

Joint Topology and Cost Optimization for Segmented Fronthaul in User-Centric Distributed Massive MIMO Systems

Felipe J. F. Pinto, Jorge G. S. Costa, André L. P. Fernandes,
André M. Cavalcante and João C. W. A. Costa

Abstract—This paper investigates topology optimization and deployment cost modeling in a user-centric distributed Massive MIMO architecture, also known as cell-free networks, where transmission and reception points (TRPs) are interconnected and linked to a central processing unit (CPU) through serial fronthaul links. A dimensioning framework is proposed, incorporating cabling and transceiver cost considerations, with optimization focused on reducing serial fronthaul chains and improving CPU placement to minimize cabling length. The solution is compared with a literature baseline under similar assumptions and evaluated under more realistic conditions. Results show the proposed approach outperforms the baseline and remains effective under more practical conditions.

Keywords—Distributed Massive MIMO, Cell-free Networks, Fronthaul, Backhaul, Serial Chains, Deployment Cost

I. INTRODUCTION

User-centric distributed massive MIMO (UC D-mMIMO), also known as cell-free massive MIMO, is a high-density wireless network in which multiple transmission and reception points (TRPs) collaborate to transmit and receive each user's signals coherently. Each TRP has one or more antennas, serves a limited number of users to ensure system scalability, and is connected to a fronthaul network that leads to one or more central processing units (CPUs), allowing signal co-processing and system coordination [1]. This design enhances spectral efficiency and provides more uniform coverage, which are crucial requirements for 6G systems [2].

A systematic problem with UC-D-mMIMO is its deployment complexity, which leads to fronthaul segmentation. This approach uses a compute-and-forward architecture that connects multiple TRPs to a CPU in a serial configuration, improving deployment scalability and preventing a chaotic fronthaul in a high-density deployment [3]. Additionally, advanced distributed processing techniques have been explored that take advantage of this serial fronthaul, achieving spectral efficiency that is comparable to centralized processing [4]. Consequently, segmentation simplifies deployment and processing tasks.

The cost-effectiveness of this approach has been analyzed in [5] and [6], for outdoor and indoor systems, respectively. In [5], an initial cost assessment was conducted under a simple fronthaul bitrate model based on an aggregate user rate. It was shown that serial architectures are more cost-efficient when the level of serialization is kept low (e.g., five or fewer TRPs per chain). In [6], the focus was on the reliability, as the serial architecture is susceptible to significant disconnections caused by a small set of potential failures. It was shown that a cross-connection failure protection mechanism was the most cost-effective deployment solution. Besides that, the no-protection base scenario becomes more effective when 12 or fewer TRPs per serial chain are considered, trailing slightly behind the cross-connection option in terms of cost-effectiveness.

Despite these advancements, both works rely on simplified, highly uniform deployment models without explicit serial-chain optimization. Recently, [7] proposed a methodology for fronthaul length reduction using a more generic deployment model. The work jointly optimizes TRP serial chain formation and CPU placement. However, the cost analysis assumes fronthaul bitrate scaling with the number of antennas, as in a centralized processing architecture. This does not invalidate the analysis of distributed processing techniques favored by serial chains, since they can be implemented in a centralized manner. However, true distributed operation requires a model in which the bitrate scales with the number of users [8].

In addition to that, [7] did not account for any backhaul considerations, assumed an infinite CPU capacity to support TRPs, and mixed optimization strategies more suitable for short TRP chains with ones effective for longer ones, hindering the resulting fronthaul length optimization. In this context, this study proposes a cost analysis methodology and a scenario construction framework for UC-DmMIMO in outdoor environments, with fronthaul bitrate modeling more suitable for distributed processing architectures, backhaul costs, CPU capacity limitation, and a more concise optimization approach compared to [7].

