Optimal placement of virtualized DUs in O-RAN architecture

Amath Ndao, Xavier Lagrange, Nicolas Huin, Géraldine Texier, Loutfi Nuaymi

Optimal placement of virtualized DUs in O-RAN architecture

Amath Ndao, Xavier Lagrange, Nicolas Huin, Geraldine Texier and Loutfi Nuaymi
IMT Atlantique, IRISA UMR CNRS 6074, F-35700 Rennes, France
firstname.lastname@imt-atlantique.fr

***Abstract*—Open Radio Access Network (O-RAN) is very promising for flexible and efficient 5G and 6G wireless networks. The O-RAN architecture consists of three main units: Radio Unit (RU), Distributed Unit (DU), and Centralized Unit (CU). In this paper, we study the placement of virtualized DUs. This placement has strong consequences on cost and delay, among others, and is thus an important challenge. First, we analyze the throughput between the O-RAN interfaces. Based on our analysis, we propose an efficient Integer Linear Programming (ILP) model. The objective is to minimize the O-RAN cost depending on the DU placement while respecting the delay and capacity constraints. We evaluate our model on a real topology. Our results provide interesting insights into the cost savings with regard to a legacy architecture. Moreover, the proposed model provides solutions in a configuration where a fully centralized Cloud RAN architecture would not. We also estimate the limits of capacity of a given configuration.**

***Index Terms*—O-RAN, Optimal placement, Virtualized Radio Access Network (vRAN), Beyond 5G**

I. Introduction

With 5G and 6G, the traffic demand is getting bigger and bigger. The challenge is to serve all the users and respect the delay and capacity constraints while optimizing the cost of the network. The existing architectures will not be sufficient for the problem. The D-RAN architecture, used in 3G/4G, is cost-inefficient for dense networks due to its expensive Radio Units (RUs) and reduced resource pooling possibilities [7]. On the other hand, the Cloud-RAN (C-RAN) architecture cannot be deployed currently on a real topology because the delay and capacity constraints are too tight.

The O-RAN reference architecture, presented by the O-RAN Alliance [5], [6], seems like a promising solution. Envisioned for the next generation of Radio Access Network (RAN) infrastructures, this new architecture features virtualized wireless access networks on open devices and artificial intelligence for radio control. It is composed of three units: the RU, which deals with the filtering and the frequency transposition; the DU, which deals with digital modulation, signal processing, and retransmission; and the CU, which deals with the transport of IP packets and signalling. These units are connected through interfaces, and their connection follows a well-defined order, i.e., RU, DU, and CU. The O-RAN architecture enables interoperability between multiple providers, this allows more connections in the network. O-RAN has already embraced virtualization and it is assumed as the next version of vRAN (Virtualized Radio Access Network) with more capabilities, more flexibility because the functions can be placed in any node of the network respecting the constraints.

Nevertheless, the configuration of the O-RAN architecture always remains a challenge because each configuration has delay and capacity constraints to respect the data transfer between the different units.

Contributions. We propose a model with one CU co-located with the network core and several RUs close to the users. In our model, the locations of the RUs and the CU are fixed, but the DU can be placed anywhere in the network, RUs and CU locations included. Firstly, we study the interfaces of the O-RAN architecture and compute the flow on the CU–DU interface and on the DU–RU interface. Secondly, we formulate our problem as an Integer Linear Program (ILP) whose objective is to place the DUs in the network by minimizing the cost of the system. Thirdly, we evaluate our model on a real topology. We show that the C-RAN does not give a solution because the delay constraints are not respected and that our model is much more beneficial than the D-RAN for the different scenarios.

Paper organization. In Section II, we present the related work. In Section III, we study the interfaces of the O-RAN architecture and compute the throughput of these interfaces. In Section IV, we define the considered problem and formulate our problem as an Integer Linear Program. We evaluate our model and analyze the results in Section V.

II. Related Work

Currently, real wireless topologies with high delay constraints do not allow C-RAN to be deployed over the entire network, so research has taken a new direction using the O-RAN architecture proposed by the O-RAN alliance. In this part, we show the challenges for the placement of functions in the RAN network.

