

Low-Latency Lattice-Reduction-Aided One-Bit Precoding Processor for 64-QAM 4×64 MU-MIMO Systems

PAO-PAO HO^{1,2}, CHIAO-EN CHEN³ (Senior Member, IEEE), AND YUAN-HAO HUANG^{1,4} (Member, IEEE)

¹Institute of Communications Engineering, National Tsing Hua University, Hsinchu 300, Taiwan

²SoC Communication Network Business Group II, RealTek Corporation, Hsinchu 300, Taiwan

³Department of Electrical Engineering, National Chung Hsing University, Taichung 402, Taiwan

⁴Department of Electrical Engineering, National Tsing Hua University, Hsinchu 300, Taiwan

This article was recommended by Guest Editor G. Manganaro.

CORRESPONDING AUTHOR: Y.-H. HUANG (e-mail: yhuang@ee.nthu.edu.tw)

This work was supported in part by the Ministry of Science and Technology, Taiwan, under Grant MOST 108-2221-E-194-006, and in part by the technically supported by Taiwan Semiconductor Research Institute (TSRI).

ABSTRACT Massive multi-user multiple-input multiple-output (MU-MIMO) communication is a crucial technique for next-generation wireless systems because it enables high-throughput and ultra-reliable data transmission. However, high power consumption is a challenging problem in conventional transceiver designs, where each RF chain at the transmitter requires a pair of high-resolution digital-to-analog converters (DACs), which are the primary power sources in front-end circuits. Therefore, quantized precoding algorithms were recently proposed to address this issue. They used low-resolution DACs at the transmitter and compensated for the distortion caused by the quantization effect at the baseband. This paper proposes a constellation-range gain-controlled lattice-reduction-aided (CR-GCLRA) one-bit precoding algorithm for massive MU-MIMO systems with high-order quadrature amplitude modulation (QAM) signaling. Lattice reduction (LR) preprocessing is adopted for performance enhancement. A novel gain-control mechanism is proposed to address the constellation range expansion problem in conventional LR preprocessing. We designed and implemented a CR-GCLRA one-bit precoding processor by using TSMC 40-nm CMOS technology to accelerate the one-bit precoding processing for 64-QAM 4×64 MU-MIMO system. The proposed one-bit precoding processor can achieve a throughput of 43.92 Mbits per second at a clock speed of 269 MHz with 271 mW power consumption and $0.51 \mu\text{s}$ latency.

INDEX TERMS Lattice reduction, multi-user MIMO, one-bit precoding, VLSI.

I. INTRODUCTION

MASSIVE multi-user multiple-input multiple-output (MU-MIMO) system enables the use of hundreds of antennas at a base station to support a large number of user equipment (UE) devices with excellent spectral efficiency and reliability [1]–[3]. One of the main challenges of massive MU-MIMO systems is the requirement of a large number of power-hungry devices, such as digital-to-analog converters (DAC) and radio frequency (RF) chain circuits, at the transceiver front-end. As DAC power dissipation grows proportionally with sampling frequency and exponentially with sampling resolution [4], novel transmission schemes that enable the use of low-resolution DACs

in new transceiver designs are urgently required. Quantized precoding (QP) can be categorized into linear quantized precoding (LQP) and nonlinear quantized precoding (NLQP). The LQP schemes can achieve acceptable performance in low-order modulation systems [5]–[7]. The NLQP algorithms such as semidefinite relaxation (SDR), squared-infinity norm Douglas-Rachford splitting (SQUID), and sphere precoding (SP) algorithms, all of which were proposed in [7], generally yield superior bit error rate (BER) performance at the expense of higher complexity relative to LQP. Several optimized quantized precoding schemes [8]–[10] had the aim of achieving optimal performance or reducing computational complexity, but are currently restricted

to systems employing binary-phase-shift-keying (BPSK) or quadrature-phase-shift-keying (QPSK) modulations. Some studies [11]–[14] have developed efficient NLQP algorithms with balanced performance and complexity tradeoff for implementation. The projected downlink beamforming (Pokemon) precoder [11] formulated a biconvex relaxation (BCR) framework and enabled highly efficient implementation without sacrificing error rate performance relative to the SQUID algorithm [7]. The biconvex 1-bit precoding (C2PO) algorithm was later proposed in [12]; C2PO can be viewed as the modification of the Pokemon algorithm aimed at facilitating hardware-friendly very large-scale integration (VLSI) implementation. In the context of lattice reduction [15]–[18], a lattice-reduction-aided successive interference cancellation (LRSIC) one-bit precoding algorithm [19] was proposed to facilitate parallel processing for high-throughput implementation. One-bit precoding algorithms supporting UE devices with multiple receive antennas were proposed in [20]. Quantized precoding that supports high-order quadrature-amplitude modulation (QAM) is another essential design issue. In [21], the constellation range (CR) was carefully designed when developing QP algorithms employing multilevel signaling. Through the exploitation of the channel hardening phenomenon, the CR was derived as a fixed constant in closed form without instantaneous gain adaptation as required in traditional QP algorithms [7]. Another QP algorithm based on a modified alternating direction method of multipliers (ADMM) was proposed in [13]. It exhibited outstanding error rate performance for high-order QAM signaling [13], but its iterative ADMM steps can result in long processing latency.

In this article, we propose a CR gain-controlled lattice-reduction-aided (CR-GCLRA) one-bit precoding algorithm, which can be viewed as the research effort in extending our LRSIC [19] precoder from PSK to high-order QAM scenarios. We propose to combine the CR method [21] with a noniterative SIC precoder to achieve both low hardware complexity and low precoding latency. We propose a new gain-control mechanism to resolve the issue of infinite constellation range in the lattice-reduced domain, to facilitate high-throughput VLSI design, and to maintain superior performance for high-order QAM signaling in fixed-point hardware implementation, thereby saving a considerable amount of power consumption with one-bit DACs in the massive MIMO transmitter.

The remainder of this paper is organized as follows. In Section II, we introduce the quantized precoding technique for massive MU-MIMO systems. In Section III, we propose the CR-GCLRA one-bit precoding algorithm; the performance and complexity of the proposed algorithm are analyzed and discussed. In Section IV, we present our proposed one-bit precoding architecture based on the CR-GCLRA algorithm and then demonstrate the VLSI design and implementation results. Finally, we conclude this article in Section V.

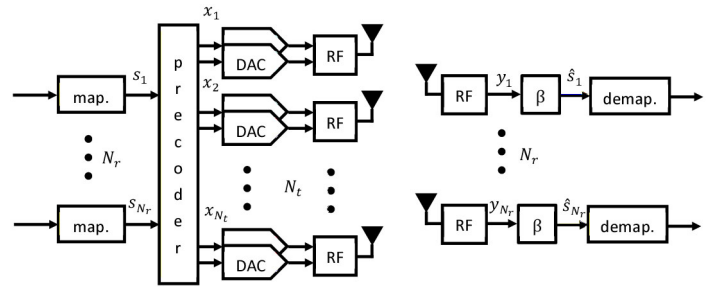


FIGURE 1. Quantized precoding for massive MU-MIMO systems.

II. QUANTIZED PRECODING FOR MASSIVE MU-MIMO

The quantized precoding system model in [7] was adopted in this study. Fig. 1 presents an $N_r \times N_r$ massive MU-MIMO system with N_r single-antenna UE devices. At the base station (BS), the symbol vector $\mathbf{s} = [s_1, s_2, \dots, s_{N_r}]^T \in \mathbb{O}^{N_r \times 1}$ is precoded to quantized $\mathbf{x} = [x_1, x_2, \dots, x_{N_r}]^T \in \mathbb{X}^{N_r \times 1}$ before transmission. Here, \mathbb{O} denotes the constellation set. $\mathbb{X}^{N_r \times 1}$ represents the set of alphabets and can be further expressed as $\mathbb{X} = \mathcal{L} + j\mathcal{L}$, where $\mathcal{L} = \{l_0, \dots, l_{L-1}\}$ denotes the set of quantized values of each DAC output, and $L = |\mathcal{L}|$. The received signal vector $\mathbf{y} = [y_1, y_2, \dots, y_{N_r}]^T \in \mathbb{C}^{N_r \times 1}$ can thus be modeled by

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

where $\mathbf{H} \in \mathbb{C}^{N_r \times N_r}$ is the Rayleigh fading channel matrix, and $\mathbf{n} \in \mathbb{C}^{N_r \times 1}$ is the noise vector generated by circularly symmetric complex Gaussian i.i.d. random variables with zero-mean and variance σ^2 . The transmitted signal \mathbf{x} follows the following the symbol-level power constraint

$$\|\mathbf{x}\|_2^2 = P, \quad (2)$$

and we define the signal-to-noise ratio (SNR) as P/σ^2 . The received signal at the u th UE device is multiplied by a real-valued rescaling factor β_u and the estimated signal vector $\hat{\mathbf{s}} = [\hat{s}_1, \hat{s}_2, \dots, \hat{s}_{N_r}]^T \in \mathbb{C}^{N_r \times 1}$ is given by

$$\hat{s}_u = \beta_u y_u, \quad \text{for } u = 1, \dots, N_r. \quad (3)$$

A. REVIEW OF THE QUANTIZED SPHERE PRECODING ALGORITHM

In the quantized sphere precoding [7], a scenario is considered with the use of one-bit DACs ($L = 2$), and thus $\mathbb{X} = \sqrt{\frac{P}{2N_r}}(\pm 1 \pm j)$. With some mathematical derivations under the assumption $\beta_u = \beta > 0$ for all $u = 1, \dots, N_r$, the optimal sum-MSE symbol-level one-bit precoder design problem can be formulated as [7]:

$$\begin{cases} \arg \min_{\mathbf{x} \in \mathbb{X}^{N_r}, \beta \in \mathbb{R}} \|\mathbf{s} - \beta \mathbf{H}\mathbf{x}\|_2^2 + \beta^2 N_r \sigma^2, \\ \text{subject to } \beta > 0. \end{cases} \quad (4)$$

In [7], the sphere precoding algorithm (Algorithm 1) was proposed to solve the quantized precoding problem. From (2), the objective function can be expressed as

$$\|\mathbf{s} - \beta \mathbf{H}\mathbf{x}\|_2^2 + \beta^2 N_r \sigma^2 = \|\mathbf{s} - \beta \mathbf{H}\mathbf{x}\|_2^2 + \beta^2 \frac{N_r \sigma^2}{P} \|\mathbf{x}\|_2^2$$

Algorithm 1 Quantized Sphere Precoding Algorithm [7]

