Minimizing delay in NFV 5G Networks by means of flexible split selection and scheduling

Luis Diez, Víctor González, Ramón Agüero *Senior Member, IEEE*
Department of Communications Engineering - University of Cantabria, Spain
Avda Castros s/n, - 39005 Santander, Spain
ldiez@tlmat.unican.es, victor.gonzalezcar@alumnos.unican.es, ramon@tlmat.unican.es

***Abstract*—It is well known that network function virtualization will be a key enabler to meet the stringent requirements of 5G networks. However, fully centralized approaches, such as Cloud Radio Access Network (C-RAN), might not be feasible, considering the particular needs of the fronthaul links and the large cost of implementing such architectural shift. In this sense, flexible functional split brings a practical solution, which trades off performance and practicability. In spite of the growing interest in flexible functional split, little attention has been paid to the interaction of split selection and scheduling. In this paper, we analyze joint strategies that minimize traffic delay. We compare the global optimum solution with partial optimizations, that can be more suitable in practical implementations, using different scenarios. According to our results, fixed scheduling behaves alike the global optimum in heterogeneous RAN scenarios. Furthermore, we observe that it is usually better to optimize the split degree for fixed scheduling setups, than deciding a scheduling policy for a particular split configuration.**

***Index Terms*—functional split, scheduling, 5G, cloud RAN, delay, NFV**

I. Introduction

One of the architectural evolutions that will characterize 5G networks comes from exploiting Software Defined Networking (SDN) techniques, leading to the so-called Network Function Virtualization (NFV) paradigm. While in prior cellular technologies, such as 4G, flat and decentralized architectures were proposed, the challenging 5G requirements call for tighter coordination of the access network elements, which can only be tackled by means of centralized solutions.

This is done by decoupling functionalities belonging to the Radio Access Network (RAN), such that part of them are virtualized and centralized, while the remaining ones stay closer to the Access Point (AP). Initially, fully centralized architectures, C-RAN, were proposed, where the AP, known as Remote Radio Head (RRH), performs only basic Radio Frequency (RF) functions. However, this approach demands high communication capacities between the RRHs and their corresponding Base-Band Unit (BBU), which may not be affordable in real scenarios. For that reason, academia, industry, and standardization bodies [1] [2] are working together to define solutions that permit a flexible selection of the centralization level, or functional split.

This paradigm shift does not only allow a notable cost reduction, but it also brings the possibility of fostering a closer cooperation between RRHs, which is mandatory, given the high density that is expected for the forthcoming cellular network deployments. When virtualizing network functions, one key decision that needs to be taken is the functional split to be used, that is to say, the particular functions moved to the processing units at the BBU and which ones remain close to the RRH. On the other hand, transmission scheduling at the BBU, to the various RRH it might manage, has also a strong impact on the delay, which needs to be kept low, in order to fulfill the stringent requirements of 5G.

In this paper we jointly analyze split configuration and scheduling of a flexible functional-split based architecture. We start from the work of Koutsopoulos [3], where a theoretical description of the problem was given. Shifting more functions to the BBU would imply higher computational load, while it might also lead to different traffic loads over the so-called fronthaul links, which connect the BBU with its corresponding RRHs. Koutsopoulos characterized the complexity of targeting the combined problem, and gave some hints to solve two particular cases, where either the functional split or the scheduling policy were fixed. However, no practical solution was given to any of the different problems.

Hence, the contributions of this paper are:

- We provide an implementation of the solutions that were briefly presented in [3] to minimize delay.
- We evaluate such solutions in different scenarios, and we study their performance using practical values.
- We study whether the use of partial optimization, which is computationally more efficient, would yield a performance similar to the optimum solution.

The paper is structured as follows: in Section II we discuss related works, and we point out the main difference with the work we present herewith. Section III introduces our system model, depicting the various problems that we pose to find the optimum split/scheduling combination. In Section IV we analyze the performance of the different strategies over various scenarios. Finally, we conclude the paper in Section V, which also provides an outlook of our future work.

