Harmony Search-Based Quantization Bit Allocation on Cell-Free Massive MIMO Systems

Minje Kim*, In-soo Kim†, and Junil Choi*
*School of Electrical Engineering, Korea Advanced Institute of Science and Technology, Daejeon, Korea
Email: {mjkim97, junil}@kaist.ac.kr
†Wireless Research & Development, Qualcomm Technologies Inc., San Diego, CA, USA
Email:{insookim}@qti.qualcomm.com

***Abstract*—In this paper, we propose a novel harmony search (HS)-based algorithm for optimizing bit allocation in cell-free massive multiple-input multiple-output (CF-mMIMO) systems, addressing the challenge of quantization errors due to fronthaul capacity limitations. By employing the quantizer, we analyze the impact of bit allocation on quantization noise and formulate an optimization problem focused on sum-rate maximization. Our algorithm, designed to tackle a non-convex combinatorial problem, dynamically updates bit configurations using exploration and exploitation strategies. Simulation results demonstrate that the proposed algorithm significantly improves sum-rate over uniform bit allocation while reducing computational complexity.**

***Index Terms*— Cell-free massive multiple-input multiple-output (CF-mMIMO), bit allocation, harmony search (HS), limited fronthaul links, quantization**

I. Introduction

Massive multiple-input multiple-output (MIMO) systems, in which huge numbers of antennas are equipped usually at a base station or an access point (AP), have garnered interest as a promising technology for energy-efficient and high-throughput communication performances [1]. In traditional cellular massive MIMO systems, undesired signals such as inter-cell interference from other APs degrade the communication performance. To deal with severe interference at cell-edges, an idea of cooperatively implementing multiple APs without dividing service area is raised in early 2010s, called *cell-free* massive MIMO (CF-mMIMO) systems [2]. The cell-free concept can mitigate the interference by jointly processing all signals from/to an entire set of APs at a control processing unit (CPU). As a result, the cell-free massive MIMO systems can flexibly utilize limited resources and lead to more fair performances for users [3].

To fully exploit the advantages of CF-mMIMO systems, precise and error-free data transmission to the CPU is essential to prevent unexpected loss. Constructing error-free fronthaul links between the APs and CPU, however, is infeasible due to practical limitations, e.g., limited fronthaul capacity and high operational costs [4]. As a solution to handle this non-perfect data transmission, signal quantization was considered in several studies [5]–[7]. According to [5], the achievable rate of the CF-mMIMO system was analyzed under the hardware impairment and limited fronthaul assumption. In [6], the codebook that minimizes the channel estimation error was proposed for the practical scenario where limited fronthaul links and low-resolution analog-to-digital converters/digital-to-analog converters (ADCs/DACs) exist. Other work [7] focused on finding effective fronthaul capacity for APs in terms of energy effieiciency. However, the analyses of these studies assumed the same number of fronthaul bits for each APs.

In this paper, we propose a fornthual bit allocation algorithm that can flexibly control the limited capability of fronthaul links. Based on the harmony-search (HS) algorithm [8], we build a population-based bit allocation algorithm, which exploits previous candidate configurations for updating solutions. Through comprehensive simulations, we demonstrate the effectiveness of allocating fronthaul bits.

II. System Model

In this section, we first introduce our system model of interest. We examine a CF-mMIMO configuration comprising M APs each with N antennas, serving K users each with a single antenna. These APs are interconnected with the CPU via fronthaul links, which enables APs to simultaneously support all the users using shared frequency and time resources. We use the same channel model as [7], and the channels between the APs and CPU are represented as $\mathbf{g}_{m,k} = \beta_{m,k}^{1/2}\mathbf{h}_{m,k}$, where $\beta_{m,k}$ is a scalar coefficient for the large-scale fading factor that includes channel path-loss and shadowing effects, and $\mathbf{h}_{m,k} \in \mathbb{C}^{N\times 1}$ is the small-scale fading factor whose elements are i.i.d with $\mathcal{CN}(0,1)$. In training phase, the m-th AP receives the reference signals and correlates with the known reference sequence $\boldsymbol{\phi}_k \in \mathbb{C}^{\tau\times 1}$ to obtain the k-th user's channel as follows

