} = \mathbf{Q}^\dagger \mathbf{y}$
 \triangleright \mathbf{H} : Complex $M \times N$ matrix, $M \geq N$
 \triangleright \mathbf{y} : Complex $M \times 1$ column vector
 \triangleright \mathbf{Q} : $M \times N$ matrix with orthonormal columns; $\mathbf{Q}^\dagger \mathbf{Q} = \mathbf{I}_N$
 \triangleright \mathbf{L} : $N \times N$ lower-triangular matrix s.t. $\mathbf{L}(k, k) \in \mathcal{R}^+$
 \triangleright $\tilde{\mathbf{y}}$: $N \times 1$ such that $\tilde{\mathbf{y}} = \mathbf{Q}^\dagger \mathbf{y}$

```

function [ $\mathbf{Q}, \mathbf{L}, \tilde{\mathbf{y}}$ ] = QLy( $\mathbf{H}, \mathbf{y}$ )
     $\mathbf{Q} \leftarrow [\mathbf{y} \ \mathbf{H}]$   $\triangleright$  augment  $\mathbf{y}$  to  $\mathbf{H}$ 
     $\mathbf{L} \leftarrow \mathbf{0}_{N \times (N+1)}$ 
    for  $k = N+1 : -1 : 2$  do  $\triangleright$  index of current column
         $\mathbf{L}(k-1, k) \leftarrow \sqrt{\mathbf{Q}(:, k)^\dagger \mathbf{Q}(:, k)}$   $\triangleright$  diagonal element
         $\mathbf{Q}(:, k) \leftarrow \mathbf{Q}(:, k) / \mathbf{L}(k-1, k)$   $\triangleright$  normalize
        for  $j = k-1 : -1 : 1$  do  $\triangleright$  all other cols to its left
             $\mathbf{L}(k-1, j) \leftarrow \mathbf{Q}(:, k)^\dagger \mathbf{Q}(:, j)$ 
             $\mathbf{Q}(:, j) \leftarrow \mathbf{Q}(:, j) - \mathbf{L}(k-1, j) \mathbf{Q}(:, k)$ 
        end for
    end for
     $\mathbf{Q} \leftarrow \mathbf{Q}(:, 2:N+1)$   $\triangleright$  last  $N$  cols of augmented  $\mathbf{Q}$ 
     $\tilde{\mathbf{y}} \leftarrow \mathbf{L}(:, 1)$   $\triangleright$  first col of augmented  $\mathbf{L}$ 
     $\mathbf{L} \leftarrow \mathbf{L}(:, 2:N+1)$   $\triangleright$  last  $N$  cols of augmented  $\mathbf{L}$ 
end function

```

SUPPLEMENT S3

QDL DECOMPOSITION ALGORITHM

Alg. 2 Optimized QDL decomposition algorithm

- ▷ Decompose \mathbf{H} as $\mathbf{H} = \tilde{\mathbf{Q}}\mathbf{D}\tilde{\mathbf{L}}$ and generate $\tilde{\mathbf{y}} = \tilde{\mathbf{Q}}^\dagger\mathbf{y}$.
- ▷ If $[\tilde{\mathbf{Q}}, \tilde{\mathbf{L}}, \tilde{\mathbf{y}}] = \text{QDLy}(\mathbf{H}, \mathbf{y})$, then
- ▷ $\tilde{\mathbf{Q}} = \mathbf{Q}\mathbf{D}^{-1/2}$, $\tilde{\mathbf{L}} = \mathbf{D}^{-1/2}\mathbf{L}$, $\tilde{\mathbf{y}} = \mathbf{D}^{-1/2}\tilde{\mathbf{y}}$.
- ▷ \mathbf{H} : Complex $M \times N$ matrix, $M \geq N$
- ▷ \mathbf{y} : Complex $M \times 1$ column vector
- ▷ $\tilde{\mathbf{Q}}$: $M \times N$ matrix with orthogonal columns s.t. $\tilde{\mathbf{Q}}^\dagger\tilde{\mathbf{Q}} = \mathbf{D}^{-1}$
- ▷ \mathbf{D} : $N \times N$ diagonal matrix with real positive entries such that
- ▷ $\mathbf{D} = \text{diag}(\mathbf{L})^2$
- ▷ $\tilde{\mathbf{L}}$: $N \times N$ unit lower-triangular matrix; $\tilde{\mathbf{L}}(k, k) = 1$
- ▷ $\tilde{\mathbf{y}}$: $N \times 1$ such that $\tilde{\mathbf{y}} = \tilde{\mathbf{Q}}^\dagger\mathbf{y} = \mathbf{D}^{-1/2}\mathbf{Q}^\dagger\mathbf{y} = \mathbf{D}^{-1/2}\tilde{\mathbf{y}}$

function $[\tilde{\mathbf{Q}}, \mathbf{D}, \tilde{\mathbf{L}}, \tilde{\mathbf{y}}] = \text{QDLy}(\mathbf{H}, \mathbf{y})$

$\tilde{\mathbf{Q}} \leftarrow [\mathbf{y} \ \mathbf{H}]$

$\mathbf{D} \leftarrow \mathbf{0}_{N \times N}$

$\tilde{\mathbf{L}} \leftarrow [\mathbf{0}_{N \times 1} \ \mathbf{I}_N]$

for $k = N+1 : -1 : 2$ **do**

$\mathbf{D}(k-1, k-1) \leftarrow \|\tilde{\mathbf{Q}}(:, k)\|^2$

for $j = k-1 : -1 : 1$ **do**

$\tilde{\mathbf{L}}(k-1, j) \leftarrow \tilde{\mathbf{Q}}(:, k)^\dagger \tilde{\mathbf{Q}}(:, j) / \mathbf{D}(k-1, k-1)$

$\tilde{\mathbf{Q}}(:, j) \leftarrow \tilde{\mathbf{Q}}(:, j) - \tilde{\mathbf{L}}(k-1, j)\tilde{\mathbf{Q}}(:, k)$

end for

$\tilde{\mathbf{Q}}(:, k) \leftarrow \tilde{\mathbf{Q}}(:, k) / \mathbf{D}(k-1, k-1)$

end for

$\tilde{\mathbf{Q}} \leftarrow \tilde{\mathbf{Q}}(:, 2 : N+1)$

$\tilde{\mathbf{y}} \leftarrow \tilde{\mathbf{L}}(:, 1)$

$\tilde{\mathbf{L}} \leftarrow \tilde{\mathbf{L}}(:, 2 : N+1)$

end function

- ▷ augment \mathbf{y} to \mathbf{H}
- ▷ normalizer diagonal matrix
- ▷ normalized augmented matrix
- ▷ index of current col
- ▷ diagonal element
- ▷ all other cols to its left

- ▷ last N cols of augmented $\tilde{\mathbf{Q}}$
- ▷ first col of augmented $\tilde{\mathbf{L}}$
- ▷ last N cols of augmented $\tilde{\mathbf{L}}$

SUPPLEMENT S4

WL DECOMPOSITION ALGORITHM

Alg. 3 Optimized WL decomposition algorithm

▷ Generate \mathbf{W} s.t. $\mathbf{W}^\dagger \mathbf{H} = \mathbf{L}_p$, $\mathbf{W}^\dagger \mathbf{y} = \mathbf{y}_p$, $\text{diag}(\mathbf{W}^\dagger \mathbf{W}) = \mathbf{I}_N$

▷ \mathbf{H} : Complex $M \times N$ matrix, $M \geq N$

▷ \mathbf{y} : Complex $M \times 1$ column vector

▷ ν : puncturing order

▷ \mathbf{L}_p : $N \times N$ punctured lower-triangular matrix; $\mathbf{L}_p(k, k) \in \mathcal{R}^+$

▷ \mathbf{y}_p : $N \times 1$ such that $\mathbf{y}_p = \mathbf{W}^\dagger \mathbf{y}$

▷ \mathbf{W} : $M \times N$ puncturing matrix such that $\text{diag}(\mathbf{W}^\dagger \mathbf{W}) = \mathbf{I}_N$

▷

▷ Note: \mathbf{W}^\dagger punctures \mathbf{H} ; in manuscript, \mathbf{W}_p punctures \mathbf{L} .

▷ The two schemes are related as follows:

$$\mathbf{W}^\dagger (\mathbf{y} - \mathbf{H}\mathbf{x}) = \mathbf{W}_p \mathbf{Q}^\dagger (\mathbf{y} - \mathbf{H}\mathbf{x}) = \mathbf{W}_p (\tilde{\mathbf{y}} - \mathbf{L}\mathbf{x})$$

$$\mathbf{W}^\dagger = \mathbf{W}_p \mathbf{Q}^\dagger, \quad \mathbf{W}_p = \mathbf{W}^\dagger \mathbf{Q}$$

▷ Also, $\mathbf{W}^\dagger \mathbf{Q}\mathbf{Q}^\dagger = \mathbf{W}^\dagger$ even though $\mathbf{Q}\mathbf{Q}^\dagger \neq \mathbf{I}$ for $M > N$. This is because the rows of \mathbf{Q}^\dagger and the cols of $(\mathbf{Q}\mathbf{Q}^\dagger - \mathbf{I})$ are orthogonal so that $\mathbf{Q}^\dagger (\mathbf{Q}\mathbf{Q}^\dagger - \mathbf{I}) = \mathbf{0}$. Hence any matrix right-multiplied by \mathbf{Q}^\dagger would have rows orthogonal to $(\mathbf{Q}\mathbf{Q}^\dagger - \mathbf{I})$. Thus $\mathbf{W}^\dagger \mathbf{Q}\mathbf{Q}^\dagger - \mathbf{W}^\dagger = \mathbf{W}^\dagger (\mathbf{Q}\mathbf{Q}^\dagger - \mathbf{I}) = \mathbf{W}_p \mathbf{Q}^\dagger (\mathbf{Q}\mathbf{Q}^\dagger - \mathbf{I}) = \mathbf{0}$.

▷

function $[\mathbf{L}_p, \mathbf{y}_p, \mathbf{W}] = \text{WL}(\mathbf{H}, \mathbf{y}, \nu)$

$[\mathbf{Q}, \mathbf{L}, \tilde{\mathbf{y}}] \leftarrow \mathbf{QLy}(\mathbf{H}, \mathbf{y})$

$\mathbf{W} \leftarrow \mathbf{Q}$, $\mathbf{L}_p \leftarrow [\tilde{\mathbf{y}} \ \mathbf{L}]$

for $k = \nu + 2 : N$ **do**

for $j = \nu + 1 : k - 1$ **do**

$\alpha \leftarrow \mathbf{L}_p(k, j + 1) / \mathbf{L}_p(j, j + 1)$

$\mathbf{W}(:, k) \leftarrow \mathbf{W}(:, k) - \alpha \mathbf{W}(:, j)$

$\mathbf{L}_p(k, 1:j+1) \leftarrow \mathbf{L}_p(k, 1:j+1) - \alpha \mathbf{L}_p(j, 1:j+1)$

end for

$\mathbf{L}_p(k, 1:k+1) \leftarrow \mathbf{L}_p(k, 1:k+1) / \|\mathbf{W}(:, k)\|$

$\mathbf{W}(:, k) \leftarrow \mathbf{W}(:, k) / \|\mathbf{W}(:, k)\|$

end for

$\mathbf{y}_p \leftarrow \mathbf{L}_p(:, 1)$

$\mathbf{L}_p \leftarrow \mathbf{L}_p(:, 2:N+1)$

end function

▷ QL dec.; here $\tilde{\mathbf{y}} = \mathbf{Q}^\dagger \mathbf{y}$, $\mathbf{Q}^\dagger \mathbf{Q} = \mathbf{I}_N$

▷ Augment $\tilde{\mathbf{y}}$ to \mathbf{L}

▷ Gaussian elimination

▷ col index to puncture

▷ first col of augmented \mathbf{L}_p

▷ last N cols of augmented \mathbf{L}_p

▷ $\mathbf{W}_p = \mathbf{W}^\dagger \mathbf{Q}$

SUPPLEMENT S5

WDL DECOMPOSITION ALGORITHM

Alg. 4 Square-root-free WDL decomposition algorithm

- ▷ Square-root free version of $\text{WL}()$ in Alg. 3
- ▷ Generate \mathbf{D} and $\tilde{\mathbf{W}}$ such that $\tilde{\mathbf{W}}^\dagger \mathbf{H} = \tilde{\mathbf{L}}_p$, $\tilde{\mathbf{W}}^\dagger \mathbf{y} = \tilde{\mathbf{y}}_p$, and
- ▷ $\text{diag}(\tilde{\mathbf{W}}^\dagger \tilde{\mathbf{W}}) = \mathbf{D}^{-1}$.
- ▷ \mathbf{H} : Complex $M \times N$ matrix, $M \geq N$
- ▷ \mathbf{y} : Complex $M \times 1$ column vector
- ▷ ν : puncturing order
- ▷ $\tilde{\mathbf{L}}_p$: $N \times N$ punctured unit lower-triangular matrix; $\tilde{\mathbf{L}}_p(k, k) = 1$
- ▷ $\tilde{\mathbf{y}}_p$: $N \times 1$ such that $\tilde{\mathbf{y}}_p = \tilde{\mathbf{W}}^\dagger \mathbf{y}$
- ▷ \mathbf{D} : $N \times N$ diagonal matrix with real positive entries such that
- ▷ $\mathbf{D} = \text{diag}(\mathbf{L}_p)^2$
- ▷ $\tilde{\mathbf{W}}$: $M \times N$ puncturing matrix such that $\text{diag}(\tilde{\mathbf{W}}^\dagger \tilde{\mathbf{W}}) = \mathbf{D}^{-1}$

▷ Note: $\tilde{\mathbf{W}}^\dagger$ punctures \mathbf{H} to form $\tilde{\mathbf{L}}_p = \tilde{\mathbf{W}}^\dagger \mathbf{H}$. In manuscript, $\tilde{\mathbf{W}}_p$ punctures $\mathbf{D}\tilde{\mathbf{L}} = \mathbf{D}^{1/2}\mathbf{L}$ to form $\mathbf{L}_p = \tilde{\mathbf{W}}_p \mathbf{D}\tilde{\mathbf{L}}$. These quantities are related as follows:

▷ If $[\tilde{\mathbf{Q}}, \mathbf{D}, \tilde{\mathbf{L}}, \tilde{\mathbf{y}}] = \text{QDLy}(\mathbf{H}, \mathbf{y})$, then:

$$\tilde{\mathbf{Q}} = \mathbf{Q}\mathbf{D}^{-1/2}, \quad \tilde{\mathbf{L}} = \mathbf{D}^{-1/2}\mathbf{L}, \quad \tilde{\mathbf{y}} = \mathbf{D}^{-1/2}\tilde{\mathbf{y}}$$

▷ If $[\mathbf{L}_p, \mathbf{y}_p, \mathbf{W}] = \text{WL}(\mathbf{H}, \mathbf{y}, \nu)$, then:

$$\begin{aligned} \mathbf{L}_p &= \mathbf{W}^\dagger \mathbf{H}, \quad \mathbf{y}_p = \mathbf{W}^\dagger \mathbf{y} \\ \tilde{\mathbf{W}} &= \mathbf{W}\mathbf{D}^{-1/2}, \quad \tilde{\mathbf{L}}_p = \mathbf{D}^{-1/2}\mathbf{L}_p, \quad \mathbf{D} = \text{diag}(\mathbf{L}_p)^2, \quad \tilde{\mathbf{y}}_p = \mathbf{D}^{-1/2}\mathbf{y}_p \\ \mathbf{W}^\dagger &= \tilde{\mathbf{W}}_p \mathbf{Q}^\dagger = \tilde{\mathbf{W}}_p \mathbf{D}\tilde{\mathbf{Q}}^\dagger, \quad \tilde{\mathbf{W}}^\dagger = \mathbf{D}^{-1/2}\tilde{\mathbf{W}}_p \mathbf{D}\tilde{\mathbf{Q}}^\dagger \end{aligned}$$

function $[\tilde{\mathbf{L}}_p, \tilde{\mathbf{y}}_p, \mathbf{D}, \tilde{\mathbf{W}}] = \text{WDL}(\mathbf{H}, \mathbf{y}, \nu)$

