A 0.58-mm^2 2.76-Gb/s 79.8-pJ/b 256-QAM Message-Passing Detector for a 128 × 32 Massive MIMO Uplink System

Wei Tang, *Member, IEEE*, Chia-Hsiang Chen, *Member, IEEE*, and Zhengya Zhang, *Senior Member, IEEE*

***Abstract*— Massive multiple-input–multiple-output (MIMO) detection uses a large number of antennas to increase spectral efficiency at a cost of large computation resources and power in a base station. In this article, we demonstrate a 0.58-mm^2 128 × 32 (it denotes 128 base station antennas and 32 single-antenna users) 256-QAM massive MIMO uplink detector based on message-passing detection (MPD). With the proposed symbol hardening processing, the complexity is reduced by more than 60% compared to the direct implementation of MPD. The detector implements a grouped layer-parallel architecture to accelerate convergence, enabling an average throughput of 2.76 Gb/s (running, on average, 4.92 iterations with early termination) at 221 mW. The chip incorporates adaptive precision control and clock gating to improve energy efficiency further by up to 43%.**

***Index Terms*— Channel hardening-exploiting message passing (CHEMP), massive multiple-input–multiple-output (MIMO), message-passing detection (MPD), MIMO detector.**

I. Introduction

The fifth-generation (5G) wireless communication systems significantly increase the network capacity and coverage and improve the spectral and energy efficiency. Massive multiple-input–multiple-output (MIMO), or large-scale MIMO, has been identified as a key disruptive technology for 5G [1]–[6]. Massive MIMO is a multi-user MIMO technique, as depicted in Fig. 1, which relies on a large number, e.g., hundreds, of base station antennas (N_r) to serve a multiplicity of, e.g., tens, of autonomous single-antenna users (N_t) in each time-frequency resource [7]. A large number of antennas provides a high spatial multiplexing gain for an increased capacity, and the radiated energy can be focused on the intended receivers for improved energy efficiency.

The optimal maximum likelihood (ML) detection exhaustively searches for all the possible points of user symbol space. Its complexity scales exponentially with the number of user antennas (N_t) and the order of modulation $|\mathcal{A}|$, i.e., $O(|\mathcal{A}|^{N_t})$. To mitigate the implementation costs, reduced-search detection, such as sphere decoders (SDs) [8]–[13], have been introduced for MIMO systems with $N_t \leq 8$. For a massive MIMO system serving more than eight users with high-order modulations, SD can still be used, but it involves a tradeoff between error rate and complexity. To support more than eight users, the sphere radius needs to be limited to control complexity, which results in an increased error rate.

A linear minimum mean square error (MMSE) detection is of lower cost, and it has demonstrated good performance for massive MIMO systems [2]–[4], [14], especially for large loading ratio (N_r/N_t) and a favorable independent identically distributed (i.i.d.) Rayleigh fading channel. However, linear MMSE detection requires matrix inversion, and the complexity of matrix inversion grows cubically with the number of user terminals, i.e., $O(N_t^3)$. Inversion of large matrices, e.g., 8 × 8 or larger, creates a performance bottleneck and requires a large silicon area, leading to an inefficient implementation. Hence, recent work has looked into approximate or implicit matrix inversion to cut the overhead of large matrix inversion while achieving near-MMSE error rate performance, such as Gauss–Seidel (GS) [15], [16], conjugate gradient (CG) [17]–[19], Richardson (RI) [20], and successive over-relaxation (SOR) [21]. Similarly, K-term Neumann series approximation (NSA)-based detection [22], [23] can reduce complexity to $O(N_t^2)$ when $K \leq 2$, but they sacrifice more than 2-dB SNR compared to a linear MMSE detection when $N_t \geq 8$.

Recently, iterative message-passing detections (MPDs) [24], [25] have emerged. In particular, the channel hardening-exploiting message passing (CHEMP) detection [25] does not require the costly matrix inversion, and it provides an improved error rate than a linear MMSE detection, achieving near-optimal detection in flat Rayleigh fading channels. As the MIMO channel becomes more correlated, it is observed that MPD starts to diverge. To deal with correlated channels, recent works, such as expectation propagation detection (EPD) [26] and large-MIMO approximate message passing (LAMA) detection [27], can achieve a good error rate at a cost of higher complexity, higher energy consumption, and slower throughput.

Fig. 2 shows uncoded bit error rate (BER) simulation of different MIMO detections under different channels. For a flat Rayleigh fading channel, MPD and EPD can approach the

Manuscript received May 19, 2020; revised September 29, 2020 and January 11, 2021; accepted March 19, 2021. Date of publication April 12, 2021; date of current version May 26, 2021. This article was approved by Associate Editor Daniel Friedman. This work was supported in part by NSF under Grant CCF-1054270 and in part by Intel. *(Corresponding author: Wei Tang.)*

Wei Tang and Zhengya Zhang are with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: weitang@umich.edu; zhengya@umich.edu).

Chia-Hsiang Chen was with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA. He is now with the Apple Wireless Connectivity Group, Cupertino, CA 95014 USA.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/JSSC.2021.3069988.

Digital Object Identifier 10.1109/JSSC.2021.3069988

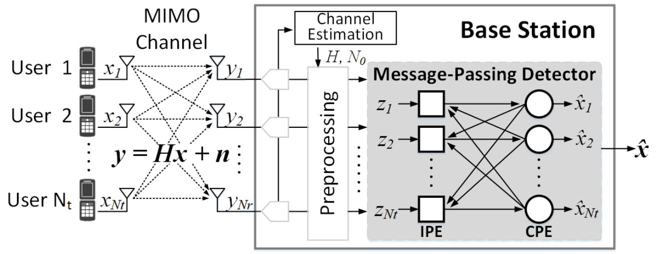


Fig. 1. Top-level block diagram of an $N_r \times N_t$ uplink massive MIMO base station with an MPD detector.

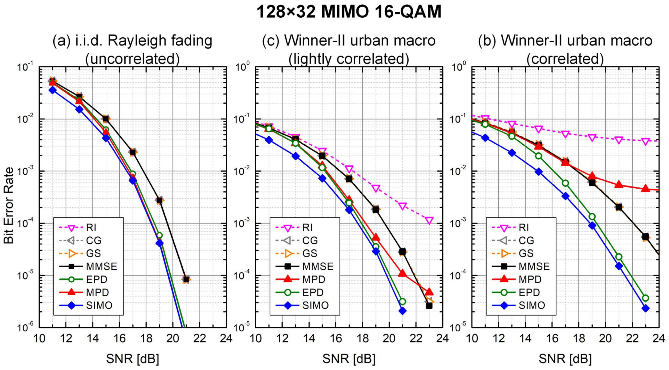


Fig. 2. Uncoded BER of 128 × 32.64-QAM uplink MIMO detections using RI [20], GS [16], CG [18], linear MMSE, expectation propagation detection (EPD) [26], MPD [25], and SIMO lower bound in three channels. (a) i.i.d. Rayleigh fading (uncorrelated), (b) Winner-II urban macro (lightly correlated) with UCA128 of 50-cm radius and a max layout range of 200 m, and (c) Winner-II urban macro (correlated) with UCA128 of 40-cm radius and a max layout range of 100 m.

SIMO lower bound, while other detection algorithms, i.e., RI, CG, and GS, can only converge to the linear MMSE result, as shown in Fig. 2(a). To simulate correlation channels, we use the Winner-II model [28] and set a base station to have a 128 uniform circular antenna (UCA) array to serve 32 single-antenna users. The base station is located at the center of an $R \times R$ area (where R is the maximum layout range), and the 32 users are randomly located within the area, moving at a velocity of 1 m/s. The c2 scenario (urban macro-cell) with a non-line-of-sight (NLOS) propagation condition is used. The level of correlation is controlled by the maximum layout range R and UCA's radius r. For a correlated channel, where $r = 40$ cm and $R = 100$ m, MPD suffers from an error floor of 10^{-2}, as shown in Fig. 2(c). In such a scenario, additional processing resources are required to resolve correlation as in an elaborate detection, e.g., EPD. The light-weighted MPD finds its appropriate use case in a dual-mode detection scheme that captures the efficiency and performance of MPD in uncorrelated channels and the accuracy of EPD in correlated channels. Moreover, MPD can be used as a pre-processor to produce the initial starting point for EPD to speed up convergence.