II. Related work

As mentioned earlier, C-RAN [4] [5] is currently deemed as one of the key enablers to meet the stringent requirements of 5G technology. In a nutshell, the main idea behind this paradigm shift is to move some of the functions that were traditionally placed at the base station to a central controller,

which might be even deployed over general purpose computing nodes. However, fully centralized approaches may be unpractical in real networks, due to the high fronthaul capacity demand, which can only be met with fiber links [6] [7]. In light of it, different initiatives are proposing the re-design of the fronthaul [8], where different splitting configurations can be chosen. The reader may refer to [9] and the references therein for a thorough review of the splits that have been proposed for 5G Networks.

One step forward was the possibility to actually use flexible functional splits. In this case, the idea is to dynamically adapt the functional split, based on the particular delay requirements of the traffic, and the fronthaul conditions. The authors of [10] [11] describe the main characteristics of this approach. On the other hand, a 5G architecture based on flexible splitting is introduced in [12], and an assessment of this concept over a lab-based testbed is discussed in [13].

Exploiting the concept of flexible functional split selection, some works, such as [14] [15], have proposed solutions that seek energy harvesting, or combining it with schemes to manage optical transport, for instance [16] [17]. Furthermore, some studies [18] [19] have highlighted the interaction of flexible functional split and scheduling. In spite of it, little attention has been paid to the joint optimization of scheduling and split selection, which is where the main contribution of this work lies.

III. System Model

As mentioned before, we adopt the system model defined in [3], which we reproduce herewith for completeness.

We consider a functional split architecture where a set of RRHs, $\mathcal{R}$, are connected to a single pool of BBUs through a set of links $\mathcal{L}$. We assume that the BBU is equipped with a single-thread processor, with computational capacity C^B, so that only one frame can be concurrently processed. Similarly, C_i denotes the computational capacity of the RRH i, and L_i is the communication capacity of the link between RRH i and the BBU. Additionally, we just consider downlink communications, although the model can be straightforwardly adapted for uplink.

We assume a slotted-time scenario where, at the beginning of each time-slot, a new frame arrives for every RRH. At the BBU, there is a central controller that globally decides the split for each frame, as well as the order in which frames are scheduled. It is assumed that all RRHs can convey the same set of splits $\mathcal{F}$.

We define $\mathcal{S}$ as the set of possible split configurations. At each slot, the controller selects a split vector $\boldsymbol{s} := \{s_1, ...s_{|R|}\}$, where $s_i \in \mathcal{F}$ is the split decision for RRH $i \in \mathcal{R}$. It is worth noting that the number of possible split configurations (cardinality of $\mathcal{S}$) exponentially grows with the number of split options, $|\mathcal{S}| = |\mathcal{F}|^{|\mathcal{R}|}$.

For a given split decision of frame i, s_i, let ω_{i,s_i} and $\hat{\omega}_{i,s_i}$ be the computational load required to process the frame at the BBU and RRH, respectively. Similarly, we define the function d_{i,s_i} as the amount of data to be transmitted over the link L_i, which depends on the particular split selection.

Similarly, we define the set of possible scheduling policies Π. A particular policy $\pi \in \Pi$ is defined as a vector, $\boldsymbol{\pi} = \{\pi_1, ..., \pi_{|R|}\}$, of integer positive numbers ($\boldsymbol{\pi} \in \mathcal{N}_+^{|\mathcal{R}|}$), where π_i indicates the serving order of frame i. For instance, in a scenario with 4 frames, the scheduling decision $\boldsymbol{\pi} = \{3, 4, 2, 1\}$ would mean that frame 4 is the first to be served, then frames 3, 1 and 2. Note that the space of the scheduling solution set $|\Pi| = |\mathcal{R}|\,!$.

As mentioned earlier, we assume that all the BBU computational resources are devoted to processing the current frame. Hence, the processing delay of a frame i at the BBU can be calculated as:

$$\delta^B_{i,s_i} = \frac{\omega_{i,s_i}}{C^B} \tag{1}$$

Similarly, considering the bits to be transmitted after applying the selected functional split to frame i, the transport delay is defined as:

$$\delta^L_{i,s_i} = \frac{d_{i,s_i}}{L_i} \tag{2}$$

Finally, the delay associated to the processing at the base station is defined as follows:

$$\delta^R_{i,s_i} = \frac{\hat{\omega}_{i,s_i}}{C_i} \tag{3}$$

Altogether, we can define the overall delay suffered by a frame i as:

