Finite-Alphabet Wiener Filter Precoding
for mmWave Massive MU-MIMO Systems

Oscar Castañeda[1], Sven Jacobsson[2,3], Giuseppe Durisi[3], Tom Goldstein[4], and Christoph Studer[1]

[1]*Cornell Tech, New York, NY; e-mail: oc66@cornell.edu, studer@cornell.edu*
[2]*Ericsson Research, Gothenburg, Sweden; e-mail: sven.jacobsson@ericsson.com*
[3]*Chalmers University of Technology, Gothenburg, Sweden; e-mail: durisi@chalmers.se*
[4]*University of Maryland, College Park, MD; e-mail: tomg@cs.umd.edu*

Abstract—**Power consumption of multi-user (MU) precoding is a major concern in all-digital massive MU multiple-input multiple-output (MIMO) base-stations with hundreds of antenna elements operating at millimeter-wave (mmWave) frequencies. We propose to replace part of the linear Wiener filter (WF) precoding matrix by a finite-alphabet WF precoding (FAWP) matrix, which enables the use of low-precision hardware that consumes low power and area. To minimize the performance loss of our approach, we present methods that efficiently compute FAWP matrices that best mimic the WF precoder. Our results show that FAWP matrices approach infinite-precision error-rate and error-vector magnitude performance with only 3-bit precoding weights, even when operating in realistic mmWave channels. Hence, FAWP is a promising approach to substantially reduce power consumption and silicon area in all-digital mmWave massive MU-MIMO systems.**

I. Introduction

Next-generation wireless systems are expected to achieve unprecedentedly high data rates by combining the large bandwidths available at millimeter-wave (mmWave) frequencies [1] with the high spectral efficiency provided by massive multi-user (MU) multiple-input multiple-output (MIMO) [2]. Unfortunately, base-station (BS) architectures for MU-MIMO systems, with hundreds of antenna elements, operating at the extreme sampling rates needed for wideband mmWave communication require excessively high power consumption and complex digital circuitry. To keep the power consumption within acceptable bounds, research has mostly focused on hybrid analog-digital architectures [3]–[5]. Such hybrid BS architectures are, however, limited in their beamforming capabilities [5]–[7], which leads to reduced spectral efficiency. Per contra, all-digital BS architectures [8]–[10] do not suffer from such limitations. While it is natural to believe that all-digital solutions are more power-hungry than hybrid architectures, recent results show that—by reducing the data-converter resolution—the radio-frequency

circuitry and data-converters in an all-digital BS (i) have similar power consumption as in a hybrid BS [7] and (ii) enable superior spectral efficiency [9]. Besides these recent results, the power consumption and silicon area of baseband processing in all-digital BS architectures are largely unexplored.

A. Finite-Alphabet Equalization

In our recent paper [11], we investigated the power consumption and silicon area required for spatial equalization in the mmWave MU-MIMO uplink, i.e., when the user equipments (UEs) transmit to the BS. We considered a system with 16 UEs and 256 BS antenna elements operating at a sampling rate of 2 G vectors/s. For such system, our implementation results in 28 nm CMOS technology showed that, even when considering data converters with only 7 bits of resolution, a simple, single-tap linear equalizer already requires 28 W and 129 mm^2 [11]. For higher sampling rates or systems with more BS antenna elements, UEs, or taps, power consumption and area will increase even further. Hence, to reduce both power and silicon area, we proposed *finite-alphabet equalization* [11], which uses coarsely quantized numbers to represent the entries of the equalization matrices, while minimizing the post-equalization mean-square error (MSE). In summary, we showed that finite-alphabet equalizers enable a reduction in power and area by a factor of 3.9× and 5.8×, respectively, while offering competitive error-rate performance to conventional, high-resolution equalizers [11].

B. Contributions

Similar to the case of equalization in the uplink, the power consumption and silicon area of precoding in the all-digital mmWave MU-MIMO downlink (BS transmits to UEs) is expected to be a major bottleneck, as high-dimensional data has to be processed at extremely high rates. In order to reduce power consumption and silicon area of the precoding operation, we apply the concept of finite-alphabet matrices used for linear spatial equalization in [11] to linear precoding. We propose two finite-alphabet precoding schemes to compute Wiener filter (WF)-optimal matrices, which are matrices that best mimic the linear WF precoder. To demonstrate the effectiveness of the framework that we call *finite-alphabet WF precoding* (FAWP), we evaluate its performance in terms of uncoded bit error-rate (BER) and error-vector magnitude (EVM) for i.i.d. Rayleigh fading, and for line-of-sight (LoS) and non-LoS mmWave channels.

C. Notation

Uppercase and lowercase boldface letters denote matrices and column vectors, respectively. For a matrix $\mathbf{A}$, its Hermitian transpose, Frobenius norm, real part and imaginary part are $\mathbf{A}^H$, $\|\mathbf{A}\|_F$, $\Re\{\mathbf{A}\}$, and $\Im\{\mathbf{A}\}$, respectively. The $M \times M$ identity matrix is $\mathbf{I}_M$. For the vector $\mathbf{a}$, its kth entry, ℓ_2-norm, and entry-wise complex conjugate are a_k, $\|\mathbf{a}\|_2$, and $\mathbf{a}^*$, respectively. The kth standard basis vector is $\mathbf{e}_k$. The set $\mathbb{R}_+$ contains the non-negative real numbers. The signum function $\mathrm{sgn}(\cdot)$ is defined as $\mathrm{sgn}(a) = +1$ for $a \in \mathbb{R}_+$ and $\mathrm{sgn}(a) = -1$ for $a \notin \mathbb{R}_+$, and is applied entry-wise to vectors. The expectation operator with respect to the random vector $\mathbf{x}$ is $\mathbb{E}_{\mathbf{x}}[\cdot]$.