In this article, we present a low-complexity MPD for a 256-QAM massive MIMO uplink system serving 32 users to demonstrate the unique advantage of MPD in certain channel environments. This work makes the following contributions: 1) a symbol hardening technique to reduce the implementation complexity of the sample design by more than 60% compared to the directly mapped design; 2) a layer-parallel architecture for MPD to double its throughput per unit silicon area, and a grouping architecture to further reduce the area; 3) adaptive precision control to reduce MPD's power consumption; and 4) a fully functional MPD chip prototype to demonstrate an average throughput of 2.76 Gb/s (running, on average, 4.92 iterations with early termination) at 221 mW.

The rest of this article is organized as follows. In Section II, we provide the background of the message-passing detection algorithm and the complexity analysis. In Sections III and IV, the algorithm–architecture co-optimization is presented for the efficient hardware implementation. In Section V, we elaborate on low-power techniques to improve the energy efficiency of the MPD detector. Silicon measurement results are presented in Section VI. Section VII concludes this article.

II. Message-Passing Detection

In a massive MIMO uplink system illustrated in Fig. 1, the base station is equipped with N_r antennas serving N_t single-antenna users at the same time and frequency resources. In the frequency domain, the received signal per-tone $\mathbf{y}^c$ can be modeled as

$$\mathbf{y}^c = \mathbf{H}^c\mathbf{x}^c + \mathbf{n}^c \tag{1}$$

where $\mathbf{x}^c \in \mathbb{C}^{N_t \times 1}$ represents the transmitted M-QAM user symbols, $\mathbf{y}^c \in \mathbb{C}^{N_r \times 1}$ represents the received symbols, and $\mathbf{n}^c$ is an i.i.d. complex Gaussian noise vector, in which each element is modeled as $\mathcal{CN}(0, \sigma_n^2)$. The channel matrix is $\mathbf{H}^c \in \mathbb{C}^{N_r \times N_t}$. Here, an i.i.d. Rayleigh fading channel is assumed to model rich-scattering environments in urban area. For a QAM modulated system, (1) is in complex, and it can be rewritten by separating the real and imaginary parts in real values

$$\begin{bmatrix} \Re(\mathbf{y}^c) \\ \Im(\mathbf{y}^c) \end{bmatrix} = \begin{bmatrix} \Re(\mathbf{H}^c) & -\Im(\mathbf{H}^c) \\ \Im(\mathbf{H}^c) & \Re(\mathbf{H}^c) \end{bmatrix} \begin{bmatrix} \Re(\mathbf{x}^c) \\ \Im(\mathbf{x}^c) \end{bmatrix} + \begin{bmatrix} \Re(\mathbf{n}^c) \\ \Im(\mathbf{n}^c) \end{bmatrix}$$
$$\Rightarrow \mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{n} \tag{2}$$

where $\Re(\cdot)$ and $\Im(\cdot)$ denote the real and imaginary parts, $\mathbf{y} \in \mathbb{R}^{2N_r \times 1}$, $\mathbf{H} \in \mathbb{R}^{2N_r \times 2N_t}$, and $\mathbf{x} \in \mathbb{R}^{2N_t \times 1}$. Note that one M-QAM symbol is decomposed to two orthogonal $\sqrt{M}$-pulse amplitude modulation (PAM) symbols: in-phase and quadrature parts. This enables MPD to process the two independent sets of symbols in parallel.

A. Pre-Processing

Before data detection, the received symbols $\mathbf{y}$ are pre-processed with matched filtering and normalization. Mathematically, the pre-processing is described by

$$\mathbf{z} = \mathbf{J}\mathbf{x} + \mathbf{w},$$
$$\mathbf{z} = \frac{\mathbf{H}^{\mathsf{H}}\mathbf{y}}{N_r}, \quad \mathbf{J} = \frac{\mathbf{H}^{\mathsf{H}}\mathbf{H}\mathbf{x}}{N_r}, \quad \mathbf{w} = \frac{\mathbf{H}^{\mathsf{H}}\mathbf{n}}{N_r} \tag{3}$$

where $\mathbf{z} \in \mathbb{R}^{2N_t \times 1}$ represents the pre-processed input to the detection, $\mathbf{J} \in \mathbb{R}^{2N_t \times 2N_t}$ is called the Gramm matrix, and $\mathbf{w} \in \mathbb{R}^{2N_t \times 1}$ is colored noise with variance σ_w^2. After the pre-processing, MPD starts the iteration of interference cancellation and constellation moment matching to improve the mean and the variance of the estimated symbol $\hat{\mathbf{x}}$ iteration by iteration. The two processing steps are detailed in the following.

B. Interference Cancellation

The pre-processed signal z_i (the ith element of $\mathbf{z}$) can be viewed as the intended symbol x_i coupled with interference from the other transmitted symbols plus noise. To see this, (3) can be rewritten in the element-wise form

$$z_i = J_{ii}x_i + \textstyle\sum_{j=1,j\neq i}^{2N_t} J_{ij}x_j + w_i. \tag{4}$$

Let k_i be the interference-plus noise of the symbol i

$$k_i = \sum_{j=1,j\neq i}^{2N_t} J_{ij}x_j + w_i. \tag{5}$$

When the number of terms, N_t, in (5) is large, as in the case of a massive MIMO uplink system, k_i can be approximated as a Gaussian random variable, i.e., $k_i \sim \mathcal{N}(\mu_{k_i}, \sigma_{k_i}^2)$, according to the central limit theorem and independence assumption. Its mean and variance are calculated by

$$\mu_{k_i} = \sum_{j=1,j\neq i}^{2N_t} J_{ij}\mathrm{E}\big[\hat{x}_j\big], \qquad \sigma_{k_i}^2 = \sum_{j=1,j\neq i}^{2N_t} J_{ij}^2\mathrm{Var}\big[\hat{x}_j\big] + \sigma_w^2 \tag{6}$$

where $\mathrm{E}[\hat{x}_j]$ and $\mathrm{Var}[\hat{x}_j]$ are the mean and the variance of the symbol estimate $\hat{x}_j$. By canceling the interference-plus noise k_i from z_i, the mean and the variance of the ith symbol $\hat{x}_i$ can be calculated as follows:

$$\mu_{\hat{x}_i} = \frac{1}{J_{ii}}\big(z_i - \mu_{k_i}\big), \qquad \sigma_{\hat{x}_i}^2 = \frac{1}{J_{ii}^2}\sigma_{k_i}^2. \tag{7}$$

Note that all the $(1/J_{ii})$, $(1/J_{ii}^2)$, and J_{ij}^2 terms remain constant within a channel coherence time interval; therefore, they are pre-computed and re-used for a stream of data detection. The interference cancellation for each M-QAM symbol takes $8N_t$ real-value multiply–accumulates (MACs).

C. Constellation Moment Matching

The soft symbol estimates obtained from interference cancellation (7) are refined in this step by considering the possible discrete constellation points. The step consists of two substeps.

First, the likelihood of each constellation point is calculated by sampling the Gaussian approximation of the symbol estimate as

$$P(\hat{x}_i = s) \propto \exp\left(\frac{-1}{2\sigma_{\hat{x}_i}^2}\big(\hat{x}_i - \mu_{\hat{x}_i}\big)^2\right) \tag{8}$$

where the in-phase and the quadrature parts are evaluated separately, and for each part, $s \in \mathbb{B}$ is a constellation point in the $\sqrt{M}$-PAM space, i.e., $\mathbb{B} = \{-(\sqrt{M}-1), -(\sqrt{M}-3), \ldots, -1, 1, \ldots, (\sqrt{M}-3), (\sqrt{M}-1)\}$.

Second, after normalized over $\sqrt{M}$ constellation points, the probability $P(\hat{x}_i = s)$ is used to refine the symbol estimate by updating its mean and variance as follows:

$$\mathrm{E}[\hat{x}_i] = \sum_{\forall s\in\mathbb{B}} sP(\hat{x}_i = s)$$
$$\mathrm{Var}[\hat{x}_i] = \sum_{\forall s\in\mathbb{B}} s^2P(\hat{x}_i = s) - \mathrm{E}[x_i]^2. \tag{9}$$

We refer to this update process as constellation moment matching, a re-match of the first and second moments (mean and variance) of the symbol estimate. The updated mean and variance will be used to calculate the interference-plus noise shown in (6) for the next MPD iteration.