Garcia-Saavedra et al. [7] propose a model with one CU and several RUs close to the users. Their challenge is to decide whether to place the functions on the RUs or on the CU. Their objective is to maximize the number of functions placed on the CU in order to minimize the cost of the system. Ojaghi et al. [8] propose Sliced-RAN, a similar approach to FluidRAN [7]. In this model, each user's function placement can be different from the others depending on what he does. For example, someone sending messages will not have the same slice as someone watching a movie. Both studies consider neither O-RAN nor

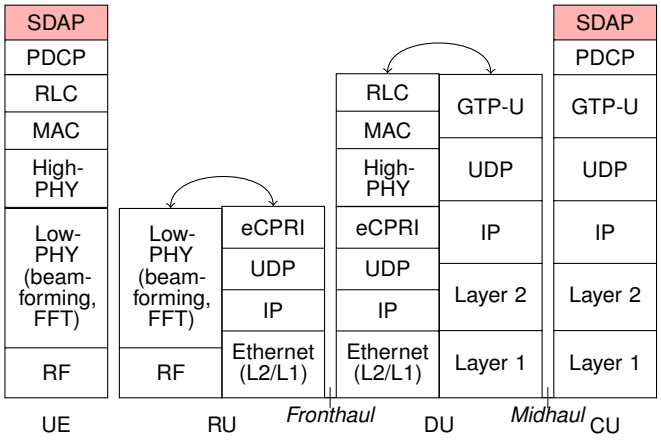

Fig. 1. User Plane protocol stack for Base Station

DU placement, even though optimizing the placement of DUs improves performance.

Murti et al. [9] consider several CUs in their model. The CUs can be placed on network nodes and can be virtualized. Their objective is to minimize the number of nodes that host CUs and to maximize the number of functions placed in the CUs. This reduces the total cost. However, the authors consider fixed DUs close to RUs. Restricting the DU placement leads to sub-optimal results.

Recently, Morais et al. [10] proposed a model for O-RAN with the three units: RU, DU, and CU, where DU and CU are virtualized. They share the same objective as [9]. However, the study considers a fixed throughput on each interface. In our model, we take into account the variation of the user load and deduce the throughput on the interfaces.

III. Analysis of the O-RAN architecture

Our objective is to estimate and analyze the effects of encapsulations on the throughputs of the different interfaces of the O-RAN architecture. We consider the network at a given load λ defined as the bit rate generated by all Internet Protocol (IP) data packets sent to each cell of the mobile network.

As previously mentioned, the O-RAN architecture is composed of three units, and their interfaces involve multiple protocols (see Fig. 1): the RU contains the RF and low-PHY layers; the DU contains the Radio Link Control (RLC), the Medium Access Control (MAC), and the High-PHY layers; the CU contains the Radio Resource Control (RRC), the Service Data Adaptation Protocol (SDAP), and the Packet Data Convergence Protocol (PDCP) layers. The RU–DU and DU–CU interfaces are defined in the Common Public Radio Interface (CPRI) forum [6] and [5], respectively.

A. Downlink throughput between CU and DU

To transmit user information between the CU and the DU, GTP-U tunnels are used (see the right part of Fig. 1): user IP packets are first encapsulated into an SDAP Protocol Data Unit (PDU), then into a PDCP PDU and finally transported in a GTP-U tunnel, which is itself transported by User Datagram Protocol (UDP) over IP. We denote the size of all layer headers between the CU and the DU as H_M (M stands for midhaul):

$$H_M = H_{SDAP} + H_{PDCP} + H_{GTP} + H_{UDP} + H_{IP} \quad (1)$$

where H_x is the header size of layer x (see Table I).

TABLE I
The header of the different layers on O-RAN architecture

Layer	Header name	Header length	Reference
SDAP	H_{SDAP}	1	[3]
PDCP	H_{PDCP}	2	[2]
GTP-U	H_{GTP-U}	8	[4]
UDP	H_{UDP}	8	Figure 3-2 of [13]
IP	H_{IP}	20	Figure 3-2 of [13]
RLC	H_{RLC}	5	[12]
MAC	H_{MAC}	2	[12]
IQ	H_{IQ}	13	Table 6-2 of [13]
eCPRI	H_{CPRI}	8	Table 3-1 of [13]
Ethernet	$H_{Ethernet}$	14	Figure 3-1 of [13]