 Input: $\mathbf{s}, \mathbf{H}, N_t, N_r, P, \sigma^2$

- 1: $\bar{\mathbf{s}} = \begin{bmatrix} \mathbf{s} \\ \mathbf{0}_{N_t} \end{bmatrix}, \bar{\mathbf{H}} = \begin{bmatrix} \mathbf{H} \\ \sqrt{\frac{N_r \sigma^2}{P}} \mathbf{I}_{N_t} \end{bmatrix}$
 - 2: $\bar{\mathbf{H}} = \mathbf{Q}\mathbf{R}$ (QR decomposition)
 - 3: $\mathbf{P} = \mathbf{H}^H (\mathbf{H}\mathbf{H}^H + \frac{N_r N_0}{P} \mathbf{I}_{N_r})^{-1}$
 - 4: $\mathbf{x}_0 = \sqrt{\frac{P}{2N_t}} (\text{sgn}(\Re\{\mathbf{P}\mathbf{s}\}) + j \text{sgn}(\Im\{\mathbf{P}\mathbf{s}\}))$
 - 5: $\beta_1(\mathbf{x}_0) = \frac{\Re\{\mathbf{s}^H \mathbf{H}\mathbf{x}_0\}}{\|\mathbf{H}\mathbf{x}_0\|_2^2 + N_r \sigma^2}$
 - 6: **for** $t = 1, \dots, t_{\max}$ **do**
 - 7: $\hat{\mathbf{x}}_t = \arg \min_{\mathbf{x} \in \mathbb{X}^{N_t}} \|\mathbf{Q}^H \bar{\mathbf{s}} - \beta_t \mathbf{R}\mathbf{x}\|_2^2$ (Sphere precoding)
 - 8: $\beta_{t+1}(\mathbf{x}_t) = \frac{\Re\{\mathbf{s}^H \mathbf{H}\mathbf{x}_t\}}{\|\mathbf{H}\mathbf{x}_t\|_2^2 + N_r \sigma^2}$
 - 9: **end for**
 - 10: $\mathbf{x} = \hat{\mathbf{x}}_{t_{\max}}, \beta = \beta_{t+1}$
- Output:
- \mathbf{x}, β

$$= \|\bar{\mathbf{s}} - \beta \bar{\mathbf{H}}\mathbf{x}\|_2^2, \quad (5)$$

where

$$\bar{\mathbf{s}} = \begin{bmatrix} \mathbf{s} \\ \mathbf{0}_{N_t} \end{bmatrix}, \bar{\mathbf{H}} = \begin{bmatrix} \mathbf{H} \\ \sqrt{\frac{N_r \sigma^2}{P}} \mathbf{I}_{N_t} \end{bmatrix}. \quad (6)$$

 For a given β , the problem (4) can thus be formulated as

$$\arg \min_{\mathbf{x} \in \mathbb{X}^{N_t}} \|\bar{\mathbf{s}} - \beta \bar{\mathbf{H}}\mathbf{x}\|_2^2. \quad (7)$$

The sphere precoding (SP) applies QR factorization to the extended channel matrix $\bar{\mathbf{H}}$ to obtain $\bar{\mathbf{H}} = \mathbf{Q}\mathbf{R}$, where $\mathbf{Q} \in \mathbb{C}^{(N_t+N_r) \times (N_t+N_r)}$ is an unitary matrix and $\mathbf{R} \in \mathbb{C}^{(N_t+N_r) \times N_t}$ is an upper-triangular matrix with non-negative diagonal entries. The problem in (7) can then be expressed as

$$\arg \min_{\mathbf{x} \in \mathbb{X}^{N_t}} \|\bar{\mathbf{s}} - \beta \bar{\mathbf{H}}\mathbf{x}\|_2^2 = \arg \min_{\mathbf{x} \in \mathbb{X}^{N_t}} \|\mathbf{Q}^H \bar{\mathbf{s}} - \beta \mathbf{R}\mathbf{x}\|_2^2, \quad (8)$$

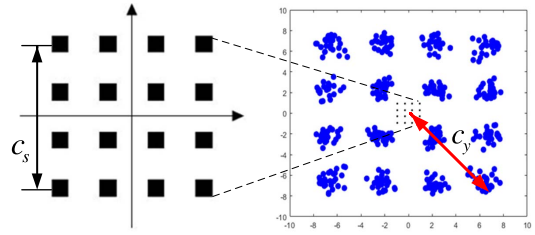
which can be solved efficiently using various preexisting tree-search algorithms, such as those in [22]. Equation (8) is formulated under the assumption of known β . For unknown β , [7] solved it heuristically through alternating minimization approach. For fixed \mathbf{x} , the optimal rescaling factor in (4) can be obtained in closed form as [7]

$$\beta(\mathbf{x}) = \frac{\Re\{\mathbf{s}^H \mathbf{H}\mathbf{x}\}}{\|\mathbf{H}\mathbf{x}\|_2^2 + N_r \sigma^2} = \frac{\Re\{\mathbf{s}^H \mathbf{H}\mathbf{x}\}}{\mathbf{x}^H (\mathbf{H}^H \mathbf{H} + \frac{N_r \sigma^2}{P} \mathbf{I}_{N_t}) \mathbf{x}}. \quad (9)$$

Then, one can then iteratively solve \mathbf{x} for given β until convergence. When β is negative, both signs of \mathbf{x} and β can be inverted for the practical implementation. The minimum mean square error (MMSE) solution is initialized for the iteration and one to three iterations were typically required for convergence in moderate-scale MIMO systems.

III. PROPOSED ONE-BIT PRECODING ALGORITHM

This section presents our one-bit precoding algorithm with high-order QAM signaling. First, we adopted the CR design


FIGURE 2. Constellation ranges of the original QAM symbols and received symbols.

method [21] to eliminate the necessity of adapting the rescaling factor β as the QAM symbol varies. Second, we proposed a gain-controlled lattice reduction mechanism with a simple successive interference cancellation (SIC) procedure to improve performance and facilitate efficient hardware implementation. The primary differences of this work from our previous algorithm [19] are twofold. First, our previous SIC precoding scheme required at least $4N_t$ candidate lattice points for SIC precoding because of the constellation range expansion problem caused by LR. Thus, the computational complexity is still very high. This work further proposes a gain-control mechanism to confine the constellation range and, therefore, only one SIC precoder is required. Second, our previous algorithm supported only QPSK modulation and the rescaling factor β must be adapted for each \mathbf{x} whereas this work supports high-order QAM modulation by adopting a modified CR method. Thus, only a fixed gain factor is used for the transceiver.

A. CR DESIGN

The CR design method [21] analyzed optimal CR for a number of QAM settings. Fig. 2 illustrates the CR design for 16-QAM; c_s and c_y represent the CRs of the original QAM symbols and received symbols, respectively. The design goal is to derive some scaling of c_s to c_y that can achieve either minimal mean square error or optimal bit-error-rate performance. First, for a targeted QAM vector \mathbf{s} , the transmitted one-bit signal \mathbf{x} is designed under a perfectly-known channel matrix \mathbf{H} by minimizing the constraints as follows:

$$\arg \min_{\mathbf{x} \in \mathbb{X}^{N_t}} \|\mathbf{H}\mathbf{x} - \mathbf{s}\|_2^2. \quad (10)$$

Then, the CR method [21] produced an infinite-resolution zero-forcing (ZF) solution $\mathbf{x}^* = \mathbf{H}^H (\mathbf{H}\mathbf{H}^H)^{-1} \mathbf{s}$ and utilized the channel hardening property of massive MIMO for evaluating the received signal gain. The transmitted signal \mathbf{x}^* must satisfy the average power constraint $\|\mathbf{x}^*\|_2^2 \leq P$. Assumed that the symbol s_k of the k -th user is an i.i.d. variable of N^2 -QAM with a constellation range of c . Then, the power mean and variance of the original symbol are given by

$$\mu_s = \mathbb{E}\{|s_k|^2\} = \frac{N+1}{6(N-1)} c^2 \quad (11)$$

and

$$\sigma_s^2 = \mathbb{E} \left\{ \left(|s_k|^2 - \mu_s \right)^2 \right\} = \frac{(N+1)(N^2-4)}{90(N-1)^3} c^4, \quad (12)$$

respectively. Assume that $\sum_{k=1}^{N_r} |s_k|^2$ equals the mean plus two times standard deviation for the power constraint, that is, $N_r \mu_s + 2\sqrt{N_r} \sigma_s = PN_t$. Thus, the CR is given in [21] as

$$c_{ZF} = \sqrt{\frac{2PN_t}{f(N_r, N)}}, \quad (13)$$

where

$$f(N_r, N) = N_r \frac{N+1}{3(N-1)} + 2\sqrt{N_r \frac{(N+1)(N^2-4)}{22.5(N-1)^3}}. \quad (14)$$

In [21], the ratio of one-bit CR c_{1-bit} to c_{ZF} is $\sqrt{\frac{2}{\pi}}$ and thus, c_{1-bit} can be expressed as

$$c_{1-bit} = \sqrt{\frac{2}{\pi}} \sqrt{\frac{2PN_t}{f(N_r, N)}}. \quad (15)$$

Notice that c_{ZF} and c_{1-bit} are the CR in radius while c is the CR in I/Q axis. Then, a fixed power gain $\beta = \sqrt{2} \frac{c}{c_{1-bit}}$ can be used in the quantized precoding problem (8) for a high-order QAM system. The CR method can efficiently improve the BER performances of single-user 256-QAM and eight-user 16-QAM systems [21], but it does not completely improve the performance of multiuser high-order QAM signaling (e.g., 256-QAM).