II. SYSTEM MODEL

We consider a square coverage area of size $R \times R$ m², containing a set of TRPs $\mathcal{L}$ serving K randomly uniformly distributed users, with each element of the set being randomly distributed with coordinates $(x_\ell, y_\ell) \sim \mathcal{U}(0, R)$. The TRPs serve a maximum of τ_p users, which is also the number

Felipe J. F. Pinto¹, Jorge G. S. Costa¹, André L. P. Fernandes², André M. Cavalcante³ and João C. W. A. Costa¹. ¹Applied Electromagnetism Laboratory, Federal University of Pará - UFPA, Belém, Brazil, ²Department of Production Engineering, Federal University of Amazonas, Manaus, Brazil, ³Ericsson Research, Ericsson Telecomunicações Ltda., Indaiatuba, Brazil. E-mails: felipe.pinto@itec.ufpa.br; jorge.costa@itec.ufpa.br; fernandes.andre@ufam.edu.br; andre.mendes.cavalcante@ericsson.com; jweyl@ufpa.br. This work was partially supported by Ericsson Telecomunicações Ltda, CNPq and CAPES.

of orthogonal pilots in the system to use for the channel estimation procedure, and are organized into G mutually exclusive groups, each composed of a subset $\mathcal{G}_i \subset \mathcal{L}$, connected serially. Each serial chain of connections contains $|\mathcal{G}_i|$ distribution fronthaul links, representing the connections between the various TRPs inside $\mathcal{G}_i$, and one feeder fronthaul link, which is connected to a CPU that performs base-band processing, as shown in Figure 1. Finally, the system contains W CPUs, assumed to be connected by backhaul links.

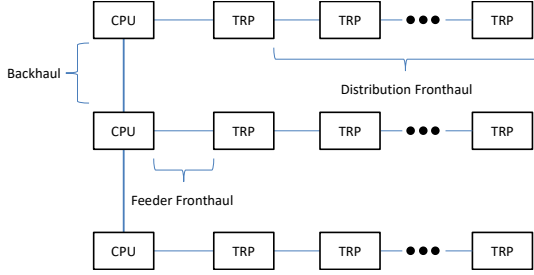


Fig. 1. Schematic illustration of the segmented fronthaul system architecture with distribution fronthaul, feeder fronthaul, and backhaul links.

A. Distribution fronthaul optimization

The formation of the serial chain of connections involves 3 steps.

Step 1: A K-means algorithm is applied to partitions $\mathcal{L}$ into $\{\mathcal{G}_1, \dots, \mathcal{G}_G\}$ preliminary groups. The number of groups is calculated as

$$G = \frac{|\mathcal{L}|}{S_{lvl}} \quad (1)$$

equation where S_{extlvl} is the serialization level that indicates the expected number of TRPs connected in series. The K-means algorithm is run with multiple initializations, and the solution with the smallest sum of squared distances from each point to the centroid of its respective cluster is selected.

Step 2: A cluster balancing procedure is performed to ensure that each serial chain contains a minimum of S_{min} and a maximum of S_{max} TRPs. Clusters exceeding S_{max} elements are partitioned into smaller clusters, while clusters with fewer than S_{min} elements are merged with adjacent groups. This process is performed iteratively until G groups are obtained, each satisfying the size constraints $S_{min} \leq S \leq S_{max}$.

Step 3: Each group $\mathcal{G}_i$ is connected through a serial topology in order to minimize the total length of its internal connections. Formally, the objective is to determine a permutation π_i of the TRPs belonging to group $\mathcal{G}_i$ that minimizes the length of an open path:

$$L_i = \sum_{k=1}^{|\mathcal{G}_i|-1} \|x_{\pi_i(k+1)} - x_{\pi_i(k)}\|. \quad (2)$$

This problem corresponds to the Minimum Hamiltonian Path problem and can be interpreted as an open variant of the *Travelling Salesman Problem* (TSP). In this work, the problem is solved through a two-stage approach:

- 1) **Initial construction (Nearest Neighbor – NN):** For each group $\mathcal{G}_i$, the NN algorithm is executed considering all TRPs as possible starting vertices. For each initial

choice, the path is iteratively constructed by selecting the nearest unvisited AP to the last inserted vertex.

- 2) **Local refinement (2-opt):** The solution generated by NN is refined using the 2-opt algorithm, which performs iterative pairwise edge exchanges whenever such swaps reduce the total path length. The process is repeated until no further improvement is possible or a maximum number of iterations is reached.

Among all solutions obtained from the different initializations, the one yielding the minimum total length is selected.