$$d_i(\boldsymbol{\pi}, \boldsymbol{s}) = \delta^B_{i,s_i} + \delta^L_{i,s_i} + \delta^R_{i,s_i} + \sum_{j:\pi_j<\pi_i} \delta^B_{j,s_j} \tag{4}$$

where $\sum_{j:\pi_j<\pi_i} \delta^B_{j,s_j}$ holds for the time that frame i is waiting in the BBU to be served. Bearing this in mind, we can define the overall system delay D as:

$$\begin{aligned} D(\boldsymbol{s}, \boldsymbol{\pi}) &= \sum_{i\in\mathcal{R}} d_i(\boldsymbol{s}, \boldsymbol{\pi}) = \\ &= \sum_{i\in\mathcal{R}} \left(\delta^B_{i,s_i} + \delta^L_{i,s_i} + \delta^R_{i,s_i} + \sum_{j:\pi_j<\pi_i} \delta^B_{j,s_j} \right) \end{aligned} \tag{5}$$

The global delay defined in Eq. 5 can be reformulated to account for the delay induced by each scheduled frame in the subsequent ones. Then, we can group the delays associated to the processing at the BBU as follows:

$$\begin{aligned} &\sum_{i\in\mathcal{R}} \left(\delta^B_{i,s_i} + \sum_{j:\pi_j<\pi_i} \delta^B_{j,s_j} \right) = \sum_{i\in\mathcal{R}} \sum_{j:\pi_j\leq\pi_i} \delta^B_{j,s_j} = \\ &= \delta^B_{1,s_1} + (\delta^B_{1,s_1} + \delta^B_{2,s_2}) + ... + (\delta^B_{1,s_1} + ... + \delta^B_{|\mathcal{R}|,s_{|\mathcal{R}|}}) = \\ &= \sum_{i\in\mathcal{R}} \delta^B_{i,s_i} \cdot (|\mathcal{R}| - \pi_i + 1) \end{aligned} \tag{6}$$

This way, we can define the system delay in terms of the delay caused by the transmission of each frame i, $g^i_{s_i,\pi_i}$, instead of that experienced by the frames:

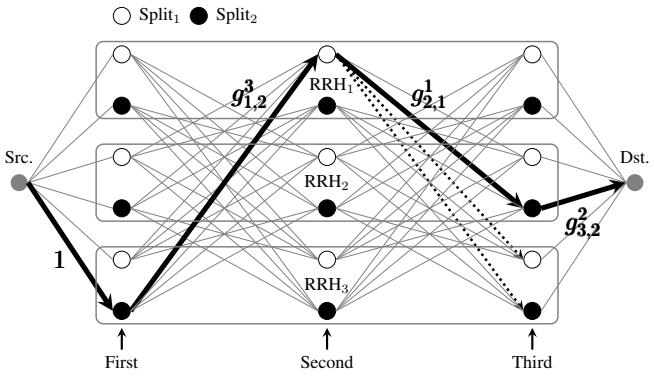


Fig. 1: Problem instance example with 2 RRH, and 2 possible splits. *Src.* and *Dst.* are the virtual nodes added to avoid the trivial solution. Solid bold arrows indicates the selected solution, while dotted arrows denote non-admissible decisions.

$$D(\boldsymbol{s},\boldsymbol{\pi}) = \sum_{i\in\mathcal{R}} g^i_{s_i,\pi_i} = \\ = \sum_{i\in\mathcal{R}} \left(\delta^L_{i,s_i} + \delta^R_{i,s_i} + \delta^B_{i,s_i}(|\mathcal{R}| - \pi_i + 1)\right) \tag{7}$$

A. Problem formulation and decision policies

Over the system model that was depicted above, we aim to minimize the overall system delay. Other objectives may be as well considered, such as minimizing the maximum delay, or maximizing the delay fairness over the different frames.

Following the model proposed in [3], the global delay minimization problem can be posed by representing the system as a Directed Acyclic Graph (DAG). The nodes in the graph form a regular grid with $|\mathcal{R}|$ columns and $|\mathcal{R}| \times |\mathcal{F}|$ rows, where each node corresponds to one split/scheduling decision for one frame, and the nodes are connected by arcs whose weight is the delay due to such decision. Furthermore, the columns represent scheduling order, while the rows correspond to the split and frame selection. In addition, two virtual nodes are added, *Source* and *Destination*, so that the global minimization problem boils down to finding the shortest route between such virtual nodes.

