A realistic coordinated scheduling scheme for the next-generation RAN

Alberto Martínez Alba, Arsany Basta, Jorge Humberto Gómez Velásquez and Wolfgang Kellerer
Lehrstuhl für Kommunikationsnetze
Technical University of Munich
Email: {alberto.martinez-alba, arsany.basta, jorge.gomez, wolfgang.kellerer}@tum.de

***Abstract*—The design of the 5G next-generation radio access network (NG-RAN) proposes the division of the next-generation eNodeB (gNB) into centralized and distributed units. Centralization should facilitate coordination of RAN functions between gNBs, but the actual benefits of it are still unclear. In this paper, we provide a study of the feasibility and benefits of coordinated downlink scheduling. We first analyze the time constraints that a coordinated scheduler has to face from a theoretical perspective, and we back them with an experimental proof-of-concept. Then, we present a lightweight scheme for coordinated link adaptation that matches the previous constraints. We show that coordination is indeed feasible with state-of-the-art technology, although very limited by time constraints. Finally, we show the results of our experimental testbed, which successfully implemented the described coordination scheme under the predicted constraints.**

I. Introduction

The fifth generation of mobile networks aims at ambitious goals, such as providing ultra-reliable low-latency communications, enhanced broadband connections, and massive machine-type deployments. In order to achieve these goals, numerous improvements to the current LTE networks have been proposed, ranging from the physical layer to the system architecture. One major improvement is the redesign of the radio access network (RAN), which is in charge of establishing and managing the wireless connection between the user equipment (UE) and the mobile core network. In LTE, the RAN consists of a single entity, the eNodeB (eNB), which implements both control and data functions from the physical to the network layers and is usually deployed close to the antenna. For the Next-Generation RAN (NG-RAN) of 5G, the 3GPP has proposed to discontinue that distributed, monolithic architecture. Instead, it is suggested to centralize a subset of the RAN functions into edge clouds.

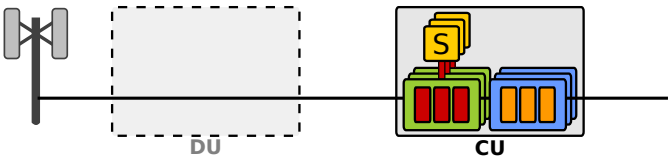
There are two main reasons for this decision. On the one hand, the centralization of functions would reduce the need of expensive, dedicated equipment on remote sites. Moreover, these functions can be virtualized and deployed into generic data centers, thus reducing resource usage. Overall, the cost of a centralized RAN would be substantially reduced with respect to a distributed one, like that of LTE. On the other hand, the centralization of RAN functions enables new techniques to be exploited. More specifically, centralized functions may exploit their proximity to coordinate with one another.

The idea of centralizing RAN functions in 5G is the consequence of previous research work. The greatest exponent of the centralization of RAN functions is the Cloud-RAN architecture [1]. In Cloud-RAN, all the RAN functions are virtualized and located in general-purpose data centers. Given the advantages of a centralized RAN, the idea of Cloud-RAN has been abundantly developed by the research community. Nonetheless, the feasibility of a totally centralized RAN has been put into question, owing to the need of a high-throughput, low-latency fronthaul network to realize such an architecture. In order to overcome this problem, a partially centralized approach has been suggested instead [2].

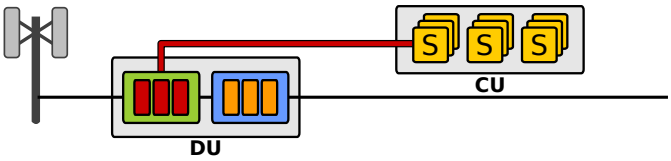
The NG-RAN architecture proposed by 3GPP builds upon this new idea of a partially centralized RAN. Indeed, the next-generation eNodeB (gNB), will be divided into two units: a distributed unit (DU), which is located close to the antenna, and a centralized unit (CU), which is located at a data center alongside other CUs [3]. However, a partially centralized RAN architecture raises major questions as well. For instance, the cost reductions of a partially centralized RAN are not clear anymore, since there is still the need of deploying equipment at the remote sites. As a consequence, the sole argument of the cost is not enough to support the partial centralization of the RAN. A close look into the feasibility and benefits of function coordination in the NG-RAN is thus needed.

Coordination has been targeted since the beginning of cellular communications, owing to the interference-prone nature of these networks. In LTE, for example, several schemes of inter-cell interference coordination (ICIC) and coordinated multi-point (CoMP) were proposed. The possibilities of realizing them in LTE was limited, but the centralization coming from the new NG-RAN architectures provides us with new opportunities, as recent research work is showing [4], [5]. However, it is still challenging to design a realistic coordination scheme that takes account of the network limitations.