Let L_{IP} is the size of the user IP packet. As illustrated by Fig. 3, $\frac{\lambda}{L_{IP}}$ is the number of user packets per second in a cell. The bit rate R_M between CU and DU is thus:

$$\begin{aligned} R_M &= (L_{IP} + H_M)\,\frac{\lambda}{L_{IP}} \\ &= 1.026\lambda. \end{aligned} \quad (2)$$

B. Downlink throughput between DU and RU

As shown in Fig. 1, the PDCP packet is processed by the RLC and MAC layers. Let H_{F_1} be the total header size at the MAC layer on the fronthaul (see Table I):

$$H_{F_1} = H_{SDAP} + H_{PDCP} + H_{RLC} + H_{MAC}. \quad (3)$$

We assume there is neither segmentation nor reassembly. The MAC PDU is one transport block. A Foward Error Correction (FEC) with a code rate r and a modulation with Q bits per symbol is applied according to the selected Modulation and Coding Scheme (MCS) (see Fig. 2): the output is a set of complex symbols (I and Q). For the sake of simplicity, we consider an average case scenario (i.e., average values of r and Q).

Using the same approach as in Section III-A, we deduce that the number of complex symbols per second transmitted to the RU is given by:

$$\frac{L_{IP} + H_{F_1}}{rQ}\frac{\lambda}{L_{IP}}. \quad (4)$$

These symbols are then quantified and transported in CPRI frames. We assume that each CPRI frame includes only one Physical Resource Block (PRB) and that each PRB occupies a whole slot on the radio interface. However, all the symbols of a PRB do not carry information: some are used for reference signals or downlink control information. Let M be the number of data symbols in a PRB. The size of CPRI frames is $H_{F_2} + 2\theta M$, where H_{F_2} is the total header size due to CPRI

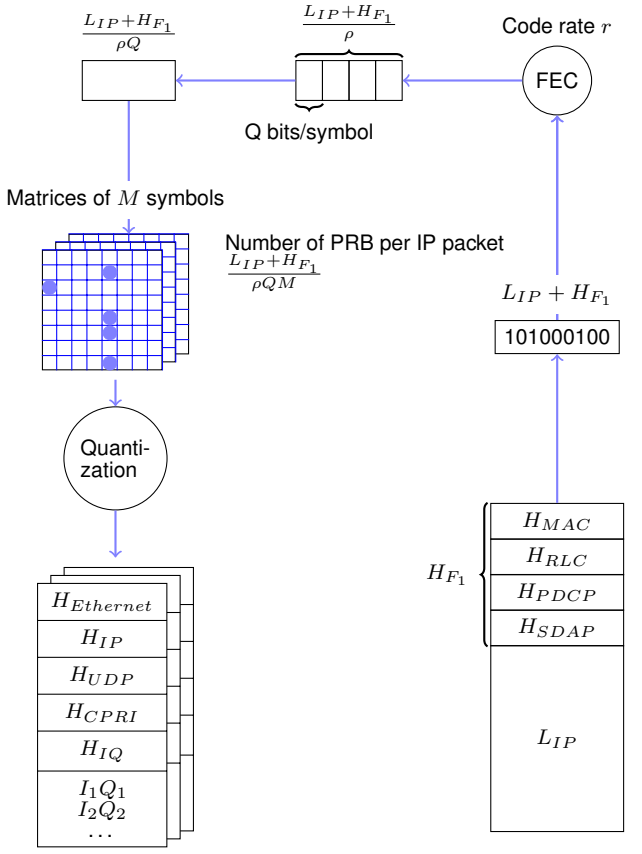

Fig. 2. Different steps to compute throughput between DU and RU

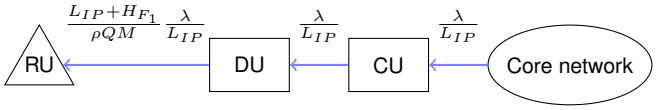

Fig. 3. Number of packet per second between the different units of an ORAN.

and companion protocols, and θ is the number of bits for quantization.