B. CPU positioning, Feeder fronthaul and backhaul optimization

The CPU positioning, formation of the feeder fronthaul links, and backhaul network involves 4 steps:

Step 1: Initial CPU positioning is performed by computing the geometric centroid of each group $\mathcal{G}_i$ and applying the K-means algorithm with W clusters, where each resulting centroid corresponds to a CPU position, denoted by μ_w , $\forall w \in \{1, \dots, W\}$.

Step 2: Association of the serial chains to a CPU is performed exclusively based on the positions of the TRPs at the endpoints of the serial connections. The distances from these endpoints to the CPUs are computed as

$$d_i^{0,w} = \|p_i^0 - \mu_w\|, \quad d_i^{1,w} = \|p_i^1 - \mu_w\|. \quad (3)$$

where p_i^0 denotes the first endpoint of the serial chain $\mathcal{G}_i$ and p_i^1 denotes the last endpoint. Based on these distances, the leader AP of the serial chain $\mathcal{G}_i$ is defined as the endpoint with the minimum distance to any CPU, i.e.,

$$\ell_i = \arg \min_{p \in \{p_i^0, p_i^1\}} \left(\min_{w=1, \dots, W} \|p - \mu_w\| \right). \quad (4)$$

The CPU associated with the serial chain $\mathcal{G}_i$ is then given by

$$w_i = \arg \min_{w=1, \dots, W} \|\ell_i - \mu_w\|. \quad (5)$$

Step 3: Final CPU positioning is derived through the leader TRPs assigned to each CPU. For each CPU w , let $\mathcal{T}_w = \{\ell_i \mid w_i = w\}$ denote the set of positions where the feeder fronthaul links connect to the first TRP in each serial chain $\mathcal{G}_i$. The updated position of CPU w is then obtained as the geometric centroid of the set $\mathcal{T}_w$.

Step 4: The backhaul topology is obtained by considering a fully connected graph with CPUs as nodes and Euclidean distances as weights, which is then post-processed by computing the Minimum Spanning Tree (MST) of this graph using Prim's algorithm.

III. COST MODEL

To evaluate the economic feasibility of the proposed serial chain association and CPU positioning, a two-tier cost model is considered. Tier 1 accounts for the infrastructure lengths of fronthaul, backhaul, and serial connections, whereas Tier 2 addresses transceiver selection.

A. Tier 1: Infrastructure Length Modeling

The Tier 1 total infrastructure length is calculated as

$$L_{T1} = \sum_{w=1}^W \sum_{\ell_i \in \mathcal{T}_w} \|\ell_i - \mu_w\| + \sum_{i=1}^G L_i^{\text{NN}+2\text{opt}} + d_{\text{MST}}^{\text{BH}}, \quad (6)$$

where $L_i^{\text{NN}+2\text{opt}}$ is the two-step solution minimizing length in a serial chain $\mathcal{G}_i$ in meters as calculated on step 3 of Subsection II-A and $d_{\text{MST}}^{\text{BH}}$ denotes the total backhaul length in meters obtained from the MST connecting all CPUs. Therefore, the Tier 1 cost can be expressed as

$$C_{T1} = \eta \cdot L_{T1}, \quad (7)$$

where η represents the cost per meter.

B. Tier 2: Transport Network Capacity Modeling and Transceiver Selection

Tier 2 cost can be expressed as

$$C_{T2} = G \left(\sum_{j=1}^{S_{\text{vl}}} \text{Pr}_T(R_{\text{front}}^j) \right) + (W - 1) \text{Pr}_T(R_{\text{back}}) \quad (8)$$

where $\text{Pr}_T(\cdot)$ represents the price of the transceiver required for a required rate, R_{front}^j is the fronthaul rate of segment j , and R_{back} is the corresponding backhaul rate.

To determine fronthaul and backhaul rates, it is necessary to estimate the maximum number of users that a serial chain of TRPs can support, as rates scale with user count under distributed processing for d-MIMO [8]. To this end, the user density in the analyzed area is defined as $\lambda_u = K/R^2$. Assuming that users are uniformly distributed, the expected coverage radius that each TRP must support in order to serve τ_p users is given by

$$r = \sqrt{\frac{\tau_p}{\pi \lambda_u}}. \quad (9)$$

From this, the expected number of users served by a serial chain of TRPs $\mathcal{G}_i$ can be obtained by multiplying the user density λ_u by the union area of the coverage regions of all TRPs in the chain. Appropriate transceiver selection must consider a worst-case scenario, i.e., the maximum expected bitrates. This occurs when the union area is maximized, which, for a fixed number of TRPs, corresponds to a linear arrangement that minimizes the overlap between coverage regions.