II. System Model and WF Precoding

A. System Model

We focus on the downlink of a mmWave massive MU-MIMO system in which a BS with B antennas serves $U < B$ single-antenna UEs in the same time-frequency resource. We consider a narrowband scenario modeled by $\mathbf{y} = \mathbf{Hx} + \mathbf{n}$, where $\mathbf{y} \in \mathbb{C}^U$ is the received vector, $\mathbf{H} \in \mathbb{C}^{U \times B}$ is the channel matrix, $\mathbf{x} \in \mathbb{C}^B$ is the precoded vector, and $\mathbf{n} \in \mathbb{C}^U$ is i.i.d. circularly-symmetric complex Gaussian noise with variance N_0 per complex entry. We assume that the channel matrix $\mathbf{H}$ is perfectly known to the BS, and that the precoded vector $\mathbf{x}$ is subject to the following average power constraint:

$$\mathbb{E}_{\mathbf{x}}\left[\|\mathbf{x}\|_2^2\right] \leq P. \tag{1}$$

B. WF Precoding

The goal of precoding is to simultaneously transmit constellation points $s_u \in \mathcal{O}$ to the $u = 1, \dots, U$ UEs while reducing MU interference. Here, s_u is assumed to have zero mean and variance E_s, and $\mathcal{O}$ denotes the constellation set (e.g., 16-QAM). The BS maps the vector $\mathbf{s}$ into the precoded vector $\mathbf{x}$ with the aid of channel state information. The precoded vector $\mathbf{x}$ is crafted such that the UEs can form an estimate $\hat{s}_u \in \mathbb{C}$ of the transmitted symbol s_u simply by scaling the received signal y_u. Specifically, as in [12], [13], we assume that each UE forms an estimate as $\hat{s}_u = \beta y_u$. Here, $\beta \in \mathbb{R}_+$ is a precoding factor that can be estimated at the UE using pilot-based transmission [13].

In what follows, we focus on linear precoders for which it holds that $\mathbf{x} = \mathbf{Ps}$, where $\mathbf{P} \in \mathbb{C}^{B \times U}$ is the precoding matrix. While the literature covers a range of optimization criteria for linear precoding [14], in this work we limit ourselves to the design of linear precoders that attempt to minimize the MSE between the estimated symbols $\hat{\mathbf{s}}$ and the transmitted symbols $\mathbf{s}$:

$$MSE = \mathbb{E}_{\mathbf{s},\mathbf{n}}\left[\|\mathbf{s} - \hat{\mathbf{s}}\|_2^2\right] = \mathbb{E}_{\mathbf{s}}\left[\|\mathbf{s} - \beta\mathbf{Hx}\|_2^2\right] + \beta^2 U N_0. \tag{2}$$

Minimizing (2) subject to the power constraint in (1) results in the so-called WF precoder [15], where the precoding matrix is given by $\mathbf{P}^{\mathrm{WF}} = \frac{1}{\beta^{\mathrm{WF}}}\mathbf{Q}^{\mathrm{WF}}$ with

$$\mathbf{Q}^{\mathrm{WF}} = \left(\mathbf{H}^H\mathbf{H} + \kappa^{\mathrm{WF}}\mathbf{I}_B\right)^{-1}\mathbf{H}^H, \tag{3}$$

$$\kappa^{\mathrm{WF}} = \frac{U N_0}{P}, \quad \text{and} \quad \beta^{\mathrm{WF}} = \sqrt{\frac{\mathrm{tr}\left((\mathbf{Q}^{\mathrm{WF}})^H\mathbf{Q}^{\mathrm{WF}}\right)E_s}{P}}. \tag{4}$$

It is important to realize that the matrix $\mathbf{Q}^{\mathrm{WF}} \in \mathbb{C}^{B \times U}$ in (3) is the solution of the following optimization problem:

$$\mathbf{Q}^{\mathrm{WF}} = \underset{\tilde{\mathbf{Q}} \in \mathbb{C}^{B \times U}}{\arg\min} \|\mathbf{I}_U - \mathbf{H}\tilde{\mathbf{Q}}\|_F^2 + \kappa^{\mathrm{WF}}\|\tilde{\mathbf{Q}}\|_F^2. \tag{5}$$

We can also obtain the columns $\mathbf{q}_u^{\mathrm{WF}} \in \mathbb{C}^B$, $u = 1, \dots, U$, of the matrix $\mathbf{Q}^{\mathrm{WF}}$ by solving

$$\mathbf{q}_u^{\mathrm{WF}} = \underset{\tilde{\mathbf{q}} \in \mathbb{C}^B}{\arg\min} \|\mathbf{e}_u - \mathbf{H}\tilde{\mathbf{q}}\|_2^2 + \kappa^{\mathrm{WF}}\|\tilde{\mathbf{q}}\|_2^2. \tag{6}$$

By applying the Woodbury identity [16] to (3), we also have that

$$\mathbf{Q}^{\mathrm{WF}} = \mathbf{H}^H\left(\mathbf{H}\mathbf{H}^H + \kappa^{\mathrm{WF}}\mathbf{I}_U\right)^{-1}, \tag{7}$$

which is the solution to the following optimization problem:

$$\mathbf{Q}^{\mathrm{WF}} = \underset{\tilde{\mathbf{Q}} \in \mathbb{C}^{B \times U}}{\arg\min} \|\mathbf{I}_B - \tilde{\mathbf{Q}}\mathbf{H}\|_F^2 + \kappa^{\mathrm{WF}}\|\tilde{\mathbf{Q}}\|_F^2. \tag{8}$$

Thus, the rows $\mathbf{q}_b^{\mathrm{r,WF}}$, $b = 1, \dots, B$, of $\mathbf{Q}^{\mathrm{WF}}$ (where the superscript r denotes a row vector) can be computed as

$$\mathbf{q}_b^{\mathrm{r,WF}} = \underset{\tilde{\mathbf{q}}^{\mathrm{r}} \in \mathbb{C}^{1 \times U}}{\arg\min} \|\mathbf{e}_b^H - \tilde{\mathbf{q}}^{\mathrm{r}}\mathbf{H}\|_2^2 + \kappa^{\mathrm{WF}}\|\tilde{\mathbf{q}}^{\mathrm{r}}\|_2^2. \tag{9}$$

The alternative optimization problems in (6) and (9) to compute the matrix $\mathbf{Q}^{\mathrm{WF}}$ will become useful in the next section.

