MSE-OPTIMAL 1-BIT PRECODING FOR MULTIUSER MIMO VIA BRANCH AND BOUND

Sven Jacobsson[☺,♫,🦇], *Weiyu Xu*[🧹], *Giuseppe Durisi*[♫], *and Christoph Studer*[☺]

[☺]Cornell University, Ithaca, NY, USA; [♫]Chalmers University of Technology, Gothenburg, Sweden;
[🦇]Ericsson Research, Gothenburg, Sweden; [🧹]The University of Iowa, Iowa City, IA, USA

ABSTRACT

In this paper, we solve the sum mean-squared error (MSE)-optimal 1-bit quantized precoding problem exactly for small-to-moderate sized multiuser multiple-input multiple-output (MU-MIMO) systems via branch and bound. To this end, we reformulate the original NP-hard precoding problem as a tree search and deploy a number of strategies that improve the pruning efficiency without sacrificing optimality. We evaluate the error-rate performance and the complexity of the resulting 1-bit branch-and-bound (BB-1) precoder, and compare its efficacy to that of existing, suboptimal algorithms for 1-bit precoding in MU-MIMO systems.

Index Terms— massive multiuser multiple-input multiple-output, 1-bit quantization, precoding, branch and bound

1. INTRODUCTION

Massive multiuser multiple-input multiple-output (MU-MIMO) technology, a scaled-up version of what is used in today's cellular communication systems, is expected to play a critical role in next-generation wireless systems [1]. In the downlink, the basestation (BS) transmits data to multiple users in the same time-frequency resource by mapping the information symbols to the antenna array via a precoder [2, 3]. For massive MU-MIMO systems, the increase in the number of BS antenna elements entails significant growths in circuit power consumption and interconnect bandwidth over the link connecting the baseband processing unit to the radio unit. These challenges are further aggravated when operating over large bandwidths at millimeter-wave frequencies [4].

The use of low-resolution digital-to-analog converters (DACs) has recently been proposed to reduce power consumption and mitigate the interconnect-bandwidth bottleneck at the BS. When low-resolution DACs are used, each entry of the precoded vector must be quantized to the low-cardinality alphabet that is supported by the transcoder in the DAC. For the special case of 1-bit DACs and frequency-flat channels, 1-bit precoding has been studied in, e.g., [5–12]; the frequency-selective scenario has been studied recently in [13, 14].

Unfortunately, the mean-squared error (MSE)-optimal precoding problem for the case where the precoded vector is quantized to a finite alphabet is, in general, NP-hard. By relaxing the finite-alphabet constraint to a convex set, suboptimal precoders (with near-optimal performance) have been developed [9, 10, 13]. In this paper, we solve the sum MSE-optimal 1-bit precoding *exactly* for small-to-moderate sized MU-MIMO systems (e.g., 12 BS antennas) that operate over frequency-flat channels without resorting to an exhaustive search. To this end, we reformulate the NP-hard problem as a tree search, which we then solve by the proposed 1-bit branch-and-bound (BB-1) precoding algorithm. We deploy a number of strategies that improve the pruning efficiency of BB-1 without sacrificing optimality, and we compare the error-rate performance and complexity of BB-1 to that of existing, suboptimal precoders.

2. MSE-OPTIMAL QUANTIZED PRECODING

2.1. The Quantized Precoding (QP) Problem

We consider quantized (or finite-alphabet) precoding for the MU-MIMO downlink. The BS is equipped with B antennas and serves U single-antenna users in the same time-frequency resource. The goal of the sum MSE-optimal quantized precoder is to compute a precoded vector $\mathbf{x} \in \mathcal{X}^B$, with $\mathcal{X}$ being the finite-cardinality transmit alphabet, by solving the following quantized precoding (QP) problem [9]:

$$\text{(QP)} \quad \begin{cases} \underset{\mathbf{x}\in\mathcal{X}^B,\, \beta\in\mathbb{R}}{\text{minimize}} & \|\mathbf{s}-\beta\mathbf{H}\mathbf{x}\|_2^2 + \beta^2 U N_0 \\ \text{subject to} & \|\mathbf{x}\|_2^2 \leq 1 \text{ and } \beta > 0. \end{cases}$$