B. LRA ONE-BIT PRECODING

Lattice reduction is a preprocessing technique that transfers the QAM symbol domain into a more orthogonal lattice grid by changing the lattice of the channel matrix for MIMO detection and precoding [18], [19]. Referring to (7), we first define $\mathbf{d} = \bar{\mathbf{s}} - \beta \bar{\mathbf{H}} \mathbf{x}$. Then, the quantized precoding problem becomes

$$\bar{\mathbf{s}} = \beta \bar{\mathbf{H}} \mathbf{x} + \mathbf{d} \quad (16)$$

Then, the LR can be easily applied to the quantized precoding algorithm similar to LR-MIMO detection [17]. The quantized symbol \mathbf{x} is converted to the integer symbol \mathbf{x}_{LR} by shift and scaling functions

$$\mathbf{x} = \left(\mathbf{x}_{LR} + \frac{1}{2}(1+j)\mathbf{1} \right) 2\sqrt{\frac{P}{2N_t}}, \quad (17)$$

where the power normalization factor p is defined as $\sqrt{\frac{P}{2N_t}}$ because the one-bit quantization of a complex-valued symbol c is defined as follows:

$$\mathcal{Q}(c) = \sqrt{\frac{P}{2N_t}} (\text{sgn}(\Re\{c\}) + j \text{sgn}(\Im\{c\})). \quad (18)$$

One can insert (17) into (16) to modify the quantized precoding problem as follows:

$$[c]_{\mathbf{s}_{LR}} = \beta \bar{\mathbf{H}} \mathbf{x}_{LR} + \frac{1}{2} \sqrt{\frac{2N_t}{P}} \mathbf{d} \quad (19)$$

Algorithm 2 SIC Procedure

Input: $\mathbf{Q} \in \mathbb{C}^{(N_t+N_r) \times (N_t+N_r)}$, $\mathbf{R} \in \mathbb{C}^{(N_t+N_r) \times N_t}$,
 $\bar{\mathbf{s}} \in \mathbb{C}^{N_t+N_r}$, β

- 1: Initialize $\mathbf{s}' = \mathbf{Q}^H \bar{\mathbf{s}} \in \mathbb{C}^{N_t+N_r}$, $\hat{z}_{N_t} = \text{Quant}(\frac{s'_{N_t}}{\beta r_{N_t, N_t}})$
- 2: for $i = N_t - 1, \dots, 1$ do
- 3: $\hat{z}_i = \text{Quant}(\frac{s'_i - \sum_{j=i+1}^{N_t} \beta r_{i,j} \hat{z}_j}{r_{i,i}})$
- 4: end

Output: $\hat{\mathbf{z}}$

where $\mathbf{s}_{LR} = \frac{1}{2} \sqrt{\frac{2N_t}{P}} \bar{\mathbf{s}} - \beta \bar{\mathbf{H}} \times \frac{1}{2} (1+j)\mathbf{1}$. Afterwards, lattice reduction can be applied to the channel matrix

$$\mathbf{s}_{LR} = \beta \bar{\mathbf{H}} \mathbf{T} \mathbf{T}^{-1} \mathbf{x}_{LR} + \frac{1}{2} \sqrt{\frac{2N_t}{P}} \mathbf{d} = \beta \mathbf{H}_{LR} \mathbf{z} + \mathbf{d}', \quad (20)$$

where \mathbf{T} is a unimodular transformation matrix, $\mathbf{H}_{LR} = \bar{\mathbf{H}} \mathbf{T}$ is the lattice-reduced channel matrix, and $\mathbf{z} = \mathbf{T}^{-1} \mathbf{x}_{LR}$ is the integral lattice-domain symbol vector. Then, the one-bit precoding problem can be reformulated as

$$\arg \min_{\mathbf{z} \in \mathbb{G}^{N_t}} \|\mathbf{s}_{LR} - \beta \mathbf{H}_{LR} \mathbf{z}\|_2^2. \quad (21)$$

Afterward, QR decomposition can be applied to \mathbf{H}_{LR} and the one-bit precoding problem (21) can be rewritten as

$$\begin{aligned} & \arg \min_{\mathbf{z} \in \mathbb{G}^{N_t}} \|\mathbf{s}_{LR} - \beta \mathbf{Q} \mathbf{R} \mathbf{z}\|_2^2 \\ & \arg \min_{\mathbf{z} \in \mathbb{G}^{N_t}} \left\| \mathbf{Q}^H \mathbf{s}_{LR} - \beta \mathbf{R} \mathbf{z} \right\|_2^2. \end{aligned} \quad (22)$$

\mathbb{G} denotes the complex integer set. We used a SIC algorithm to detect $\hat{\mathbf{z}}$ as shown in **Algorithm 2** to derive $\hat{\mathbf{z}}$, was then transformed back to the constellation domain by

$$\hat{\mathbf{x}} = 2p \left(\mathbf{T} \hat{\mathbf{z}} + \frac{1}{2} (1+j)\mathbf{1} \right). \quad (23)$$

Finally, the symbol $\hat{\mathbf{x}}$ was converted by one-bit DACs as the transmitted signal.

C. JOINT QR DECOMPOSITION AND LATTICE REDUCTION

We adopted a joint QR decomposition and constant-throughput Lenstra, Lenstra, and Lovasz (CTLL) algorithm [15], [24] to realize (20) and (22) together for parallel processing as shown in **Algorithm 3**. In the first part, the Givens rotation-based QR decomposition and the size reduction of the LLL algorithm were combined in a column-wise iteration loop. In each iteration, GR nullified the lower triangular entries in \mathbf{R} . Then, the ratio of off-diagonal and diagonal entries in each row were used to reduce the nondiagonal entry values by linear operations for size reduction. In the second part, the Siegel condition was checked to ensure that $|R_{n,n}|^2 / |R_{n-1,n-1}|^2$ was larger than a ratio δ , which was set to be 1.0 in this work. If that test was not true, two columns were swapped to maintain the increasing trends of diagonal entries; the low-triangular entries of \mathbf{R} caused by swapping were finally nullified by GRs. Although we used

Algorithm 3 Joint QR and CTLLL Algorithm

Input: \mathbf{H} stage, δ

- 1: **Initialize** $\mathbf{R} = \mathbf{H}$, $\mathbf{Q} = \mathbf{I}_N$, $\mathbf{T} = \mathbf{I}_N$, $i = 3$, $j = N - 1$
- 2: **for** $m = 1, \dots, N$ **do**
- 3: update the m^{th} column of \mathbf{R} and \mathbf{Q} by using GR
- 4: **for** $n = m + 1, \dots, 2$ **do** % full size reduction
- 5: $\mu = \lceil R_{n-1,m+1}/R_{n-1,n-1} \rceil$
- 6: $R_{1:n,m+1} \leftarrow R_{1:n,m+1} - \mu R_{1:n,n-1}$
- 7: $\mathbf{T}_{:,m+1} \leftarrow \mathbf{T}_{:,m+1} - \mu \mathbf{T}_{:,n-1}$
- 8: **end**
- 9: **end**
- 10: **for** $s = 1, \dots, \text{stage}$ % E-CTLLL
- 11: **for** $n = i, i + 2, \dots, j$
- 12: **if** $\delta |R_{n-1,n-1}|^2 > |R_{n,n}|^2$
- 13: $\mu = \lceil R_{n-1,n}/R_{n-1,n-1} \rceil$
- 14: $R_{1:n,n} \leftarrow R_{1:n,n} - \mu R_{1:n,n-1}$
- 15: $\mathbf{T}_{:,n} \leftarrow \mathbf{T}_{:,n} - \mu \mathbf{T}_{:,n-1}$
- 16: $\text{swap}(\mathbf{R}(:, n - 1), \mathbf{R}(:, n))$
- 17: $\text{swap}(\mathbf{T}(:, n - 1), \mathbf{T}(:, n))$
- 18: update \mathbf{R} and \mathbf{Q} by using GR
- 19: **end**
- 20: **end**
- 21: **if** $(i == 3)$ $i = 2$
- 22: **else** $i = 3$ **end**
- 23: **if** $(j == N - 1)$ $j = N$
- 24: **else** $j = N - 1$ **end**
- 25: **end**

Output: \mathbf{Q} , \mathbf{R} , \mathbf{T}

the same joint QR decomposition and lattice reduction algorithm as [24], the processed matrix dimension (68×64) of this work is much larger than that in the 8×8 MIMO detector. The major circuit design consideration will be discussed in the following section.

D. GAIN CONTROL MECHANISM FOR LRA ONE-BIT PRECODING

In LR preprocessing, a finite QAM constellation may be transformed to an infinite integer domain by \mathbf{T} because μ requires an infinite range to accommodate the division result. Therefore, the range of \mathbf{z} may be expanded, as illustrated in Fig. 3. The primary reason is that quantized precoding channel $\bar{\mathbf{H}}$ is not a typical MIMO channel; it contains an expanded noise submatrix as described in (6). In particular, when the noise σ^2 is small, the diminutive noise term makes the lattice range unpredictably expand because the off-diagonal entry is divided by the small diagonal element of noise in the size reduction. The range expansion tends to cause performance degradation due to word-length limits in the hardware implementation. Multiple candidates for lattice-reduction-aided one-bit precoding [19] might address this problem by comparing numerous candidates but the candidate size would be too large (usually more than 100) to be implemented.

The constellation range expansion problem is very serious in LRA quantized precoding. Thus, a simple regularization method was proposed to confine constellation points in a certain range to maintain numerical stability for the fixed point implementation. We regularized the noise submatrix

by replacing the expanded noise term of $\sqrt{\frac{N_r \sigma^2}{P}}$ with

$$\tilde{N}_0 = \max \left\{ \sqrt{\frac{N_r \sigma^2}{P}}, \sqrt{\frac{\|\mathbf{H}\|_F^2}{N_t N_r}} \right\}. \quad (24)$$

Then, the modified channel matrix becomes

$$\bar{\mathbf{H}} = \begin{bmatrix} \mathbf{H} \\ \tilde{N}_0 \mathbf{I}_{N_t} \end{bmatrix}. \quad (25)$$

$\sqrt{\frac{\|\mathbf{H}\|_F^2}{N_t N_r}}$ is the normalized root-mean-square of the channel \mathbf{H} .

The utilization of $\sqrt{\frac{\|\mathbf{H}\|_F^2}{N_t N_r}}$ in (24) can provide the adaptation of noise term to the channel gain for a specific \mathbf{H} . Thus, (24) was used to limit the gain reduction in the noise term of $\bar{\mathbf{H}}$ so that expansion factor μ could be controlled within a certain range. Fig. 3 (a)–(b) shows the original shifted-and-scaled constellation, lattice-reduced constellation, and gain-controlled lattice-reduced constellation under different SNR values. The simulation randomly generated 1000 sets of QPSK symbol and Gaussian channel \mathbf{H} to plot the symbol distributions. It can be observed that the CR of the \mathbf{z} domain expands widely with the increase in the SNR value for the original lattice-reduction-aided one-bit precoding while the CR with gain control can be kept in a limited range, as shown in Fig. 3 (c). This phenomenon leads to the requirement of long word-length for hardware implementation; if that requirement is not met, BER performance tends to degrade, especially for high-order QAM modulation.