There is a CPRI header and a specific header when transmitting IQ symbols (see Table I). Since CPRI works over UDP/IP over Ethernet, H_{F_2} is given by:

$$H_{F_2} = H_{IQ} + H_{CPRI} + H_{UDP} + H_{IP} + H_{Ethernet}. \quad (5)$$

The bit rate R_F on the fronthaul is given by:

$$\begin{aligned} R_F &= (H_{F_2} + 2\theta M) \times \frac{L_{IP} + H_{F_1}}{rQM} \times \frac{\lambda}{L_{IP}} \\ &= 8.3728\lambda. \end{aligned} \quad (6)$$

All variables used for computing rates are given in Table II. The flows calculated in (2) and (6) are used in the formulation of our problem, especially in the routing decision.

TABLE II
THE VARIABLES USED TO CALCULATE THE THROUGHPUT

Variables	Value	Description
L_{IP}	1500	Size of a packet
Q	6	Number of bits per symbol (MCS=15)
r	0.65	Coding rate (MCS=15)
θ	16	Bits used to quantify either an I or a Q symbol

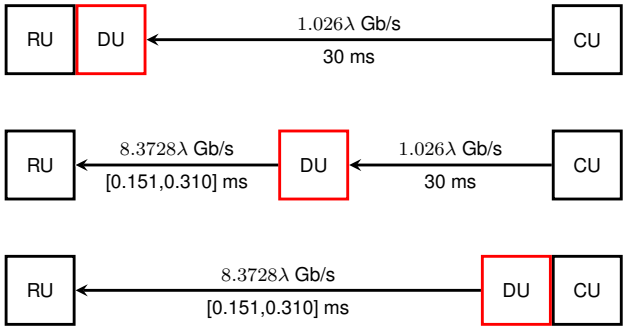

Fig. 4. Bandwidth and latency requirements of the main splits

IV. Problem Statement and Model

A. Problem statement

We represent the network as a directed graph $G = (V, A)$ where V represents the set of nodes and A the set of links. Each node $v \in V$, depending on its processing capacities C_v, can host RUs, DUs, and CUs. Each arc $a \in A$ has a bandwidth B_a and a delay δ_a. In our model, we consider a single CU co-located with the core network and RUs already deployed on the nodes of the network; only the DUs need to be placed. Let v_0 be the node hosting the CU, and $V_R \subseteq V$ be the set of nodes hosting the RUs. Each demand represents the traffic from the CU to an RU $n \in V_R$, which requires an amount of bandwidth denoted by λ_n.

DUs are implemented in virtual machines (VMs). The upkeep cost of using a VM depends on the hardware. CUs are in central facilities and use high-end servers; hence this cost will be lower compared to RUs. We denote the average cost for instantiating a VM as α_v (monetary units), the average cost for serving each request for each node as β_v (monetary units/cycle) (as in [7]), and the processing loads of DU as ρ_{DU} (cycle per Gbps).

Our goal is to place a set of DUs in the network and assign each RU to a DU such that we can serve all demands while minimizing the network cost. The assignment of RU to DU is constrained by the delay and bandwidth requirements of the functional split, as described in Fig. 4.

B. Problem Formulation

Function placement. We denote the binary variable indicating the placement of the DU serving a RU n on the node v as y_{nv}. Let z_v be the activation variable of a DU on node v. When the DU is deployed with the CU, it can serve many RU according to the delay of each request.

An RU can be assigned at most to one DU:

$$\sum_{v \in V} y_{nv} \leq 1 \qquad \forall n \in V_R \tag{7}$$

$$y_{nv} \leq z_v \qquad \forall n \in V_R, \forall v \in V. \tag{8}$$

The placements need to ensure that the computing capacity at each location is satisfied, hence:

$$\sum_{n \in V_R} \lambda_n y_{nv} \rho_{DU} \leq C_v \qquad \forall v \in V. \tag{9}$$

Routing decisions. We use two sets of binary variables for routing decisions: $x^M_{an} \in \{0,1\}$ indicates the arcs taken between CU and DU, and $x^F_{an} \in \{0,1\}$ indicates the arcs taken between DU and RU. The flow constraints are defined below, for each node $v \in V$ and $n \in V_R$:

$$\sum_{a \in \omega^+(v)} x^M_{an} - \sum_{a \in \omega^-(v)} x^M_{an} = \begin{cases} 1 - y_{nv} & v{=}v_0 \\ -y_{nv} & \text{otherwise.} \end{cases} \tag{10}$$

$$\sum_{a \in \omega^+(v)} x^F_{an} - \sum_{a \in \omega^-(v)} x^F_{an} = \begin{cases} y_{nv} - 1 & v = n \\ y_{nv} & \text{otherwise.} \end{cases} \tag{11}$$

where $\omega^-(v)$ is the incoming flow for node $v \in V$, and $\omega^+(v)$ the outgoing flow for node $v \in V$.

