Strategic Deployment and Optimization of Cloud-RAN Architectures for Future Wireless Networks

Youssef Katif [1,2], Youssouf Hadhbi [1], Ivana Ljubić [3], Sonia Vanier [2] and Jean Christophe Albenque [1]

***Abstract*— This study investigates optimization models and algorithms aimed at enhancing the deployment of Cloud Radio Access Network architectures, a key enabler for next-generation wireless communication systems. This study addresses key challenges such as efficient data routing, effective resource management, and the scalability of the network infrastructure. To achieve this, we aim to optimize the placement of Distributed Units and Centralized Units, along with the data routing processes, to minimize latency and maximize data transfer efficiency. The problem is initially formulated as a combinatorial optimization problem for which a compact model based on an integer linear programming formulation is proposed. Based on this model, we develop an optimization framework aimed at identifying the most efficient strategies for deploying Cloud-RAN. Furthermore, we conduct computational experiments across various instances and scenarios to evaluate the effectiveness of the proposed approach.**

I. INTRODUCTION

Data traffic in telecommunication networks is expected to increase by up to $10,000$ times by 2030 [7], which poses a significant challenge to traditional *Radio Access Networks* (RANs). These networks are therefore shifting towards cloudified and open solutions, with *Cloud-RAN* being a key technology supported by various initiatives [6]. They bring together a diverse range of academic and industrial partners dedicated to advancing Mobile Cloud Computing [2] through open, flexible, and programmable RAN architectures [13]. Its goal is to overcome existing limitations by fostering more adaptable, cost-efficient, and interoperable network deployments [2]. This evolution is motivated by the need to enhance connectivity and meet the surging demand for digital services. Users increasingly expect higher quality of service (QoS), while network operators face complex challenges related to capacity, performance, and resource management. To deal with this, cloudification offers promising solutions, including increased programmability, automation, and embedded intelligence within networks. These advancements enable operators to customize services based on user needs, automate management and deployment processes, and optimize network performance in real-time.

The transition from traditional base stations (BBUs) to cloud-based architectures involves decomposing traditional base stations into functional components such as Distributed Units (DUs) and Centralized Units (CUs). In this architecture, the CU can be connected to several DUs, forming a star topology [14]. This segmentation allows for more efficient resource allocation and protocol function management.

In this context, Cloud-RAN features a fully centralized architecture where baseband processing is decomposed it into functional units: Distributed Units (DUs), located near antennas, handle local radio processing, while Centralized Units (CUs) manage higher-level control functions from data centers. In the following section, we address the critical deployment challenges in Cloud-RAN, motivated by the need for cost-effective, scalable, and reliable network solutions to support the increasing demand for high-quality wireless services.

II. DEPLOYMENT CHALLENGES AND OPTIMIZATION IN CLOUD-RAN

Despite the advantages of Cloud-RAN, deploying this architecture involves several technical and operational challenges that need to be addressed to ensure efficient and reliable network performance. Two primary optimization problems arise in Cloud-RAN deployment: the *placement of DUs and CUs*, and the *routing* of data across the network. The placement of CUs and DUs needs to consider constraints such as latency (affected by the distance between units), processing power, storage capacity, and bandwidth. Ensuring low latency is critical for real-time applications like video streaming and online gaming. Sufficient computational resources at Edge Clouds and data centers are essential to meet processing demands, while adequate storage and high-bandwidth links (fiber optics) are necessary for efficient data transmission.

Additionally, efficient routing of data across the network is crucial, requiring seamless interactions between base stations, Radio Units (RUs), DUs, CUs, and core routers to ensure reliable and secure data transmission (see Fig. 1).

There exist two well-known concepts called *distributed Cloud-RAN* and *centralized Cloud-RAN* [3]. In a centralized Cloud-RAN architecture, data from multiple base stations is aggregated and processed at a central location, which enables efficient resource management, simplified network updates, and improved coordination across the network. However, this approach can introduce higher latency due to the long-distance data transmission, which may impact latency-sensitive applications. On the other hand, a distributed Cloud-RAN architecture processes data locally at the edge, closer to the users, significantly reducing latency and improving real-time performance. The drawback is that this setup can

*This work was supported by Orange Research

[1] Orange Research, 92320 Châtillon, France firstname.lastname@orange.com

[2] Ecole Polytechnique, IP Paris, LIX UMR 7161, 91120 Palaiseau, France firstname.lastname@polytechnique.edu

[3] ESSEC Business School, 95000 Cergy-Pontoise, France ivana.ljubic@essec.edu

be more complex to manage, with increased infrastructure costs and potential difficulties in maintaining consistency and coordination across multiple edge sites.