Here, $\mathbf{s} \in \mathcal{O}^U$ is the (known) data vector to be transmitted to the U users, $\mathcal{O}$ is the constellation alphabet (e.g., 16-QAM), $\mathbf{H} \in \mathbb{C}^{U\times B}$ is the (known) downlink channel matrix, and N_0 is the noise variance at each user (assumed to be equal for all users and known at the BS). We define the signal-to-noise ratio (SNR) as $\mathit{SNR} = 1/N_0$. The *precoding factor* β takes into account the gain of the channel [9]. We note that for a given value of β, the (QP) problem is a closest

The work of SJ and GD was supported in part by the Swedish Foundation for Strategic Research under grant ID14-0022, and by the Swedish Governmental Agency for Innovation Systems (VINNOVA) within the competence center ChaseOn. SJ's research visit at Cornell was sponsored in part by Cornell's College of Engineering. The work of CS was supported in part by Xilinx, Inc. and by the US National Science Foundation (NSF) under grants ECCS-1408006, CCF-1535897, CAREER CCF-1652065, and CNS-1717559.

vector problem (CVP) that is NP hard [15]. As a consequence, solving (QP) via an exhaustive search requires evaluating the objective function in (QP) for $|\mathcal{X}|^B$ candidate vectors, which is infeasible for moderate-to-large B. Hence, more efficient precoding algorithms are required in practice.

2.2. Rewriting the (QP) Problem

We start by using the fact that the precoding factor $\beta > 0$ in (QP) is a continuous parameter. Hence, given a precoded vector $\mathbf{x}$ for which $\Re\{\mathbf{x}^H\mathbf{H}^H\mathbf{s}\} > 0$, the optimal associated precoding factor can be readily computed as

$$\hat{\beta}(\mathbf{x}) = \frac{\Re\{\mathbf{x}^H\mathbf{H}^H\mathbf{s}\}}{\|\mathbf{H}\mathbf{x}\|_2^2 + N_0 U}.$$

By inserting this optimal precoding factor $\hat{\beta}(\mathbf{x})$ into the objective function of (QP), we obtain

$$\|\mathbf{s} - \hat{\beta}(\mathbf{x})\mathbf{H}\mathbf{x}\|_2^2 + \hat{\beta}(\mathbf{x})^2 U N_0 = \|\mathbf{s}\|^2 - \frac{\Re\{\mathbf{x}^H\mathbf{H}^H\mathbf{s}\}^2}{\|\mathbf{H}\mathbf{x}\|_2^2 + N_0 U}.$$

Consequently, solving the problem (QP) is equivalent to solving the following optimization problem:

$$(\text{QP}^*) \quad \begin{cases} \underset{\mathbf{x}\in\mathcal{X}^B}{\text{minimize}} & \dfrac{\|\mathbf{H}\mathbf{x}\|_2^2 + N_0 U}{\Re\{\mathbf{x}^H\mathbf{H}^H\mathbf{s}\}^2} \\ \text{subject to} & \|\mathbf{x}\|_2^2 \leq 1. \end{cases}$$

Let $\hat{\mathbf{x}}$ denote the optimal solution to the problem (QP*). Note that the corresponding precoding factor $\hat{\beta}(\hat{\mathbf{x}})$ can be negative. In this case, we use that $\hat{\beta}(-\hat{\mathbf{x}}) = -\hat{\beta}(\hat{\mathbf{x}})$ to simply flip the sign of the solution $\hat{\mathbf{x}}$. For this not to affect the solution, we require that $\mathcal{X}$ is symmetric, i.e., that $x \in \mathcal{X}$ implies $-x \in \mathcal{X}$.

3. BB-1: 1-BIT BRANCH-AND-BOUND PRECODER

3.1. Simplifying (QP*) for Constant-Modulus Alphabets

To arrive at a formulation of (QP*) that is amenable to branch and bound, we triangularize the problem. For this to work, we require constant-modulus (CM) transmit alphabets $\mathcal{X}$, i.e., that $|x|^2 = 1/B$ for all $x \in \mathcal{X}$, which implies that $\|\mathbf{x}\|_2^2 = 1$. We use this property to rewrite the objective of (QP*) as follows:

$$\frac{\|\mathbf{H}\mathbf{x}\|_2^2 + N_0 U}{\Re\{\mathbf{x}^H\mathbf{H}^H\mathbf{s}\}^2} = \frac{\|\mathbf{H}\mathbf{x}\|_2^2 + N_0 U\|\mathbf{x}\|_2^2}{\Re\{\mathbf{x}^H\mathbf{H}^H\mathbf{s}\}^2} = \frac{\|\widetilde{\mathbf{H}}\mathbf{x}\|_2^2}{\Re\{\mathbf{x}^H\mathbf{z}^{\text{MRT}}\}^2}.$$