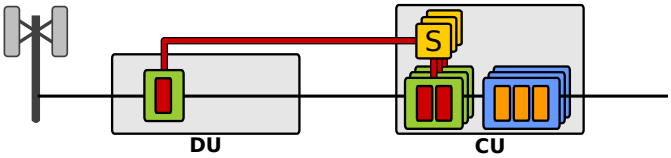
In this paper, we provide a study of the feasibility and benefits of function coordination in the NG-RAN with a focus on coordinated downlink scheduling. We first lay out the different architecture options for the NG-RAN, and explain how they influence the possibilities of coordination. Then, we present a theoretical and empirical analysis about the time limitations that coordinated schedulers have to face. Next, we use this information to design a simple, resource-efficient scheme for coordinated link adaptation. Finally, we present simulation and implementation results to show the effectiveness of the proposed scheme.



(a) Cloud-RAN architecture.



(b) Pure SD-RAN architecture.



(c) Example of a partially centralized architecture (also SD-RAN).

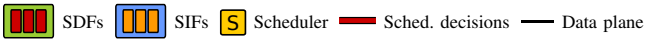


Fig. 1. Possible architectures for the NG-RAN, according to how they may affect coordination between schedulers. Three kinds of functions are represented: scheduling-dependent functions (SDFs), scheduling-independent functions (SIFs), and the schedulers.

The rest of the paper is structured as follows. Sec. II explains the different architecture options for the NG-RAN. In Sec. III, the details of the system model and overall assumptions are presented. Sec. IV contains an analysis of the time constraints for coordinated scheduling. In Sec. V, a simple scheme for coordinated link adaptation is proposed. Finally, in Sec. VI we provide evaluation results, and the paper is concluded in Sec. VII.

II. NG-RAN ARCHITECTURE

As with other communication networks, the mobile RAN can be described in terms of its network *functions*. In the NG-RAN, functions can either be centralized (at the CU) or distributed (at the DU) [3]. Centralization promises cost reductions and performance improvements by means of function coordination, but it also poses important challenges to the fronthaul network [1]. In order to relax the requirements of the fronthaul, one could centralize only those functions benefiting the most from coordination, leaving the others distributed.

The scheduler is one of the functions that can benefit the most from centralization, as coordinated scheduling is useful to mitigate inter-cell interference. Nonetheless, the effectiveness of coordinated scheduling also depends on the location of other functions. In fact, we can classify the RAN functions into three types: (i) scheduler-dependent functions (SDFs), including PHY, MAC, and low RLC layers; (ii) scheduler-independent functions (SIFs), including IP, RRC, PDCP, and high RLC layers; and (iii) the scheduler itself. The feasibility and earnings of coordination will depend on the relative distribution of the SDFs and the scheduler, whereas the location of the SIFs is, in principle, irrelevant.

In order to enable coordination at all, the scheduler has to be centralized. Therefore, the architecture options are defined by the level of centralization of the SDFs. If all the SDFs are centralized (implying also that all SIFs are centralized), we get a Cloud-RAN architecture, depicted in Fig. 1 (a). This is the best possible scenario for coordinated schedulers, since the schedulers are able to communicate among themselves and with SDFs through high-capacity, low-latency links.

Conversely, if all the SDFs are distributed, we get the scenario shown in Fig. 1 (b). This option allows for fast communication among schedulers, but the slower communication with SDFs may reduce the coordination possibilities. This architecture is called SD-RAN, since it resembles the idea of software-defined networking (SDN), as the scheduler belongs to the control plane [6], [7].

Finally, we could face the case in which not all of the SDFs are centralized, as in Fig. 1 (c). Although the performance of the RAN may be different in other aspects, regarding coordination it would behave the same as in the previous SD-RAN architecture. The reason is simple: coordination is limited by the latency to the farthest SDF, as all the SDFs need to face the same time constraints. Therefore, a partially centralized architecture will be also considered as SD-RAN. As a result, we have only two essentially different architectures: Cloud-RAN and SD-RAN.

Regarding the internal architecture of the scheduler, there are two options: a logically centralized scheduler, or distributed but physically co-located schedulers. A logically centralized scheduler is a single function that receives information from all the UEs in the network to compute the best allocation of resources. The alternative is to have one scheduler per cell, which performs local decisions based on information from its own UEs, and afterwards it exchanges information with neighbor schedulers. In this work, we will consider that the schedulers are distributed. The possibility of having a centralized scheduler is left for future work.

III. SYSTEM MODEL

A. General network description

We consider a frequency-division duplex (FDD) network with M gNBs, each one in charge of one cell. The transmitters of these cells are located close together, thus avoiding power gaps at the cell edges. In principle, this means that high downlink throughput can be achieved. However, the proximity of the transmitters implies that UEs at the cell edges may receive downlink interference from neighbor cells, hence preventing the achievement of high throughputs. We will refer to those neighbor gNBs as *interfering gNBs*, whereas the term *serving gNB* will denote the gNB on which the selected UE is camping. We assume that cells are synchronized, and that, in general, cells are not fully loaded.

