# In-Memory Computing Based Ultra-Efficient Massive MIMO Precoding: Memristor Crossbar Circuits, Conductance Mapping Strategies, and Programming Latency Estimation

Yu-Xin Zhang, Shaoshi Yang, *Senior Member, IEEE*, Yi-Hang Ren, Sheng Chen, *Life Fellow, IEEE*, and Ping Zhang, *Fellow, IEEE*

***Abstract*—Matrix inversion in massive multiple-input multiple-output (MIMO) precoding imposes significant burden on the energy efficiency and processing latency of baseband units. In this paper, we first propose a memristor crossbar based in-memory computing circuit capable of supporting both zero-forcing (ZF) and minimum mean square error (MMSE) precoding. The circuit features a reduced matrix size and enables faster one-step computation without the need for timing control. Secondly, to address the computational inaccuracy caused by the limited conductance range of memristors, we develop an optimized matrix-to-conductance mapping scheme that jointly considers device physical constraints and matrix statistics, achieving over 60% reduction in relative computation error compared with baseline scheme. An associated lightweight circuit enhancement ensures compatibility with practical crossbar architectures, without incurring significant hardware overhead. Thirdly, we establish a memristor programming time model grounded in device-level potentiation and depression dynamics. The analysis yields closed-form expressions for the expected programming time and its upper bound, and is further validated through Monte Carlo simulations, enabling accurate estimation of the system throughput. Simulation results demonstrate that the proposed circuit achieves a bit error rate comparable to that of 64-bit floating-point precoding, while delivering over 100× improvement in both energy and area efficiency compared with the NVIDIA RTX A2000 graphics processing unit (GPU).**

***Index Terms*—Massive MIMO, precoding, ZF and MMSE integration, memristor crossbar, in-memory computing.**

## I. Introduction

MASSIVE multiple-input multiple-output (MIMO) technology has become a key feature of modern cellular communication systems, as exemplified by the 5G cellular networks [1]. Compared with conventional small-scale MIMO systems, massive MIMO offers superior interference mitigation capabilities and higher spectral efficiency. In the 6G mobile networks, massive MIMO is expected to scale up to hundreds or even thousands of antennas to further enhance system performance.

In time-division duplexing (TDD) systems, precoding techniques utilize uplink channel state information (CSI) to preprocess downlink signals, effectively reducing inter-user interference [2]. Common linear precoding algorithms include maximum ratio transmission (MRT), zero-forcing (ZF) precoding, and minimum mean square error (MMSE) precoding. Notably, ZF and MMSE algorithms involve matrix inversion operations with cubic computational complexity [3], leading to significant increases in processing latency and energy consumption in the baseband units of massive MIMO systems.

The emergence of memristor crossbars offers a promising approach to accelerating matrix operations. Memristors can perform matrix-vector multiplications (MVM) with in-memory computing architecture [4], leveraging highly parallel processing to substantially reduce computational complexity, energy consumption, and data transfer latency between memory and processors [5].

Previous studies explored the use of memristor crossbar circuits to implement ZF and regularized ZF algorithms based on ridge regression [6]. Later research designed memristor-accelerated ZF precoding with timing control, where the matrix inversion (INV) and matrix-vector multiplication processes are separated and executed sequentially [7]. Efforts also focused on employing memristor crossbars for high-parallelism maximum likelihood (ML) detection in massive MIMO systems [8]. More recently, hybrid analog-digital computing architectures were introduced to develop memristor-based circuits for successive interference cancellation (SIC) detection in massive MIMO systems [9]. To further enhance robustness against conductance deviations, an amplifier-augmented detector architecture was proposed to decouple the processing of large-scale and small-scale fading matrices, achieving improved bit error rate (BER) with negligible power overhead increase [10]. A follow-up work further introduced refined

This work was supported by National Natural Science Foundation (Grant No. 62550173) and Beijing Municipal Natural Science Foundation (Grant No. L242013). *(Corresponding author: Shaoshi Yang.)*

Y.-X. Zhang is with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China, with the Key Laboratory of Mathematics and Information Networks, Ministry of Education, Beijing 100876, China, and also with the Department of IoT Technologies and Applications, China Mobile Research Institute, Beijing 100053, China (e-mail: yuxin.zhang@bupt.edu.cn).

S. Yang and Y.-H. Ren are with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China, and also with the Key Laboratory of Mathematics and Information Networks, Ministry of Education, Beijing 100876, China (e-mails: shaoshi.yang@bupt.edu.cn, renyihang@bupt.edu.cn).

P. Zhang is with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China, and also with the State Key Laboratory of Networking and Switching Technology, Beijing 100876, China (e-mail: pzhang@bupt.edu.cn).

S. Chen is with the School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK (e-mail: sqc@ecs.soton.ac.uk).

mathematical modeling and conductance mapping schemes to enhance detection accuracy under memristor deviations [11].

Despite these advancements, these prior works have not provided a detailed assessment of how memristor non-idealities affect precoding performance. For instance, the work [7] focused on overall BER trends under noise, but did not explicitly isolate the impact of conductance quantization or programming uncertainty. Similarly, the study [6] emphasized errors under varying amplifier gain and memory precision, while only offering very limited analysis on how circuit performance is affected by quantization constraints or practical conductance ranges in low-bit precision regimes.

Moreover, few studies have systematically assessed how memristor non-idealities, such as limited switching ratio (the ratio of high resistance state to low resistance state) and programming variability, affect the performance of precoding circuits. Notably, the work [12] proposed a resistive random access memory (RRAM) based baseband processor for MIMO orthogonal frequency-division multiplexing (OFDM) systems and introduced a model to estimate the row-wise programming time of memristor crossbars. Their analysis reveals that the latency scales sublinearly with the number of transmit antennas. While their work marks a valuable step toward programming time modeling, it adopts simplified conductance-programming step model and focuses mainly on asymptotic upper-bound estimation, without considering the impact of realistic programming behavior on the latency.

To bridge these gaps, we propose a memristor crossbar circuit that supports both ZF and MMSE precoding. The proposed circuit reduces energy consumption by minimizing the crossbar size and peripheral circuitry, and it enables fast one-step computation without requiring timing control. To complement the circuit design, we further develop a probabilistic model, supported by both theoretical analysis and simulation results, to more accurately evaluate the memristor programming time under realistic device behaviors, such as potentiation and depression dynamics.

These contributions bridge the modeling gap between physical device constraints and system-level performance, enabling more accurate design trade-offs for future memristor-based massive MIMO baseband processors. The main contributions of this paper are summarized as follows.

- We propose a one-step memristor crossbar circuit that supports both ZF and MMSE precoding, featuring a reduced array size and simplified peripheral circuitry. By eliminating the need for timing control, the circuit further accelerates the processing speed through efficient one-step computation.
- To mitigate the degradation of computational accuracy caused by the limited switching ratio of memristors, we develop an optimized matrix-to-conductance mapping scheme that jointly considers the physical constraints of memristors and the statistical properties of the precoding matrix. Furthermore, the high conductance values assigned to diagonal elements, as dictated by the proposed mapping scheme, motivate a resistor-parallel circuit architecture selectively applied to diagonal memristor cells via metal-oxide-semiconductor field-effect transistor (MOSFET) regulated switching, thereby minimizing hardware overhead. The proposed mapping scheme achieves over 60% reduction in relative computation error compared with the baseline linear mapping scheme, demonstrating superior robustness under various memristor physical constraints.
- We propose a probabilistic model for estimating the memristor programming time, which accounts for the potentiation and depression characteristics of memristors. Based on this model, we derive closed-form expressions for the statistical expectation of programming time and its upper bound, thereby providing theoretical support for the time estimation. The consistency between the theoretical predictions and simulation results validates the accuracy of the proposed model. This estimation scheme offers a reliable approach for the quantitative analysis of circuit throughput, energy efficiency and area efficiency. Based on the theoretical analysis and simulation results, the throughput is comparable to the advanced commercial processors and both the energy and area efficiency of our design is 100 times higher than these processors.

## II. Preliminaries

### A. System Model

The transmitter considered in this paper is located at the base station (BS) of a massive MIMO system operating in the downlink scenario. The channel matrix $\widetilde{\mathbf{H}} \in \mathbb{C}^{N_{\mathrm{rx}} \times N_{\mathrm{tx}}}$ is used to characterize the channel gains between the BS and the user equipment (UE), where $N_{\mathrm{tx}}$ denotes the number of transmit antennas at the BS and $N_{\mathrm{rx}}$ denotes the number of receive antennas at the UE. The tilde symbol, e.g., $\widetilde{\mathbf{H}}$, is used to indicate complex-valued variables throughout the paper. Each element $\widetilde{h}_{n_{\mathrm{rx}},n_{\mathrm{tx}}}$ of the channel matrix $\widetilde{\mathbf{H}}$ represents the subchannel gain between the $n_{\mathrm{tx}}$-th transmit antenna at the BS and the $n_{\mathrm{rx}}$-th receive antenna at the UE.

Under ideal conditions, where the distance between transmit and receive antennas is sufficiently large and the channel environment is highly random, the subchannel gains in the MIMO system can be assumed to be uncorrelated. Such channel conditions can be modeled by an independent and identically distributed Rayleigh fading channel, which is $\widetilde{h}_{n_{\mathrm{rx}},n_{\mathrm{tx}}} \sim \mathcal{CN}(0,1)$.

The received signals at the $N_{\mathrm{rx}}$ antennas of the UE are given by:

$$\widetilde{\mathbf{y}} = \sqrt{\rho_{\mathrm{T}}}\widetilde{\mathbf{H}}\widetilde{\mathbf{x}} + \widetilde{\mathbf{n}}, \tag{1}$$

where $\sqrt{\rho_{\mathrm{T}}}$ denotes the transmit power factor, $\widetilde{\mathbf{x}}$ represents the transmitted complex signal vector from the $N_{\mathrm{tx}}$ antennas, and $\widetilde{\mathbf{n}} \sim \mathcal{CN}(0, \sigma_n^2 \mathbf{I}_{N_{\mathrm{rx}}})$ denotes the additive white Gaussian noise with $\mathbf{I}_{N_{\mathrm{rx}}}$ being the $N_{\mathrm{rx}} \times N_{\mathrm{rx}}$ identity matrix.

### B. Precoding Algorithms

The linear precoding operation is given by $\widetilde{\mathbf{x}} = \widetilde{\mathbf{W}}\widetilde{\mathbf{s}}$, where $\widetilde{\mathbf{W}}$ denotes the precoding matrix and $\widetilde{\mathbf{s}}$ denotes the transmitted symbol vector.

ZF precoding aims to direct the transmitted signal energy toward the intended user while minimizing inter-antenna interference by forcing the signals in other directions to near zero [13]. The ZF precoding matrix is given by:

$$\widetilde{\mathbf{W}} = \gamma \widetilde{\mathbf{H}}^H (\widetilde{\mathbf{H}} \widetilde{\mathbf{H}}^H)^{-1}, \tag{2}$$

where $\gamma$ is a scalar factor ensuring the transmit power constraint.