In this case, the union area can be expressed in closed form as

$$A_{\text{union}} = \begin{cases} \pi r^2 + 2r(S_{\text{vl}} - 1)d, & d \leq 2r \text{ and } S_{\text{vl}} > 1, \\ S_{\text{vl}} \pi r^2, & \text{otherwise,} \end{cases} \quad (10)$$

where $d = \sqrt{\frac{R^2}{|\mathcal{L}|}}$ represents the expected inter-site distance (ISD) assuming a uniform spatial distribution of TRPs over the area of interest.

As an additional consideration for designing chains under worst-case conditions, the coverage radius r can be conservatively adjusted to account for regions of higher-than-average

user density. This is achieved by modeling the number of users within a given coverage area as a binomial random variable $N_c \sim \text{Binomial}\left(K, \frac{\pi r^2}{R^2}\right)$, and by determining a value $K_{\text{high}}^{(\text{circle})}$ greater than τ_p that occurs with small probability. Specifically,

$$\mathbb{P}\left(N_c \leq K_{\text{high}}^{(\text{circle})}\right) \geq \epsilon, \quad (11)$$

where ϵ defines the confidence level between 0 and 1. An ϵ of 0.99 means that, under the average density assumption, the selected transceiver supports 99% of user density variations. This does not directly correspond to a 1% blocking probability, but is likely smaller, as the remaining configurations correspond to extremely high-density cases that occur with very low probability under typical spatial models.

Then, the corresponding high-density coverage radius is given by

$$r_{\text{high}} = \sqrt{\frac{\tau_p}{\pi \lambda_u^{\text{high}}}}, \quad (12)$$

where $\lambda_u^{\text{high}} = \frac{K_{\text{high}}^{(\text{circle})}}{\pi r^2}$ is the high user density margin, and the final user count expected in a serial chain for transceiver design is

$$K_{\text{chain}} = A_{\text{union}, r_{\text{high}}} \lambda_u^{\text{high}}, \quad (13)$$

where $A_{\text{union}, r_{\text{high}}}$ is calculated as in (10) but with r_{high} instead of r .

At the CPU level, the number of users associated with each CPU can be approximated by a grid with S_{vl} columns, and an expected number of rows per CPU given by

$$N_{\text{sb}}^{\text{CPU}} = \frac{|\mathcal{L}|}{S_{\text{vl}} W}. \quad (14)$$

The grid points are spaced by d , and the same circle union approach is applied using either r or r_{high} . In this work, the area of this union is obtained numerically using MATLAB 2025a. When a non-integer $N_{\text{sb}}^{\text{CPU}}$ occurred, linear interpolation between the nearest integer grid configurations was applied. The corresponding user counts are then computed using λ_u or λ_u^{high} , respectively.

For backhaul traffic, a simple overhead factor α is considered over the expected number of users per CPU, such that the backhaul user load is given by

$$U_{\text{BH}} = (1 + \alpha) U_{\text{CPU}}, \quad (15)$$

where U_{CPU} denotes the expected number of users associated with a CPU.

The estimated number of users is then used to determine the required fronthaul and backhaul rates, which can be calculated as [6]

$$R = 2B \left(1 - \frac{\tau_p}{T_c}\right) (u \cdot N_{\text{bits}}), \quad (16)$$

where B is the system bandwidth, τ_p is the number of pilot symbols, T_c is the coherence block length, N_{bits} is the number of bits per symbol, and u is the number of users whose data is transported through the considered link. In the fronthaul case, $u = K_{\text{chain}}$, resulting in segment rates R_{front}^j , while in the backhaul case $u = U_{\text{BH}}$, resulting in the aggregated rate R_{back} . The factor 2 accounts for in-phase and quadrature components.

Furthermore, in the fronthaul case, different values of K_{chain} can be evaluated by varying S_{IVL} in (10) from 1 up to its maximum value, enabling segment-wise transceiver dimensioning along the chain, i.e., the evaluation of R_{front}^j .