As an example, Figure 1 depicts the resulting graph of a system with 3 RRHs and 2 split levels. As can be observed, two rows are used for all the possible split/scheduling decision for every RRH. In the figure we highlight a possible solution with solid arrows, where the cost of each link corresponds to the delay associated to the decision represented by the source vertex. In the example, frame 3 is first scheduled with split 2, then frame 1 with split 1 and finally frame 2 with split 2. It is worth noting that, after first selecting frame 3, constraints need to be added, to ensure that such frame is not scheduled again. This aspect is reflected in Figure 1 with dotted arrows.

Let $G(\mathcal{V},\mathcal{A})$ be the equivalent system graph, where $\mathcal{A}$ is the set of arcs, and $\mathcal{V}$ the set of vertices, being v_0 and $v_{|\mathcal{R}|+1}$ the *Source* and *Destination* virtual nodes, respectively ($|\mathcal{V}| = |\mathcal{R}|+2$). In addition, we define $\mathcal{V}_i \subseteq \mathcal{V}$ as the subset of nodes that correspond to a particular RRH i. We define a binary decision variable $x_{i,j}$, which takes value 1 if link between nodes i and j is selected, and 0 otherwise. For simplicity, we use w_{ij} to denote the cost of the arc connecting any pair of nodes (i,j). Then, the global delay minimization problem can be defined as:

Problem 1 (Joint scheduling and split decision)**.**

$$\textbf{\textit{min.}} \quad \sum_{i,j} x_{ij}\cdot w_{ij} \tag{8}$$

$$\textbf{\textit{s.t.}} \quad \sum_{i\in\mathcal{V}/k} x_{ik} + \sum_{i\in\mathcal{B}/k} x_{ki} = T_k \qquad \forall k\in\mathcal{V} \tag{9}$$

$$\sum_{i\in\mathcal{V}/\mathcal{V}_i} x_{ik} = 1 \qquad \forall k\in\mathcal{V}_i \tag{10}$$

$$x_{ij}\in\{0,1\} \qquad \forall i,j\in\mathcal{V} \tag{11}$$

where Eq. 9 is the flow conservation constraint. The constant T_k indicates the net incoming and outgoing traffic for each node, which equals 1 and -1 for the *Source* and *Destination*, respectively ($T_0 = 1; T_{|\mathcal{R}|+1} = -1$), and 0 for all other nodes. Then, Eq. 10 ensures that only one decision is taken for each RRH, as shown in Figure 1 by the dotted arrows. The value of w_{ij} follows the same pattern depicted in Figure 1, taking infinite value for the those cases that are not represented. As can be observed, the resulting problem is a Binary Linear Program (BLP), which is known to be *np-complete* and so difficult to solve. Furthermore, the size of the joint problem grows exponentially with both the number of RRHs and possible splits, being the space of candidate solutions $|\mathcal{F}|^{|\mathcal{R}|}\times|\mathcal{R}|!$ and the number of variables $2|\mathcal{R}||\mathcal{F}| + (|\mathcal{R}|-1)\times|\mathcal{F}||\mathcal{R}| \times (|\mathcal{F}||\mathcal{R}| - |\mathcal{F}|)$[1].

In the following, we elaborate how the general joint problem can be simplified when either scheduling or split configuration are fixed.

B. Fixed split selection

If we fix the split of each RRH, we can know the delay values, so the objective is to minimize the product of the delay in the BBU by the multiplicative factor associated with the scheduling. It can be observed that the global minimization is achieved with the shortest-job-first policy.

C. Fixed scheduling

When the scheduling is fixed, the problem dimension is reduced, so it boils down to select the split that minimizes the expression $\delta^L_{i,s_i} + \delta^R_{i,s_i} + \delta^B_{i,s_i}(|\mathcal{R}| - \pi_i + 1)$, where π_i is known, for each RRH. Considering that, in practice, the number of possible splits is low, this task can be efficiently accomplished by regular search algorithms.

