IEEE Open Journal of
Circuits and Systems

Received 2 May 2020; revised 29 July 2020 and 26 August 2020; accepted 31 August 2020.
Date of publication 8 September 2020; date of current version 6 October 2020.

Digital Object Identifier 10.1109/OJCAS.2020.3022514

Algorithm and VLSI Design for 1-Bit Data Detection in Massive MIMO-OFDM

SEYED HADI MIRFARSHBAFAN[1,4], MAHDI SHABANY[2], SEYED ALIREZA NEZAMALHOSSEINI[3], AND CHRISTOPH STUDER[1,4] (Senior Member, IEEE)

[1]Department of Information Technology and Electrical Engineering, ETH Zurich, 8092 Zurich, Switzerland

[2]Department of Electrical Engineering, Sharif University of Technology, Tehran 1458889694, Iran

[3]Department of Electrical Engineering, Iran University of Science and Technology, Tehran 1684613114, Iran

[4]Department of Electrical and Computer Engineering, Cornell University, Ithaca, NY 14850, USA

This article was recommended by Associate Editor V. Chen.

CORRESPONDING AUTHOR: C. STUDER (e-mail: studer@ethz.ch)

The work of Seyed Hadi Mirfarshbafan and Christoph Studer was supported in part by Xilinx, Inc., and in part by the U.S. National Science Foundation (NSF) under Grant ECCS-1408006, Grant CCF-1535897, Grant CCF-1652065, Grant CNS-1717559, and Grant ECCS-1824379. Parts of this article have been presented at IEEE ISCAS 2018 [1]; In this article, we extend our results to frequency-selective channels using OFDM. A MATLAB simulator to reproduce the results of this article is available on GitHub: https://github.com/IIP-Group/1BOX-MIMO-OFDM-simulator.

ABSTRACT The use of low-resolution data converters in the radio-frequency (RF) chains of all-digital massive multiple-input multiple-output (MIMO) basestations promises significant reductions in power consumption, hardware costs, and interconnect bandwidth. We propose a quantization-aware data-detection algorithm which mitigates the performance loss of 1-bit quantized massive MIMO orthogonal frequency-division multiplexing (OFDM) systems. Since the system performance heavily depends on the quality of channel estimates, we also develop a nonlinear 1-bit channel estimation algorithm that builds upon the proposed data detection algorithm. We show that the proposed algorithms significantly outperform linear data detectors and channel estimators in terms of bit error rate. For the proposed nonlinear data detection algorithm, we develop a very large scale integration (VLSI) architecture and present implementation results on a Xilinx Virtex-7 field programmable gate array (FPGA). Our implementation results are, to the best of our knowledge, the first for 1-bit massive MU-MIMO-OFDM systems and demonstrate comparable hardware efficiency with respect to state-of-the-art linear data detectors designed for systems with high-resolution data converters, while achieving lower bit error rate.

INDEX TERMS 1-bit analog-to-digital converter (ADC), channel estimation, data detection, FPGA implementation, massive multiple-input multiple-output (MIMO), orthogonal frequency-division multiplexing (OFDM), VLSI design.

I. INTRODUCTION

MASSIVE multi-user multiple-input multiple-output (MU-MIMO) is one of the core technologies of fifth-generation (5G) wireless systems as it promises significant improvements in spectral efficiency and link reliability, compared to traditional, small-scale MIMO [2], [3]. In massive MU-MIMO, the infrastructure basestations (BSs) are equipped with hundreds of antenna elements that simultaneously serve tens of user equipments (UEs) in the same frequency band. In all-digital massive MU-MIMO basestation architectures [4], [5], each radio-frequency (RF) chain is equipped with a pair of high-resolution (e.g., 10 bit to 12 bit) analog-to-digital converters (ADCs). The presence of hundreds of such high-quality RF chains inevitably results in high power consumption, interconnect data rates, and hardware costs, especially when deployed for the large bandwidths offered at millimeter-wave (mmWave) frequencies [6], [7].

To mitigate these issues, one can deploy low-resolution ADCs [8]–[11], which is motivated by the observation that the power consumption of ADCs scales exponentially with the number of quantization bits [12]. Another benefit of

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

deploying low-resolution ADCs is that the data rates on the fronthaul link, which connects the baseband unit and remote radio head, can be lowered significantly. In addition, the quality requirements on the RF circuitry (e.g., low-noise amplifiers, mixers, filters) can be relaxed, which enables further power and cost savings. All these benefits are attained at their greatest extent for the case of 1-bit ADCs, which can be implemented simply with 1-bit comparators and they eliminate the need for automatic gain control (AGC) circuits. This results in even more savings in RF chain power consumption and cost. However, due to the strong nonlinearity introduced by 1-bit ADCs, baseband processing tasks including data detection become more challenging in these systems. In this article, we focus on data detection and channel estimation in massive MU-MIMO systems with 1-bit ADCs, operating over frequency-selective channels. However, our results can be extended to the case of multi-bit ADCs using the general framework in [13], without significantly affecting the algorithm complexity. A detailed study of multi-bit case is left for future work.

A. RELATED PREVIOUS WORK

Linear channel estimation and data detection algorithms for 1-bit massive MU-MIMO systems, such as maximum-ratio combining (MRC) and linear minimum mean square error (L-MMSE), have been studied in [8], [14]–[16] for systems operating in frequency-flat channels. These papers demonstrate that reliable multiuser communications is possible, even for higher-order constellations [16]. Furthermore, the results in [8] show that 3-bit to 4-bit ADC resolution is sufficient to approach the achievable rates of infinite-resolution data converters.

The practically more relevant case of frequency-selective channels has been studied in [13], [17]. In [17], it has been demonstrated that linear data detection achieves acceptable performance for wideband systems with 1-bit ADCs, assuming that the channel has a sufficiently large number of taps. For massive MU-MIMO systems with orthogonal frequency division multiplexing (OFDM), it was shown in [13] that 4-bit to 6-bit ADCs are sufficient to achieve similar performance as systems with infinite-resolution data converters. While these results show that linear channel estimators and data detectors can be used in conjunction with 4-bit to 6-bit ADCs, sophisticated nonlinear channel estimation and data detection algorithms are necessary for systems that use ADCs with 3 bits or less.

To improve the performance of 1-bit massive MU-MIMO systems, an L-MMSE channel estimator based on Bussgang's decomposition [18] has been developed in [19]. Sophisticated nonlinear channel estimation and data detection algorithms have been proposed in [10], [11], [13], [20]–[23] for systems with low-resolution ADCs. The methods in [20]–[23] perform data detection using generalized approximate message passing, which enables excellent error-rate performance for Rayleigh-fading channels, but at the cost of high complexity and rather poor performance in correlated or line-of-sight propagation conditions—the less complex methods in [10], [11] are only suitable for frequency-flat systems. The channel estimators and data detectors in [13] rely on a convex-optimization procedure, which perform well under realistic propagation conditions in coarsely-quantized massive MU-MIMO-OFDM systems but at the cost of high complexity. However, to the best of our knowledge, none of the above algorithms have been implemented in hardware.

A large number of data detector hardware designs for massive MU-MIMO systems have been proposed in the past; see, e.g., [24]–[31]. All of these data detectors have been designed for BS architectures with high-resolution ADCs. Furthermore, these implementations are suitable for frequency-flat channels, or rely on OFDM or single-carrier frequency-division multiple access (SC-FDMA) to decompose frequency-selective channels into orthogonal, frequency-flat subcarriers. However, due to the severe distortion caused by 1-bit ADCs, OFDM and SC-FDMA processing does no longer result in orthogonal and frequency-flat subcarriers [13]. Hence, the use of conventional data detectors that have been designed with frequency-flat channels and high-resolution ADCs in mind, inevitably result in poor performance in BS architectures that use 1-bit ADCs.

B. CONTRIBUTIONS

We propose a new data detection algorithm and develop a corresponding VLSI design specialized for 1-bit massive MU-MIMO-OFDM systems operating over frequency-selective channels. Our contributions are summarized as follows:

- We propose a nonlinear quantization-aware data detection algorithm that solves a relaxed version of the ML detection problem in 1-bit massive MU-MIMO-OFDM systems. The proposed algorithm includes optimizations that enable an efficient VLSI implementation.
- Based on our data detection algorithm, we develop a nonlinear channel estimation algorithm that mitigates the performance loss under 1-bit quantization. We further improve the quality of the channel estimates using time-domain maximum likelihood estimator, which exploits correlation across subcarriers to denoise the channel estimates.
- We use simulations to demonstrate that the proposed channel estimation and data detection methods outperform linear algorithms for frequency-selective channels.
- We present an efficient VLSI architecture for the proposed data detection algorithm and show the first implementation results of a 1-bit massive MU-MIMO-OFDM data detector on a field programmable gate array (FPGA).

Our simulations and FPGA implementation results show that our design achieves comparable hardware efficiency but (often significantly) lower error rate compared to data detectors that have been designed for systems with high-resolution ADCs.

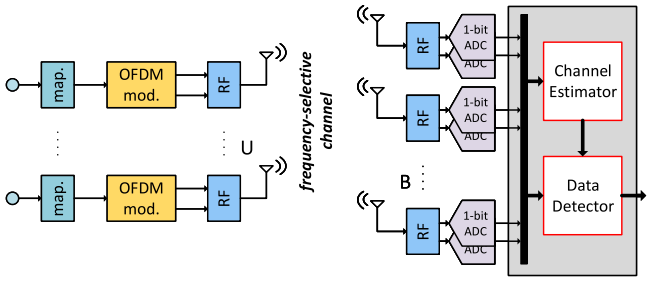


FIGURE 1. Overview of a 1-bit massive MU-MIMO-OFDM uplink system. Left: *U* user equipments (UEs) with OFDM transmission over a frequency-selective channel. Right: basestation (BS) with *B* antennas; each BS antenna quantizes the time-domain baseband signals with a pair of 1-bit ADCs prior to channel estimation and data detection.

C. NOTATION

Boldface lowercase and uppercase letters represent column vectors and matrices, respectively. For a matrix $\mathbf{A}$, the transpose and Hermitian transpose are denoted by $\mathbf{A}^{\mathrm{T}}$ and $\mathbf{A}^{\mathrm{H}}$, respectively, the kth column is $\mathbf{a}_k = [\mathbf{A}]_k$, and the entry on the mth row and nth column is $A_{m,n} = [\mathbf{A}]_{m,n}$. The ℓ_2-norm of a vector $\mathbf{a}$ and the Frobenius norm of a matrix $\mathbf{A}$ are $\|\mathbf{a}\|_2$ and $\|\mathbf{A}\|_F$, respectively. The diagonal matrix with main diagonal given by the vector $\mathbf{a}$ is $\mathbf{A} = \mathrm{diag}(\mathbf{a})$. The $M \times N$ all-zeros and $N \times N$ identity matrices are $\mathbf{0}_{M\times N}$ and $\mathbf{I}_N$, respectively. The $N \times N$ discrete Fourier transform (DFT) matrix is denoted by $\mathbf{F}$ and normalized so that $\mathbf{F}\mathbf{F}^H = \mathbf{I}_N$. For a vector $\mathbf{a}$, the kth entry is denoted by $a_k = [\mathbf{a}]_k$, and the real and imaginary parts are $\Re(\mathbf{a}) = \mathbf{a}^R$ and $\Im(\mathbf{a}) = \mathbf{a}^I$, respectively. We use $\boxdot$ to define an extended dot product that takes two $M \times N$ matrices $\mathbf{A}$ and $\mathbf{B}$ and returns an $M \times 1$ vector $\mathbf{c} = \mathbf{A} \boxdot \mathbf{B}$, whose mth element is the dot product of the mth rows of $\mathbf{A}$ and $\mathbf{B}$, i.e., $c_m = [\mathbf{A}^{\mathrm{T}}]_m^{\mathrm{T}}[\mathbf{B}^{\mathrm{T}}]_m$. We use $\odot$ to define a Hadamard product that takes two $M \times N$ matrices $\mathbf{A}$ and $\mathbf{B}$ and returns an $M \times N$ matrix $\mathbf{C} = \mathbf{A} \odot \mathbf{B}$ whose entry on the mth row and nth column is $C_{m,n} = A^R_{m,n}B^R_{m,n} + jA^I_{m,n}B^I_{m,n}$. The signum function $\mathrm{sign}(\cdot)$ operates entry-wise on vectors and for each entry x returns $+1$ if $x > 0$ and -1 otherwise. A proper complex-valued zero-mean Gaussian vector $\mathbf{a}$ with covariance matrix $\mathbf{\Sigma}$ is denoted by $\mathbf{a} \sim \mathcal{CN}(\mathbf{0}, \mathbf{\Sigma})$.