MMSE precoding aims to minimize the mean square error between the transmitted and received signals by considering both inter-antenna interference and noise [14]. The MMSE precoding matrix is given by:

$$\widetilde{\mathbf{W}} = \gamma \widetilde{\mathbf{H}}^H (\widetilde{\mathbf{H}} \widetilde{\mathbf{H}}^H + \lambda \mathbf{I}_{N_{\mathrm{rx}}})^{-1}, \tag{3}$$

where $\lambda = \frac{\sigma_s^2}{\sigma_n^2}$ is the ratio of signal power to noise power, which can be regarded as a regularization parameter.

Considering the physical constraints that conductance, current, and voltage are real-valued in memristor crossbar circuits, complex-valued vectors/matrices must be converted into real-valued forms, in order to perform computation within memristor crossbar circuits. For vector $\widetilde{\mathbf{v}}$ and matrix $\widetilde{\mathbf{M}}$, their corresponding real-valued forms are given by

$$\mathbf{\Omega}_{\mathbf{v}} \triangleq \left[\Re(\widetilde{\mathbf{v}})^T \ \Im(\widetilde{\mathbf{v}})^T\right]^T, \ \mathbf{\Omega}_{\mathbf{M}} \triangleq \begin{bmatrix} \Re(\widetilde{\mathbf{M}}) & -\Im(\widetilde{\mathbf{M}}) \\ \Im(\widetilde{\mathbf{M}}) & \Re(\widetilde{\mathbf{M}}) \end{bmatrix}, \tag{4}$$

where $\Re(\cdot)$ and $\Im(\cdot)$ denote the real and imaginary parts, respectively. We also define $\boldsymbol{x}^+ \triangleq \max\{\boldsymbol{x}, 0\}$ and $\boldsymbol{x}^- \triangleq -\min\{\boldsymbol{x}, 0\}$.

### C. Basic Characteristics of Memristors

The minimum and maximum available conductance values of the memristor are denoted by $G_{\min}$ and $G_{\max}$. By applying appropriate electrical excitation, the conductance can be adjusted to a target value $G_{\mathrm{target}} \in [G_{\min}, G_{\max}]$. The actual conductance is denoted as $G_{\mathrm{final}}$. Experimental results have shown that the conductance variation follows a zero-mean Gaussian distribution with a variance that remains constant regardless of the conductance value [15]. To simplify the analysis, the memristor conductance variation is modeled as $G_{\mathrm{final}} = \delta G_{\mathrm{target}}$, where $\delta$ denotes the conductance variation factor and $\delta \sim \mathcal{N}(1, \sigma_{\mathrm{norm}}^2)$. The normalized standard deviation is given by $\sigma_{\mathrm{norm}} = \epsilon_{\mathrm{p}}/G_{\mathrm{target}}$, where $\epsilon_{\mathrm{p}}$ represents the small residual error after programming process, and it has the same unit as $G_{\mathrm{target}}$.

The conductance resolution is represented by bit precision, which specifies the number of resolvable conductance levels, and it is influenced by the device variation during programming process. In practical implementations, considering the limitation of programming time, the programming process relies on a predefined lookup table, and the target conductance $G_{\text{target}}$ is restricted to a discrete set of values rather than being continuously tunable. Typically, the conductance range is divided into $L = 2^{\epsilon_{\mathrm{bp}}}$ levels [16], and $\epsilon_{\mathrm{bp}}$ denotes the bit precision of the memristor.

### D. Basic Circuit Theory for Memristor Crossbar

**Matrix-Vector Multiplication:** The MVM circuit executes the operation $\mathbf{M} \cdot \mathbf{s} = \mathbf{n}$. Its core component is a memristor crossbar consisting of $i$ word lines (rows) and $j$ bit lines (columns). The conductance at each crosspoint forms the conductance matrix $\mathbf{G} \in \mathbb{R}^{i \times j}$, which is linearly mapped from $\mathbf{M} \in \mathbb{R}^{i \times j}$. According to Kirchhoff's current law, the relationship between the output current vector $\mathbf{i}$, the conductance matrix $\mathbf{G}$, and the input voltage vector $\mathbf{v}$ is given by:

$$\mathbf{i} = \mathbf{G} \cdot \mathbf{v}. \tag{5}$$

**Matrix Inversion:** The matrix inversion circuit shares a similar structure with the MVM circuit but operates on square matrices. By applying Kirchhoff's laws, the relationship is given as:

$$-\mathbf{G}^{-1} \cdot \mathbf{i} = \mathbf{v}. \tag{6}$$

This circuit can also be regarded as a linear equation solver, capable of computing $\mathbf{M}^{-1} \cdot \mathbf{s} = \mathbf{n}$ in a single step, which is particularly valuable for channel inversion in linear precoding.

## III. Proposed Memristor Crossbar-Based MIMO ZF & MMSE Precoder Circuit

The circuit consists of two main components: the INV component (including $\mathbf{N}_{\mathrm{INV}}$, $\mathbf{D}_{\mathrm{mem}}$ and $\mathbf{P}_{\mathrm{INV}}$) and the MVM component (including $\mathbf{N}_{\mathrm{MVM}}$ and $\mathbf{P}_{\mathrm{MVM}}$), as illustrated in Fig. 1. In the INV component, the diagonal matrix $\mathbf{D}_{\mathrm{mem}}$ can be regarded as the sum of a balancing matrix and a regularization matrix. The balancing matrix addresses the conductance imbalance caused by the diagonally dominant matrix $\widetilde{\mathbf{H}}\widetilde{\mathbf{H}}^H$, while the regularization matrix maps the regularization term ($\lambda\mathbf{I}$) in the MMSE algorithm. In the MVM component, the operational transconductance amplifiers (OTAs) convert the output voltage $\mathbf{v}_{\mathrm{x}}$ to current feedback. By applying Ohm's law and Kirchhoff's laws to the circuit shown in Fig. 1, the following equation is obtained:

$$(\mathbf{P}_{\mathrm{INV}} - \mathbf{N}_{\mathrm{INV}} + \mathbf{D}_{\mathrm{mem}})^{-1} \mathbf{i}_{\mathrm{s}} = (\mathbf{P}_{\mathrm{MVM}} - \mathbf{N}_{\mathrm{MVM}})^{-1} \mathbf{v}_{\mathrm{x}}. \tag{7}$$

Denote $\widetilde{\mathbf{Z}} = \widetilde{\mathbf{H}}\widetilde{\mathbf{H}}^{\mathrm{H}}$ and construct the matrix $\mathbf{\Omega}_{\mathrm{M}} = \frac{2\mathbf{\Omega}_{\mathrm{Z}}}{N_{\mathrm{tx}}} - 2\mathbf{I}_{2N_{\mathrm{rx}}}$. Then the proposed circuit mapping scheme is given in (8) at the bottom of this page, where the factor $\alpha$ is employed to establish the correspondence between matrix elements and memristor conductance values.

For $\lambda = 0$, the proposed circuit performs the ZF precoding algorithm, while for $\lambda \neq 0$, the proposed circuit performs the MMSE precoding algorithm.

$$\begin{gathered} \mathbf{i}_{\mathrm{s}} = -\mathbf{\Omega}_{\mathrm{s}}, \quad \mathbf{v}_{\mathrm{x}} = \mathbf{\Omega}_{\mathrm{x}}, \quad \mathbf{D}_{\mathrm{mem}} = \alpha \left(2 + \frac{2\lambda}{N_{\mathrm{tx}}}\right) \mathbf{I}_{2N_{\mathrm{rx}}}, \quad \mathbf{P}_{\mathrm{INV}} = \alpha\, \mathbf{\Omega}_{\mathrm{M}}^{+}, \\ \mathbf{N}_{\mathrm{INV}} = \alpha\, \mathbf{\Omega}_{\mathrm{M}}^{-}, \quad \mathbf{P}_{\mathrm{MVM}} = \alpha \frac{2\,\mathbf{\Omega}_{\mathrm{H^H}}^{+}}{N_{\mathrm{tx}}}, \quad \mathbf{N}_{\mathrm{MVM}} = \alpha \frac{2\,\mathbf{\Omega}_{\mathrm{H^H}}^{-}}{N_{\mathrm{tx}}}, \end{gathered} \tag{8}$$

$$\mathbf{i}_{\mathrm{s}} = -\frac{\mathbf{\Omega}_{\mathrm{s}}}{\kappa}, \quad \mathbf{v}_{\mathrm{x}} = \mathbf{\Omega}_{\mathrm{x}}, \quad \mathbf{D}_{\mathrm{mem}} = \alpha\left(N_{\mathrm{d}} + \frac{\lambda}{r_{\mathrm{ds}}}\right)\mathbf{I}_{2N_{\mathrm{rx}}}, \quad \mathbf{P}_{\mathrm{INV}} = \alpha\left(\frac{\mathbf{\Omega}_{\mathrm{Z}}}{r_{\mathrm{ds}}} - N_{\mathrm{d}}\,\mathbf{I}_{2N_{\mathrm{rx}}}\right)^{+},$$
$$\mathbf{N}_{\mathrm{INV}} = \alpha\left(\frac{\mathbf{\Omega}_{\mathrm{Z}}}{r_{\mathrm{ds}}} - N_{\mathrm{d}}\,\mathbf{I}_{2N_{\mathrm{rx}}}\right)^{-}, \quad \mathbf{P}_{\mathrm{MVM}} = \frac{\kappa\,\mathbf{\Omega}^{+}_{\mathrm{H^H}}}{r_{\mathrm{ds}}}, \quad \mathbf{N}_{\mathrm{MVM}} = \frac{\kappa\,\mathbf{\Omega}^{-}_{\mathrm{H^H}}}{r_{\mathrm{ds}}}, \tag{9}$$

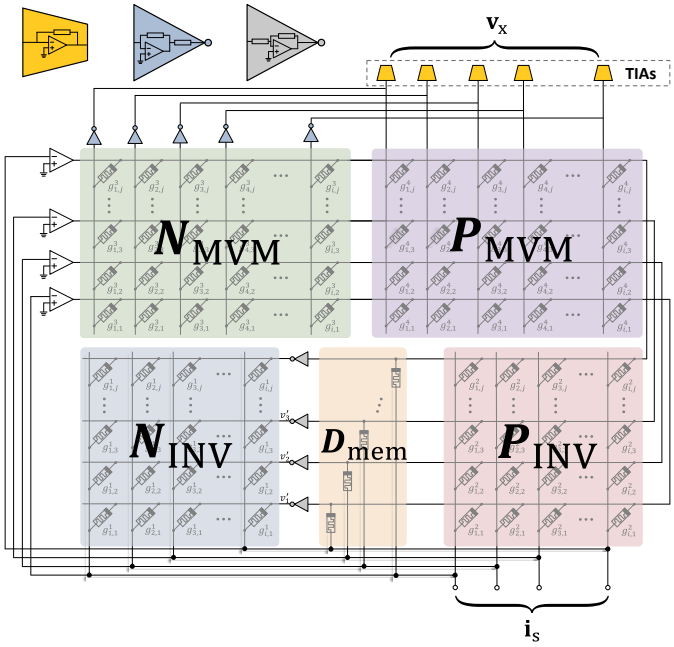


Fig. 1. Proposed memristor crossbar-based MIMO ZF & MMSE precoder circuit.