B. Requirements from the serving gNB

Any form of coordinated downlink transmission is only applicable when the identities of the interfering gNBs of every UE are known to the serving gNB. The serving gNB needs this information to know whether the decisions of other gNBs affect its UEs. There are two possible ways in which this information can be acquired: either the UEs explicitly send messages informing about the interferers, or the serving gNB deduces them from the location of the UEs. Either way, we will assume that the interfering gNBs of every UE are known to the serving gNB.

Apart from the identity of the interferers, information about state of the channel for each UE is needed in order to perform the scheduling. This is obtained via the channel quality information (CQI) reports that the UE sends to the gNB, both periodically and on demand.

C. Coordination network

We assume that all CUs belonging to the same NG-RAN and located in the same data center are connected via a virtual network. The throughput of this virtual network should be at least 10 Gbps, since high-speed link technologies can be used within the data center [8].

With the purpose of coordinating the downlink scheduling, performing handovers, and other tasks, an inter-gNB communication protocol must be in place. In 5G, this protocol is the Xn Application Protocol (Xn-AP). For messages exchanged between coordinated schedulers, we propose message similar to the X2 Load Indication of LTE. This message reports a bit for every resource block (RB) in the cell, indicating whether it is going to be used for transmission in the next subframe.

IV. Time analysis

The scheduler in a 5G gNB produces a scheduling decision every subframe, that is, every millisecond. This time is comparable to the time it takes to process and exchange information even in the best-performing data centers. Therefore, any form of coordination between centralized schedulers needs to take into account the timing of each step. In this section, we analyze the time constraints of a pair of coordinated schedulers S_1 and S_2, as depicted in Fig. 2. We will use the insights gained from this analysis to design a coordinated scheduling scheme.

A. Coordination timeline

The scheduling process lasts in total t_S, and can be divided into four stages. The first stage is the initial processing phase, when the scheduler uses information from the upper and lower layers (such as buffer status and CQI reports) to produce a local scheduling decision. This phase lasts t_{p0}^m for the scheduler S_m, $m \in 1, 2$. The next stage is the exchange phase, when the scheduling decision is transmitted to the neighbor gNB and, at the same time, the corresponding decision of the neighbor gNB is received. As it is shown in Fig. 2, the duration of this phase depends on the previous processing time and the transmission time of each packet t_e, which is the same for all

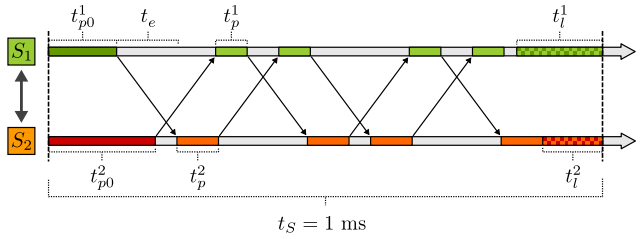


Fig. 2. Timeline of the scheduling process for $K = 4$ iterations and two schedulers. The initial processing time, t_{p0}^m, is in dark green or red; the exchange time, in gray; the subsequent processing times, t_p^m, in light green or red; and the submission time, t_l^m, in a checkered pattern. The black arrows represent the transmission of temporary scheduling decisions.

the schedulers in a symmetrical network. The third stage is a new processing phase, but this time the scheduler takes into account the decisions of neighbor gNBs to produce a new scheduling decision. This phase can have a duration different from that of the initial one, lasting t_p^m. The second and third stages can be repeated K times, before the fourth stage, which is the submission of the final scheduling decision to the SDFs. The duration of this submission is represented by t_l^m.

The total time required for the coordinated scheduling of S_1 can be derived from Fig. 2:

$$t_S^1 = \begin{cases} t_{p0}^1 + Kt_e + \frac{K}{2}(t_p^1 + t_p^2) + t_l^1 & \text{if } K \text{ even,} \\ t_{p0}^2 + Kt_e + \frac{K+1}{2}t_p^1 + \frac{K-1}{2}t_p^2 + t_l^2 & \text{if } K \text{ odd.} \end{cases} \quad (1)$$

If we assume, for the sake of simplicity, that the two schedulers are identical ($S_1 \equiv S_2$), (1) simplifies into:

$$t_S = t_{p0} + K \cdot (t_e + t_p) + t_l. \quad (2)$$

Equation (2) is a generic expression for t_S, but a more accurate expression can be derived if we take into account the RAN architecture. Indeed, the value of t_l is directly related to the architecture of the RAN. In SD-RAN, t_l reflects the transmission delay between the CU and the DU, as the scheduler and one or more SDFs are physically separated. In contrast, all the SDFs are centralized in Cloud-RAN. Therefore, the communication between the scheduler and the other layers takes the same time as the communication between schedulers, that is, $t_l = t_e$. If we apply that into (1), we get:

$$t_S = \begin{cases} t_{p0} + K \cdot (t_e + t_p) + t_l & \text{for SD-RAN,} \\ t_{p0} + (K+1) \cdot t_e + Kt_p & \text{for Cloud-RAN.} \end{cases} \quad (3)$$

Finally, we face the following constraint for any architecture:

$$t_S \leq 1 \text{ ms.} \quad (4)$$

If (4) is satisfied for $K > 0$, some kind of coordinated scheduling is possible in the network. Otherwise, coordination is not possible, regardless of the architecture.