We remind that $1.026\lambda_n$ is the flow in Gbps from CU to DU and $8.372\lambda_n$ the flow in Gbps from DU v to RU n (see Section III-A and Section III-B). The routing decisions need then to respect the link capacities, giving the following constraints:

$$\sum_{n \in V_R} K^M \lambda_n x^M_{an} + K^F \lambda_n x^F_{an} \leq B_a \qquad \forall a \in A \tag{12}$$

with K^M=1.026 and K^F=8.372.

Delay constraints. The DU placement determines which arcs the flow takes from the CU to the RU. According to [12], when the DU is placed with the RU the delay constraints equal 30 ms, and, according to [13], when the DU is placed with the CU, the delay is included in an interval $[0.151, 0.310]$ ms. In our study, we use a value of 0.25 ms as in FluidRAN. The delay constraints are as follows:

$$\sum_{a \in A} \delta_a x^M_{an} \leq 30 \qquad \forall n \in V_R \tag{13}$$

$$\sum_{a \in A} \delta_a x^F_{an} \leq 0.25 \qquad \forall n \in V_R. \tag{14}$$

Objective function. Our goal is to place a set of DUs in the network and assign each RU to a DU such that we can serve all demands while minimizing the network cost. The deployment of the DU with the RU, with the CU, or in between incurs a computational cost given by:

$$R_v(y) = \alpha_v z_v + \rho_{DU} \beta_v \sum_{n \in V_R} \lambda_n \times y_{nv} \qquad \forall v \in V \tag{15}$$

TABLE III
THE VARIABLES USED IN OUR MODEL

Variable	Definition
A	The set of links
B_a	Maximum bandwidth of link a
C_v	Processing capacity of node $v \in V$
R_v	Computing cost of node $v \in V$
V	The set of routers
v_0	Node containing the CU
V_R	The set of RU
ω_v^-	The incoming flow for node $u \in V$
ω_v^+	The outgoing flow for node $u \in V$
x^M_{an}	Arc a between CU and DU
x^F_{an}	Arc a between DU and RU
y_{nv}	DU placement and RU n assigned to DU v
δ_a	The delay of arc a
ρ_{DU}	Processing loads (cycles per Mb/s) of DU
λ_n	The load for each RU n

where $y = (y_{nv} \in \{0,1\} : \quad n \in V_R, v \in V)$.

All variables are defined in Table III, and our problem is formulated as follows:

$$\min_{x,y,z} \quad \alpha_v z_v + \beta_v \sum_{n \in V_R} \lambda_n y_{nv}$$

$$s.t. \quad \sum_{n \in V_R} \lambda_n y_{nv} \leq C_v \qquad \forall v \in V$$

$$\sum_{n \in V_R} K^M \lambda_n x^M_{an} + K^F \lambda_n x^F_{an} \leq B_a \qquad \forall a \in A$$

$$\sum_{a \in \omega^+(v)} x^M_{an} - \sum_{a \in \omega^-(v)} x^M_{an} = \begin{cases} 1 - y_{nv} & \text{if } v = v_0 \\ -y_{nv} & \text{otherwise} \end{cases}$$
$$\forall n \in V_R, \forall v \in V$$

$$\sum_{a \in \omega^+(v)} x^F_{an} - \sum_{a \in \omega^-(v)} x^F_{an} = \begin{cases} y_{nv} - 1 & \text{if } v = n \\ y_{nv} & \text{otherwise} \end{cases}$$
$$\forall n \in V_R, \forall v \in V$$