## IV. Optimized Mapping Scheme for MIMO Precoder Circuit

It is widely acknowledged that memristors cannot directly represent negative values, and the adopted positive-negative separation scheme results in approximately 50% of the target conductance values being zero. Consequently, limited by the memristor switching ratio, different mapping scales can significantly affect computational errors. In fact, through a series of appropriate linear scaling operations, the proposed circuit retains the capability to accurately perform the intended computation. Based on the circuit structure, a more general mapping expression can be derived as (9) at the top of the next page, where the factor $\kappa$ has the similar role to $\alpha$, $r_{\mathrm{ds}} = \frac{N_{\mathrm{tx}}}{N_{\mathrm{d}}}$ represents the matrix scaling factor, and $N_{\mathrm{d}}$ denotes the scalar that corresponds to the identical diagonal entries of the balancing matrix.

We simulate the computation errors under different mapping ratios and maximum conductance to verify the effectiveness of the proposed mapping scheme. In the simulation, $\alpha$ is set to $100\,\mu$S, $\kappa$ is set to $\frac{r_{\mathrm{ds}}G_{\max}}{2\sqrt{2}}$, $G_{\max}$ is expressed in unit of $100\,\mu$S, $\mathbf{i}_{\mathrm{s}}$ is expressed in milliamperes, and $\mathbf{v}_{\mathrm{x}}$ is expressed in millivolts. The simulation results are shown in Fig. 2(a), which indicate that the relative computation errors of the proposed circuit decreases as $N_{\mathrm{d}}$ increases within a certain range. However, when $N_{\mathrm{d}}$ exceeds this range, the relative error increases rapidly. This is because limited by the conductance range of the memristor, an excessively large $N_{\mathrm{d}}$ significantly raises the probability of conductance exceeding the conductance range, leading to computational inaccuracies. As $G_{\max}$ increases, the feasible range of $N_{\mathrm{d}}$ also expands, which facilitates a further reduction in computation error. Therefore, when selecting the mapping ratio, a trade-off should be considered based on the characteristics of the memristor device. The simulation results demonstrate that the optimized mapping scheme reduces the relative computation error by more than 60% compared to the baseline approaches ($N_{\mathrm{d}} = 2$) in [7], demonstrating superior robustness across various memristor physical constraints.

To ensure that the probability of conductance exceeding the limitation range of memristor conductance remains below 0.3%, the theoretical optimal value of $N_{\mathrm{d}}$ can be determined through calculation, and the optimal value $N_{\mathrm{d}}^{*}$ is given by

$$N_{\mathrm{d}}^{*} = \xi \cdot \frac{\sqrt{2N_{\mathrm{tx}}}}{3} \cdot G_{\max}, \tag{10}$$

where $\xi$ is an experimentally determined coefficient to compensate for the random disturbances, such as residual error $\boldsymbol{\epsilon}_p$, that affect circuit performance. In the simulation, $\xi$ is set to 0.8.

To further validate the robustness of the proposed mapping scheme under spatially correlated MIMO channels, additional simulations are conducted based on the Kronecker channel model, where the channel matrix is given by $\widetilde{\mathbf{H}} = \mathbf{R}_{\mathrm{r}}^{1/2}\widetilde{\mathbf{W}}\mathbf{R}_{\mathrm{t}}^{1/2}$. In this model, $\widetilde{\mathbf{W}}$ denotes a Rayleigh fading matrix with i.i.d. complex Gaussian entries, and both the transmit and receive correlation matrices $\mathbf{R}_{\mathrm{t}}$ and $\mathbf{R}_{\mathrm{r}}$ follow a Toeplitz exponential model. The $(i,j)$-th entry of $\mathbf{R}$ is given by $[\mathbf{R}]_{i,j} = \rho^{|i-j|}$. The transmit and receive correlation coefficients are assumed to be identical ($\rho_{\mathrm{t}} = \rho_{\mathrm{r}} = \rho$).

The corresponding results are shown in Figs. 2(b)–(d), which illustrate the variation of relative computation error with respect to $N_{\mathrm{d}}$ for different correlation coefficients $\rho$, under maximum conductance of 200 $\mu$S, 300 $\mu$S, and 400 $\mu$S, respectively. Similar to the Rayleigh case, the relative error first decreases and then increases as $N_{\mathrm{d}}$ grows. In addition, higher spatial correlation not only results in slightly larger computation errors due to the uneven distribution of equivalent channel gains, but also narrows the feasible range of $N_{\mathrm{d}}$. Consequently, although increasing $G_{\max}$ still enlarges the valid $N_{\mathrm{d}}$ range and helps further reduce the overall error, the benefit gradually diminishes under stronger correlation. These results demonstrate that the proposed mapping scheme remains reasonably effective under correlated channel conditions, particularly in relatively rich-scattering channels.

Similar to the Rayleigh channel case, for the Kronecker channel model, the optimal value $N_{\mathrm{d}}^{*}$ is expressed as

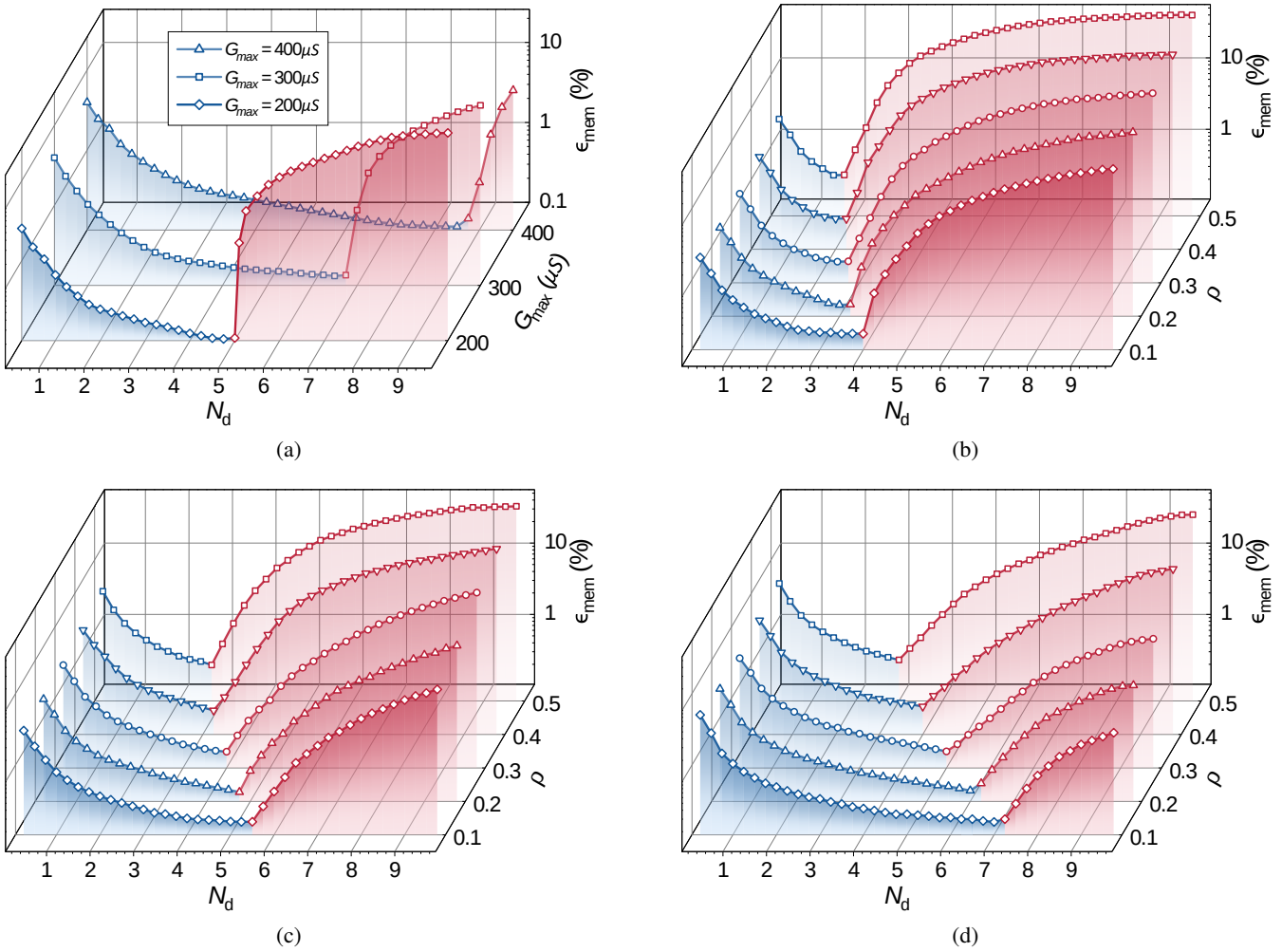


Fig. 2. Circuit computation errors under different mapping ratios and maximum conductance (with same minimum conductance). (a) Rayleigh fading channel, (b) Kronecker channel model ($G_{\max} = 200\mu$S), (c) Kronecker channel model ($G_{\max} = 300\mu$S), (d) Kronecker channel model ($G_{\max} = 400\mu$S).

$$N_{\mathrm{d}}^{*} = \xi \cdot \frac{G_{\max} N_{\mathrm{t}}}{\eta\rho + 3\sqrt{\frac{\zeta}{2}(1+\rho)}}, \tag{11}$$

where $\eta \triangleq \mathrm{tr}(R_{\mathrm{t}})$ and $\zeta \triangleq \mathrm{tr}(R_{\mathrm{t}}^2)$ are determined by the transmit correlation matrix $R_{\mathrm{t}}$. Compared with the Rayleigh channel case, the presence of spatial correlation increases the denominator term in (11), thereby reducing the feasible range of $N_{\mathrm{d}}$ and resulting in a smaller theoretical optimum $N_{\mathrm{d}}^{*}$. This analytical result is consistent with the simulation observations in Figs. 2(b)–(d), confirming that higher correlation levels constrain the allowable mapping ratio and thereby affect the achievable computation accuracy.

Unlike the target conductance of the memristors in the MVM and INV components shown in Fig. 1, the diagonal matrix $\mathbf{D}_{\mathrm{mem}}$ corresponds to a relatively higher set of memristor conductance values. To accommodate this feature while minimizing hardware overhead, a lightweight circuit enhancement is introduced: upon considering the limited number of diagonal elements in $\mathbf{D}_{\mathrm{mem}}$, each memristor in the diagonal matrix is connected in parallel with multiple resistors of fixed conductance $G_{\max}$, while MOSFET switches are utilized to regulate their connection to the circuit and flexibly realize the desired conductance values. The corresponding circuit structure is illustrated in Fig. 3.

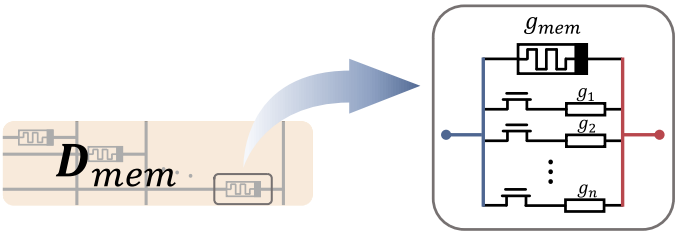


Fig. 3. Circuit structure of one diagonal memristor cell.