B. Time estimations and measurements

In order to provide realistic estimations of t_{p0}, t_p, t_e, and t_l, we developed a 5G coordination testbed based on OpenAirInterface and OpenStack, whose architecture is depicted in Fig. 3. It consists of a serving gNB, an interfering gNB, and a

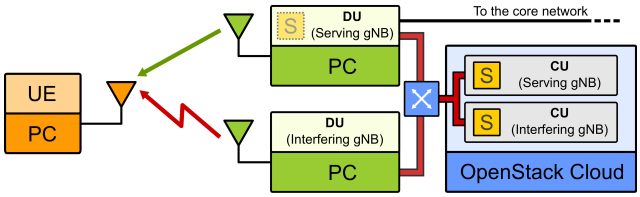


Fig. 3. Architecture of the 5G coordination testbed. It consists of a serving gNB (implemented with OpenAirInterface and FlexRAN), an interfering gNB, and a CU-hosting cloud.

UE. Each gNB is divided into a DU and a CU. For the serving gNB, the DU is implemented with OpenAirInterface and a FlexRAN agent [7], whereas the CU is a FlexRAN controller. FlexRAN enables to dynamically move the serving scheduler from the CU to the DU and vice versa. As a result, the architecture of the testbed can switch between SD-RAN and Cloud-RAN. For the interfering gNB, the DU is a gnuradio script designed to mimic the interference produced by a gNB, whereas the CU is a C program that schedules the interference and sends information to the serving's DU.

All the CUs are implemented as virtual machines (VMs) hosted in an OpenStack-managed cloud, consisting of five servers with 48-core Intel Xeon E5-2650. The DUs are deployed on stand-alone Intel i7-6700 PCs. They are connected by an OpenFlow switch and 10 GbE links, which is in charge of transporting the control plane.

The processing times t_{p0} and t_p depend on the scheduling algorithm and the computing platform; therefore, their value can change from implementation to implementation. For our estimation, we measured the time that takes for the scheduler of OpenAirInterface to allocate 25 RBs and set the modulation and coding scheme (MCS) for one UE. For more UEs or RBs, this time would scale linearly in a proportional-fair scheduling algorithm, as it spends the same amount of time computing the priority of each UE for each RB. The measurements were repeated 1000 times under full load to ensure confidence. As it is shown in Fig. 4, they mostly range from 0.05 to 0.12 ms.

Regarding the exchange time t_e, it can be measured as half the round-trip time (RTT) between schedulers, that is, the RTT between VMs in the cloud. We measured the RTT of UDP packets between the serving and the interfering CU, and the results turned out to be mostly between 0.12 ms and 0.22 ms after 1000 repetitions. Thus, the values of t_e range between 0.06 and 0.11 ms, which are those depicted in Fig. 4.

As mentioned previously, the value of t_l may only be substantially different from t_e if we assume an SD-RAN architecture. For that case, t_l can be computed as half the RTT between the VM hosting the CU and the PC containing the DU. Our measurements showed that 95% of the t_l samples were between 0.26 and 0.45 ms after 1000 repetitions, although a maximum delay of 0.25 ms is recommended according to [9].

C. Maximum number of iterations

After the measurements, we have actual values to plug in (3). This will allow us to derive an estimate of the number

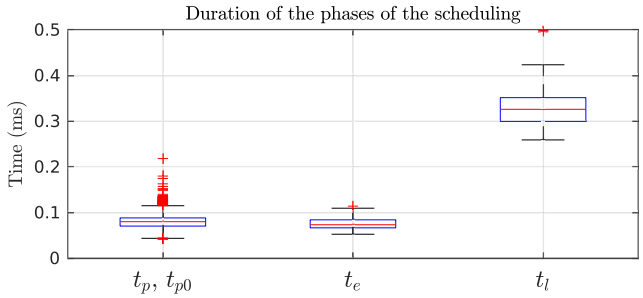


Fig. 4. Durations of the four stages of the scheduling, measured from the 5G coordination testbed.