$$\mathbf{y}_m = \left(\sqrt{\rho_u\tau}\sum_{s=1}^{K}\mathbf{g}_{m,s}\boldsymbol{\phi}_s^{\mathrm{H}}\right)\boldsymbol{\phi}_k + \mathbf{n}_m, \tag{1}$$

where ρ_u denotes the uplink transmit power, τ is a length of reference sequence, $\mathbf{n} \sim \mathcal{CN}(0, \mathbf{I}_N)$ is an additive white Gaussian noise (AWGN). To avoid pilot contamination effect, with condition $\tau \geq K$, we utilize mutually orthogonal reference signals, i.e. $\boldsymbol{\phi}_i^{\mathrm{H}}\boldsymbol{\phi}_j = \delta_{ij}$. Assuming use of the linear minimum mean square error (LMMSE) for the channel estimation, we can obtain the estimate of channel as

$$\hat{\mathbf{g}}_{m,k} = \frac{\sqrt{\rho_u\tau}\beta_{m,k}}{1+(\rho_u\tau\beta_{m,k})}\mathbf{y}_m. \tag{2}$$

Here, the fact that the estimate of channel and estimation error are uncorrelated is well known, and their distributions can be described by

$$\hat{\mathbf{g}}_{m,k} \sim \mathcal{CN}(0, \eta_{m,k}\mathbf{I}_N), \tag{3}$$
$$\tilde{\mathbf{g}}_{m,k} \sim \mathcal{CN}(0, (\beta_{m,k} - \eta_{m,k})\mathbf{I}_N), \tag{4}$$

where $\tilde{\mathbf{g}}_{m,k} = \mathbf{g}_{m,k} - \hat{\mathbf{g}}_{m,k}$ denotes the estimation error and $\eta_{m,k} = \tau_p p_p \beta_{m,k}^2/(1 + \tau_p p_p \beta_{m,k})$. Then, with the estimate of channel, linear precoding techniques such as maximal ratio combining (MRC) can be applied at the APs.

In considering the practicality of CF-mMIMO system, we factor in the limitations of fronthaul links, characterized by constrained number of available bits as follows

$$\sum_{m=1}^{M} b_m \leq B_{\max}, \tag{5}$$

where $B_{\max}$ represents the maximun number of bits accessible to the CPU, and b_m indicates the bit allocation for the fronthaul link of the m-th AP. Due to the finite capacity, it is necessary for the APs and the CPU to compress the data before it is transmitted over the fronthaul links. Thus, we employ an additive quantiation noise model (AQNM)-based quantization to represent data compression. Relying upon the analyses from [9], the output of AQNM quantizer can be expressed as

$$\mathcal{Q}(x) = (1 - \alpha)x + \varepsilon_q, \tag{6}$$

where α refers to the scaling factor, and ε_q denotes the uncorrelated additive quantization noise. The value of α depends on the number of allocated bits for quantization and converges to zero when infinite bits are available. The number of quantization bits also affects the variance of quantization noise as follows

$$\mathbb{E}\left\{|\varepsilon_q|^2\right\} \approx \alpha(1 - \alpha)C_x, \tag{7}$$

where $C_x = \mathbb{E}\{xx^*\}$ is the variance of input signal. Note that the quantization noise variance encompasses the power of the input signal. Thus, the normalized variance of the quantization noise is calculated using the formula $\sigma_q^2 = \mathbb{E}\left\{|\varepsilon_q|^2\right\}/C_x$. Table I shows the designated values for the parameters α and σ_q^2 corresponding to the number of bits.

