HARQ-aware allocation of computing resources in C-RAN

Francesca Bassi, Hatem Ibn-Khedher

HARQ-aware allocation of computing resources in C-RAN

Francesca Bassi* and Hatem Ibn Khedher†
*Institut de Recherche Technologique SystemX, 8 avenue de la Vauve, 91120 Palaiseau, France
†LIPIADE, University of Paris Descartes, Sorbonne Paris Cité, 45 rue des Saints Pères, 75006 Paris, France
francesca.bassi@irt-systemx.fr, hatem.ibn-khedher@parisdescartes.fr

***Abstract*—The principal tenet of C-RAN is the softwarization of the base-band signal processing, which enables the sharing of computing resources among multiple radio heads. When the aggregate demand exceeds the processing capacity, a fraction of the radio packets is lost at PHY layer. Traditional computing resource allocation policies aim to minimize the packet loss rate.**

Dropping a PHY packet triggers a retransmission, unless the lost packet corresponds to the last available HARQ round, in which case the entirety of the radio resources spent on the multiple transmissions go to waste. This suggests that allocating computing resource accounting also for the HARQ transmission history may make a more efficient use of the bandwidth.

We consider a simplified LTE uplink setting, and we measure the performance at the lower MAC layer (accuracy, goodput and average delay). We first compare the PHY-layer loss rate minimization and the cross-layer approaches using an ILP formulation. The cross-layer approach brings a tangible improvement, especially in accuracy. This suggests, for future work, that joint radio and computing resource allocation may further enhance spectral efficiency. We finally propose a probabilistic algorithm amenable to real-time operation which allows to mix strategies via parameter tuning, and we use it to explore the region of achievable goodput/accuracy trade-offs.

I. Introduction

Cloud-RAN or C-RAN [1], [2] is a radio access paradigm for mobile networks capable of responding to the economic challenges of network densification [3]. In the traditional mobile architecture users access the network by connecting to a radio base station. The base station can be functionnally separated into radio functions, for the transmission of the analog radio-frequency waveform, and base-band functions, for the digital processing before radio-frequency conversion. The enabler of C-RAN is the softwarization of the base-band functions, which allows to split the base station into a Radio Head (RH) and a Base-Band Unit (BBU). As the base station in the traditional architecture, the RH needs to be geographically deployed to provide radio coverage, while the BBU can now run remotely, in a data center that simoultaneously serves a multitude of RHs.

A big challenge for C-RAN is the respect of the latency constraints imposed by the communication protocol stack [4]. In LTE, for instance, the HARQ protocol triggers the retransmission of a radio packet if the correct reception of the message has not been acknowledged within 8 milliseconds. This imposes a hard deadline of 3 milliseconds to complete the base-band processing of a received subframe, accounting for both the propagation time in the fronthaul and the processing time in the BBU. Latency, bandwith and cost issues of the fronthaul are the object of many studies (see [5] and references therein). In order to solve some of these issues, alternative functional splits have been proposed [6], where also a portion of the base-band processing takes place in the distributed unit, along with the radio functions. In this work we stick to the standard functional split considered in [1], [2], with remote RHs and central BBUs.

The policy of allocation of the shared computing resources has an impact in many respects. Some works investigate strategies to optimize the power consumption [7]; some consider demand anticipation and resource provision [8]; some try to to reduce the overall processing time exploiting parallelization [9], [10]; some focus on the orchestration of the virtualized functions [11]. The problem is especially interesting in the uplink, where the central BBU is in charge of channel decoding. Since each user transmits with a different Modulation Coding Scheme (MCS), and since the processing times for different MCSs vary widely [12], it may be difficult to anticipate the instantaneous demand and allocate enough computing resources, so that packets may not get processed within the required deadline. We model these events as packet losses at PHY layer. The computing resource allocation algorithms found in the literature generally aim to minimize the packet loss rate [9], [14], which is a very natural choice.

A packet loss triggers a negative acknowledgement in the HARQ protocol at MAC layer. Hence, if the lost packet corresponds to the last allowed transmission, the message will never be correctly received, and the system squandered all the radio resources spent on it with the last *and previous* transmissions. Starting from this simple observation, we explore whether a cross-layer policy for allocating the computing resources may use the bandwidth more efficiently. We still assume that radio scheduling and computing resource allocation are separate processes, but we account in the latter for the retransmission age of the message, *i.e.*, for the HARQ round.