C. Total Cost

Finally, the total network deployment cost is obtained by combining both tiers, resulting in

$$C_{\text{total}} = C_{\text{T1}} + C_{\text{T2}}, \quad (17)$$

where C_{T1} represents the infrastructure-related cost, including fronthaul, backhaul, and serial connections, and C_{T2} corresponds to the cost associated with transceiver selection and capacity provisioning.

IV. NUMERICAL RESULTS

A. Case Study

The considered study setup is a 2×2 km area with 100 users and variable number of TRPs, each serving $\tau_p = 10$ users and being organized in such a way that the expected number of TRPs per serial chain is 5 or 10 [7], [9]. The serial segments in each chain use single mode optical fiber communication with transceiver adjusted adequate capacity. In addition to that, for the fronthaul bitrate calculations, the system bandwidth is fixed at $B = 100$ MHz, the coherence block length is $T_c = 200$ samples, and the number of bits per sample transmitted over the fronthaul (N_{bits}) is set to either 3 or 6.

For the cost analysis, we consider a cost unit (CU) based on the price per meter for aerial fiber installation. Using [10], the median cost is taken as US\$34.11 per meter, obtained by scaling the reported US\$21.49 per meter for labor and materials to include make-ready, engineering, and permitting expenses. The cost of underground fiber installation is set to 2.33 CUs, based on the corresponding trenching median cost from the same reference, similarly adjusted for engineering and permitting expenses.

Four types of optical transceivers in terms of bitrate capacity are considered, and their corresponding prices are summarized in Table I. The prices are the values returned by the function $\text{Pr}_T(R_{\text{front}}^j)$ for the required fronthaul rate R_{front}^j , which returns the cost of the lowest-capacity transceiver whose capacity is greater than or equal to the required [11].

TABLE I
NORMALIZED TRANSCEIVER COSTS

100 Gbps	40 Gbps	25 Gbps	10 Gbps
13.47	9.65	2.17	0.94

For comparison purposes, we analyze three scenarios: A, B, and C. Scenario A involves a setup with 8 CPUs, with TRP serial chain formation and CPU placement exactly as in [7]. Scenario B also has 8 CPUs but employs the 2-opt method discussed in Subsection II-A for the TRP serial chain formation, along with the proposed CPU placement described in Subsection II-B. Scenario C builds upon Scenario B by introducing a physical constraint on CPU capacity, which results in a variable number of CPUs, specifically with capacities of 12, 24 and 32 TRPs considered.

B. Results

Fig. 2 illustrates the cabling costs of fronthaul for both aerial and trenching deployments in networks ranging from 100 to 1000 TRPs. In scenarios A, B, and C, aerial deployment costs remain below 100 kCU for all network sizes. In contrast, trenching deployment costs stay below this threshold only up to a maximum network size of approximately 460–485 TRPs in scenario A, 570–600 TRPs in scenario B, and 650–700 TRPs in scenario C, depending on the serialization level. Regarding this parameter, using 10 TRPs in series is approximately 17–22% cheaper than using 5 TRPs in scenarios A and B, depending on the network size. Scenario C is more variable, with 10 TRPs resulting in lower costs in most cases. When comparing the baseline scenarios, Scenario B achieves cost reductions of about 6–12% relative to Scenario A, demonstrating the effectiveness of the proposed optimization methodology under identical deployment conditions. Scenario C further enhances cost efficiency, offering additional savings of approximately 5–12% compared to Scenario B and 10–18% compared to Scenario A.

Fig. 3 presents the backhaul cabling cost. For Scenarios A and B, the costs are constant and similar, so CPU repositioning in Scenario B affects only the fronthaul. In Scenario C, the cost increases with TRP count, due to limited CPU capacity, surpassing A and B at around 180 TRPs. However, as more CPUs are deployed, they are increasingly placed between

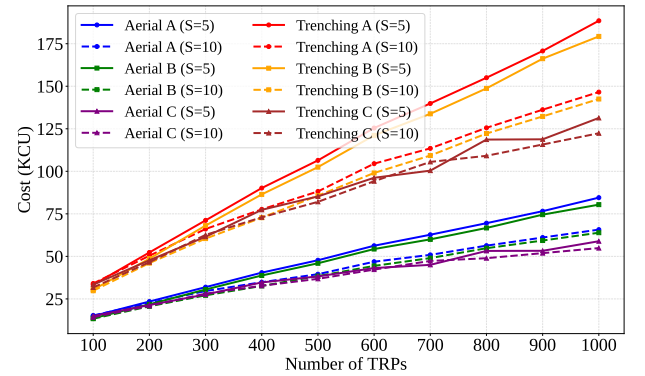


Fig. 2. Fronthaul cost for Scenarios A, B, and C, considering both aerial and trenching deployment strategies, as a function of the number of TRPs.