III. Finite-Alphabet WF Precoding (FAWP)

WF precoding computes $\mathbf{x} = \mathbf{P}^{\mathrm{WF}}\mathbf{s} = \frac{1}{\beta^{\mathrm{WF}}}\mathbf{Q}^{\mathrm{WF}}\mathbf{s}$ for each transmitted vector $\mathbf{s}$. Unfortunately, digital precoding circuitry will be power hungry and large as mmWave MU-MIMO systems operate with high-dimensional data and extremely high sampling rates. As a remedy, FAWP proposes to represent the matrix $\mathbf{Q}^{\mathrm{WF}}$ using coarsely quantized numbers, with the objective of reducing the hardware complexity of the matrix-vector product $\mathbf{Q}^{\mathrm{WF}}\mathbf{s}$. Unfortunately, a direct quantization of the matrix $\mathbf{Q}^{\mathrm{WF}}$ typically leads to a significant error-rate degradation.

In order to design low-resolution matrices that are WF-optimal, i.e., that best mimic the infinite-precision WF-precoding matrix $\mathbf{Q}^{\mathrm{WF}}$, we propose to use the so-called *finite-alphabet matrices*, initially proposed in [11] for spatial equalization in the mmWave MU-MIMO uplink. Since we will apply finite-alphabet matrices to imitate the WF-precoding matrix $\mathbf{Q}^{\mathrm{WF}}$, we will refer to them as *FAWP matrices*. FAWP matrices introduce a *few* high-resolution scaling factors that help to bring a low-resolution matrix to the right scale. While the work in [11] studied one form of finite-alphabet matrices, we will now consider two distinct FAWP matrix structures, namely pre-FAWP and post-FAWP matrices.

A. Pre-FAWP Matrix

Definition 1. *We define a* pre-FAWP matrix *as a $B \times U$ matrix with the structure*

$$\mathbf{Q} = \mathbf{A}\,\mathrm{diag}(\boldsymbol{\alpha}^*), \tag{10}$$

where $\mathbf{A} \in \mathcal{X}^{B \times U}$ is a low-resolution matrix with entries taken from the finite alphabet $\mathcal{X}$ and $\boldsymbol{\alpha} \in \mathbb{C}^U$ is a vector with per-UE scaling factors.

By using a pre-FAWP matrix, the matrix-vector product $\mathbf{Qs}$ becomes $\mathbf{A}(\mathrm{diag}(\boldsymbol{\alpha}^*)\mathbf{s})$. We call such matrix pre-FAWP as the U

179

entries of the transmitted symbol vector $\mathbf{s}$ are scaled by the entries of $\boldsymbol{\alpha}^*$ *before* getting multiplied with the matrix $\mathbf{A}$. Pre-FAWP reduces hardware complexity of $\mathbf{Qs}$ since the matrix $\mathbf{A}$ has low-resolution entries. Consider, for example, the case in which the entries of $\mathbf{A}$ are chosen from the 1-bit alphabet $\mathcal{X} = \{\pm 1 \pm j\}$; multiplying this matrix $\mathbf{A}$ with the vector $\operatorname{diag}(\boldsymbol{\alpha}^*)\mathbf{s}$ does not require hardware multipliers, but only adders and subtractors.

To calculate pre-FAWP matrices that are WF-optimal, we solve the problem in (6) by assuming that $\mathbf{Q}$ has the form given by (10). By doing so, we arrive at the following procedure:

Lemma 1. *The problem in (5) is equivalent to solving the following optimization problem for each UE $u = 1, \ldots, U$:*

$$\mathbf{a}_u = \underset{\tilde{\mathbf{a}} \in \mathcal{X}^B}{\arg \min} \frac{\|\mathbf{H}\tilde{\mathbf{a}}\|_2^2 + \kappa^{\mathrm{WF}}\|\tilde{\mathbf{a}}\|_2^2}{|\mathbf{h}_u^{\mathrm{r}}\tilde{\mathbf{a}}|^2}. \quad (11)$$

Here, $\mathbf{a}_u$ is the uth column of $\mathbf{A}$, $\mathbf{h}_u^{\mathrm{r}}$ is the uth row of $\mathbf{H}$, and the associated optimal scaling factor is given by

$$\alpha_u(\mathbf{a}_u) = \frac{\mathbf{h}_u^{\mathrm{r}}\mathbf{a}_u}{\|\mathbf{H}\mathbf{a}_u\|_2^2 + \kappa^{\mathrm{WF}}\|\mathbf{a}_u\|_2^2}. \quad (12)$$

Lemma 1 can be established by first plugging (10) into (6). Then, we obtain (12) by taking the Wirtinger derivative with respect to α_u. Substituting (12) in (6) gives (11); the proof is analogous to that in [11] for finite-alphabet equalizers.

B. Post-FAWP Matrix

Definition 2. *We define a* post-FAWP *matrix as a $B \times U$ matrix with the structure*

$$\mathbf{Q} = \operatorname{diag}(\boldsymbol{\zeta})\,\mathbf{Z}^H, \quad (13)$$

where $\mathbf{Z} \in \mathcal{X}^{U \times B}$ is a low-resolution matrix with entries taken from the finite alphabet $\mathcal{X}$ and $\boldsymbol{\zeta} \in \mathbb{C}^B$ is a vector with per-BS-antenna scaling factors.

By using a post-FAWP matrix, the matrix-vector product $\mathbf{Qs}$ becomes $\operatorname{diag}(\boldsymbol{\zeta})(\mathbf{Z}^H\mathbf{s})$. We call such matrix post-FAWP as the B scaling factors in $\boldsymbol{\zeta}$ are applied *after* multiplying the matrix $\mathbf{Z}^H$ with the vector $\mathbf{s}$. Post-FAWP reduces the hardware complexity of $\mathbf{Qs}$ since the $B \times U$ matrix-vector product $\mathbf{Z}^H\mathbf{s}$ can be implemented using exclusively low-resolution arithmetic units. The results of $\mathbf{Z}^H\mathbf{s}$ are then entry-wise scaled by $\boldsymbol{\zeta}$, which requires only B high-resolution scalar multiplications.