We consider a simplified version of the LTE uplink, where each subframe is occupied by a single user, and we measure the performance through the Bit Error Rate (BER), the average delay and the Goodput at MAC layer. The Goodput is defined as the (long-term) average number of correctly received bits per unit of time [13]. The system and simulations setups are described in detail in Sections II and III, respectively. The allocation policy that minimizes the PHY packet losses is

formulated as an Integer Linear Program [14] in Section IV, where we turn it in a cross-layer policy by incorporating the HARQ retransmission age of the message in the objective function. This achieves a consistent improvement in the BER, for the same Goodput, at the expense of the average delay. Resource allocation via ILP is not practically viable because of its complexity. In Section V we propose a probabilistic algorithm, which is amenable to real-time operation and allows to implement the same strategies of the ILP formulation by appropriately tuning its three parameters, incurring a modest performance degradation. The low complexity of the probabilistic algorithm allows to exhaustively explore the space of the parameters, in Section V-A. This corresponds to probing all mixed strategies, which is unfeasible with the ILP formulation. It is confirmed that the strategy providing the the most effective Goodput/accuracy trade-off consists in allocating computing resources in priority to subframes with high information content and old retransmission age.

II. System setup and problem definition

We consider a C-RAN architecture [1], [2] and we focus on the uplink from multiple RHs towards a centralized BBU. We assume that the radio scheduling, the PHY layer and the lower MAC layer work as in LTE. Let $\mathcal{N}$ be the set of the RHs, which we assume operate in the 20 MHz bandwidth, so that the number of physical resource blocks per subframe is 100 [12], [14], for all RH. We assume that each subframe carries data of a single user and uses only one MCS.

In LTE the PHY subframes originate from the transmission of MAC messages via a stop-and-wait HARQ process [15]. We consider synchronous HARQ, and each RH treats $P = 8$ parallel messages in a pre-defined order. The maximum number of transmissions per message is $T_{\max} = 4$. Each subframe i is tagged by $r_i \in \{1, 2, ..., T_{\max}\}$, which indicates which HARQ round the subframe represents. Since the HARQ is synchronous the tag is known at the BBU and does not need to be signalled explicitly. We assume non-adaptive HARQ, so that the MCS stays constant across HARQ retransmissions. To allow HARQ control messages to be received on time each subframe must be processed by the BBU within $d = 2$ TTIs from reception, otherwise it is lost.

The BBU pool consists of the set $\mathcal{C}$ of CPU cores. Each core in $\mathcal{C}$ has the same execution speed and can process at most one subframe at a time. Processing is not parallelized and a subframe fully occupies a core for the entire duration of its processing. The MCS of subframe i determines its processing time t_i, as well as its information content of b_i information bytes [12]. Each RH presents a subframe to the BBU each TTI, which in LTE has a duration of 1 ms. For simplicity, we assume that $K = \frac{|\mathcal{C}|}{d}$ is an integer value. At each TTI the resource allocation problem consists in reserving, for each of the $|\mathcal{N}|$ most recently received subframes, the appropriate amount of time in one of the K available CPU cores, so that the subframe is processed within the deadline d. At each TTI, if the system is in moderate load the expected value of the demand of processing time is smaller or equal to the available processing time, i.e. $\sum_{i \in \mathcal{N}} \mathbb{E}[t_i] \leq dK$. The expected number of PHY subframes losses in the TTI is zero. Otherwise, the system is in overload, and losses of PHY subframes are expected. Since in our setup all the RHs present a subframe at each TTI, all the subsets $\mathcal{K}$ experience the same level of load (moderate or overload), depending on the value $|\mathcal{N}|$. So the resource allocation can be performed in independent subgroups of K cores each, without loss of optimality.

We want to investigate the impact of the computing resource allocation policy on the metrics observed at MAC layer. We consider the Bit Error Rate (BER), defined as the number of incorrectly received information bits over the total number of transmitted information bits (an information bit is considered incorrectly received if the subframe it belongs to has not been ACKed after $T_{\max}$ transmissions); the Goodput, defined as the average number of information bits correctly received per TTI; the average delay, defined as the average number of transmissions for successfully received MAC logical unit. At PHY layer, we consider the Subframe Rejection Rate (SRR), defined as the number of lost (because of the resource allocation policy) subframes over the total number of subframes demanding processing; and the Bit Rejection Rate (BRR), defined as the number of lost information information bits over the total number of information bits demanding processing.