Under the assumption of Rayleigh channel, the number of constant resistors $N_{\mathrm{R}}$ contained in each memristor cell can be determined according to

$$N_{\mathrm{R}} = \left\lceil \xi \cdot \left( \frac{\lambda}{N_{\mathrm{tx}}} + 1 \right) \frac{\sqrt{2N_{\mathrm{tx}}}}{3} \right\rceil, \tag{12}$$

where $\lceil \cdot \rceil$ denotes the ceiling function. The value of $N_{\mathrm{R}}$ primarily depends on the memristor crossbar scale and is

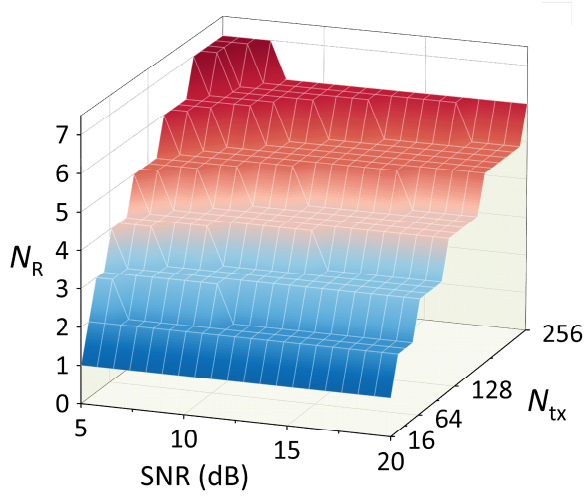


Fig. 4. Number of constant resistors in each memristor cell under Rayleigh channel.

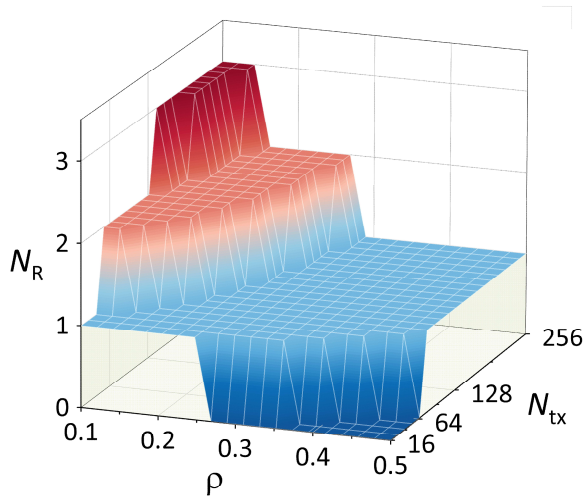


Fig. 5. Number of constant resistors in each memristor cell under Kronecker channel model, with SNR fixed at 10 dB.

relatively less sensitive to the SNR. Fig. 4 illustrates the number of constant resistors required per memristor cell in the proposed circuit when executing the MMSE precoding algorithm under different crossbar scales and SNR conditions.

When spatial correlation is introduced in the Kronecker channel model, the estimation of $N_{\mathrm{R}}$ is modified as

$$N_{\mathrm{R}} = \left\lceil \xi \cdot \left( \frac{\lambda}{N_{\mathrm{tx}}} + 1 \right) \cdot \frac{N_{\mathrm{t}}}{\eta\rho + 3\sqrt{\frac{\zeta}{2}(1+\rho)}} \right\rceil. \tag{13}$$

Fig. 5 further illustrates this relationship by fixing the SNR at 10 dB and plotting the variation of $N_{\mathrm{R}}$ with respect to different crossbar scales and correlation coefficients.

## V. A Monte Carlo Approach for Memristor Programming Time Estimation

### A. Closed-Form Expression of Programming Time

Considering the proposed circuit in the downlink scenario, programming time constitutes a significant portion of the overall computational latency. However, to the best of our knowledge, there has been few prior works that provide a detailed analysis of the programming time. In this subsection, we derives a closed-form expression for the memristor programming time and validates it through Monte Carlo simulations.

Throughout the derivation, $w$ denotes the normalized ratio of programming steps, $T_{\mathrm{s}}$ represent the duration of a single programming pulse, and $S_{\mathrm{total}}$ denote the total number of steps required to switch a memristor from its lowest to highest resistance state. The memristor crossbar commonly implements row-wise parallel programming [8]. Since the impact of residual errors on the programming time is relatively minor, they are excluded in the following simplified model.

Building on the preceding discussion, we now derive a closed-form expression for the expected number of programming steps. We take the matrix $\mathbf{P}_{\mathrm{INV}}$ as an example for derivation. The off-diagonal elements of this matrix follow a Gaussian distribution, while the diagonal elements follow a centralized Gamma distribution. Based on this distinction, the modeling of diagonal and off-diagonal elements is carried out separately as follows.

**Off-diagonal:** Each off-diagonal element is modeled as a zero-mean Gaussian random variable:

$$P \sim \mathcal{N}(0, \sigma^2), \quad \sigma = \alpha \frac{N_{\mathrm{d}}}{\sqrt{2N_{\mathrm{tx}}}}. \tag{14}$$

**Diagonal:** Each diagonal element is modeled as a centralized Gamma random variable with a scaling factor:

$$P \sim \Gamma(k = N_t, \theta) - \alpha N_{\mathrm{d}}, \quad \theta = \alpha \frac{2N_{\mathrm{d}}}{N_{\mathrm{tx}}}. \tag{15}$$

All conductance values are represented by the same variable $P$ for the sake of illustration, and the unit of $P$ is $\mu$S. The conductance variable $P$ is quantized over a predefined range:

$$\begin{cases} G_k = G_{\min} + k\Delta G, \ k = 0, 1, \ldots, L-1, \\ \Delta G = \frac{G_{\max} - G_{\min}}{L}. \end{cases} \tag{16}$$

The quantization function is defined as:

$$Q(P) = \begin{cases} G_0, & P \le G_0, \\ G_k, & G_k < P \le G_{k+1}, \ 1 \le k \le L-1, \\ G_{L-1}, & P > G_{L-1}. \end{cases} \tag{17}$$

To more accurately estimate the programming time of memristors, we adopt the programming model proposed in [17]. Let $\alpha_{\mathrm{p}}$ and $\alpha_{\mathrm{d}}$ represent the potentiation and depression coefficients respectively, which characterize the nonlinearity of the memristor programming process. As shown in (18) and (19), the functions $G_{\mathrm{pot}}(w)$ and $G_{\mathrm{dep}}(w)$ characterize the evolution of memristor conductance with respect to the normalized number of programming steps under the potentiation and depression operations, respectively.

$$G_{\mathrm{pot}}(w) = \left( \left( G_{\max}^{\alpha_{\mathrm{p}}} - G_{\min}^{\alpha_{\mathrm{p}}} \right) w + G_{\min}^{\alpha_{\mathrm{p}}} \right)^{\frac{1}{\alpha_{\mathrm{p}}}}, w \in [0, 1], \tag{18}$$

$$G_{\text{dep}}(w) = \left(\left(G_{\max}^{\alpha_{\text{d}}} - G_{\min}^{\alpha_{\text{d}}}\right)w + G_{\min}^{\alpha_{\text{d}}}\right)^{\frac{1}{\alpha_{\text{d}}}}, w \in [0,1]. \quad (19)$$

From these two equations, we obtain the inverse function of (18) and (19):

$$w(G) = \begin{cases} \dfrac{G^{\alpha_{\text{p}}} - G_{\min}^{\alpha_{\text{p}}}}{G_{\max}^{\alpha_{\text{p}}} - G_{\min}^{\alpha_{\text{p}}}}, & G_{\text{tar}} > G_{\text{cur}}, \\ 1 - \dfrac{G^{\alpha_{\text{d}}} - G_{\min}^{\alpha_{\text{d}}}}{G_{\max}^{\alpha_{\text{d}}} - G_{\min}^{\alpha_{\text{d}}}}, & G_{\text{tar}} \le G_{\text{cur}}, \end{cases} \quad (20)$$

where $G_{\text{cur}}$ denotes the current conductance, and $G_{\text{tar}}$ denotes the target conductance. From $G_{\text{cur}}$ to $G_{\text{tar}}$, the required number of programming steps is given by

$$S_{\text{prog}} = S_{\text{total}} \cdot |w(G_{\text{tar}}) - w(G_{\text{cur}})|. \quad (21)$$

For each quantized level $G_k$, the corresponding probability mass function, $p_k = \Pr(G_k \le P < G_{k+1})$, is detailed below.

**Off-diagonal:**

$$p_k = \begin{cases} \Phi\left(\dfrac{G_1}{\sigma}\right), & k = 0, \\ \Phi\left(\dfrac{G_{k+1}}{\sigma}\right) - \Phi\left(\dfrac{G_k}{\sigma}\right), & 1 \le k \le L-2, \\ 1 - \Phi\left(\dfrac{G_{L-1}}{\sigma}\right), & k = L-1. \end{cases} \quad (22)$$

**Diagonal:**

$$p_k = \begin{cases} F(G_1 + \alpha N_{\text{d}}), & k = 0, \\ F(G_{k+1} + \alpha N_{\text{d}}) - F(G_k + \alpha N_{\text{d}}), & 1 \le k \le L-2, \\ 1 - F(G_{L-1} + \alpha N_{\text{d}}), & k = L-1. \end{cases} \quad (23)$$

where $\Phi(\cdot)$ and $F(\cdot)$ are the cumulative distribution functions of the Gaussian and Gamma distributions, respectively. The probabilities are normalized

$$p_k \leftarrow \frac{p_k}{\sum_{i=0}^{L-1} p_i}, \quad (24)$$

to ensure $\sum_{k=0}^{L-1} p_k = 1$.

Assuming independently sampled conductance $G_m$ and $G_k$, the expected programming step is given by:

$$\mathbb{E}[S_{\text{prog}}] = S_{\text{total}} \sum_{k=0}^{L-1} \sum_{m=0}^{L-1} p_k p_m \, |w(G_m) - w(G_k)|. \quad (25)$$

As $L \to \infty$, the expected number of programming steps can be expressed as

$$\mathbb{E}[S_{\text{prog}}] = S_{\text{total}} \iint |w(y) - w(x)| \, f_P(x) f_P(y) \, dx \, dy, \quad (26)$$

where the probability density function (PDF) $f_P(x)$ is defined over the interval $[G_{\min}, G_{\max}]$. The PDFs are given as follows.