III. Bit Allocation for Fronthaul Links

In this section, we analyze the effect of the scaling factor and quantization noise in terms of a signal-to-interference-plus-noise ratio (SINR) and propose a bit allocation algorithm based on the HS algorithm to enhance the sum-rate of CF-mMIMO system. We consider the MRC beamforming made by the estimate of channel $\hat{\mathbf{g}}_{m,k}$ at the APs for the local estimation of the k-th user signal, and the CPU linearly combines the local estimates with weight coefficients $\mathbf{w}_k = [w_{1,k}, \cdots, w_{m,k}]^{\mathrm{T}} \in \mathbb{C}^{M\times 1}$. Let the estimated uplink signal of the k-th user at the m-th AP be $\ell_{m,k}$. Then, $\ell_{m,k}$ is represented as

TABLE I
Quantization Noise and Its Variance of Optimal Unifrom Quantizer with Normalized Input [10]

b	σ_q^2	α
1	0.2313	0.3634
2	0.10469	0.11885
3	0.036037	0.03744
4	0.011409	0.01154
5	0.003482	0.003490

$$\ell_{m,k} = \hat{\mathbf{g}}_{m,k}\left(\sum_{s=1}^{K}\sqrt{\rho_u}\mathbf{g}_{m,s}x_s + \mathbf{n}_m\right), \tag{8}$$

where x_k is the k-th user's symbol satisfying $\mathbb{E}\{|x_k|^2\} = 1$.

Since locally obtained signals are, after compression, passed to the CPU through limited fronthaul links, those signals are distorted by the scaling factor and quantization noise. Then, the k-th user's collective signal with weight coefficients at the CPU can be formulated as

$$r_k = \sum_{m=1}^{M} w_{m,k}\mathcal{Q}(\ell_{m,k}) = \sum_{m=1}^{M} w_{m,k}((1 - \alpha_m)\ell_{m,k} + \varepsilon_{m,q}). \tag{9}$$

Note that the APs can be allocated different number of bits, leading to the distinct scaling factors and quantization noise levels for each AP. This consideration is captured by the notations, i.e., α_m and $\varepsilon_{m,q}$. Given the received signal at the CPU, we can compute SINR of CF-mMIMO with limited fronthaul links. For the sake of tractable analysis, we utilize the use-and-then-forget (UatF) bound [11] for deriving the closed-form expression of SINR. Then, the SINR expression is formulated as (10) shown at the bottom of this page. Derivation details can be found in [7] and omitted in this paper due to the space limitation. Variables in (10) are defined as follows

$$\boldsymbol{\eta}_k = [\eta_{1,k}, \cdots, \eta_{M,k}]^{\mathrm{T}} \in \mathbb{C}^{M\times 1}, \tag{11}$$
$$\boldsymbol{\Pi}_k = \mathrm{diag}\left([\eta_{1,k}, \cdots, \eta_{M,k}]^{\mathrm{T}}\right) \in \mathbb{C}^{M\times M}, \tag{12}$$
$$\boldsymbol{\Delta}_{ks} = \mathrm{diag}\left([\eta_{1,k}\beta_{1,s}, \cdots, \eta_{M,k}\beta_{M,s}]^{\mathrm{T}}\right) \in \mathbb{C}^{M\times M}, \tag{13}$$
$$\mathbf{Q} = \mathrm{diag}\left([(1 - \alpha_1), \cdots, (1 - \alpha_M)]^{\mathrm{T}}\right) \in \mathbb{R}^{M\times M}. \tag{14}$$