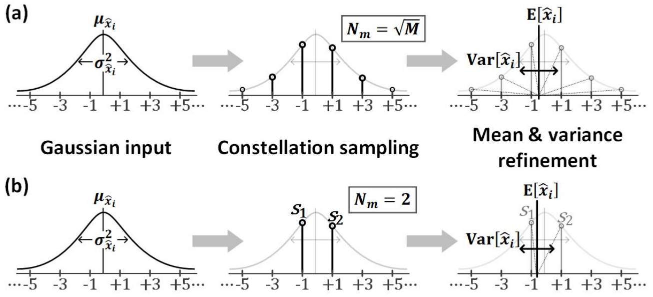


Fig. 3. (a) Constellation matching in exact MPD. (b) Approximate constellation match using two nearest neighbors.

For an M-QAM constellation, the Gaussian evaluation (8) and the mean and variance update (9) are done for the in-phase and the quadrature parts independently. The constellation moment matching takes $2\sqrt{M}$ Gaussian evaluations and $4\sqrt{M} + 2$ MACs. Gaussian evaluations are commonly implemented using table lookups.

D. Message-Passing Implementation

To carry out an MPD iteration, messages are passed between two types of processing elements (PEs), as shown in Fig. 1 An interference cancellation PE (IPE) collects other symbol estimates to compute the interference-plus noise and cancels the interference-plus noise from pre-processed receiver inputs to obtain the updated symbol estimate, as in (7). The updated symbol estimate is passed as a message to the constellation matching PEs (CPEs). In a CPE, the symbol estimate is refined by considering $2\sqrt{M}$ in-phase and quadrature components of constellation points and re-matching the given symbol's first and second moment, as in (9). The refined symbol estimate is passed as a message to IPEs. With more iterations, the accuracy of the symbol estimates is improved.

A complete MPD for a M-QAM massive MIMO uplink system for N_t users requires N_t IPEs and N_t CPEs. The complexities of the IPE and CPE are listed in Table I. To implement a complete 256-QAM 128 × 32 ($N_r = 128$ and $N_t = 32$) massive MIMO uplink system, 32 IPEs and 32 CPEs are needed. An IPE requires 256 MACs, and a CPE requires 66 MACs and 32 Gaussian evaluations. This setup will be used as the baseline for comparison.

III. Symbol-Hardening MPD

The exact constellation processing in a CPE requires exhaustive evaluation of the probabilities of all $2\sqrt{M}$ PAM symbols, in-phase and quadrature, as in (8), and re-matching of the mean and the variance using (9). An illustration is shown in Fig. 3(a).

In this work, we present symbol-hardening MPD to reduce the complexity by more than a half compared to the exact implementation. The symbol hardening technique is derived from the nearest-neighbor approximation, as elaborated in the following.

TABLE I
COMPARISON OF EXACT MPD AND APPROXIMATE MPDS

	Exact MPD $(N_m = \sqrt{M})$	Two-nearest-neighbor approx. MPD $(N_m = 2)$	Symbol-hardening MPD $(N_m = 1)$
#IPEs, #CPEs	N_t IPEs, N_t CPEs	N_t IPEs, N_t CPEs	N_t IPEs
IPE	$8N_t$ MACs	$8N_t$ MACs	$4N_t$ MACs
CPE	$4\sqrt{M}+2$ MACs $2\sqrt{M}$ Gaussian lookups	10 MACs 4 Gaussian lookups	Wiring only

A. Nearest-Neighbor Approximation

Instead of going through $2\sqrt{M}$ symbols in-phase and quadrature, we choose to process only N_m most likely symbols (where $N_m \ll \sqrt{M}$) neighboring to the soft symbol estimate $\mu_{\hat{x}}$. For example, we can choose $N_m = 2$ most likely symbols, s_1 and s_2, and evaluate their probabilities, $P(\hat{x}_i = s_1)$ and $P(\hat{x}_i = s_2)$, as shown in Fig. 3(b). Then, the constellation processing can be approximated by

$$\begin{aligned} \mathrm{E}[\hat{x}_i] &\approx s_1 \; P(\hat{x}_i = s_1) + s_2 \; P(\hat{x}_i = s_2), \\ \mathrm{Var}[\hat{x}_i] &\approx s_1^2 \; P(\hat{x}_i = s_1) + s_2^2 \; P(\hat{x}_i = s_2) - \mathrm{E}[x_i]^2. \end{aligned} \quad (10)$$

The nearest-neighbor approximation reduces the complexity of a CPE from $O(\sqrt{M})$ to $O(N_m)$. For the two-nearest-neighbor approximation, a CPE is reduced to ten MACs and four Gaussian evaluations, which costs about 7× fewer compute units than the baseline.

B. Symbol Hardening

With channel hardening and the diagonal dominance of the Gram matrix [25] in massive MIMO, the variance of the symbol estimates converges at a fast, nearly but below exponential pace. This effect allows for aggressively using $N_m = 1$ in the nearest-neighbor approximation and bypassing the Gaussian evaluation and variance calculation entirely. With $N_m = 1$, constellation processing is simplified to making one hard decision on the symbol estimate, i.e., finding the constellation point that is closest to the symbol estimate. This is termed symbol hardening

$$\mathrm{E}[\hat{x}_i] \approx \lceil \mu_{\hat{x}_i} \rfloor, \quad \mathrm{Var}[\hat{x}_i] \approx 0 \quad (11)$$

where $\lceil . \rfloor$ denotes the hard decision operation. In implementing the CPE, the hard decision can be simple as bit slicing, costing no hardware. In this work, we use 256-QAM and 8-b symbol estimates, so a hard decision can be made by taking the 5 MSBs of the 8-b symbol estimate. Since variance is approximated to be 0 with symbol hardening, the variance calculation in both the CPE and the IPE is eliminated.

The complexity of the exact MPD, the two-nearest-neighbor approximate MPD, and the symbol-hardening MPD are summarized in Table I. In a symbol-hardening MPD, the CPEs are transformed to wiring only, and the number of MACs in IPEs is reduced by half. Compared to the exact MPD, the symbol-hardening MPD sacrifices less than 0.1 dB in SNR for a 128×32.256-QAM massive MIMO uplink system based on the simulation for the i.i.d. Rayleigh fading channel shown in Fig. 11(a).

C. Soft Outputs Computation

In this work, the symbol-hardening MPD computes hard symbols. It can also be extended to compute soft outputs for a coded MIMO system. The approach follows [29], where the interference variation is approximated by scaling the maximal value of the non-diagonal elements in $\mathbf{J}$. The scaling parameter β is optimally determined by the loading factor and channel properties, as discussed in [29]. The soft outputs can be derived from the soft symbol calculated by interference cancellation (8).

IV. ARCHITECTURE AND SCHEDULE OPTIMIZATION

Like the implementation of message-passing decoding of low-density parity-check (LDPC) codes, there are flooding and layered scheduling [30] and the corresponding architecture choices in implementing MPD. In the following, we explore the scheduling and the architecture choices with the goal of maximizing throughput at a low area cost.