E. PROPOSED CR-GCLRA ONE-BIT PRECODING ALGORITHM

Algorithm 4 presents the proposed CR-GCLRA one-bit precoding algorithm. The constellation rescaling factor β_{CR} is initially calculated off-line based on Equation (26) for N^2 -QAM modulation as follows:

$$\beta_{CR} = \sqrt{2} \frac{c_s}{c_{1-bit}} = \sqrt{\pi} \sqrt{\frac{6(N-1)}{N+1}} \sqrt{\frac{f(N_r, N)}{2PN_t}}, \quad (26)$$

where $f(N_r, N)$ can be expressed in Equation (14); $c_s = \sqrt{6(N-1)/(N+1)}$ is the CR of N^2 -QAM symbol after power normalization; c_{1-bit} is the CR of one-bit precoding (15); and $\sqrt{2}$ is the conversion factor from the complex to the real value of β . The gain control factor \tilde{N}_0 is then derived to construct the matrix $\bar{\mathbf{H}}$ for LR preprocessing. Afterwards, the lattice reduction and QR decomposition processing in Lines 3 and 4 are performed by the joint QR decomposition and E-CTLLL algorithm (**Algorithm 3**), and the transmitted QAM symbol vector $\bar{\mathbf{s}}$ is shifted and scaled to the lattice domain on the basis of $\bar{\mathbf{H}}$ and β_{CR} . Then, a simple SIC detector (**Algorithm 2**) is used to compute the solution $\hat{\mathbf{z}}$, which is finally converted to the original constellation domain and quantized by one-bit DACs as the transmitted signal vector \mathbf{x} .

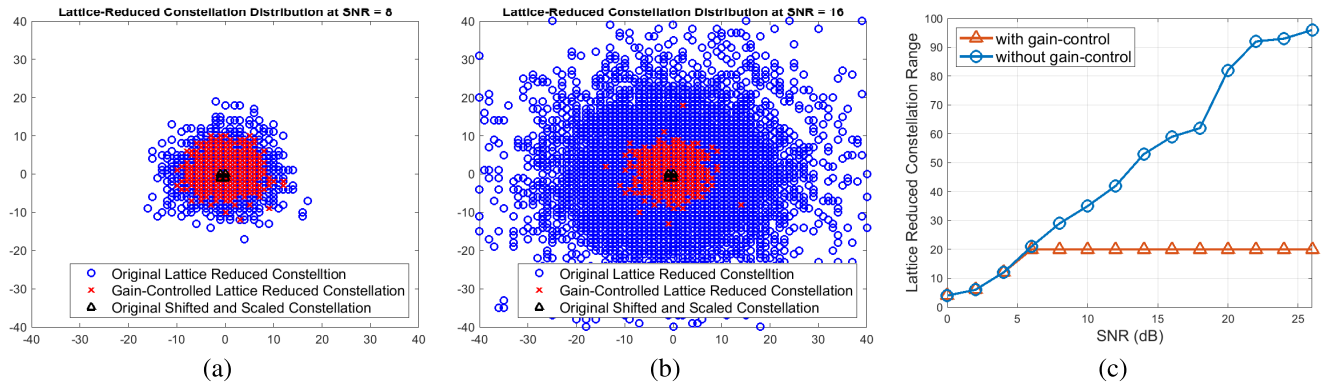


FIGURE 3. Lattice-reduced constellations of the original shifted and scaled symbols, lattice-reduced symbols, gain-controlled lattice-reduced symbols under (a) SNR = 4 dB and SNR = 16 dB with $(N_t, N_r) = (64, 4)$. One-thousand sets of QPSK symbols and channel matrices \mathbf{H} were randomly generated to compute the lattice-reduced constellations. (c) Average lattice-reduction domain detection range versus SNR with $(N_t, N_r) = (64, 4)$.

Algorithm 4 CR-GCLRA One-Bit Precoding Algorithm

v Input: $\mathbf{s}, \mathbf{H}, N_t, N_r, P, \sigma^2, \beta_{CR}$

- 1: $\tilde{N}_0 = \max \left\{ \sqrt{\frac{N_r \sigma^2}{P}}, \sqrt{\frac{\|\mathbf{H}\|_F^2}{N_t N_r}} \right\}$
- 2: $\tilde{\mathbf{s}} = \begin{bmatrix} \mathbf{s} \\ \mathbf{0}_{N_t} \end{bmatrix}, \tilde{\mathbf{H}} = \begin{bmatrix} \mathbf{H} \\ \tilde{N}_0 \mathbf{I}_{N_t} \end{bmatrix}$
- 3: $\mathbf{H}_{LR} = \tilde{\mathbf{H}} \mathbf{T}$
- 4: $\mathbf{H}_{LR} = \mathbf{Q}_{LR} \mathbf{R}_{LR}$
- 5: $\mathbf{s}_{LR} = \frac{1}{2} \sqrt{\frac{2N_t}{P}} \tilde{\mathbf{s}} - \beta_{CR} \tilde{\mathbf{H}} \times \frac{1}{2} (1 + j) \mathbf{1}$
- 6: $\hat{\mathbf{z}} = \arg \min_{\mathbf{z} \in \mathbb{G}^{N_t}} \|\mathbf{Q}_{LR}^H \mathbf{s}_{LR} - \beta_{CR} \mathbf{R}_{LR} \mathbf{z}\|_2^2$
- 7: $\mathbf{x} = \sqrt{\frac{P}{2N_t}} \mathbf{Q}(\mathbf{T} \hat{\mathbf{z}} + \frac{1}{2} (1 + j) \mathbf{1})$

Output: \mathbf{x}

F. PERFORMANCE AND COMPLEXITY ANALYSIS

1) PERFORMANCE ANALYSIS

Fig. 4 presents the BER performances of the proposed CR-GCLRA one-bit precoding algorithm for 4×64 MU-MIMO systems. The gain-control technique notably produced one-order BER improvements at more than 21 and 26 dB for 64-QAM and 256-QAM, respectively, because it effectively reduced the range expansion problem in the high-SNR region. The CR design for β_{CR} not only achieved better BER performance for high-QAM modulation but also resulted in noniterative precoding procedure, which was beneficial to the following low-latency high-throughput hardware implementation. Another advantage is that the CR-GCLRA did not need to update constellation gain per QAM symbol. Fig. 5 compares the proposed CR-GCLRA algorithm with other one-bit precoding algorithms in the literature. The parameter settings were tuned for optimal performance levels. In practice, we used the suggested iteration number for a compared one-bit precoder if the iteration number is available in the literature paper. Otherwise, we used the minimum number of iterations to achieve the optimal performance for a one-bit precoder. Accordingly, the fast C2PO algorithm in [12] used 24 iterations. The low-complexity iterative

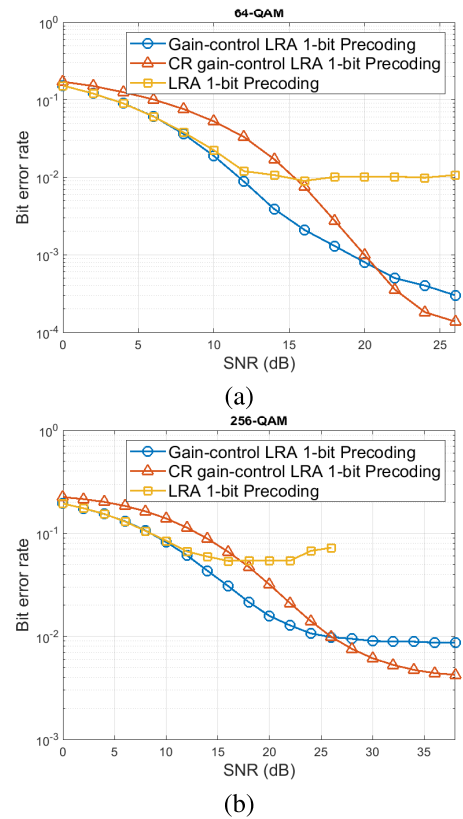


FIGURE 4. BER performance of the CR-GCLRA one-bit precoding algorithms for (a) 64-QAM and (b) 256-QAM MU-MIMO systems with $N_t = 64$ and $N_r = 4$.

discrete estimation (IDE2) algorithm in [13] was simulated with 50 iterations. The CR algorithm in [21] estimated the fixed rescaling factor for the quantized precoding and used an SIC procedure to derive the solution. The LRSIC in [19] used lattice reduction and the listed SIC precoding. A K-best strategy with $K = 32$ (32-Best), which realized the fixed-throughput sphere precoder with 3 iterations, was simulated as a benchmark. We also simulated more other iteration numbers to compare iterative precoders. One-bit quantized

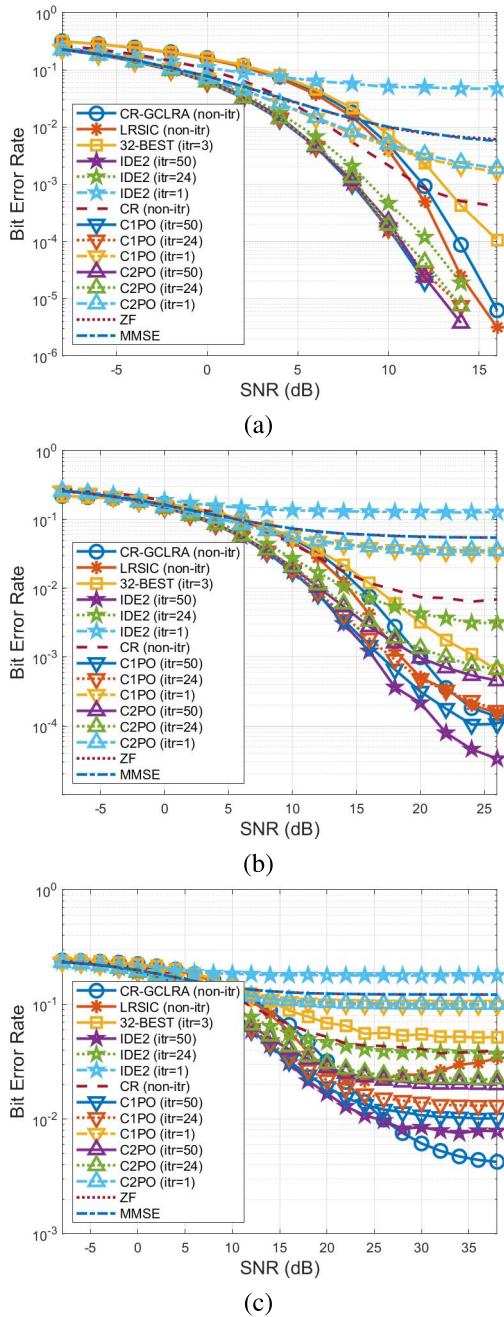


FIGURE 5. BER performance comparison of 1-bit precoding algorithms for MU-MIMO systems with $N_t = 64$, $N_r = 4$ for (a) 16-QAM, (b) 64-QAM, and (c) 256-QAM.

MMSE and ZF precoders were also simulated for reference in Fig. 5.