**Off-diagonal:**

$$f_P(z) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{z^2}{2\sigma^2}\right), \quad (27)$$

**Diagonal:**

$$f_P(z) = \frac{1}{\alpha \Gamma(k) \theta^k} \left(\frac{z + \alpha N_{\text{d}}}{\alpha}\right)^{k-1} \exp\left(-\frac{z + \alpha N_{\text{d}}}{\alpha\theta}\right). \quad (28)$$

The programming time $T_{\text{prog}}$ of a single memristor is given by:

$$T_{\text{prog}} = T_{\text{s}} \cdot \bar{S}_{\text{prog}} = T_{\text{s}} \cdot \mathbb{E}[S_{\text{prog}}]. \quad (29)$$

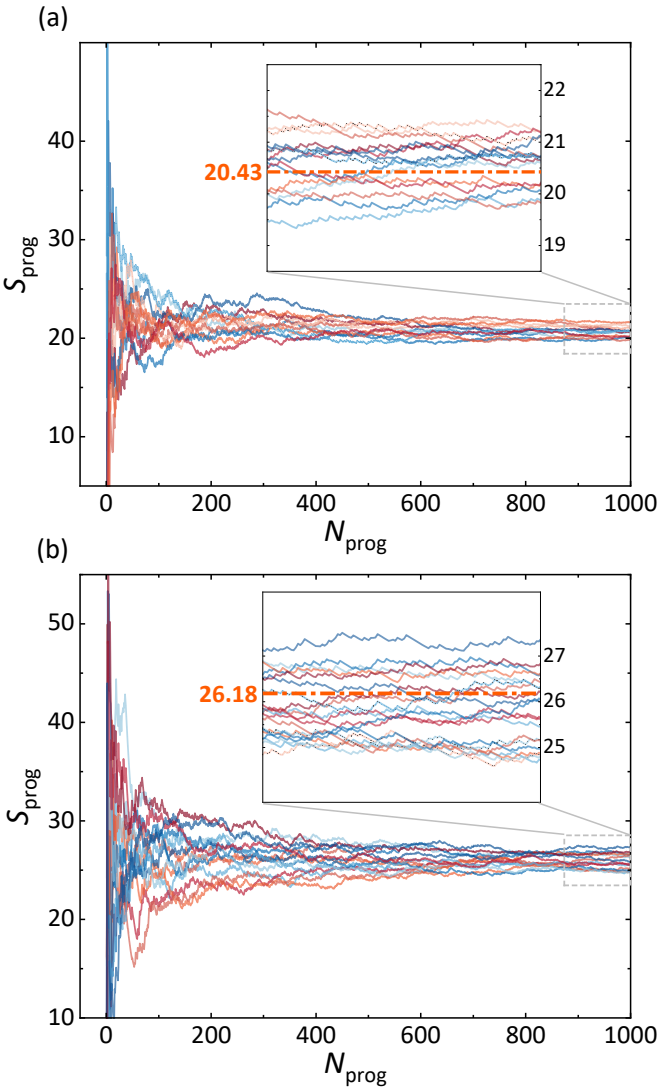


Fig. 6. Convergence of the average number of programming steps obtained via Monte Carlo simulations: (a) $\mathbf{P}_{\text{INV}}$ matrix, and (b) $\mathbf{P}_{\text{MVM}}$ matrix. Each curve represents a different initial conductance state. In each subfigure, the theoretical value is shown in orange for comparison.

### *B. Validation by Monte Carlo Simulation*

To validate the above expression, we conducted a Monte Carlo simulation, as illustrated in Fig. 6, where the theoretical values are highlighted in orange. The simulation parameters are set as $T_{\text{s}} = 1$ ns, $S_{\text{total}} = 100$, $N_{\text{tx}} = 32$ and $N_{\text{d}} = N_{\text{d}}^*$.

For each curve, each point represents the average number of programming steps over the first $N_{\text{prog}}$ independent experiments. As the number of independent programming experiments increases, $\bar{S}_{\text{prog}}$ gradually stabilizes. When repeating the experiments with different initial conductance states, $\bar{S}_{\text{prog}}$ consistently converges with less than 5% deviation from the theoretical value. The Monte Carlo results thus validate that our closed-form solution is accurate.

### *C. Crossbar Programming Time*

Under the row-wise parallel programming scheme, all memristors within a row are updated simultaneously, while mem-

ristor crossbar is sequentially programmed from one row to the next. Hence, the total programming time for the crossbar equals to the number of rows times the programming time per row.

For the $2N_{\mathrm{rx}} \times 2N_{\mathrm{rx}}$ INV crossbar, each row contains one diagonal memristor (programmed with a latency of $T_{\text{INV-d}}$) and multiple off-diagonal memristors (each programmed in $T_{\text{INV-od}}$). Since all memristors within a row are programmed simultaneously, the row programming time is determined by the largest of $T_{\text{INV-d}}$ and $T_{\text{INV-od}}$. Consequently, the overall programming time of the INV crossbar can be expressed as:

$$T_{\mathrm{INV}} = 2N_{\mathrm{rx}} \cdot \max\{T_{\text{INV-d}}, T_{\text{INV-od}}\}. \tag{30}$$

For the $2N_{\mathrm{rx}} \times 2N_{\mathrm{tx}}$ MVM crossbar, all elements follow the same distribution, and thus the overall programming time of the MVM crossbar is given by

$$T_{\mathrm{MVM}} = 2N_{\mathrm{rx}} \cdot T_{\text{MVM-row}}, \tag{31}$$

where $T_{\text{MVM-row}}$ denotes the programming time for each row of MVM crossbar.

The overall programming latency is determined by the slowest crossbar, namely,

$$\begin{aligned} T_{\mathrm{prog}} =& \max\{T_{\mathrm{INV}}, T_{\mathrm{MVM}}\} \\ =& 2N_{\mathrm{rx}} \cdot \max\{T_{\text{INV-d}}, T_{\text{INV-od}}, T_{\text{MVM-row}}\}. \end{aligned} \tag{32}$$

### *D. Programming Time of Memristor Crossbar in Massive MIMO Systems*

In massive MIMO systems, the impact of memristor variability on programming steps is more significant due to the large number of memristors updating in parallel. While Section V-A focuses on the programming time of a single memristor cell, this alone does not reflect the actual programming latency. Specifically, since all memristors in a row are programmed simultaneously, the row programming latency is determined by the slowest device, i.e., the one requiring the maximum number of steps. This effect becomes more significant as the crossbar size increases. To demonstrate the time complexity, we employ extreme value theory to model the statistical distribution of the maximum programming steps.

We take the programming time of matrix $\mathbf{P}_{\text{INV}}$ as an example for derivation. As derived in (25), the mean $\mu_S$ and variance $\sigma_S^2$ of $S_{\text{prog}}$ can be obtained through:

$$\mu_S = \mathbb{E}[S_{\text{prog}}], \quad \sigma_S^2 = \mathbb{E}[S_{\text{prog}}^2] - (\mathbb{E}[S_{\text{prog}}])^2. \tag{33}$$

Since $S_{\text{prog}}$ is bounded between 0 and $S_{\text{total}}$, it satisfies a sub-Gaussian tail bound, expressed as

$$\Pr\{S_{\text{prog}} - \mu_S > x\} \le \exp\left(-\frac{x^2}{2\sigma_S^2}\right), \quad x \ge 0. \tag{34}$$

Assume that there are $M_{\text{off}} = 2N_t - 1$ off-diagonal memristors in each row. Let $S_{\max}$ denote the maximum programming steps among these memristors, which is expressed as:

$$S_{\max} = \max_{1 \le j \le M_{\text{off}}} S_j, \tag{35}$$

where $S_j$ are i.i.d. copies of $S_{\text{prog}}$. Applying the sub-Gaussian tail inequality (34), the survival function of $S_{\max}$ is upper bounded by

$$\Pr\{S_{\max} > \mu_S + x\} \le M \exp\left(-\frac{x^2}{2\sigma_S^2}\right). \tag{36}$$

The expected maximum programming steps can be expressed as

$$\mathbb{E}[S_{\max}] = \mu_S + \int_0^\infty \Pr\{S_{\max} > \mu_S + x\} dx. \tag{37}$$

To tightly bound this integral, we select a threshold $x_0 = \sigma_S\sqrt{2\ln M_{\text{off}}}$ and partition the integration domain:

$$\mathbb{E}[S_{\max}] \le \mu_S + x_0 + \int_{x_0}^\infty M_{\text{off}} \exp\left(-\frac{x^2}{2\sigma_S^2}\right) dx. \tag{38}$$

By using the standard upper bound for the Q-function and evaluating the Gaussian tail integral, we obtain:

$$\mathbb{E}[S_{\max}] \le \mu_S + \sigma_S\sqrt{2\ln M_{\text{off}}} + \frac{\sigma_S}{\sqrt{2\pi \ln M_{\text{off}}}}. \tag{39}$$

Consequently, the programming time of a single row of $\mathbf{P}_{\text{INV}}$ can be approximated as

$$T_{\text{INV-row}} = \max\left\{T_{\text{INV-d}},\ T_s\ \mathbb{E}[S_{\max}]\right\}. \tag{40}$$

The above analysis demonstrates that the time complexity for programming time is $O(\sqrt{\ln N_t})$, which depends on the number of transmit antennas in MIMO systems. This result enables more accurate estimation of the programming time in massive MIMO scenarios, and the derived expressions provide closed-form guideline for assessing the programming time overhead in memristor-based implementations.

## VI. Performance of Proposed MIMO Precoding Circuit

To evaluate the performance of the proposed circuit in executing precoding algorithms, simulation experiments are conducted to analyze the impact of memristor non-idealities on BER in the downlink scenario. In the simulation, the modulation scheme is quadrature amplitude modulation (QAM) with the modulation order $M$.

We first evaluate the impact of $\epsilon_p$ and $\epsilon_{bp}$ on system BER when executing the MMSE precoding algorithm using the proposed circuit with $N_{\mathrm{rx}} = 16$, $N_{\mathrm{tx}} = 32$, $M = 16$ and the signal-to-noise ratio (SNR) of 16 dB. Fig. 7 shows that the system BER decreases as $\epsilon_{\mathrm{bp}}$ increases or as $\epsilon_{\mathrm{p}}$ decreases. As shown in the dark blue region, when $\epsilon_{\mathrm{bp}} \ge 6$ and $\epsilon_{\mathrm{p}} \le 3$, the system BER remains at a low level. Under these conditions, the BER degradation of the proposed circuit is less than 5%, compared to FP64 precoding executed on graphics processing unit (GPU).

Next we evaluate the impact of $\epsilon_p$ and $\epsilon_{bp}$ with different modulation schemes on BER. The yellow contour line in Fig. 8 marks the BER obtained under FP64 computation, serving as a performance benchmark for comparison. The simulation is conducted with $N_{\mathrm{rx}} = 4$, $N_{\mathrm{tx}} = 32$, and $\mathrm{SNR} = 10\,\mathrm{dB}$, under 4QAM, circular 8QAM, rectangular 8QAM, and 16QAM modulation schemes. As the modulation order decreases, the

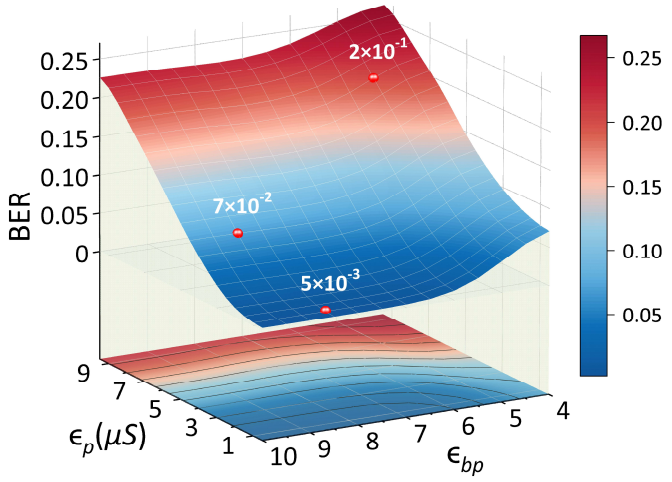


Fig. 7. Impact of $\epsilon_p$ and $\epsilon_{bp}$ on system BER when executing the MMSE precoding algorithm using the proposed circuit.