$[\tilde{\mathbf{Q}}, \mathbf{D}, \tilde{\mathbf{L}}, \tilde{\mathbf{y}}] \leftarrow \text{QDLy}(\mathbf{H}, \mathbf{y})$

$\tilde{\mathbf{W}} \leftarrow \tilde{\mathbf{Q}}$

$\tilde{\mathbf{L}}_p \leftarrow [\tilde{\mathbf{y}} \quad \tilde{\mathbf{L}}]$

for $k = \nu + 2 : N$ **do**

for $j = \nu + 1 : k - 1$ **do**

$\alpha \leftarrow \tilde{\mathbf{L}}_p(k, j + 1)$

$\tilde{\mathbf{W}}(:, k) \leftarrow \tilde{\mathbf{W}}(:, k) - \alpha^\dagger \tilde{\mathbf{W}}(:, j)$

$\tilde{\mathbf{L}}_p(k, 1:j+1) \leftarrow \tilde{\mathbf{L}}_p(k, 1:j+1) - \alpha \tilde{\mathbf{L}}_p(j, 1:j+1)$

end for

$\mathbf{D}(k, k) \leftarrow 1 / \|\tilde{\mathbf{W}}(:, k)\|^2$

end for

$\tilde{\mathbf{y}}_p \leftarrow \tilde{\mathbf{L}}_p(:, 1)$

$\tilde{\mathbf{L}}_p \leftarrow \tilde{\mathbf{L}}_p(:, 2:N+1)$

end function

▷ QDL dec.; here $\tilde{\mathbf{y}} = \tilde{\mathbf{Q}}^\dagger \mathbf{y}$

▷ copy in case $\tilde{\mathbf{Q}}$ is needed

▷ Augment $\tilde{\mathbf{y}}$ to $\tilde{\mathbf{L}}$

▷ Gaussian elimination

▷ col index to puncture

▷ first col of augmented $\tilde{\mathbf{L}}_p$

▷ last N cols of augmented $\tilde{\mathbf{L}}_p$

SUPPLEMENT S6

QLZ DECOMPOSITION ALGORITHM

Alg. 5 QLZ decomposition algorithm with right reduction

- ▷ Decompose \mathbf{H} as $\mathbf{HZ} = \mathbf{QL}$ or $\mathbf{H} = \mathbf{QLZ}^{-1}$
- ▷ \mathbf{H} : Complex $M \times N$ matrix, $M \geq N$
- ▷ \mathbf{y} : Complex $M \times 1$ column vector
- ▷ c : reduction control parameter
- ▷ \mathbf{Q} : $M \times N$ matrix with orthonormal columns; $\mathbf{Q}^\dagger \mathbf{Q} = \mathbf{I}_N$
- ▷ \mathbf{L} : $N \times N$ lower-triangular matrix satisfying reduction conditions
- ▷ 1) $|\Re\{L(k, j)\}| \leq 2^{-(c+1/2)} L(k, k)$ for all $j, k : j < k$
- ▷ 2) $|\Im\{L(k, j)\}| \leq 2^{-(c+1/2)} L(k, k)$ for all $j, k : j < k$
- ▷ Note: $\mathbf{L}(k, k) \in \mathcal{R}^+$
- ▷ $\tilde{\mathbf{y}} = \mathbf{Q}^\dagger \mathbf{y}$
- ▷ \mathbf{Z} : $N \times N$ unimodular matrix with $\det \mathbf{Z} = 1$
- ▷ \mathbf{Z}^{-1} : inverse of \mathbf{Z} ; $N \times N$ unimodular matrix with $\det \mathbf{Z}^{-1} = 1$

function $[\mathbf{Q}, \mathbf{L}, \tilde{\mathbf{y}}, \mathbf{Z}, \mathbf{Z}^{-1}] = \text{QLZy}(\mathbf{H}, \mathbf{y}, c)$

```

 $\mathbf{Z} \leftarrow \mathbf{I}_N, \mathbf{Z}^{-1} \leftarrow \mathbf{I}_N$ 
 $[\mathbf{Q}, \mathbf{L}, \tilde{\mathbf{y}}] \leftarrow \text{QLy}(\mathbf{H}, \mathbf{y})$ 
for  $k=2:N$  do
    for  $j=1:k-1$  do
         $\zeta \leftarrow 2^{-c} \lfloor 2^c \frac{L(k,j)}{L(k,k)} \rfloor$ 
        if  $\zeta \neq 0$  then
             $\mathbf{L}(k:N, j) \leftarrow \mathbf{L}(k:N, j) - \zeta \cdot \mathbf{L}(k:N, k)$ 
             $\mathbf{Z}(k:N, j) \leftarrow \mathbf{Z}(k:N, j) - \zeta \cdot \mathbf{Z}(k:N, k)$ 
             $\mathbf{Z}^{-1}(k, 1:j) \leftarrow \mathbf{Z}^{-1}(k, 1:j) + \zeta \cdot \mathbf{Z}^{-1}(j, 1:j)$ 
        end if
    end for
end for
end function

```

- ▷ Gauss matrix and its inverse
- ▷ QL-decompose
- ▷ row index
- ▷ col index
- ▷ Reduction factor

SUPPLEMENT S7

OPTIMIZED TWO-SIDED WLZ DECOMPOSITION ALGORITHM

Alg. 6 Two-sided WLZ decomposition algorithm

- ▷ Generate \mathbf{W}, \mathbf{Z} such that $\mathbf{L}_p = \mathbf{W}^\dagger \mathbf{H} \mathbf{Z}$
- ▷ If $\mathbf{H} = \mathbf{Q} \mathbf{L}$, then $\mathbf{L}_z \triangleq \mathbf{L} \mathbf{Z}$ satisfies the reduction conditions
- ▷ 1) $|\Re\{\mathbf{L}_z(k, j)\}| \leq 2^{-(c+1/2)} \mathbf{L}_z(k, k)$ for all $j, k : \nu < j < k$
- ▷ 2) $|\Im\{\mathbf{L}_z(k, j)\}| \leq 2^{-(c+1/2)} \mathbf{L}_z(k, k)$ for all $j, k : \nu < j < k$
- ▷ \mathbf{H} : Complex $M \times N$ matrix, $M \geq N$
- ▷ \mathbf{y} : Complex $M \times 1$ column vector
- ▷ ν : puncturing order
- ▷ c : reduction control parameter
- ▷ \mathbf{L}_p : $N \times N$ punctured lower-triangular matrix; $\mathbf{L}_p(k, k) \in \mathcal{R}^+$
- ▷ $\mathbf{y}_p = \mathbf{W}^\dagger \mathbf{y}$
- ▷ \mathbf{W} : $M \times N$ matrix such that $\text{diag}(\mathbf{W}^\dagger \mathbf{W}) = \mathbf{I}_N$
- ▷ \mathbf{Z} : $N \times N$ unimodular matrix with $\det \mathbf{Z} = 1$
- ▷ \mathbf{Z}^{-1} : inverse of \mathbf{Z} ; $N \times N$ unimodular matrix with $\det \mathbf{Z}^{-1} = 1$
- ▷ _____
- ▷ Note: \mathbf{W}^\dagger punctures \mathbf{H} ; in manuscript, \mathbf{W}_p in (31) punctures \mathbf{L} . The two matrices are related as $\mathbf{W}^\dagger = \mathbf{W}_p \mathbf{Q}^\dagger$.
- ▷ _____

function $[\mathbf{L}_p, \mathbf{y}_p, \mathbf{W}, \mathbf{Z}, \mathbf{Z}^{-1}] = \text{WLZ}(\mathbf{H}, \mathbf{y}, \nu, c)$

$[\mathbf{W}, \mathbf{L}_p, \mathbf{y}_p] \leftarrow \text{QLy}(\mathbf{H}, \mathbf{y})$

for $k = \nu + 2 : N$ **do**

for $j = \nu + 1 : k - 1$ **do**

▷ Reduction step

$\zeta \leftarrow 2^{-c} \lfloor 2^c \frac{\mathbf{L}(k, j)}{\mathbf{L}(k, k)} \rfloor$

if $\zeta \neq 0$ **then**

$\mathbf{L}(k:N, j) \leftarrow \mathbf{L}(k:N, j) - \zeta \cdot \mathbf{L}(k:N, k)$

$\mathbf{Z}(k:N, j) \leftarrow \mathbf{Z}(k:N, j) - \zeta \cdot \mathbf{Z}(k:N, k)$

$\mathbf{Z}^{-1}(k, 1:j) \leftarrow \mathbf{Z}^{-1}(k, 1:j) + \zeta \cdot \mathbf{Z}^{-1}(j, 1:j)$

end if

▷ Elimination step

$\omega \leftarrow \mathbf{L}_p(k, j) / \mathbf{L}_p(j, j)$

$\mathbf{W}(:, k) \leftarrow \mathbf{W}(:, k) - \omega^\dagger \cdot \mathbf{W}(:, j)$

$\mathbf{L}_p(k, 1:j) \leftarrow \mathbf{L}_p(k, 1:j) - \omega \cdot \mathbf{L}_p(j, 1:j)$

$\mathbf{y}_p(k) \leftarrow \mathbf{y}_p(k) - \omega \cdot \mathbf{y}_p(j)$

▷ update \mathbf{y}_p

end for

$\mathbf{L}_p(k, 1:k) \leftarrow \mathbf{L}_p(k, 1:k) / \|\mathbf{W}(:, k)\|$

▷ normalize

$\mathbf{y}_p(k) \leftarrow \mathbf{y}_p(k) / \|\mathbf{W}(:, k)\|$

$\mathbf{W}(:, k) \leftarrow \mathbf{W}(:, k) / \|\mathbf{W}(:, k)\|$

end for

end function

SUPPLEMENT TABLE TII
SUMMARY OF DETECTION ALGORITHMS

Algorithm	Decomp. Scheme	Channel	Metric	Marginalization	Tree	Metric Update
Alg. 7 WLdetector	1-sided WL ν parents	left-punctured	$\mu_p = -\frac{1}{N_0} \ \mathbf{y}_p - \mathbf{L}_p \mathbf{x}\ ^2$	LS on leaves	multi-tree: N/ν trees all child nodes are leaves	local within tree only metrics not preserved with col permutations
Alg. 8 WLZdetector	2-sided WLZ ν parents reduction param. c	left-punctured right-reduced	$\mu_{pz} = -\frac{1}{N_0} \ \mathbf{y}_p - \mathbf{L}_p \mathbf{Z}^{-1} \mathbf{x}\ ^2$	LS on leaves	multi-tree: N/ν trees all child nodes are leaves	global across all trees metrics <i>almost</i> preserved with col permutations as c increases
Alg. 9 AWDLdetector	1-sided WDL ν parents	augmented left-punctured	$\mu_{ap} = \frac{1}{E_s} \ \mathbf{x}\ ^2 - \ \tilde{\mathbf{y}}_{ap} - \tilde{\mathbf{L}}_{ap} \mathbf{x}\ _D^2$	LS on leaves	multi-tree: N/ν trees all child nodes are leaves	local within tree only metrics not preserved with col permutations
Alg. 10 AWDLXdetector	1-sided WDL fixed $\nu=2$ parents window size η	augmented left-punctured	$\mu_{ap} = \frac{1}{E_s} \ \mathbf{x}\ ^2 - \ \tilde{\mathbf{y}}_{ap} - \tilde{\mathbf{L}}_{ap} \mathbf{x}\ _D^2$	enumerate on parent 1 ZF+window on parent 2 ZF-DF on leaves	multi-tree: $2 \times N/2$ trees 2 trees per parent pair all child nodes are leaves	global between parent tree pairs metrics not preserved across tree pairs
Alg. 11 LORDdetector	1-sided QLy ν parents	true	$\mu = -\frac{1}{N_0} \ \tilde{\mathbf{y}} - \mathbf{L} \mathbf{x}\ ^2$	ZF-DF on child nodes	multi-tree: N/ν trees	local within tree only
Alg. 12 LORDXdetector	1-sided QLy ν parents	true	$\mu = -\frac{1}{N_0} \ \tilde{\mathbf{y}} - \mathbf{L} \mathbf{x}\ ^2$	ZF-DF on child nodes	multi-tree: N/ν trees	global across trees metrics preserved with col permutations

SUPPLEMENT S8

WLD-BASED MIMO DETECTION ALGORITHM

Alg. 7 One-sided WLD MIMO detection algorithm

▷ Perform soft-output MIMO detection by puncturing \mathbf{H} using 1-sided WL() decomposition scheme of Alg. 3. Process ν parent layers at a time. In each run, layers are permuted so that a new group of ν symbols are chosen as parent symbols. N/ν independent runs are performed. Metrics of **parent layer symbols only** are updated in each run. This is because, for every layer ordering of \mathbf{H} , \mathbf{W}_p^\dagger changes and is not unitary. Hence Euclidean distance metrics of the form $\|\mathbf{W}_p^\dagger(\mathbf{y} - \mathbf{H}\mathbf{x})\| = \|\mathbf{y}_p - \mathbf{L}_p\mathbf{x}\|$ are not preserved when the columns of \mathbf{H} are permuted.

▷ \mathbf{H} : Complex $M \times N$ matrix, $M \geq N$

▷ \mathbf{y} : Complex $M \times 1$ column vector

▷ N_0 : noise variance

▷ \mathcal{X} : set of Q modulation constellation symbols; $|\mathcal{X}| = Q = 2^q$

▷ ν : puncturing order (assume N is a multiple of ν)

▷ Λ : $qN \times 1$ bit LLR vector

▷ Note: Distance computation on line 14 is expressed in this form for brevity. It can be simplified since \mathbf{L}_p is punctured and sparse.

```

1: function  $\Lambda = \text{WLDetector}(\mathbf{H}, \mathbf{y}, N_0, \mathcal{X}, \nu)$ 
2:    $Q \leftarrow |\mathcal{X}|, q \leftarrow \log_2 Q$ 
3:    $\mathbf{X} \leftarrow$  all  $\nu \times 1$  vectors in  $\mathcal{X}$ 
4:    $\mathbf{x} \leftarrow \mathbf{0}_{N \times 1}$ 
5:    $\mu_1, \mu_0 \leftarrow -\infty_{qN \times 1}$ 
6:   for  $t = 1 : N/\nu$  do
7:      $\pi \leftarrow [\nu(t-1)+1 : N, 1 : \nu(t-1)]$ 
8:      $[\mathbf{L}_p, \mathbf{y}_p, \sim] \leftarrow \text{WL}(\mathbf{H}(:, \pi), \mathbf{y}, \nu)$ 
9:     for  $j = 1 : Q^\nu$  do
10:       $\mathbf{x}(1:\nu) \leftarrow \mathbf{X}(1:\nu, j)$ 
11:      for  $i = \nu+1 : N$  do
12:         $\mathbf{x}(i) \leftarrow \left\lfloor \frac{\mathbf{y}_p(i) - \mathbf{L}_p(i, 1:\nu)\mathbf{x}(1:\nu)}{\mathbf{L}_p(i, i)} \right\rfloor$ 
13:      end for
14:       $\mu \leftarrow -\|\mathbf{y}_p - \mathbf{L}_p\mathbf{x}\|^2$ 
15:       $\mathbf{b} \leftarrow \text{binary}(\mathbf{x}(1:\nu))$ 
16:      for  $k = 1 : q\nu$  do
17:        if  $\mathbf{b}(k) = 1$  then
18:           $\mu_1(q\nu(t-1) + k) \leftarrow \max\{\mu_1(q\nu(t-1) + k), \mu\}$ 
19:        else
20:           $\mu_0(q\nu(t-1) + k) \leftarrow \max\{\mu_0(q\nu(t-1) + k), \mu\}$ 
21:        end if
22:      end for
23:    end for
24:  end for
25:   $\Lambda \leftarrow (\mu_1 - \mu_0)/N_0$ 
26: end function

```

▷ $\nu \times Q^\nu$ symbol matrix
▷ $N \times 1$ column symbol vector
▷ $qN \times 1$ metric vec. initialized to $-\infty$
▷ process ν parent layers at a time
▷ col permutation
▷ permuted cols
▷ loop over all $\nu \times 1$ vectors in \mathcal{X}^ν
▷ ν parent layer symbols
▷ $N - \nu$ child layer symbols
▷ slice
▷ metric using punctured \mathbf{L}_p
▷ $q\nu \times 1$ binary representation
▷ metrics for $q\nu$ parent symbol bits
▷ k loop
▷ j loop
▷ t loop
▷ $qN \times 1$ vector of LLRs

SUPPLEMENT S9

WLZ-BASED MIMO DETECTION ALGORITHM

Alg. 8 Two-sided WLZ MIMO detection algorithm

▷ Perform soft-output MIMO detection by puncturing \mathbf{H} using 2-sided WLZ() decomposition scheme of Alg. 6. Process ν parent layers at a time. Each run detects a new group of ν symbols chosen as parent symbols. N/ν runs are performed. Metrics of **all layer symbols** are updated in each run. This approximation is possible in this case because of the right reduction step by \mathbf{Z} . For large c , $\mathbf{W}_{\text{ap}}\mathbf{W}_{\text{ap}}^\dagger \approx \mathbf{I}$ (i.e., almost unitary), and hence distance metrics of the form $\|\mathbf{W}_{\text{ap}}^\dagger(\mathbf{y}_a - \mathbf{H}_a\mathbf{x})\|^2 \propto \|\mathbf{W}_{\text{ap}}^\dagger\mathbf{L}_a(\mathbf{M}\mathbf{y} - \mathbf{x})\|^2$ are almost preserved when the columns of \mathbf{H} are permuted.