III. Simulation setup and parameter choice

The performance of the considered computing resource allocation strategies are obtained using Monte Carlo simulations, whose setup is as follows. Each of the $|\mathcal{N}|$ RHs manages a synchronous HARQ process with $P = 8$ parallel data streams. All transmissions of the same message use the same MCS (see Section III-A on MCS sampling). At each TTI each RH presents a subframe to the BBU, and the computing resource allocation algorithm specifies the set of subframes which get processed. The algorithm may use information about retransmission tags (known because of synchronous HARQ), information content (dependent on MCS) and processing time (dependent on MCS) of the subframes (see Section III-B). Subframes rejected by the BBU trigger NACKs in the HARQ process. Subframes accepted by the BBU might be successfully (triggering ACK) or unsuccessfully (triggering NACK) decoded. The decoding outcome is sampled using precomputed probabilities dependent on the retransmission tag and MCS of the subframe (see Section III-C). The maximum number of HARQ transmissions is $T_{\max} = 4$. The simulation lasts 500 TTI, and the performance is evaluated in terms of BER, Goodput, average delay and SRR, BRR. The simulations use the GLPK optimization library to solve the ILP.

A. MCS sampling

When a MAC message is generated it is associated with an MCS, which is constant for all HARQ transmissions. For MCS sampling we use, as in [14], an empirical distribution evaluated from the data set in [16]. The data set [16] records real traffic in 4 mobile cells and reports the MCS occurrence. The resulting empirical distribution has a median MCS equal

to 8, and a mean MCS equal to 8.62. The sampled MCS is between 4 and 14 with probability 0.9.

B. Processing time and information content

Both the information content b_i in information bits and the processing time t_i of subframe i can be determined by its MCS. As the MCS increases, both t_i and b_i increase. In this work we use the numerical values found in [12].

C. Decoding failure probabilities

In order to sample the decoding outcome (ACK or NACK) of processed subframes, the Monte Carlo simulation needs π_k, defined as be the probability that a message is NACKed after the k-th transmission. To estimate these probabilities we make use of a data set obtained by running the LTE PHY layer on Open Air Interface (OAI). The experiment concerns the transmission in uplink (from User Equipment to RH) of a MAC message. The PHY subframe is composed by 100 resource blocks. The MCS of the subframe and the SINR are constant across HARQ retransmissions. The receiver performs Chase Combining and the maximum number of transmissions is $T_{\text{max}} = 4$. The experiment is repeated 500 times for any configuration of $(\text{SINR}, \text{MCS})$, with $\text{MCS} \in \{0, 2, 4, ..., 26\}$ and $\text{SINR} \in [-5, 18]$ dB. The data set records the average Block Error Rate (BLER_k) after transmission k, $k \in \{1, 2, ..., T_{\text{max}}\}$. BLER_k is the proportion of experiments where the MAC message is in NACK after k transmissions have elapsed. An estimate $\overline{\pi}_k$ is obtained dividing the number of times NACK happened after the k-th transmission by the total number of times a unit has been transmitted at the k-th round. This is calculated from the dataset as $\overline{\pi}_1 = \text{BLER}_1$ for $k = 1$, and for $k > 1$ as $\overline{\pi}_k = \frac{\text{BLER}_k}{\text{BLER}_{k-1}}$. From the dataset we hence get the estimates $\overline{\pi}_k(\text{MCS}, \text{SINR})$ for $k \in \{1, ..., T_{\text{max}}\}$. In LTE the radio scheduler allocates the radio resources (radio resource blocks, power and MCS) such that $\text{BLER}_1 \leq 0.1$. Using this information, we select for each MCS the SINR which provides $\text{BLER}_1 \leq 0.1$. We then obtain estimates $\overline{\pi}_k(\text{MCS})$ parametrized only by the MCS.

IV. Resource allocation as an optimization problem

Given the K available cores and the set of instantaneous processing demands, the resource allocation problem consists in identifying the subset of subframes which will be processed and their execution cores. The problem can be formulated as an Integer Linear Program whose solution (evaluated each TTI) is the computing resource allocation policy.

This formulation has already been explored in [14], where the optimization problem is expressed as follows:

$$\text{maximize} \quad \sum_{i \in \mathcal{N}} \sum_{c \in \mathcal{K}} x_i^c \tag{1}$$

$$\text{subject to} \quad x_i^c \in \{0, 1\}, \ \forall i \in \mathcal{N}, \ \forall c \in \mathcal{K}, \tag{2}$$

$$\sum_{c \in \mathcal{K}} x_i^c \leq 1, \ \forall i \in \mathcal{N}, \tag{3}$$

$$\sum_{i \in \mathcal{N}} t_i \ x_i^c \leq d, \ \forall c \in \mathcal{K}. \tag{4}$$

Constraint (2) says that x_i^c is a binary quantity. The value $x_i^c = 1$ means that subframe i gets assigned to core c for processing. Constraint (3) imposes that each subframe is processed at most once, and constraint (4) imposes that the processing of all the subframes allocated to the same core is completed within the deadline d. The objective (1) is to schedule a maximum number of subframes. In case of moderate load all requests get satisfied. In case of overload subframes associated with a smaller t_i get processed with higher priority, which maximizes the number of subframes which get processed. Since t_i increases with the MCS [12], subframes with high MCS have a bigger risk of being lost.