Here, $\widetilde{\mathbf{H}} = \left[\mathbf{H}^T, \sqrt{N_0 U}\,\mathbf{I}_B\right]^T$ is the $(U+B)\times B$ augmented channel matrix and $\mathbf{z}^{\text{MRT}} = \mathbf{H}^H\mathbf{s}$ is the B-dimensional maximal-ratio transmission (MRT) vector. By applying the QR factorization $\widetilde{\mathbf{H}} = \mathbf{Q}\mathbf{R}$, where $\mathbf{Q} \in \mathbb{C}^{(U+B)\times B}$ has unitary rows and $\mathbf{R} \in \mathbb{C}^{B\times B}$ is an upper-triangular matrix with nonnegative values on the main diagonal, we can formulate the CM quantized precoding problem as

$$(\text{CMQP}) \quad \underset{\mathbf{x}\in\mathcal{X}^B}{\text{minimize}} \ \frac{\|\mathbf{R}\mathbf{x}\|_2^2}{\Re\{\mathbf{x}^H\mathbf{z}^{\text{MRT}}\}^2},$$

which is sum MSE-optimal for CM transmit alphabets $\mathcal{X}$.

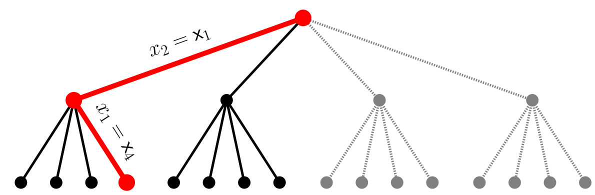

Fig. 1: Quaternary tree for the 1-bit-quantized case and for $B = 2$. Note that the root of the tree corresponds to the last entry of the vector $\mathbf{x}$, and that half of the tree can by prepruned by symmetry (see Section 4.5). The path highlighted in red corresponds to the precoded vector $\mathbf{x} = [\mathrm{x}_4, \mathrm{x}_1]^T$.

3.2. Branch-and-Bound Procedure

The branch-and-bound procedure proposed in this paper finds the optimal solution $\hat{\mathbf{x}} \in \mathcal{X}^B$ to the problem (CMQP). For the sake of brevity, we will focus exclusively on the 1-bit quantized case, where $\mathcal{X} = \{\mathrm{x}_1, \mathrm{x}_2, \mathrm{x}_3, \mathrm{x}_4\}$ and $\mathrm{x}_m = \frac{1}{\sqrt{B}} e^{j\pi(\frac{m}{2} - \frac{1}{4})}$.

The goal of using branch and bound to solve the problem (CMQP) is to reformulate it as a tree-search problem for which we can prune large parts of the tree in order to reduce the computational complexity. It is key to realize that the problem (CMQP) can be associated with a B-level $|\mathcal{X}|$-tree; see Fig. 1 for an illustration. Each node in the tree at level L can be uniquely described by the partial symbol vector (PSV) $\mathbf{x}^{(L)} = [x_L, x_{L+1}, \ldots, x_B]^T \in \mathcal{X}^{B-L+1}$. Consider branching out from a node in the tree at level $L+1$. The goals are to decide (i) which child node should be visited next and (ii) which child nodes can be pruned. To this end, we need a cost that represents the objective function in (CMQP) given a previously chosen PSV $\mathbf{x}^{(L+1)}$ and the potential child nodes $\tilde{x} \in \mathcal{X}$ so that we can prune whenever the cost associated with a node at level L exceeds some bound. With this in mind, we lower-bound the numerator of the objective function in (CMQP) as

$$\|\mathbf{R}\mathbf{x}\|^2 = \textstyle\sum_{b=1}^B \left|\sum_{k=b}^B R_{b,k} x_k\right|^2 \geq n_L\big(\tilde{x}; \mathbf{x}^{(L+1)}\big),$$

where $n_L(\tilde{x}; \mathbf{x}^{(L+1)}) \geq 0$ depends only on the potential child node $\tilde{x} \in \mathcal{X}$ and the PSV $\mathbf{x}^{(L+1)}$. Similarly, we upper-bound the denominator of the objective function in (CMQP) as