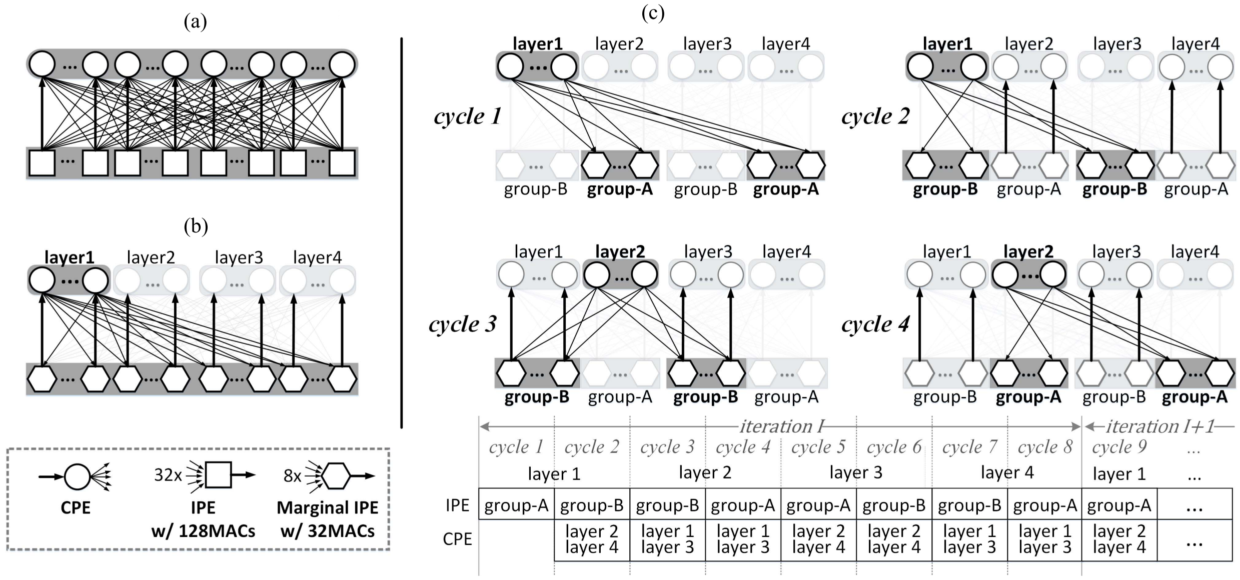
A. Flooding Schedule and Fully Parallel Architecture

A directly mapped fully parallel architecture consists of a set of N_t IPEs and a set of N_t CPEs. One IPE (IPE$_i$) and one CPE (CPE$_i$) are dedicated to a user i. IPE$_i$ receives messages from CPE$_j$, $j \in \{0, 1, \ldots, N_t - 1\}$ and $j \neq i$. CPE$_i$ receives message from IPE$_i$ only.

In this baseline architecture, the two sets of PEs pass messages of symbol estimates between each other following the flooding schedule. In the flooding schedule, all N_t IPEs pass messages of user estimates to all N_t CPEs. After CPE updates, the N_t CPEs pass messages of updated user estimates to all N_t IPEs. This flooding scheduling requires instantiating N_t IPEs and N_t CPEs (in the symbol-hardening MPD, the CPEs are wiring only), costing a large number of hardware resources.

For a 256-QAM 128 × 32 massive MIMO uplink system, a baseline MPD architecture consists of 32 IPEs and 32 CPEs (wiring only), as shown in Fig. 4(a). All the IPEs and CPEs work in parallel. The 32 IPEs send messages at the same time. Even with the symbol hardening approximation, an IPE uses $4N_t = 128$ MACs and has 31 incoming connections and one outgoing connection receive and send messages. The entire baseline architecture requires nearly 4k MACs and 1k connections between the PEs. Assuming 8-b messages, the 1k connections translate to 8k wires.

The detection throughput of the fully parallel architecture can reach $32 f_{\mathrm{clk}}/N_{\mathrm{it}}$ symbol/s, where f_{clk} is the clock frequency, N_{it} is the number of message passing iterations that typically range from 10 to 20, and one clock cycle per iteration is assumed. The throughput of the fully parallel architecture



	cycle 1	cycle 2	cycle 3	cycle 4	cycle 5	cycle 6	cycle 7	cycle 8	cycle 9	...
	layer 1		layer 2		layer 3		layer 4		layer 1	...
IPE	group-A	group-B	group-B	group-A	group-A	group-B	group-B	group-A	group-A	...
CPE		layer 2 layer 4	layer 1 layer 3	layer 1 layer 3	layer 2 layer 4	layer 2 layer 4	layer 1 layer 3	layer 1 layer 3	layer 2 layer 4	...

Fig. 4. MPD architecture optimizations from (a) fully parallel architecture following flooding schedule to (b) layer-parallel architecture, and (c) grouped layer-parallel architecture with two-stage pipeline schedule shown on the bottom.

is high, but the wiring overhead is also high, resulting in a large silicon area, low clock frequency, and high power consumption.

B. Layered Schedule

In LDPC decoding, the flooding schedule requires in each iteration all check node operations to be completed followed by all variable node operations, so all variable node messages of a given iteration are propagated together, and all check node messages of a given iteration are propagated together. Using the layered schedule, check node operations can be done sequentially, and the latest check node messages are used immediately in variable node operations in the same iteration to produce updated variable node messages. The layered schedule allows faster propagation of messages between variable nodes and check nodes, resulting in a faster decoding convergence.

We applied a similar concept from layered LDPC decoding to detection. In the layered schedule, the N_t messages from the IPEs are divided into N_l layers—only a portion of the N_t messages are sent and processed at a time. Each layer of processing updates the partial interference contributed by the users in the layer. Right after a layer is done, the updated symbol estimates are used in the same iteration for interference cancellation of the next layer. This is called intra-iteration forwarding.

In comparison, the layer schedule has two advantages over the flooding schedule.

1) The layer-by-layer processing allows updated user estimates to propagate immediately within the current iteration without having to wait until the next iteration, resulting in faster convergence by nearly 2× compared to the flooding schedule, as shown in Fig 11(a). The symbol-hardening MPD with layered scheduling ($N_l = 4$) requires only seven iterations to achieve nearly the same BER performance as the MPD with a flooding schedule running 13 iterations.
2) Since only a portion of the messages are sent and processed, fewer IPEs are needed, and these IPEs can be time-multiplexed between layers, requiring approximately N_l times less hardware.

C. Layer-Parallel Architecture

With layered scheduling, we can divide the fully parallel architecture into layers for processing. For example, the 32 users can be divided into four layers with eight users per layer ($N_l = 4$). Since only 1/4 of the users are processed per layer, the IPE can be reduced to having 1/4 as many MACs. We call these "marginal" IPEs. A marginal IPE contains only 32 MACs. The layer-parallel MPD architecture is shown in Fig. 4(b). It contains 1k MACs.

The interference cancellation is done layer by layer. In one layer processing, all 32 marginal IPEs compute the partial sum of the interference plus noise from the eight users in the layer and subtract the partial sum. After a layer is done, the updated symbol estimates are used in the next layer of processing.

Since each iteration is divided into four layers, the throughput is reduced by a factor of 4 to $8 f_{\text{clk}}/N_{\text{it}}$ symbol/s, where one clock cycle per layer is assumed. However, as noted previously, layered processing requires half as many iterations to converge, so the throughput is reduced by a factor of 2 only. Considering that the layer-parallel MPD architecture uses approximately 1/4 the number of MACs, the throughput per unit silicon area can be doubled.

D. Grouped Layer-Parallel Architecture

To reduce the area further, we can halve the number of marginal IPEs and time-multiplex their use between two groups of users over two clock cycles. In each cycle, only one group, i.e., half of the users, computes the partial interference and performs the cancellation. The two groups interleave and share the same IPE hardware.

As illustrated in Fig. 4(c), the marginal IPEs are grouped into group-A and group-B. Group-A contains the even

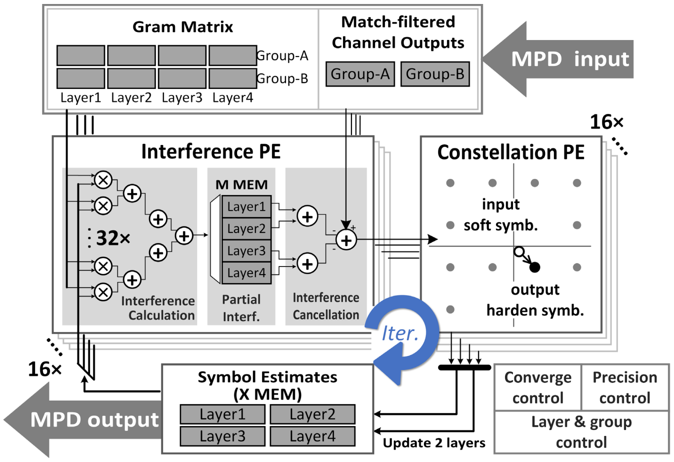


Fig. 5. Block diagram of four-group four-layer MPD design using 16 IPEs and 16 CPEs.

layers 2 and 4, and group-B contains the odd layers 1 and 3. Each layer processing is divided into two substeps, taking one clock cycle per substep. In cycle 1, group-A marginal IPEs compute the partial interference from the users in layer 1 and perform the interference cancellation. In cycle 2, group-B marginal IPEs compute the interference from the users in layer 1 and perform the interference cancellation. Also, in cycle 2, group-A user estimates are updated by hardening the latest symbol estimates from group-A marginal IPEs. The interleaving between group-A and group-B avoids pipeline stalls, as shown in Fig. 4(c).