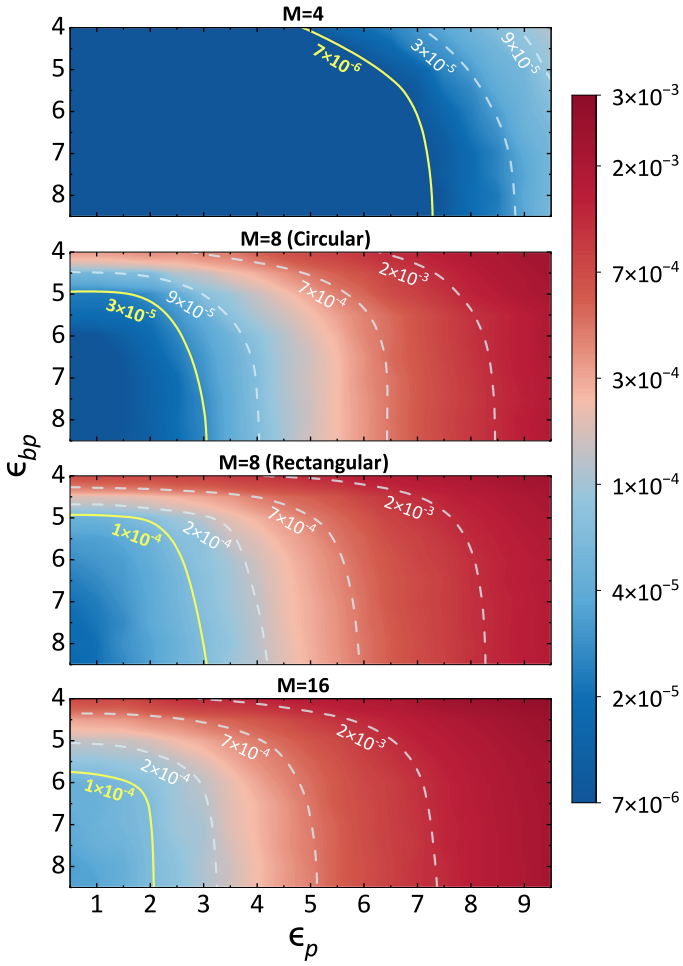


Fig. 8. BER performance of the proposed circuit under different modulation orders when executing the MMSE precoding algorithm.

system BER decreases accordingly. Under 16-QAM modulation, the proposed circuit achieves a BER comparable to that of FP64 computation when $\epsilon_{\rm bp}$ exceeds 6 bits and $\epsilon_{\rm p}$ is below 2 $\mu$S. Under 8-QAM modulation, comparable performance is observed with $\epsilon_{\rm bp}$ above 5 bits and $\epsilon_{\rm p}$ below 3 $\mu$S. For 4-QAM modulation, the circuit achieves a comparable BER even with $\epsilon_{\rm bp}$ reduced to 4 bits and $\epsilon_{\rm p}$ increased to 7 $\mu$S. These results demonstrate that reducing the modulation order significantly relaxes the precision requirements for the memristor.

In Fig. 9, we further compare the BER performance of the proposed circuit when executing ZF and MMSE algorithms under different $\epsilon_{\rm bp}$ values, with $N_{\rm tx} = 32$, $N_{\rm rx} = 16$, M $= 4$ and $\epsilon_{\rm p} = 1\,\mu$S. The shaded regions in the figure represent the inter-quartile range, highlighting the spread between the 25th and 75th percentiles of the BER across Monte Carlo trials.

At low SNRs, as expected, no significant difference is observed between the proposed circuit and the FP64 computation. As the SNR increases, the BER reduction rate of the proposed circuit is significantly suppressed due to the limited bit precision of the memristor. At medium-to-high SNRs, the circuit with 4-bit memristors exhibits a significantly higher BER than that with 6-bit memristors, whereas replacing 6-bit memristors with 8-bit ones yields only marginal performance improvement.

## VII. Throughput and Energy Efficiency

For both ZF and MMSE precoding algorithms, the primary computational burden arises from matrix inversion, leading to a time complexity of $O(n^3)$, where $n$ is the dimension of the square matrix $\widetilde{\mathbf{Z}} = \widetilde{\mathbf{H}}\widetilde{\mathbf{H}}^{\rm H}$. Under ideal conditions, the proposed circuit can execute ZF or MMSE precoding within a single convergence time, which remains constant regardless of the matrix size, which corresponds to a time complexity of $O(1)$. In the following, we provide a detailed analysis of the throughput and energy efficiency of the proposed circuit.

As an example, we determine the numbers of floating-point operations (FLOPs) required by the two precoding algorithms given $N_{\rm tx} = 32$ and $N_{\rm rx} = 16$. Considering the operations accelerated by the proposed circuit, completing one group of symbols requires 25,600 FLOPs for ZF precoding and 25,856 FLOPs for MMSE precoding.

TABLE I
Power Consumption of Main Components in the Proposed MIMO Precoder Circuit

| Main Component | Key Specifications | Power Consumption | Reference |
|---|---|---|---|
| Current digital-to-analog converter (DAC) | 8-bit, 0.4 ns conversion delay | 1.6 mW | [18] |
| Voltage analog-to-digital converter (ADC) | 10-bit, 0.5 ns conversion delay | 41.3 $\mu$W | [19] |
| Operational Amplifier (OA) | 500 MHz gain-bandwidth product, 80 dB open-loop gain | 12 $\mu$W | [6] |
| Memristor Programming | 6-bit | 0.6 pJ | [20] |

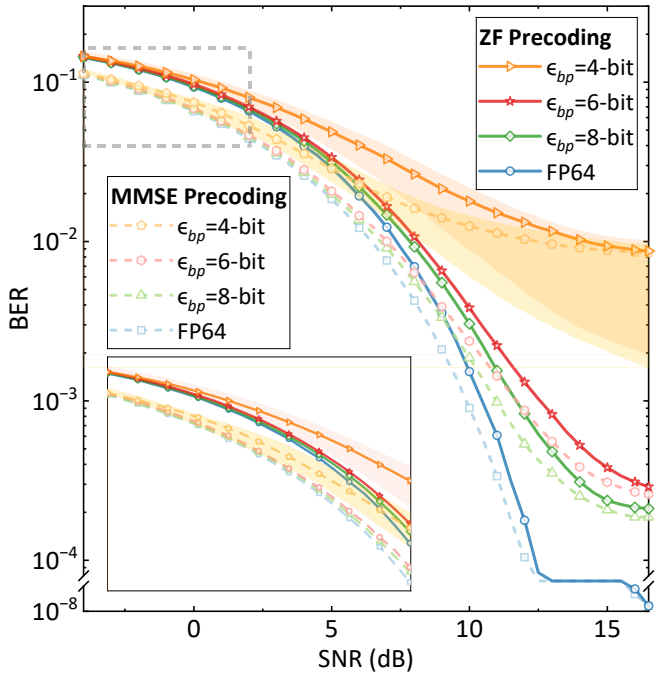


Fig. 9. BER comparison between the proposed circuit and FP64 computation under ZF and MMSE algorithms.

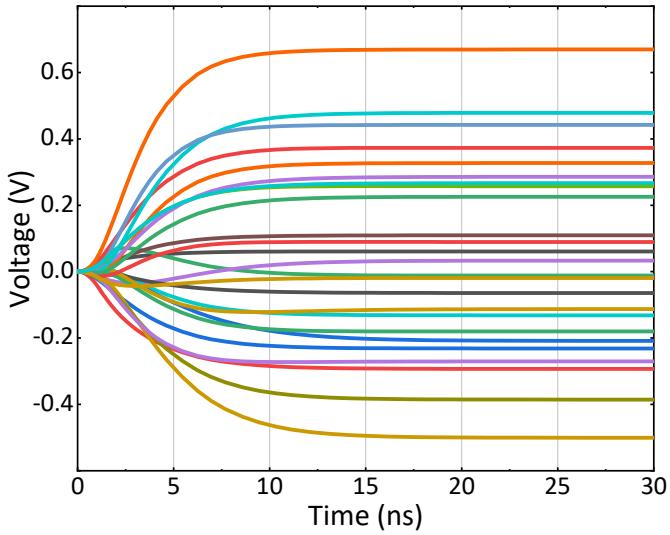


Fig. 10. Convergence time of the proposed circuit over different samples.

Studies have shown that the convergence time of the resistor crossbar is independent of the crossbar size, but depends on the factors, such as the distribution of conductance values and the eigenvalues of the correlation matrix associated with the conductance matrix [21], [22]. LTspice® transient simulation results indicate that the convergence time of the proposed circuit is less than 20 ns, as illustrated in Fig. 10, where each curve represents the temporal evolution of the output voltage vector $\mathbf{v}_{\mathrm{x}}$ in response to one group of input symbols.

Based on the estimation of programming time $T_{\mathrm{prog}}$ described in Section V and the single convergence time $T_{\mathrm{conv}}$ in

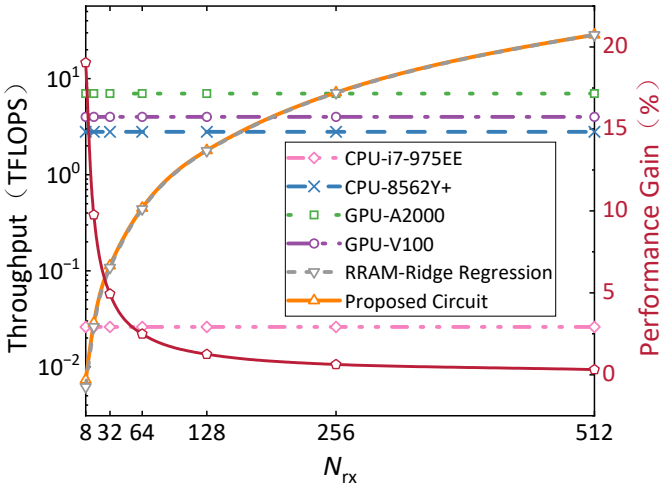


Fig. 11. Throughput of the proposed circuit compared with commercial processors and an RRAM IMC baseline under varying matrix sizes ($N_{\mathrm{tx}} = 2N_{\mathrm{rx}}$). The right-axis scale indicates the performance gain of the proposed circuit relative to the RRAM IMC baseline.

Fig. 10, the total computation time $T_{\text{total-mem}}$ can be obtained by $T_{\text{total-mem}} = T_{\text{prog}} + T_{\text{conv}}$. For commercial processors, the total computation time $T_{\text{total-pro}}$ consists of data transfer time $T_{\text{tran}}$ and floating-point computation time $T_{\text{comp}}$. According to the estimation in [6], the total computation time is approximately twice the floating-point computation time $T_{\text{total-pro}} \approx 2 \cdot T_{\text{comp}}$.