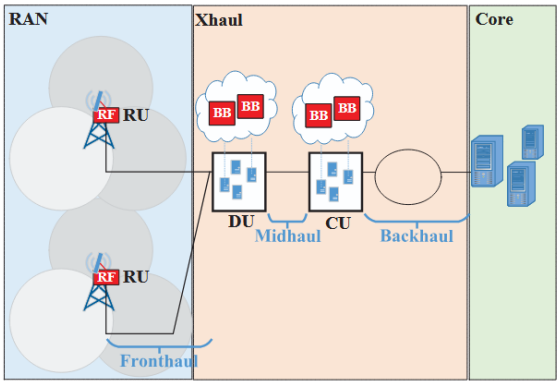

Fig. 1. Cloud-RAN architecture for 5G [13].

This ongoing transformation draws the interest of new vendors and operators, driving innovation and competition within the telecommunications sector. In France and Spain, *Orange* prioritize network virtualization within their "*Lead the Future*" strategy, viewing cloudification as a key enabler for delivering higher quality services.

This landscape highlights the critical importance of strategic design and optimization in Cloud-RAN architectures to meet the growing demands for performance and reliability, while also accommodating technological advancements.

In this work, we model the deployment of Cloud-RAN as a *combinatorial optimization* problem and address it using an optimization framework built on efficient mathematical models and algorithms. This approach considers several real-world inputs and incorporates various technological constraints that have often been neglected in the existing literature but are critical for operators today. This study aims to address the challenge of optimizing Cloud-RAN deployment, with a focus on improving network performance and reducing operational costs in the context of growing demands in modern wireless communication.

Furthermore, the computational complexity of this problem is briefly discussed, and numerical experiments are conducted on various instances to assess the performance of the proposed solution method.

The following section reviews some related works.

III. STATE OF THE ART

The virtualization and disaggregation of RAN into DUs and CUs have gained attention for enhancing flexibility, scalability, and cost-efficiency. The main challenge is balancing deployment costs, resource utilization, and latency requirements.

Several studies focus on minimizing deployment costs [3]. For example, Ndao et al. [9] used an *integer linear programming* (ILP) formulation for DU placement with a fixed CU, limiting flexibility. Njanda et al. [11] jointly optimized DU and CU placement, showing that stricter latency increases deployment costs; they used MILP for small cases and a GRASP heuristic for larger ones. Murti et al. [8] explored multi-CU setups with flexible splits, solved via *Benders decomposition*, balancing cost and centralization. Hojeij et al. [4] maximized user admission under QoS constraints, combining ILP with an *Recurrent Neural Network heuristic* for faster solutions, though their model simplifies latency and bandwidth considerations. Klinkowski [5] proposed a MILP considering all network links, aiming to minimize active sites, but pre-computed routing limits optimality.

Despite significant research efforts, many existing models rely on simplifying assumptions, such as fixing DU or CU locations or focusing only on certain segments of the communication chain. These approaches often overlook processing, queuing delays, and full end-to-end constraints. There is a need for a complete model that captures the entire RU $\rightarrow$ DU $\rightarrow$ CU chain with per-segment distance and latency constraints, as well as capacity considerations over nodes and fiber links.

In this study, we address these gaps with an advanced optimization framework for the Cloud-RAN deployment. We jointly optimize DU and CU placement along with end-to-end traffic routing. We explicitly enforce the RU $\rightarrow$ DU $\rightarrow$ CU precedency and impose per-segment distance limits on both fronthaul and midhaul, along with end-to-end latency constraints, a feature often missing from prior work.

This problem, which combines classical network design and facility location ingredients, constitutes a large-scale optimization problem. It involves the integration of several *NP-hard* subproblems, including *facility location* (determining which sites to activate), and constrained routing (finding paths that satisfy various constraints). The explicit precedence constraints such as RU $\rightarrow$ DU $\rightarrow$ CU and per-segment distance limits significantly increase the problem's complexity compared to classical network design problems. Furthermore, satisfying capacity constraints at each site and along each fiber-link relates to the well-known *Knapsack problem*, which is also NP-hard. Overall, the combination of these NP-hard sub-problems and the intricate constraints makes the Cloud-RAN deployment problem computationally challenging especially for large-scale instances.