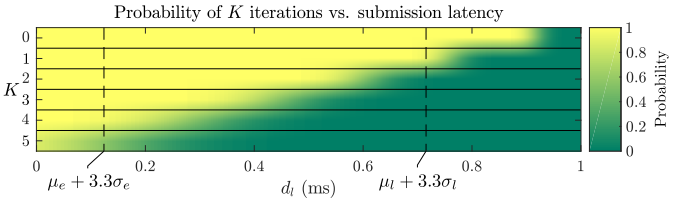


Fig. 5. Probability of accomplishing K iterations as a function of the allocated submission time d_l.

of coordination iterations K, that is, the number of messages that can be exchanged by the schedulers in 1 ms. This number is specially limited by t_l, the delay between CU and DU, as some time $d_l > t_l$ must be reserved at the end of the scheduling process to ensure that it is done on time. That is, when time reaches $t = 1 - d_l$ the coordination process must stop and the latest scheduling decision is submitted to the SDFs. The time d_l may be defined according to some reliability goal ρ, such that $\Pr\{t_l > d_l\} \leq \rho$.

Given this hard deadline, we need to find the probability of having time for K iterations. For simplicity, let us assume that times t_{p0}, t_e, and t_p are normally-distributed independent random variables with means μ_{p0}, μ_e, μ_p, and variances σ^2_{p0}, σ^2_e, σ^2_p, respectively. We define:

$$t_{pe} = t_{p0} + K(t_e + t_p). \tag{5}$$

Then, t_{pe} is also normally distributed with mean μ_{pe} and variance σ^2_{pe}:

$$\mu_{pe} = \mu_{p0} + K(\mu_e + \mu_p), \quad \sigma^2_{pe} = \sigma^2_{p0} + K^2(\sigma^2_e + \sigma^2_p). \tag{6}$$

Now, we can use t_{pe} to calculate the probability of not surpassing the deadline with K iterations:

$$\Pr\{t_{pe} \leq 1 - d_l \mid K\} = \frac{1 + \operatorname{erf}\left(\frac{1-d_l-\mu_{pe}}{\sigma_{pe}\sqrt{2}}\right)}{2}. \tag{7}$$

Fig. 5 shows values of (7) for $K \in \{1, ..., 5\}$ and $d_l \in [0, 1]$ ms. The discontinuous lines show two values of d_l that are 3.3 standard deviations away from the mean of t_e and t_l. This covers 99.9% of the values of t_l for Cloud-RAN and SD-RAN, respectively. We can conclude that, in order not to surpass the time allocated for the scheduling, $K \leq 1$ for SD-RAN and $K \leq 4$ for Cloud-RAN.

V. Coordination scheme

From the analysis of last section, we can draw two conclusions. First, coordination is indeed possible in the NG-RAN, assuming state-of-the-art technology. Second, the amount of information that can be exchanged between gNBs is very limited, not exceeding one iteration for SD-RAN and four iterations for Cloud-RAN. Since the architecture of NG-RAN is likely to be partially centralized, the limitation of SD-RAN is actually more realistic. Under these conditions, the challenge is to find a suitable coordinated scheduling technique.

As mentioned in Sec. III-C, the gNBs exchange messages indicating whether they intend to transmit in a certain RB in the next subframe in every iteration. In response to these messages, the gNBs could try to cooperatively allocate RBs and assign the appropriate MCS, or just assign the MCS. We refer to the former as coordinated resource allocation (CRA), and to the latter as coordinated link adaptation (CLA). CRA requires solving the allocation conflicts that may occur among schedulers. This could take much more than one iteration, so it would be possible only in Cloud-RAN or in high-performing RAN deployments. CLA, however, requires only one iteration, which is exactly what we can afford. Therefore, in order to complete our path towards a realistic coordination scheme in 5G, in this section we propose a lightweight CLA scheme.

A. Algorithm for coordinated link adaptation

The objective of CLA is the assignment of the appropriate MCS according to the presence of interference. In a nutshell, the scheduler has to predict the channel state with and without interference in order for the correct MCS to be selected. In LTE Rel.10, a CLA scheme was introduced, called CSI Interference Management (CSI-IM) [10], which proposes for the UE to measure and inform the gNB about the channel state in the two cases. To enable this, all the gNBs in the network are configured to transmit reference signals at some resource blocks and remain quiet at others. In this way, all possible interference combinations can be detected by the UE.

In principle, we might adopt CSI-IM as our coordination scheme. However, we see some problems in it that could counter the earnings of coordination. Specifically, the resources used by CSI-IM may grow too large for more than two gNBs. In addition, it forces the UE to measure and report the channel state twice, as if the gNB knew nothing about the conditions on which the measurements were performed. However, the gNB can actually configure when the UE measures the channel and, if it is coordinated with other gNBs, it also knows whether the UE received interference during such measurements. Therefore, the gNB knows precisely whether a given CQI report reflects interference or not. As a consequence, the UE could transmit just a single CQI report, which would be then used directly by the gNB (if the interference situation repeats at the time of transmission), or corrected (if the situation changes). This is the intuition behind our proposed CLA scheme.

In order to decide if a CQI correction is needed, the scheduler in the serving gNB has to find out whether the reported CQI matches the state of the channel for the next transmission. In other words, it needs to know if an interfering gNB transmitted in the subframe when the CQI was measured, and if an interfering gNB will transmit in the next subframe. To that end, the serving gNB should keep a record of past scheduling decisions of interfering gNBs. By consulting the record, the serving gNB will learn whether the reported CQI reflects an interference situation or not.