$$\Re\{\mathbf{x}^H\mathbf{z}^{\text{MRT}}\}^2 = \left(\textstyle\sum_{b=1}^B \Re\{x_b^* z_b^{\text{MRT}}\}\right)^2 \leq d_L\big(\tilde{x}; \mathbf{x}^{(L+1)}\big),$$

where z_b^{MRT} is the bth entry of $\mathbf{z}^{\text{MRT}}$ and $d_L(\tilde{x}; \mathbf{x}^{(L+1)}) \geq 0$ depends only on the potential child node $\tilde{x} \in \mathcal{X}$ and the PSV $\mathbf{x}^{(L+1)}$. Given these two quantities we can define the cost $c_L(\tilde{x}; \mathbf{x}^{(L+1)})$ as follows:

$$c_L\big(\tilde{x}; \mathbf{x}^{(L+1)}\big) = \frac{n_L\big(\tilde{x}; \mathbf{x}^{(L+1)}\big)}{d_L\big(\tilde{x}; \mathbf{x}^{(L+1)}\big)} \leq \frac{\|\mathbf{R}\mathbf{x}\|^2}{\Re\{\mathbf{x}^H\mathbf{z}^{\text{MRT}}\}^2}.$$

Whenever $c_L(\tilde{x}; \mathbf{x}^{(L+1)}) > \rho$, where $\rho > 0$ is a suitably chosen treshold (we shall discuss how to set ρ in Sections 4.1 and 4.2), we can prune the corresponding subtree. Next, we provide specific choices for $n_L(\tilde{x}; \mathbf{x}^{(L+1)})$ and $d_L(\tilde{x}; \mathbf{x}^{(L+1)})$.

3.3. Bounding The Cost Function

Given a PSV $\mathbf{x}^{(L+1)} \in \mathcal{X}^{B-L}$ and a candidate child $\tilde{x}$, we write the numerator of the cost function $c_L(\cdot,\cdot)$ as a sum of three parts—past, present, and future—as follows:

$$n_L\big(\tilde{x};\mathbf{x}^{(L+1)}\big) = n_L^{\text{past}}(\mathbf{x}^{(L+1)}) + n_L^{\text{present}}\big(\tilde{x};\mathbf{x}^{(L+1)}\big) + n_L^{\text{future}}\big(\tilde{x};\mathbf{x}^{(L+1)}\big).$$

The past is determined by the previously chosen PSV $\mathbf{x}^{(L+1)}$ and is given by $n_L^{\text{past}}(\mathbf{x}^{(L+1)}) = \sum_{b=L+1}^{B}\big|\sum_{\ell=b}^{B} R_{b,\ell}x_\ell\big|^2$. The present depends on the choice of the child node $\tilde{x} \in \mathcal{X}$ and on the PSV $\mathbf{x}^{(L+1)}$ and is given by $n_L^{\text{present}}(\tilde{x};\mathbf{x}^{(L+1)}) = \big|R_{L,L}\tilde{x} + \sum_{\ell=L+1}^{B} R_{L,\ell}x_\ell\big|^2$. The future depends on the cost of all possible leaf nodes and is given by

$$n_L^{\text{future}}\big(\tilde{x};\mathbf{x}^{(L+1)}\big) = \min_{\bar{\mathbf{x}}\in\mathcal{X}^{L-1}} \sum_{b=1}^{L-1}\Big|\sum_{\ell=b}^{L-1} R_{b,\ell}\bar{x}_\ell + R_{b,L}\tilde{x} + \sum_{\ell=L+1}^{B} R_{b,\ell}x_\ell\Big|^2.$$

Unfortunately, computing the future cost exactly is as hard as solving the original precoding problem. A trivial lower bound is obtained by setting $n_L^{\text{future}}(\tilde{x};\mathbf{x}^{(L+1)}) = 0$, which results, however, in a poor pruning behavior. In Section 4.4, we provide a more sophisticated approach that improves the pruning behavior.