To tackle this problem, we first develop a compact formulation based on an integer linear program that incorporates key technological constraints, such as distance, latency, and resource limitations, ensuring a realistic and practical model. Using this, we develop an optimization framework in which the *Branch-and-Cut* (BC) algorithm embedded in a state-of-the-art optimization solver is employed, leveraging its *cutting-plane* techniques to enhance solution efficiency, while ensuring key operational constraints are maintained without compromising solution quality. The computational study has been conducted to analyze the performance of our approach across various instances and scenarios.

The rest of this paper is organized as follows. In Section IV, we present the problem formulation, including the inputs, constraints, and objectives. Section V presents the compact formulation. Numerical results are provided in Section VI.

Finally, Section VII offers conclusions and discusses directions for future work.

IV. PROBLEM STATEMENT

We consider a telecom operator managing a network modeled as a directed graph $G = (V, E)$. Nodes in V represent RU nodes, destination core node, and intermediate nodes used for routing, as well as potential hosting sites such as data centers or aggregation points. Edges in E are fiber links, each with distance d_e (in kms), latency l_e (in ms), and capacity C_e (in Gbps).

The network serves a set R of geographically distributed Radio Units. Each RU $r \in R$ sits at an origin node $o(r) \in V$ and sends traffic to a common destination $t \in V$, which is typically the core network gateway. We assume that all RUs share the same destination but have distinct origins. Each RU generates traffic with bandwidth w_r (Gbps) and has a maximum tolerable latency $\bar{l}_r$ (ms).

Regarding hosting eligibility, we denote the set of candidate sites for DUs by $V^{DU} \subseteq V$ and for CUs by $V^{CU} \subseteq V$, with $V^{DU} \cup V^{CU} \subseteq V$ and $t \notin V^{DU} \cup V^{CU}$. This indicates that not all nodes are suitable for hosting these functions due to constraints such as processing capacity, latency requirements, and available infrastructure. Note that each site $v \in V^{DU} \cup V^{CU}$ has a limited physical capacity $\bar{m}_v$, representing the maximum number of DU and CU instances it can host, constrained by factors such as rack space, power, and cooling. Each DU can aggregate traffic from multiple RUs but is limited by a processing capacity $\bar{W}_{DU}$ (in Gbps). Similarly, CUs are subject to a processing capacity constraint $\bar{W}_{CU}$ (in Gbps).

The operator must place DU and CU instances while meeting several constraints:

- *Precedency*: each RU r needs a path from $o(r)$ to t that traverses exactly one DU and one CU, in that order: RU $\rightarrow$ DU $\rightarrow$ CU $\rightarrow$ Core. This chain is fundamental and must be enforced explicitly.
- *Distance limits*: physical separation faces strict bounds due to fronthaul and midhaul synchronization needs. The RU-to-DU distance cannot exceed $\bar{d}_{RU}^{DU}$ km (fronthaul), and DU-to-CU distance is capped at $\bar{d}_{DU}^{CU}$ km (midhaul). It is important to note that these per-segment distance constraints are independent of the end-to-end latency constraint. While latency captures the total propagation delay along the entire RU $\rightarrow$ DU $\rightarrow$ CU $\rightarrow$ Core path, the distance limits reflect strict physical and technological requirements imposed by fronthaul and midhaul synchronization protocols, which mandate that DUs remain within a maximum physical reach from their associated RUs, regardless of the overall end-to-end latency. In practice, a path could satisfy the total latency budget while violating a per-segment distance limit, making both constraints necessary and complementary for real-world deployment feasibility.
- *Latency*: the end-to-end latency is a sum of propagation delay (L_{prop}) and processing delays at the DU and CU ($L_{\text{proc}}^{\text{DU}}$, $L_{\text{proc}}^{\text{CU}}$). In our model, we consider only the propagation delay, defined as $\sum_{e \in E} l_e x_e^r$. This is justified because processing delays are hardware-dependent constants that do not vary with topological decisions. The propagation delay, which scales linearly with fiber path length, is the sole topology-dependent variable. This approach is standard in RAN planning literature and aligns with the 3GPP TR 38.801 framework.
- *Capacity:* traffic on any fiber link e cannot exceed C_e. Also, bandwidth assigned to any DU or CU must respect $\bar{W}_{DU}$ and $\bar{W}_{CU}$. And, at each node v, the total number of instances cannot exceed $\bar{m}_v$.

The CAPEX includes fixed installation costs C_v^{DU} and C_v^{CU} (in euros), incurred when activating a DU or CU at site v.