D. ARTICLE OUTLINE

The rest of this article is organized as follows. Section II introduces the system model. Section III presents our quantization-aware data detection and channel estimation algorithms for 1-bit massive MU-MIMO-OFDM systems. Section IV describes the VLSI architecture and shows FPGA implementation results of the data detector. Section V concludes this article.

II. SYSTEM MODEL

We consider the uplink of a 1-bit massive MU-MIMO-OFDM system illustrated in Figure 1, where U single-antenna UEs communicate with a BS that is equipped with $B \gg U$ antennas. We assume a block-fading scenario and communication over a frequency-selective channel using OFDM. Each OFDM symbol consists of $W = W_{\mathrm{used}} + W_{\mathrm{guard}}$ subcarriers, where W_{used} refers to the number of subcarriers used to carry data or pilot symbols, and W_{guard} refers to the number of guard subcarriers. The set of subcarriers used for data or pilot symbols is denoted by Ω_{used} and the set of subcarriers used as guard tones is denoted by Ω_{guard}.

Remark 1: In what follows, we assume perfect timing and frequency synchronization at the BS. Due to the severe distortion of 1-bit quantized received signals, timing and frequency synchronization is challenging. A recent study in [32] has shown that accurate synchronization is feasible using the conventional Schmidl-Cox algorithm for the downlink of OFDM-based systems with 1-bit DACs. To the best of our knowledge, not much is known about uplink synchronization with 1-bit ADCs, but we expect that results in [32] can be extended to our scenario. A detailed study of synchronization with 1-bit quantized signals is left for future work.

The channel is modeled in the time domain by the matrices $\mathbf{H}_t \in \mathbb{C}^{B\times U}$, $t = 1, \ldots, L$, where L is the number of taps of the channel's impulse response. The frequency-domain channel matrices $\mathbf{H}_w \in \mathbb{C}^{B\times U}$, $w = 1, \ldots, W$, are obtained from the time-domain representation via a DFT:

$$\mathbf{H}_w = \frac{1}{\sqrt{W}} \sum_{t=1}^{L} \mathbf{H}_t \exp\left(-\frac{j2\pi}{W}(t-1)(w-1-W/2)\right), \quad (1)$$

noting that $\mathbf{H}_t = \mathbf{0}_{B\times U}$, for $t = L+1, \ldots, W$. To simplify notation, we often use the frequency-domain channel matrices $\mathbf{H}_b \in \mathbb{C}^{W\times U}$, $b = 1, \ldots, B$, corresponding to each BS antenna. The wth row of $\mathbf{H}_b$ is the channel vector between the bth BS antenna and all users on the wth subcarrier for all $w = 1, \ldots, W$. Throughout the paper, channel matrices with the subscript b correspond to the $W \times U$ frequency-domain channel matrices associated with each BS antenna; channel matrices with the subscript w correspond to the $B \times U$ frequency-domain channel matrices associated with each subcarrier.

Uplink communication within each channel coherence interval is divided into two phases. In the first phase, which consists of $N_t = UT$ OFDM symbols, the UEs send pilot signals. The parameter T determines the number of training symbols per UE. In the second phase, the UEs transmit N_d data-carrying OFDM symbols. In what follows, we describe the transmission model for the duration of one OFDM symbol, which is the same for both channel training and data transmission—the only difference is the choice of frequency-domain symbols.

During the transmission of each OFDM symbol, each UE generates its own frequency-domain symbol vector $\mathbf{s}_u \in \mathbb{C}^W$. For the subcarriers reserved as guard tones, these symbols are zero, i.e., $[\mathbf{s}_u]_w = 0$ for $w \in \Omega_{\mathrm{guard}}$. For the other subcarriers $w \in \Omega_{\mathrm{used}}$, these symbols are chosen from a constellation set $\mathcal{X}$ (or pilot constellation set $\mathcal{X}_t$), i.e., $[\mathbf{s}_u]_w \in \mathcal{X}$ (or $[\mathbf{s}_u]_w \in \mathcal{X}_t$) and are normalized as $\mathbb{E}[|[\mathbf{s}_u]_w|^2] = E_s$ for

all w. Each UE then converts its frequency-domain vector $\mathbf{s}_u$ into the time domain using a W-point inverse DFT, and transmits the resulting vector after prepending a cyclic prefix (CP) of length P. We assume perfect synchronization and a CP length of $P \geq L-1$, which is sufficient to avoid inter-symbol interference.

At the BS-side, each antenna receives a noisy superposition of the UEs' signals. To simplify notation, we will use the $W \times U$ matrix $\mathbf{S}$ whose uth column contains the frequency-domain symbols of the uth UE. Let $\mathbf{y}_b \in \mathbb{C}^W$ denote the (unquantized) signal vector received at the bth BS antenna after removal of the cyclic prefix. This vector can be modeled as

$$\mathbf{y}_b = \mathbf{F}^{\mathrm{H}}\mathbf{z}_b + \mathbf{n}_b, \tag{2}$$

where $\mathbf{n}_b \sim \mathcal{CN}(\mathbf{0}_W, N_0\mathbf{I}_W)$ is the thermal receive noise at the bth BS antenna with variance N_0 per complex entry, and $\mathbf{z}_b \in \mathbb{C}^W$ is the vector of noiseless frequency-domain signals associated with the bth BS antenna given by

$$\mathbf{z}_b = \mathbf{H}_b \boxdot \mathbf{S}, \quad b = 1, \ldots, B, \tag{3}$$

where $\boxdot$ is the extended dot product defined in Section I-C.

In what follows, we assume that the in-phase and quadrature baseband signals at the output of each BS RF chain are quantized by a pair of zero-threshold 1-bit ADCs. For a complex-valued scalar z, we model this quantization operation as $r = Q(z) = \mathrm{sign}(z^R) + j\,\mathrm{sign}(z^I)$, which is applied element-wise to vectors and matrices. The W-dimensional vector of the 1-bit quantized observations at the bth BS antenna for the duration of one OFDM symbol is thus given by

$$\mathbf{r}_b = Q(\mathbf{y}_b) = Q\big(\mathbf{F}^{\mathrm{H}}\mathbf{z}_b + \mathbf{n}_b\big). \tag{4}$$

Assuming $\mathbb{E}[\|\mathbf{H}_b\|_F] = WUE_h$ for $b = 1, \ldots, B$, the average receive signal-to-noise ratio (SNR) at each BS antenna prior to quantization is given by $\rho = W_{\text{used}}UE_sE_h/(WN_0)$.

III. 1-BIT DATA DETECTION AND CHANNEL ESTIMATION

We now present our data detection and channel estimation algorithms for 1-bit massive MU-MIMO-OFDM systems, and we demonstrate their effectiveness via simulation results.

A. QUANTIZATION-AWARE DATA DETECTION WITH BOX CONSTRAINTS

In order to derive the quantization-aware data detection algorithm, we first formulate the ML problem and then relax its constraints to arrive at a problem that can be solved efficiently. We start with the likelihood of the output of a 1-bit quantizer $r = Q(\mu + n) = \mathrm{sign}(\mu + n)$, given the noiseless input μ and assuming that n is circularly-symmetric complex Gaussian noise with variance $N_0 = \sigma^2$. For this model, the likelihood function is given by the following expression [13], [33], [34]:

$$p(r|\mu) = p\Big(r^R|\mu^R\Big)p\Big(r^I|\mu^I\Big)$$

$$= \Phi\left(\frac{\sqrt{2}}{\sigma}r^R\mu^R\right)\Phi\left(\frac{\sqrt{2}}{\sigma}r^I\mu^I\right). \tag{5}$$

Here, $\Phi(t) = \int_{-\infty}^{t}\frac{1}{\sqrt{2\pi}}\exp(-u^2/2)\mathrm{d}u$, is the cumulative distribution function of a standard normal random variable. Now let us assume that $\mathbf{r} = Q(\boldsymbol{\mu} + \mathbf{n})$, where $\mathbf{r}$, $\boldsymbol{\mu}$, and $\mathbf{n}$ are N-dimensional vectors, and the noise vector is distributed according to $\mathbf{n} \sim \mathcal{CN}(\mathbf{0}, N_0\mathbf{I})$. The likelihood function of the vector $\mathbf{r}$ is given by $p(\mathbf{r}|\boldsymbol{\mu}) = \prod_{n=1}^{N} p(r_n|\mu_n)$. Hence, the ML data detection problem corresponds to [13]

$$\begin{aligned}\hat{\mathbf{S}} = \underset{\tilde{\mathbf{S}}\in\mathbb{C}^{W\times U}}{\arg\max} \; & \prod_{b=1}^{B} p\Big(\mathbf{r}_b|\mathbf{F}^{\mathrm{H}}(\mathbf{H}_b \boxdot \tilde{\mathbf{S}})\Big) \\ \text{subject to} \; & \Big[\tilde{\mathbf{S}}^{\mathrm{T}}\Big]_w \in \mathcal{X}^U, \quad w \in \Omega_{\text{used}} \\ & \Big[\tilde{\mathbf{S}}^{\mathrm{T}}\Big]_w = \mathbf{0}_{U\times 1}, \quad w \in \Omega_{\text{guard}},\end{aligned} \tag{6}$$

where $\mathbf{r}_b$ is the vector of 1-bit observations at the bth BS antenna as defined in (4). Note that this problem is NP-hard, as the ML data detection problem for the infinite-resolution case, and an exhaustive search would require one to evaluate the objective function $|\mathcal{X}|^{WU}$ times. In order to overcome this prohibitive complexity, we relax the discrete constellation constraints. The same idea has been used in the special case of frequency-flat channels in [1], [10], [11], [24], and in a more general framework with multi-bit ADCs in [13].