▷ \mathbf{H} : Complex $M \times N$ matrix, $M \geq N$

▷ \mathbf{y} : Complex $M \times 1$ column vector

▷ N_0 : noise variance

▷ \mathcal{X} : set of Q modulation constellation symbols; $|\mathcal{X}| = Q = 2^q$

▷ ν : puncturing order (assume N is a multiple of ν)

▷ c : reduction control parameter

▷ Λ : $qN \times 1$ bit LLR vector

▷ Note: Operation $\mathbf{L}_p\mathbf{Z}^{-1}$ on line 9 is simply integer addition and scaling operations by powers-of-2. Also, distance computation on line 15 is expressed in this form for brevity. It can be simplified since \mathbf{L}_z is punctured and sparse.

```

1: function  $\Lambda = \text{WLZdetector}(\mathbf{H}, \mathbf{y}, N_0, \mathcal{X}, \nu, c)$ 
2:    $Q \leftarrow |\mathcal{X}|, q \leftarrow \log_2 Q$ 
3:    $\mathbf{X} \leftarrow$  all  $\nu \times 1$  vectors in  $\mathcal{X}^\nu$ 
4:    $\mathbf{x} \leftarrow \mathbf{0}_{N \times 1}$ 
5:    $\mu_1, \mu_0 \leftarrow -\infty_{qN \times 1}$ 
6:   for  $t = 1 : N/\nu$  do
7:      $\pi \leftarrow [\nu(t-1)+1 : N, 1 : \nu(t-1)]$ 
8:      $[\mathbf{L}_p, \mathbf{y}_p, \sim, \sim, \mathbf{Z}^{-1}] \leftarrow \text{WLZ}(\mathbf{H}(:, \pi), \mathbf{y}, \nu, c)$ 
9:      $\mathbf{L}_z \leftarrow \mathbf{L}_p\mathbf{Z}^{-1}$ 
10:    for  $j = 1 : Q^\nu$  do
11:       $\mathbf{x}(1:\nu) \leftarrow \mathbf{X}(1:\nu, j)$ 
12:      for  $i = \nu+1 : N$  do
13:         $\mathbf{x}(i) \leftarrow \left\lfloor \frac{\mathbf{y}_p(i) - \mathbf{L}_z(i, 1:\nu)\mathbf{x}(1:\nu)}{\mathbf{L}_z(i, i)} \right\rfloor$ 
14:      end for
15:       $\mu \leftarrow -\|\mathbf{y}_p - \mathbf{L}_z\mathbf{x}\|^2$ 
16:       $\mathbf{b} \leftarrow \text{binary}(\mathbf{x})$ 
17:      for  $k = 1 : qN$  do
18:        if  $\mathbf{b}(k) = 1$  then
19:           $\mu_1(k) \leftarrow \max\{\mu_1(k), \mu\}$ 
20:        else
21:           $\mu_0(k) \leftarrow \max\{\mu_0(k), \mu\}$ 
22:        end if
23:      end for
24:    end for
25:  end for
26:   $\Lambda \leftarrow (\mu_1 - \mu_0)/N_0$ 
27: end function
```

▷ $\nu \times Q^\nu$ symbol matrix
 ▷ $N \times 1$ column symbol vector
 ▷ $qN \times 1$ metric vec. initialized to $-\infty$
 ▷ process ν parent layers at a time
 ▷ col permutation
 ▷ Integer addition/scaling operations
 ▷ loop over all $\nu \times 1$ vectors in \mathcal{X}^ν
 ▷ ν parent layer symbols
 ▷ $N - \nu$ child layer symbols
 ▷ slice
 ▷ metric using punctured \mathbf{L}_z
 ▷ $qN \times 1$ binary rep. of all \mathbf{x}
 ▷ update metrics for all symbol bits
 ▷ k loop
 ▷ j loop
 ▷ t loop
 ▷ $qN \times 1$ vector of LLRs

SUPPLEMENT S10

AWDL MIMO DETECTION ALGORITHMS

Alg. 9 AWDL MIMO detection algorithm

\triangleright Perform soft-output MIMO detection by puncturing the augmented matrix \mathbf{H}_a using 1-sided square-root-free WDL() decomposition scheme of Alg. 4. Process ν parent layers at a time. In each run, layers are permuted so that a new group of ν symbols are chosen as parent symbols. N/ν independent runs are performed. Metrics of **parent layer symbols only** are updated in each run. This is because, for every layer ordering of \mathbf{H}_a , the puncturing matrix changes and is not unitary. Hence the required metrics are not preserved when the columns of \mathbf{H} are permuted.

\triangleright \mathbf{H} : Complex $M \times N$ matrix, $M \geq N$
 \triangleright \mathbf{y} : Complex $M \times 1$ column vector
 \triangleright N_0 : noise variance
 \triangleright \mathcal{X} : set of Q modulation constellation symbols; $|\mathcal{X}| = Q = 2^q$
 \triangleright ν : puncturing order (assume N is a multiple of ν)
 \triangleright Λ : $qN \times 1$ bit LLR vector
 \triangleright Note: Metric computation on line 16 is expressed in this form for brevity. It can be simplified since $\tilde{\mathbf{L}}_{ap}$ is punctured and sparse.

```

1: function  $\Lambda = \text{AWDLdetector}(\mathbf{H}, \mathbf{y}, N_0, \mathcal{X}, \nu)$ 
2:    $Q \leftarrow |\mathcal{X}|$ ,  $q \leftarrow \log_2 Q$ 
3:    $E_s \leftarrow \frac{1}{Q} \sum_{x \in \mathcal{X}} |x|^2$   $\triangleright$  Avg. symbol energy
4:    $\mathbf{X} \leftarrow$  all  $\nu \times 1$  vectors in  $\mathcal{X}^\nu$   $\triangleright \nu \times Q^\nu$  matrix of symbols
5:    $\mathbf{x} \leftarrow \mathbf{0}_{N \times 1}$   $\triangleright N \times 1$  column symbol vector
6:    $\mu_1, \mu_0 \leftarrow -\infty_{qN \times 1}$   $\triangleright qN \times 1$  metric vec. initialized to  $-\infty$ 
7:    $\mathbf{H}_a \leftarrow \begin{bmatrix} \frac{1}{\sqrt{N_0}} \mathbf{H} \\ \frac{1}{\sqrt{E_s}} \mathbf{I}_N \end{bmatrix}$ ,  $\mathbf{y}_a \leftarrow \frac{1}{\sqrt{N_0}} \begin{bmatrix} \mathbf{y} \\ \mathbf{0}_{N \times 1} \end{bmatrix}$   $\triangleright$  augmented  $\mathbf{H}_a, \mathbf{y}_a$ 
8:   for  $t = 1 : N/\nu$  do  $\triangleright$  process  $\nu$  parent layers at a time
9:      $\pi \leftarrow [\nu(t-1)+1 : N, 1 : \nu(t-1)]$   $\triangleright$  column permutation
10:     $[\tilde{\mathbf{L}}_{ap}, \tilde{\mathbf{y}}_{ap}, \mathbf{D}] \leftarrow \text{WDL}(\mathbf{H}_a(:, \pi), \mathbf{y}_a, \nu)$   $\triangleright$  permuted cols
11:    for  $j = 1 : Q^\nu$  do  $\triangleright$  loop over all  $\nu \times 1$  vectors in  $\mathcal{X}^\nu$ 
12:       $\mathbf{x}(1:\nu) \leftarrow \mathbf{X}(1:\nu, j)$   $\triangleright \nu$  parent layer symbols
13:      for  $i = \nu+1 : N$  do  $\triangleright N - \nu$  child layer symbols
14:         $\mathbf{x}(i) \leftarrow \left[ \frac{\tilde{\mathbf{y}}_{ap}(i) - \tilde{\mathbf{L}}_{ap}(i, 1:\nu) \mathbf{x}(1:\nu)}{1 - 1/(E_s \mathbf{D}(i, i))} \right]$   $\triangleright$  slice
15:      end for
16:       $\mu \leftarrow \frac{1}{E_s} \|\mathbf{x}\|^2 - (\tilde{\mathbf{y}}_{ap} - \tilde{\mathbf{L}}_{ap} \mathbf{x})^\dagger \mathbf{D} (\tilde{\mathbf{y}}_{ap} - \tilde{\mathbf{L}}_{ap} \mathbf{x})$   $\triangleright$  metric
17:       $\mathbf{b} \leftarrow \text{binary}(\mathbf{x}(1:\nu))$   $\triangleright q\nu \times 1$  binary representation
18:      for  $k = 1 : q\nu$  do  $\triangleright$  metrics for  $q\nu$  parent symbol bits
19:        if  $\mathbf{b}(k) = 1$  then
20:           $\mu_1(q\nu(t-1)+k) \leftarrow \max\{\mu_1(q\nu(t-1)+k), \mu\}$ 
21:        else
22:           $\mu_0(q\nu(t-1)+k) \leftarrow \max\{\mu_0(q\nu(t-1)+k), \mu\}$ 
23:        end if
24:      end for  $\triangleright k$  loop
25:    end for  $\triangleright j$  loop
26:  end for  $\triangleright t$  loop
27:   $\Lambda \leftarrow \mu_1 - \mu_0$   $\triangleright qN \times 1$  vector of LLRs
28: end function
    
```

SUPPLEMENT S11

AWDL-BOX MIMO DETECTION ALGORITHMS

Alg. 10 AWDL-BOX MIMO detection algorithm

\triangleright *Optimized version of AWDLdetector for $\nu=2$. Process 2 parent layers at a time, by enumerating over parent 1 and doing ZF-DF for parent 2. The search for parent 2 is expanded to a window of size η around the ZF solution. The parents are switched and the process is repeated for a second run. In each pair of runs, a new pair symbols is chosen as parents. $N/2$ pairs of runs are performed. Metrics of **all symbols** are updated in each pair of runs. This is because metrics are preserved if parent layers are permuted and child layers are permuted independently, but metrics are not preserved for col arbitrary permutations.*
 \triangleright **H**: Complex $M \times N$ matrix, $M \geq N$
 \triangleright **y**: Complex $M \times 1$ column vector
 \triangleright N_0 : noise variance
 \triangleright \mathcal{X} : set of Q modulation constellation symbols; $|\mathcal{X}|=Q=2^q$
 \triangleright η : window size around ZF solution for parent 2
 \triangleright Λ : $qN \times 1$ bit LLR vector
 \triangleright *Note: WDL decomposition on line 9 can be optimized for each pair of runs since right-most $N-2$ cols of $\tilde{\mathbf{L}}_{\text{ap}}$ do not change.*
 \triangleright *Note: Metric computation on line 18 is expressed in this form for brevity. It can be simplified since $\tilde{\mathbf{L}}_{\text{ap}}$ is punctured and sparse.*

```

1: function  $\Lambda = \text{AWDLXdetector}(\mathbf{H}, \mathbf{y}, N_0, \mathcal{X}, \eta)$ 
2:    $Q \leftarrow |\mathcal{X}|$ ,  $q \leftarrow \log_2 Q$ ,  $E_s \leftarrow \frac{1}{Q} \sum_{x \in \mathcal{X}} |x|^2$ 
3:    $\mathbf{x} \leftarrow \mathbf{0}_{N \times 1}$ 
4:    $\mu_1, \mu_0 \leftarrow -\infty_{qN \times 1}$ 
5:    $\mathbf{H}_a \leftarrow \begin{bmatrix} \frac{1}{\sqrt{N_0}} \mathbf{H} \\ \frac{1}{\sqrt{E_s}} \mathbf{I}_N \end{bmatrix}$ ,  $\mathbf{y}_a \leftarrow \frac{1}{\sqrt{N_0}} \begin{bmatrix} \mathbf{y} \\ \mathbf{0}_{N \times 1} \end{bmatrix}$ 
6:   for  $t = 1 : N/2$  do
7:     for  $p = 1 : 2$  do
8:        $\pi \leftarrow [2t-2+p, 2t-p+1, 2t+1 : N, 1 : 2t-2]$ 
9:        $[\tilde{\mathbf{L}}_{\text{ap}}, \tilde{\mathbf{y}}_{\text{ap}}, \mathbf{D}] \leftarrow \text{WDL}(\mathbf{H}_a(:, \pi), \mathbf{y}_a, 2)$ 
10:      for  $j = 1 : Q$  do
11:         $\mathbf{x}(1) \leftarrow \mathcal{X}(j)$ 
12:         $z \leftarrow \left[ \frac{\tilde{\mathbf{y}}_{\text{ap}}(2) - \tilde{\mathbf{L}}_{\text{ap}}(2,1)\mathbf{x}(1)}{1 - 1/(E_s \mathbf{D}(2,2))} \right]$ 
13:         $\mathcal{W}(z) \leftarrow \eta$  closest symbols in  $\mathcal{X}$  to  $z$ 
14:        for all  $\omega \in \mathcal{W}(z)$  do
15:           $\mathbf{x}(2) \leftarrow \omega$ 
16:          for  $i = 3 : N$  do
17:             $\mathbf{x}(i) \leftarrow \left[ \frac{\tilde{\mathbf{y}}_{\text{ap}}(i) - \tilde{\mathbf{L}}_{\text{ap}}(i,1:2)\mathbf{x}(1:2)}{1 - 1/(E_s \mathbf{D}(i,i))} \right]$ 
18:             $\mu \leftarrow \frac{1}{E_s} \|\mathbf{x}\|^2 - \|\tilde{\mathbf{y}}_{\text{ap}} - \tilde{\mathbf{L}}_{\text{ap}}\mathbf{x}\|_{\mathbf{D}}^2$ 
19:             $\mathbf{b} \leftarrow \text{binary}(\mathbf{x}(1:2, 1))$ 
20:            for  $k = 1 : 2q$  do
21:               $r \leftarrow (k-1 + (p-1)q) \% (2q) + 1$ 
22:              if  $\mathbf{b}(k) = 1$  then
23:                 $\mu_1(2q(t-1)+r) \leftarrow \max\{\mu_1(2q(t-1)+r), \mu\}$ 
24:              else
25:                 $\mu_0(2q(t-1)+r) \leftarrow \max\{\mu_0(2q(t-1)+r), \mu\}$ 
26:              end if
27:            end if
28:          end for
29:        end for
30:      end for
31:    end for
32:  end for
33:   $\Lambda \leftarrow \mu_1 - \mu_0$ 
34: end function