Given the bit allocation, i.e., given $\{\alpha_1, \cdots, \alpha_M\}$, the only factor that affects the SINR values is the weight coefficients for linear combination at the CPU. By introducing concise

$$\mathrm{SINR}_k = \frac{\rho_u N^2 \mathbf{w}_k^{\mathrm{H}}(\mathbf{Q}^{\mathrm{H}}\boldsymbol{\eta}_k\boldsymbol{\eta}_k^{\mathrm{H}}\mathbf{Q})\mathbf{w}_k}{\mathbf{w}_k^{\mathrm{H}}\left(\rho_u N^2(\mathbf{I}_N - \mathbf{Q}^{\mathrm{H}})\boldsymbol{\Pi}_k^2 + \sum_{s=1}^{K}\rho_u N\boldsymbol{\Delta}_{ks} + N\boldsymbol{\Pi}_k\right)\mathbf{Q}\mathbf{w}_k}. \tag{10}$$

notations for the components of the SINR, equation (10) can be transformed into a more simplified expression as

$$\mathrm{SINR}_k = \frac{\mathbf{w}_k^{\mathrm{H}}\mathbf{A}_k\mathbf{w}_k}{\mathbf{w}_k^{\mathrm{H}}\mathbf{B}_k\mathbf{w}_k}, \quad (15)$$

where

$$\mathbf{A}_k = \rho_u N^2 \mathbf{Q}^{\mathrm{H}} \boldsymbol{\eta}_k \boldsymbol{\eta}_k^{\mathrm{H}} \mathbf{Q}, \quad (16)$$

$$\mathbf{B}_k = \rho_u N^2 \Big((\mathbf{I}_N - \mathbf{Q}^{\mathrm{H}})\boldsymbol{\Pi}_k^2 + \sum_{s=1}^{K} \frac{\boldsymbol{\Delta}_{ks}}{N} + \frac{\boldsymbol{\Pi}_k}{\rho_u N} \Big)\mathbf{Q}. \quad (17)$$

This reformulation aligns with the structure of the generalized Rayleigh quotient, indicating that the optimal weights can be obtained as

$$\mathbf{w}_k^{\star} = c \cdot \underset{\mathbf{v}}{\mathrm{argmax}} \{\lambda : \mathbf{U}\mathbf{v} = \lambda\mathbf{v}, \mathbf{U} = (\mathbf{B}_k)^{-1}\mathbf{A}_k\}, \quad (18)$$

where c is the constant for normalization.

We now propose a bit allocation algorithm designed for fronthaul link transmission, aimed at improving the SINR. It is noteworthy that the number of allocated bits influences both the desired signal and interference components, as depicted in the numerator and denominator of (10), respectively. Given that this formula does not exhibit convexity, the optimal bit allocation using conventional convex optimization methods becomes challenging. To tackle this issue, we leverage a meta-heuristic approach, specifically the HS method, which is capable of finding a near-optimal solution with low computational complexity. Our proposed algorithm is structured around two core phases: initialization and update.

Initialization: In the initialization phase, the algorithm generates a collection of feasible solutions, termed as harmony memory (HM). In our proposed algorithm, the HM is defined as follows

$$\mathbf{H}_0 = \begin{bmatrix} b_{1,1} & & b_{n,1} \\ \vdots & \cdots & \vdots \\ b_{1,M} & & b_{n,M} \\ f_{\mathrm{eval}}(\mathbf{b}_1) & & f_{\mathrm{eval}}(\mathbf{b}_n) \end{bmatrix} \in \begin{bmatrix} \mathbb{N}^{M \times n} \\ \mathbb{C}^{1 \times n} \end{bmatrix}, \quad (19)$$

where n determines the total size of the initial population. The i-th column of the HM comprises two main components: the variable part, which is the vector of allocated bits $\mathbf{b}_i = [b_{i,1}, \cdots, b_{i,M}]^{\mathrm{T}}$ for all APs, and its associated evaluation metric, denoted by $f_{\mathrm{eval}}(\mathbf{b}_i)$. Since we are focusing on enhancing the sum-rate, the evaluation function is defined as $f_{\mathrm{eval}}(\mathbf{b}_i) = \Sigma_{k=1}^{K} \log_2(1+\mathrm{SINR}_k(\mathbf{b}_i))$. This metric quantifies the performance or suitability of the given bit allocation vector $\mathbf{b}_i$.