$$\sum_{a \in A} \delta_a x^M_{an} \leq 30 \qquad \forall n \in V_R$$

$$\sum_{a \in A} \delta_a x^F_{an} \leq 0.25 \quad \forall n \in V_R$$

$$\sum_{v \in V} y_{nv} \leq 1 \qquad \forall n \in V_R$$

$$y_{nv} \leq z_v \qquad \forall n \in V_R, \forall v \in V.$$

V. Evaluation and Analysis

A. Network topology and scenarios

In this part, we evaluate our model on a real topology. The T_{5Gx} topology, located in a northern region of Italy, is composed of 51 nodes connected through a ring structure [1], [10]. The T_{5Gx} topology is composed of 61 links, and their capacity varies from 40 Gbps to 400 Gbps between the aggregation nodes (AG) and from 10 Gbps to 40 Gbps between the access nodes (AC). Fig. 5 shows the topology: the red nodes represent the access nodes, which are connected to a RU; the

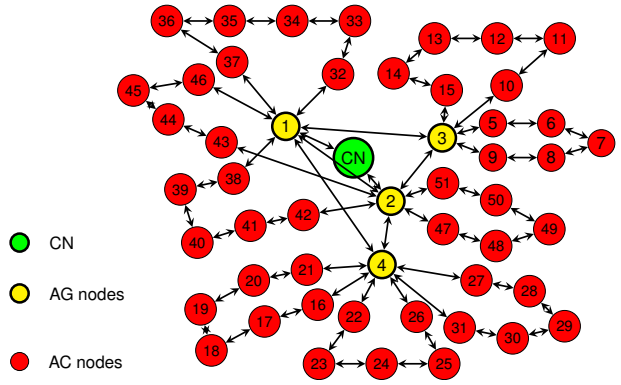

Fig. 5. T_{5Gx} topology

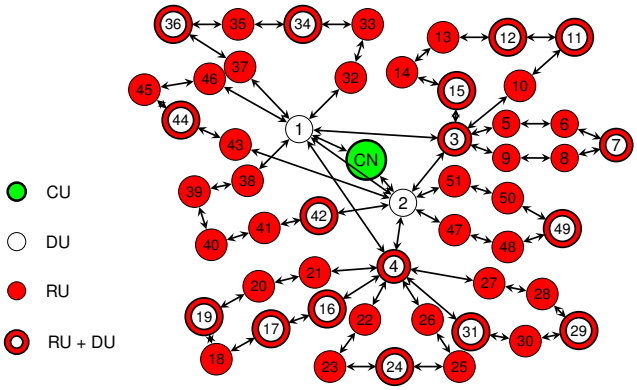

Fig. 6. DU placement for High capacity

TABLE IV
SCENARIOS EMPLOYED IN THE EVALUATION [10]

	Low capacity		High capacity	
Transport nodes	AC	AG	AC	AG
Computing resources	8	16	16	32
Link capacity (Gbps)	10/25	40/100	25/40	100/400
Computing/Fiber latency (μs)	2/0.005		50/0.005	

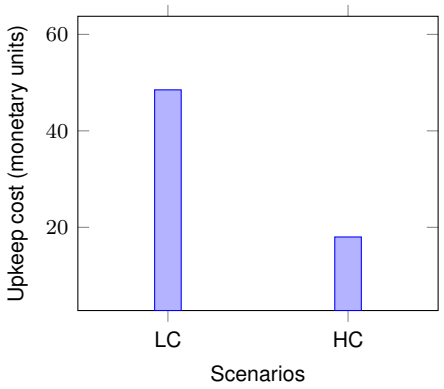

Fig. 7. Upkeep cost for a load $\lambda = 0.15$ Gbps

green node represents the core of the network (CN); and the yellow nodes represent the aggregation nodes. Among the aggregation nodes, nodes 1 and 2 are not connected to a RU while nodes 3 and 4 are connected to a RU.

Table IV summarizes the values of the different scenarios used to evaluate our model. There are two types of scenarios: Low capacity and High capacity. According to [10], T_{5Gx} uses only the Computing and Fiber components. Since in T_{5Gx} all link distances are available, the propagation delay is directly computed as a deterministic function of the distance.