In order to arrive at an efficient method, we relax the discrete constellation to its bounding box (convex hull), which is, for quadrature-amplitude modulation (QAM), given by

$$\mathcal{B} = \Big\{x \in \mathbb{C} \mid \max\Big\{\Big|x^R\Big|, \Big|x^I\Big|\Big\} \leq S_{\mathcal{X}}\Big\}, \tag{7}$$

where $S_{\mathcal{X}} = \max_{s\in\mathcal{X}}\{|s^R|, |s^I|\}$. Concretely, we replace the constraints $[\tilde{\mathbf{S}}^{\mathrm{T}}]_w \in \mathcal{X}^U, w \in \Omega_{\text{used}}$ in (6) with $[\tilde{\mathbf{S}}^{\mathrm{T}}]_w \in \mathcal{B}^U, w \in \Omega_{\text{used}}$. The resulting optimization problem is convex and can be formulated in equivalent form as follows:

$$\begin{aligned}\hat{\mathbf{S}}_{\text{1BOX}} = \underset{\tilde{\mathbf{S}}\in\mathbb{C}^{W\times U}}{\arg\min} \; & f(\tilde{\mathbf{S}}) + \sum_{w\in\Omega_{\text{used}}} \mathcal{I}\Big(\Big[\tilde{\mathbf{S}}^{\mathrm{T}}\Big]_w \in \mathcal{B}^U\Big) \\ & + \sum_{w\in\Omega_{\text{guard}}} \mathcal{I}\Big(\Big[\tilde{\mathbf{S}}^{\mathrm{T}}\Big]_w = \mathbf{0}_{U\times 1}\Big).\end{aligned} \tag{8}$$

Here, $f(\tilde{\mathbf{S}}) = -\sum_{b=1}^{B}\log p(\mathbf{r}_b|\mathbf{F}^{\mathrm{H}}(\mathbf{H}_b \boxdot \tilde{\mathbf{S}}))$ is the negative logarithm of the likelihood function in (6), and $\mathcal{I}(\cdot)$ is the indicator function which outputs zero if its input is satisfied and infinity otherwise. Since magnitude is lost completely in 1-bit measurements, we re-scale the output $\hat{\mathbf{S}}_{\text{1BOX}}$ according to

$$\hat{\mathbf{S}}^{\text{norm}} = \frac{E_s\sqrt{UW_{\text{used}}}}{\|\hat{\mathbf{S}}_{\text{1BOX}}\|_F}\hat{\mathbf{S}}_{\text{1BOX}}, \tag{9}$$

followed by mapping the normalized output $\hat{\mathbf{S}}^{\text{norm}}$ to the nearest constellation point in the discrete constellation set $\mathcal{X}$.

The optimization problem in (8) can be solved using the forward-backward splitting (FBS) framework [35], [36]. We use FBS to design an iterative algorithm that we call

1-Bit OFDM boX (1BOX) detector, which is summarized in Algorithm 1. In each iteration of 1BOX, lines 4 to 11 calculate the (scaled) negative gradient of the objective function $f(\tilde{\mathbf{S}})$ of (8), and line 12 increments the estimates of the current iteration in the direction of the negative gradient by a step-size of κ and projects the result onto the box constraint set. Calculating the gradient of $f(\tilde{\mathbf{S}})$ requires the inverse Mills ratio for a normal random variable defined as

$$\omega(x) = \frac{\exp\left(-x^2/2\right)}{\sqrt{2\pi}\,\Phi(x)}, \tag{10}$$

and its complex version $\omega_c(z) = \omega(z^R) + j\omega(z^I)$, which is applied entry-wise to vectors. With this definition, the negative gradient of the objective function $f(\tilde{\mathbf{S}})$, with respect to the estimated matrix $\tilde{\mathbf{S}}$, is a $W \times U$ matrix whose wth row is

$$\left[(\nabla f)^{\mathrm{T}}\right]_w = \frac{\sqrt{2}}{2\sigma}\mathbf{H}_w^{\mathrm{H}}[\mathbf{V}]_w. \tag{11}$$

Here, $\mathbf{V}$ is a $B \times W$ matrix whose bth row is given by

$$\left[\mathbf{V}^{\mathrm{T}}\right]_b = \mathbf{F}\left(\mathbf{r}_b \odot \omega_c\left(\frac{\sqrt{2}}{\sigma}\mathbf{r}_b \odot \mathbf{F}^{\mathrm{H}}\mathbf{z}_b\right)\right), \tag{12}$$

and $\mathbf{z}_b$ is defined in (3). The quantity $\mathbf{G}$ on line 10 is a scaled version of the gradient, where we omit the prefactor $\sqrt{2}/(2\sigma)$, which is absorbed into the step-size κ. The function $\mathrm{proj}_c(\cdot)$ on line 12 of Algorithm 1, implements a projection onto the box constraints in (7). This function operates element-wise on the entries of its input and simply performs the operation

$$\mathrm{proj}_c(x, \mathcal{B}) = \mathrm{proj}\left(x^R, \mathcal{B}\right) + j\,\mathrm{proj}\left(x^I, \mathcal{B}\right), \tag{13}$$

where proj(.) is given by $\mathrm{proj}(x, \mathcal{B}) = \mathrm{sign}(x)\min(|x|, S_{\mathcal{X}})$.

Remark 2: We note that for constant-modulus modulation schemes, such as 8-PSK, one could achieve better performance by using a circle-like polytope as the constraint set rather than a rectangular box as in (7). However, to develop simpler VLSI implementations, we have used box constraints regardless of the modulation scheme. Projection onto more complex circle-like polytopes significantly increases hardware complexity [37].

B. OPTIMIZATION FOR HARDWARE IMPLEMENTATION

The 1BOX algorithm summarized in Algorithm 1 has two drawbacks from a hardware implementation perspective. We next introduce two solutions that address these issues.

1) STABLE GRADIENT CALCULATION

The first issue arises from the inverse Mills ratio $\omega(x)$ in (10), which is numerically unstable for negative inputs of large absolute value. Since both the nominator and denominator of (10) asymptotically approach zero for large negative values of x, such input values result in zero-over-zero division with finite-precision arithmetic. In order to circumvent this issue, one can use *l'Hôpital*'s rule to find the negative infinity limit of $\omega(x)$, which is equal to $-x$. For sufficiently large positive values of x, $\omega(x)$ produces outputs very close to zero. As illustrated in Fig. 2, these properties of $\omega(x)$ can be exploited to simplify a hardware implementation using the approximation:

$$\tilde{\omega}(x) = \begin{cases} 0, & x \geq t_p \\ \omega(x), & t_n < x < t_p \\ -x, & x \leq t_n. \end{cases} \tag{14}$$

The thresholds $t_p = 4$ and $t_n = -4$ have been selected based on simulations to minimize the performance degradation due to the approximation (14). This approximation enables efficient hardware designs with a small look-up-table (LUT) that stores only the function values between t_n and t_p.

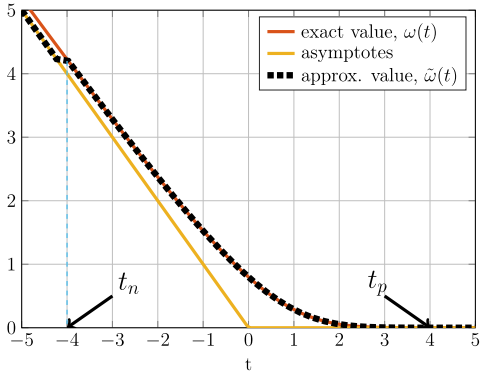


FIGURE 2. Mills ratio $\omega(x)$ defined in (10) along with its asymptotes and approximated values used in our hardware design.

Algorithm 1 The 1BOX Algorithm to Solve (8) via FBS

1: **Inputs:** σ, $\{\mathbf{H}_w\}_{w=1}^{W}$, $\{\mathbf{r}_b\}_{b=1}^{B}$, κ, and K
2: **Initialize:** $\mathbf{S}^{(0)} = \mathbf{0}_{U\times W}$
3: **for** $k = 1, \ldots, K$ **do**
4: **for** $b = 1, \ldots, B$ **do**
5: $\mathbf{z}_b^{(k)} = \mathbf{H}_b \boxdot \mathbf{S}^{(k-1)}$
6: $\boldsymbol{\alpha}_b^{(k)} = \frac{\sqrt{2}}{\sigma}\mathbf{r}_b \odot (\mathbf{F}^{\mathrm{H}}\mathbf{z}_b^{(k)})$
7: $[\mathbf{V}^{\mathrm{T}}]_b = \mathbf{F}(\mathbf{r}_b \odot \omega_c(\boldsymbol{\alpha}_b^{(k)}))$
8: **end for**
9: **for** $w = 1, \ldots, W$ **do**
10: $[\mathbf{G}^{\mathrm{T}}]_w = \mathbf{H}_w^{\mathrm{H}}[\mathbf{V}]_w$
11: **end for**
12: $\mathbf{S}^{(k)} = \mathrm{proj}_c(\mathbf{S}^{(k-1)} + \kappa\mathbf{G}, \mathcal{B})$
13: **end for**
14: **return:** $\hat{\mathbf{S}}_{1\mathrm{BOX}} = \mathbf{S}^{(K)}$

2) LIMITING THE NOISE STANDARD DEVIATION

The second problem arises from convergence issues at high SNR with fixed-point arithmetic. Adaptive step-size rules are able to deal with such convergence issues; see, e.g., [11], [36]. Such rules, however, entail excessively high complexity as they require a search over suitable step sizes in every iteration that includes repeatedly evaluating the objective function (which per se requires high complexity). Instead, we propose a simple modification to Algorithm 1

that simplifies our hardware design in Section IV. Due to the large dynamic range of the objective function at high SNR, small step-sizes are required to ensure convergence. From a hardware perspective, it is desirable to have a fixed step size that can be implemented with simple arithmetic shift operations. Since at high SNR, the role of step size is to control the dynamic range of the entries of $\mathbf{G}$ to be added to $\mathbf{S}^{(k-1)}$ on line 12 of Algorithm 1, it is equivalent to limiting the dynamic range of the entries of $\boldsymbol{\alpha}_b$ on line 6. This effect is accomplished by thresholding the value of the noise standard deviation σ, i.e., we replace the value of σ with σ' whenever $\sigma < \sigma'$. Our simulations have shown that suitable values for σ' correspond to an SNR between 10 dB and 15 dB. In Section IV-B3, we will denote the thresholded noise standard deviation by $\tilde{\sigma}$, i.e., we set $\tilde{\sigma} = \sigma'$ if $\sigma < \sigma'$ and $\tilde{\sigma} = \sigma$ otherwise.

C. LINEAR-QUANTIZED DATA DETECTION

Since linear data detection algorithms have been studied extensively in the literature for both full-resolution and low-resolution converters [8], [17], [38], we use them as a benchmark to evaluate the performance of the 1BOX data detector. Zero-forcing and L-MMSE detectors allow for efficient hardware implementations [26], [31] and achieve similar error-rate performance in massive MU-MIMO systems where $B \gg U$. Hence, we consider ZF detection (referred to as ZF-DET), which first converts the 1-bit received data at each BS antenna $\mathbf{r}_b$, $b = 1, \ldots, B$, into the frequency domain using a DFT: $\tilde{\mathbf{r}}_b = \mathbf{F}\mathbf{r}_b$. Let $\tilde{\mathbf{R}}$ be a $W \times B$ matrix whose bth column is $\tilde{\mathbf{r}}_b$. Then, ZF is applied on each active subcarrier $w \in \Omega_{\text{used}}$ as follows:

$$\left[\hat{\mathbf{S}}^{\mathrm{T}}\right]_w = \left(\mathbf{H}_w^{\mathrm{H}}\mathbf{H}_w\right)^{-1}\mathbf{H}_w^{\mathrm{H}}\left[\tilde{\mathbf{R}}^{\mathrm{T}}\right]_w. \tag{15}$$

The outputs are then normalized as in (9) and mapped to the nearest points in the constellation $\mathcal{X}$.