The interleaving between the two groups of IPEs reduces the number of marginal IPEs by half to 16. Since each marginal IPE contains 32 MACs, the entire grouped layer-parallel architecture contains 512 MACs. The complete architecture is shown in Fig. 5. Since each iteration is twice as long as the layer-parallel architecture, the throughput is reduced by 2 to $4f_{\text{clk}}/N_{\text{it}}$ symbol/s. Considering that the grouped layer-parallel architecture uses half as many MACs as the layer-parallel architecture, the throughput per unit silicon area remains the same. The grouped layer-parallel architecture offers a way to scale down the silicon area when the number of users N_t grows.

In the implementation, the input Gram matrix is quantized to 12 b, match-filtered output to 13 b, and the partial interference to 12 b, without deteriorating the error rate performance more than 0.5 dB. The most compute intense block is the interference calculation block in each IPE that performs a 32-wide inner product. The critical paths are dominated by the 12 b×4 b multiplications and accumulations. We optimized the pipeline depth and targeted 425 MHz clock frequency to balance the throughput and power under the area constraints.

Based on place-and-route results, the architectural optimization from the baseline [see Fig. 13(a)] to the grouped layer-parallel [see Fig. 13(c)] architecture reduces the silicon area and power by 4.24× and 2.84×, respectively, at a cost of 2.41× lower throughput.

V. Power Reduction Techniques

The MPD implementation's datapath power is dominated by the 512 MACs, especially the multipliers. To reduce the power consumption, we exploit some unique features of iterative MPD processing. First, being iterative, early termination can be applied to stop processing when convergence criteria are met. Second, the convergence behavior allows the processing to be adapted from early to late iterations.

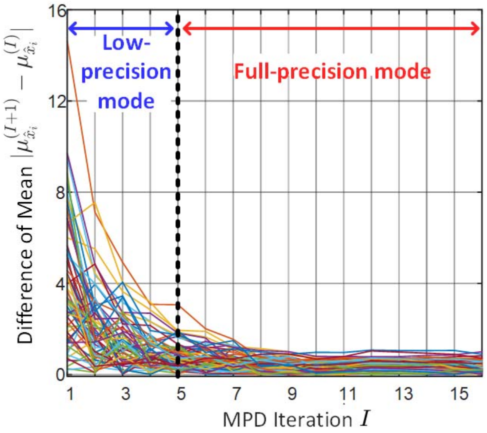


Fig. 6. Convergence of user symbol estimate $\mu_{\hat{x}_i}$ with MPD iterations. The switching from low-precision mode to full-precision mode is set to five MPD iterations in this an example.

A. Adaptive Precision Control

In early iterations, the user symbol estimates are noisy and unstable, and the detector makes coarse symbol estimates, i.e., the symbol updates result in large jumps from one constellation point to another. As iteration progresses, the symbol estimates gradually become more stable, and the detector makes fine-tuning of the symbol estimates. The symbol updates result in small movements near the estimates from the previous iteration. An example of the convergence behavior is illustrated in Fig. 6.

This convergence behavior suggests that the MPD needs only low-precision multiplications in early iterations and full-precision multiplications in late iterations. Therefore, we design the multipliers to support two precision modes: a full-precision mode that supports 12 b × 4 b multiplication and a low-precision mode with the LSBs disabled to support 6 b × 2 b multiplication, as shown in Fig. 7. Compared to the full-precision mode, the low-precision mode saves 75% of the switching activity and dynamic power. The precision switching time is designed to be fine-grained. Detection starts in the low-precision mode. After N_{prec} cycles, the full-precision mode is switched ON. When to switch ON the full-precision mode can affect the BER performance. It is left as a knob that can be set depending on the performance requirement.

B. Clock Gating and Early Termination

Registers are used in the MPD test chip design as data memory to support the wide access required by the parallel architecture. The memory access is deterministic and regular, as shown in Fig. 8, e.g., the 3-Kb marginal IPE memory (M MEM) is only updated once every eight cycles, and the 512-b symbol estimate memory (X MEM) is updated once every two cycles. Therefore, we implement a simple clock gating controller to follow the deterministic timing to turn

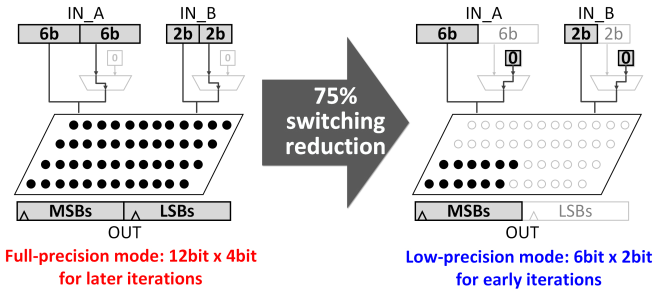


Fig. 7. Multiplier with full-precision and low-precision modes.

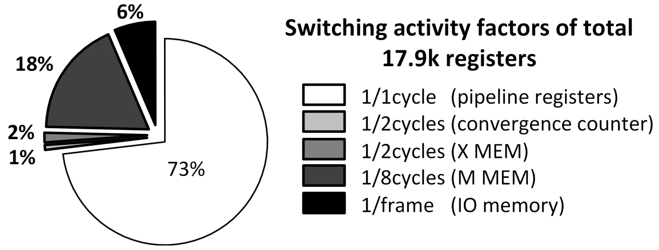


Fig. 8. Register power breakdown and activities.

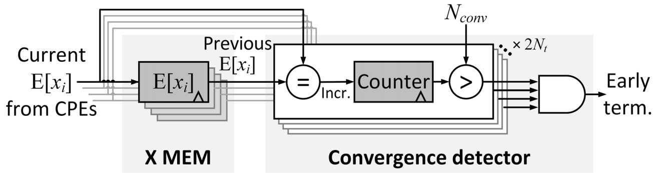


Fig. 9. Implementation of convergence detector and early termination.

off the clock input when the memory is not updated to save dynamic power. The lower the activity, the more the power savings.

In designing the MPD, we add a convergence detector, as shown in Fig. 9, to check if the estimate of a symbol in the current iteration matches the estimate from the previous iteration. If they match, the counter increases. When a symbol estimate remains the same over N_{conv} cycles, we consider that the symbol has converged. If all the symbols in a frame have converged, we consider that the detection converges, and the detection iteration can be early terminated. The value of N_{conv} can be optimally chosen to have a preferable tradeoff between BER performance and power saving from the early termination. The convergence detector is implemented next to the symbol estimate memory (X MEM) by an early termination controller that checks the convergence for each symbol. Early termination can be used to improve throughput or reduce power consumption or both. When used for improving throughput, input and output buffers and a controller are needed to accommodate the varying decoding latency and keep it transparent from the user.

VI. Chip Measurement Results and Comparison

A prototype of the grouped layer-parallel MPD architecture for a 256-QAM 128 × 32 massive MIMO uplink system was designed and fabricated in 40-nm CMOS technology. The microphotograph is shown in Fig. 10 (left). The chip includes a 0.58-mm^2 detector core, a PLL to provide the clock, and a test block with memories and scan chains for storing test vectors and off-chip communication. The PLL is a hard IP that provides a low-jitter and high-frequency on-chip clock for the MPD chip. The PLL enables a reliable on-chip clock source and convenient frequency tuning for chip measurement at a range of clock frequencies. In the detector core, the partial interference calculation block dominates the core area, as shown in Fig. 10 (right). It contains costly multipliers and creates wire congestion hotspots, which leads to a 74% of the core area consumption.