Using a similar approach, we decompose the denominator of the cost associated with branching out from a node at level $L+1$ into three parts: past, present, and future. To arrive at an upper bound on the denominator of the cost function, we use the triangle inequality to bound $\big(\sum_{b=1}^{B}\Re\{x_b^* z_b^{\text{MRT}}\}\big)^2$ by

$$d_L(\tilde{x};\mathbf{x}^{(L+1)}) = \Big(\big|d_L^{\text{past}}(\mathbf{x}^{(L+1)}) + d_L^{\text{present}}(\tilde{x})\big| + \big|d_L^{\text{future}}\big|\Big)^2,$$

where the past and present are given by $d_L^{\text{past}}(\mathbf{x}^{(L+1)}) = \sum_{\ell=L+1}^{B}\Re\{x_\ell^* z_\ell^{\text{MRT}}\}$ and $d_L^{\text{present}}(\tilde{x}) = \Re\{\tilde{x}^* z_L^{\text{MRT}}\}$, respectively. Finally, the future cost is

$$d_L^{\text{future}} = \max_{\bar{\mathbf{x}}\in\mathcal{X}^{L-1}} \sum_{\ell=1}^{L-1}\Re\{\bar{x}_\ell^* z_\ell^{\text{MRT}}\}.$$

It can be shown that the maximum is achieved by $\bar{x}_\ell = x_\ell^{\text{MRT}}$, where $x_\ell^{\text{MRT}} = \text{argmin}_{x\in\mathcal{X}}|x - z_\ell^{\text{MRT}}|^2$.

4. FIVE TRICKS THAT MAKE BB-1 FASTER

We now propose five tricks that improve the pruning behavior of the proposed algorithm without sacrificing optimality.

4.1. Trick 1: Depth-First Best-First Tree Traversal with Radius Reduction

We traverse the search tree in the following manner: at level $L+1$, we pick the $\tilde{x}$ that minimizes the current cost $c_L(\tilde{x};\mathbf{x}^{(L+1)})$; we then proceed in a depth-first manner. Whenever a valid leaf node $\mathbf{x}^{(1)}$ is found, we update the radius (bound) to

$$\rho \leftarrow \frac{\|\mathbf{R}\mathbf{x}^{(1)}\|_2^2}{\Re\big\{(\mathbf{x}^{(1)})^H\mathbf{z}^{\text{MRT}}\big\}^2}$$

and we perform backtracking by proceeding upwards and selecting the next-best symbol, excluding branches that have been explored or with a cost that exceeds the new radius.

Remark 1 *Any other tree-traversal strategy could be used, such as breadth-first used in the (suboptimal) K-best algorithm that can be implemented efficiently in hardware [16, 17].*

4.2. Trick 2: Radius Initialization

The pruning efficiency can be improved by initializing the tree search with some radius $\rho < \infty$, which is sufficiently large not to exclude the optimal solution [18]. We initialize the radius using the Wiener-filter (WF) solution, which can be computed at low complexity and can be shown to be optimal in the low-SNR regime. Specifically, we initialize

$$\rho = \frac{\|\mathbf{R}\mathbf{x}^{\text{WF}}\|_2^2}{\Re\big\{\big(\mathbf{x}^{\text{WF}}\big)^H\mathbf{z}^{\text{MRT}}\big\}^2}.$$

Here, the bth entry of $\mathbf{x}^{\text{WF}}$ is $x_b^{\text{WF}} = \text{argmin}_{x\in\mathcal{X}}|x - z_b^{\text{WF}}|^2$ where z_b^{WF} is the bth entry of $\mathbf{z}^{\text{WF}} = \mathbf{H}^H(\mathbf{H}\mathbf{H}^H + UN_0\mathbf{I}_U)^{-1}$.

4.3. Trick 3: Sorted QR Decomposition

We permute the columns of $\widetilde{\mathbf{H}}$ (and the corresponding entries in $\mathbf{x}$) using the sorted-QR-decomposition algorithm put forward in [19], so that the diagonal elements of $\mathbf{R}$ are sorted in ascending order. This approach improves substantially the pruning behavior for nodes close to the root because larger part of the search tree can be pruned early on.