In [14] it is also considered an objective function accounting for the information content of the subframe. The constraints (2), (3), (4) remain the same, while the objective function is

$$\text{maximize} \quad \sum_{i \in \mathcal{N}} \sum_{c \in \mathcal{K}} x_i^c \ b_i, \tag{5}$$

where b_i is the information content in bytes of subframe i. The allocation strategy corresponding to the ILP with objective function (5) gives higher priority to subframes with high information content (high MCS). Since these are also the most demanding in processing time, this strategy must find the best trade-off between information content and execution time.

We propose a cross-layer resource allocation policy via ILP formulation, with constraints (2), (3), (4) and the following objective function:

$$\text{maximize} \quad \sum_{i \in \mathcal{N}} \sum_{c \in \mathcal{K}} x_i^c \ \left(\epsilon + \alpha \ \frac{r_i}{T_{\text{max}}} + \beta \ \frac{b_i}{B} \right), \tag{6}$$

where ϵ, α and β are parameters taking values in $\{0, 1\}$. The parameter B in (6) is the maximum information content of a subframe, which corresponds to the information content of a subframe with maximum MCS [12]. The quantity b_i/B is hence the normalized information content of subframe i. Similarly, r_i/T_{max} indicates the normalized transmission age of subframe i. Notice that for $(\epsilon, \alpha, \beta) = (1, 0, 0)$ the objective function (6) becomes (1); for $(\epsilon, \alpha, \beta) = (0, 0, 1)$ the objective function (6) becomes equivalent to (5). When $(\epsilon, \alpha, \beta) = (0, 1, 0)$ we obtain a scheduling policy where older subframes in the HARQ protocol have higher priority. We also consider $(\epsilon, \alpha, \beta) = (0, 1, 1)$, strategy where the highest priority is given to subframes with big information content and also old in the HARQ process.

The solid lines in Figures 1 and 2 present the metrics, at PHY layer and MAC layer respectively, of the resource allocation algorithm via ILP formulation (6), (2), (3), (4). In this setting $K = 1$. The legend indicates the vector $(\epsilon, \alpha, \beta)$ used to define the objective function (6). All metrics are shown as a function of the number of connected RHs. As it can be seen in Figure 1 from the average CPU load (expressed as the fraction of the scheduled computation resources), when more than 25 RHs are connected the system is in overload. The SRR in Figure 1 shows that all considered strategies provide similar results on the number of processed PHY subframes. As expected $(\epsilon, \alpha, \beta) = (1, 0, 0)$ (yellow curve) gives the best