D. 1-BIT CHANNEL ESTIMATION

The performance of coherent data detectors depends heavily on the accuracy of the available channel estimates—this is even more critical in 1-bit quantized systems which suffer from nonlinear distortions. We next develop a method that relies on the same tools of 1BOX to calculate improved channel estimates. As discussed in Section II, during the channel training phase, all UEs transmit $N_t = UT$ pilot OFDM symbols, concurrently. This results in UTW training symbols in total, as each OFDM symbol contains W frequency-domain symbols. The frequency-domain pilot symbols of all UEs transmitted during the nth pilot OFDM symbol are gathered in the matrix $\mathbf{T}_n \in \mathcal{X}_t^{W\times U}$, where $\mathcal{X}_t$ is the set of pilot symbols, augmented with zero to take into account the zero-symbol used for guard subcarriers. To simplify exposition, we additionally introduce the per-subcarrier matrix $\mathbf{T}_w \in \mathcal{X}_t^{N_t\times U}$ whose uth column contains the frequency-domain pilot symbols of the UE u transmitted over N_t pilot OFDM symbols on the wth subcarrier. Let $\mathbf{Y}_b$ be the $W \times N_t$ matrix associated with the bth BS antenna, whose nth column contains the unquantized time-domain samples (after removing the cyclic prefix), received during the nth training OFDM symbol. Then, the 1-bit quantized observations at the bth BS antenna are given by

$$\mathbf{R}_b = Q(\mathbf{Y}_b) = Q\big(\mathbf{F}^{\mathrm{H}}\mathbf{Z}_b + \mathbf{N}_b\big), \tag{16}$$

where $\mathbf{Z}_b$ is a $W \times N_t$ matrix, whose nth column is given by $[\mathbf{Z}_b]_n = \mathbf{H}_b \boxdot \mathbf{T}_n$, and the entries of $\mathbf{N}_b$ are i.i.d. circularly-symmetric complex Gaussian with variance N_0. Our objective is to obtain channel estimates $\{\hat{\mathbf{H}}_w\}_{w=1}^W$ based on the 1-bit observations $\{\mathbf{R}_b\}_{b=1}^B$ and the known pilot matrices $\{\mathbf{T}_n\}_{n=1}^{N_t}$.

1) LINEAR-QUANTIZED CHANNEL ESTIMATION

A naïve way for obtaining channel estimates from 1-bit measurements is to ignore quantization altogether and perform W independent channel estimation tasks relying on the orthogonality of OFDM together with linear channel estimators, such as ZF or L-MMSE, on a per-subcarrier basis [13], [17]. Since ZF and L-MMSE channel estimation provides similar performance in massive MU-MIMO, we focus on ZF channel estimation. Similar to ZF-DET, ZF channel estimation (called ZF-CHEST) first converts the 1-bit measurements at each BS antenna $\mathbf{R}_b$ into the frequency domain according to $\tilde{\mathbf{R}}_b = \mathbf{F}\mathbf{R}_b$, $b = 1, \ldots, B$. Then, the channel estimates for each BS antenna b and for each active subcarrier $w \in \Omega_{\text{used}}$ are obtained as follows:

$$\left[\left(\hat{\mathbf{H}}_w\right)^{\mathrm{T}}\right]_b = \left(\mathbf{T}_w^{\mathrm{H}}\mathbf{T}_w\right)^{-1}\mathbf{T}_w^{\mathrm{H}}\left[\tilde{\mathbf{R}}_b^{\mathrm{T}}\right]_w. \tag{17}$$

We note that a Bussgang-based L-MMSE (BL-MMSE) channel estimator has been proposed in [19] that achieves superior performance than the conventional L-MMSE channel estimator by taking into account the nonlinearities caused by 1-bit ADCs. A similar approach can be used to design BL-MMSE-based data detectors [39], [40]. Unfortunately, the complexity of these methods is prohibitive in OFDM systems, as they require the inversion of a $UW \times UW$ matrix, caused by the fact that 1-bit quantization destroys orthogonality that enables one to decouple the estimation problem into independent problems for each subcarrier. Furthermore, the matrix to be inverted applies a nonlinear arcsine function to each entry, which prevents efficient iterative methods to find the inverse.

2) ML-BASED CHANNEL ESTIMATION

Another approach to obtain improved channel estimates $\hat{\mathbf{H}}_b$, $b = 1, \ldots, B$, is to directly solve the ML channel estimation problem

$$\hat{\mathbf{H}}_b = \underset{\tilde{\mathbf{H}}_b \in \mathbb{C}^{W\times U}}{\arg\max} \prod_{n=1}^{N_t} p\Big([\mathbf{R}_b]_n | \mathbf{F}^{\mathrm{H}}\Big(\mathbf{T}_n \boxdot \tilde{\mathbf{H}}_b\Big)\Big), \tag{18}$$

which is derived analogously to (6). The idea of ML-based channel estimation with 1-bit quantized measurements for

frequency-flat channels has been put forward in [10]. We emphasize, however, that the method in [10] is not directly applicable to OFDM systems, and the general ML problem needs to be derived for such systems, as done in (18). The problem in (18) is convex and we propose to solve it using normalized gradient descent (NGD), which we call NGD-CHEST. The algorithm resembles that of the 1BOX data detector in (6), except that (i) there are no constraints on the channel matrices (and hence no projection operation required), (ii) we assume that the pilot matrices $\mathbf{T}_n$, $n = 1, \ldots, N_t$, are known, (iii) we initialize the algorithm with the ZF-CHEST estimates to improve convergence. Due to the structural similarity between 1BOX and NGD-CHEST, it would be possible, with only a few modifications, to use the same VLSI architecture proposed for 1BOX in Section IV to also carry out the NGD-CHEST algorithm. However, due to the short time available for channel training in each channel coherence interval, it would not be practical to carry out both NGD-CHEST and 1BOX data detection with the same hardware instance—separate instances should be used for each task. Next, we describe channel denoising and normalization techniques that are applied to NGD-CHEST and ZF-CHEST outputs to improve the channel estimates.

3) CHANNEL DENOISING AND NORMALIZATION

In most practical systems, the number L of channel taps in the time domain is less than the number of OFDM subcarriers—this implies that adjacent subcarriers are correlated. One can exploit this property to denoise the channel estimates by means of a time-domain maximum likelihood estimator (TDMLE) [41]. For the frequency-domain channel $\hat{\mathbf{h}}_{b,u} \in \mathbb{C}^{W_{\text{used}} \times 1}$ between the bth BS antenna and uth user, a denoised channel estimate can be obtained as follows:

$$\hat{\mathbf{h}}_{b,u}^{\text{denoised}} = \mathbf{F}_{\Omega_{\text{used}}} \left(\mathbf{F}_{\Omega_{\text{used}}}^{\text{H}} \mathbf{F}_{\Omega_{\text{used}}} \right)^{-1} \mathbf{F}_{\Omega_{\text{used}}}^{\text{H}} \hat{\mathbf{h}}_{b,u}. \quad (19)$$

Here, $\mathbf{F}_{\Omega_{\text{used}}}$ is a $W_{\text{used}} \times L$ matrix, constructed from a W-point DFT matrix by taking its first L columns and the rows indexed by Ω_{used}. The resulting denoised channel estimates are collected in the matrices $\hat{\mathbf{H}}_b^{\text{denoised}}$, $b = 1, \ldots, B$.

Since for 1-bit quantizers amplitude information is lost completely, we apply a re-scaling procedure analogous to (9). More concretely, we assume that the BS is able to acquire an estimate of the average channel gain at each BS antenna $\gamma_b = \mathbb{E}[\|\mathbf{H}_b\|_F]$.[1] Due to the spatial proximity of BS antennas, we assume that $\gamma_b = \gamma$ for all $b = 1, \ldots, B$. In order to minimize the mean squared error (MSE) between the exact channel matrices $\{\mathbf{H}_b\}_{b=1}^B$ and their estimates $\{\hat{\mathbf{H}}_b\}_{b=1}^B$, the BS re-scales the channel estimates as follows:

$$\hat{\mathbf{H}}_b^{\text{norm}} = \frac{\gamma}{\left\| \hat{\mathbf{H}}_b^{\text{denoised}} \right\|_F} \hat{\mathbf{H}}_b^{\text{denoised}}, \quad b = 1, \ldots, B. \quad (20)$$

1. Such information can, for example, be acquired by using one or multiple higher-resolution ADCs during a training phase.

E. COMPLEXITY ANALYSIS

In this section, we provide analytic complexity expressions for the 1BOX and NGD-CHEST algorithms, measured in terms of the number of real-valued multiplications. Each iteration of 1BOX, as shown in Algorithm 1, consists of two for-loops with B and W iterations. Line 5 of Algorithm 1 involves W inner products between U-entry vectors. By assuming that each complex-valued multiplication requires four real-valued multiplications, this line requires $4UW$ real-valued multiplications. The computations on line 6 and 7 involve DFT operations on W-entry vectors that can be implemented using fast Fourier transforms (FFT). Each W-point FFT requires $2W \log_2 W$ real-valued multiplications, assuming a radix-2 implementation. The complexity of calculating ω_c for a scalar input using look-up-tables can be approximated by one real-valued multiplication. The Hadamard products with the vectors $\mathbf{r}_b$ on lines 6 and 7 do not require actual multiplications and can be implemented efficiently with conditional negations as described in Section IV-B3. Line 10 consists of the product of a $U \times B$ matrix by a $B \times 1$ vector, which requires $4UB$ real-valued multiplications. Multiplications with the real-valued constant κ of all entries of $\mathbf{G}$ on line 12, requires $2UW$ real-valued multiplications. In total, 1BOX involves $8BUW + 4BW \log_2 W + BW + 2UW$ real-valued multiplications per algorithm iteration.

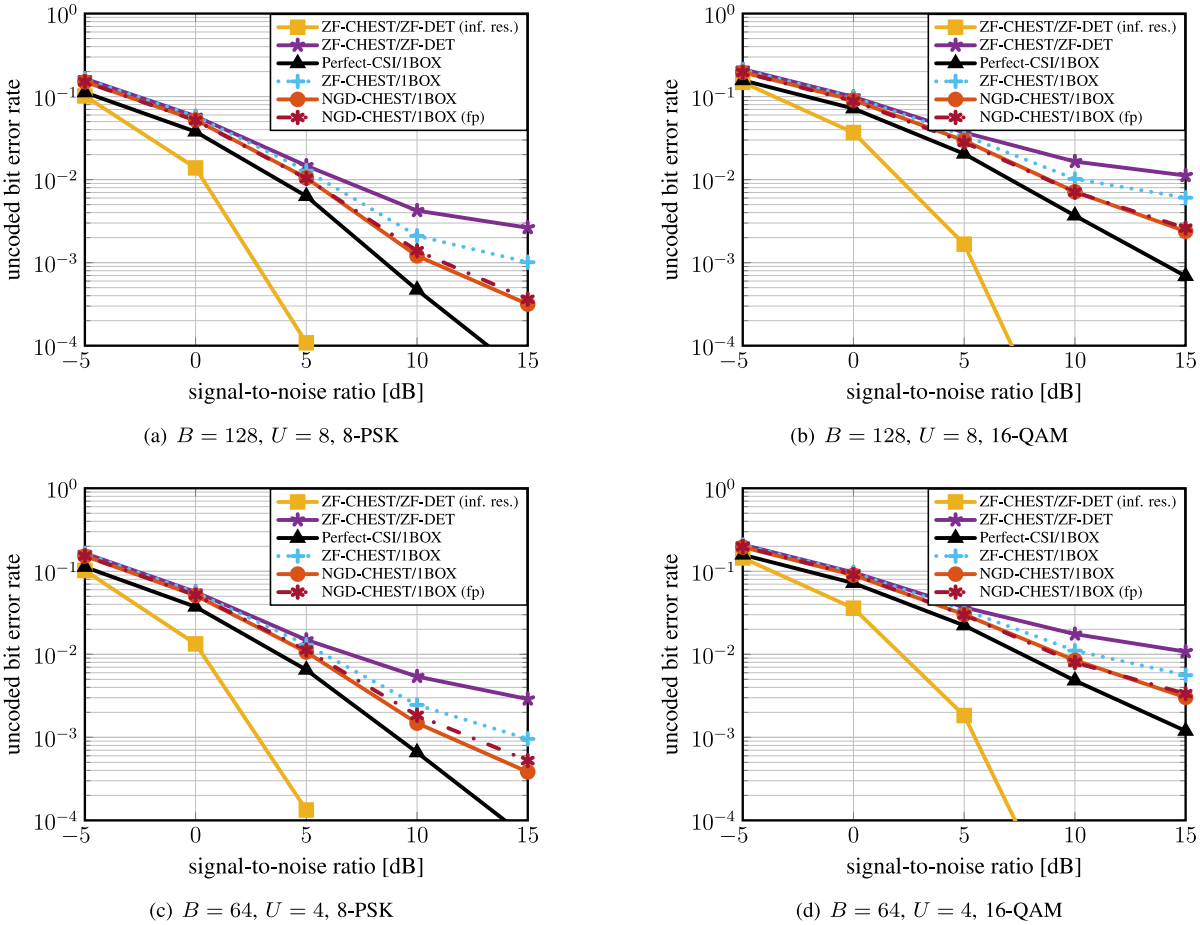
To obtain an estimate of each of the $W \times U$ channel matrices $\mathbf{H}_b$, $b = 1, 2, \ldots, B$, we need to run NGD-CHEST, which is similar to 1BOX, with the exception that the dimension B is replaced by N_t. Therefore, the overall complexity of obtaining estimates for B channel matrices is given by $BK(8N_tUW + 4N_tW \log_2 W + N_tW + 2UW)$ for K algorithm iterations.