For 16-QAM, the CR-GCLRA was inferior to the LRSIC, C2PO (iteration=24) and IDE2 (iteration=50) by 1 dB, 3 dB, and 3.5 dB, respectively, at a BER of 10^{-4} . For 64-QAM, the CR-GCLRA exhibited performance gaps of 1 dB and 3 dB relative to the LRSIC and IDE2 (iteration=50) and showed a performance gain of 2 dB relative to C2PO (iteration=24) at a BER of 10^{-3} . However, the LRSIC tended to converge to the error rate floor and the CR-GCLRA algorithm started

to approximate LRSIC at a BER of 10^{-4} . For 256-QAM, the CR-GCLRA outperformed the LRSIC because of the CR design at the high-SNR region. The CR-GCLRA also surpassed the IDE2, C1PO, and C2PO when the SNR was larger than 26 dB and achieved a lower error rate floor than the IDE2. When the modulation order was high, the proposed CR-GCLRA algorithm exhibited better performance than its competitors did. The performance of the CR was worse than those of most one-bit precoders at a BER of 10^{-4} because the CR is more appropriate for a single-user system than for a multi-user system especially when the modulation order is higher than 16-QAM. The C1PO and C2PO achieved desirable performance for 16-QAM and 64-QAM but failed to reach a BER lower than 10^{-2} for 256-QAM. IDE2 demonstrated performance superior to other algorithms in most cases. However, it required many more iterations than other algorithms did, and was not suitable for low-latency high-throughput implementation.

2) COMPLEXITY ANALYSIS

Table 1 lists the computational complexity of multiplications of various algorithms. For the K-best precoder, t is the number of iterations and K is the candidate number in the K-best algorithm. For the C2PO algorithm, t is also the number of iterations. For IDE2, M represents the number of quantized symbols; t_1 denotes the number of iterations under the same β ; and t_2 denotes the number of iterations for β updating. For LRSIC and CR-GCLRA algorithms, $stage$ denotes the stage number for the joint QR decomposition and E-CTLLL algorithm. The computation in the preprocessing part requires computation only once per frame length (typically lasting a few hundreds of symbols in modern communication standards) as long as the channel remains static. CR, C2PO, and IDE2 algorithms only require matrix multiplications in the preprocessing part. K-best, LRSIC, and CR-GCLRA algorithms require QR decomposition; LRSIC and CR-GCLRA algorithms perform additional LR preprocessing.

The computational complexity values of various algorithms are presented in Fig. 6 for different frame sizes in 4×32 , 4×64 , and 8×128 MU-MIMO systems. Because the channel can be assumed to be fixed in a frame, the preprocessing complexity can be shared among all transmitted symbol vectors in a frame. Although the preprocessing has very high computational complexity, the CR-GCLRA algorithm has lower complexity than other algorithms for frames with more than 20 symbol vectors. For 4×64 MU-MIMO, when a frame has 100 symbols, the CR-GCLRA algorithm requires approximately 35.3%, 18.3%, 16.7%, 11.4%, and 4.2% of the complexity values of CR, IDE2, 32-best, C2PO, and LRSIC algorithms, respectively. Compared to the other algorithms, the CR-GCLRA algorithm tends to be more suitable for frame-based or packet-based transmission in most practical systems.

Notice that the complexity results in Table 1 were analyzed without considering the utilized hardware platform. The degree of parallelism of multiplication-and-accumulation

TABLE 1. Computational complexity analysis of the proposed CR-GCLRA and other various one-bit precoding algorithms.

Complexity (Number of multiplications)		
Algorithm	Pre-processing	Precoding
CR [21]	0	$4N_r N_t^2$
C2PO [12]	$N_r N_t^2 + N_r N_t + 2N_r$	$t(N_t^2 + 3N_t)$
IDE2 [13]	$N_r N_t$	$t_2(4N_r N_t + 2N_t + t_1(2N_r N_t + 2N_t))$
K-Best	$N_r N_t \left(\frac{N_t^2 + N_t}{2} + N_r \right)$	$t \left(K \left(\frac{N_t^2 + N_t}{2} \right) + N_t K M \right) + N_r N_t$
LRSIC [19]	$\left(N_r N_t + \frac{stage}{2} \right) \left(\frac{N_t^2 + N_t}{2} + N_r \right) + 2N_t^3 + N_r N_t^2 + \frac{3N_t^2 - N_t}{2} + 2N_r N_t$	$N_t^3 + 3N_r N_t^2 + 3N_t^2 + \frac{3N_t^2 + N_t}{2}$
CR-GCLRA	$\left(N_r N_t + \frac{stage}{2} \right) \left(\frac{N_t^2 + N_t}{2} + N_r \right) + \frac{3N_t^2 - N_t}{2} + 2N_r N_t$	$N_r N_t + \frac{3N_t^2 - N_t}{2}$

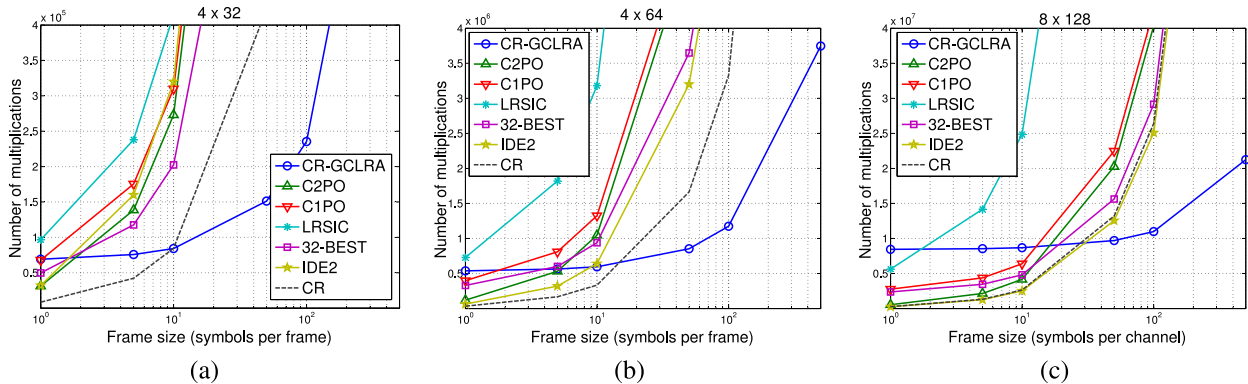


FIGURE 6. Computational complexity values of the proposed CR-GCLRA and other various one-bit precoding algorithms for different frame sizes in a 64-QAM MU-MIMO system with (a) $N_t = 32$ and $N_r = 4$, (b) $N_t = 64$ and $N_r = 4$ and (c) $N_t = 128$ and $N_r = 8$.

(MAC) processing elements in the hardware determines the processing latencies of the preprocessing and precoding parts. For the preprocessing part, the major complexity lies in Line 13-18 of **Algorithm 3** and is related to the matrix \mathbf{T} , \mathbf{R} , and \mathbf{Q} . The sizes of \mathbf{T} , \mathbf{R} , and \mathbf{Q} are $(N_t + N_r) \times (N_t + N_r)$, $(N_t + N_r) \times N_t$, and $(N_t + N_r) \times (N_t + N_r)$, respectively. Because the preprocessing part is performed only once in a coherent frame, the parallel processing of MAC is not required for most fixed and low-mobility systems. For the precoding part, each SIC detection depends on the previous detected symbol. Thus, only the interference summation of $\sum_{j=i+1}^{N_t} \beta r_{i,j} \hat{z}_j$ in Line 3 of **Algorithm 2** can be executed in parallel, but the summation count increases along the iteration number. In order to strike a balance between hardware efficiency and throughput gain, $N_t/2$ MAC processing elements can be utilized in interference computation. For high-throughput systems, more than $N_t/2$ MAC elements can be utilized at the sacrifice of hardware efficiency while few than $N_t/2$ MAC elements should be used for low-throughput and power-efficient systems.

IV. PROPOSED CR-GCLRA PROCESSOR

Fig. 7 shows the architecture of the proposed CR-GCLRA one-bit precoding processor, which supports 4×64 downlink massive MU-MIMO systems with 64-QAM. The preprocessing and precoding parts of the processor execute Lines 1 to 4 and Lines 5 to 7, respectively, in **Algorithm 4**.

In the preprocessing part, we used three dual-port SRAMs for the storage of \mathbf{R} (or \mathbf{H}), \mathbf{Q} , and \mathbf{T} matrices; we designed

the GR block, condition check block, and size reduction block to perform the joint QR decomposition and E-CTLLL algorithm in **Algorithm 3**. The Gram-Schmidt (GS) [25] and GR algorithms [24] are two primary schemes to perform QR decomposition in the literature, but the GS requires a large number of division operations, which cause the numeric instability and significantly increase signal word-lengths and hardware cost. The GR-based scheme primarily consists of numerically stable vector rotation operations, which can also be shared with the size reduction operations for lattice reduction. Thus, the GR-based circuit architecture was used.

In the precoding part, the SIC processing and output stages were pipelined to calculate \mathbf{z} and \mathbf{x} , respectively. In the SIC processing, each precoding output sequentially depends on the previous output, leading to the linearly increased latency along with N_t . Data-dependency of SIC is not suitable for parallel processing because of the trade-off between its hardware efficiency and throughput. However, the proposed CR-GCLRA one-bit precoder does not need iterative precoding process, it is still a very promising low-latency precoder.

A. GAIN-CONTROLLED QR DECOMPOSITION AND LR

The preprocessing part was controlled by three processing states, gain control (GC) state, QR decomposition state, and ECTLLL state. In the GC state, the GC block calculated the Frobenius norm of \mathbf{H} to obtain the noise gain \tilde{N}_0 in $\tilde{\mathbf{H}}$, as shown in Fig. 8. Additionally, the GC block computed the transformation factor from the QAM symbol domain to the

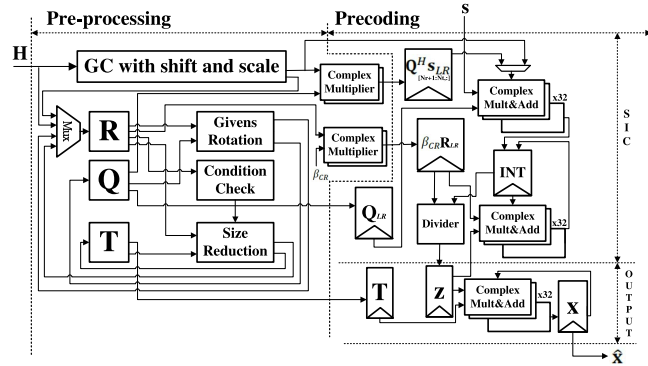


FIGURE 7. Proposed CR-GCLRA one-bit precoding processor architecture.