```

$\triangleright N \times 1$ column symbol vector
 $\triangleright qN \times 1$ metric vec. initialized to $-\infty$
 \triangleright process 2 parent layers at a time
 \triangleright parent layers order: [1, 2] or [2, 1]
 $\triangleright \nu=2$
 \triangleright loop over all symbols in \mathcal{X}
 \triangleright parent layer symbol
 \triangleright slice layer 2
 $\triangleright \eta$ closest symbols to z
 \triangleright set as layer 2 symbol
 $\triangleright N-2$ child layer symbols
 \triangleright slice
 \triangleright metric
 \triangleright binary repres.
 \triangleright parent bits metrics
 \triangleright index
 $\triangleright k$ loop
 $\triangleright i$ loop
 $\triangleright \omega$ loop
 $\triangleright j$ loop
 $\triangleright p$ loop
 $\triangleright t$ loop
 $\triangleright qN \times 1$ vector of LLRs

SUPPLEMENT S12

LORD MIMO DETECTION ALGORITHM

Alg. 11 LORD MIMO detection algorithm

▷ *LORD soft-output MIMO detection using QLy() decomposition scheme of Alg. 1. Process ν parent layers at a time. In each run, layers are permuted so that a new group of ν symbols are chosen as parent symbols. N/ν independent runs are performed. Metrics of **parent layer symbols only** are updated in each run.*

▷ **H**: Complex $M \times N$ matrix, $M \geq N$

▷ **y**: Complex $M \times 1$ column vector

▷ N_0 : noise variance

▷ \mathcal{X} : set of Q modulation constellation symbols; $|\mathcal{X}| = Q = 2^q$

▷ ν : puncturing order (assume N is a multiple of ν)

▷ Λ : $qN \times 1$ bit LLR vector

▷ *Note: Distance computation on line 14 is expressed in this form for brevity. It can be simplified since **L** is lower-triangular.*

```

1: function  $\Lambda = \text{LORDdetector}(\mathbf{H}, \mathbf{y}, N_0, \mathcal{X}, \nu)$ 
2:    $Q \leftarrow |\mathcal{X}|, q \leftarrow \log_2 Q$ 
3:    $\mathbf{X} \leftarrow$  all  $\nu \times 1$  vectors in  $\mathcal{X}^\nu$ 
4:    $\mathbf{x} \leftarrow \mathbf{0}_{N \times 1}$ 
5:    $\mu_1, \mu_0 \leftarrow -\infty_{qN \times 1}$ 
6:   for  $t = 1 : N/\nu$  do
7:      $\pi \leftarrow [\nu(t-1)+1 : N, 1 : \nu(t-1)]$ 
8:      $[\tilde{\sim}, \tilde{\mathbf{L}}, \tilde{\mathbf{y}}] \leftarrow \text{QLy}(\mathbf{H}(:, \pi), \mathbf{y}, \nu)$ 
9:     for  $j = 1 : Q^\nu$  do
10:       $\mathbf{x}(1:\nu) \leftarrow \mathbf{X}(1:\nu, j)$ 
11:      for  $i = \nu + 1 : N$  do
12:         $\mathbf{x}(i) \leftarrow \left\lfloor \frac{\tilde{\mathbf{y}}(i) - \tilde{\mathbf{L}}(i, 1:\nu)\mathbf{x}(1:\nu)}{\tilde{\mathbf{L}}(i, i)} \right\rfloor$ 
13:      end for
14:       $\mu \leftarrow -\|\tilde{\mathbf{y}} - \tilde{\mathbf{L}}\mathbf{x}\|^2$ 
15:       $\mathbf{b} \leftarrow \text{binary}(\mathbf{x}(1:\nu))$ 
16:      for  $k = 1 : q\nu$  do
17:        if  $\mathbf{b}(k) = 1$  then
18:           $\mu_1(q\nu(t-1) + k) \leftarrow \max\{\mu_1(q\nu(t-1) + k), \mu\}$ 
19:        else
20:           $\mu_0(q\nu(t-1) + k) \leftarrow \max\{\mu_0(q\nu(t-1) + k), \mu\}$ 
21:        end if
22:      end for
23:    end for
24:     $\Lambda \leftarrow (\mu_1 - \mu_0) / N_0$ 
25:  end function

```

▷ $\nu \times Q^\nu$ matrix of symbols
 ▷ $N \times 1$ column symbol vector
 ▷ $qN \times 1$ metric vec. initialized to $-\infty$
 ▷ process ν parent layers at a time
 ▷ column permutation
 ▷ permuted cols
 ▷ loop over all $\nu \times 1$ vectors in \mathcal{X}^ν
 ▷ ν parent layer symbols
 ▷ $N - \nu$ child layer symbols
 ▷ slice
 ▷ metric using full **L**
 ▷ $q\nu \times 1$ binary representation
 ▷ metrics for $q\nu$ parent symbol bits
 ▷ k loop
 ▷ j loop
 ▷ t loop
 ▷ $qN \times 1$ vector of LLRs

SUPPLEMENT S13

OPTIMIZED LORD MIMO DETECTION ALGORITHM

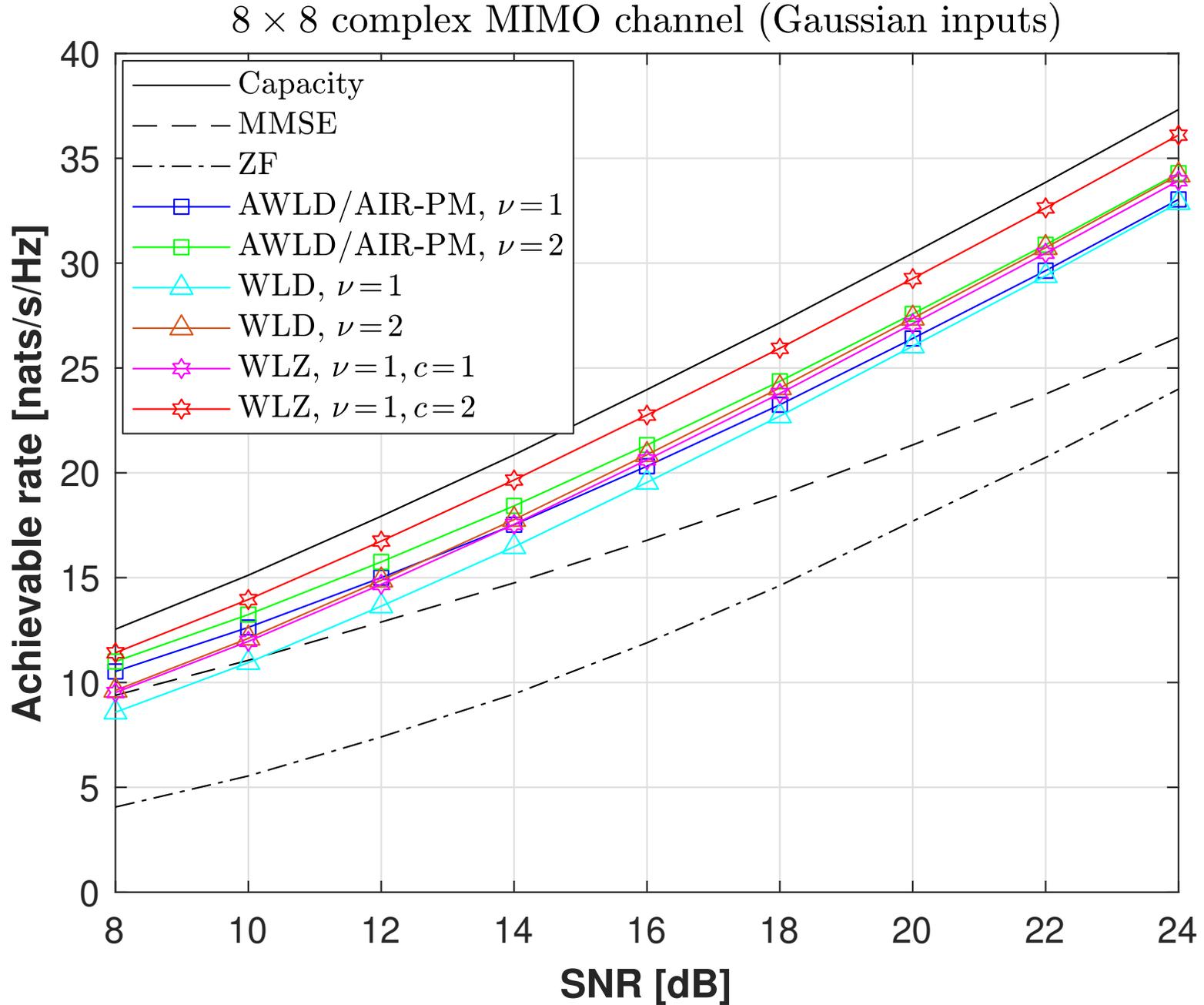
Alg. 12 Optimized LORD MIMO detection algorithm

\triangleright *Optimized version of LORDdetector in Alg. 11 to globally update metrics in each run. Process ν parent layers at a time. In each run, layers are permuted so that a new group of ν symbols are chosen as parent symbols. N/ν independent runs are performed. Metrics of **all layer symbols** are updated in each run. This is possible because Euclidean distance metrics do not change under column permutation of \mathbf{H} .*
 \triangleright \mathbf{H} : Complex $M \times N$ matrix, $M \geq N$
 \triangleright \mathbf{y} : Complex $M \times 1$ column vector
 \triangleright N_0 : noise variance
 \triangleright \mathcal{X} : set of Q modulation constellation symbols; $|\mathcal{X}| = Q = 2^q$
 \triangleright ν : puncturing order (assume N is a multiple of ν)
 \triangleright Λ : $qN \times 1$ bit LLR vector
 \triangleright *Note: Distance computation on line 14 is expressed in this form for brevity. It can be simplified since \mathbf{L} is lower-triangular.*

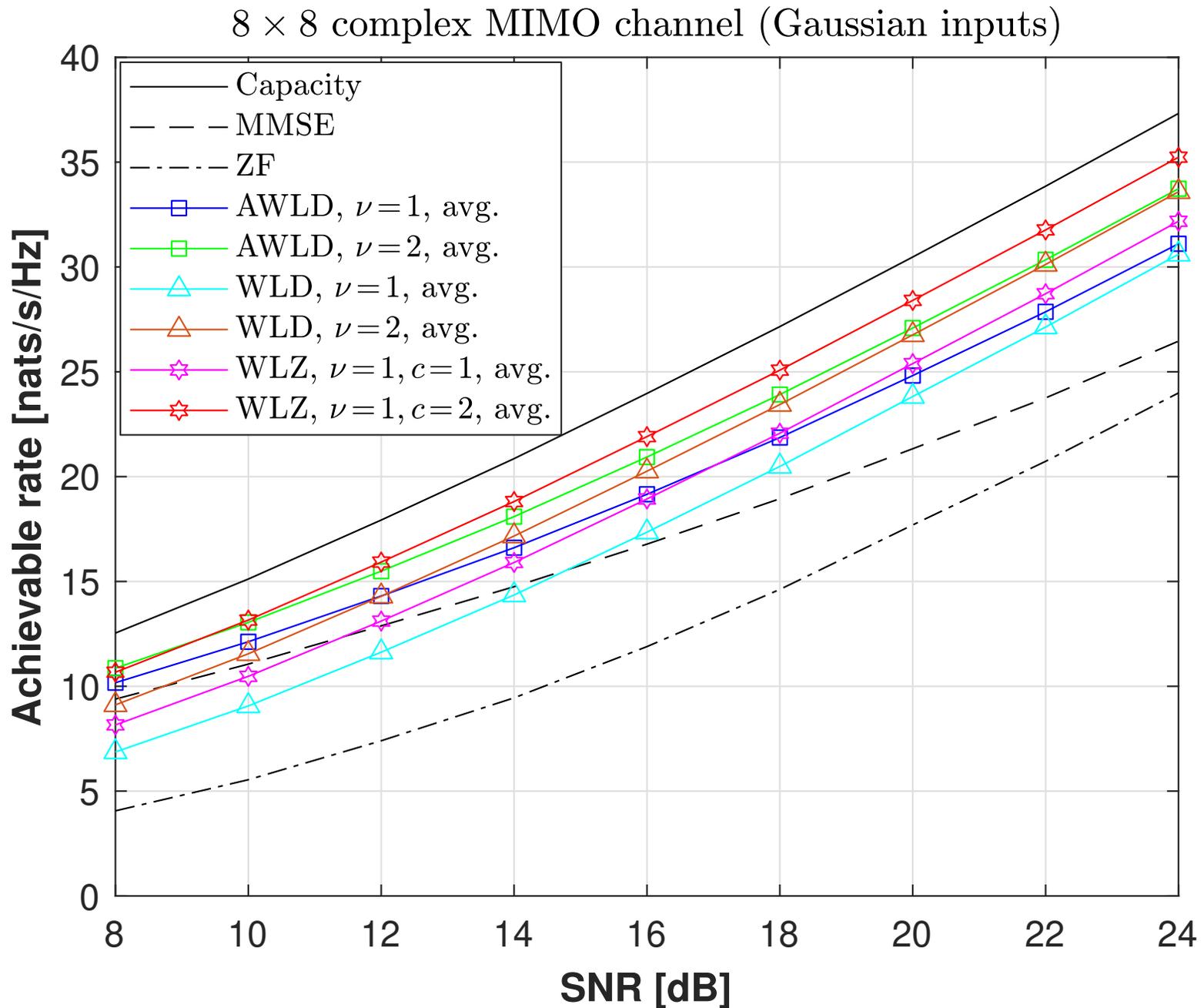
```

1: function  $\Lambda = \text{LORDXdetector}(\mathbf{H}, \mathbf{y}, N_0, \mathcal{X}, \nu)$ 
2:    $Q \leftarrow |\mathcal{X}|, q \leftarrow \log_2 Q$ 
3:    $\mathbf{X} \leftarrow$  all  $\nu \times 1$  vectors in  $\mathcal{X}^\nu$   $\triangleright \nu \times Q^\nu$  matrix of symbols
4:    $\mathbf{x} \leftarrow \mathbf{0}_{N \times 1}$   $\triangleright N \times 1$  column symbol vector
5:    $\mu_1, \mu_0 \leftarrow -\infty_{qN \times 1}$   $\triangleright qN \times 1$  metric vec. initialized to  $-\infty$ 
6:   for  $t = 1 : N/\nu$  do  $\triangleright$  process  $\nu$  parent layers at a time
7:      $\pi \leftarrow [\nu(t-1)+1 : N, 1 : \nu(t-1)]$   $\triangleright$  column permutation
8:      $[\tilde{\sim}, \tilde{\mathbf{L}}, \tilde{\mathbf{y}}] \leftarrow \text{QLy}(\mathbf{H}(:, \pi), \mathbf{y}, \nu)$   $\triangleright$  permuted cols
9:     for  $j = 1 : Q^\nu$  do  $\triangleright$  loop over all  $\nu \times 1$  vectors in  $\mathcal{X}^\nu$ 
10:       $\mathbf{x}(1:\nu) \leftarrow \mathbf{X}(1:\nu, j)$   $\triangleright \nu$  parent layer symbols
11:      for  $i = \nu+1 : N$  do  $\triangleright N - \nu$  child layer symbols
12:         $\mathbf{x}(i) \leftarrow \left\lfloor \frac{\tilde{\mathbf{y}}(i) - \tilde{\mathbf{L}}(i, 1:\nu)\mathbf{x}(1:\nu)}{\tilde{\mathbf{L}}(i, i)} \right\rfloor$   $\triangleright$  slice
13:      end for
14:       $\mu \leftarrow -\|\tilde{\mathbf{y}} - \tilde{\mathbf{L}}\mathbf{x}\|^2$   $\triangleright$  metric using full  $\tilde{\mathbf{L}}$ 
15:       $\mathbf{b} \leftarrow \text{binary}(\mathbf{x})$   $\triangleright qN \times 1$  binary rep. of all  $\mathbf{x}$ 
16:      for  $k = 1 : qN$  do  $\triangleright$  update metrics for all symbol bits
17:        if  $\mathbf{b}(k) = 1$  then
18:           $\mu_1(k) \leftarrow \max\{\mu_1(k), \mu\}$ 
19:        else
20:           $\mu_0(k) \leftarrow \max\{\mu_0(k), \mu\}$ 
21:        end if
22:      end for  $\triangleright k$  loop
23:    end for  $\triangleright j$  loop
24:  end for  $\triangleright t$  loop
25:   $\Lambda \leftarrow (\mu_1 - \mu_0)/N_0$   $\triangleright qN \times 1$  vector of LLRs
26: end function