F. SIMULATION RESULTS

To demonstrate the effectiveness of NGD-CHEST and the 1BOX data detector, we now present simulation results and a comparison with linear channel estimators and data detectors designed to operate with high-resolution ADCs.

1) SIMULATION SETTINGS

We consider a 1-bit massive MU-MIMO-OFDM system with $W = 128$ subcarriers, whose middle part of $W_{\text{used}} = 100$ subcarriers are used for data and pilots, and the remaining $W_{\text{guard}} = 28$ subcarriers at the two sides are left unused. During the channel estimation phase, all UEs simultaneously transmit pilots. Since non-sparse pilot matrices perform better than diagonal training matrices, a phenomenon that has been observed in [13], we set the frequency-domain pilot matrices $\mathbf{T}_n$, $n = 1, \ldots, N_t$, to contain random QPSK symbols that are known to the BS. In our simulations, we use $N_t = UT$ training OFDM symbols with $T = 2$. For NGD-CHEST, we set the number of algorithm iterations to $K = 5$ with a fixed step-size of $1/16$. For both ZF-CHEST and NGD-CHEST, we use TDMLE channel denoising and normalization techniques outlined in Section III-D3, to improve



(a) $B = 128$, $U = 8$, 8-PSK

(b) $B = 128$, $U = 8$, 16-QAM

(c) $B = 64$, $U = 4$, 8-PSK

(d) $B = 64$, $U = 4$, 16-QAM

FIGURE 3. Uncoded bit error rate (BER) of the proposed 1BOX detector with perfect-CSI and channel estimates obtained from ZF-CHEST or NGD-CHEST. As a reference, the BER performance of ZF detector with ZF-CHEST, both with 1-bit and infinite resolution ADCs is also included.

the channel estimates. During the data transmission phase, the UEs generate frequency-domain symbols from either 8-PSK or 16-QAM constellations. For the 1BOX detector, we use $K = 3$ iterations and set the step-size to $\kappa = \sqrt{2}/64$ for the floating-point experiments; we set $\kappa = 1/32$ for our fixed-point results as we absorb the $\sqrt{2}$ factor into the scaling schedule of the FFT hardware design. In addition, the choice of $\kappa = 1/32$ simplifies fixed point design as it can be implemented with trivial arithmetic right shifts.

2) ERROR-RATE PERFORMANCE

Figure 3 shows uncoded bit error rate (BER) results for (i) a systems with $B = 128$ BS antennas and $U = 8$ UEs and (ii) a systems with $B = 64$ BS antennas and $U = 4$ UEs. For both systems, we use Gray-coded 8-PSK and 16-QAM. Each plot in Figure 3, contains six curves: (i) ZF-CHEST followed by ZF-DET, with infinite resolution ADCs for both channel estimation and data detection (used as a reference), (ii) 1-bit ZF-CHEST followed by 1-bit ZF-DET, (iii) 1BOX detector with perfect CSI, (iv) 1-bit ZF-CHEST followed by 1BOX detection, (v) NGD-CHEST followed by 1BOX detection and (iv) NGD-CHEST with the fixed-point version of 1BOX detection (denoted by "(fp)" on the figure legends) which uses the fixed-point implementation parameters detailed in Section IV-D. As we can see from Figure 3, the 1BOX detector combined with NGD-CHEST achieves significantly better error-rate performance compared to linear quantized uplink processing, i.e., ZF-DET with ZF-CHEST. The SNR gap at $BER = 10^{-2}$ ranges from 1.5 dB for the 128×8 setup with 8-PSK signaling to 8 dB in the 128×8 setup with 16-QAM signaling. Additionally, we see that obtaining channel estimates from NGD-CHEST results in significantly better error-rate performance than those from ZF-CHEST.

In Figure 3, we observe that the performance of the fixed-point version of 1BOX that corresponds to the hardware implementation detailed in Section IV, closely match those of floating-point performance. Finally, we observe that the proposed NGD-CHEST and 1BOX algorithms, need higher SNRs to achieve the same BER as that of ZF-CHEST and ZF-DET for high-resolution systems. This indicates the existence of a trade-off between the error-rate performance and the reduction in cost and power consumption of basestation RF chains, when using 1-bit ADCs. A detailed study of this

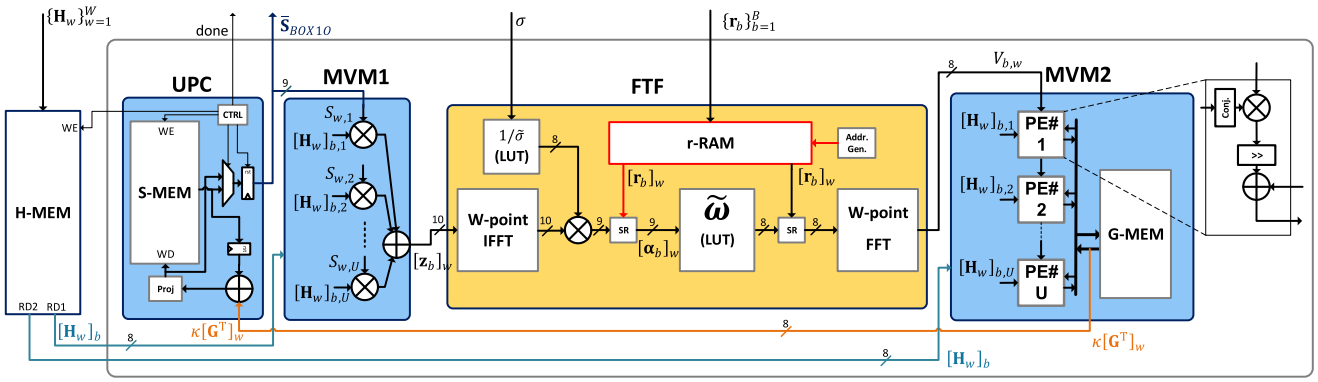


FIGURE 4. VLSI architecture of the 1BOX data detection algorithm for 1-bit massive MU-MIMO-OFDM systems.

trade-off and an assessment of the overall system cost and power consumption as a function of ADC resolution is left for future work.

Remark 3: A key limitation of data detection with 1-bit ADCs is that supporting higher-order modulation schemes (such as 64-QAM or higher) is challenging [13]. While the proposed algorithm shows acceptable performance with 8-PSK and 16-QAM, it does not perform well for higher-order modulations schemes, especially for SNR values typically encountered in mmWave systems. Therefore, 1-bit data detection is suitable for systems that operate at lower per-user data rates in exchange for reduced RF chain cost and power consumption.

IV. ARCHITECTURE AND FPGA IMPLEMENTATION

We now present a VLSI architecture for the 1BOX data detection algorithm and show reference FPGA implementation results. We then provide a comparison with existing linear data detectors that have been developed for infinite-resolution massive MU-MIMO systems. To the best of our knowledge, this is the first data detector implementation for 1-bit massive MU-MIMO systems reported in the open literature.

A. ARCHITECTURE OVERVIEW AND OPERATION PRINCIPLES

The proposed VLSI architecture is shown in Figure 4 and consists of the following five modules: (i) UPC (short for update, project, and control), (ii) MVM1 (short for matrix-vector multiplication unit 1), (iii) FTF (short for frequency-time-frequency), (iv) MVM2 (short for matrix-vector multiplication unit 2), and (v) H-MEM (short for **H**-memory). The UPC module carries out the operations on line 12 of Algorithm 1, as well as preparing the control signals; the MVM1 module is responsible for the matrix-vector multiplication on line 5; the FTF module performs the operations on lines 6 and 7; the MVM2 module is responsible for the matrix-vector multiplication on line 10; and the H-MEM module stores the frequency-domain channel matrices $\{\hat{\mathbf{H}}_w\}_{w=1}^{W}$. The signal names in Figure 4 correspond to the variables in Algorithm 1, e.g., $[\mathbf{H}_w]_b$ represents the channel vector of the bth BS antenna over the wth subcarrier.

The proposed architecture has been optimized in terms of hardware efficiency, measured in throughput per FPGA resources. With this optimization strategy, our goal is to minimize the resource consumption needed to achieve a specific throughput. Therefore, even though each instance of the proposed architecture may achieve relatively low throughput, it is possible to scale up the throughput by replication, i.e., by instantiating N parallel designs that process different sets of receive signals. This approach enables us to increase the throughput by a factor of N while maintaining the same hardware efficiency.

In addition, the proposed architecture operates in streaming fashion, which reduces the overhead of control and data buffering. As illustrated in Figure 5, the architecture processes the data of the bth BS antenna over W consecutive clock cycles—one clock cycle for each of the W subcarriers. For example, the elements of the $W \times 1$ vector $\mathbf{z}_b^{(k)}$ on line 5 of Algorithm 1, are produced in MVM1 sequentially over W consecutive clock cycles. Consequently, it takes BW clock cycles to compute all vectors $\mathbf{z}_b^{(k)}$, for $b = 1, 2, \ldots, B$, in the kth iteration of 1BOX. Therefore, each algorithm iteration requires $D = BW + L_{\text{UPC}} + L_{\text{MVM1}} + L_{\text{FTF}} + L_{\text{MVM2}}$ clock cycles, where L_{X} denotes the latency of module "X," caused by pipelining. Consequently, our data detector architecture achieves a sustained throughput of

$$\Theta = \frac{|\mathcal{X}| U W_{\text{used}} f}{KD} \quad \text{[Mb/s]}, \tag{21}$$

where $|\mathcal{X}|$ is cardinality of the constellation set and f is the circuit's clock frequency in MHz. A detailed discussion of the timing schedule is provided in Section IV-C.

B. ARCHITECTURE DETAILS

The operating principles and implementation details of the five modules are detailed next.