B. Channel quality correction

The CQI reported by the UE is just an index directly proportional to the measured SINR, as shown in [11]. Thus, correcting the CQI is equivalent to converting the CQI into SINR, correcting the SINR, and then converting the SINR back into CQI. In order to simplify the subsequent analysis, we consider that UEs receive interference mainly from a single neighbor gNB, while other neighbor gNBs do not interfere noticeably. For a generic resource block in subframe τ, we define the random variables Γ_I and Γ_N, representing the SINR experienced by the UE in the case of receiving and not receiving interference, respectively:

$$\Gamma_I = \frac{H_S \cdot p_S}{\eta + H_I \cdot p_I}, \qquad \Gamma_N = \frac{H_S \cdot p_S}{\eta}, \tag{8}$$

where p_S and p_I are the transmission powers of the serving and interfering gNBs, respectively; H_S and H_I are the channel power responses of the serving and interfering signals, respectively; and η is the received noise power.

If the measured SINR needs to be corrected, there are two possible cases. The UE may have measured the SINR while not being interfered, whereas at the time of transmission there is going to be interference, and vice versa. We will henceforth refer to the former case as Scenario IM (Interference Measured), and the latter as Scenario NIM (No Interference Measured). In Scenario NIM, we have $\Gamma_N = \gamma_N$ and we need an estimate of Γ_I. From (8) it follows that:

$$\Gamma_I = \frac{\eta}{\eta + H_I \cdot p_I}\gamma_N. \tag{9}$$

The channel H_I is usually modeled as a combination of slow fading m_I and fast fading R_I [12]:

$$H_I = m_I \cdot R_I. \tag{10}$$

We assume that slow fading remains constant every subframe, hence the only random variable is R_I. The value of m_I can be calculated from parameters readily available to the serving gNB by using a radio propagation model. Regarding fast fading, Rayleigh fading is commonly assumed in mobile communications. Therefore, R_I denotes the power coefficient of a Rayleigh fading channel. This implies that $R_I \sim Exp(2\sigma^2)$, since it is the square of the amplitude of the channel response, which follows a Rayleigh distribution. The value of σ^2 depends on the selected channel model, but it is usual to assume that $\mathrm{E}\{R_I\} = 1$, which means that the average gain of the channel is just $\mathrm{E}\{H_I\} = m_I$ [12]. For us,

TABLE I
SIMULATION PARAMETERS

Parameter	Value
SNR	10 dB
Delay profile	EVA
CQI reporting mode	Wideband CQI
Doppler frequency	5 Hz

that implies $\sigma^2 = \frac{1}{2}$ and, therefore, a very simple cumulative distribution function of R_I:

$$F_{R_I}(r_I) = 1 - e^{-r_I}. \tag{11}$$

From this equation, we can derive the cumulative distribution function of Γ_I, given $\Gamma_N = \gamma_N$:

$$\begin{aligned} F_{\Gamma_I}(\gamma_I|\Gamma_N = \gamma_N) &= 1 - F_{R_I}\left(\frac{n(\gamma_N - \gamma_I)}{\gamma_I m_I p_I}\right) \\ &= e^{-\frac{n(\gamma_N-\gamma_I)}{\gamma_I m_I p_I}}, \end{aligned} \tag{12}$$

and also the probability density function:

$$f_{\Gamma_I}(\gamma_I|\Gamma_N = \gamma_N) = \frac{n\gamma_N}{\gamma_I^2 m_I p_I} e^{-\frac{n(\gamma_N-\gamma_I)}{\gamma_I m_I p_I}}. \tag{13}$$

Once we have $f_{\Gamma_I}(\gamma_I|\Gamma_N = \gamma_N)$ we can find the mode of Γ_I, that is, the most likely value of Γ_I given $\Gamma_N = \gamma_N$, and use it as our estimator $\hat{\gamma}_I$. It is easy to show that the mode of (13) is:

$$\hat{\gamma}_I = \frac{\eta}{m_I p_I}\gamma_N. \tag{14}$$

For Scenario IM, an equivalent process leads to:

$$\hat{\gamma}_N = \frac{m_I p_I}{\eta}\gamma_I. \tag{15}$$

By applying (14) and (15), the SINR reported by the UE can be corrected at the time of transmission. As we have seen already, the serving gNB will know if such correction is necessary after coordinating with other gNBs.

VI. EVALUATION

The feasibility and earnings of the CLA scheme above were tested by means of simulations and physical implementations. The objective of the simulations was to assess the fitness of the estimators, whereas the implementation confirmed that it was indeed realizable and showed us the actual impact of CLA in the downlink throughput.