[1]The first term, $2|\mathcal{R}||\mathcal{F}|$, corresponds to the arcs from and towards virtual nodes. In the second term we multiply the number of columns $(|\mathcal{R}|-1)$ by the number of rows $|\mathcal{R}||\mathcal{F}|$ and outgoing arcs from each node $(|\mathcal{F}||\mathcal{R}|-|\mathcal{F}|)$.

IV. Performance Evaluation

As we have seen before, the complexity of the minimum delay problem, in particular for large networks, may actually hinder its practical use. For that reason, it becomes necessary to explore alternative approaches and to analyze under which circumstances those could be applied.

In this section we analyze the performance of the global delay minimization posed in Problem 1, and we compare it with the partial solutions described in the previous section. In this sense, we deploy 3 different scenarios where we vary the two main parameters that affect the underlying optimization problem: computational capacity of the RRHs compared to that of the BBU, and frame lengths.

To better understand the impact that the scenario characteristics play in the behavior of the optimization schemes, we define them relative to the computational capacity of the BBU, C^B. For that, we use the ratio between RRH and BBU processing capacities, $r_i^R = C_i/C^B$, and the BBU processing delay, δ^B_{i,s_i}, to model the scenario. As for the transport delay, we assume that high capacity links are used [20], and we thus neglect the communication delay, compared to others.

In general, the scenarios are made of 1 BBU and 10 RRHs, and for each configuration and optimization scheme we run 1000 independent experiments. Network element computational capacities are based on real models described in [21], [22], while we used typical distributions of Internet packets [23], [24] to establish frame lenghts. In addition, the split or centralization level is defined as the relation of the computational load required to process the frames. We thus define the centralization degree of RRH i for a particular split s, as $w_{i,s_i}/(w_{i,s_i} + \hat{w}_{i,s_i})$, see [25], [26] for further details. The joint optimization described in Problem 1 is solved using the GLPK library [27], and for each scenario its outcome is compared with:

- Fixed scheduling scheme where frames are scheduled according to their length, from shortest to largest.
- Fixed split solutions, applying centralization degree of 0, 50, and 100%.

A. Homogeneous RRH and heterogeneous traffic

In the first scenario we consider a homogeneous access network where the computational capacity of the RRHs is the same for all the access components ($r_i^R = r^R \quad \forall i \in \mathcal{R}$), while the BBU processing delay, δ^B, is uniformly distributed within the range $[1, 1000]$ μs in each experiment.

Figure 2 depicts the average total delay of the frames, as well as the 95% confidence interval, for different relative computational capacity of the RRHs. As can be seen, the fixed scheduling scheme (shortest to largest) yields results similar to the global optimum, regardless of r^B. On the other hand, the fixed split solution needs to be tailored to the ratio of RRH and BBU computational capacities. In this sense, when the RRHs are less capable, the best behavior is observed for the highest centralization degree (i.e. c-RAN solution), while more functionalities should be shifted to the RRHs as they become more capable.

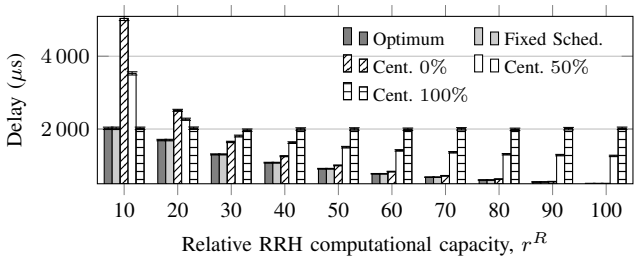


Fig. 2: Average delay per frame with heterogeneous frame lengths and for different relative computational capacity of RRHs

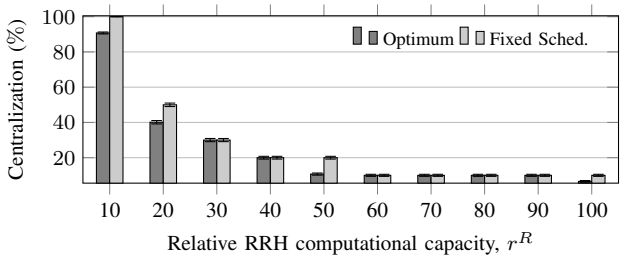


Fig. 3: Average centralization degree with heterogeneous frame lengths and for different relative computational capacity of RRHs

As for the split selection, Figure 3 shows the centralization degree obtained by both the global optimum and the fixed scheduling approaches. As can be observed, both solutions have a very close behavior, and they are able to adapt the split choice according to the particular scenario characteristics.