```

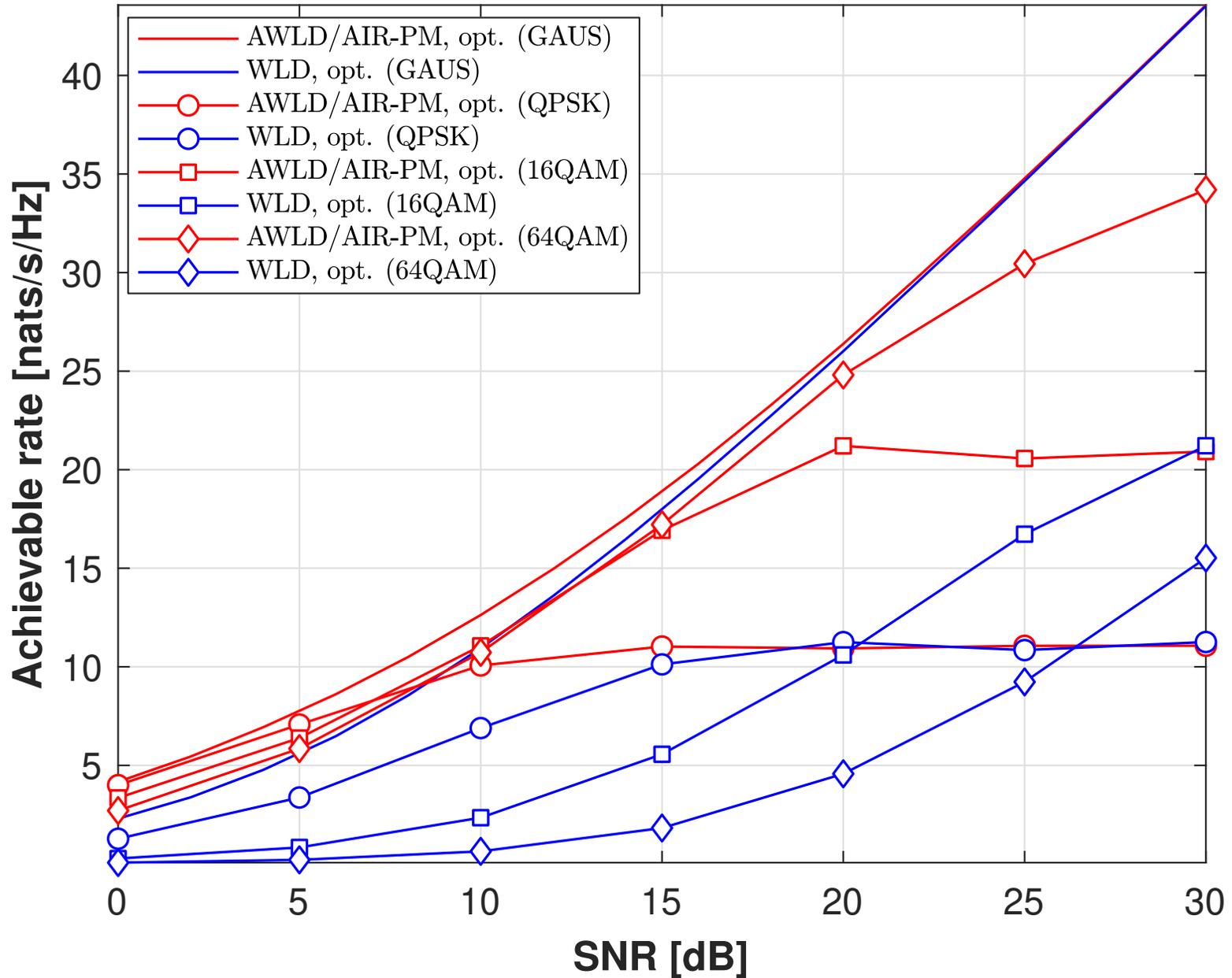


Supplement Figure F9. Comparison of AIRs for 8×8 MIMO channels with Gaussian inputs. For the AWLD, WLD, and WLZ algorithms, parent layers are optimally selected so as to maximize I_{LB}^{WLD} in (41).



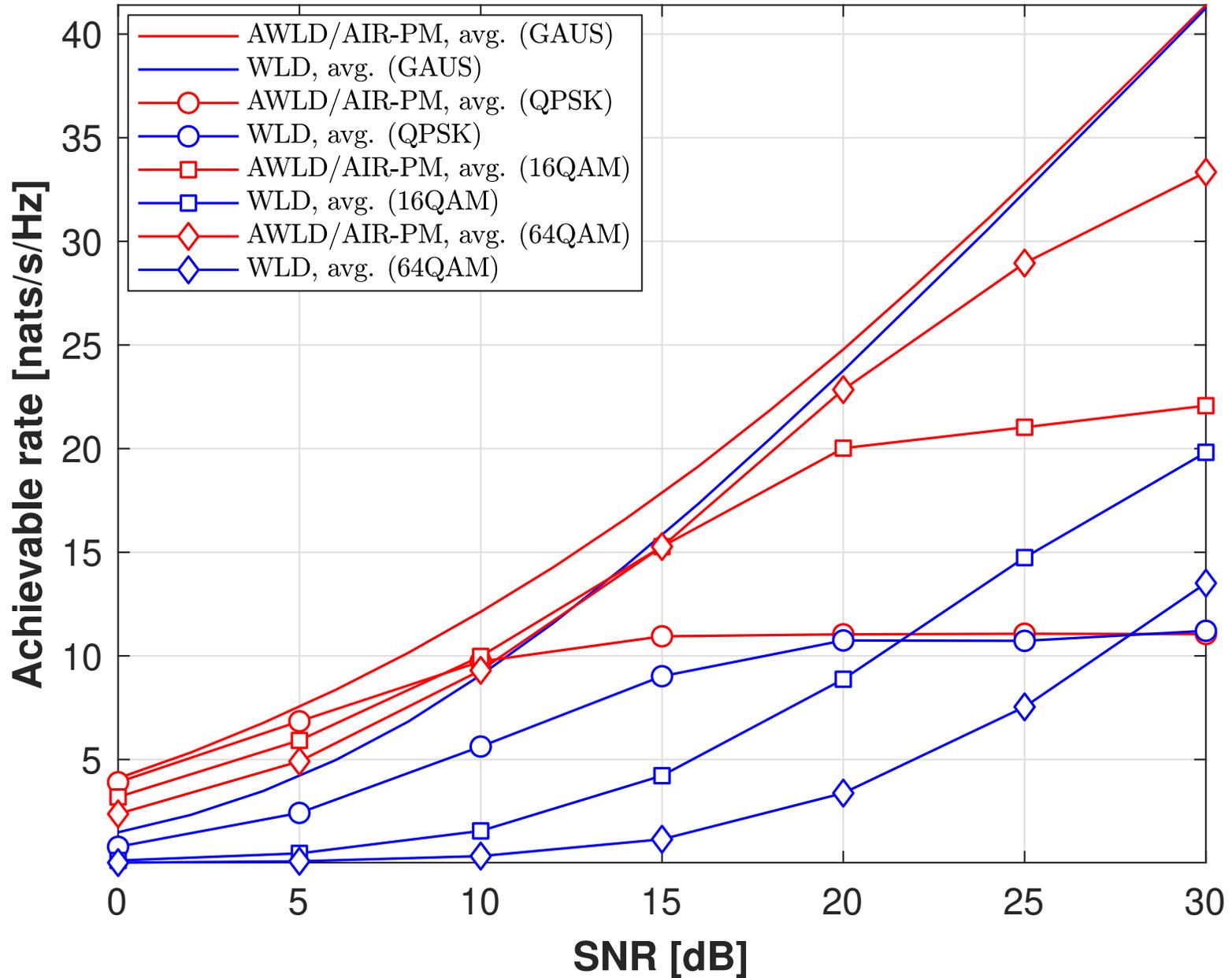
Supplement Figure F10. Comparison of AIRs for 8×8 MIMO channels with Gaussian inputs. The AIRs for the AWLD, WLD, and WLZ algorithms are averaged over all possible parent layer selections.

8×8 complex MIMO channel (finite inputs)

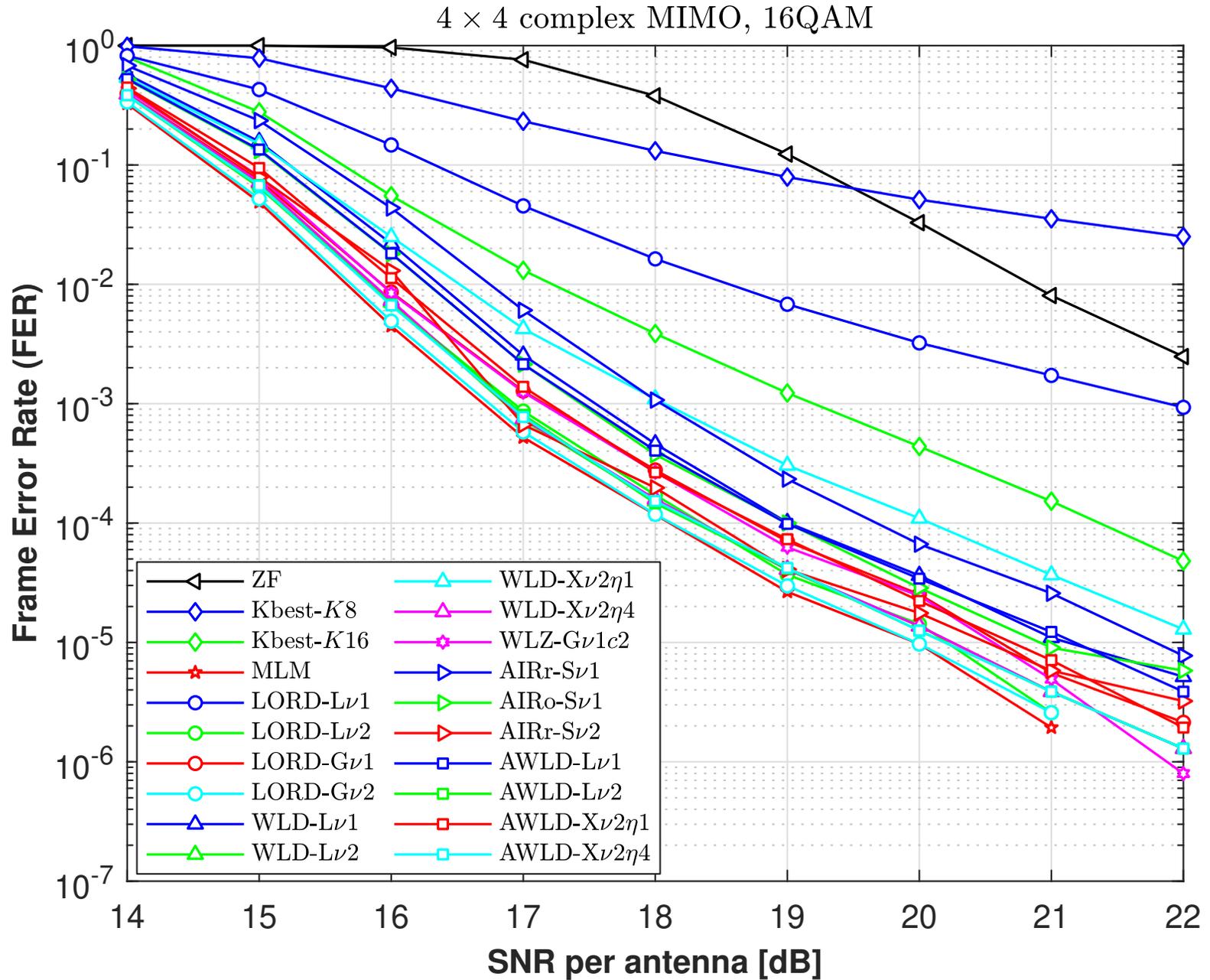


Supplement Figure F11. Comparison of AIRs for 8×8 MIMO channels with finite inputs. For the AWLD, WLD, and WLZ algorithms with QPSK, 16QAM, and 64QAM inputs, parent layers are selected so as to maximize $I_{\text{LB}}^{\text{WLD}}$ in (41) if Gaussian inputs were assumed.

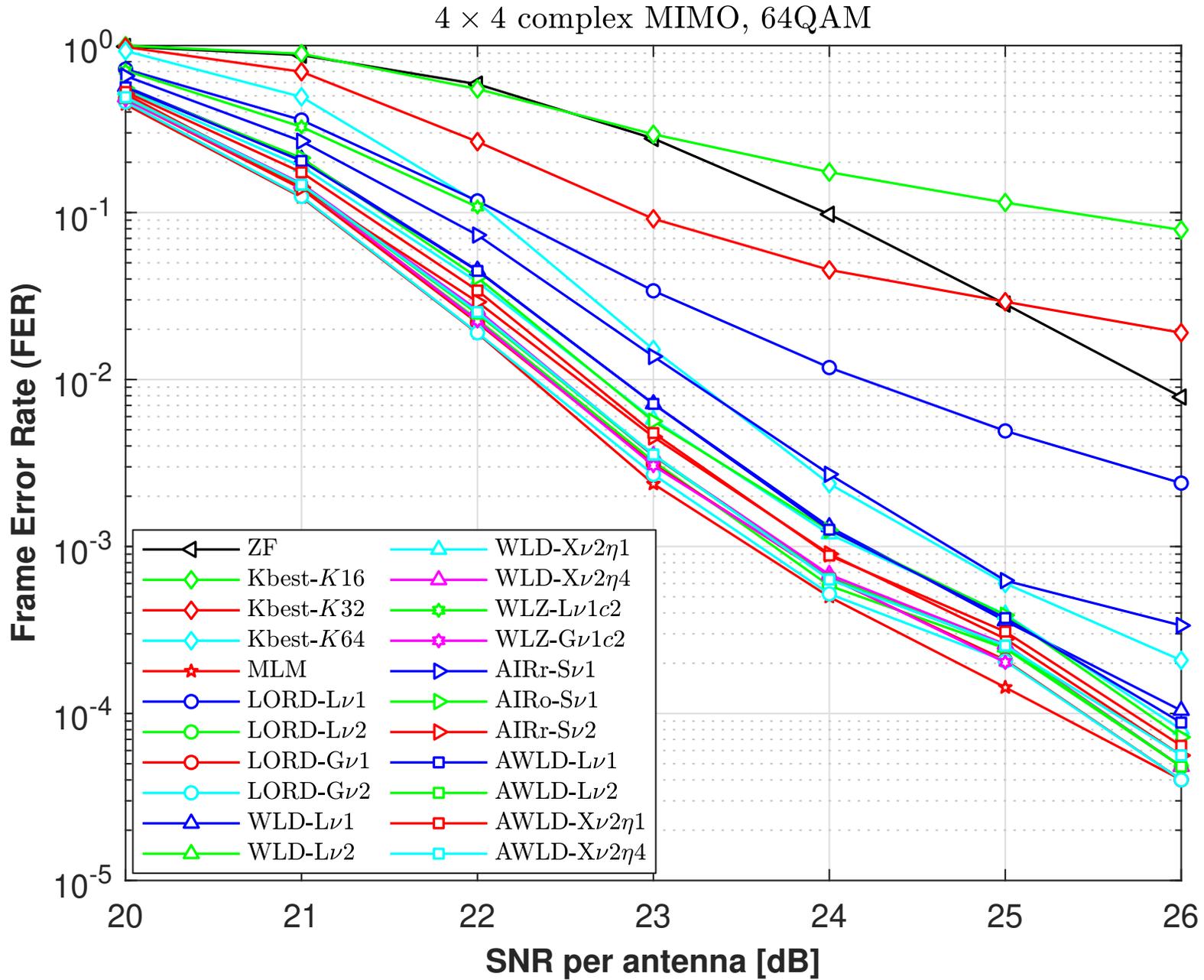
8×8 complex MIMO channel (finite inputs)