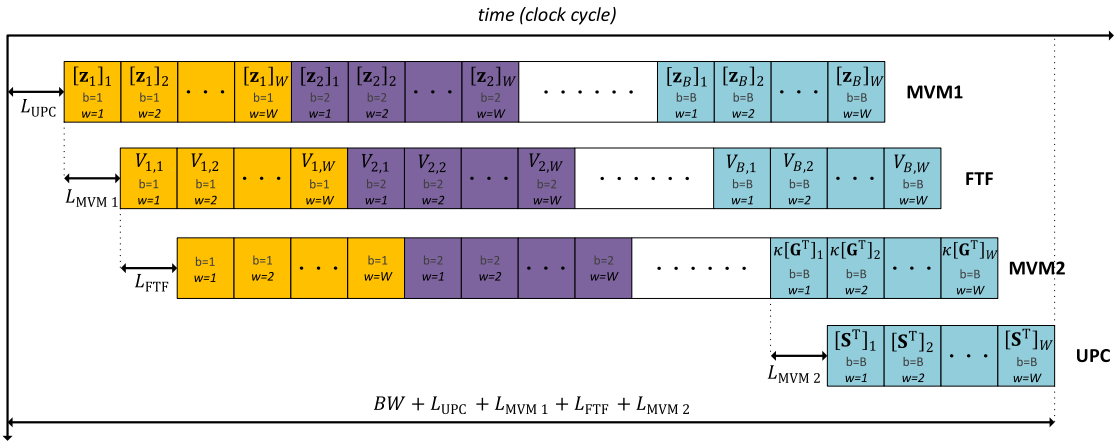


FIGURE 5. Sequence of operations carried out by the submodules of the proposed architecture within one iteration of 1BOX.

1) UPC

This module contains a $U \times W$ memory block labeled S-MEM that stores the matrix of estimated symbols $\mathbf{S}^{(k)}$ at iteration k. All entries of this memory block are updated at the end of each iteration with the values obtained from the MVM2 module. The control unit, labeled "CTRL", is responsible for synchronizing operations of the entire architecture, as all other modules mainly consist of steaming data paths. Since the matrix $\mathbf{S}^{(k)}$ accumulates the estimates over iterations as in Algorithm 1, it has to be initialized at the beginning of the first iteration. In order to avoid wasting any clock cycles for erasing the content of the S-MEM block and to allow for continuous processing, the control unit asserts the reset signal of the output register of the UPC module, so that initial values of zero are provided to the next module during the first W clock cycles of the first iteration.

At the end of each iteration, the results of the MVM2 module are ready in G-MEM. The UPC module receives the content of G-MEM during the last W clock cycles of each iteration. In the wth cycle, UPC receives the vector $\kappa[\mathbf{G}^{\mathrm{T}}]_w$, retrieves the wth column of S-MEM which contains $[\mathbf{S}^{\mathrm{T}}]_w$, adds it with $\kappa[\mathbf{G}^{\mathrm{T}}]_w$, and writes back the results to wth column of S-MEM after applying the projection operation, as shown on line 12 of Algorithm 1. When updating the S-MEM block with the result from the MVM2 module during the first iteration, the control unit asserts the reset signal of the register before the accumulator to ensure that the S-MEM content from the previous detection task does not get accumulated. In addition, during the first W clock cycles of each iteration, the control unit sets the select signal of the multiplexer (MUX) at the output of the UPC module so that its output comes directly from the projection operation, since during these clock cycles the content of S-MEM is being updated and hence not ready to be passed to the output.

2) MVM1

This module consists of U complex-valued multipliers and a balanced adder tree that sums the results of U multipliers. In every clock cycle, two U-dimensional vectors enter the module and their inner product is computed after L_{MVM1} clock cycles, including additional clock cycles caused by pipelining. The value of L_{MVM1} depends on the number of UEs U. For $U = 4$ and $U = 8$, for example, the number of clock cycles are 4 and 5, respectively.

3) FTF

The FTF module contains the FFT and inverse FFT (IFFT) submodules, which are implemented using the Xilinx LogiCORE FFT IP with radix-2 *pipelined streaming I/O* architecture. This streaming FFT architecture achieves high throughput and provides continuous processing capability, i.e., accepts one sample of a W-point vector per clock cycle and produces one entry of the resulting transform per clock cycle, after a latency of L_{FFT} clock cycles. The IFFT and FFT submodules carry out the operations on lines 6 and 7 of Algorithm 1, respectively. Each output of the IFFT core, is multiplied by $1/\tilde{\sigma}$ (see Section III-B2) and then sign-refined[2] by the corresponding 1-bit received signal $[\mathbf{r}_b]_w$ (carried out with the logic labeled with "SR" on Figure 4) to produce the vector $\boldsymbol{\alpha}_b$ as shown on line 6. Note that the missing factor of $\sqrt{2}$ is absorbed to the scaling schedule of the IFFT implementation. The result, which is in the time domain, is then fed to the nonlinear function $\tilde{\omega}$ defined in (14), which is implemented by a small LUT, as detailed in Section III-A. The output of the $\tilde{\omega}$ submodule is once again sign-refined before being converted back to the frequency-domain by the FFT core. For an OFDM system with $W = 128$ subcarriers, the FTF module has a latency of $L_{\text{FTF}} = 702$ clock cycles, which is the sum of the latencies of all submodules.

4) MVM2

This module is responsible for the matrix-vector multiplications on line 10 of Algorithm 1, as well as the multiplication by the step size κ, i.e., computes $\kappa\mathbf{G}$. To this end, the MVM2 module contains U processing elements (PEs), which consists of a complex-valued multiplier, a complex-valued adder, and logic to perform complex conjugation

2. Sign-refinement of a complex signal x with a complex 1-bit signal r refers to the operation $x \odot r$; each real and imaginary part of x is negated if the corresponding part of r is -1 and stays unaltered otherwise.

TABLE 1. Implementation results and comparison on a Xilinx Virtex-7 XC7VX690T FPGA.

Detector	1BOX	1BOX	NS-SCFDMA [31]	OCD [26]	PGS [42]	RCG [27]	ESDBB [43]
System dimension ($B \times U$)	64×4	128×8	128×8	128×8	128×8	128×8	128×8
Designed for 1-bit ADCs?	**yes**	**yes**	no	no	no	no	no
OFDM or SC-FDMA	OFDM	OFDM	SC-FMDA	no	no	no	no
Slices	1 201 (1.11%)	1 627 (1.5%)	48 244 (45%)	11 094 (10%)	NA[a]	NA[a]	NA[a]
LUTs	3 125 (0.72%)	3 952 (0.91%)	148 797 (34%)	23 914 (5.5%)	10 085 (2.3%)	4 587 (1.0%)	3 631 (0.83%)
FFs	4 412 (0.51%)	5 221 (0.6%)	161 934 (19%)	43 008 (4.96%)	12 074 (1.4%)	18 782 (2.1%)	7 515 (0.85%)
DSP48 units	52 (1.44%)	84 (2.33%)	1 016 (28%)	774 (21.5%)	266 (7.4%)	972 (27.0%)	272 (7.42%)
Block RAMs	22.5	72	16	2	4	NA[a]	NA[a]
Max. clock frequency [MHz]	303	303	317	258	322	210	210
Latency [clock cycles]	26 706	51 282	196	795	204	NA	322
Throughput [Mb/s]	18.14 [b]	18.89 [b]	414 [c]	250 [c]	51 [c]	420 [c]	31.3
Throughput/LUT	5 808	4 784	2 782	10 398	5 024	91 563	8 619
Throughput/NRC [d]	820	578	695	881	527	1 421	358

[a]Number not reported.
[b]The throughput of the 1BOX detector is reported for $K = 3$ iterations and 16-QAM.
[c]The throughput of the design has been scaled to 16-QAM.
[d]Normalized Resource Consumption (NRC) is calculated as NRC = LUT + FF + 280 × DSP48 [44].

(denoted by "conj.") and shifting to carry out the multiplication by the step size (κ), since the step size is chosen to be a negative power of two (i.e., 1/32). The architecture details of a PE is shown on the right side of Figure 4.

We note that the sequence of operations carried out by the MVM2 module is not exactly the same as shown by the for-loop between lines 9 to 11 of Algorithm 1; this is because the MVM2 module receives the elements of the $B \times W$ matrix $\mathbf{V}$, defined on line 7 of Algorithm 1, in a row-by-row fashion, over BW clock cycles, while the line 10 of the algorithm shows the multiplication with one column of $\mathbf{V}$ at a time. The module receives the bth row $[\mathbf{V}^T]_b$, during clock cycles $(b-1)W+1$ to bW. As soon as it receives the bth element of $[\mathbf{V}]_w$ (which happens W clock cycles after receiving the $(b-1)$th element of $[\mathbf{V}]_w$), the U PEs multiply it with the bth column of $\mathbf{H}_w^{\mathrm{H}}$, right-shift the product by $\log(\kappa^{-1})$ bits and accumulate the result in the wth column of G-MEM. This process continues until all BW elements of $\mathbf{V}$ have been received and the wth column of G-MEM contains the result of the matrix-vector multiplication $\mathbf{H}_w^{\mathrm{H}}[\mathbf{V}]_w$, for $w = 1, \ldots, W$. The MVM2 module has a latency of $L_{\mathrm{MVM2}} = 4$, due to its internal pipeline stages.

5) H-MEM

This module is a $BW \times U$ memory that stores the W frequency-domain channel matrices $\{\hat{\mathbf{H}}_w\}_{w=1}^{W}$. During the last iteration of each data detection task, in which the entries of the channel matrices are used for the last time, the control unit within the UPC module signals the channel estimation module that the H-MEM is ready to receive new matrices pertaining to the next data detection problem and asserts the write-enable (WE) signal of the H-MEM block.

Remark 4: All modules of the proposed architecture have been carefully pipelined in order to improve hardware efficiency. As a result, our designs achieve relatively high clock frequencies. We used pipelining registers at the inputs and outputs of each multiplier, implemented by DSP48 units of the FPGA. However, in order to keep our architecture diagrams simple, we do not show these pipelining registers in Figure 4. The critical path of the design is shown with red color in Figure 4 and passes through the memory array "r-RAM" within the FTF module.

C. TIMING SCHEDULE

Figure 5 illustrates the sequence of operations carried out within one iteration of Algorithm 1, by each of the four main modules detailed in Section IV-B. The horizontal axis shows the progress of time measured in clock cycles. Each row of this diagram corresponds to one of the modules, indicated by its name on the right. The small boxes in each row show the operations of one clock cycle of the corresponding module (if it is active). Each box, contains the antenna (b) and frequency (w) indices of the data being processed in that clock cycle, as well as the output of the module in that cycle (if it exists). Additionally, the diagram shows the timing schedule of the modules relative to each other. For example, at the beginning of each iteration, the MVM1 module begins to compute the first entry of $\mathbf{z}_1$, i.e., $[\mathbf{z}_1]_1$, after a latency of L_{UPC} clock cycles. Subsequently, the FTF module begins its operation once it receives a valid output from the MVM1 module, which requires a latency of $L_{\mathrm{UPC}} + L_{\mathrm{MVM1}}$ clock cycles, counted from the beginning of the iteration. It is important to note that the MVM2 module produces valid outputs only during the last W clock cycles of its operation in each iteration, due to its multiply-accumulate approach for computing the matrix-vector product; see Section IV-B for more details. Once the result ($\kappa\mathbf{G}$) of the MVM2 module is ready, they are transferred to the UPC module in order to update the estimate matrix $\mathbf{S}^{(k-1)}$ of the previous iteration, according to line 12 of Algorithm 1. In total, each algorithm iteration requires $BW + L_{\mathrm{UPC}} + L_{\mathrm{MVM1}} + L_{\mathrm{FTF}} + L_{\mathrm{MVM2}}$ clock cycles as discussed in Section IV-A.

Remark 5: Typical values of B and W encountered in massive MU-MIMO-OFDM systems, result in relatively high latency and low per-instance throughput, as seen in Table 1.