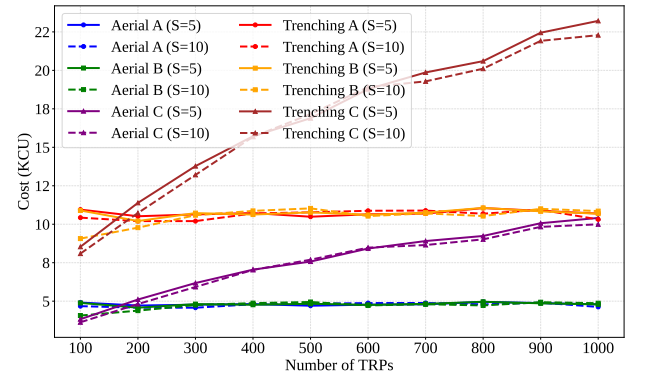


Fig. 3. Backhaul cost for Scenarios A, B, and C, considering both aerial and trenching deployment strategies, as a function of the number of TRPs.

existing ones, contributing less to cost growth. Overall, the increase in Scenario C does not offset its fronthaul savings within Tier 1 costs.

Fig. 4 shows the total transceiver cost for both fronthaul and backhaul. The overall cost remains similar across Scenarios A, B, and C, with a slightly higher level in C due to the dependency of CPU and TRP counts. Interestingly, the curves have a general growth tendency, but do not increase monotonically, as in some small intervals the cost decreases with TRP count. This behavior arises from the trade-off between CPU/TRP counts and chain configuration, which vary with TRP count, leading to shifts in fronthaul demand that, when close to transceiver capacity limits, can trigger transitions from higher-cost transceivers to lower-cost ones.

Finally, Fig. 5 presents the total cost (Subsection II-B) under different capacity configurations. The figure demonstrates that increasing the capacity from 12 to 32 TRPs per CPU substantially reduces the total cost. This reduction is primarily driven by the need for fewer CPUs, which directly lowers the number of transceivers and associated infrastructure costs. Although there is a small cost difference between serializations ($S=5$ and $S=10$) across all capacity levels, capacity is the key factor influencing total cost, rather than serialization. This trend is observed for both aerial and trenching deployments, with trenching consistently incurring higher costs due to installation

expenses.

V. CONCLUSIONS

This work presents a topology-length optimization scheme for segmented fronthaul in UC-mMIMO systems that improves serial chain formation and CPU placement, along with cost and system models that are better suited for a more realistic evaluation of distributed processing architectures. The results are evaluated against a literature baseline, considering both cabling and transceiver costs. The results show that the proposed scheme outperforms the baseline under unlimited CPU capacity, as typically assumed in the literature. When practical CPU constraints are considered, the proposed scheme achieves further performance gains, as increased backhaul costs from additional CPUs are offset by reduced fronthaul length and improved transceiver selection. Future work may consider more realistic fronthaul cabling distances, beyond Euclidean assumptions, as well as hybrid wired-wireless fronthaul solutions.