Akin to the case of pre-FAWP matrices, we obtain post-FAWP matrices that are WF-optimal by solving the problem in (9) with a matrix $\mathbf{Q}$ that has the form given in (13). By doing so, we arrive at the following procedure:

Lemma 2. *The problem in (8) is equivalent to solving the following optimization problem for each BS antenna $b = 1, \ldots, B$:*

$$\mathbf{z}_b = \underset{\tilde{\mathbf{z}} \in \mathcal{X}^U}{\arg \min} \frac{\|\mathbf{H}^H\tilde{\mathbf{z}}\|_2^2 + \kappa^{\mathrm{WF}}\|\tilde{\mathbf{z}}\|_2^2}{|\mathbf{h}_b^H\tilde{\mathbf{z}}|^2}. \quad (14)$$

Here, $\mathbf{z}_b$ is the bth column of $\mathbf{Z}$, $\mathbf{h}_b$ is the bth column of $\mathbf{H}$, and the associated optimal scaling factor is given by

$$\zeta_b(\mathbf{z}_b) = \frac{\mathbf{h}_b^H\mathbf{z}_b}{\|\mathbf{H}^H\mathbf{z}_b\|_2^2 + \kappa^{WF}\|\mathbf{z}_b\|_2^2}. \quad (15)$$

The proof of Lemma 2 parallels that of Lemma 1.

In summary, both pre-FAWP and post-FAWP matrices are composed by a low-resolution matrix and a set of scaling factors. The difference is that a pre-FAWP matrix applies its U scaling factors *before* the multiplication with the low-resolution matrix, whereas a post-FAWP matrix applies its B scaling factors *after* matrix multiplication. As $B \gg U$ in typical massive MU-MIMO systems, a pre-FAWP matrix performs fewer high-resolution scaling operations than a post-FAWP matrix. However, the matrix-vector product is simpler with a post-FAWP matrix than with a pre-FAWP matrix, since the vector has a lower resolution as the symbols in $\mathbf{s}$ are not scaled yet. Thus, neither pre-FAWP nor post-FAWP matrices have a clear advantage over the other in terms of hardware complexity.[1] Nonetheless, both FAWP matrix structures are expected to reduce hardware complexity when compared to traditional precoding, as the low-resolution matrices in both structures have *coarsely quantized* entries.

IV. COMPUTING FAWP MATRICES

We now propose different methods to compute pre-FAWP and post-FAWP matrices defined in (10) and (13), respectively. We also discuss means to estimate the precoding factor β.

A. FAWP by Quantizing the WF-Precoding Matrix

For pre-FAWP and post-FAWP matrices, the scaling factors are computed by means of (12) and (15), respectively, regardless of how the low-resolution matrix ($\mathbf{A}$ for pre-FAWP and $\mathbf{Z}$ for post-FAWP) is computed. Instead of solving the problems in (11) or (14), a simple approach is to directly quantize the infinite-precision matrix $\mathbf{Q}^{\mathrm{WF}}$. We call this approach *FAWP-WF*; more specifically, pre-FAWP-WF and post-FAWP-WF when applied to pre-FAWP and post-FAWP matrices, respectively.

We quantize $\mathbf{Q}^{\mathrm{WF}}$ following the method put forward in [11]. For pre-FAWP-WF, we first find the maximum value $w_{\max}$ of $[|\Re\{\mathbf{q}_u^{\mathrm{WF}}\}|; |\Im\{\mathbf{q}_u^{\mathrm{WF}}\}|]$ for each column $\mathbf{q}_u^{\mathrm{WF}}$ of $\mathbf{Q}^{\mathrm{WF}}$. We then divide the range $[-w_{\max}, w_{\max}]$ into uniform-width bins, where each bin is represented by its centroid value. The centroid values are scaled by the same factor so that they are integer numbers, which preserves the objective value in (11) and results in the low-resolution entries of the column $\mathbf{a}_u$. For post-FAWP-WF, we apply the same procedure on a per-row basis: Each quantized row of $\mathbf{Q}^{\mathrm{WF}}$ corresponds to one row of $\mathbf{Z}^H$.

Since the problems in (11) and (14) are NP-hard, FAWP-WF significantly reduces complexity. Concretely, FAWP-WF requires the same complexity of $\mathcal{O}(BU^2)$ as computing the infinite-precision $\mathbf{Q}^{\mathrm{WF}}$ in (7). As a result, we will use FAWP-WF as a baseline to evaluate the performance of the algorithm proposed next, which tackles the problems in (11) and (14).

B. FAWP via Forward-Backward Splitting (FBS)

Similar to finite-alphabet equalization matrices in [11], we can also approximately solve the FAWP problems in (11) and (14) using forward-backward splitting (FBS), an approach dubbed

---

[1]In contrast, for the uplink considered in [11], post-equalization scaling requires fewer scaling factors *and* does not increase the resolution of the received vector.

FAWP-FBS. In what follows, we will present pre-FAWP-FBS, an algorithm for computing the low-resolution part of a pre-FAWP matrix starting from the problem in (11). The algorithm for post-FAWP matrices, dubbed post-FAWP-FBS, can be derived in a similar way starting from (14).

As in [11], we assume that the optimal objective value γ_u of (11), $u = 1, \ldots, U$, is known. Then, solving the problem in (11) is equivalent to solving the following problem:

$$\mathbf{a}_u = \arg\min_{\tilde{\mathbf{a}} \in \mathcal{X}^B} \frac{1}{2}\|\mathbf{H}\tilde{\mathbf{a}}\|_2^2 + \frac{\kappa^{\mathrm{WF}}}{2}\|\tilde{\mathbf{a}}\|_2^2 - \frac{\gamma_u}{2}|\mathbf{h}_u^{\mathrm{r}}\tilde{\mathbf{a}}|^2. \quad (16)$$

As γ_u is unknown, we will use it as a parameter that can be tuned to empirically improve the performance of our algorithm.