Update: The update procedure in the proposed algorithm employs both exploitation and exploration to generate new solutions through two primary methods: random generation and random element selection. The choice between these methods is determined by the Harmony Memory Considering Rate (HMCR), denoted as δ. According to this parameter, the

Algorithm 1 Proposed bit allocation algorithm

Initialization: for a given $B_{\max}$
for $i = 1, 2 \cdots, n$ **do**
  Generate $\mathbf{b}_i$ satisfying constraints (5)
  Compute $\mathbf{w}_k^{\star}$ using (18)
  Compute $f_{\mathrm{eval}}(\mathbf{b}_i)$
end for
Update: for a given δ
for $\mathrm{iter} = 1, 2, \cdots, \mathcal{I}$ **do**
  Generate $\mathbf{b}_{\mathrm{new}}$ using (20)
  Compute $\mathbf{w}_k^{\star}$ using (18)
  Compute $f_{\mathrm{eval}}(\mathbf{b}_{\mathrm{new}})$
  If $f_{\mathrm{eval}}(\mathbf{b}_{\mathrm{new}}) > f_{\mathrm{eval}}(\mathbf{b}_n)$
    $\mathbf{H}_{\mathrm{iter}}$: Replace $\mathbf{b}_n$ with $\mathbf{b}_{\mathrm{new}}$
  Else if $f_{\mathrm{eval}}(\mathbf{b}_{\mathrm{new}}) \leq f_{\mathrm{eval}}(\mathbf{b}_n)$
    $\mathbf{H}_{\mathrm{iter}} = \mathbf{H}_{\mathrm{iter}-1}$
  Sort $\mathbf{H}_{\mathrm{iter}}$ in descending order by evaluation values
end for
Choose the best bit configuration in the last HM, $\mathbf{H}_{\mathcal{I}}$

process for generating a new bit configuration, $\mathbf{b}_{\mathrm{new}}$, can be described as follows

$$\mathbf{b}_{\mathrm{new}} = \begin{cases} [b_{\mathrm{new},1}, \cdots, b_{\mathrm{new},M}] & \text{with probability } 1-\delta, \\ [b_{t_1,1}, \cdots, b_{t_M,M}] & \text{with probability } \delta. \end{cases} \quad (20)$$

The approach of random generation, the first case of (20), enhances the exploration of the solution space by introducing completely new solution elements. Conversely, the random element selection, the second case of (20), is designed to leverage previously successful solutions stored in the HM, which utilizes the exploitation capability. In this random selection, the selection indices $t_1, \cdots, t_M$ are chosen independently for each element, with each t_j taking a value from the set $\{1, 2, \cdots, n\}$. With the random selection, the combined bit configuration might not comply with the constraint on the available number of bits in (5). Thus, the algorithm iteratively deducts one bit from a randomly selected element until the bit configurations become feasible.

Upon generating a new solution, its effectiveness is also gauged through the evaluation function as $f_{\mathrm{eval}}(\mathbf{b}_{\mathrm{new}})$. Subsequently, based on this evaluation, the HM is updated to include superior solutions. Assuming the HM is sorted in descending order by evaluation values as follows

$$f_{\mathrm{eval}}(\mathbf{b}_1) \geq f_{\mathrm{eval}}(\mathbf{b}_2) \geq \cdots \geq f_{\mathrm{eval}}(\mathbf{b}_n), \quad (21)$$

where the n-th solution is considered as the least effective solution. We then compare $f_{\mathrm{eval}}(\mathbf{b}_{\mathrm{new}})$ with $f_{\mathrm{eval}}(\mathbf{b}_n)$ to decide on the inclusion of the new solution. If $f_{\mathrm{eval}}(\mathbf{b}_{\mathrm{new}}) > f_{\mathrm{eval}}(\mathbf{b}_n)$, the HM is updated by replacing the least effective solution with the new one. Otherwise, if $f_{\mathrm{eval}}(\mathbf{b}_{\mathrm{new}}) \leq f_{\mathrm{eval}}(\mathbf{b}_n)$, the current HM is preserved without incorporating the new solution.