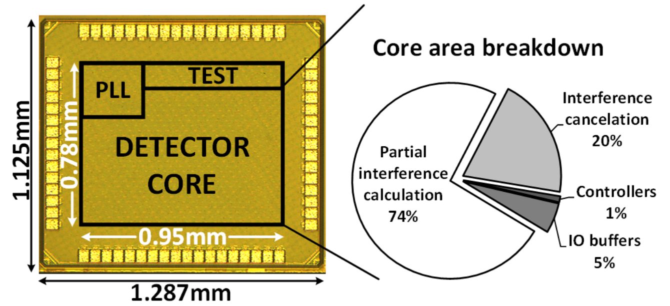


Fig. 10. Left: microphotograph of the 40-nm MPD test chip containing a detector core, a PLL, and a testing block. Right: area breakdown of the detector core.

In our testing setup, we generated the 12-b Gram matrix and the 13-b match-filtered data input offline for different channel realizations, SNRs, and modulation sizes. We assumed OFDM and used a flat Rayleigh fading model to produce the channel and Gram matrix. Input vectors were based on the channel model and match-filtered offline. The values were quantized and fed to the MPD chip through a scan interface. A batch of inputs following the same channel parameters and SNR was generated at a time and loaded on-chip for one batch of continuously testing. The chip was measured to run at a maximum frequency of 425 MHz at the nominal supply voltage of 0.9 V in room temperature (about 20 °C), dissipating 220.6 mW.

A. Chip Measurement Results

Fig. 11 shows the bit-true error rate plot of the SIMO lower bound, linear MMSE, the original floating-point MPD, and the fixed-point symbol-hardening MPD incorporating layered schedule, early termination, and adaptive precision control. For the i.i.d. Rayleigh fading channel shown in Fig. 11(a), we set the maximum iteration $N_{\text{iter}} = 7$, the convergence threshold for early termination $N_{\text{conv}} = 5$, and the adaptive precision threshold $N_{\text{prec}} = 10$. The result shows that our MPD chip incurs only about 0.5-dB SNR loss at BER $< 10^{-4}$ compared to the original floating-point MPD [25]. For the Winner-II channel shown in Fig. 11(b), the MPD chip adopting symbol hardening ($N_m = 1$) has worse performance than MMSE. The MPD performance can be partially recovered by adjusting N_m to 2 or higher at a proportionally higher implementation cost over the symbol-hardening MPD.

Fig. 12 shows the average throughput in Gb/s and energy in pJ/b with voltage and frequency scaling from the nominal supply of 0.9 V to the minimum supply of 0.55 V. Here, the throughput and energy are measured and averaged over the test vectors. With early termination enabled on-chip, detection

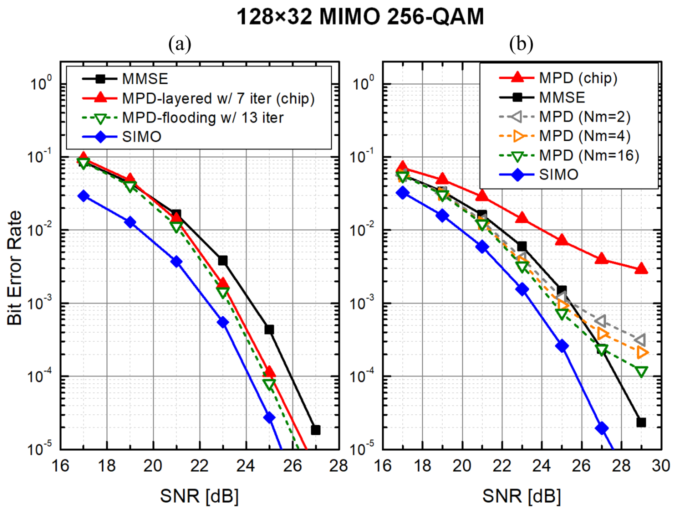


Fig. 11. Uncoded BER of 128 × 32.256-QAM MIMO detections (a) for i.i.d. Rayleigh fading channel using linear MMSE (black), original MPD [25] (green), and symbol-hardening MPD with layer schedule, early termination, and adaptive precision control (red) and (b) for Winner-II channel (urban macro-cell) with the max layout range of 200 m and the UCA128 radius of 50 cm. The performance loss of symbol hardening (red) can be partially recovered by using $N_m = 2$ (gray) at a higher implementation cost.

converges in 5.7, 5.2, and 4.9 iterations on average at an SNR of 23, 25, and 27 dB, respectively. In general, fewer iterations are needed to reach convergence at a higher SNR. Also, at a higher SNR, the symbol estimates are less noisy and stable, which allows the use of low-precision processing to reduce energy consumption. The MPD chip achieves the peak throughput of 2.76 Gb/s, while consuming 79.8 pJ/b.

Fig. 13 shows our MPD chip area, power, throughput, and energy compared to the other architectures presented in Section III. From the fully parallel architecture [see Fig. 13(a)] to the four-layer two-group architecture [see Fig. 13(c)], the area and power are reduced by 4.24× and 2.83×, while the throughput is reduced by 2.41×. However, by adopting layered message-passing scheduling and early termination, our MPD detector [see Fig. 13(d)] converges 2× faster and is terminated early to recoup the throughput loss. Moreover, by incorporating power reduction techniques presented in Section V, the MPD chip achieves a 13.3% power reduction, a 51.6% throughput increase, and a 43.1% energy efficiency improvement compared to the baseline [see Fig. 13(c)].

The measured power-saving improvement is summarized in Table II. The clock gating combined with adaptive precision control saves 6.7% of the total power. By enabling the early termination, redundant switching activities are eliminated, reducing the power consumption further by 13.3%.

B. Comparison to Prior Arts

The results are compared with state-of-the-art MIMO detector chips in Table III. Here, we also include the MPD chip's throughput and area efficiency with and without the early termination to have a fair comparison with other works. Most of the previous approaches, including sphere decoding [11] and MMSE [19], [31], only support up to eight users. These designs are unsuitable for massive MIMO systems because their implementation cost does not scale as the number of users grows. Note that a linear MMSE detector only needs to perform matrix inversion once in every coherence time. The complexity of the matrix inversion part of a linear MMSE detector is dominant, i.e., $O(N_t^3)$, while the filtering part is $O(N_t^2)$. The coherence time needs to be long enough to amortize the cost of the matrix inversion to make it more competitive than an MPD dectector, which does not require matrix inversion.

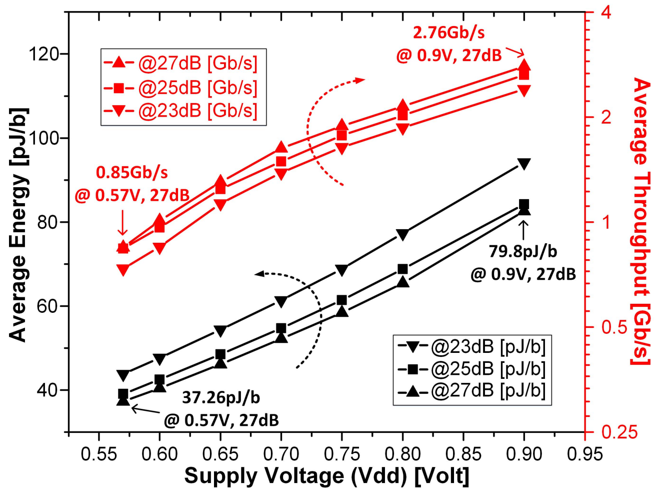


Fig. 12. Measured average throughput (red lines) and average energy (black lines) with voltage scaling at different SNR levels.

TABLE II
POWER SAVINGS OF PROPOSED TECHNIQUES

Techniques	Power	Saving
Baseline (c) in Fig. 13	254.9 mW	0.0%
+ clock gating	249.8 mW	2.0%
+ adaptive precision	237.9 mW	6.7%
+ early termination	220.6 mW	13.3%

Compared to the 28-nm 256-QAM 128 × 8 uplink detector [31] and the 65-nm 64-QAM 128 × 8 uplink detector [19], our 40-nm MPD chip supports a much larger 256-QAM 128 × 32 configuration while providing 1.3×–6.4× higher throughput and 3.0×–8.7× better energy (measured in pJ/b/TX antenna). In [29], the 40-nm MPD chip demonstrates a higher throughput and a better energy efficiency, but it is done by limiting the maximum number of iterations to 2, and it supports only eight users and a QPSK modulation. To support more users and a higher-order modulation, more computation and iterations are needed, which will necessarily lower the throughput and worsen the energy efficiency.