While the overall trend leading to less centralization as the RRHs become more capable was expected, the results also indicate that a fixed scheduling policy migth indeed be used in scenarios where the access elements are similar, in terms of computational capacity.

B. Heterogeneous RRH and homogeneous traffic

In the second scenario we fix the frame length, leading to a homogeneous traffic setup, which we represent by means of a fixed value of the BBU computational delay, δ^B. On the other hand, we randomly select the relative computational capacity of the RRHs, r_i^R, in each deployment, following a uniform distribution within the range $[0.1, 1]$.

First, in Figure 4 we illustrate the average delay affecting the frames, along with the confidence interval for each setup and optimization scheme. As could have been expected, the average frame delay increases with the size of the frames, regardless the algorithm applied. In addition, we can clearly see that the fixed scheduling policy always outperforms all the fixed centralization setups.

Similarly to the previous scenario, Figure 5 depicts the average centralization degree chosen by the global optimization and the fixed scheduling solutions. As can be seen, for shorter frames (lower value of δ^B) the optimum solution leads to

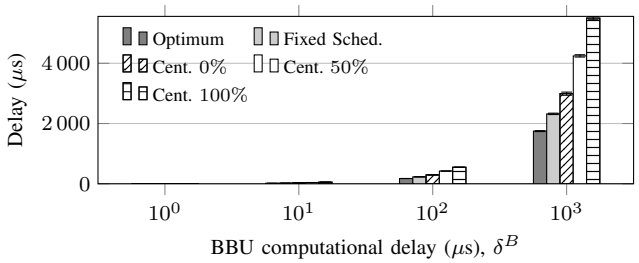


Fig. 4: Average delay per frame with heterogeneous RRHs relative computational capacity and for different frame lengths represented by BBU delay

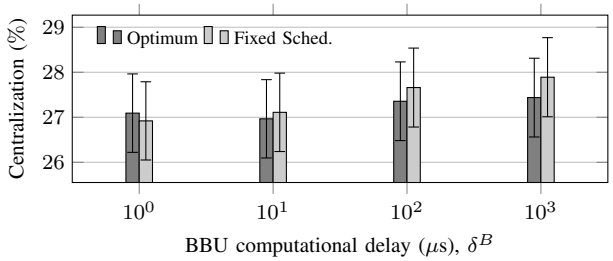


Fig. 5: Average split level with heterogeneous RRHs relative computational capacity and for different frame lengths represented by BBU delay

higher centralization degree, while the behavior is the opposite one as the frame length increases.

C. Heterogeneous RRH and heterogeneous traffic

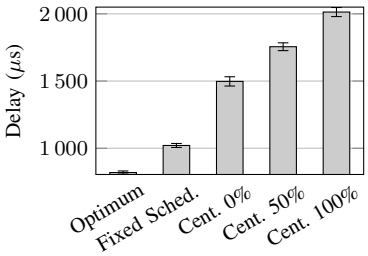
In the last scenario we randomly choose both the frame length and RRH computational capacity for each experiment. We use the distributions of the previous scenarios, so that the relative computational capacity of the RRH, r_i^R, and the BBU processing delay, δ^B, are within the intervals $[0.1, 1]$ and $[1, 1000]$ μs, respectively.

In Figure 6a we depict the average delay experienced by the frames when using the different optimization schemes. Similarly to the heterogeneous RRH scenario, we can observe that the fixed scheduling policy, although far from the optimum behavior, always outperforms the fixed centralization alternatives.

In addition, Figure 6b shows that, in average, the optimum behavior tends to higher centralization degrees.

V. Conclusion

We have analyzed global delay minimization algorithms for flexible functional split scenarios, where both split selection and frame scheduling have a strong impact on traffic delay. Hence it becomes necessary to jointly consider both aspects to yield the best possible performance in terms of delay. However, the underlying optimization problem results in a BLP, which hinders its applicability in practical scenarios, since it is known to be *np-complete*. We have thus considered some alternatives that, thanks to some assumptions, notably simplify the previous problem.