To evaluate our model, we used a processor 11th Gen Intel® Core™ i7-1165G7 @ 2.80 GHz 2.80 GHz. To solve our problem, we use Python 3.9.7 and IBM CPLEX 12.8.0. Based on [7], [10], we estimated the processing load of a DU at $\rho_{DU} = 1.6$ cycles per Gbps. According to FluidRAN, the upkeep cost is approximately half when done in the core network, i.e., $\alpha_0 = \frac{\alpha_n}{2}$, with $\alpha_n = 1$ the upkeep cost of a VM in the RUs. To have the upkeep cost of a VM at the aggregation nodes between CU and RU, we take a value $\alpha_v = 0.75 \in [\alpha_0, \alpha_n]$. Based on [7], [11], we estimate the CU processing cost to $\beta_0 = 0.017\beta_n$ (linear regression in [[11], Fig.6a]) with $\beta_n = 1$ the processing cost of RU. To have the processing cost of DU, we take a value $\beta_v = 0.08 \in [\beta_0, \beta_n]$.

B. Analysis of results

For the low capacity scenario, all DUs are placed with RUs because the delay constraints are very high. Even when we increase the load, the number of DUs is equal to the number of RUs, i.e., one DU serves one RU (the number of DUs is equal to 49). When the capacity is increased (i.e., in the high capacity scenario), the number of DUs placed in the network is much smaller and increases proportionally with the load until the maximum number is reached. For a small load $\lambda = 0.15$ Gbps, the number of DUs is equal to 19, and one DU serves, on average, two RUs (see Fig. 6).

Fig. 7 shows that the upkeep cost for low capacity is much higher than for high capacity. This cost depends on the number of DUs and their location in the network. For high capacity, some DUs are placed in aggregation nodes (see Fig. 6), which have a much lower upkeep cost α_v and can serve several RUs whereas, for low capacity, the DUs are co-located with the RUs. This means that the high capacity scenario provides more mutualization gains than the low capacity.

In Fig. 8 we compare our model with the D-RAN model for the different scenarios. The D-RAN architecture is used in current networks and all functions are placed at the base station. In another word, the RUs, DUs and CUs are co-located together which makes the D-RAN model does not take into account the delay constraints between the interfaces.

Fig. 8 shows that when the load is low, i.e., $\lambda \leq 0.1$ Gbps, the cost of the D-RAN is close to the Low capacity but less beneficial, because in the D-RAN model, the number of CU is equal to the number of RU while in our model there is only one CU for all RU. The system cost for our model with high capacity scenario is more beneficial than the D-RAN and our model with low capacity because one DU can serve several

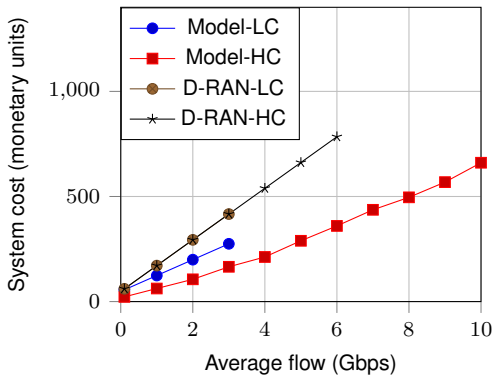

Fig. 8. Comparison of the system cost as a function of the load between our model and D-RAN.

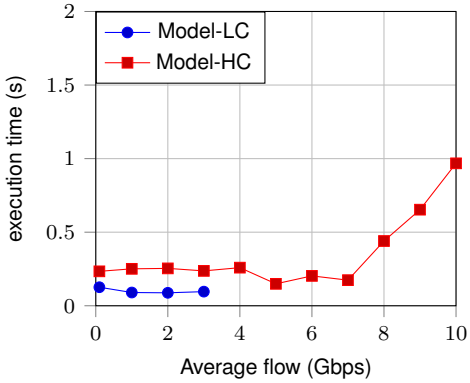

Fig. 9. Execution time for low and high capacity

RUs, and they are all served by one CU.

When λ increases, the gap between D-RAN, our low-capacity scenario model and our high-capacity scenario model increases. The results of our high-capacity scenario model become better than those of the D-RAN and our low-capacity scenario model to the extent of a 65.2% and 29.7% advantage of the latter.