The main goal is to optimize DU and CU deployment to minimize CAPEX of active units, enhancing centralization, resource sharing, and operational efficiency. Minimizing deployed units also lowers operational costs, making the network more cost-effective and sustainable.

In the following section, we introduce a compact formulation for the Cloud-RAN deployment problem.

V. COMPACT FORMULATION

This is based on the following sets of decision variables to model the placement of DU and CU and routing decisions:

- $x_e^r \in \{0, 1\}$: a binary variable that equals 1 if the routing path for RU $r \in R$ uses the fiber link $e \in E$, and 0 otherwise.
- $a_{r,v}^{DU} \in \{0, 1\}$: a binary variable that equals 1 if RU $r \in R$ is assigned to a DU instance located at node $v \in V^{DU}$, and 0 otherwise.
- $a_{r,v}^{CU} \in \{0, 1\}$: a binary variable that equals 1 if the DU serving RU $r \in R$ is associated with a CU instance located at node $v \in V^{CU}$, and 0 otherwise.
- $n_v^{DU}, n_v^{CU} \in \mathbb{N}$: integer variables representing the number of DU and CU instances, respectively, deployed at node $v \in V$.
- $b_{r,v}^{DU} \in \{0, 1\}$: an auxiliary binary variable that equals 1 if a DU serving RU $r \in R$ is placed at or before node $v \in V$ along its routing path, and 0 otherwise. This variable is essential for modeling the precedency constraint.
- $y_e^r, z_e^r \in \{0, 1\}$: auxiliary binary variables used to track the path segments. y_e^r equals 1 if link $e \in E$ is part of the RU-to-DU path for RU $r \in R$, while z_e^r equals 1 if link e is part of the DU-to-CU path for RU r.These variables are used to compute the distance between RU-DU and DU-CU respectively.

The Cloud-RAN deployment problem is equivalent to the following program

$$\min \sum_{v \in V} (C_v^{DU} n_v^{DU} + C_v^{CU} n_v^{CU}). \tag{1}$$

subject to:

$$\sum_{e\in\delta^+(v)} x_e^r - \sum_{e\in\delta^-(v)} x_e^r = \alpha_v^r, \quad \forall r \in R, \forall v \in V, \tag{2}$$

$$\sum_{e\in\delta^-(o(r))} x_e^r = 0, \quad \forall r \in R, \tag{3}$$

$$\sum_{e\in\delta^+(t)} x_e^r = 0, \quad \forall r \in R, \tag{4}$$

$$\sum_{v\in V^{DU}} a_{r,v}^{DU} = 1, \quad \forall r \in R, \tag{5}$$

$$\sum_{v\in V^{CU}} a_{r,v}^{CU} = 1, \quad \forall r \in R, \tag{6}$$

$$\sum_{r\in R} w_r a_{r,v}^{DU} \le \bar{W}_{DU} n_v^{DU}, \quad \forall v \in V^{DU}, \tag{7}$$

$$\sum_{r\in R} w_r a_{r,v}^{CU} \le \bar{W}_{CU} n_v^{CU}, \quad \forall v \in V^{CU}, \tag{8}$$

$$\sum_{r\in R} w_r (x_{vu}^r + x_{uv}^r) \le C_e, \quad \forall (v,u) \in E, \tag{9}$$

$$n_v^{DU} + n_v^{CU} \le \bar{m}_v, \quad \forall v \in V, \tag{10}$$

$$\sum_{e\in\delta^-(v)} x_e^r \le 1, \quad \forall r \in R, \forall v \in V \tag{11}$$

$$a_{r,v}^{DU} \le \sum_{e\in\delta^-(v)} x_e^r, \quad \forall r \in R, \forall v \in V \setminus \{o(r), t\}, \tag{12}$$

$$a_{r,v}^{CU} \le \sum_{e\in\delta^-(v)} x_e^r, \quad \forall r \in R, \forall v \in V \setminus \{o(r), t\}, \tag{13}$$

$$\sum_{e\in E} l_e x_e^r \le \bar{l}_r, \quad \forall r \in R, \tag{14}$$

$$a_{r,v}^{CU} \le b_{r,v}^{DU}, \quad \forall r \in R, \forall v \in V, \tag{15}$$

$$a_{r,v}^{DU} \le b_{r,v}^{DU}, \quad \forall r \in R, \forall v \in V, \tag{16}$$

$$b_{r,v}^{DU} + x_{vu}^r - 1 \le b_{r,u}^{DU}, \quad \forall r \in R, \forall (v,u) \in E, \tag{17}$$