We next relax the finite-alphabet constraint $\tilde{\mathbf{a}} \in \mathcal{X}^B$ in (16) to $\tilde{\mathbf{a}} \in \mathcal{B}^B$, where $\mathcal{B}$ represents the convex hull of $\mathcal{X}$. By doing so, the all-zeros vector $\mathbf{0}_{B \times 1}$ becomes a trivial solution. To avoid this solution, we follow the approach in [17] and include in (16) the term $-\frac{\delta}{2}\|\tilde{\mathbf{a}}\|_2^2$, with $\delta > 0$, to encourage large entries in the vector $\tilde{\mathbf{a}}$. The resulting optimization problem is

$$\mathbf{a}_u = \arg\min_{\tilde{\mathbf{a}} \in \mathcal{B}^B} \frac{1}{2}\|\mathbf{H}\tilde{\mathbf{a}}\|_2^2 - \frac{\gamma_u}{2}|\mathbf{h}_u^{\mathrm{r}}\tilde{\mathbf{a}}|^2 + \frac{\kappa^{\mathrm{WF}} - \delta}{2}\|\tilde{\mathbf{a}}\|_2^2. \quad (17)$$

We are now ready to apply FBS [18], [19]. FBS is an efficient procedure for solving convex problems of the form $\hat{\mathbf{a}} = \arg\min_{\tilde{\mathbf{a}}} f(\tilde{\mathbf{a}}) + g(\tilde{\mathbf{a}})$, where both functions f and g are convex, but f is smooth and g is not necessarily smooth or bounded. FBS is an iterative method that runs for $t_{\max}$ iterations or until convergence [19]. In each iteration t, FBS computes

$$\tilde{\mathbf{v}}^{(t+1)} = \tilde{\mathbf{a}}^{(t)} - \tau^{(t)}\nabla f(\tilde{\mathbf{a}}^{(t)}), \quad (18)$$

$$\tilde{\mathbf{a}}^{(t+1)} = \mathrm{prox}_g\left(\tilde{\mathbf{v}}^{(t+1)}; \tau^{(t)}\right), \quad (19)$$

where $\nabla f(\tilde{\mathbf{a}}^{(t)})$ is the gradient of the function f and $\{\tau^{(t)} > 0\}$ is a sequence of step sizes. The proximal operator of the function g is defined as $\mathrm{prox}_g(\tilde{\mathbf{v}}; \tau) = \arg\min_{\tilde{\mathbf{a}}}\left\{\tau g(\tilde{\mathbf{a}}) + \frac{1}{2}\|\tilde{\mathbf{a}} - \tilde{\mathbf{v}}\|_2^2\right\}$ [20].

Since the problem in (17) is non-convex, FBS is not guaranteed to converge to an optimal solution. Nevertheless, we use FBS to approximately solve (17) by setting

$$f(\tilde{\mathbf{a}}) = \frac{1}{2}\|\mathbf{H}\tilde{\mathbf{a}}\|_2^2 - \frac{\gamma_u}{2}|\mathbf{h}_u^{\mathrm{r}}\tilde{\mathbf{a}}|^2, \quad (20)$$

$$g(\tilde{\mathbf{a}}) = \mathbb{I}_{\mathcal{B}^B}(\tilde{\mathbf{a}}) + \frac{\kappa^{\mathrm{WF}} - \delta}{2}\|\tilde{\mathbf{a}}\|_2^2, \quad (21)$$

where $\mathbb{I}_{\mathcal{B}^B}(\tilde{\mathbf{a}})$ is the indicator function, which is zero if $\tilde{\mathbf{a}} \in \mathcal{B}^B$ and infinity otherwise. We use the indicator function to incorporate the convex constraint $\tilde{\mathbf{a}} \in \mathcal{B}^B$ in (17) into the function $g(\tilde{\mathbf{a}})$. These choices for $f(\tilde{\mathbf{a}})$ and $g(\tilde{\mathbf{a}})$ result in:

$$\nabla f(\tilde{\mathbf{a}}) = \mathbf{H}^H\mathbf{H}\tilde{\mathbf{a}} - \gamma_u(\mathbf{h}_u^{\mathrm{r}})^H\mathbf{h}_u^{\mathrm{r}}\tilde{\mathbf{a}} \quad (22)$$

$$\mathrm{prox}_g(\tilde{v}) = \mathrm{sgn}(\Re\{\tilde{v}\})\min\left\{\nu^{(t)}|\Re\{\tilde{v}\}|, 1\right\} + j\,\mathrm{sgn}(\Im\{\tilde{v}\})\min\left\{\nu^{(t)}|\Im\{\tilde{v}\}|, 1\right\}, \quad (23)$$

where $\nu^{(t)} = (1 + \tau^{(t)}(\kappa^{\mathrm{WF}} - \delta))^{-1}$ and (23) is applied element-wise to $\tilde{\mathbf{v}}$. Pre-FAWP-FBS can be summarized as follows:

Algorithm 1 (Pre-FAWP-FBS). *Initialize $\tilde{\mathbf{a}}^{(1)}$ with either the maximum-ratio transmission (MRT) solution $(\mathbf{h}_u^{\mathrm{r}})^H$ or the pre-FAWP-WF solution $\mathbf{a}_u^{\mathrm{WF}}$, and fix the sets of parameters $\{\tau^{(t)}\}$, $\{\nu^{(t)}\}$, and $\{\gamma^{(t)}\}$. Then, for each iteration $t = 1, 2, \ldots, t_{\max}$, compute*

$$\tilde{\mathbf{v}}^{(t+1)} = \left(\mathbf{I}_B - \tau^{(t)}\mathbf{H}^H(\mathbf{I}_U - \gamma^{(t)}\mathbf{e}_u\mathbf{e}_u^H)\mathbf{H}\right)\tilde{\mathbf{a}}^{(t)} \quad (24)$$

$$\tilde{\mathbf{a}}^{(t+1)} = \mathrm{prox}_g(\tilde{\mathbf{v}}^{(t+1)}). \quad (25)$$

The result $\tilde{\mathbf{a}}^{(t_{max}+1)}$ is projected onto the finite alphabet $\mathcal{X}$ to obtain $\mathbf{a}_u$. The optimal scalar α_u is computed using (12). This procedure is repeated for each UE $u = 1, \ldots, U$.

To tune the algorithm parameters $\{\tau^{(t)}\}$, $\{\nu^{(t)}\}$, and $\{\gamma^{(t)}\}$, we use a neural-network-based approach as put forward in [21]. Note that we have replaced γ_u with $\gamma^{(t)}$ in Algorithm 1 in order to (i) keep the algorithm general for different user locations and (ii) to increase flexibility during optimization.

We now summarize post-FAWP-FBS, which can be derived following similar steps as for the derivation of pre-FAWP-FBS.