4.4. Trick 4: Predicting the Future

The pruning efficiency can further be improved by finding a nontrivial lower bound on $n_L^{\text{future}}(x_L;\mathbf{x}^{(L+1)})$. We denote by $\mathbf{R}_{L-1} \in \mathbb{C}^{L-1\times L-1}$ the submatrix of $\mathbf{R}$ whose entry on the bth row ($b = 1,2,\ldots,L-1$) and on the ℓth column ($\ell = 1,2,\ldots,L-1$) is $R_{b,\ell}$ Furthermore, we denote by $\mathbf{b}_L(\tilde{x};\mathbf{x}^{(L+1)}) \in \mathbb{C}^{L-1}$ the vector whose entry on the bth row ($b = 1,2,\ldots,L-1$) is $R_{b,L}\tilde{x} + \sum_{\ell=L+1}^{B} R_{b,\ell}x_\ell$. With these definitions, we find a lower bound on $n_L^{\text{future}}(x_L;\mathbf{x}^{(L+1)})$ using the eigenbound technique in [20, Sec. VII]. Specifically,

$$n_L^{\text{future}}\big(\tilde{x};\mathbf{x}^{(L+1)}\big) \geq \lambda_{L-1}^{\min}(\mathbf{R}_{L-1}^H\mathbf{R}_{L-1}) \times \min_{\bar{\mathbf{x}}\in\mathcal{X}^{L-1}}\|\bar{\mathbf{x}} - \mathbf{R}_{L-1}^{-1}\mathbf{b}_L(\tilde{x};\mathbf{x}^{(L+1)})\|_2^2.$$

Here, $\lambda_{L-1}^{\min}(\mathbf{R}_{L-1}^H\mathbf{R}_{L-1})$ is the smallest eigenvalue of the Gram matrix $\mathbf{R}_{L-1}^H\mathbf{R}_{L-1}$. The vector $\bar{\mathbf{x}} \in \mathcal{X}^{L-1}$ that achieves the minimum is readily obtained by quantizing $\mathbf{R}_{L-1}^{-1}\mathbf{b}_L(\tilde{x};\mathbf{x}^{(L+1)})$ to the nearest vector in the set $\mathcal{X}^{L-1}$.

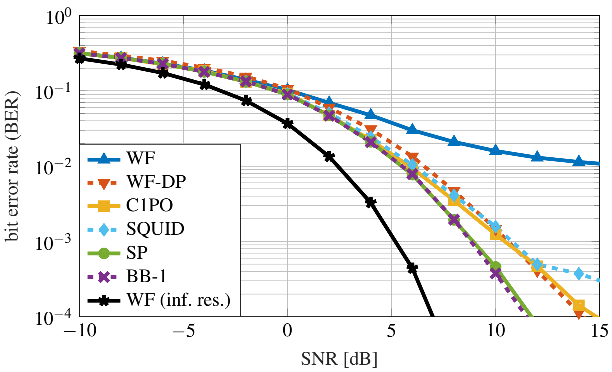

Fig. 2: Uncoded BER with QPSK; $B = 12$ and $U = 3$. Low uncoded BERs can be achieved with the BB-1 precoder and by recently developed state-of-the-art 1-bit precoders.

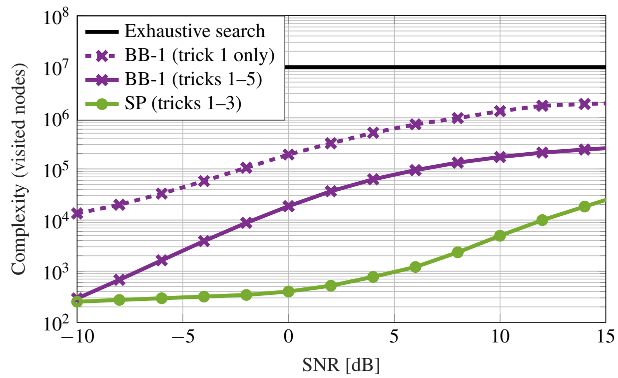

Fig. 3: Complexity of BB-1 with and without the five tricks proposed in Section 4; QPSK, $B = 12$, and $U = 3$.

4.5. Trick 5: Preprune the Search Tree

For $\mathcal{X} = \{\mathsf{x}_1, \mathsf{x}_2, \mathsf{x}_3, \mathsf{x}_4\}$ as in Section 3.2, large parts of the search tree are redundant since, by symmetry, it holds that $-\mathsf{x}_1 = \mathsf{x}_3$ and that $-\mathsf{x}_2 = \mathsf{x}_4$. Therefore, we can preprune the search tree by excluding symmetric solutions without sacrificing optimality. As illustrated in Fig. 1, we exclude all branches (and corresponding subtrees) stemming from $x_B \in \{\mathsf{x}_3, \mathsf{x}_4\}$.