$$\sum_{e\in\delta^+(o(r))} y_e^r = 1 - a_{r,o(r)}^{DU}, \quad \forall r \in R, \tag{18}$$

$$\sum_{e\in\delta^+(v)} y_e^r - \sum_{e\in\delta^-(v)} y_e^r = -a_{r,v}^{DU}, \quad \forall r \in R, \forall v \in V \setminus \{o(r)\}, \tag{19}$$

$$y_e^r \le x_e^r, \quad \forall r \in R, \forall e \in E, \tag{20}$$

$$\sum_{e\in E} d_e y_e^r \le \bar{d}_{RU}^{DU}, \quad \forall r \in R, \tag{21}$$

$$\sum_{e\in\delta^+(v)} z_e^r - \sum_{e\in\delta^-(v)} z_e^r = a_{r,v}^{DU} - a_{r,v}^{CU}, \quad \forall r \in R, \forall v \in V, \tag{22}$$

$$z_e^r \le x_e^r, \quad \forall r \in R, \forall e \in E, \tag{23}$$

$$\sum_{e\in E} d_e z_e^r \le \bar{d}_{DU}^{CU}, \quad \forall r \in R, \tag{24}$$

$$\sum_{n\in V\setminus V^{DU}} n_v^{DU} + \sum_{n\in V\setminus V^{CU}} n_v^{CU} = 0, \tag{25}$$

where $\alpha_v^r = 1$ if $v = o(r)$, -1 if $v = t$, and 0 otherwise. The objective function (1) aims to minimize the total number of deployed DU and CU instances across the network. Constraints (2) are the flow conservation constraints, ensuring that for each RU r, a single path is established from its origin $o(r)$ to the core network destination t. Constraints (5) and (6) guarantee that each RU is assigned to exactly one DU and one CU, respectively, among the available candidate sites. The processing capacity of the deployed DU and CU instances is enforced by constraints (7) and (8), which link the assignment variables to the number of deployed instances. Constraint (9) ensures that the total bandwidth of all traffic traversing a fiber link e does not exceed its capacity C_e. The physical capacity of each site is modeled by constraint (10), which limits the total number of DU and CU instances that can be co-located at any node v. Constraint (11) prevents cycles and multiple traversals through any node, including potential DU and CU hosting sites. The end-to-end latency requirement for each RU is guaranteed by constraint (14). The crucial RU $\rightarrow$ DU $\rightarrow$ CU precedency is modeled by constraints (15)-(17). We ensure in (15) that a CU for RU r can only be placed at node v if a DU has already been encountered on the path (i.e., $b_{r,v}^{DU} = 1$). Constraint (16) states that if a DU for RU r is placed at node v, then $b_{r,v}^{DU}$ must be 1. Constraint (17) propagates the $b_{r,v}^{DU}$ variable along the selected path, ensuring that if a DU has been seen at node v and the path continues to node u, then $b_{r,u}^{DU}$ is also forced to 1. By combining constraints (11), (2), and (15)–(17), we ensure each RU path passes through exactly one DU and then one CU, in strict order. Finally, constraints (18)–(21) define a sub-flow y_e^r from $o(r)$ to the assigned DU, with a distance bound. Similarly, for the midhaul segment via constraints (22)–(24) using variables z_e^r. This model is then integrated into a *Branch-and-Bound* (B&B) framework and tested as shown in the next section.

VI. COMPUTATIONAL STUDY

This section reports computational results from our C++ implementation of the compact model using IBM ILOG CPLEX 12.10 [10]. Tests were run on a Debian GNU/Linux 9.13 system with an Intel Xeon E5-2650 v2 (32 cores at 2.60 GHz) and 248 GB RAM. Experiments used parallel search with up to 32 threads, with CPLEX automatically applying internal cuts (e.g., Gomory, clique, cover, lifted-cover) within its *Branch-and-Cut* (B&C) framework to tighten LP relaxations and speed up convergence. The time limit was set to 7 hours ($25,200$ seconds).

The test graphs in Table I were derived from SNDlib [12].

TABLE I
GRAPHS FROM SNDLIB.

Graph	Nodes	Links
Atlanta	16	26
Germany	17	26
Norway	27	51
Europe	28	41
Giul	39	172
Pioro	40	89
Zib	54	81
Ta	65	108

They are divided into two groups: small-scale networks (Atlanta, Germany, Norway, and Europe) and moderate-size networks (Giul, Pioro, Zib, and Ta).