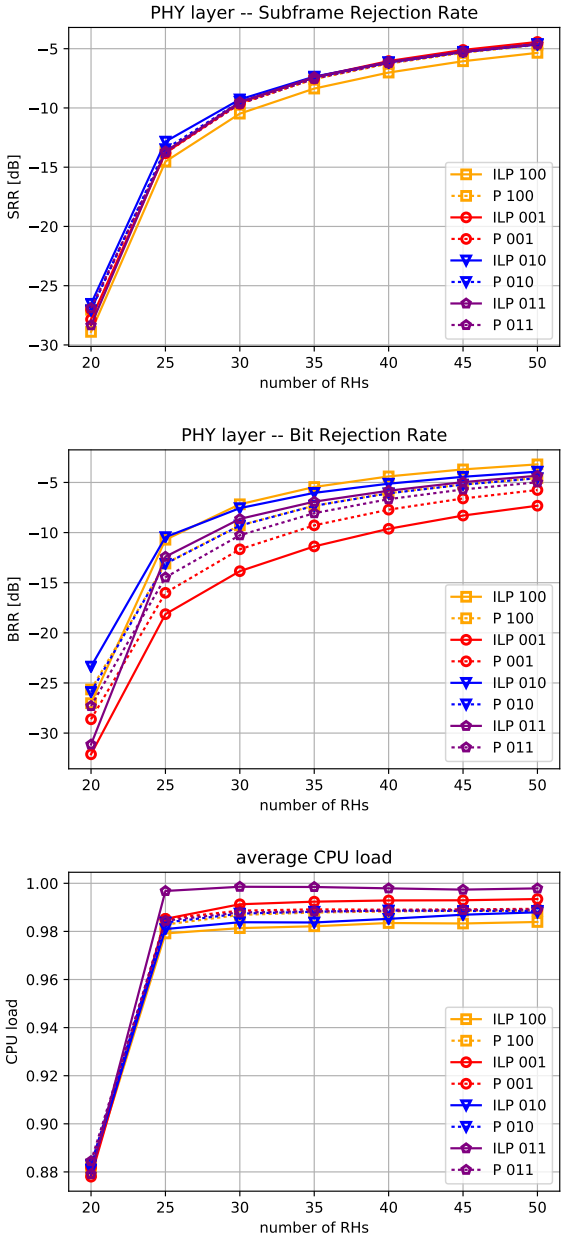

Fig. 1. PHY-layer performance of the scheduling algorithm via ILP.

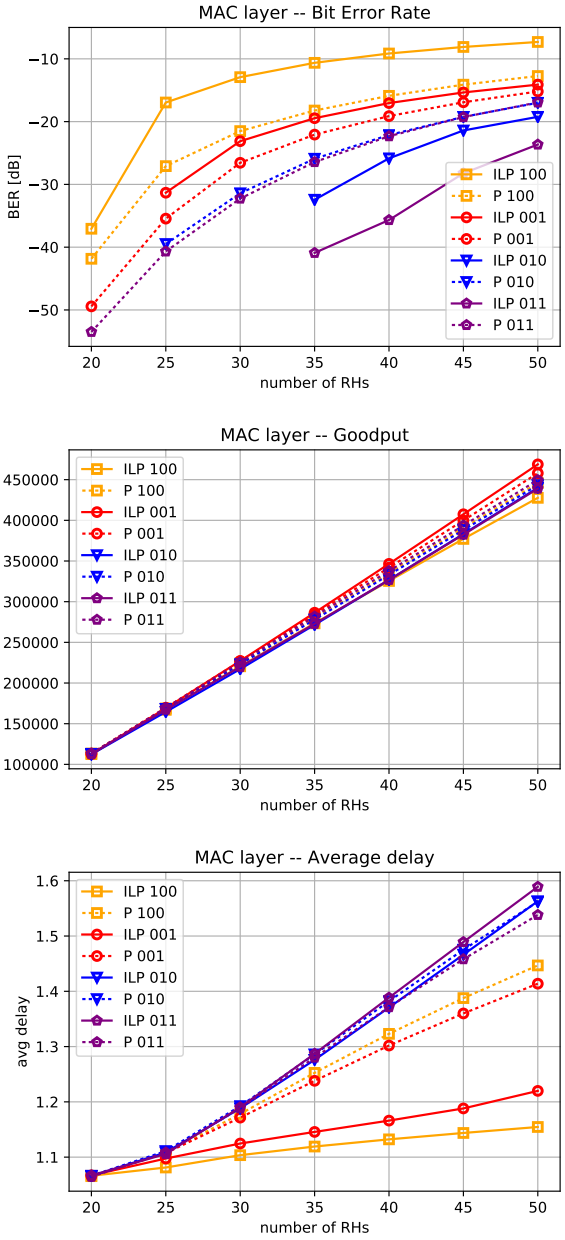

Fig. 2. MAC-layer performance of the scheduling algorithm via ILP.

SRR, which is however only less than 1 dB away of the worst case. This suggests that PHY losses are not a very good metric to guide the choice of the allocation policy, since there is not a strong reason to prefer a policy over the others based on SRR only. On the other hand, the BRR in Figure 1 shows that the ILP for $(\epsilon, \alpha, \beta) = (0, 0, 1)$ (red curve), provides the best BRR [14], and that the improvement over to the worst case (yellow curve) is substantial (over 5 dB). The intuition that optimizing the number of processed information bits a PHY layer is a good proxy for optimizing the Goodput at MAC layer is confirmed by the plot of the Goodput in Figure 2, where we can see that the red curve is the best performance.

Figure 2 shows the MAC layer metrics, on which we evaluate, in this work, the performance of the allocation policy. Inspecting the BER, we can see that maximizing the number of processed subframes (yellow curve) and even maximizing the number of processed information bits (red curve) are suboptimal with respect to ILP for $(\epsilon, \alpha, \beta) = (0, 1, 0)$ (blue curve), which gives priority to older subframes. This confirms that considering also the retransmission age brings the performance improvement in accuracy we hoped for. This comes at the cost of a degradation of the average delay, which however does not impact too dramatically the Goodput, as visible in Figure 2. Finally, notice that optimizing the number of

processed subframes (yellow curve) gives, albeit achieving the best average delay, the worst Goodput among the considered strategies, due to the very poor accuracy, as seen in the plots in Figure 2.