5. SIMULATION RESULTS

We now investigate the bit error rate (BER) and the complexity of the proposed BB-1 precoder. For the sake of brevity, we focus on a limited set of system parameters.[1] Specifically, we use $B = 12$ BS antennas and $U = 3$ users. We consider Rayleigh fading, i.e., the entries of $\mathbf{H}$ are i.i.d. complex circularly symmetric Gaussian distributed with unit variance.

5.1. BER Performance

In Fig. 2, we plot the uncoded BER with QPSK for BB-1 as a function of the SNR. For comparison, we also evaluate the BER of state-of-the-art 1-bit precoders. Specifically, we consider WF precoding, WF precoding with direct perturbation (WF-DP) [7, Sec. 3.1], convex 1-bit precoding (C1PO) [10, Sec. III-C], squared-infinity norm Douglas-Rachford splitting (SQUID) [9, Sec. IV-B], and sphere precoding (SP) [9, Sec. IV-C]. We also show the BER with WF precoding for the infinite-resolution (no quantization) case. First of all, we note that low uncoded BER can be achieved with BB-1. Indeed, the gap to infinite-resolution performance is only 4 dB for a target BER of 10^{-3}. We further note that several of the state-of-the-art precoders, which can be implemented at low computational complexity (see, e.g, [10, Sec. VI-D]), perform close to the BB-1 precoder for low-to-moderate SNR values.

[1] To explore other system configurations, our simulation framework is available on GitHub (https://github.com/quantizedmassivemimo/1bit_precoding).

5.2. Complexity Impact of the Five Tricks

In Fig. 3, we show the complexity (measured in terms of the number of nodes visited during a tree search) as a function of the SNR, with and without the tricks presented in Section 4. We also show the complexity for exhaustive search, for which $\frac{7}{3} \cdot 4^{B-1} - \frac{4}{3}$ nodes are visited during a tree search, and for SP. We note that by traversing the tree as in Section 4.1, BB-1 has to visit orders-of-magnitude fewer nodes compared to an exhaustive search, especially at moderate to low SNR values. Indeed, if N_0 is small, then the augmented channel matrix $\widetilde{\mathbf{H}}$ is ill-conditioned and many eigenvalues of the Gram matrix are small, resulting in poor pruning behavior.

By using the tricks proposed in Section 4.2–Section 4.5, the complexity of BB-1 is further reduced drastically. Note that the tricks in Sections 4.1 to 4.3 can be used also for SP. Further note that the complexity of SP, which delivers near-optimal performance (cf. Fig. 2), is noticeably lower than that of BB-1.

6. CONCLUSIONS AND USES OF BB-1

We have shown how the sum MSE-optimal 1-bit precoding problem can be transformed into a tree search, which is solved exactly for small-to-moderate sized MU-MIMO systems via branch and bound. The resulting BB-1 precoder can be used as a benchmark for other precoding algorithms. Note that while we focused on 1-bit precoding and QPSK, the proposed branch-and-bound procedure can be applied for any CM transmit alphabet and for any constellation (e.g., 16-QAM).

We have also shown how the complexity of the branch-and-bound procedure can be significantly reduced by five tricks. To further reduce complexity at the cost of optimality, other tree-traversal strategies (e.g., K-best) can be used. For millimeter-wave applications where the channel remains constant over multiple symbol intervals, a practical implementation of BB-1 may involve precomputing the precoded vectors for all $|\mathcal{O}|^U$ possible symbol realizations, and storing them in a codebook.

7. REFERENCES

[1] F. Boccardi, R. W. Heath Jr., A. Lozano, T. L. Marzetta, and P. Popovski, “Five disruptive technology directions for 5G,” *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 74–80, Feb. 2014.

[2] Q. H. Spencer, C. B. Peel, A. L. Swindlehurst, and M. Haardt, “An introduction to the multi-user MIMO downlink,” *IEEE Commun. Mag.*, vol. 42, no. 10, pp. 60–67, Oct. 2004.

[3] F. Rusek, D. Persson, B. Kiong, E. G. Larsson, T. L. Marzetta, O. Edfors, and F. Tufvesson, “Scaling up MIMO: Opportunities and challenges with very large large arrays,” *IEEE Signal Process. Mag.*, vol. 30, no. 1, pp. 40–60, Jan. 2013.