The 28-nm LAMA detector [27] supports 32 users, but it incurs a higher complexity. The throughput of the 28-nm chip is 5.5× lower, and the energy is 5.3× worse compared to our 40-nm chip even without technology normalization. Compared to EPD [26], MPD avoids matrix inversion and has lower complexity. Therefore, EPD is much more compute-intensive than MPD. As a result, it only supports at most 16 users, and in comparison, the MPD chips support twice as many users. Although the EPD chip was designed in a more advanced 28-nm technology, the silicon area is significantly larger, and the throughput is lower. Without technology normalization,

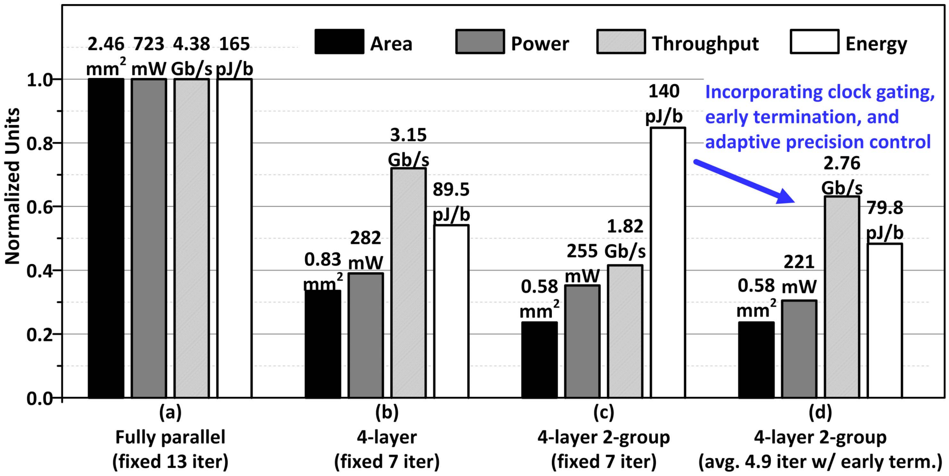


Fig. 13. Design optimization steps. (a) and (b) Place-and-route results. (c) and (d) Measured results. Improvement of (d) is attributed to adaptive precision control, clock gating, and early termination.

TABLE III

COMPARISON OF STATE-OF-THE-ART MIMO DETECTOR DESIGNS

Detector	Liao [11]	Prabhu [31]	Peng [19]	Chen [29]	Tang [26]	Jeon [27]	This Work
Algorithm	SD[a]	MMSE-CHD[b]	MMSE-RCG[c]	MPD	EPD[d]	LAMA[e]	MPD
Performance in correlated channels	Near-optimal	Sub-optimal	Sub-optimal	Limited	Near-optimal	Near-optimal	Limited
MIMO size	8 × 8	128 × 8	128 × 8	128 × 8	128 × 16	256 × 32	128 × 32
QAM size	64	256	64	4	256	256	256
Technology [nm]	130	28	65	40	28	28	40
Core area [mm^2]	1.77	-	-	0.076[f]	2.0	0.37	0.58
Frequency [MHz]	198	300	500	500	569	400	425
Power [mW]	74.8	18	120	77.89	127	151	220.6
Throughput [Gb/s]	0.429	0.300	1.5	8.0[g]	1.80	0.354	1.94[h] - 2.76[i]
Area efficiency [Gb/s/mm^2]	0.242	-	-	105.3[f]	0.9	0.95	3.34 - 4.76
Energy efficiency [pJ/b]	174.3	60	80	9.73	70	426	79.8
Energy efficiency [pJ/b/TX antenna]	21.8	7.5	10	1.22	4.38	13.31	2.49

[a] Sphere decoding. [b] Cholesky decomposition. [c] Recursion conjugate gradient.
[d] Expectation Propagation Detector. [e] Large-MIMO approximate message passing.
[f] IO buffer not included. [g] Maximum number of iterations = 2. [h] Maximum number of iterations = 7 without early termination.
[i] Early termination with average 4.92 iterations and minimal 3.25 iterations at SNR = 27 dB.

the energy efficiency of the 28-nm EPD chip is 70 pJ/b, which is only 1.14× smaller than this 40-nm MPD chip, and supports half as many users. The EPD chip has an area efficiency of only 0.95 Gb/s/mm^2; while the MPD chip is 3.34 Gb/s/mm^2, which is 3.7× higher.

As the simulation results in Fig. 2 show, MPD diverges in correlated channels. The main advantages of MPD are its high throughput, low power, small area, and superior energy efficiency. However, MPD's use is limited to uncorrelated channels. For correlated channels, EPD [26] and LAMA [27] would be more suitable. This result highlights the need for a dual-mode detection scheme depending on the channels: MPD for uncorrelated channels to achieve more efficient, high-performance detection; and EPD for correlated channels, which costs higher energy and lower throughput. We also propose MPD to be used as a pre-processor to produce the initial starting point for an elaborate detection, such as EPD, or search-based detection to speed up convergence.

VII. CONCLUSION

We demonstrate a 0.58-mm^2 MPD test chip for a 256-QAM 128 × 32 massive MIMO uplink detector. With the proposed symbol hardening approximation, the complexity is reduced by more than 60%. The detector implements a pipelined grouped layer-parallel architecture using a layered schedule to accelerate convergence, enabling an average throughput of 2.76 Gb/s (running, on average, 4.92 iterations with early termination) at 220.6 mW. The chip incorporates adaptive precision control and clock gating to improve energy efficiency further by up to 43%. Compared to the state-of-the-art 28-nm massive MIMO uplink detector, this design provides 1.7× higher energy efficiency per TX antenna and 3.7× higher area efficiency.

REFERENCES

[1] T. Marzetta, "Noncooperative cellular wireless with unlimited numbers of base station antennas," *IEEE Trans. Wireless Commun.*, vol. 9, no. 11, pp. 3590–3600, Nov. 2010.

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

[3] H. Q. Ngo, E. G. Larsson, and T. L. Marzetta, "Energy and spectral efficiency of very large multiuser MIMO systems," *IEEE Trans. Commun.*, vol. 61, no. 4, pp. 1436–1449, Apr. 2013.

[4] J. Hoydis, S. ten Brink, and M. Debbah, "Massive MIMO in the UL/DL of cellular networks: How many antennas do we need?" *IEEE J. Sel. Areas Commun.*, vol. 31, no. 2, pp. 160–171, Feb. 2013.

[5] E. Bjornson, J. Hoydis, M. Kountouris, and M. Debbah, "Massive MIMO systems with non-ideal hardware: Energy efficiency, estimation, and capacity limits," *IEEE Trans. Inf. Theory*, vol. 60, no. 11, pp. 7112–7139, Nov. 2014.

[6] E. Björnson, E. G. Larsson, and T. L. Marzetta, "Massive MIMO: Ten myths and one critical question," *IEEE Commun. Mag.*, vol. 54, no. 2, pp. 114–123, Feb. 2016.

[7] 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.

[8] A. Burg, M. Borgmann, M. Wenk, M. Zellweger, W. Fichtner, and H. Bolcskei, "VLSI implementation of MIMO detection using the sphere decoding algorithm," *IEEE J. Solid-State Circuits*, vol. 40, no. 7, pp. 1566–1577, Jul. 2005.

[9] B. Hassibi and H. Vikalo, "On the sphere-decoding algorithm I. Expected complexity," *IEEE Trans. Signal Process.*, vol. 53, no. 8, pp. 2806–2818, Aug. 2005.

[10] Z. Guo and P. Nilsson, "Algorithm and implementation of the K-best sphere decoding for MIMO detection," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 3, pp. 491–503, Mar. 2006.

[11] C. Liao, T. Wang, and T. Chiueh, "A 74.8 mW soft-output detector IC for 8×8 spatial-multiplexing MIMO communications," *IEEE J. Solid-State Circuits*, vol. 45, no. 2, pp. 411–421, Feb. 2010.