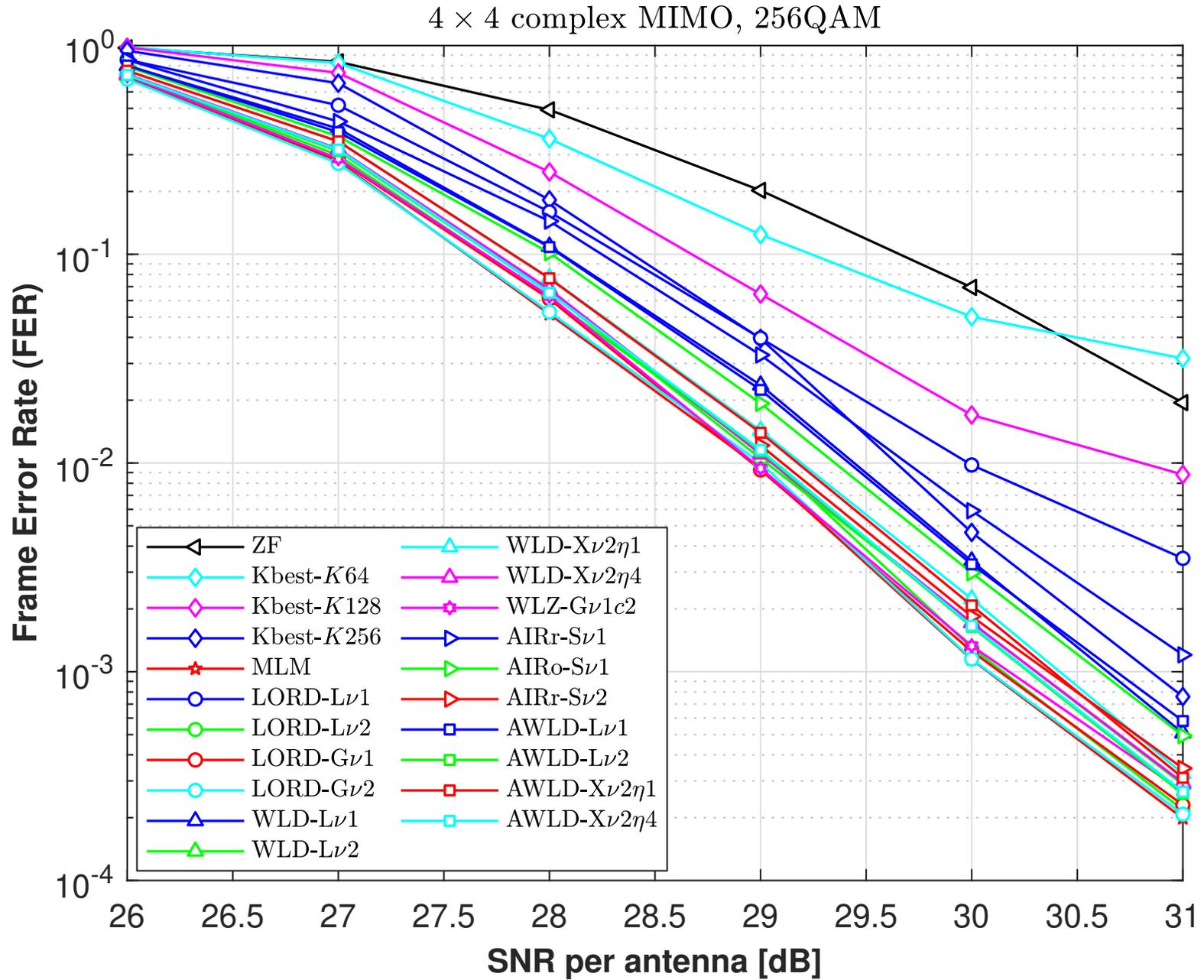
Supplement Figure F12. Comparison of AIRs for 8×8 MIMO channels with finite inputs. The AIRs for the AWLD, WLD, and WLZ algorithms are averaged over all possible parent layer selections.



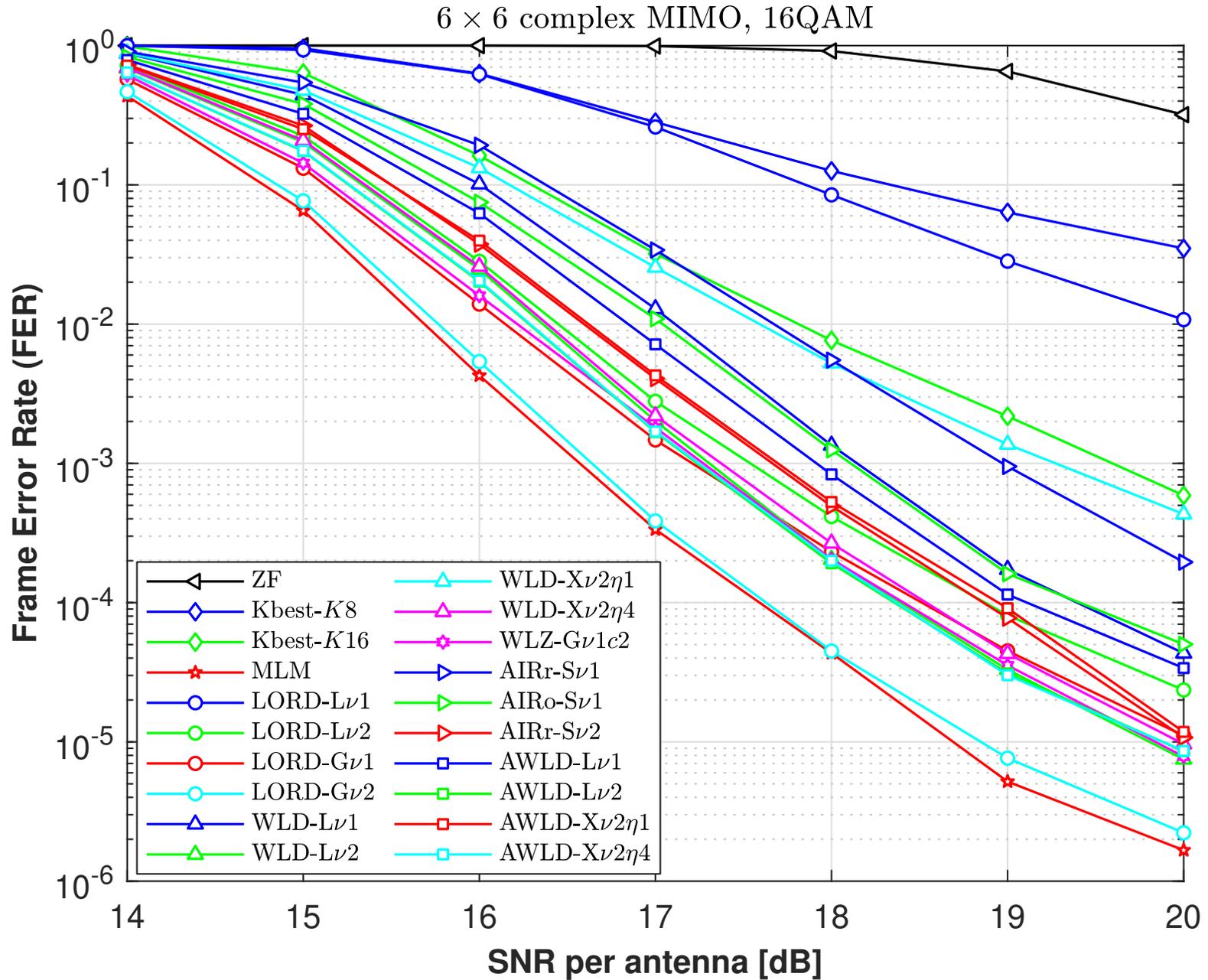
Supplement Figure F13. Frame error-rate of 4×4 complex MIMO channels, 16QAM



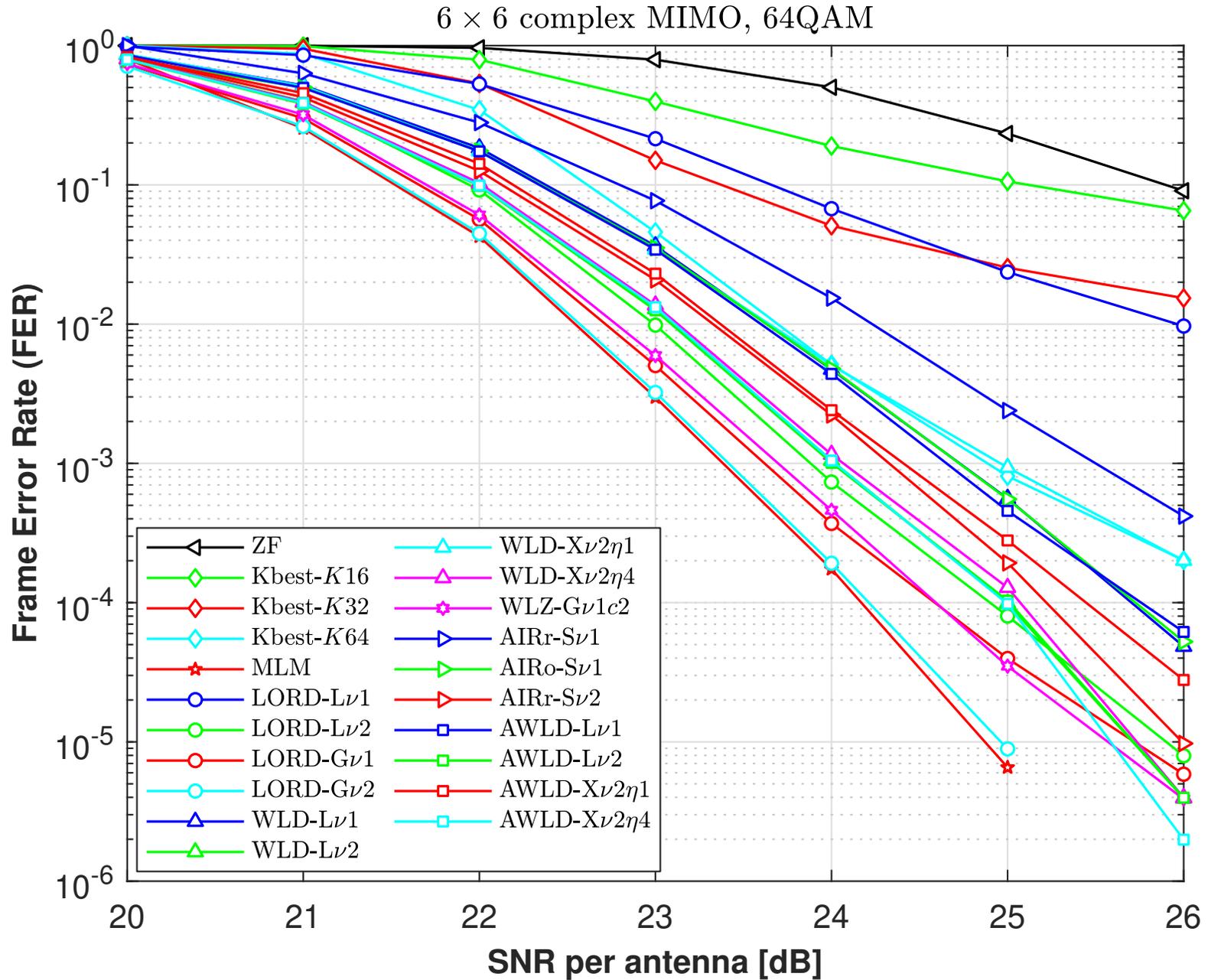
Supplement Figure F14. Frame error-rate of 4×4 complex MIMO channels, 64QAM



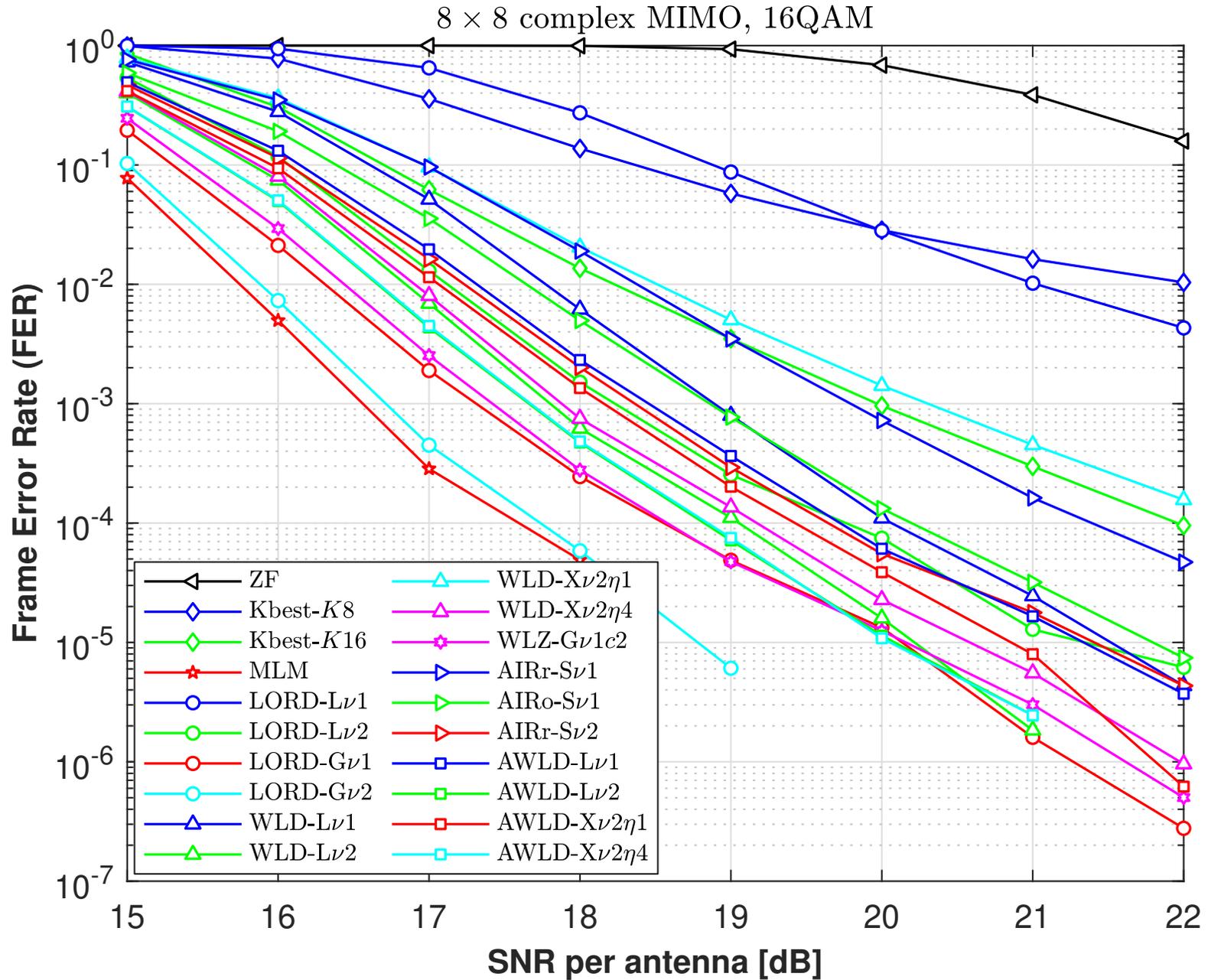
Supplement Figure F15. Frame error-rate of 4×4 complex MIMO channels, 256QAM