A. Simulation results

A MATLAB simulator was developed with the goal of testing the accuracy of the estimators. The LTE Toolbox was used to generate an end-to-end RAN as complete as possible, which included actual resource allocation and modulations as they are performed in 4G and 5G. Moreover, a Rayleigh fading channel and a signal from an interfering gNB were simulated. The SINR and CQI were computed by using the built-in functions available in the LTE Toolbox. The most relevant parameters of the simulation are shown in Table I. The two scenarios of CQI correction were simulated at the same time. For both of them, the actual SINR was computed and averaged over 200 subframes, along with the corrected SINR after applying (14) and (15). In Fig. 6 we can see the behavior of the corrected SINR as we increase the signal-to-interference ratio (SIR). We conclude that the corrected SINR closely resemble the actual SINR values, confirming the validity of the proposed scheme.

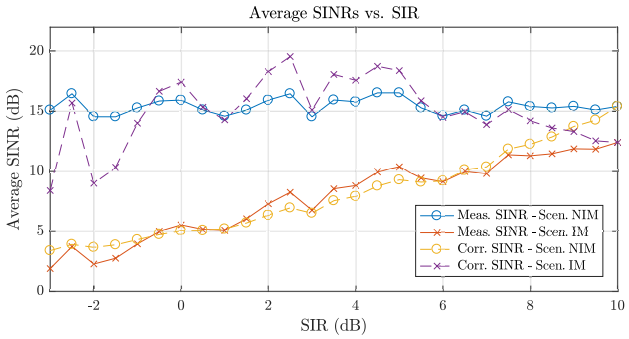


Fig. 6. Comparison between measured and corrected SINRs for Scenarios IM and NIM and different interference levels.

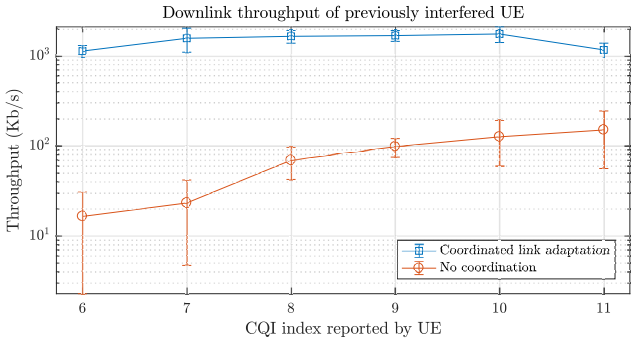


Fig. 7. Downlink throughput of a UE in Scenario IM, for coordinated and uncoordinated gNBs and different interference levels.

B. Implementation results

In Sec. IV-B we introduced our testbed, which is depicted in Fig. 3. This testbed is an end-to-end implementation of a coordinated NG-RAN, in which the time constraints of Sec. IV and the proposed CLA scheme of Sec. V were tested.

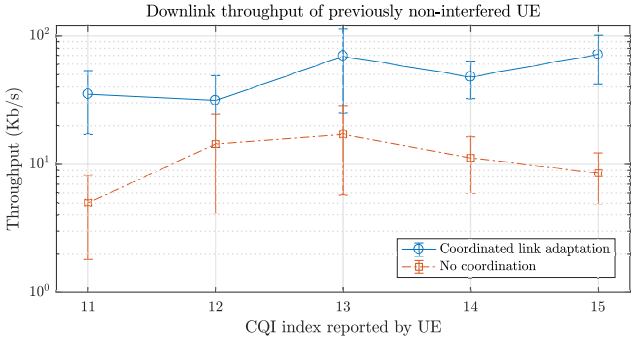


Fig. 8. Downlink throughput of a UE in Scenario NIM, for coordinated and uncoordinated gNBs and different interference levels.

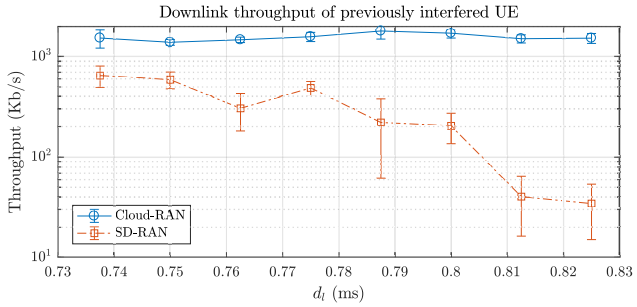


Fig. 9. Downlink throughput of a UE in Scenario IM, for Cloud-RAN and SD-RAN and several values of d_l, the time reserved for the communication between CU and DU.

The most notable innovation of our testbed is the connection between the schedulers of the serving and interfering gNBs, which allows for the serving gNB to be informed about the presence of interference in each RB every millisecond

In our first experiment, we emulated Scenario IM. In order to reproduce it, the schedulers of the interfering and serving gNB followed the same transmission pattern. In this way, the UE always measured the channel when interference was present, but downlink transmissions were interference-free. The transmission power of the interfering gNB was configured for the UE to experience an SINR between 10 and 20 dB at the subframes configured for CQI measurement, which resulted in CQIs between 6 and 11, as it can be seen in Fig. 7. We measured the throughput achieved when downloading a file from a remote HTTP server for 20 seconds, and the measurements were repeated 10 times for each CQI value. The results confirm that our scheme is realizable and that it improves the RAN performance at the cell edge, achieving between 5 and 50 times more throughput.