[12] S. Chen, T. Zhang, and Y. Xin, "Relaxed k-best MIMO signal detector design and VLSI implementation," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 15, no. 3, pp. 328–337, Mar. 2007.

[13] X. Chen, G. He, and J. Ma, "VLSI implementation of a high-throughput iterative fixed-complexity sphere decoder," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 60, no. 5, pp. 272–276, May 2013.

[14] S. Yang and L. Hanzo, "Fifty years of MIMO detection: The road to large-scale MIMOs," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 1941–1988, Sep. 2015.

[15] L. Dai, X. Gao, X. Su, S. Han, C.-L. I, and Z. Wang, "Low-complexity soft-output signal detection based on Gauss–Seidel method for uplink multiuser large-scale MIMO systems," *IEEE Trans. Veh. Technol.*, vol. 64, no. 10, pp. 4839–4845, Oct. 2015.

[16] C. Zhang, Z. Wu, C. Studer, Z. Zhang, and X. You, "Efficient soft-output Gauss-Seidel data detector for massive MIMO systems," *IEEE Trans. Circuits Syst. I, Reg. Papers*, early access, Oct. 26, 2019, doi: 10.1109/TCSI.2018.2875741.

[17] 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.*, Dec. 2014, pp. 3696–3701.

[18] B. Yin, M. Wu, J. R. Cavallaro, and C. Studer, "VLSI design of large-scale soft-output MIMO detection using conjugate gradients," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2015, pp. 1498–1501.

[19] G. Peng, L. Liu, Q. Wei, Y. Wang, S. Yin, and S. Wei, "A 2.69 Mbps/mW 1.09 Mbps/kGE conjugate gradient-based MMSE detector for 64-QAM 128×8 massive MIMO systems," in *Proc. IEEE Asian Solid-State Circuits Conf. (A-SSCC)*, Nov. 2018, pp. 191–194.

[20] X. Gao, L. Dai, Y. Ma, and Z. Wang, "Low-complexity near-optimal signal detection for uplink large-scale MIMO systems," *Electron. Lett.*, vol. 50, no. 18, pp. 1326–1328, Aug. 2014.

[21] T. Xie, L. Dai, X. Gao, X. Dai, and Y. Zhao, "Low-complexity SSOR-based precoding for massive MIMO systems," *IEEE Commun. Lett.*, vol. 20, no. 4, pp. 744–747, Apr. 2016.

[22] B. Yin, M. Wu, G. Wang, C. Dick, J. R. Cavallaro, and C. Studer, "A 3.8Gb/s large-scale MIMO detector for 3GPP LTE-advanced," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2014, pp. 3879–3883.

[23] M. Wu, B. Yin, G. Wang, C. Dick, J. R. Cavallaro, and C. Studer, "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.

[24] S. Wu, L. Kuang, Z. Ni, J. Lu, D. Huang, and Q. Guo, "Low-complexity iterative detection for large-scale multiuser MIMO-OFDM systems using approximate message passing," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 902–915, Oct. 2014.

[25] T. L. Narasimhan and A. Chockalingam, "Channel hardening-exploiting message passing (CHEMP) receiver in large-scale MIMO systems," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 847–860, Oct. 2014.

[26] W. Tang, H. Prabhu, L. Liu, V. Öwall, and Z. Zhang, "A 1.8 Gb/s 70.6 pJ/b 128×16 link-adaptive near-optimal massive MIMO detector in 28nm UTBB-FDSOI," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2018, pp. 224–226.

[27] C. Jeon, O. Castañeda, and C. Studer, "A 354 Mb/s 0.37 mm^2 151 mW 32-user 256-QAM near-MAP soft-input soft-output massive MU-MIMO data detector in 28 nm CMOS," *IEEE Solid-State Circuits Lett.*, vol. 2, no. 9, pp. 127–130, Sep. 2019.

[28] P. Kyosti *et al.*, "WINNER II channel models. D1.1.2 V1.2," 2007. [Online]. Available: https://ieeexplore.ieee.org/document/8877927/references#references

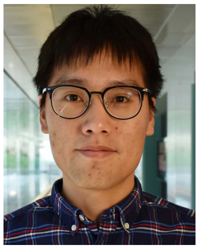
[29] Y.-T. Chen, C.-C. Cheng, T.-L. Tsai, W.-C. Sun, Y.-L. Ueng, and C.-H. Yang, "A 501 mW 7.6lGb/s integrated message-passing detector and decoder for polar-coded massive MIMO systems," in *Proc. Symp. VLSI Circuits*, Jun. 2017, pp. C330–C331.

[30] D. E. Hocevar, "A reduced complexity decoder architecture via layered decoding of LDPC codes," in *Proc. IEEE Workshop Signal Process. Syst. (SIPS)*, Oct. 2004, pp. 107–112.

[31] H. Prabhu, J. N. Rodrigues, L. Liu, and O. Edfors, "3.6 A 60 pJ/b 300 Mb/s 128×8 massive MIMO precoder-detector in 28 nm FD-SOI," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2017, pp. 60–61.

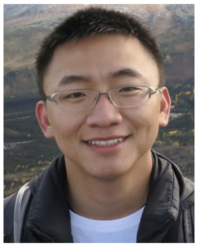
[32] X. Gao, O. Edfors, F. Rusek, and F. Tufvesson, "Massive MIMO performance evaluation based on measured propagation data," *IEEE Trans. Wireless Commun.*, vol. 14, no. 7, pp. 3899–3911, Jul. 2015.

[33] G. Peng, L. Liu, S. Zhou, S. Yin, and S. Wei, "A 1.58 Gbps/W 0.40 Gbps/mm^2 ASIC implementation of MMSE detection for 128×8 64-QAM massive MIMO in 65 nm CMOS," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 65, no. 5, pp. 1717–1730, May 2018.



Wei Tang (Member, IEEE) received the B.S. degree from National Chiao Tung University, Hsinchu, Taiwan, in 2011, and the M.S. and Ph.D. degrees in electrical engineering from the University of Michigan, Ann Arbor, MI, USA, in 2019.

He was a Visiting Ph.D. Student with Lund University, Lund, Sweden, and a Graduate Research Intern with Intel Labs, Santa Clara, CA, USA. He is currently a Research Fellow with the Department of Electrical Engineering and Computer Science, University of Michigan. His research interests are in the high-speed, energy-efficient detector and forward error correction (FEC) decoder designs for small-scale multiple-input–multiple-output (MIMO) and massive MIMO systems.



Chia-Hsiang Chen (Member, IEEE) received the B.S. degree from National Chiao Tung University, Hsinchu, Taiwan, in 2008, and the M.S. and Ph.D. degrees in electrical engineering from the University of Michigan, Ann Arbor, MI, USA, in 2012 and 2014, respectively.

From 2015 to 2017, he joined Intel Labs, Santa Clara, CA, USA, with a focus on architecture and systems for low-power and wireless communication. He is currently with the Apple Wireless Connectivity Group, Cupertino, CA, USA.

Zhengya Zhang (Senior Member, IEEE) received the B.A.Sc. degree in computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2003, and the M.S. and Ph.D. degrees in electrical engineering from the University of California at Berkeley (UC Berkeley), Berkeley, CA, USA, in 2005 and 2009, respectively.

He has been a Faculty Member with the University of Michigan, Ann Arbor, MI, USA, since 2009, where he is currently an Associate Professor with the Department of Electrical Engineering and Computer Science. His research interests include low-power and high-performance VLSI circuits and systems for computing, communications, and signal processing.

Dr. Zhang was a recipient of the David J. Sakrison Memorial Prize from UC Berkeley in 2009, the National Science Foundation CAREER Award in 2011, the Intel Early Career Faculty Award in 2013, and the University of Michigan College of Engineering Neil Van Eenam Memorial Award in 2019. He was an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: REGULAR PAPERS from 2013 to 2015 and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS from 2014 to 2015. He has been an Associate Editor of the IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS since 2015. He has been serving on the Technical Program Committees of the Symposium on VLSI Circuits and the IEEE Custom Integrated Circuits Conference (CICC) since 2018.