Algorithm 2 (Post-FAWP-FBS). *Initialize $\tilde{\mathbf{z}}^{(1)}$ with either the MRT solution $\mathbf{h}_b$ or the post-FAWP-WF solution $\mathbf{z}_b^{WF}$, and fix the sets of parameters $\{\tau^{(t)}\}$, $\{\nu^{(t)}\}$, and $\{\gamma^{(t)}\}$. Then, for each iteration $t = 1, 2, \ldots, t_{max}$, compute*

$$\tilde{\mathbf{v}}^{(t+1)} = \left(\mathbf{I}_U - \tau^{(t)}\mathbf{H}(\mathbf{I}_B - \gamma^{(t)}\mathbf{e}_b\mathbf{e}_b^H)\mathbf{H}^H\right)\tilde{\mathbf{z}}^{(t)} \quad (26)$$

$$\tilde{\mathbf{z}}^{(t+1)} = \mathrm{prox}_g(\tilde{\mathbf{v}}^{(t+1)}). \quad (27)$$

The result $\tilde{\mathbf{z}}^{(t_{max}+1)}$ is projected onto the finite alphabet $\mathcal{X}$ to obtain $\mathbf{z}_b$. The optimal scale ζ_u is computed with (15). This procedure is done for each BS antenna $b = 1, \ldots, B$.

We note that both FAWP-FBS algorithms have the same complexity order of $\mathcal{O}(BU^2)$ as WF and FAWP-WF.

C. Estimating the Precoding Factor β

While the BS is able to compute the precoding factor β via (4) with a FAWP matrix $\mathbf{Q}$ instead of $\mathbf{Q}^{\mathrm{WF}}$, the UEs need to estimate such precoding factor in order to correctly estimate the transmitted symbols in $\mathbf{s}$. As shown in [13], estimation can be achieved in a block-fading scenario by transmitting a pilot symbol that is known at the UE side. Specifically, the BS will transmit the pilot $s_u = \sqrt{E_s}$, $u = 1, \ldots, U$. Then, the uth UE will receive $y_u = \beta^{-1}\mathbf{h}_u^{\mathrm{r}}\mathbf{q}_u s_u + \check{e}_u + n_u$, where $\check{e}_u$ represents residual interference from the other UEs. The objective now is for the UE to find a $\hat{\beta}_u \in \mathbb{R}_+$ such that it generates an unbiased estimate $\hat{s}_u$ of s_u, i.e., $\hat{s}_u = \hat{\beta}_u y_u \approx s_u$. By taking into account that the transmitted pilot symbol s_u is known to be $\sqrt{E_s}$ and by assuming that $\check{e}_u + n_u$ is zero-mean Gaussian distributed and independent of s_u, the UE can compute a maximum likelihood estimate (MLE) of $\hat{\beta}_u$ as [13]:

$$\hat{\beta}_u^{\mathrm{MLE}} = \mathbb{R}\{\sqrt{E_s}/y_u\}. \quad (28)$$
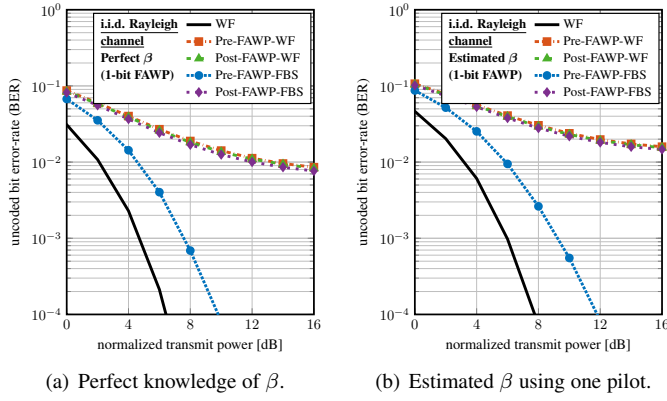
181

Fig. 1. Uncoded bit-error rate (BER) for a $B = 256$ BS-antenna, $U = 16$ UE, 16-QAM system operating over an i.i.d. Rayleigh fading channel. All the FAWP-based approaches use 1-bit FAWP matrices. Pre- and post-FAWP-FBS run for $t_{\max} = 10$ iterations starting from the MRT solution $\mathbf{H}^H$.

While more pilots could be transmitted to form a better estimate $\hat{\beta}_u^{\mathrm{MLE}}$, our results in Section V show that one pilot is sufficient to achieve reliable downlink communication.

V. NUMERICAL RESULTS

We now present simulation results for both pre-FAWP and post-FAWP matrices generated by either FAWP-WF or FAWP-FBS. We perform a comparison in terms of BER and EVM versus normalized transmit power, which we define as P/N_0. For simplicity, we restrict our evaluation on a mmWave system with $B = 256$ BS antennas serving $U = 16$ UEs.

A. 1-bit FAWP BER Performance and β-Estimation

Fig. 1 shows the uncoded BER for the considered system when using 16-QAM in an i.i.d. Rayleigh fading channel. For the FAWP-based approaches, we use a 1-bit alphabet. In Fig. 1(a), we consider the case where the UEs have genie-aided access to the exact $\hat{\beta}_u$ precoding scaling factor. Here, we can see that both FAWP-WF approaches result in an error floor. In fact, the FAWP-WF precoders are significantly outperformed by pre-FAWP-FBS, which computes WF-optimal pre-FAWP matrices. However, post-FAWP-FBS is unable to outperform post-FAWP-WF, a surprising behavior that we observe consistently across all our experiments—a detailed investigation of this behavior is left for future work. Hence, we exclude post-FAWP-FBS in the ensuing discussion.

In Fig. 1(b), we consider the same scenario as before, but this time $\hat{\beta}_u$ is estimated from a single pilot transmission as described in Section IV-C. We can see that all precoders (including the infinite-precision WF) suffer from roughly a 2 dB loss. In what follows, we assume that $\hat{\beta}_u$ is estimated using a single pilot.

B. Multi-Bit FAWP EVM Performance

Fig. 2 shows the EVM performance for the different FAWP precoders and $\{1, 2, 3\}$-bit alphabets. The red dashed lines represent the per-modulation EVM requirements as specified by the 3GPP 5G NR standard [22]. Fig. 2(a) confirms what we previously observed in Fig. 1 for the 1-bit alphabet: While FAWP-WF suffers a high error-floor that prevents such approach from reaching the EVM requirement even for QPSK, pre-FAWP-FBS almost meets

the EVM requirement for 64-QAM. By increasing the number of bits used for the finite alphabet, the gap between the FAWP approaches and the infinite-precision WF decreases—to the point shown in Fig. 2(c) where all FAWP approaches meet the 64-QAM EVM requirement when using a 3-bit alphabet. It is interesting to observe that post-FAWP-WF outperforms pre-FAWP-WF when using finite alphabets with more than 1 bit. Nonetheless, post-FAWP-WF is unable to outperform pre-FAWP-FBS.