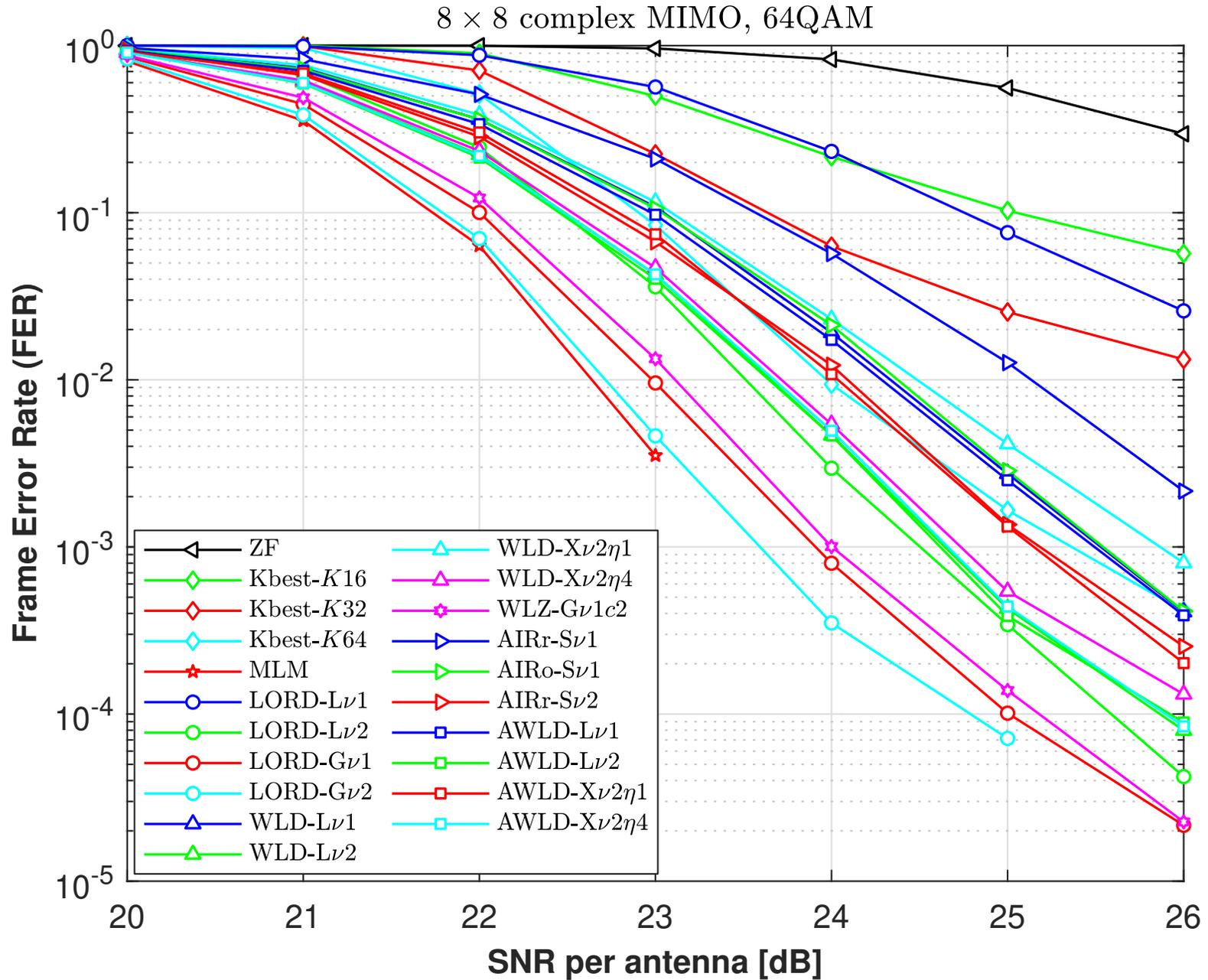
Supplement Figure F16. Frame error-rate of 6×6 complex MIMO channels, 16QAM



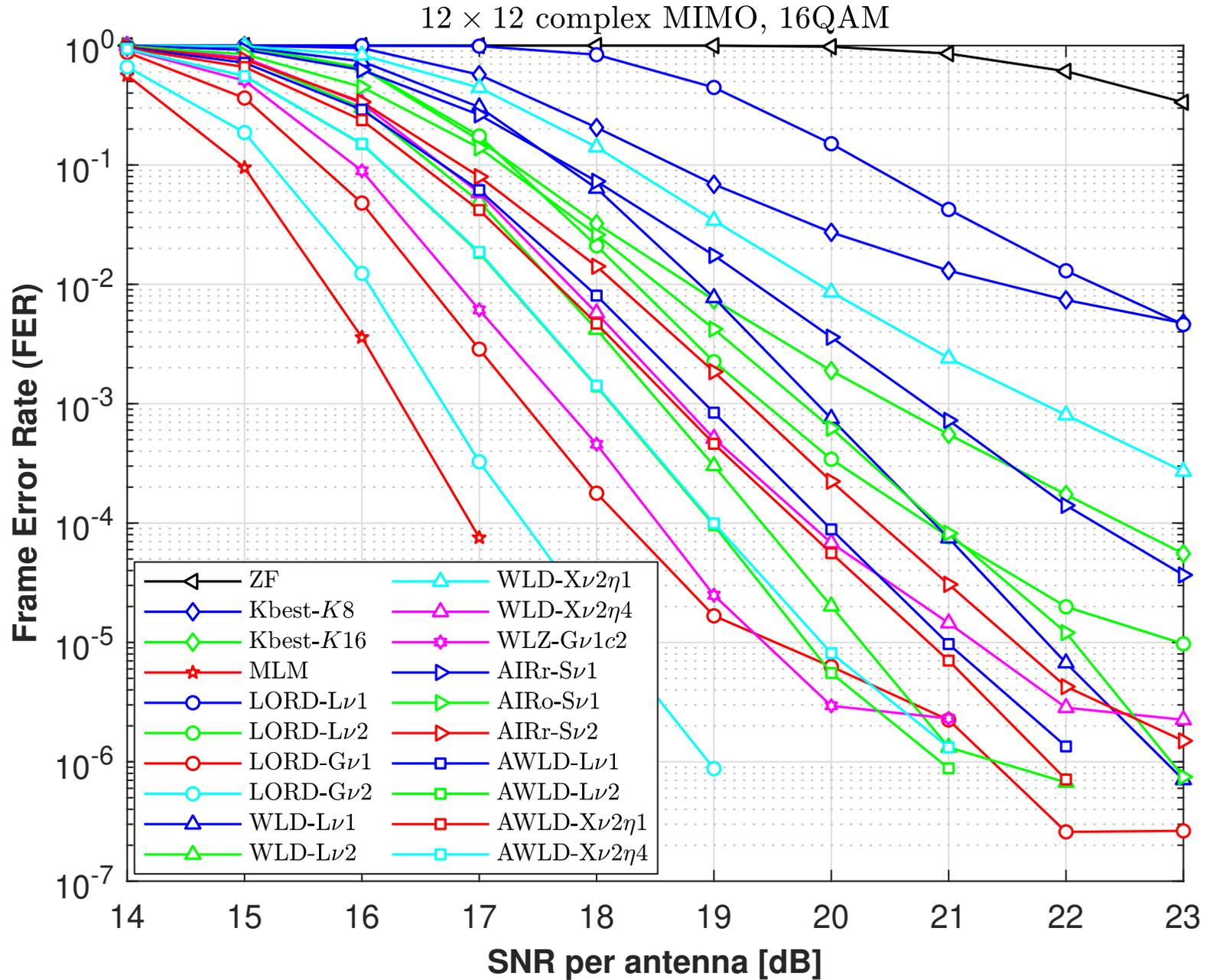
Supplement Figure F17. Frame error-rate of 6×6 complex MIMO channels, 64QAM



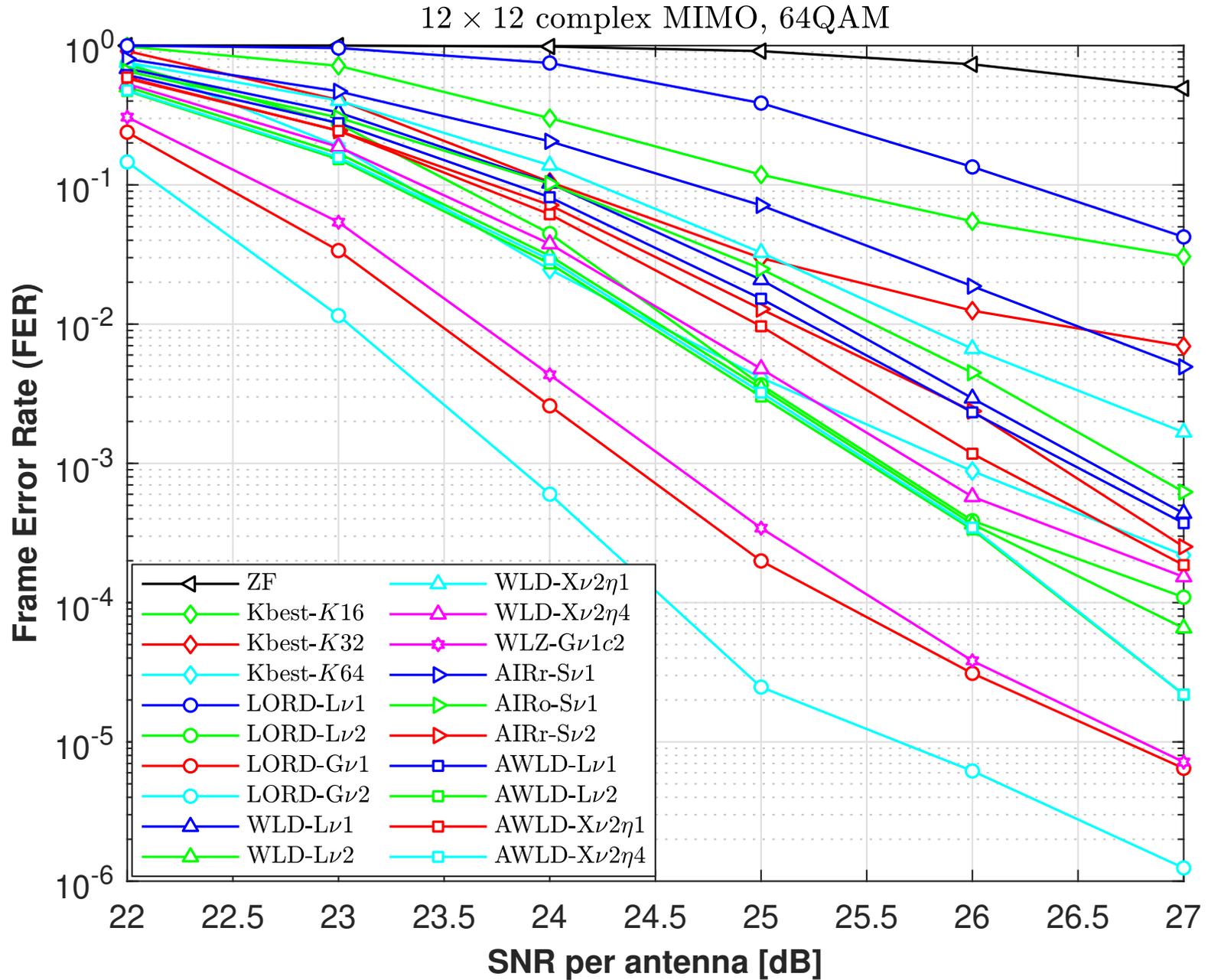
Supplement Figure F18. Frame error-rate of 8×8 complex MIMO channels, 16QAM



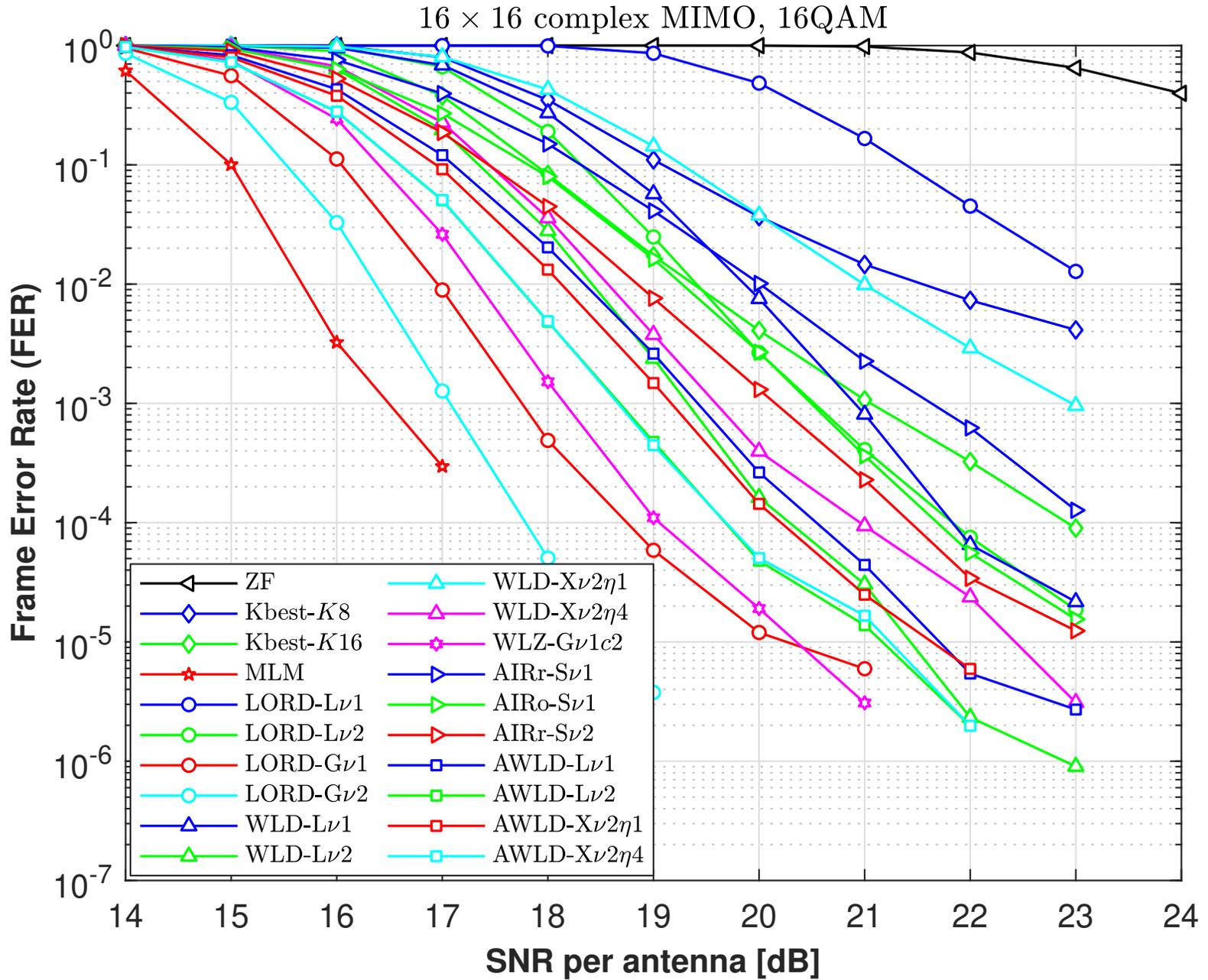
Supplement Figure F19. Frame error-rate of 8×8 complex MIMO channels, 64QAM