This stems from the sequential nature of the proposed 1BOX algorithm, which exhibits stringent data dependencies caused by the repeated time-to-frequency and frequency-to-time transforms. Therefore, it is non-trivial to decompose our algorithm into independent tasks, as opposed to linear data detectors that transform the frequency-selective system into a set of independent frequency-flat systems. Nevertheless, a straightforward solution to increase the throughput would be to deploy multiple parallel instances that operate on different sets of received data. Besides that, it is also possible to increase the throughput per instance of the proposed data detector, by using highly-parallel FFT architectures and matrix-vector product engines. Further possible improvements are (i) more aggressive pipelining and (ii) interleaved processing of multiple symbols concurrently in a single instance. Combining these techniques with an ASIC implementation in a modern CMOS technology node might be able to achieve Gb/s throughputs. The design of such improved architectures is left for future work.

D. FIXED-POINT PARAMETERS

In order to maximize hardware efficiency, we deploy fixed-point arithmetic. The word-length of each signal has been optimized based on BER simulations to minimize the loss compared to a floating-point reference model, while maintaining low area. In what follows, we use the notation $[x.y]$ to denote the 2's complement binary format of a fixed-point signal that has x integer bits (including the sign bit) and y fractional bits with a total word length of $x + y$ bits. The reported word lengths are for each of the real and imaginary components of complex-valued signals. The channel entries $[\mathbf{H}_w]_b$ are in format [4.4]. The entries of the vectors $\mathbf{z}_b$ are in format [5.5]. The entries of the matrices $\mathbf{V}$, $\mathbf{G}$, and $\mathbf{S}^{(k)}$ are in format [4.4], [1.7], and [2.7], respectively. For the entries of $\boldsymbol{\alpha}_b$, we use format [5.4]. The $\tilde{\omega}$ LUT storing the fixed-point values of the function $\tilde{\omega}(t)$ for $t_n < t < t_p$, consists of 128 entries in format [3.4]. The LUT containing the values of $1/\tilde{\sigma}$ consists of 256 entries in format [2.6]. In Figure 4, we show the word length of real and imaginary parts of each complex-valued signal with a number next to the signal connection. The BER performance corresponding to the fixed-point hardware implementation is shown in Figure 3.

E. IMPLEMENTATION RESULTS AND COMPARISON

In order to demonstrate the efficacy of the proposed architecture for the 1BOX algorithm, we now present implementation results on a Xilinx Virtex-7 XC7VX690T FPGA. Table 1 summarizes the FPGA implementation results of the 1BOX detector for two system dimensions $B \times U$, i.e., 64×4 and 128×8. We reiterate that these are, to the best of our knowledge, first hardware implementation results of a 1-bit data detector, which renders a fair comparison with existing designs difficult. Nevertheless, we provide a comparison with state-of-the-art massive MIMO data detectors that have been designed for massive MU-MIMO systems with high-resolution ADCs. We reiterate that this comparison is not entirely fair and its purpose is to show the overhead in hardware efficiency caused by the proposed iterative detection algorithm that supports massive MU-MIMO-OFDM systems with 1-bit ADCs. We emphasize that the reference linear data detectors are not specialized to operate in 1-bit systems and therefore perform poorly in such systems; see Section III-F for the details.

The design in [31] implements an L-MMSE data detector for single-carrier frequency-division multiple access (SC-FDMA)-based massive MU-MIMO systems. This design, which is referred to as "NS-SCFDMA" in this article, uses three iterations of Neumann series to perform an approximate matrix inversion per subcarrier. The design in [26] implements an optimized coordinate descent (OCD) algorithm that approximates a box-constrained detection problem for massive MU-MIMO-OFDM systems—frequency-domain conversion circuitry is, however, excluded. The design in [42] implements a parallel Gauss-Seidel (PGS) algorithm that iteratively solves an L-MMSE data detection problem for frequency-flat channels. In Table 1, we also include two of the most recent, state-of-the-art linear detectors for massive MU-MIMO systems, proposed in [27] and [43]. The design presented in [27], implements a recursive conjugate-gradient-based L-MMSE detector, called RCG, and the design in [43] implements an algorithm based on steepest descent and Barzilai-Borwein algorithms, abbreviated to ESDBB, that solves the L-MMSE detection problem iteratively. All of the compared designs consider a system with $B = 128$ BS antennas and $U = 8$ UEs. However, the throughputs reported in the reference designs NS-SCFDMA, OCD, PGS and RCG are for 64-QAM, while we consider 16-QAM for the 1BOX data detector. Therefore, in Table 1, we scale the reported throughput of these reference designs by a factor of $4/6$ so that the throughput of all designs is with respect to 16-QAM. We include the throughput for ESDBB as it was reported in [43], since the modulation scheme was not specified for that design. We note that PGS, NS-SCFDMA and RCG designs contain hardware to compute post-equalization signal-to-noise-plus-interference ratio (SINR) and log-likelihood ratios (LLRs), which is not present in our design. Table 1 includes two metrics for comparing the hardware efficiency of different designs: (i) throughput per look-up table (LUT) utilization and (ii) throughput per normalized resource consumption (NRC), which is calculated as $\text{NRC} = \text{LUT} + \text{FF} + 280 \times \text{DSP48}$ [44]. The NRC is a more accurate measure for assessing the resource consumption than the LUT count alone as it takes into account the other FPGA resources as well.

As shown in Table 1, the throughput and latency results of the 1BOX data detector fall short of the reference data detectors, which are specialized for BS architectures with high-resolution ADCs. However, our design achieves similar hardware efficiency in terms of throughput/NRC compared to other designs, and even higher than that of PGS and ESDBB.

TABLE 2. FPGA resource breakdown for the proposed 1BOX data detector on a Xilinx Virtex-7 XC7VX690T FPGA.

System dim.	64×4					128×8				
Module	H-MEM	UPC	MVM1	FTF	MVM2	H-MEM	UPC	MVM1	FTF	MVM2
LUTs	32 (1%)	373 (12%)	226 (7%)	2 263 (72%)	231 (8%)	64 (1%)	712 (18%)	482 (12%)	2 263 (58%)	431 (11%)
FFs	0 (0%)	441 (10%)	154 (3%)	3 615 (82%)	232 (5%)	0 (0%)	849 (16%)	346 (6%)	3 616 (70%)	410 (8%)
DSP48 units	0 (0%)	0 (0%)	16 (30%)	20 (40%)	16 (30%)	0 (0%)	0 (0%)	32 (38%)	20 (24%)	32 (38%)
Block RAMs	16 (72%)	0 (0%)	0 (0%)	5 (22%)	1.5 (6%)	64 (89%)	0 (0%)	0 (0%)	5 (7%)	3 (4%)

The RCG implementation achieves the highest throughput and efficiency, but excludes OFDM processing circuitry. It is important to discern that the proposed 1BOX implementation directly operates on 1-bit measurements, while the reference designs perform linear data detection based on high-resolution measurements. Consequently, the error-rate performance of the reference designs would, at best, be close to that of ZF-DET shown in Section III-F for 1-bit massive MU-MIMO systems. Additionally, our 1BOX implementation is designed for OFDM systems and includes DFT processing, while the other designs do not include DFT processing—except for NS-SCFDMA, which includes logic for SC-FDMA processing. As a result, their hardware efficiency would be lower if OFDM processing circuitry would be included. We note that it is not possible to exclude the submodules corresponding to OFDM processing in the FTF module from 1BOX detector to enable a meaningful comparison with the reference designs that do not include OFDM processing. This is due to the fact that the operations in the FTF submodule are an integrated part of each iteration of 1BOX algorithm and they are not separable from the rest of the algorithm. This is in stark contrast to linear detectors for high-resolution OFDM-based systems, that decompose the system into independent frequency-flat subsystems, and a comparison between detectors designed for OFDM-based systems and detectors designed for frequency-flat systems is possible.

In order to shed more light on the FPGA resource utilization of the proposed 1BOX data detector, Table 2 shows a breakdown of resources corresponding to the individual modules. We note that the FTF module consumes the largest portion of resources. In turn, roughly 80% of the FTF resources are consumed by the FFT and IFFT cores that carry out the time-domain to frequency-domain transform and vice versa. This confirms that a direct comparison with other data detectors that exclude such time-to-frequency conversion is not accurate. In addition, Table 2 shows that most of the block RAMs are used in the H-MEM module, which contains the channel matrices for all subcarriers—the reference FPGA designs in Table 1 for high-resolution ADCs do not provide information on the amount of storage allocated for channel matrices.

V. CONCLUSION

We have proposed a quantization-aware data detection algorithm, called 1BOX, for massive MU-MIMO-OFDM systems operating over frequency-selective channels with 1-bit ADCs. To improve the channel estimates for such architectures, we have also proposed a channel estimation algorithm referred to as NGD-CHEST. We have shown using simulations that NGD-CHEST combined with the 1BOX data detector outperform conventional linear channel estimation and data detection methods in terms of error-rate performance. Furthermore, we have developed a reference VLSI architecture and presented corresponding FPGA implementations for 1BOX detector. Our results demonstrate that the proposed design achieves comparable hardware efficiency to data detectors that have been designed for high-resolution systems. The proposed channel estimation and data detection algorithms are particularly useful in all-digital massive MU-MIMO-OFDM systems operating at mmWave or terahertz (THz) frequencies, which require large antenna arrays and high bandwidths. Other use cases for the proposed algorithms and hardware designs could be radar or imaging systems for automotive or robotics applications that operate at high carrier frequencies (e.g., mmWave or THz), in which the RF power consumption is expected to be a bottleneck due to the high bandwidths and the large number of antennas.

There are numerous avenues for future work. Each instance of the 1BOX data detector achieves throughputs of tens of Mb/s, which is insufficient for next-generation wireless systems operating at millimeter-wave frequencies. However, there exist several techniques as outlined in Section IV-C, that can significantly increase the throughput per instance of our architecture. The design of such high-throughput architectures is part of ongoing work. Another important research direction is to explore the overall performance-complexity-cost trade-off in basestations that use low-resolution data converters. In such a study, the effect of reduced data converter resolution on the overall system power consumption and cost should be considered. Finally, the design of efficient algorithms for uplink timing and frequency synchronization as well as parameter estimation (such as SNR or noise variance) for systems with 1-bit ADCs, is an important open research problem.

REFERENCES

[1] S. H. Mirfarshbafan, M. Shabany, A. Amini, and S. A. Nezamalhosseini, "Near-ML detection in massive MIMO systems with one-bit ADCs: Algorithm and VLSI design," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Oct. 2018, pp. 1–5.

[2] 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.

[3] 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.

[4] P. Skrimponis *et al.*, "Power consumption analysis for mobile mmWave and sub-THz receivers," in *Proc. IEEE 6G Wireless Summit*, Mar. 2020, pp. 1–5.

[5] S. Dutta, C. N. Barati, A. Dhananjay, D. A. Ramirez, J. F. Buckwalter, and S. Rangan, "A case for digital beamforming at mmWave," *IEEE Trans. Wireless Commun.*, vol. 19, no. 2, pp. 756–770, Feb. 2020.

[6] T. S. Rappaport *et al.*, "Millimeter wave mobile communications for 5G cellular: It will work!" *IEEE Access*, vol. 1, pp. 335–349, 2013.

[7] 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.

[8] 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.

[9] A. K. Saxena, I. Fijalkow, and A. L. Swindlehurst, "Analysis of one-bit quantized precoding for the multiuser massive MIMO downlink," *IEEE Trans. Signal Process.*, vol. 65, no. 17, pp. 4624–4634, Sep. 2017.

[10] J. Choi, J. Mo, and R. W. Heath, Jr., "Near maximum-likelihood detector and channel estimator for uplink multiuser massive MIMO systems with one-bit ADCs," *IEEE Trans. Commun.*, vol. 64, no. 5, pp. 2005–2018, May 2016.

[11] S. Wang, Y. Li, and J. Wang, "Convex optimization based multiuser detection for uplink large-scale MIMO under low-resolution quantization," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2014, pp. 4789–4794.

[12] R. H. Walden, "Analog-to-digital converter survey and analysis," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 4, pp. 539–550, Apr. 1999.