The two good strategies that emerged so far can be combined. Maximizing the processed information bits while giving priority to older subframes, as done in ILP with $(\epsilon, \alpha, \beta) = (0, 1, 1)$ (purple curve), further increases the gain in terms of BER of the $(0, 0, 1)$ strategy (blue curve), without degrading in a very significant way the average delay and keeping the Goodput close to the blue curve. This strategy globally achieves a good trade-off between accuracy and Goodput.

V. The probabilistic algorithm

Computing resource allocation via ILP formulation is not a viable solution for real systems because it is a computationally complex problem which needs to be solved very frequently (at each TTI). In this section we propose an alternative algorithm, which draws inspiration from the results of Section IV. The algorithm consists in repeatedly sampling elements from the set of received subframes and scheduling them for processing as long as computing resources are still available. More precisely, let $\mathcal{S}$ be the set of subframes demanding processing resources at a given TTI, and let $\mathcal{T}_\mathcal{K}$ be the processing time available in the BBU. The algorithm works in rounds. At the beginning of each round, the subframe i is sampled with uniform distribution from $\mathcal{S}$. If $t_i \leq \mathcal{T}_\mathcal{K}$, *i.e.* enough processing resources are still available, subframe i gets scheduled with probability

$$\varphi_i = \frac{1}{(\epsilon + \alpha + \beta)} \left(\epsilon + \alpha \, \frac{r_i}{T_{\max}} + \beta \, \frac{b_i}{B} \right). \tag{7}$$

If subframe i is scheduled, $\mathcal{T}_\mathcal{K}$ is diminished by t_i and subframe i is removed from $\mathcal{S}$. If i is not scheduled the available computing resources stay unchanged and the subframe is not removed from $\mathcal{S}$. The algorithm repeats rounds until the remaining resources in $\mathcal{T}_\mathcal{K}$ are not sufficient to process the fastest element still in $\mathcal{S}$, or until no subframe has been scheduled in the last L consecutive samplings. The expected number $\overline{\nu}$ of rounds to complete the algorithm is upper bounded by

$$\overline{\nu} \leq \frac{\mathcal{T}_\mathcal{K}}{\min_i\{t_i\}} \frac{1}{\min_i\{\varphi_i\}} + L. \tag{8}$$

The computational cost of the probabilistic algorithm can be hence approximated by the cost of performing $\overline{\nu}$ random samplings.

The parameters $\epsilon \in \{0, 1\}$, $\alpha \in \{0, 1\}$ and $\beta \in \{0, 1\}$ in (7) play the same role as in the ILP objective function (6). For $(\epsilon, \alpha, \beta) = (1, 0, 0)$ the probabilistic algorithm samples subframes from $\mathcal{S}$ and allocates them with probability 1 until the computing resources are occupied. For $(\epsilon, \alpha, \beta) = (0, 0, 1)$ the higher is the information content of the sampled subframe, the higher is the probability that it gets scheduled. Similarly, for $(\epsilon, \alpha, \beta) = (0, 1, 0)$ the older is the HARQ transmission history of the sampled subframe, the higher is the probability that it gets scheduled. For $(\epsilon, \alpha, \beta) = (0, 1, 1)$ the probability of getting scheduled increases both with the information content and with the retransmission age.

The performance of the probabilistic algorithm is indicated by the dotted curves in Figures 1 and 2. The curves keep the same color code of their ILP counterparts. As expected, for the SRR the performance of the probabilistic algorithm for $(1, 0, 0)$ (yellow curves) is suboptimal with respect to the performance of the ILP algorithm; similarly, the ILP algorithm outperforms the probabilistic algorithm for $(0, 0, 1)$ (red curves) for the BRR. However we observe that in both cases the probabilistic algorithm performs better with respect to the BER observed at MAC layer, as visible in Figure 2. For the $(0, 1, 0)$ strategy, the ILP and probabilistic algorithms achieve comparable performance in terms of average delay and Goodput, even if the BER is mildly degraded for the probabilistic algorithm. Combining strategies, as in $(\epsilon, \alpha, \beta) = (0, 1, 1)$ does not provide, with the probabilistic algorithm, the same gain over the $(0, 1, 0)$ strategy as in the ILP formulation, but still remains the best option to minimize the BER.