Our second experiment was Scenario NIM. The reported CQI was artificially reduced in order to measure a value different from 15, which is the maximum. In addition, the transmission power of the interfering gNB was set to correspond to a CQI of 10. The results are shown in Fig. 8, where we see that our CLA scheme led to a tenfold increase of the downlink throughput.

Finally, we repeated Scenario IM with one modification. We wanted to test the performance of our CLA scheme for our two considered architectures, Cloud-RAN and SD-RAN, when the latency between CU and DU is high. We emulated an increase of d_l, as explained in Sec. IV-C, by decreasing the time window allocated to receive the interference message from the interfering gNB. Additionally, we repeated the measurements after moving the scheduler to the CU. This corresponds to a legacy LTE architecture, but it is essentially the same architecture as Cloud-RAN from the coordination perspective. The result, shown in Fig. 9, matched very closely what was predicted in Fig. 5. At around $d_l = 0.75$ ms, the probability of accomplishing $K = 1$ iterations starts to reduce quickly, and so does the throughput. This shows that the earnings of coordination in an SD-RAN architecture can be very sensitive to the latency between CU and DU.

VII. Conclusion

In this paper, we investigate the feasibility of implementing a simple type of coordinated scheduling in the NG-RAN. We first classify the different architecture options for the NG-RAN according to their impact on the coordination aspect. Then, we provide an analysis on the latency constraints that coordinated schedulers have to face. We use experimental data from a state-of-the-art deployment to estimate the number of times that a scheduler can send a coordination message to a neighbor in one subframe. We conclude that between one and four iterations are possible, depending on the level of centralization of the NG-RAN. With this limitation in mind, we propose a simple CLA scheme to leverage the possibility to coordinate. Finally, we implement our proposed coordination scheme in a NG-RAN testbed, which confirms that coordinated scheduling is indeed possible and allows for substantial throughput increases in interfered UEs.

Acknowledgment

This work is part of a project that has received funding from the European Research Council (ERC) under the European Unions Horizon 2020 research and innovation program (grant agreement No 647158 - FlexNets “Quantifying Flexibility for Communication Networks”).

References

[1] 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, 2015.

[2] D. Sabella, P. Rost, Y. Sheng, E. Pateromichelakis, U. Salim, P. Guitton-Ouhamou, M. Di Girolamo, and G. Giuliani, “Ran as a service: Challenges of designing a flexible ran architecture in a cloud-based heterogeneous mobile network,” in *Future Network and Mobile Summit (FutureNetworkSummit), 2013*. IEEE, 2013, pp. 1–8.

[3] 3GPP, “NG-RAN; Architecture description,” 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 38.401. [Online]. Available: http://www.3gpp.org/DynaReport/38401.htm

[4] B. Soret, A. De Domenico, S. Bazzi, N. H. Mahmood, and K. I. Pedersen, “Interference coordination for 5g new radio,” *IEEE Wireless Communications*, 2017.

[5] G. Nardini, G. Stea, A. Virdis, A. Frangioni, L. Galli, D. Sabella, and G. Dell’Aera, “Scalability and energy efficiency of coordinated scheduling in cellular networks towards 5g,” in *Cloud Technologies and Energy Efficiency in Mobile Communication Networks (CLEEN), 2017 Fifth International Workshop on*. IEEE, 2017, pp. 1–6.

[6] A. Gudipati, D. Perry, L. E. Li, and S. Katti, “Softran: Software defined radio access network,” in *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking*. ACM, 2013, pp. 25–30.

[7] X. Foukas, N. Nikaein, M. M. Kassem, M. K. Marina, and K. Kontovasilis, “Flexran: A flexible and programmable platform for software-defined radio access networks,” in *Proceedings of the 12th International on Conference on emerging Networking EXperiments and Technologies*. ACM, 2016, pp. 427–441.

[8] IDC, “The new need for speed in the datacenter network,” March 2015.

[9] N. Alliance, “Further study on critical c-ran technologies,” *Next Generation Mobile Networks*, 2015.

[10] E. Dahlman, S. Parkvall, and J. Skold, *4G: LTE/LTE-Advanced for Mobile Broadband*. Elsevier Science, 2013.

[11] K. Arshad, “Lte system level performance in the presence of cqi feedback uplink delay and mobility,” in *Communications, Signal Processing, and their Applications (ICCSPA), 2015 International Conference on*. IEEE, 2015, pp. 1–5.

[12] B. Sklar, “Rayleigh fading channels,” *Mobile Communications Handbook (Ed. SS Suthersan). CRC Press*, 1999.