[13] C. Studer and G. Durisi, "Quantized massive MU-MIMO-OFDM uplink," *IEEE Trans. Commun.*, vol. 64, no. 6, pp. 2387–2399, Jun. 2016.

[14] C. Risi, D. Persson, and E. G. Larsson. (Apr. 2014). *Massive MIMO With 1-Bit ADC*. [Online]. Available: http://arxiv.org/abs/1404.7736

[15] Y. Li, C. Tao, L. Liu, G. Seco-Granados, and A. L. Swindlehurst, "Channel estimation and uplink achievable rates in one-bit massive MIMO systems," in *Proc. IEEE Sensor Array Multichannel Signal Process. Workshop (SAM)*, Rio de Janeiro, Brazil, Jul. 2016, pp. 1286–1289.

[16] S. Jacobsson, G. Durisi, M. Coldrey, U. Gustavsson, and C. Studer, "One-bit massive MIMO: Channel estimation and high-order modulations," in *Proc. IEEE Int. Conf. Commun. Workshop (ICCW)*, London, U.K., Jun. 2015, pp. 1304–1309.

[17] C. Mollén, J. Choi, E. G. Larsson, and R. W. Heath, Jr., "Uplink performance of wideband massive MIMO with one-bit ADCs," *IEEE Trans. Wireless Commun.*, vol. 16, no. 1, pp. 87–100, Jan. 2017.

[18] J. J. Bussgang, "Crosscorrelation functions of amplitude-distorted Gaussian signals," Res. Lab. Elect., Cambridge, MA, USA, Rep. 216, Mar. 1952.

[19] Y. Li, C. Tao, G. Seco-Granados, A. Mezghani, A. L. Swindlehurst, and L. Liu, "Channel estimation and performance analysis of one-bit massive MIMO systems," *IEEE Trans. Signal Process.*, vol. 65, no. 15, pp. 4075–4089, Aug. 2017.

[20] C.-K. Wen, C.-J. Wang, S. Jin, K.-K. Wong, and P. Ting, "Bayes-optimal joint channel-and-data estimation for massive MIMO with low-precision ADCs," *IEEE Trans. Signal Process.*, vol. 64, no. 10, pp. 2541–2556, Jul. 2015.

[21] S. Wang, Y. Li, and J. Wang, "Multiuser detection for uplink large-scale MIMO under one-bit quantization," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Sydney, NSW, Australia, Jun. 2014, pp. 4460–4465.

[22] C. Cao, H. Li, and Z. Hu, "An AMP based decoder for massive MU-MIMO-OFDM with low-resolution ADCs," in *Proc. IEEE Int. Conf. Comput. Netw. Commun. (ICNC)*, Jan. 2017, pp. 449–453.

[23] H. Wang, T. Liu, C. K. Wen, and S. Jin, "Optimal data detection for OFDM system with low-resolution quantization," in *Proc. IEEE Int. Conf. Commun. Syst. (ICCS)*, Dec. 2016, pp. 1–6.

[24] S. Shahabuddin, M. Juntti, and C. Studer, "ADMM-based infinity norm detection for large MU-MIMO: Algorithm and VLSI architecture," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Sep. 2017, pp. 1–4.

[25] K. Li, C. Jeon, J. R. Cavallaro, and C. Studer, "Decentralized equalization for massive MU-MIMO on FPGA," in *Proc. Asilomar Conf. Signals Syst. Comput.*, Apr. 2017, pp. 1532–1536.

[26] M. Wu, C. Dick, J. R. Cavallaro, and C. Studer, "High-throughput data detection for massive MU-MIMO-OFDM using coordinate descent," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 63, no. 12, pp. 2357–2367, Dec. 2016.

[27] L. Liu *et al.*, "Energy- and area-efficient recursive-conjugate-gradient-based MMSE detector for massive MIMO systems," *IEEE Trans. Signal Process.*, vol. 68, pp. 573–588, Jan. 2020.

[28] G. Peng, L. Liu, P. Zhang, S. Yin, and S. Wei, "Low-computing-load, high-parallelism detection method based on Chebyshev iteration for massive MIMO systems with VLSI architecture," *IEEE Trans. Signal Process.*, vol. 65, no. 14, pp. 3775–3788, Apr. 2017.

[29] G. Peng, L. Liu, S. Zhou, S. Yin, and S. Wei, "A 2.92-Gb/s/W and 0.43-Gb/s/MG flexible and scalable CGRA-based baseband processor for massive MIMO detection," *IEEE J. Solid-State Circuits*, vol. 55, no. 2, pp. 505–519, Nov. 2019.

[30] C. Jeon, G. Mirza, R. Ghods, A. Maleki, and C. Studer, "VLSI design of a nonparametric equalizer for massive MU-MIMO," in *Proc. Asilomar Conf. Signals Syst. Comput.*, Apr. 2018, pp. 1504–1508.

[31] M. Wu *et al.*, "Large-scale MIMO detection for 3GPP LTE: Algorithms and FPGA implementations," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 916–929, Oct. 2014.

[32] S. Jacobsson, C. Lindquist, G. Durisi, T. Eriksson, and C. Studer, "Timing and frequency synchronization for 1-bit massive MU-MIMO-OFDM downlink," in *Proc. IEEE Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Cannes, France, Jul. 2019, pp. 4670–4684.

[33] A. Zymnis, S. Boyd, and E. Candès, "Compressed sensing with quantized measurements," *IEEE Signal Process. Lett.*, vol. 17, no. 2, pp. 149–152, Feb. 2010.

[34] J. Choi, D. J. Love, D. R. Brown, and M. Boutin, "Quantized distributed reception for MIMO wireless systems using spatial multiplexing," *IEEE Trans. Signal Process.*, vol. 63, no. 13, pp. 3537–3548, Jul. 2015.

[35] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, Jan. 2009.

[36] T. Goldstein, C. Studer, and R. G. Baraniuk. (Feb. 2016). *A Field Guide to Forward-Backward Splitting With a FASTA Implementation*. [Online]. Available: https://arxiv.org/abs/1411.3406

[37] O. Castañeda, S. Jacobsson, G. Durisi, T. Goldstein, and C. Studer, "VLSI design of a 3-bit constant-modulus precoder for massive MU-MIMO," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Florence, Italy, May 2018, pp. 683–686.

[38] B. Yin, M. Wu, J. R. Cavallaro, and C. Studer, "Conjugate gradient-based soft-output detection and precoding in massive MIMO systems," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2014, pp. 3696–3701.

[39] A. Abdallah, M. M. Mansour, A. Chehab, and L. M. A. Jalloul, "MMSE detection for 1-bit quantized massive MIMO with imperfect channel estimation," in *Proc. IEEE Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Jun. 2018, pp. 1–5.

[40] L. V. Nguyen and D. H. Nguyen. (Jul. 2019). *Linear Receivers for Massive MIMO Systems With One-Bit ADCs*. [Online]. Available: https://arxiv.org/abs/1907.06664

[41] S. Haene, A. Burg, N. Felber, and W. Fichtner, "OFDM channel estimation algorithm and ASIC implementation," in *Proc. IEEE Eur. Conf. Circuits Syst. Commun. (ECCSC)*, Jul. 2008, pp. 270–275.

[42] Z. Wu, Y. Xue, X. You, and C. Zhang, "Hardware efficient detection for massive MIMO uplink with parallel Gauss–Seidel method," in *Proc. IEEE Int. Conf. Digit. Signal Process. (DSP)*, Aug. 2017, pp. 1–5.

[43] X. Tan *et al.*, "Enhanced linear iterative detector for massive multiuser MIMO uplink," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 67, no. 2, pp. 540–552, Feb. 2020.

[44] 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.

SEYED HADI MIRFARSHBAFAN received the B.Sc. and M.Sc. degrees in electrical engineering from the Sharif University of Technology, Tehran, Iran, in 2016 and 2018, respectively, and the M.Sc. degree in electrical and computer engineering from Cornell University, Ithaca, NY, USA, in 2020. He is currently pursuing the Ph.D. degree with the Integrated Information Processing Group, ETH Zürich, Zürich, Switzerland. His research interests include wireless communications and design of digital VLSI circuits and systems.

MAHDI SHABANY received the B.Sc. degree in electrical engineering from the Sharif University of Technology, Tehran, Iran, in 2002, and the M.Sc. and Ph.D. degrees in electrical engineering from the University of Toronto, Toronto, Canada, in 2004 and 2008, respectively. He is an Associate Professor with the Electrical Engineering Department, Sharif University of Technology. He also works with the University of Toronto periodically as a Visiting Researcher. From 2007 to 2008, he was with Redline Communications Company, Toronto, where he developed and patented designs for WiMAX systems. He also served as a Postdoctoral Fellow with the University of Toronto in 2009. He spent a year as a Senior ASIC Designer with Synopsys in 2019. He holds three U.S. patents. His main research interests include digital electronics, VLSI architecture/algorithm design for broadband communication systems, and efficient implementation of signal processing algorithms for various applications, including imaging and bio-oriented systems.

SEYED ALIREZA NEZAMALHOSSEINI received the B.Sc. degree in electrical engineering from the Amirkabir University of Technology, Tehran, Iran, in 2006, and the M.Sc. and Ph.D. degrees in electrical engineering from the Sharif University of Technology, Tehran, in 2008 and 2013, respectively. He is currently an Assistant Professor with the Iran University of Science and Technology, Tehran. His research interests include underwater wireless optical communications, mode-division multiplexing inăoptical fibers, and visible light communications.

CHRISTOPH STUDER (Senior Member, IEEE) received the Ph.D. degree in information technology and electrical engineering from ETH Zürich, Zürich, Switzerland, in 2009.

He is an Associate Professor with the Department of Information Technology and Electrical Engineering, ETH Zürich. In 2005, he was a Visiting Researcher with the Smart Antennas Research Group, Stanford University. From 2006 to 2009, he was a Research Assistant with the Integrated Systems Laboratory and the Communication Technology Laboratory (CTL), ETH Zürich. From 2009 to 2012, he was a Postdoctoral Researcher with CTL, ETH Zürich, and the Digital Signal Processing Group, Rice University. In 2013, he was a Research Scientist with Rice University. From 2014 to 2019, he was an Assistant Professor with Cornell University. From 2019 to 2020, he was an Associate Professor with Cornell University and Cornell Tech, New York, NY, USA. In 2020, he joined ETH Zürich and since 2014, he has also been an Adjunct Professor with Rice University. His research interests include the design of application-specific integrated circuits, as well as wireless communications, digital signal processing, optimization, and machine learning.

Dr. Studer received the ETH Medals for his M.S. and Ph.D. theses in 2006 and 2009, respectively. He received the Swiss National Science Foundation Fellowship for Advanced Researchers in 2011 and the U.S. National Science Foundation CAREER Award in 2017. He won the Michael Tien '72 Excellence in Teaching Award from the College of Engineering, Cornell University in 2016. He shared the Swisscom/ICTnet Innovations Award in 2010 and 2013. He was the winner of the Student Paper Contest of the 2007 Asilomar Conference on Signals, Systems, and Computers, received the Best Student Paper Award of the 2008 International Symposium on Circuits and Systems (ISCAS), and shared the Best Live Demonstration Award at IEEE ISCAS in 2013. He is currently an Associate Editor of the IEEE OPEN JOURNAL OF CIRCUITS AND SYSTEMS. In 2019, he was the Technical Program Chair of the Asilomar Conference on Signals, Systems, and Computers, and a Technical Program Co-Chair of the IEEE International Workshop on Signal Processing Systems. Since 2019, he has been an Associate Editor of the IEEE OPEN JOURNAL OF CIRCUITS AND SYSTEMS.