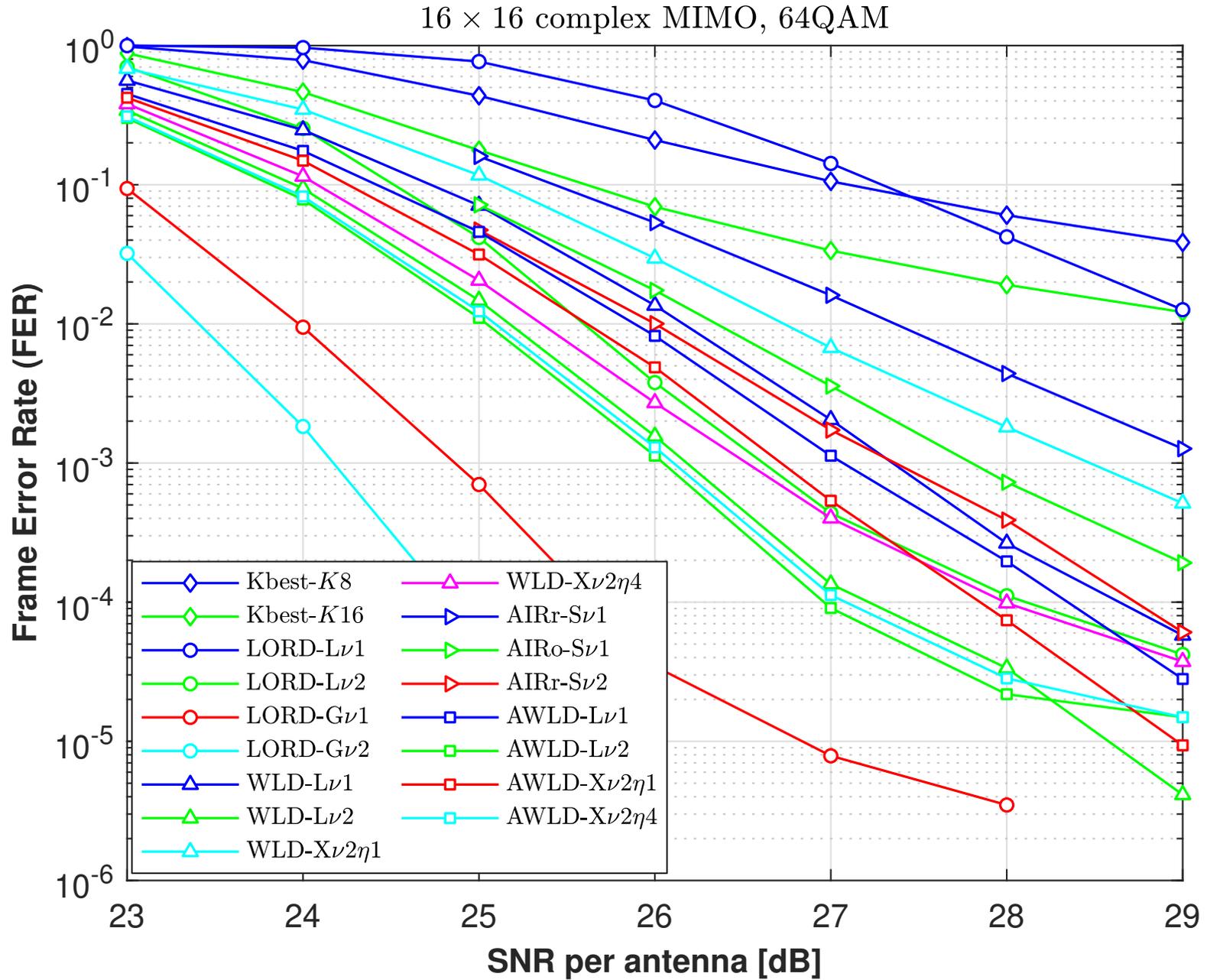
Supplement Figure F20. Frame error-rate of 12×12 complex MIMO channels, 16QAM



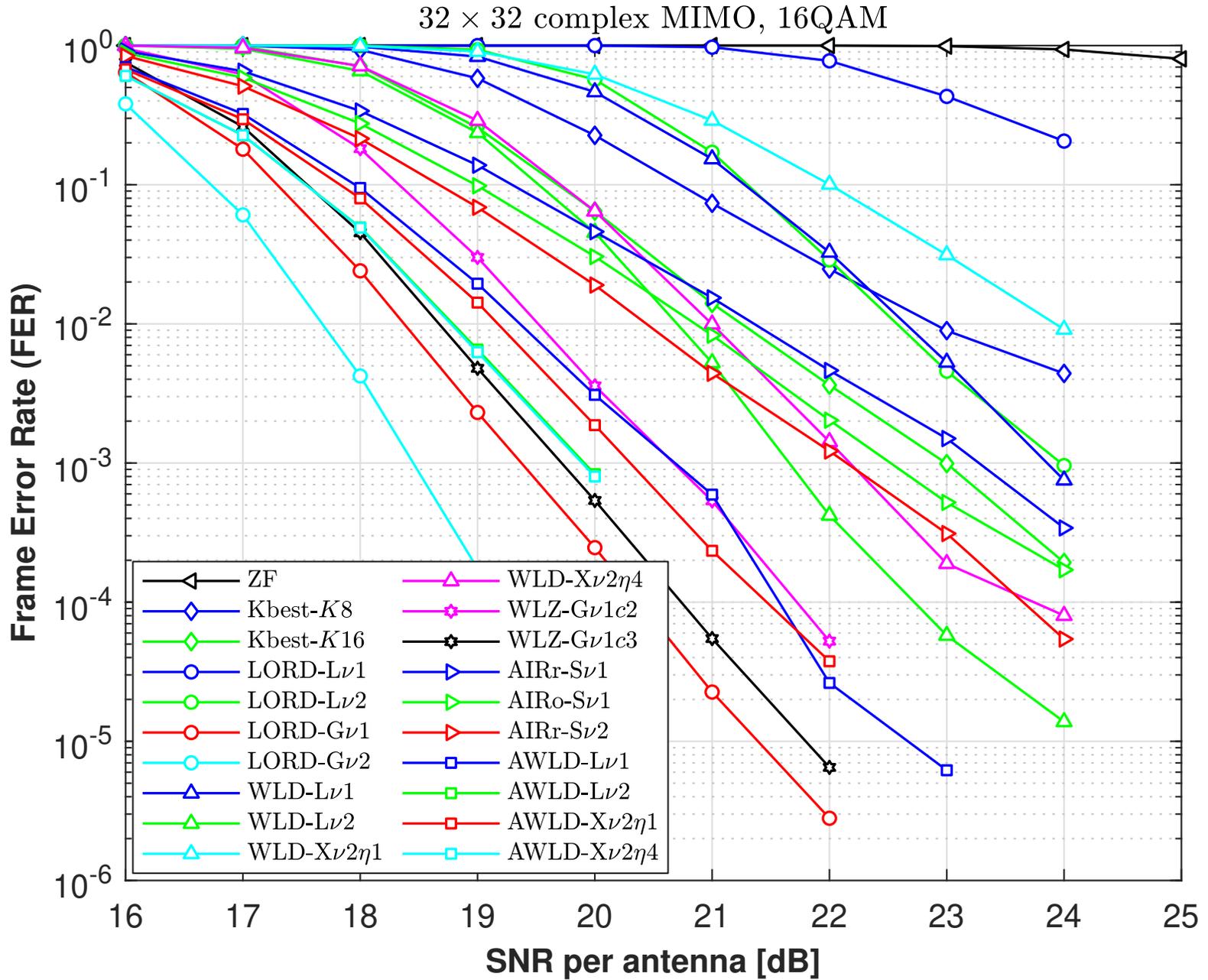
Supplement Figure F21. Frame error-rate of 12×12 complex MIMO channels, 64QAM



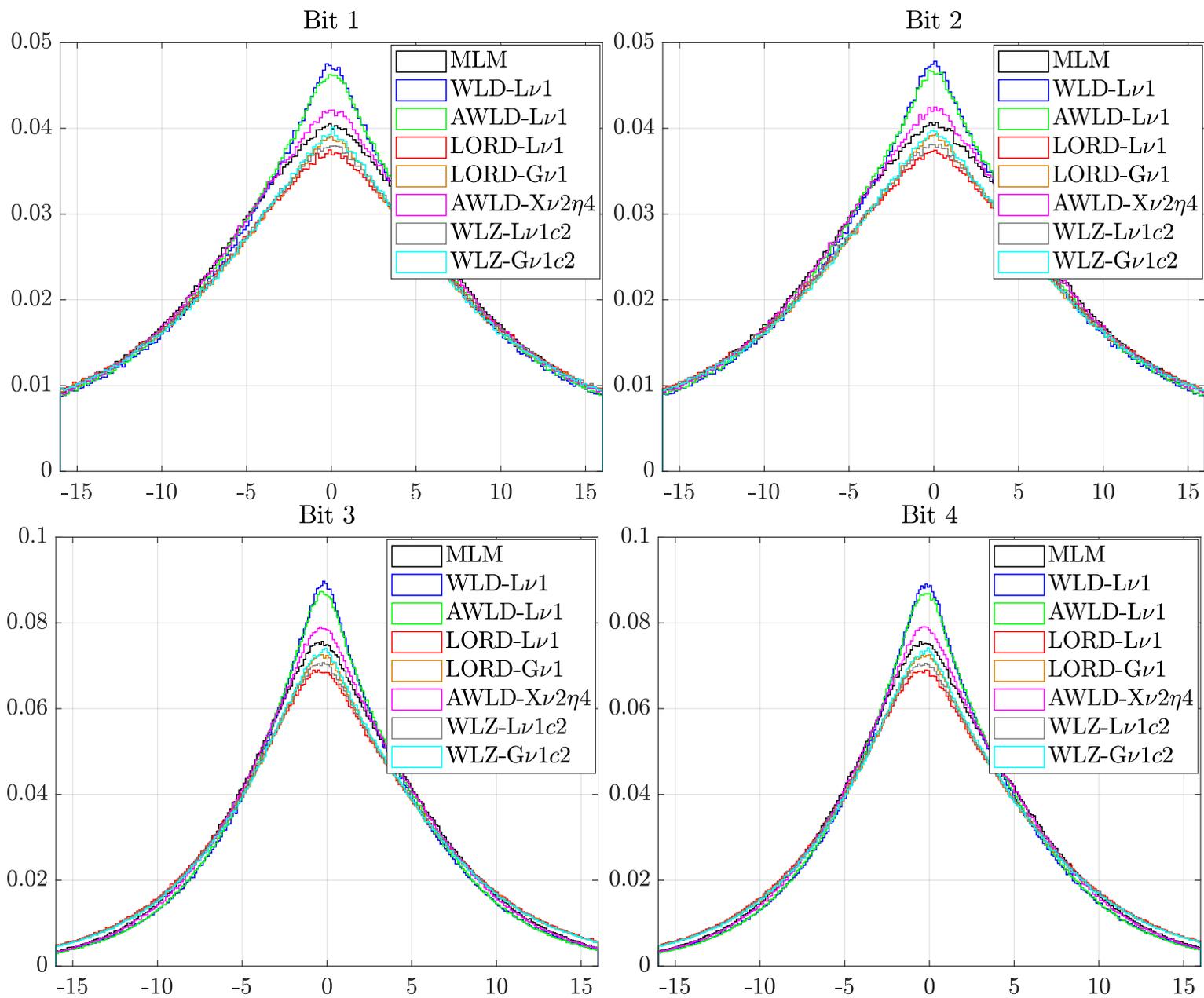
Supplement Figure F22. Frame error-rate of 16×16 complex MIMO channels, 16QAM



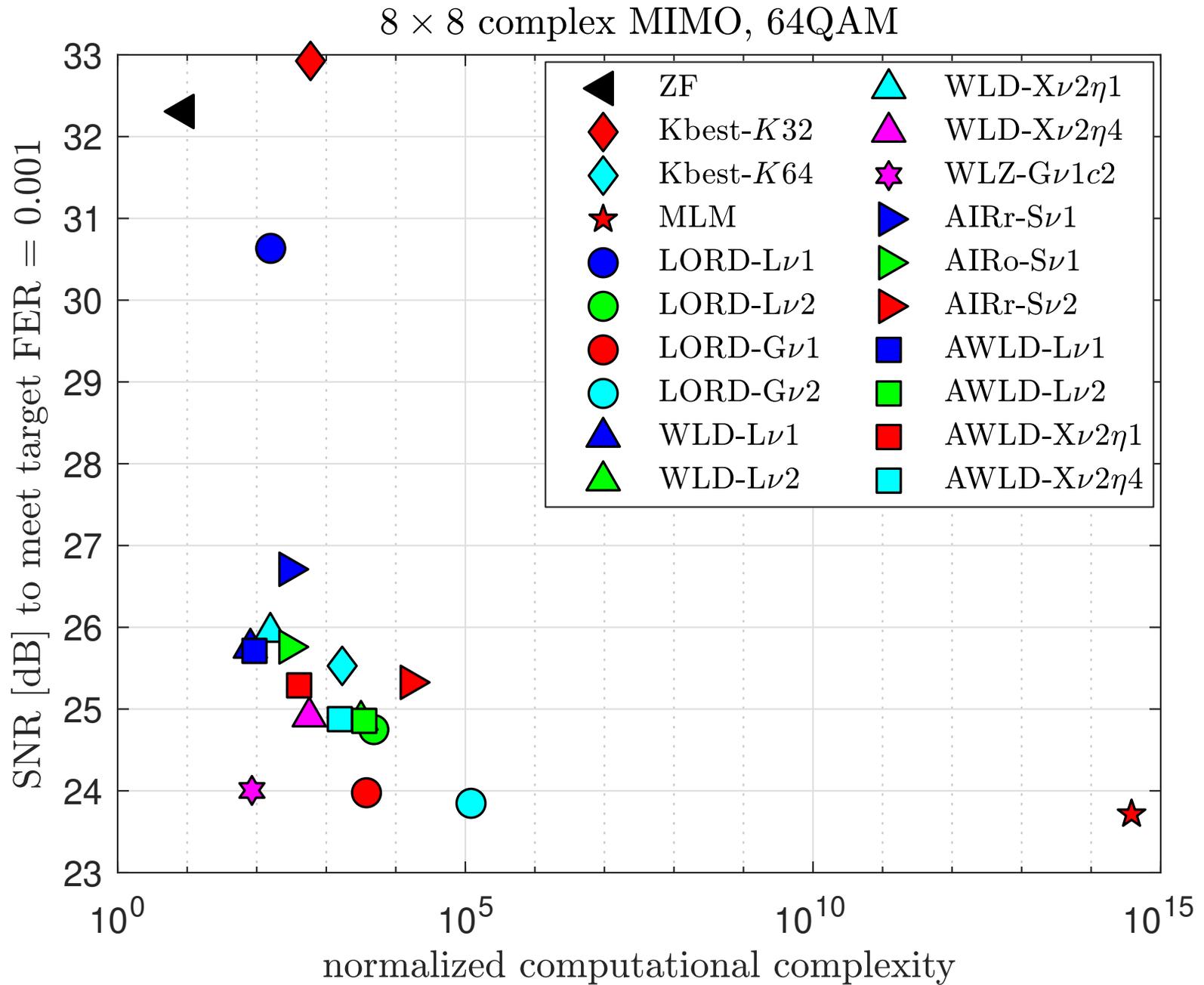
Supplement Figure F23. Frame error-rate of 16×16 complex MIMO channels, 64QAM



Supplement Figure F24. Frame error-rate of 32×32 complex MIMO channels, 16QAM



Supplement Figure F25. Distribution of bit LLRs of one symbol: 4×4 complex MIMO channel, 16QAM, SNR = 20 dB.



Supplement Figure F26. SNR to meet target FER of 0.1% versus complexity.