The proposed algorithm iteratively updates until reaching a preset number of iterations, $\mathcal{I}$. During iterations, the

TABLE II
COMPUTATIONAL COMPLEXITY ANALYSIS

Algorithm	Complexity
Proposed	$\mathcal{O}\Big((n+\mathcal{I})KM^3\Big)$
Brute-force	$\mathcal{O}\Big(\frac{(M+B_{\max}-1)!}{(M-1)!(B_{\max})!}KM^3\Big)$

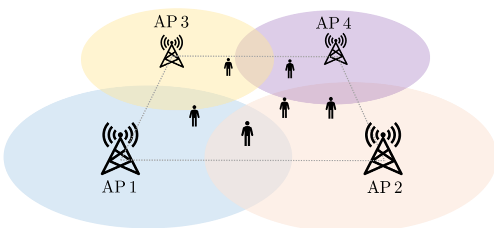

Fig. 1: Simulation setting.

proposed algorithm gradually enhances solutions within the HM, focusing on refining fronthaul bit allocation to improve system performance. The algorithm procedure is summarized in Algorithm 1.

IV. Numerical Results

In this section, we detail the performance of our proposed algorithm through simulation results. For channel generation, the following pathloss model (in [dB] scale) is used [12]

$$\mathrm{PL} = 36.7\log_{10}(d) + 22.7 + 26\log_{10}(f_c) + \mathcal{X}_{\mathrm{shad}}, \quad (22)$$

where d is the distance to the transmitter, f_c denotes the carrier frequency, and $\mathcal{X}_{\mathrm{shad}} \sim \mathcal{N}(0, (4\mathrm{dB})^2)$ represents the shadowing factor. The simulations are configured with a carrier frequency (f_c) of 2.1 GHz, a bandwidth (BW) of 20 [MHz], a transmit power (p_{tx}) of 15 [dBm], and noise power (σ_n^2) defined as $N_0 \times \mathrm{BW}$, with N_0 being -174 [dBm/Hz]. This setup leads to a transmit SNR (ρ_u) calculated by $\frac{p_{\mathrm{tx}}}{\sigma_n^2}$. The algorithm's simulation parameters are intentionally kept modest to ensure low complexity ($B_{\max} = 8$, $n = 10$, $\delta = 0.9$, and $\mathcal{I} = 20$).

We benchmark the performance of our algorithm against two allocation strategies: uniform allocation, which acts as a basic comparison due to its lack of channel information reliance and thus represents a lower performance bound, and brute-force allocation, which assesses all possible bit configurations for the upper performance limit. This comparison aims to illustrate our algorithm's efficiency in striking a balance between computational simplicity and optimal performance. The comparative analysis of computational complexities between the proposed algorithm and the brute-force approach is detailed in Table II.

In Fig. 2, we assess the performance of our proposed algorithm in relation to the number of users, setting the number of APs at $M = 4$ and their antennas at $N = \{32, 64, 128\}$. The

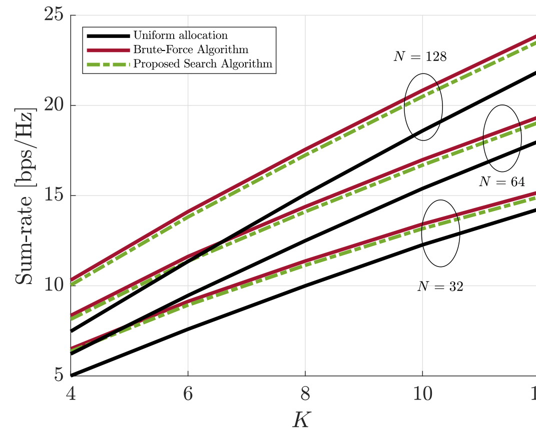

Fig. 2: Sum-rate performances of algorithms with respect to the number of users.