Starting from the original SNDlib graph, we created instances by retaining nodes and links, then adding attributes such as RU nodes, DU/CU eligibility, capacities, link lengths, and latencies. Each instance included RU requests with origin, shared destination, bandwidth, latency, and segment constraints. Link lengths were estimated from coordinates, with latencies derived proportionally. The objective was to minimize activated DU and CU nodes while meeting routing, capacity, precedence, latency, and distance constraints.

Table II summarizes the main results. For each topology, we tested instances with 5 to 50 RUs, each representing an RRH site with multiple carriers and sectors. A typical 25 km² baseband pool includes 300–600 sectors [1]. Each instance has two scenarios: S1 with loose link capacities and S2 with tighter capacities, making the network more constrained. Instances are labeled by graph name, number of RUs, and scenario (e.g., Atlanta_10_2 for 10 RUs in S2). Table II shows the number of nodes (Nd) in the B&B tree, the lower bound (LB), the upper bound (UB) at termination (best activated DU and CU, with C_v^{DU} and C_v^{CU} set to 1), the final optimality gap, and total CPU time (TT).

Results show that we solved 65 out of 72 instances to optimality (28 out of 28 for small networks and 37 out of 44 for medium networks) within a 7-hour limit (average CPU time: 140 sec for small, 7431 sec for medium). This indicates the formulation performs well for the tested sizes, with the number of active nodes (UB) much smaller than the RUs.

In addition, we observe that active nodes grow slowly compared to RUs, demonstrating the model's ability to reuse nodes efficiently. For example, in Norway, solving for 15 RUs requires only 4 active nodes, the same as for 10 RUs. Regarding computational performance, solution times vary by topology and scenario. Smaller, less dense graphs like German and Norway are solved quickly, while larger, denser graphs like Pioro, Zib, and Ta take longer but often reach optimality. Overall, topology structure and problem complexity, rather than size or density alone, mainly impact performance.

On the other hand, we conducted a sensitivity analysis to assess how network size, number of RUs, and fiber link capacities affect computational performance.

Regarding the influence of network size, results show that the model is efficient on small networks, with CPU times under a few minutes and branch-and-bound nodes below 10. Upper and lower bounds are close, often achieving optimality in seconds. Conversely, networks like Pioro, Zib, and Ta pose challenges; for example, Pioro with 40 nodes and 89 links required up to 25,200 seconds (7 hours), with Nd exceeding 9,000. Zib and Ta, with thousands of nodes and links, often hit the time limit, with UB values much higher than LB, indicating increased difficulty.

TABLE II

PRELIMINARY RESULTS UNDER TWO SCENARIOS.

Instance	Nd	LB	UB	Gap	TT
Atlanta_5_1	1	4	4	0	0.67
Atlanta_5_2	1	5	5	0	1.39
Atlanta_10_1	1806	6	6	0	9.21
Atlanta_10_2	915	6	6	0	15.52
Atlanta_15_1	827	6	6	0	13.83
Atlanta_15_2	30089	7	7	0	265.79
German_5_1	1	4	4	0	8.83
German_5_2	1	4	4	0	2.61
German_10_1	371	5	5	0	3.16
German_10_2	1934	5	5	0	14.89
German_15_1	1019	6	6	0	6.85
German_15_2	12061	6	6	0	78.40
Norway_5_1	40	2	2	0	3.55
Norway_5_2	110	2	2	0	3.82
Norway_10_1	939	4	4	0	17.34
Norway_10_2	2037	4	4	0	14.63
Norway_15_1	4318	4	4	0	39.75
Norway_15_2	2862	4	4	0	18.64
Norway_20_1	5182	6	6	0	35.67
Norway_20_2	4694	6	6	0	46.76
Europe_5_1	1	4	4	0	1.08
Europe_5_2	1	4	4	0	2.18
Europe_10_1	299	5	5	0	3.77
Europe_10_2	1002	5	5	0	9.73
Europe_15_1	2258	6	6	0	23.88
Europe_15_2	4769	6	6	0	23.32
Europe_20_1	7827	8	8	0	1706.57
Europe_20_2	5722	8	8	0	1540.13
Giul_5_1	1	2	2	0	1.70
Giul_5_2	1	2	2	0	2.17
Giul_10_1	608	4	4	0	33.38
Giul_10_2	9	4	4	0	62.64
Giul_15_1	1	6	6	0	37.73
Giul_15_2	1	6	6	0	37.74
Giul_20_1	2447	8	8	0	173.53
Giul_20_2	1	8	8	0	176.04
Giul_30_1	1959	10	10	0	33.38
Giul_30_2	1656	10	10	0	175.35
Pioro_5_1	1	4	4	0	23.97
Pioro_5_2	1	4	4	0	6.62
Pioro_10_1	4500	6	6	0	1270.96
Pioro_10_2	14179	6	6	0	385.67
Pioro_15_1	506	6	6	0	1036.82
Pioro_15_2	3523	7	7	0	2355.83
Pioro_20_1	40743	6.7	8	16.25	25200
Pioro_20_2	69477	6.6	8	17.5	25200
Pioro_30_1	80942	9	9	0	9917.61
Pioro_30_2	90094	7.98	11	27.45	25200
Zib_5_1	1	3	3	0	35.05
Zib_5_2	1	3	3	0	6.39
Zib_10_1	4486	4	4	0	284.57
Zib_10_2	2481	4	4	0	260.30
Zib_15_1	20858	7	7	0	2541.95
Zib_15_2	76727	8	8	0	22837.53
Zib_20_1	58213	8	8	0	8911.48
Zib_20_2	2258	31	31	0	5.61
Zib_30_1	58793	9	9	0	15853.50
Zib_30_2	90094	6.7	10	33	25200
Zib_50_1	24624	12	12	0	2155.84
Zib_50_2	43378	16.56	19	12.86	25200
Ta_5_1	1527	3	3	0	85.87
Ta_5_2	233	3	3	0	138.72
Ta_10_1	4497	5	5	0	945.63
Ta_10_2	20944	6	6	0	601.68
Ta_15_1	163062	6	6	0	25002.51
Ta_15_2	72235	6	6	0	10399.81
Ta_20_1	87518	6	6	0	4078.30
Ta_20_2	48454	6	6	0	5744.25
Ta_30_1	176945	7	7	0	16006.69
Ta_30_2	52309	7	7	0	24141.51
Ta_50_1	76065	10	13	23.08	25200
Ta_50_2	55622	10	15	33.33	25200