(a) Average delay

(b) Average centralization

Fig. 6: Algorithms' performance with heterogeneous frame length and computational capacity of RRHs

Based on the work of Koutsopoulos [3], we implemented the joint problem, as well as partial optimization alternatives, where either the scheduling or the functional split are fixed. We then analyzed the performance of the simplified alternatives in different scenarios, by comparing the observed delays with those offered by the global optimum. The results show that, in scenarios where the access network is homogeneous in terms of computational capacity, the delay obtained when using a fixed scheduling policy is statistically similar to that brought by the global optimum. In addition, we also saw that the particular split configuration that was found when the scheduling was fixed is also rather close to the optimum one. On the other hand, fixed split policies yield worse performances, and they are not able to adapt to different scenario setups.

We also studied the performance of the different alternatives in heterogeneous networking scenarios. In this case, the performance obtained by the partial optimizations are not as good as those yielded by the global optimization approach. Nevertheless, our results show that fixed scheduling policies always lead to lower delays than alternatives where the degree of centralization is constant.

In the future we will analyze additional alternatives to the global optimization approach, specially in scenarios with access elements having different computational capacities. In particular, clusterization techniques, where frames or access elements with similar properties are grouped together, may simplify the corresponding problem. Furthermore, we will tackle more complex scenarios, considering the temporary evolution of frame arrival at the BBU. In this case, we will need to use online solutions, exploiting results from queueing theory.

Acknowledgment

This work has been supported by the Spanish Government (Ministerio de Economía y Competitividad, Fondo Europeo de Desarrollo Regional, FEDER) by means of the projects ADVICE: Dynamic provisioning of connectivity in high density 5G wireless scenarios (TEC2015-71329-C2-1-R) and Future Internet Enabled Resilient Cities (FIERCE).

REFERENCES

[1] I. . W. Group, "Next generation fronthaul interface." [Online]. Available: http://sites.ieee.org/sagroups-1914/

[2] "Study on new radio access technology: Radio access architecture and interfaces," 3rd Generation Partnership Project (3GPP), TR 38.801, 2017.

[3] I. Koutsopoulos, "Optimal functional split selection and scheduling policies in 5g radio access networks," in *2017 IEEE International Conference on Communications Workshops (ICC Workshops)*, May 2017, pp. 993–998.

[4] J. Wu, Z. Zhang, Y. Hong, and Y. Wen, "Cloud radio access network (c-ran): a primer," *IEEE Network*, vol. 29, no. 1, pp. 35–41, Jan 2015.

[5] 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*, vol. 17, no. 1, pp. 405–426, Firstquarter 2015.

[6] G. O. Pérez, J. A. Hernández, and D. Larrabeiti, "Fronthaul network modeling and dimensioning meeting ultra-low latency requirements for 5g," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 10, no. 6, pp. 573–581, June 2018.

[7] A. Garcia-Saavedra, J. X. Salvat, X. Li, and X. Costa-Perez, "Wizhaul: On the centralization degree of cloud ran next generation fronthaul," *IEEE Transactions on Mobile Computing*, vol. 17, no. 10, pp. 2452–2466, Oct 2018.

[8] C. I, Y. Yuan, J. Huang, S. Ma, C. Cui, and R. Duan, "Rethink fronthaul for soft ran," *IEEE Communications Magazine*, vol. 53, no. 9, pp. 82–88, Sep. 2015.

[9] L. M. P. Larsen, A. Checko, and H. L. Christiansen, "A survey of the functional splits proposed for 5g mobile crosshaul networks," *IEEE Communications Surveys Tutorials*, vol. 21, no. 1, pp. 146–172, Firstquarter 2019.

[10] D. Harutyunyan and R. Riggio, "Flex5g: Flexible functional split in 5g networks," *IEEE Transactions on Network and Service Management*, vol. 15, no. 3, pp. 961–975, Sep. 2018.

[11] ——, "Flexible functional split in 5g networks," in *2017 13th International Conference on Network and Service Management (CNSM)*, Nov 2017, pp. 1–9.