### C. Performance Under Realistic Propagation Conditions

We now evaluate FAWP under more realistic mmWave propagation conditions. We use the QuaDRiGa channel model [23] to simulate communication in the "mmMAGIC_UMi" scenario when using a 60 GHz carrier frequency for both non-LoS and LoS propagation conditions. We randomly place the UEs 10 m to 110 m away from the BS in a $120°$ circular sector, with a minimum angular separation of $4°$. Furthermore, we assume perfect power control, i.e., all the users receive the same signal power. Our simulation results are shown in Fig. 3. The trends we observed in the i.i.d. Rayleigh fading scenario are confirmed: Pre-FAWP-FBS is able to outperform both FAWP-WF precoders, although the gains of the former (as well as the gap to the WF precoder) reduce when using more bits for the finite alphabet. An interesting observation is that, for the LoS scenario illustrated in Fig. 3(b), the performance of pre-FAWP-WF is on par with that of post-FAWP-WF, which was not the case for the non-LoS and i.i.d. Rayleigh fading scenarios. Regardless, the results in Fig. 3 demonstrate that FAWP remains to perform well with realistic mmWave channels, which holds the promise of FAWP enabling low-power and area-efficient precoding circuitry.

VI. CONCLUSIONS

To enable energy- and area-efficient circuitry, we have proposed FAWP, an approach that replaces part of the linear WF precoder with a low-resolution matrix. We have developed two structures for FAWP matrices, pre-FAWP and post-FAWP, as well as two methods to craft such matrices. Our simulation results have shown that the sophisticated pre-FAWP-FBS algorithm is able to significantly outperform a simple quantization of the WF-precoding matrix, especially when using extremely low-resolution alphabets, and that it approaches the performance of the infinite-precision WF precoder with as few as 3 bits of resolution. Pre-FAWP-FBS accomplishes such feats while exhibiting the same asymptotical complexity as the WF precoder. As for post-FAWP matrices, our simulation results have shown that post-FAWP-FBS does not outperform the simple quantization of the WF precoder. We have verified these results under realistic conditions, such as LoS and non-LoS mmWave channels, as well as with estimation of the precoding factor β. Thus, FAWP matrices are a promising approach to reduce hardware complexity and power consumption of precoding in mmWave MU-MIMO systems. However, in order to quantify the real-world benefits of FAWP, a hardware-level evaluation is necessary—such an evaluation is part of ongoing work. Since our FAWP approach performs matrix-vector products with coarsely quantized numbers, corresponding hardware implementations
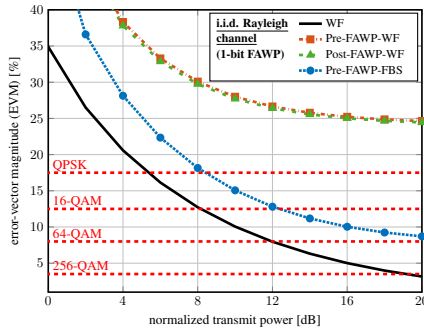
(a) 1-bit FAWP.
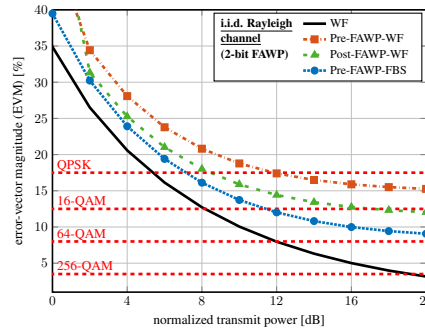
(b) 2-bit FAWP.
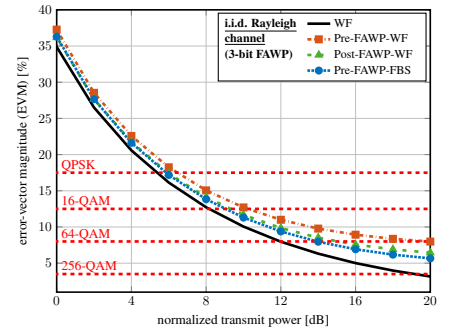
(c) 3-bit FAWP.

Fig. 2. Error-vector magnitude (EVM) for a $B = 256$ BS-antenna, $U = 16$ UE system operating in an i.i.d. Rayleigh fading channel. The red dashed lines represent the EVM requirements established by the 3GPP 5G NR technical specification [22]. We consider FAWP using $\{1, 2, 3\}$-bit alphabets. For $\{2, 3\}$-bit FAWP, pre-FAWP-FBS is initialized with the MRT solution $\mathbf{H}^H$ and runs for $t_{\max} = 5$ iterations. The details for 1-bit pre-FAWP-FBS are given in Fig. 1.
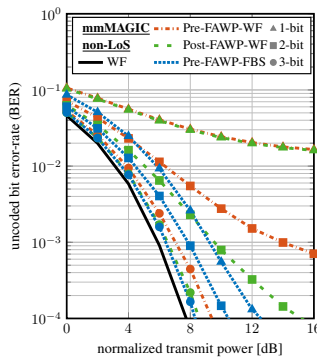
(a) QuaDRiGa non-LoS.
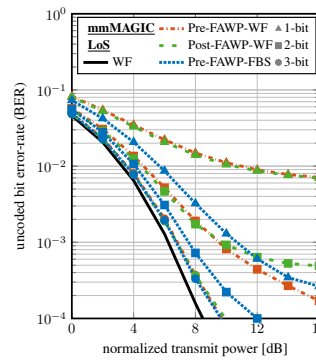
(b) QuaDRiGa LoS.