Fig. 8 shows the limits of our model and the D-RAN model. For the low capacity scenario, when the load $\lambda > 3$ Gbps, our model does not find a solution because the capacities of some links are exceeded. For the high capacity scenario, when the load exceeds 10 Gbps, the processing capacity constraints are no longer respected. For the D-RAN model with the low capacity scenario, the limits of some links are exceeded when the load $\lambda = 3$ Gbps and beyond this load, the D-RAN model does not give any solution. For the high capacity scenario, the limits of the processing capacity constraint are reached when the load $\lambda = 6$ Gbps and beyond this load, the D-RAN model does not give any solution.

In order to estimate the execution time, we simulate our model 10 times and then calculate the average execution time for each load. The execution time for our model with low capacity scenario is almost constant whenever a solution exists (load up to 3 Gbps approx.). This is because the locations of the DUs do not change when the load increases. For the high capacity scenario, the execution time is around 0.25 s up to a load of approx. 7 Gbps. As we reach higher loads, the execution time starts to increase due to the tighter capacity constraints (see Fig. 9).

VI. Conclusion

In this work, we analyzed the interfaces of O-RAN architecture and computed the throughput on the CU-DU and DU-RU interfaces. We formulated the problem of DU placement in the network, which still remains a challenge. The evaluation of our model on a real topology showed that C-RAN couldn't be deployed in the network because of tight delay constraints. The analysis of our results showed that our model is much more economical when increasing the load compared to D-RAN for the different scenarios. The O-RAN architecture offers more flexibility in the network, and in particular, we want to focus our future work on the scenario where a user connects to multiple RUs.

Acknowledgment

This work was carried out in the context of Beyond5G, a project funded by the French government as part of the economic recovery plan, namely "France Relance" and the investments for the future program.

References

[1] 5G-Crosshaul, d1.2: final 5G-Crosshaul system design and economic analysis, 2017.
[2] 3GPP. 5G; NR; Packet Data Convergence Protocol (PDCP) specification. Technical Specification (TS) 38.323, 3rd Generation Partnership Project (3GPP), 11 2020. Version 16.2.0 Release 16.
[3] 3GPP. LTE; 5G; Evolved Universal Terrestrial Radio Access (E-UTRA) and NR; Service Data Adaptation Protocol (SDAP) specification. Technical Specification (TS) 37.324, 3rd Generation Partnership Project (3GPP), 11 2020. Version 16.2.0 Release 16.
[4] 3GPP. Universal Mobile Telecommunications System (UMTS); LTE; 5G; General Packet Radio System (GPRS) Tunnelling Protocol User Plane (GTPv1-U). Technical Specification (TS) 29.281, 3rd Generation Partnership Project (3GPP), 01 2020. Version 15.7.0 Release 15.
[5] 3GPP. NG-RAN; F1 Application Protocol (F1AP). Technical Specification (TS) 38.473, 3rd Generation Partnership Project (3GPP), 12 2021. Version 15.16.0.
[6] O-RAN Alliance. O-RAN architecture description v04.00. Technical report o-ran.wg1, O-RAN Alliance, 2021.
[7] A. Garcia-Saavedra et al. Fluidran: Optimized vran/mec orchestration. In *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*. IEEE, 04 2018.
[8] B. Ojaghi et al. Sliced-RAN: Joint slicing and functional split in future 5g radio access networks. In *ICC 2019-2019 IEEE International Conference on Communications (ICC)*. IEEE, 05 2019.
[9] F. Wisnu Murti et al. On the optimization of multi-cloud virtualized radio access networks. In *ICC 2020-2020 IEEE International Conference on Communications (ICC)*. IEEE, 06 2020.
[10] F. Zanferrari Morais et al. PlaceRAN: optimal placement of virtualized network functions in beyond 5g radio access networks. *IEEE Transactions on Mobile Computing*, 04 2022.
[11] P. Rost et al. The complexity-rate tradeoff of centralized radio access networks. In *IEEE Trans. on Wireless Comm., 14 (11)*, 2015.
[12] Small Cell Forum. Small cell virtualization functional splits and use cases. Technical report, Small Cell Forum, January 2016.
[13] O-RAN Fronthaul Working Group. Control, user and synchronization plane specification. O-ran.wg4.cus.0-v06.00, O-RAN Alliance, 2021.