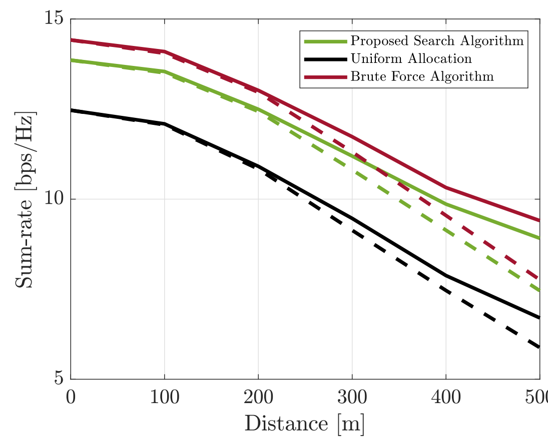

Fig. 3: Sum-rate performances of algorithms with respect to distance along two directions, upward (solid line) and diagonal (dashed line).

figure illustrates that the sum-rate escalates across all methods as the user count increases. Notably, our algorithm exhibits performance that closely achieves the brute-force approach. The disparity between the uniform allocation algorithm and the brute-force algorithm narrows with the support for more users. This is attributed to the equal significance of all APs in scenarios with a more uniformly scattered user setting. Furthermore, we explore the impact of increasing the number of antennas on performance. It is confirmed that augmenting N enhances performance, as more antennas expand the channel dimensions between the AP and the user, thereby improving the likelihood of establishing favorable channel conditions.

In Fig. 3, we analyze the performance of our proposed algorithm across various scenarios where the center of the user distribution shifts away from the origin by distances ranging up to 500 meters. For the simulations, we consider two directions of displacement: upward (solid lines) and diagonal (dashed lines). with user locations uniformly distributed within

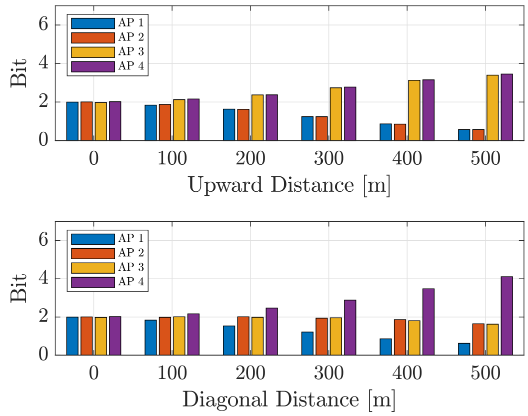

Fig. 4: Average allocated bits of each AP versus the distances along two directions, upward and diagonal.

a 250 m^2 area. Based on Fig. 1, the upward direction is aligned through the area between AP 3 and AP 4, while the diagonal direction heads towards AP 4. According to Fig. 3, the total sum-rate diminishes as the center of the user distribution moves farther from the origin. This decrease occurs because users relocating away from the origin encounter increased pathloss to certain APs, while those near the APs cannot reduce their distance further, leading to decreased performance. The upward shift of the user distribution tends to maintain shorter distances between the APs and users compared to moving diagonally, resulting in a more gradual decline in performance. Despite these differences, the gap between the proposed algorithm and the uniform bit allocation persists in both displacement scenarios.

To examine how the fronthaul bits are distributed by the proposed algorithm, Fig. 4 displays the average number of bits allocated to each AP. The figures reveal that a significant number of bits are allocated to the primary APs nearest to the user distribution, such as AP 3 and AP 4 in the upward direction, and AP 4 in the diagonal direction. While the average allocation might appear uniform at zero distance, this does not imply that the algorithm opts for an equal bit distribution across all APs. Instead, The algorithm optimizes the bit configuration for each specific channel realization, tailoring the allocation to achieve the best possible performance.