Table I lists the energy consumption of the main components in our MIMO precoder circuit, and Table II presents the energy consumption of the four baseline general-purpose processors. The energy consumption of the proposed circuit is evaluated with reference to the method proposed in [6], focusing on its key components, while the energy consumption of the commercial processors are estimated based on their datasheets.

In addition to the energy consumption, we also quantify the area footprint of the main components in the proposed circuit. To facilitate comparison, all circuit areas are evaluated using the same 14-nm CMOS technology node adopted in [6]. Specifically, the operational amplifier occupies approximately 50 $\mu\mathrm{m}^2$ [6], the 8-bit DAC requires 3.07 $\mu\mathrm{m}^2$ [18], the 10-bit ADC occupies 0.01 $\mathrm{mm}^2$ [19]. Regarding the memory elements, the area footprint of a regular RRAM cell is estimated based on the $4F^2$ scaling assumption reported in [20]. For the proposed diagonal cell, its area is estimated by proportionally scaling the 0.46 $\mu\mathrm{m}^2$ footprint reported for the 6T6R cell with parallel resistors in [6].

Furthermore, throughput, energy efficiency, and area efficiency are adopted as evaluation metrics to comprehensively assess the performance of the proposed circuit. They are respectively defined as

$$\text{Throughput} = \text{FLOP}/\text{Total computation time}, \quad (41)$$

$$\text{Energy efficiency} = \text{FLOP}/\text{Energy consumption}, \quad (42)$$

$$\text{Area efficiency} = \text{Throughput}/\text{Area}. \quad (43)$$

Figs. 11–13 compare the throughput, energy efficiency, and area efficiency of the proposed circuit with those of four representative commercial processors (Intel Core i7-975EE,

TABLE II
POWER CONSUMPTION OF COMMERCIAL PROCESSORS

| Processor Type | Model | Power Consumption | Peak Floating-Point Performance | Die Size | Reference |
|---|---|---|---|---|---|
| CPU | Intel Core i7-975EE | 130 W | 53.28 GFLOPS | 263 $mm^2$ | [23] |
| CPU | Intel Xeon Platinum 8562Y+ | 300 W | 5.6 TFLOPS | N/A | [24] |
| GPU | NVIDIA RTX A2000 | 70 W | 8 TFLOPS | 276 $mm^2$ | [25] |
| GPU | NVIDIA TESLA V100 | 250 W | 14 TFLOPS | 815 $mm^2$ | [26] |

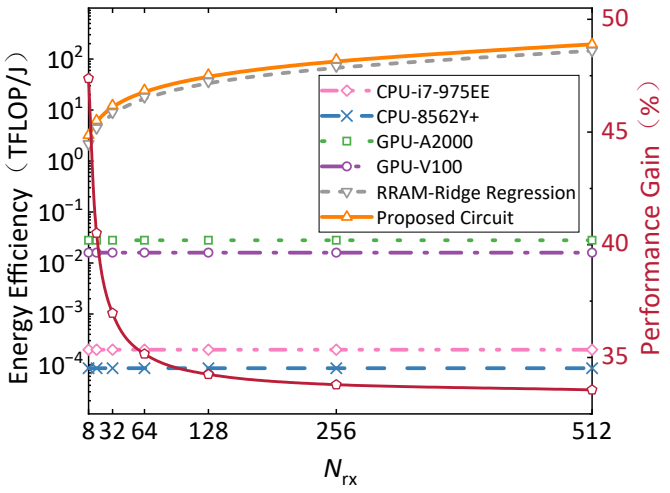


Fig. 12. Energy efficiency of the proposed circuit compared with commercial processors and an RRAM IMC baseline under varying matrix sizes ($N_{tx} = 2N_{rx}$). The right-axis scale indicates the performance gain of the proposed circuit relative to the RRAM IMC baseline.

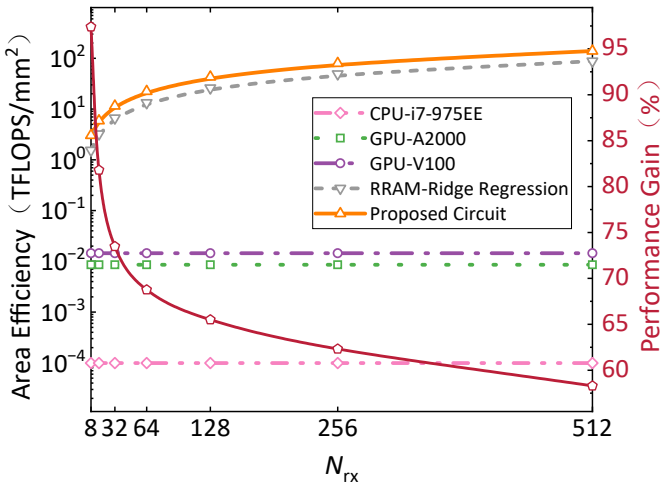


Fig. 13. Area efficiency of the proposed circuit compared with commercial processors and an RRAM IMC baseline under varying matrix sizes ($N_{tx} = 2N_{rx}$). The right-axis scale indicates the performance gain of the proposed circuit relative to the RRAM IMC baseline.

Intel Xeon Platinum 8562Y+, NVIDIA RTX A2000, and NVIDIA Tesla V100) as well as an analog RRAM-based baseline circuit that implements the same MMSE algorithm through ridge-regression computation [6]. The specifications of the commercial processors were obtained from their official datasheets [23]–[26] and are summarized in Table II. These processors serve as representative general-purpose central processing units (CPUs) and GPUs, while the analog RRAM design serves as a representative in-memory-computing (IMC) baseline for the same algorithmic functionality. As can be seen from Figs. 11–13, the throughput, energy efficiency, and area efficiency of commercial processors remain nearly constant regardless of the matrix dimension, while those of the proposed memristor crossbar improve significantly as the matrix size increases.

When the matrix size $N_{rx}$ is smaller than 16, the throughput of the proposed circuit is lower than that of the Intel Core i7-975EE. At $N_{rx} = 256$, the throughput becomes comparable to that of state-of-the-art commercial processors. As the matrix size increases further, the throughput of the memristor crossbar surpasses all the selected baseline processors. Compared with the RRAM IMC baseline, the proposed design achieves a slightly higher throughput, as reflected by the performance gain curve with its scale shown on the right axis.

In terms of energy efficiency, the memristor crossbar consistently demonstrates advantages of several orders of magnitude across all matrix sizes to commercial processors. For example, the energy efficiency of the proposed circuit is 100 times higher than that of the NVIDIA RTX A2000 when $N_{rx} = 8$. Similar to the RRAM IMC baseline, the proposed design exhibits an increasing trend as $N_{rx}$ grows, yet consistently achieves a 30%–50% improvement across all matrix sizes.

Regarding area efficiency, the proposed circuit outperforms the selected CPU and GPU baselines by two to three orders of magnitude. Compared with the RRAM IMC baseline, the proposed design follows the same increasing trend as the matrix size grows, while achieving 60%–100% higher efficiency across the evaluated range of $N_{rx}$.

In addition, although the integration of parallel resistors may raise concerns regarding potential area overhead, our design confines these resistive elements to the diagonal positions of the crossbar, resulting in a negligible footprint relative to the full crossbar. The relative area proportions among the three core components (Operational Amplifier, regular RRAM cell, proposed diagonal cell) of the proposed circuit are illustrated in Fig. 14, for small-scale crossbars (e.g. $N_{rx} = 8$), the area occupied by the proposed diagonal cells is comparable to that of the regular RRAM cells, while both together constitute only

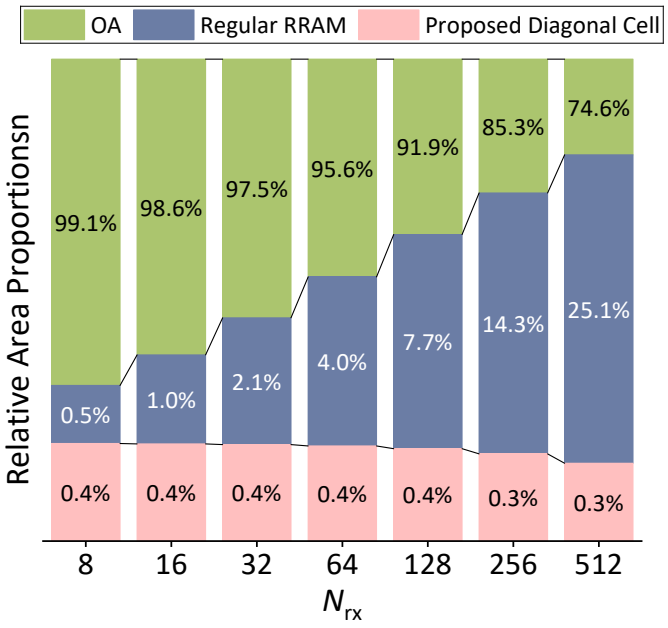


Fig. 14. Area composition of the three core components (operational amplifier, regular RRAM cell, proposed diagonal memristor cell) in the proposed circuit under varying matrix sizes ($N_{\rm tx} = 2N_{\rm rx}$).

a minor portion within the considered circuit components, whose overall area is largely dominated by the operational amplifiers. As the array dimension increases, the number of diagonal cells grows only linearly with $N_{\rm rx}$, whereas the RRAM cells increase quadratically, leading to a clear divergence between the two. Consequently, the fraction of the diagonal cells remains nearly constant around $0.3\% - 0.4\%$ and even slightly decreases, indicating that their contribution to the area of the proposed circuit becomes progressively less significant. Therefore, the integration of these additional resistors does not compromise the density advantage of the proposed IMC architecture.

## VIII. Conclusion

In this paper, we have proposed a memristor crossbar-based precoder circuit for accelerating linear precoding algorithm in downlink massive MIMO systems. To address the computational inaccuracies arising from the limited conductance range of memristors, we have optimized the mapping scheme that jointly considers matrix characteristics and device constraints under both Rayleigh and Kronecker channel models, reducing the relative computation error by more than 60% compared to baseline approaches. Furthermore, we have developed a probabilistic model for estimating the programming time associated with the proposed circuit. Based on this model, the closed-form upper bound and the complexity for the programming time have been obtained. Simulation results have shown that when $\epsilon_{\rm p} \leq 3$ and $\epsilon_{\rm bp} \geq 6$, the BER of the proposed circuit only degrades within 5% compared to FP64 GPU. Additionally, the proposed circuit achieves high throughput while significantly enhances both energy and area efficiency in massive MIMO configurations, outperforming conventional digital processors under comparable conditions.

In summary, this work has identified and addressed key enabling factors for ultra-efficient precoding in massive MIMO systems. The proposed mapping scheme and probabilistic model together bridge the gap between memristor-level non-idealities and system-level performance, laying a foundation for scalable and energy-aware baseband signal processing acceleration.

## References


[1] H. Jahanbakhsh, N. Ojaroudi Parchin, and C. H. See, "Antenna design and optimization for 5G, 6G, and IoT," *Sensors*, vol. 25, p. 1494, Feb. 2025.