The analysis also reveals that the problem's difficulty increases significantly with the number of RUs. For instance, the Pioro network with 5 RUs is solved in 23 seconds, but increasing to 10 RUs requires over 1270 seconds and explores nearly 4,500 nodes. With 20 RUs, the solution time reaches the time limit, resulting in a final gap of 16.25%. Similarly, the Europe network, despite being smaller, requires substantial effort at 20 RUs. However, the Giul network remains easier to solve even at larger sizes because its topology offers more routing options for RUs, facilitating routing and leading to faster convergence.

This difference in computational behavior is due to the networks' topological properties. The Giul network (39 nodes, 172 links) is denser (density: 0.23, average degree: 8.82) than Pioro (40 nodes, 89 links; density: 0.11, average degree: 4.45) and Zib (54 nodes, 81 links; density: 0.05, average degree: 3.00). Giul's high connectivity offers many routing options, easing feasible placements and routes. In contrast, limited path diversity in Pioro and Zib makes the problem highly combinatorial, with single routing decisions potentially blocking others, leading to larger search spaces and longer computation times.

Regarding capacity influence, results show that tighter capacities significantly increase problem difficulty. Stricter constraints tend to raise the number of activated DU/CU nodes and the objective value. For example, in Zib with 20 RUs, the optimal active nodes increase from 8 to 31 under tighter constraints. Conversely, in some cases like Norway, tighter capacities unexpectedly improve solution times by reducing the search space. A detailed analysis of Norway with 15 RUs shows that in the loose capacity scenario (S1), CU functions are centralized on nodes $\{13, 25\}$, creating high traffic on links feeding node 25. In the tighter scenario (S2), this centralization becomes infeasible due to link capacity violations, which prunes a large part of the search space. The solver then finds an alternative optimal solution with a more distributed CU placement on nodes $\{9, 15\}$, leading to a smaller search tree and faster solution. For larger instances like Zib with 50 RUs, the problem becomes more challenging under tighter capacities, and optimality is often not reached within the time limit.

Overall, these findings demonstrate the model's effectiveness but also its limitations on large instances, highlighting the need for techniques like cutting-plane algorithms and heuristics to improve performance.

VII. CONCLUSION AND FUTURE WORKS

In this paper, we addressed the strategic deployment of Cloud-RAN by optimizing DU and CU placement and traffic routing. Our framework models the entire RU-DU-CU service chain, incorporating key constraints such as fiber capacity, segment distance limits, and end-to-end latency.