The performance obtained with the probabilistic algorithm confirms as well that considering the retransmission age of the subframes in the allocation of the computation resources may improve significatively the performance of the system, especially in terms of accuracy.

A. Parameter optimization

To allow the comparison between the ILP-based and the probabilistic algorithms we have considered, so far, only binary values for the parameters $(\epsilon, \alpha, \beta)$. In this subsection we relax this requirement and we allow $(\epsilon, \alpha, \beta)$ to take real values in the interval $[0, 1]^3$, to investigate whether better performance trade-offs are possible with the probabilistic algorithm. Because of the normalizing factor in (7) the different strategy operated by the probabilistic algorithm depends on the relative magnitudes of the elements in $(\epsilon, \alpha, \beta)$ and not on their absolute values, so that restricting the dynamics to $[0, 1]$ is done without loss of optimality.

We focus on the case of $K = 1$ and $|\mathcal{N}| = 30$, and we simulate the behavior of the probabilistic algorithm for various configurations of the triple $(\epsilon, \alpha, \beta)$. Figure 3 represents the achievable BER at MAC layer, as a function of the corresponding BRR at the PHY layer. Each dot in the scatter plot corresponds to a configuration of $(\epsilon, \alpha, \beta)$. The plot shows that the previously considered cases roughly describe the boundaries of the regions of the achievable trade-offs. The dynamic of the BER axis in Figure 3 spans about 10 dB, which indicates that the choice of the parameters may greatly affect the accuracy of the algorithm. As already suggested by Figure 1, the strategy $(\epsilon, \alpha, \beta) = (0, 1, 1)$ minimizes the BER. From Figure 3 is evident that optimizing the accuracy at the PHY layer (red dot) leaves some potential untapped, and choosing a cross-layer strategy where also the HARQ retransmission age is considered allows to gain up to 5 dB.

By looking at Figure 3 the strategies $(\epsilon, \alpha, \beta) = (0, 1, 0)$ (blue dot) and $(\epsilon, \alpha, \beta) = (0, 1, 1)$ (purple dot) appear

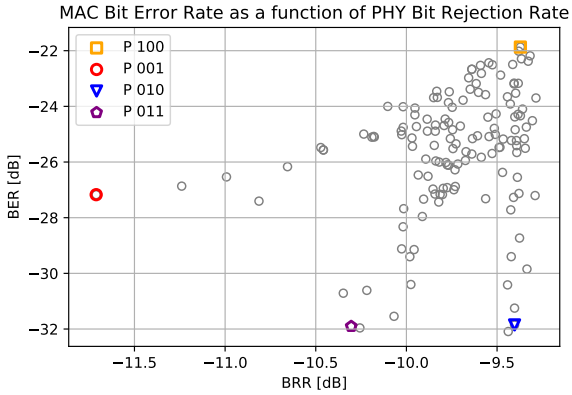

Fig. 3. Achievable BER at MAC layer as a function of the BRR at the PHY layer (bottom), varying the parameters of the probabilistic algorithm.

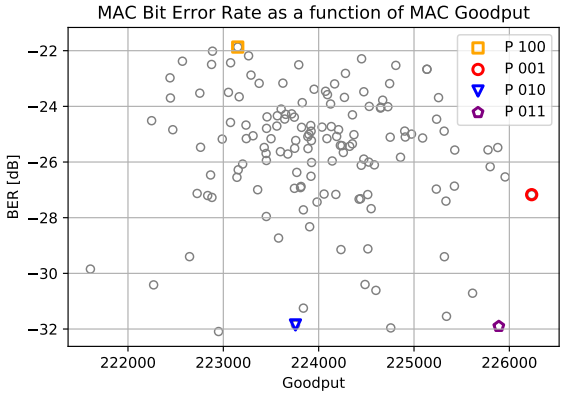

Fig. 4. Achievable BER at MAC layer as a function of the Goodput at PHY layer, varying the parameters of the probabilistic algorithm.

equivalent. Figure 4 shows the trade-off achievable between BER and Goodput, and shows that the cross-layer strategy $(\epsilon, \alpha, \beta) = (0, 1, 1)$ is able to minimize the BER while very nearly achieving the maximum Goodput. It is however worth noticing that the maximum improvement in Goodput achievable by varying the parameters is of about 2%. This confirms that the choice of the strategy is most critical with respect to accuracy. Finally, the position of the yellow dot in Figure 4 shows that the policy of limiting the subframes losses at the PHY layer, however a natural choice, achieves one of the least satisfying performance trade-off.