[2] R. Kumar, D. Sinwar, and V. Singh, "Experimental evaluation of MU-MIMO in TDD environment for 5G NR using exploiting channel reciprocity," *International Journal of Information Technology*, vol. 16, pp. 4651–4666, Jun. 2024.

[3] Z. Sun, G. Pedretti, E. Ambrosi *et al.*, "Solving matrix equations in one step with cross-point resistive arrays," *Proceedings of the National Academy of Sciences*, vol. 116, no. 10, pp. 4123–4128, Feb. 2019.

[4] L. Xia, P. Gu, B. Li *et al.*, "Technological exploration of RRAM crossbar array for matrix-vector multiplication," *Journal of Computer Science and Technology*, vol. 31, no. 1, pp. 3–19, Mar. 2016.

[5] G. Yuan, C. Ding, R. Cai *et al.*, "Memristor crossbar-based ultra-efficient next-generation baseband processors," in *Proc. IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, Boston, MA, USA, Aug. 2017, pp. 1121–1124.

[6] P. Mannocci, E. Melacarne, and D. Ielmini, "An analogue in-memory ridge regression circuit with application to massive MIMO acceleration," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 12, no. 4, pp. 952–962, Dec. 2022.

[7] P. Zuo, Z. Sun, and R. Huang, "Extremely-fast, energy-efficient massive MIMO precoding with analog RRAM matrix computing," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 70, no. 7, pp. 2335–2339, Jul. 2023.

[8] Y.-H. Ren, S. Yang, J.-H. Bi, and Y.-X. Zhang, "Accelerating maximum-likelihood detection in massive MIMO: A new paradigm with memristor crossbar based in-memory computing circuit," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 12, pp. 19 745–19 750, Dec. 2024.

[9] J.-H. Bi, S. Yang, S. Chen, and P. Zhang, "High-speed ultra-energy-efficient memristor-based massive MIMO SIC detector circuit with hybrid analog-digital computing architecture," *IEEE Transactions on Vehicular Technology*, early access, May 2025.

[10] J.-H. Bi, S. Yang, P. Zhang *et al.*, "Amplifier-enhanced memristive massive MIMO linear detector circuit: An ultra-energy-efficient and robust-to-conductance-error design," in *Proc. IEEE Global Communications Conference (GLOBECOM 2024)*, Cape Town, South Africa, Dec. 2024, pp. 3968–3973.

[11] J.-H. Bi, S. Yang, P. Zhang *et al.*, "In-memory massive MIMO linear detector circuit with extremely high energy efficiency and strong memristive conductance deviation robustness," in *Proc. IEEE Global Communications Conference (GLOBECOM 2024)*, Cape Town, South Africa, Dec. 2024, pp. 728–733.

[12] Q. Zeng, J. Liu, M. Jiang *et al.*, "Realizing in-memory baseband processing for ultrafast and energy-efficient 6G," *IEEE Internet of Things Journal*, vol. 11, no. 3, pp. 5169–5183, Feb. 2024.

[13] M. A. Albreem, A. H. Al Habbash, A. M. Abu-Hudrouss, and S. S. Ikki, "Overview of precoding techniques for massive MIMO," *IEEE Access*, vol. 9, pp. 60 764–60 801, Apr. 2021.

[14] H. Lee, I. Sohn, D. Kim, and K. B. Lee, "Generalized MMSE beamforming for downlink MIMO systems," in *Proc. IEEE International Conference on Communications (ICC)*, Kyoto, Japan, Jun. 2011, pp. 1–6.

[15] C. Li, M. Hu, Y. Li *et al.*, "Analogue signal and image processing with large memristor crossbars," *Nature Electronics*, vol. 1, no. 1, pp. 52–59, Jan. 2018.

[16] M. Rao, H. Tang, J. Wu *et al.*, "Thousands of conductance levels in memristors integrated on CMOS," *Nature*, vol. 615, no. 7954, pp. 823–829, Mar. 2023.

[17] J.-W. Jang, S. Park, G. W. Burr *et al.*, "Optimization of conductance change in Pr 1–x Ca x MnO 3-based synaptic devices for neuromorphic systems," *IEEE Electron Device Letters*, vol. 36, no. 5, pp. 457–459, May 2015.

[18] S. I. Huq, S. Islam, N. Saqib, and S. N. Biswas, "Design of low power 8-bit DAC using PTM-LP technology," in *Proc. International Conference on Recent Trends in Electrical, Electronics and Computing Technologies (ICRTEECT)*, Warangal, Telangana, India, Jul. 2017, pp. 64–69.
[19] A. Wang and C.-J. R. Shi, "A 10-bit 50-MS/s SAR ADC with 1 fJ/conversion in 14 nm SOI finFET CMOS," *Integration*, vol. 62, pp. 246–257, Mar. 2018.
[20] F. Zahoor, T. Z. Azni Zulkifli, and F. A. Khanday, "Resistive random access memory (RRAM): An overview of materials, switching mechanism, performance, multilevel cell (MLC) storage, modeling, and applications," *Nanoscale Research Letters*, vol. 15, pp. 1–26, Apr. 2020.
[21] Z. Sun, G. Pedretti, P. Mannocci *et al.*, "Time complexity of in-memory solution of linear systems," *IEEE Transactions on Electron Devices*, vol. 67, no. 7, pp. 2945–2951, May 2020.
[22] Z. Sun and R. Huang, "Time complexity of in-memory matrix-vector multiplication," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 8, pp. 2785–2789, Aug. 2021.
[23] "Data sheet: i7-975 processor extreme edition," Intel, 2024. [Online]. Available: https://ark.intel.com/content/www/us/en/ark/products/37153/intel-core-i7-975-processor-extreme-edition-8m-cache-3-33-ghz-6-40-gt-s-intel-qpi.html.
[24] "Data sheet: Platinum 8562Y+ processor," Intel, 2024. [Online]. Available: https://www.intel.com/content/www/us/en/products/sku/237558/intel-xeon-platinum-8562y-processor-60m-cache-2-80-ghz/specifications.html.
[25] "Data sheet: RTX A2000," NVIDIA, 2021. [Online]. Available: https://www.nvidia.com/content/dam/en-zz/Solutions/design-visualization/rtx-a2000/nvidia-rtx-a2000-datasheet.pdf.
[26] "Data sheet: TESLA V100," NVIDIA, 2017. [Online]. Available: https://www.nvidia.com/content/dam/en-zz/zh_cn/Solutions/nvidia/data-center/tesla/tesla-volta-v100-datasheet-a4-fnl-web-cn.pdf.

**Yu-Xin Zhang** received the B.Eng. degree in communications engineering from Beijing University of Posts and Telecommunications (BUPT), China, in 2024. He is currently pursuing the joint Ph.D. degree in information and communication engineering with the School of Information and Communication Engineering, BUPT, with the Key Laboratory of Mathematics and Information Networks, Ministry of Education, and also with the Department of IoT Technologies and Applications, China Mobile Research Institute. His research interests focus on baseband algorithms and circuit design with memristor and in-memory computing.

**Shaoshi Yang** (Senior Member, IEEE) received the B.Eng. degree in information engineering from Beijing University of Posts and Telecommunications (BUPT), China, in 2006, and the Ph.D. degree in electronics and electrical engineering from the University of Southampton, U.K., in 2013. From 2008 to 2009, he was a Researcher with Intel Labs China. From 2013 to 2016, he was a Research Fellow with the School of Electronics and Computer Science, University of Southampton. From 2016 to 2018, he was a Principal Engineer with Huawei Technologies Co. Ltd., where he made significant contributions to the products, solutions, and standardization of 5G, wideband IoT, and cloud gaming/VR. He was a Guest Researcher with the Isaac Newton Institute for Mathematical Sciences, University of Cambridge. He is currently a Full Professor with BUPT. His research interests include 5G/5G-A/6G, massive MIMO, mobile ad hoc networks, distributed artificial intelligence, and cloud gaming/VR. He is a Deputy Director of the Key Laboratory of Mathematics and Information Networks, Ministry of Education, and a Standing Committee Member of the CCF Technical Committee on Distributed Computing and Systems. He received the Dean's Award for Early Career Research Excellence from the University of Southampton in 2015, the Huawei President Award for Wireless Innovations in 2018, the IEEE TCGCC Best Journal Paper Award in 2019, the IEEE Communications Society Best Survey Paper Award in 2020, the Xiaomi Young Scholars Award in 2023, the CAI Invention and Entrepreneurship Award in 2023, the CIUR Industry-University-Research Cooperation and Innovation Award in 2023, and the First Prize of Beijing Municipal Science and Technology Advancement Award in 2023. He is an Editor of *IEEE Transactions on Communications*, *IEEE Transactions on Vehicular Technology*, and *Signal Processing* (Elsevier). He was also an Editor of *IEEE Systems Journal* and *IEEE Wireless Communications Letters*. For more details on his research progress, please refer to https://shaoshiyang.weebly.com/.

**Yi-Hang Ren** received the B.Eng. degree in communications engineering from Beijing University of Posts and Telecommunications (BUPT), China, in 2023. He is currently pursuing the Ph.D. degree in information and communication engineering with the School of Information and Communication Engineering, BUPT, and the Key Laboratory of Mathematics and Information Networks, Ministry of Education. His current research interests include baseband algorithms with in-memory computing, memristor-based circuit, and AI algorithms with in-memory computing.

**Sheng Chen** (Life Fellow, IEEE) received the B.Eng. degree in control engineering from East China Petroleum Institute, Dongying, China, in 1982, the Ph.D. degree in control engineering from the City, University of London in 1986, and the Doctor of Sciences (D.Sc.) degree from the University of Southampton, Southampton, U.K., in 2005. From 1986 to 1999, he held research and academic appointments at The University of Sheffield, U.K., The University of Edinburgh, U.K., and the University of Portsmouth, U.K. Since 1999, he has been with the School of Electronics and Computer Science, University of Southampton, where he is currently a Professor of intelligent systems and signal processing. He has published over 700 research articles. He has more than 22,000 Web of Science citations with an H-index of 64 and more than 42,000 Google Scholar citations with an H-index of 87. His research interests include adaptive signal processing, wireless communications, modeling and identification of nonlinear systems, neural network and machine learning, evolutionary computation methods, and optimization. He is also a Fellow of the Royal Academy of Engineering, U.K., a Fellow of Asia–Pacific Artificial Intelligence Association (AAIA), and a Fellow of IET. He was one of the original ISI Highly Cited Researchers in Engineering in March 2004.

**Ping-Zhang** (Fellow, IEEE) is currently a Professor with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, the Director of State Key Laboratory of Networking and Switching Technology, a Member of IMT-2020 (5G) Experts Panel, a Member of Experts Panel for China's 6G development. He served as Chief Scientist of National Basic Research Program (973 Program), an Expert in Information Technology Division of National High-tech R&D Program (863 Program), and a Member of Consultant Committee on International Cooperation of National Natural Science Foundation of China. His research interests mainly focus on wireless communications. He is an Academician of the Chinese Academy of Engineering (CAE).