Fig. 3. Uncoded bit-error rate (BER) for a $B = 256$ BS-antenna, $U = 16$ UE, 16-QAM system operating in realistic mmWave channel models. 1-bit pre-FAWP-FBS runs $t_{\max} = 10$ iterations; $\{2, 3\}$-bit pre-FAWP-FBS run no more than $t_{\max} = 5$ iterations. Pre-FAWP-FBS is initialized with $\mathbf{H}^H$ for all cases but the $\{2, 3\}$-bit LoS ones, which use $\mathbf{A}^{WF}$ from pre-FAWP-WF.

could benefit from emerging processing-in-memory architectures, such as the one proposed in [24].

REFERENCES

[1] A. L. Swindlehurst, E. Ayanoglu, P. Heydari, and F. Capolino, "Millimeter-wave massive MIMO: The next wireless revolution?" *IEEE Commun. Mag.*, vol. 52, no. 9, pp. 56–62, Sep. 2014.

[2] E. G. Larsson, F. Tufvesson, O. Edfors, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 186–195, Feb. 2014.

[3] W. Roh, J.-Y. Seol, J. Park, B. Lee, J. Lee, Y. Kim, J. Cho, K. Cheun, and F. Aryanfar, "Millimeter-wave beamforming as an enabling technology for 5G cellular communications: Theoretical feasibility and prototype results," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 106–113, Feb. 2014.

[4] B. Sadhu, Y. Tousi, J. Hallin, S. Sahl, S. Reynolds, Ö. Renström, K. Sjögren, O. Haapalahti, N. Mazor, B. Bokinge, G. Weibull, H. Bengtsson, A. Carlinger, E. Westesson, J. Thillberg, L. Rexberg, M. Yeck, X. Gu, D. Friedman, and A. Valdes-Garcia, "A 28GHz 32-element phased-array transceiver IC with concurrent dual polarized beams and 1.4 degree beam-steering resolution for 5G communication," in *IEEE Int. Solid-State Circuits Conf. (ISSCC)*, Feb. 2017, pp. 128–129.

[5] A. Alkhateeb, J. Mo, N. González-Prelcic, and R. W. Heath Jr., "MIMO precoding and combining solutions for millimeter-wave systems," *IEEE Commun. Mag.*, vol. 52, no. 12, pp. 122–131, Dec. 2014.

[6] E. Björnson, L. Van der Perre, S. Buzzi, and E. G. Larsson, "Massive MIMO in sub-6 GHz and mmWave: Physical, practical, and use-case differences," *IEEE Wireless Commun.*, vol. 26, no. 2, pp. 100–108, Apr. 2019.

[7] S. Dutta, C. N. Barati, A. Dhananjay, D. A. Ramirez, J. F. Buckwalter, and S. Rangan, "A case for digital beamforming at mmWave," *arXiv preprint arXiv:1901.08693*, Jan. 2019.

[8] J. Mo and R. W. Heath Jr., "Capacity analysis of one-bit quantized MIMO systems with transmitter channel state information," *IEEE Trans. Signal Process.*, vol. 63, no. 20, pp. 5498–5512, Oct. 2015.

[9] K. Roth and J. A. Nossek, "Achievable rate and energy efficiency of hybrid and digital beamforming receivers with low resolution ADC," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 9, pp. 2056–2068, Sep. 2017.

[10] S. Jacobsson, G. Durisi, M. Coldrey, U. Gustavsson, and C. Studer, "Throughput analysis of massive MIMO uplink with low-resolution ADCs," *IEEE Trans. Wireless Commun.*, vol. 16, no. 6, pp. 4038–4051, Jun. 2017.

[11] O. Castañeda, S. Jacobsson, G. Durisi, T. Goldstein, and C. Studer, "Finite-alphabet MMSE equalization for all-digital massive MU-MIMO mmWave communication," submitted to a journal.

[12] 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.

[13] ——, "Nonlinear 1-bit precoding for massive MU-MIMO with higher-order modulation," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, Pacific Grove, CA, USA, Nov. 2016, pp. 763–767.

[14] N. Fatema, G. Hua, Y. Xiang, D. Peng, and I. Natgunanathan, "Massive MIMO linear precoding: A survey," *IEEE Syst. J.*, vol. 12, no. 4, pp. 3920–3931, Dec. 2018.

[15] M. Joham, W. Utschick, and J. A. Nossek, "Linear transmit processing in MIMO communications systems," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 2700–2712, Aug. 2005.

[16] K. B. Petersen and M. S. Pedersen, "The matrix cookbook," Nov. 2012.

[17] S. Shah, A. K. Yadav, C. D. Castillo, D. W. Jacobs, C. Studer, and T. Goldstein, "Biconvex relaxation for semidefinite programming in computer vision," in *Eur. Conf. Comput. Vision*, Sep. 2016, pp. 717–735.

[18] T. Goldstein and S. Setzer, "High-order methods for basis pursuit," *UCLA CAM Report*, pp. 10–41, 2010.

[19] T. Goldstein, C. Studer, and R. G. Baraniuk, "A field guide to forward-backward splitting with a FASTA implementation," Nov. 2014. [Online]. Available: https://arxiv.org/abs/1411.3406

[20] N. Parikh and S. Boyd, "Proximal algorithms," *Found. Trends Optim.*, vol. 1, no. 3, pp. 127–239, Jan. 2014.

[21] A. Balatsoukas-Stimming, O. Castañeda, S. Jacobsson, G. Durisi, and C. Studer, "Neural-network optimized 1-bit precoding for massive MU-MIMO," in *IEEE Int. Workshop Signal Process. Advances Wireless Commun. (SPAWC)*, Cannes, France, Jul. 2019.

[22] 3GPP, "5G; NR; base station (BS) radio transmission and reception," May 2019, TS 38.104 version 15.5.0 Rel. 15.

[23] S. Jaeckel, L. Raschkowski, K. Börner, and L. Thiele, "QuaDRiGa: A 3-D multi-cell channel model with time evolution for enabling virtual field trials," *IEEE Trans. Antennas Propag.*, vol. 62, no. 6, pp. 3242–3256, Jun. 2014.

[24] O. Castañeda, M. Bobbett, A. Gallyas-Sanhueza, and C. Studer, "PPAC: A versatile in-memory accelerator for matrix-vector-product-like operations," in *IEEE Int. Conf. Appl.-specific Syst., Archit., Processors (ASAP)*, Jul. 2019, pp. 149–156.