Preliminary experiments showed that the model effectively solved most tested instances to optimality, confirming its robustness for identifying optimal deployment strategies in Cloud-RAN networks. In addition, results show that the model promotes significant consolidation of DU and CU functions, a key goal for reducing CAPEX and improving resource utilization. While solution times are fast for smaller and medium instances, they also indicate that scalability can become a challenge for larger or more complex instances, highlighting the combinatorial difficulty of the problem.

Based on these promising results, several avenues for future work emerge. A primary direction is to study the behavior of our compact model while using further larger instances (larger graphs and larger number of RUs), and identify its drawbacks. As a next step, we plan to develop a path-based formulation solved via *column generation* algorithm. We also plan to develop heuristic approaches to enable comparisons with the exact model and further assess its performance on larger instances. Additionally, we aim to explore decomposition techniques like *Benders* and *Dantzig-Wolfe*, combined with cutting-plane methods, to enhance scalability and solution efficiency.

REFERENCES

[1] C-RAN the Road towards Green RAN.

[2] O. Chabbouh, S. B. Rejeb, N. Agoulmine, and Z. Choukair. Cloud RAN Architecture Model Based upon Flexible RAN Functionalities Split for 5G Networks. In *2017 31st International Conference on Advanced Information Networking and Applications Workshops (WAINA)*, pages 184–188, 2017.

[3] A. Checko, H. L. Christiansen, Y. Yan, L. Scolari, G. Kardaras, M. S. Berger, and L. Dittmann. Cloud RAN for Mobile Networks—A Technology Overview. *IEEE Communications Surveys Tutorials*, 17(1):405–426, 2015.

[4] H. Hojeij, M. Sharara, S. Hoteit, and V. Vèque. On Flexible Placement of O-CU and O-DU Functionalities in Open-RAN Architecture. *IEEE Trans. on Netw. and Serv. Manag.*, 22(1):660–674, Feb. 2025.

[5] M. Klinkowski. Latency-Aware DU/CU Placement in Convergent Packet-Based 5G Fronthaul Transport Networks. *Applied Sciences*, 10(21), 2020.

[6] L. M. P. Larsen, H. L. Christiansen, S. Ruepp, and M. S. Berger. Deployment Guidelines for Cloud-RAN in Future Mobile Networks. In *Proceedings of 11th International Conference on Cloud Networking*, pages 141–149, United States, Nov. 2022. IEEE. 11th IEEE International Conference on Cloud Networking, CloudNet 2022 ; Conference date: 07-11-2022 Through 10-11-2022.

[7] A. Maeder, A. Ali, A. Bedekar, A. F. Cattoni, D. Chandramouli, S. Chandrashekar, L. Du, M. Hesse, C. Sartori, and S. Turtinen. A Scalable and Flexible Radio Access Network Architecture for Fifth Generation Mobile Networks. *Comm. Mag.*, 54(11):16–23, Nov. 2016.

[8] F. W. Murti, J. A. Ayala-Romero, A. Garcia-Saavedra, X. Costa-Pérez, and G. Iosifidis. An Optimal Deployment Framework for Multi-Cloud Virtualized Radio Access Networks. *IEEE Transactions on Wireless Communications*, 20(4):2251–2265, 2021.

[9] A. Ndao, X. Lagrange, N. Huin, G. Texier, and L. Nuaymi. Optimal Placement of Virtualized DUs in O-RAN Architecture. In *2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring)*, pages 1–6, 2023.

[10] S. Nickel, C. Steinhardt, H. Schlenker, and W. Burkart. *IBM ILOG CPLEX Optimization Studio—A primer*, pages 9–21. Springer Berlin Heidelberg, Berlin, Heidelberg, 2022.

[11] A. J. N. Njanda, J. F. Muchanga, H. Baghban, and K. D. R. Assis. Practical DU-CU Placement in an Enhanced Mobile BroadBand Request During 5G Traffic. *IEEE Access*, 13:137317–137326, 2025.

[12] S. Orlowski, R. Wessäly, M. Pióro, and A. Tomaszewski. SNDlib 1.0—Survivable Network Design Library. *Netw.*, 55(3):276–286, May 2010.

[13] M. Polese, L. Bonati, S. D'Oro, S. Basagni, and T. Melodia. Understanding O-RAN: Architecture, Interfaces, Algorithms, Security, and Research Challenges. 2022.

[14] B. H. Prananto, Iskandar, and A. Kurniawan. Low Split Cloud RAN Opportunities and Challenges. In *2019 IEEE 13th International Conference on Telecommunication Systems, Services, and Applications (TSSA)*, pages 119–123, 2019.