REFERENCES

- [1] Ö. Demir, E. Björnson, and Sanguinetti, *Foundations of User-Centric Cell-Free Massive MIMO*. Foundations and Trends® in Signal Processing, 2021, vol. 14, no. 3-4.
- [2] H. Q. Ngo, G. Interdonato, E. G. Larsson, G. Caire, J. G. Andrews, "Ultradense Cell-Free Massive MIMO for 6G: Technical Overview and Open Questions", *Proceedings of the IEEE*, vol. 112, no. 7, pp. 805–831, July 2024.
- [3] G. Interdonato, E. Björnson, H. Q. Ngo et al., "Ubiquitous Cell-Free Massive MIMO Communications", *EURASIP J. Wireless Commun.*, vol.2019, no.1, Aug. 2019.
- [4] Z. H. Shaik, E. Björnson and E. G. Larsson, "MMSE-Optimal Sequential Processing for Cell-Free Massive MIMO With Radio Stripes," in *IEEE Transactions on Communications*, vol. 69, no. 11, pp. 7775–7789, Nov. 2021.
- [5] L. Furtado, A. Fernandes, A. Ohashi, F. Farias, A. Cavalcante and J. Costa, "Cell-Free Massive MIMO Deployments: Fronthaul Topology Options and Techno-Economic Aspects," *Proc. 16th European Conf. on Antennas and Propagation (EuCAP)*, Madrid, Spain, 2022, pp. 1–5.
- [6] J. G. S. Costa, Y. K. C. Costa, A. L. P. Fernandes, A. M. Cavalcante, and J. C. W. A. Costa, "Network Survivability in User-Centric Distributed Massive MIMO with Segmented Fronthaul: Centralized and Distributed Indoor Processing" in in *Proc. Braz. Symp. on Telecommun. Signal Process. (SBrT)*, 2025.
- [7] A. S. Mohammed, K. S. Tharakan, H. A. Ammar, H. ElSawy and H. S. Hassanein, "Fronthaul Network Planning for Hierarchical and Radio-Stripes-Enabled CF-mMIMO in O-RAN," in *IEEE Transactions on Wireless Communications*, vol. 25, pp. 14781–14796, 2026.
- [8] E. Björnson and L. Sanguinetti, "Making Cell-Free Massive MIMO Competitive With MMSE Processing and Centralized Implementation," in *IEEE Transactions on Wireless Communications*, vol. 19, no. 1, pp. 77–90, Jan. 2020.
- [9] E. Björnson and L. Sanguinetti, "Scalable Cell-Free Massive MIMO Systems," in *IEEE Transactions on Communications*, vol. 68, no. 7, pp. 4247–4261, July 2020.
- [10] Fiber Broadband Association and Cartesian, *Fiber Deployment Cost Annual Report 2024*. Washington, DC, USA, 2024. [Online]. Available: <https://fiberbroadband.org/resources/2024-fiber-deployment-cost-annual-report/>
- [11] FS, "Optical Transceivers." [Online]. Available: <https://www.fs.com/>. Accessed: Mar. 2026.

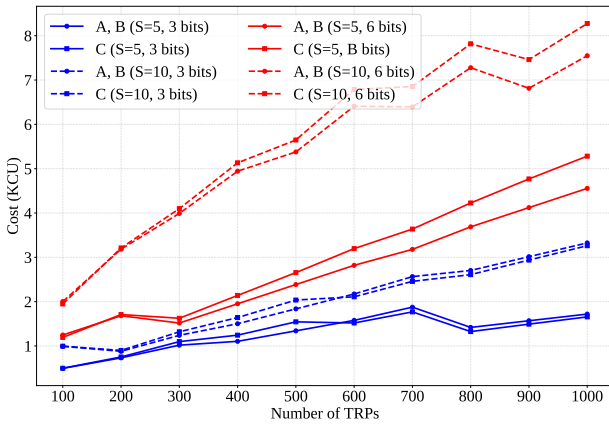


Fig. 4. Transceiver cost as a function of the number of TRPs for A, B and C CPU configurations, considering 3-bit and 6-bit quantization levels.

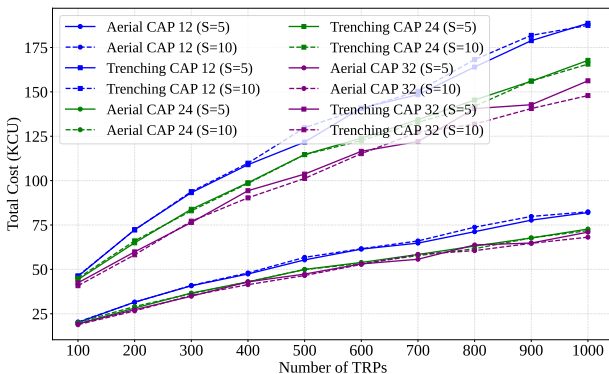


Fig. 5. Total Cost for Scenario C with Different CPU Capacities as a function of the number of TRPs.