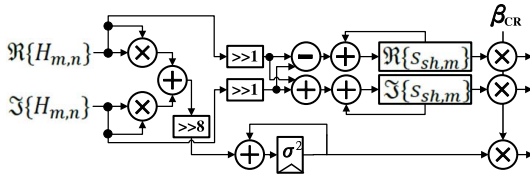


FIGURE 8. Gain control and shift-and-scale processing circuits.

lattice-reduced domain through shift-and-scale operations as follows:

$$\mathbf{s}_{sh} = \begin{bmatrix} s_{sh,1} \\ s_{sh,2} \\ s_{sh,3} \\ s_{sh,4} \\ \mathbf{N}_{0sh} \end{bmatrix} = \bar{\mathbf{H}} \times \frac{1}{2}(1+j)\mathbf{1} = \begin{bmatrix} \mathbf{H} \\ \tilde{\mathbf{N}}_0 \mathbf{I}_{N_t} \end{bmatrix} \times \frac{1}{2}(1+j)\mathbf{1}, \quad (27)$$

where \mathbf{N}_{0sh} is the shifted-and-scaled gain-controlled noise submatrix; $\mathbf{1}$ is a matrix with all entries of 1. Then, β_{CR} was multiplied with $\bar{\mathbf{H}} \times \frac{1}{2}(1+j)\mathbf{1}$ to obtain the second term in Line 5 of **Algorithm 4**, and the result of $\beta_{CR}\bar{\mathbf{H}} \times \frac{1}{2}(1+j)\mathbf{1}$ was delivered to the multiplier-and-adder in the precoding part as an initial value.

The designed joint QR decomposition and LR preprocessor was designed based on the algorithm in [24] but the processed channel matrix dimension (68×64) was much larger than that (8×8) in [24], in which data register files of \mathbf{Q} , \mathbf{R} , and \mathbf{T} could be used for the parallel row-wise LR processing. However, when matrix dimension increases up to 64, the storage of \mathbf{Q} , \mathbf{R} , and \mathbf{T} should be implemented with SRAMs for hardware efficiency. Thus, dual-port SRAMs were used for accessing these matrices in this work but the parallel row-wise LR processing architecture could no longer be realized due to the low access bandwidth of dual-port SRAMs. Therefore, a pipelined paired vector/rotation-mode CORDIC circuit was proposed to increase the throughput of the LR preprocessing by using only shift and add operations at each pipeline stage, and it also reduced hardware cost compared to the direct GR vector rotation with matrix multipliers.

TABLE 2. Timing schedule of vector mode 1.

Clock	1	2	3	4	5
x	$A_{1,1}$	$A_{2,1}$		$\{A'_{1,1}, A'_{2,1}\}$	
$ x $		$A'_{1,1}$	$A'_{2,1}$		$R_{1,1}$
$\angle x$		θ_1	θ_2		θ_3

TABLE 3. Timing schedule of rotation mode 1.

Clock	1	2	3	4	
Complex Multiplier 1	in1	$H_{1,2}$	$\{\Re\{T_1\}, \Re\{T_2\}\}$	$H_{1,3}$	$\{\Re\{T_1\}, \Re\{T_2\}\}$
	in2	θ_1	θ_3	θ_1	θ_3
Complex Multiplier 2	in1	$H_{2,2}$	$\{\Im\{T_1\}, \Im\{T_2\}\}$	$H_{2,3}$	$\{\Im\{T_1\}, \Im\{T_2\}\}$
	in2	θ_2	θ_3	θ_2	θ_3
out	T_1	$\{\Re\{R_{1,2}\}, \Re\{R_{2,2}\}\}$	T_1	$\{\Re\{R_{1,3}\}, \Re\{R_{2,3}\}\}$	
out	T_2	$\{\Im\{R_{1,2}\}, \Im\{R_{2,2}\}\}$	T_2	$\{\Im\{R_{1,3}\}, \Im\{R_{2,3}\}\}$	

In the QR mode, QR decomposition was performed by a GR block consisting of a paired CORDIC circuit, as illustrated in Fig. 10, and two complex multipliers. The timing diagram that performs the QR decomposition with the CORDIC processor for a complex 2×2 submatrix \mathbf{A} from matrix $\bar{\mathbf{H}}$ is presented in Fig. 11. In Vector Mode 1, the vector-mode CORDIC first converts the complex $A_{1,1}$ to the real $A'_{1,1}$ and outputs the rotation angle θ_1 . Similarly, the complex $A_{2,1}$ is converted to the real $A'_{2,1}$ and the system outputs the rotation angle θ_2 . Finally, $A'_{2,1}$ is nullified by using the angle of $(A'_{1,1}, A'_{2,1})$ with a vector-mode CORDIC circuit, and the system outputs the rotation angle θ_3 . Table 2 shows the timing schedule of Vector Mode 1. Afterward, in Rotation Mode 1, two complex multipliers calculate the phase rotations of the following column pairs with the corresponding θ_1 , θ_2 , or θ_3 generated in Vector Mode 1. Table 3 shows the timing schedule of Rotation Mode 1. In Vector Mode 2 and Rotation Mode 2, the CORDIC circuit performs similar GRs. After the entries in the low-triangular part of \mathbf{R} in a column are nullified, the full size reduction must be checked and executed in the size reduction block. In E-CTLL mode, the Siegel condition check (Lines 12 and 13 in **Algorithm 3**) block contains a divider, as presented in Fig. 12 and a complex multiplier-and-adder to compute the linear size reduction operation in Lines 14 and 15 of **Algorithm 3**. The divider's output is then combined with a rounding operation due to the integer computation for size reduction.

B. SIC PRECODING PROCESSOR

The proposed SIC precoding processor computed the lattice-reduced-domain solution \mathbf{z} in Line 6 of **Algorithm 4** by using the SIC procedure in **Algorithm 2**, which was then transformed back to solution \mathbf{x} in the QAM symbol domain. In the SIC procedure, the first set of 32 complex multiplier-and-adders calculated \mathbf{Q}_{SLR}^H and the second set of 32 complex multiplier-and-adders calculated the interference (INT) contribution in each row of $\beta_{CR}\mathbf{R}_{LR}$. Then, the results of two complex multiplier-and-adders were subtracted and divided row-wise to derive the lattice-domain integer solution \mathbf{z} . The proposed SIC was appropriately pipelined and scheduled to compute the interference component of each entry in an upper triangular matrix for the SIC procedure,

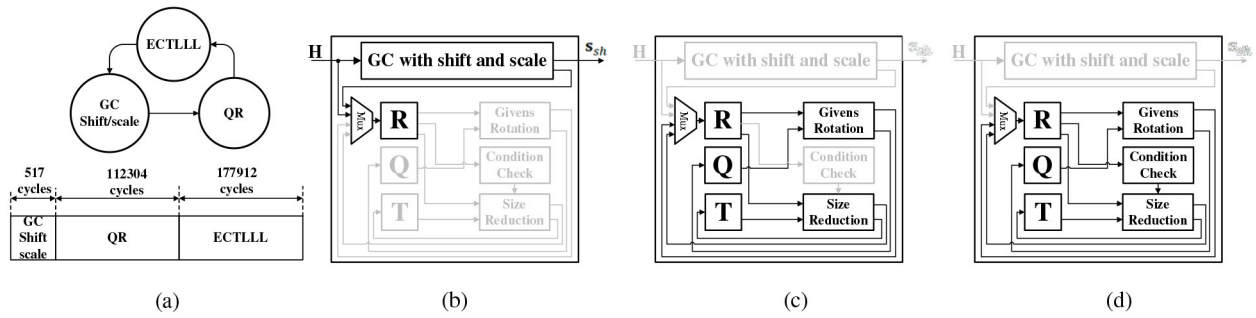


FIGURE 9. Gain-controlled QR decomposition and LR preprocessor with (a) finite-state control with (b) the state of gain control with shift-and-scale, (c) state of QR decomposition, and (d) state of E-CTLLL lattice reduction.

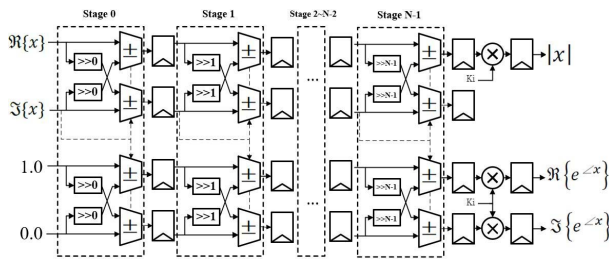


FIGURE 10. Paired CORDIC circuit.

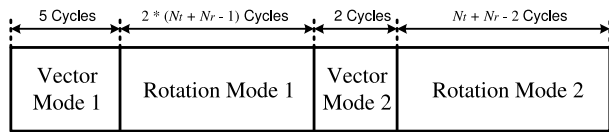


FIGURE 11. GR timing schedule.

as illustrated in Fig. 13. Because the OUTPUT circuit, which transforms \mathbf{z} to \mathbf{x} by matrix multiplication with \mathbf{T} , cannot be executed before the SIC circuit finishes the computation of \mathbf{z} , we must try to keep the latency difference of two circuits as small as possible in order to achieve the highest precoding throughput. Considering the increasing MAC operations in the interference computation for each iteration, we used $N_i/2$ multiplier-and-adders in the SIC circuits to strike a balance between hardware utilization efficiency and processing throughput. Subsequently, in order to achieve approximately the same processing latency as the SIC circuit, 32 complex multiplier-and-adders at the output stage were used to implement Line 7 of Algorithm 4 to obtain the solution \mathbf{x} .

C. TIMING SCHEDULE

Fig. 13 (b) shows the timing diagrams of the proposed CR-GCLRA one-bit precoding processor for a frame-based transmission system, in which a channel can be assumed to be fixed within a frame. Therefore, the preprocessing part required only one computation in a frame of several successive transmitted symbol vectors. The preprocessing part required a notably long processing latency of 274,592 cycles; therefore, the throughput would be limited for the fast-fading channel. However, in practice, the

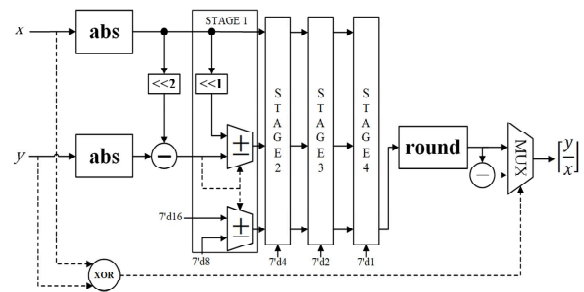


FIGURE 12. CORDIC-based divider.