[12] P. Arnold, N. Bayer, J. Belschner, and G. Zimmermann, "5g radio access network architecture based on flexible functional control / user plane splits," in *2017 European Conference on Networks and Communications (EuCNC)*, June 2017, pp. 1–5.

[13] Y. Alfadhli, M. Xu, S. Liu, F. Lu, P. Peng, and G. Chang, "Real-time demonstration of adaptive functional split in 5g flexible mobile fronthaul networks," in *2018 Optical Fiber Communications Conference and Exposition (OFC)*, March 2018, pp. 1–3.

[14] D. A. Temesgene, M. Miozzo, and P. Dini, "Dynamic functional split selection in energy harvesting virtual small cells using temporal difference learning," in *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Sep. 2018, pp. 1813–1819.

[15] L. Wang and S. Zhou, "Flexible functional split in c-ran with renewable energy powered remote radio units," in *2018 IEEE International Conference on Communications Workshops (ICC Workshops)*, May 2018, pp. 1–6.

[16] A. Marotta, D. Cassioli, K. Kondepu, C. Antonelli, and L. Valcarenghi, "Efficient management of flexible functional split through software defined 5g converged access," in *2018 IEEE International Conference on Communications (ICC)*, May 2018, pp. 1–6.

[17] Y. Li, J. Mårtensson, M. Fiorani, B. Skubic, Z. Ghebretensaé, Y. Zhao, J. Zhang, L. Wosinska, and P. Monti, "Flexible ran: A radio access network concept with flexible functional splits and a programmable optical transport," in *2017 European Conference on Optical Communication (ECOC)*, Sep. 2017, pp. 1–3.

[18] C. Chang, N. Nikaein, and T. Spyropoulos, "Impact of packetization and scheduling on c-ran fronthaul performance," in *2016 IEEE Global Communications Conference (GLOBECOM)*, Dec 2016, pp. 1–7.

[19] C. Chang, N. Nikaein, R. Knopp, T. Spyropoulos, and S. S. Kumar, "Flexcran: A flexible functional split framework over ethernet fronthaul in cloud-ran," in *2017 IEEE International Conference on Communications (ICC)*, May 2017, pp. 1–7.

[20] Q. C. Li, H. Niu, A. T. Papathanassiou, and G. Wu, "5g network capacity: Key elements and technologies," *IEEE Vehicular Technology Magazine*, vol. 9, no. 1, pp. 71–78, Mar. 2014.

[21] P. Rost, I. Berberana, A. Maeder, H. Paul, V. Suryaprakash, M. Valenti, D. Wübben, A. Dekorsy, and G. Fettweis, "Benefits and challenges of virtualization in 5g radio access networks," *IEEE Communications Magazine*, vol. 53, no. 12, pp. 75–82, Dec. 2015.

[22] K. Wang, K. Yang, H. Chen, and L. Zhang, "Computation diversity in emerging networking paradigms," *IEEE Wireless Communications*, vol. 24, no. 1, pp. 88–94, Feb. 2017.

[23] X.-L. Wu, W.-M. Li, F. Liu, and H. Yuand, "Packet size distribution of typical internet applications," in *2012 International Conference on Wavelet Active Media Technology and Information Processing (ICWAMTIP)*, Dec. 2012, pp. 276–281.

[24] Z. Sun, D. He, L. Liang, and H. Cruickshank, "Internet qos and traffic modelling," *IEE Proceedings - Software*, vol. 151, no. 5, pp. 248–255, Oct. 2004.

[25] N. Makris, P. Basaras, T. Korakis, N. Nikaein, and L. Tassiulas, "Experimental evaluation of functional splits for 5g cloud-rans," in *2017 IEEE International Conference on Communications (ICC)*, May 2017, pp. 1–6.

[26] D. Wubben, P. Rost, J. S. Bartelt, M. Lalam, V. Savin, M. Gorgoglione, A. Dekorsy, and G. Fettweis, "Benefits and impact of cloud computing on 5g signal processing: Flexible centralization through cloud-ran," *IEEE Signal Processing Magazine*, vol. 31, no. 6, pp. 35–44, Nov. 2014.

[27] G. Project, "Gnu linear programming kit." [Online]. Available: https://www.gnu.org/software/glpk/