[4] R. W. Heath Jr., N. González Prelcic, S. Rangan, W. Roh, and A. Sayeed, “An overview of signal processing techniques for millimeter wave MIMO systems,” *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 3, pp. 436–453, Feb. 2016.

[5] A. Mezghani, R. Ghiat, and J. A. Nossek, “Transmit processing with low resolution D/A-converters,” in *Proc. IEEE Int. Conf. Electron., Circuits, Syst. (ICECS)*, Yasmine Hammamet, Tunisia, Dec. 2009, pp. 683–686.

[6] H. Jedda, J. A. Nossek, and A. Mezghani, “Minimum BER precoding in 1-bit massive MIMO systems,” in *IEEE Sensor Array and Multichannel Signal Process. Workshop (SAM)*, Rio de Janeiro, Brazil, Jul. 2016.

[7] A. L. Swindlehurst, A. K. Saxena, A. Mezghani, and I. Fijalkow, “Minimum probability-of-error perturbation precoding for the one-bit massive MIMO downlink,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, New Orleans, LA, USA, Mar. 2017, pp. 6483–6487.

[8] L. Landau and R. C. de Lamare, “Branch-and-bound precoding for multiuser MIMO systems with 1-bit quantization,” *IEEE Wireless Commun. Lett.*, vol. 6, no. 6, pp. 770–773, Dec. 2017.

[9] S. Jacobsson, G. Durisi, M. Coldrey, T. Goldstein, and C. Studer, “Quantized precoding for massive MU-MIMO,” *IEEE Trans. Commun.*, vol. 65, no. 11, pp. 4670–4684, Nov. 2017.

[10] O. Castañeda, S. Jacobsson, G. Durisi, M. Coldrey, T. Goldstein, and C. Studer, “1-bit massive MU-MIMO precoding in VLSI,” *IEEE J. Emerging Sel. Topics Circuits Syst.*, vol. 7, no. 4, pp. 508–522, Dec. 2017.

[11] Y. Li, C. Tao, A. L. Swindlehurst, A. Mezghani, and L. Liu, “Downlink achievable rate analysis in massive MIMO systems with one-bit DACs,” *IEEE Commun. Lett.*, vol. 21, no. 7, pp. 1669–1672, Jul. 2017.

[12] O. Tirkkonen and C. Studer, “Subset-codebook precoding for 1-bit massive multiuser MIMO,” in *Conf. Inf. Sciences Syst. (CISS)*, Baltimore, MD, USA, Mar. 2017.

[13] S. Jacobsson, O. Castañeda, C. Jeon, G. Durisi, and C. Studer, “Nonlinear precoding for phase-quantized constant-envelope massive MU-MIMO-OFDM,” Jan. 2018. [Online]. Available: https://arxiv.org/abs/1710.06825

[14] S. Jacobsson, G. Durisi, M. Coldrey, and C. Studer, “Massive MU-MIMO-OFDM downlink with one-bit DACs and linear precoding,” in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Singapore, Singapore, Dec. 2017.

[15] S. Verdú, “Computational complexity of multiuser detection,” *Algorithmica*, vol. 4, no. 1, pp. 303–312, 1989.

[16] K.-W. Wong, C.-Y. Tsui, R. S.-K. Cheng, and W.-H. Mow, “A VLSI architecture of a K-best lattice decoding algorithm for MIMO channels,” in *Proc. IEEE Int. Symp. Circuits and Syst. (ISCAS)*, vol. 3, Phoenix, AZ, USA, May 2002, pp. 273–276.

[17] M. Wenk, M. Zellweger, A. Burg, N. Relber, and W. Fichtner, “K-best MIMO detection VLSI architectures achieving up to 424 Mbps,” in *Proc. IEEE Int. Symp. Circuits and Syst. (ISCAS)*, Kos, Greece, May 2006, pp. 1151–1154.

[18] A. Burg, M. Borgmann, M. Wenk, M. Zellweger, W. Fichtner, and H. Bölcskei, “VLSI implementation of MIMO detection using the sphere decoding algorithm,” vol. 40, no. 7, pp. 1566–1577, Jul. 2005.

[19] D. Wübben, R. Böhnke, 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.

[20] M. Stojnic, H. Vikalo, and B. Hassibi, “Speeding up the sphere decoder with H^∞ and SDP inspired lower bounds,” *IEEE Trans. Signal Process.*, vol. 56, no. 2, pp. 712–726, Feb. 2008.