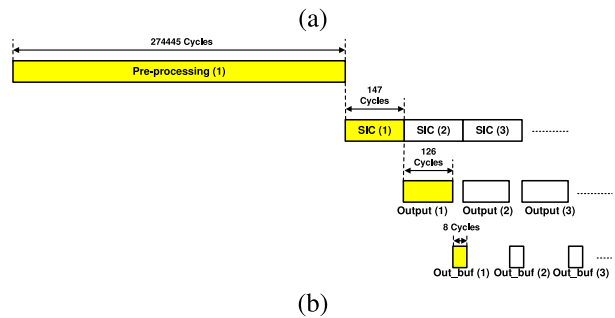
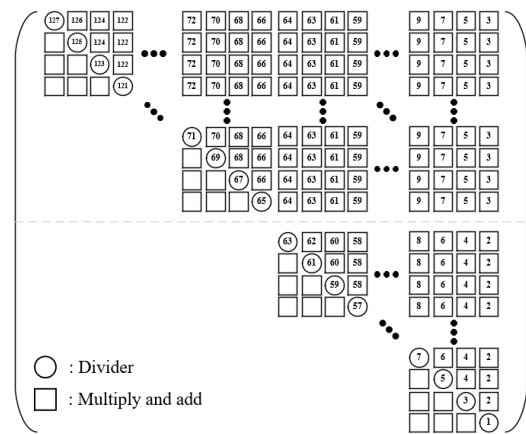


FIGURE 13. (a) Timing diagram of the SIC precoder and (b) timing schedule of the CR-GCLRA one-bit precoding processor.

average throughput is determined by the coherent frame size, which is usually determined by practical communication specifications. Through coarse-grained pipelining of

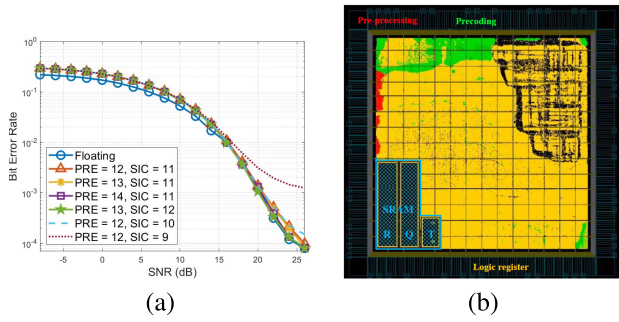


FIGURE 14. (a) BER performances of the floating-point and fixed-point CR-GCLRA algorithms. PRE and SIC indicate the numbers of fractional bits used in the preprocessing and precoding parts, respectively. (b) Layout and distribution of the implemented CRG-CLRA one-bit precoding processor.

the SIC precoder and output stage, the average precoding throughput was determined by the minimum coarse-grained latency, and the critical latency was shortened from 280 to 147 cycles. Fig. 14(a) shows the fixed-point simulation results of the proposed CR-GCLRA one-bit precoding processor for a 64-QAM MU-MIMO system with $(N_t, N_r) = (64, 4)$. The minimum word-lengths for the chip implementation were determined by requiring that performance loss be smaller than a SNR of 1 dB at a BER of 10^{-4} . Accordingly, 12 and 11 fractional bits were used for the implementation of preprocessing and precoding parts, respectively. Because the preprocessing part is not related to the input symbol \mathbf{s} , the same word-length is needed for BSPK/QPSK/16QAM/256QAM or even higher modulation order. As for the precoding part, for N^2 -QAM modulation, it approximately requires more bits with the growth order of $O(\log(N - 1)^2)$ in the fraction part.

D. FPGA AND CHIP RESULTS

The proposed CR-GCLRA one-bit precoding processor was designed and implemented by using Xilinx Vertex 7 FPGA platform and TSMC 40-nm CMOS Technology. The designed processor chip was synthesized using the Synopsys Design Compiler and the layout was placed and routed with the Synopsys IC Design Compiler. The layout and distribution of the preprocessing and precoding parts of the chip are illustrated in Fig. 14 (b). The preprocessing part had a 4352×32 dual-port SRAM for the \mathbf{H} and \mathbf{R} matrices, a 4624×32 dual-port SRAM for the \mathbf{Q} matrix, and a 4096×10 dual-port SRAM for the \mathbf{T} matrix.

1) FPGA RESULTS

Table 4 shows the FPGA implementation, IC synthesis, and IC postlayout simulation results of the proposed chip. To the best of the authors' knowledge, the FPGA-based C2PO processor is the only one-bit precoding VLSI design in the literature. Although the proposed FPGA implementation yielded lower clock rate, it still achieved comparable bit rate due to the lower latency of 147 cycles. For the C2PO processor, the latency of one iteration is only 40 clock cycles, which is much smaller that of the proposed precoder, but it

requires 24 iterations for completing one precoding channel matrix.

The proposed FPGA-based CR-GCLRA precoding processor has lower clock rate mainly because the CORDIC circuits utilized more LUTs, causing higher routing complexity than C2PO processor. However, the C2PO processors for various N_r s were mainly composed of MAC operations and required more DSP48 units in FPGA. Therefore, the CR-GCLRA processor achieved better normalized hardware efficiency than the C2PO processors in terms of DSP48 units for either large or small N_r . The proposed CR-GCLRA FPGA processor has a comparable normalized throughput to the C2PO processor for 64×16 MU-MIMO. However, the proposed does not require any iterative precoding procedure; therefore it has a smaller clock latency of $1.91 \mu\text{s}$ than the C2PO processor by 41%.

2) CHIP RESULTS

The synthesized precoder processor achieved a throughput of 7.78 M symbols/s at a 286-MHz clock frequency. The post-layout processor yielded a throughput of 7.31 M symbols/s at a 269-MHz clock frequency with power consumption of 271 mW. The preprocessing part requires 274,592 processing cycles and thus the FPGA, synthesized, and postlayout preprocessors can finish the preprocessing of a channel matrix in 3.5 ms, 0.96 ms, and 1.02 ms, respectively, which satisfy slow-fading scenarios such as IEEE 802.11ax [28]. The IC chip results of the proposed processor show that the CORDIC-based circuit is more suitable for IC implementation because simple logic functions of multiplexing, shifting, and addition can be optimized more flexibly by the routing tool and would not be limited by fixed routing tracks in an FPGA VLSI architecture. The symbol-rate and bit-rate throughput metric have been normalized by the user number N_r . Fig. 15 shows the area and power distributions of the proposed CR-GCLRA precoding processor chip. We can see that the preprocessing part occupies 34% area and 42% power of the precoding processors while the precoding part occupies 66% area and 58% power consumption of this chip. Assume that clock speed is fixed. The area and power of the preprocessing part approximately remains the same when N_t or N_r increases because the preprocessing part is an iterative LR processor. The SRAM area increases along with a scaling factor of $(N_t \times N_r)^2$. For the precoding part, if the circuit design follows the aforementioned design strategies mentioned in Section IV-B, the area and power increase along with the increased N_r . Because the QAM modulation order only influences the fixed-point wordlength in the SIC precoder, the area/power increase caused by modulation order scaling is relatively very minor. In summary, the proposed one-bit precoding chip utilized smaller preprocessing cost and power to reduce the precoding latency by avoiding iterative processing. Although extra preprocessing latency is required, this design strategy conforms to the practical coherent frame-based communication systems in which the preprocessing needs to be performed only once during

TABLE 4. Comparison of one-bit precoding processors.

	C2PO [12]			This chip		
	(32,16)	(64,16)	(128,16)	(64,4)		
Modulation	BPSK/16-QAM			16-QAM/64-QAM/*256-QAM		
IC Process	28nm (FPGA Tech.)			28nm (FPGA Tech.)	TSMC 40nm CMOS	
Design Tech.	Xilinx VIRTEX-7 XC7VX690T			Xilinx VIRTEX-7 XC7VX690T	Design Compiler (synthesis)	IC Compiler (post-layout)
Core Voltage	1.0 V			1.0 V	0.9 V	0.9 V
Clock Speed	222 MHz	206 MHz	208 MHz	77 MHz	286 MHz	269 MHz
Hardware Cost	Slices : 3375 LUTs : 10817 Flipflops : 5677 DSP48 : 136	Slices : 6519 LUTs : 21920 Flipflops : 12461 DSP48 : 272	Slices : 12639 LUTs : 43710 Flipflops : 26083 DSP48 : 544	Slices : 6721 LUTs : 36796 Flipflops : 16153 DSP48 : 195	3.0 M Gate Count	4.848 mm ²
Precoding Latency (clock cycles)(second)	39×24 4.22μ	40×24 4.66μ	41×24 4.73μ	147 1.91μ	147 0.51μ	147 0.55μ
Normalized Symbol Throughput (symbols/s)	1.89 M	3.42 M	6.76 M	2.10 M	7.78 M	7.31 M
Normalized Bit Throughput (bits/s)	7.59 M (16-QAM)	13.68 M (16-QAM)	27.06 M (16-QAM)	12.57 M/*16.76 M (64-QAM/256-QAM)	46.69 M/*62.26 M (64-QAM/256-QAM)	43.92 M/*58.56 M (64-QAM/256-QAM)
Normalized Hardware Efficiency (bits/s)/FPGA cost	1.63 K	1.45 K	1.42 K	1.31 K/*1.75 K	-	-

Precoding Latency is measured for completing one symbol precoding.

Normalized Symbol Throughput is defined as $\frac{N_T \times \text{clock speed}}{\text{Latency}}$.

Normalized Bit Throughput is defined as Normalized Symbol Throughput $\times \frac{N_T}{64} \times \text{Bits/Symbol}$.

Normalized Hardware Efficiency is defined as Normalized Bit Throughput per FPGA cost of (LUTs + DSPs*280 + FFs) [29].

* denotes the performance of BER > 10⁻³.

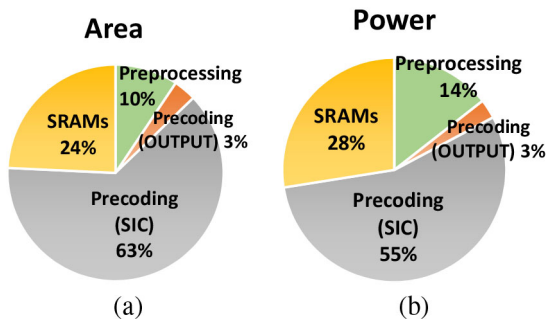


FIGURE 15. (a) Area and (b) power distributions of the chip.

a frame of symbols. In summary, the proposed CR-GCLRA one-bit precoding processor yielded a low latency and a high bit-throughput rate because the proposed algorithm supported noniterative precoding processing and high-order QAM signals.