V. Conclusion

In this paper, we investigated the effects of bit allocation on fronthaul-limited CF-mMIMO systems. The AQNM quantizer was employed to understand how allocated bits influence quantization noise. Through this exploration, we derived an SINR expression to construct an optimization problem aimed at maximizing the sum-rate by adjusting bit allocations. A HS-based algorithm was proposed to tackle this combinatorial and non-convex problem effectively. The algorithm utilized exploration and exploitation techniques to update bit configurations dynamically. The simulation results demonstrated that the proposed algorithm provided satisfactory performance while significantly reducing computational complexity. This work underscored the importance of strategic bit allocation in enhancing the efficiency and reliability of CF-mMIMO systems, establishing a foundation for future communication systmes such as cloud radio access network (C-RAN) and open radio access network (O-RAN).

VI. Acknowledgement

This research was supported in part by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2024-2020-0-01787) supervised by the IITP (Institute of Information & Communications Technology Planning & Evaluation) and in part by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2021-000269, Development of sub-THz band wireless transmission and access core technology for 6G Tbps data rate).

References

[1] T. L. Marzetta, “Noncooperative Cellular Wireless with Unlimited Numbers of Base Station Antennas,” *IEEE Transactions on Wireless Communications*, vol. 9, no. 11, pp. 3590–3600, Nov. 2010.

[2] H. Q. Ngo, A. Ashikhmin, H. Yang, E. G. Larsson, and T. L. Marzetta, “Cell-Free Massive MIMO versus Small Cells,” *IEEE Transactions on Wireless Communications*, vol. 16, no. 3, pp. 1834–1850, Mar. 2017.

[3] K. T. Truong and R. W. Heath, “The Viability of Distributed Antennas for Massive MIMO Systems,” in *2013 Asilomar Conference on Signals, Systems and Computers*, Nov. 2013, pp. 1318–1323.

[4] V. N. Ha, L. B. Le, and N.-D. Dao, “Coordinated Multipoint Transmission Design for Cloud-RANs with Limited Fronthaul Capacity Constraints,” *IEEE Transactions on Vehicular Technology*, vol. 65, no. 9, pp. 7432–7447, Sep. 2016.

[5] H. Masoumi and M. J. Emadi, “Performance Analysis of Cell-Free Massive MIMO System with Limited Fronthaul Capacity and Hardware Impairments,” *IEEE Transactions on Wireless Communications*, vol. 19, no. 2, pp. 1038–1053, Feb. 2020.

[6] I.-S. Kim, M. Bennis, and J. Choi, “Cell-Free mmWave Massive MIMO Systems with Low-Capacity Fronthaul Links and Low-Resolution ADC/DACs,” *IEEE Transactions on Vehicular Technology*, vol. 71, no. 10, pp. 10 512–10 526, Oct. 2022.

[7] M. Bashar, K. Cumanan, A. G. Burr, H. Q. Ngo, E. G. Larsson, and P. Xiao, “Energy Efficiency of the Cell-Free Massive MIMO Uplink with Optimal Uniform Quantization,” *IEEE Transactions on Green Communications and Networking*, vol. 3, no. 4, pp. 971–987, Dec. 2019.

[8] Z. W. Geem, J. H. Kim, and G. V. Loganathan, “A New Heuristic Optimization Algorithm: Harmony Search,” *Simulation*, vol. 76, no. 2, pp. 60–68, Feb. 2001.

[9] A. Mezghani and J. A. Nossek, “Capacity Lower Bound of MIMO Channels with Output Quantization and Correlated Noise,” in *Proceedings of IEEE International Symposium on Information Theory*, 2012, pp. 1–5.

[10] J. Max, “Quantizing for Minimum Distortion,” *IRE Transactions on Information Theory*, vol. 6, no. 1, pp. 7–12, Mar. 1960.

[11] E. Björnson, J. Hoydis, and L. Sanguinetti, “Massive MIMO Networks: Spectral, Energy, and Hardware Efficiency,” *Foundations and Trends in Signal Processing*, vol. 11, no. 3-4, pp. 154–655, Nov. 2017.

[12] Report ITU-R M.2412-0, “Guidelines for Evaluation of Radio Interface Technologies for IMT-2020,” ITU-R, Tech. Rep., 2017.