VI. Conclusion

We have considered computing resources allocation for the uplink in a C-RAN architecture. When the system is in overload, it is necessary to draw a policy to decide which subframes will not be processed. We have considered a system where the computation resources are allocated separately from the radio resources. Minimizing the data losses at PHY layer, although a very natural strategy, has been proven suboptimal when the performance of the system is gauged at the upper layer. A cross-layer strategy accounting also for the HARQ retransmission process at the upper layer has allowed to sensibly improve the performance, especially with respect to accuracy.

We have considered here a simplified setting where each subframe hosts data of one user. In real systems the same subframe may simultaneously host data of multiple users. The probabilistic algorithm can be adapted to this case, by extending the values that the tag r_i may take to account for the "efficient normalized age" of the subframe.

This study showed that the cross-layer strategy may improve the overall spectral efficiency. This suggest that tackling the scheduling of the radio and the computing resources jointly is a good direction for future work.

References

[1] Y. Lin, L. Shao, Z. Zhu, Q. Wang, and R. K. Sabhikhi, "Wireless network cloud: Architecture and system requirements," IBM Journal of Research and Development, January-February 2010.

[2] X. Wang, "C-RAN: The Road Towards Green RAN," China Communications Journal, June 2010.

[3] A. Checko, H. L. Christiansen, Y. Yan, L. Scolari, G. Kardaras, and M. S. Berger, L. Dittmann, "Cloud RAN for Mobile Networks — A Technology Overview," IEEE Communications Survey & Tutorials 17, no. 1, pp. 405 – 426, 2015.

[4] N. Nikaein, "Processing radio access network functions in the cloud: Critical issues and modeling," Proc. 6th Int. Workshop on Mobile Cloud Computing and Services, pp. 36–43, September 2015.

[5] I. A. Alimi, A. L. Teixeira, and P. Pereira Monteiro. "Toward an efficient C-RAN optical fronthaul for the future networks: a tutorial on technologies, requirements, challenges, and solutions," IEEE Communications Surveys & Tutorials 20, no. 1, pp. 708-769, 2018.

[6] M. Peng, C. Wang, V. Lau, and H. V. Poor, "Fronthaul-constrained cloud radio access networks: insights and challenges," IEEE Wireless Communications, vol. 22, pp. 152160, April 2015.

[7] M. Qian, H. Wibowo, S. Jinglin, and V. Branka, "Baseband processing units virtualization for cloud radio access networks," IEEE Wireless Communications Letters 4, no. 2, pp. 189–192, 2015.

[8] D. Pompili, H. Abolfazl, and X. T. Tuyen, "Elastic resource utilization framework for high capacity and energy efficiency in cloud RAN," IEEE Communications Magazine 54, no. 1, pp. 26–32, 2016.

[9] K. C. Garikipati, F. Kassem, and G. S. Kang, "RT-OPEX: Flexible scheduling for Cloud-RAN processing," In Proceedings of the 12th International on Conference on emerging Networking Experiments and Technologies, pp. 267 –280, 2016.

[10] V. Q. Rodriguez, and F. Guillemin, "Cloud-RAN modeling based on parallel processing," IEEE Journal on Selected Areas in Communications 36, no. 3, pp. 457–468, 2018.

[11] S. Matoussi, I. Fajjari, S. Costanzo, N. Aitsaadi, and R. Langar, "A User Centric Virtual Network Function Orchestration for Agile 5G Cloud-RAN," IEEE International Conference on Communications (ICC), pp. 1–7, 2018.

[12] H. Khedher, S. Hoteit, P. Brown, R. Krishnaswamy, W. Diego, and V. Veque, "Processing time evaluation and prediction in Cloud-RAN," Proc. IEEE Int. Conf. on Communications (ICC), May 2019.

[13] P. Wu, and N. Jindal, "Coding versus ARQ in fading channels: How reliable should the PHY be?," IEEE Transactions on Communications 59, no. 12, pp. 3363-3374, 2011.

[14] H. Khedher, S. Hoteit, P. Brown, V. Vèque, R. Krishnaswamy, W. Diego, and M. Hadji, "Real Traffic-Aware Scheduling of Computing Resources in Cloud-RAN," International Conference on Computing, Networking and Communications (ICNC), Big Island, Hawaii, 2020.

[15] A. Larmo A, M. Lindstrom, M. Meyer, G. Pelletier, J. Torsner, and H. Wiemann, "The LTE link-layer design," IEEE Communications magazine 47, no.4, pp. 52–59, May 2009.

[16] H. D. Trinh, N. Bui, J. Widmer, L. Giupponi, and P. Dini, "Analysis and modeling of mobile traffic using real traces," in Proc. IEEE PIMRC, 2017.