V. CONCLUSION

This paper presents the CR-GCLRA one-bit precoding algorithm and processor for massive MU-MIMO systems. We adopted a CR design for high-order QAM signaling and used lattice reduction preprocessing to enhance the channel matrix orthogonality, thereby enabling the replacement of a highly complex iterative precoder with a simple SIC precoder. We proposed gain control technology to solve the constellation expansion problem in the lattice-reduced domain, which facilitated the hardware implementation. Finally, the proposed CR-GCLRA one-bit precoding processor chip was designed and implemented to verify its practical performance in massive MU-MIMO systems. Therefore, we believe that the proposed one-bit precoding processor would be beneficial to next-generation massive MU-MIMO communications.

There are some problems and potential extensions of this work. First, the BER of the CR-GCLRA for 256-QAM is

not smaller than 10⁻³. Thus, it does not conform to modern communication standards although it outperforms other counterparts in high-order modulations. Accurate CR estimation methods can be further investigated in the future. Second, the quantized precoding has been applied to MIMO-OFDM precoding system in [26], [27], in which three to five bits were still required to achieve the optimal performance for the QPSK modulation. The proposed lattice-reduction-aided precoding with the modified CR method will be a potential technique to reduce the quantized bits and implementation cost for high-order QAM modulations. In addition, the proposed CR-GCLRA processor supports single QAM-modulation for all UEs. Actually, β_{CR} values can be easily derived for various QAM modulations of UEs because the CR method is based on the channel hardening property. More accurate β_{CR} values can be further derived by considering the interference power to the transmitted signal in the CR estimation process, thereby improving the spectral utilization efficiency.

ACKNOWLEDGMENT

The authors would like to thank the Taiwan Semiconductor Research Institute (TSRI) for Technical Support.

REFERENCES

- [1] F. Rusek *et al.*, "Scaling up MIMO: Opportunities and challenges with very large arrays," *IEEE Signal Process. Mag.*, vol. 30, no. 1, pp. 40–60, Jan. 2013.
- [2] E. G. Larsson, O. Edfors, F. Tufvesson, and T. L. Marzetta, "Massive MIMO for next generation wireless systems," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 186–195, Feb. 2014.
- [3] L. Lu, G. Y. Li, A. L. Swindlehurst, A. Ashikhmin, and R. Zhang, "An overview of massive MIMO: Benefits and challenges," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 742–758, Oct. 2014.
- [4] S. Jörgensen, "Modelling of power dissipation in CMOS DACs," Institutionen för Systemteknik, Linköping Univ., Linköping, Sweden, Rep. LiTH-ISY-EX-3275-2002, 2002.

- [5] A. K. Saxena, I. Fijalkow, and A. L. Swindlehurst, "On one-bit quantized ZF precoding for the multiuser massive MIMO downlink," in *Proc. IEEE Sens. Array Multichannel Signal Process Workshop (SAM)*, 2016, pp. 1–5.
- [6] O. B. Usman, H. Jedda, A. Mezghani, and J. A. Nossek, "MMSE precoder for massive MIMO using 1-bit quantization," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, Shanghai, China, 2016, pp. 3381–3385.
- [7] 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.
- [8] S. Jacobsson, G. Durisi, W. Xu, and C. Studer, "MSE-optimal 1-bit precoding for multiuser MIMO via branch and bound," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, Calgary, AB, Canada, 2018, pp. 3589–3593.
- [9] A. C. H. Sarrivedeen, 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.
- [10] J.-C. Chen, "Alternating minimization algorithm for one-bit precoding in massive multiuser MIMO system," *IEEE Trans. Veh. Technol.*, vol. 67, no. 8, pp. 7394–7406, Aug. 2018.
- [11] O. Castaneda, T. Goldstein, and C. Studer, "POKEMON: A non-linear beamforming algorithm for 1-bit massive MIMO," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, 2017, pp. 3464–3468.
- [12] O. Castaneda, S. Jacobsson, G. Durisi, M. Coldrey, T. Goldstein, and C. Studer, "1-bit massive MU-MIMO precoding in VLSI," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 7, no. 4, pp. 508–522, Dec. 2017.
- [13] C. Wang, C. Wen, S. Jin, and S. Tsai, "Finite-alphabet precoding for massive MU-MIMO with low-resolution DACs," *IEEE Trans. Wireless Commun.*, vol. 17, no. 7, pp. 4706–4720, Jul. 2018.
- [14] M. Shao, Q. Li, and W.-K. Ma, "One-bit massive MIMO precoding via a minimum symbol-error probability design," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, Calgary, AB, Canada, 2018, pp. 3579–3583.
- [15] D. Wubben, D. Seethaler, J. Jalden, and G. Matz, "Lattice reduction," *IEEE Signal Process. Mag.*, vol. 28, no. 3, pp. 70–91, May 2011.
- [16] Y. H. Gan, C. Ling, and W. H. Mow, "Complex lattice reduction algorithm for low-complexity full-diversity MIMO detection," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2701–2710, Jul. 2009.
- [17] H. Yao and G. W. Wornell, "Lattice-reduction-aided detectors for MIMO communication systems," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Taipei, Taiwan, 2002, pp. 424–428.
- [18] L. Babai, "On Lovász' lattice reduction and the nearest lattice reduction point problem," *Combinatorica*, vol. 6, no. 1, pp. 1–13, 1986.
- [19] C.-E. Chen and Y.-H. Huang, "Lattice-reduction-aided one-bit precoding for massive MU-MIMO systems," *IEEE Trans. Veh. Technol.*, vol. 68, no. 7, pp. 7184–7188, Jul. 2019.
- [20] C.-E. Chen, "MMSE one-bit precoding for MU-MIMO systems with enhanced receive processing," *IEEE Wireless Commun. Lett.*, vol. 9, no. 4, pp. 548–552, Apr. 2020.
- [21] F. Sotrabadi, Y. Liu, and W. Yu, "One-bit precoding and constellation range design for massive MIMO with QAM signaling," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 3, pp. 557–570, Jun. 2018.
- [22] E. Agrell, T. Eriksson, A. Vardy, and K. Zeger, "Closest point search in lattice," *IEEE Trans. Inf. Theory*, vol. 48, no. 8, pp. 2201–2214, Aug. 2002.
- [23] S. Jacobsson, G. Durisi, M. Coldrey, T. Goldstein, and C. Studer, "Nonlinear 1-bit precoding for massive MU-MIMO with higher-order modulation," in *Proc. 50th Asilomar Conf. Signals Syst. Comput.*, Pacific Grove, CA, USA, 2016, pp. 763–767.
- [24] C. H. Liao, J. Y. Wang, and Y. H. Huang, "A 0.18nJ/matrix QR decomposition and lattice reduction processor for 8×8 MIMO pre-processing," in *Proc. IEEE Asian Solid-State Circuits Conf. (A-SSCC)*, Singapore, 2013, pp. 161–164.
- [25] P. Luethi *et al.*, "Gram-Schmidt-based QR decomposition for MIMO detection: VLSI implementation and comparison," in *Proc. IEEE Asian Solid-State Circuits Conf. (A-SSCC)*, Macao, China, 2008, pp. 830–833.
- [26] N. Khalid, B. Mondal, R. W. Heath, G. Leus, and F. Petre, "Quantized multi-mode precoding for spatial multiplexing MIMO-OFDM system," in *Proc. IEEE 62nd Veh. Technol. Conf. (VTC-Fall)*, Dallas, TX, USA, 2005, pp. 867–871.
- [27] P. R. Botha, D. J. Louw, and B. T. Maharaj, "Achievable diversity limits in a quantized MIMO-OFDM linear pre-coded system," in *Proc. IEEE 5th Int. Symp. Wireless Pervasive Comput.*, Modena, Italy, 2010, pp. 455–459.
- [28] E. Khorov, A. Kiryanov, A. Lyakhov, and G. Bianchi, "A tutorial on IEEE 802.11ax high efficiency WLANs," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 1, pp. 197–216, 1st Quart., 2019.
- [29] J. Chen, Z. Zhang, H. Lu, J. Hu, and G. E. Sobelman, "An intra-iterative interference cancellation detector for large-scale MIMO communications based on convex optimization," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 63, no. 11, pp. 2062–2072, Nov. 2016.



PAO-PAO HO was born in Taiwan, in 1995. He received the B.S. degree in engineering and system science and the M.S. degree in communications engineering from National Tsing Hua University, Taiwan, in 2017 and in 2019, respectively. He currently works with Realtek. His research interests are VLSI design and implementation of MIMO communication systems.



CHIAO-EN CHEN (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees in electrical engineering from National Taiwan University, Taipei, Taiwan, in 1998 and 2000, respectively, and the Ph.D. degree from the University of California at Los Angeles, Los Angeles.

He was with the Electrical Engineering Department, University of California at Los Angeles from 2003 to 2008. He joined National Chung Cheng University, Chiayi, Taiwan, as an Assistant Professor in 2008, and was promoted to an Associate Professor and a Full Professor in 2013 and 2018, respectively. In 2020, he joined the faculty of the Department of Electrical Engineering, National Chung Hsing University, Taichung, Taiwan. He has coauthored the book *Detection and Estimation for Communication and Radar Systems* (Cambridge University Press, 2012). His current research interests include statistical signal processing, optimization, and multiple-input-multiple-output communications. He was a co-recipient of the Best Paper Award in IEEE WCNC 2012. He is currently the Vice Chair of the Tainan Chapter for the IEEE Vehicular Technology Society.



YUAN-HAO HUANG (Member, IEEE) was born in Taiwan in 1973. He received the B.S. and Ph.D. degrees in electrical engineering from National Taiwan University, Taipei, Taiwan, in 1995 and 2001, respectively.

He was a Member of the Technical Staff with VXIS Technology Corporation from 2001 to 2005. Since 2005, he has been with the Department of Electrical Engineering and the Institute of Communications Engineering, National Tsing Hua University, Taiwan, where he is currently a

Professor. His research interests include VLSI design for digital signal processing and telecommunication systems and biomedical signal processing. He has also served in the Seasonal School Subcommittee of the Membership Board of the IEEE Signal Processing Society since 2016 and has been an associate editor of *IEEE Signal Processing Magazine* since 2019. He is also a Member of the VLSI Systems and Applications and Circuits and Systems for Communications technical committees of the IEEE Circuits and Systems Society. He is currently a Member of the Advisory Board of Design and Implementation of Signal Processing Systems Technical Committee of the IEEE Signal Processing Society.