Contention management for Cloud RAN over an optical ring

Dominique Barth[1] and Maël Guiraud [1,2] and Yann Strozecki [1]

[1]*David Laboratory, UVSQ*
[2]*Nokia Bell Labs France*

N-GREEN project has for goal to design a low cost optical ring technology with good performances (throughput, latency ...) without using expensive end to end connections. We study the compatibility of such a technology with the development of the Cloud RAN, a latency critical application which is a major aspect of 5G deployment. We show that deterministically managing Cloud RAN traffic minimizes its latency without impacting the latency of others traffics.

Keywords : Optical ring, Latency, C-RAN, SDN

TODO: ça marque et dans le titre, ça ne devrait pas
Il faut citer l'ANR NGREEN + n'importe quel truc de charoni qui explique leur architecture NGREEN

1 Introduction

Network providers have to design inexpensive networks while the amount of data and online applications significantly increases. Much of these applications have QoS criteria, like a minimal throughput or a maximal latency. The N-GREEN project aims to design a high performing optical ring while ensuring a minimal cost for providers. The current solutions with good QoS [1], establishes end to end connection (E2E) between the nodes, which is extremely expensive. The N-GREEN optical ring is designed to ensure good performances and the hardware it requires scales linearly with the number of nodes while E2E scales quadratically making it impractical for more than a few nodes.

In this article, we study a Cloud RAN (C-RAN) application in the N-GREEN optical ring [2]. C-RAN is one of the major area of development for 5G, that consists in centralizing the computation units or **BaseBand Units** (BBU) of the **Remote Radio Heads** (RRH) in one datacenter. The latency of the messages between the BBU and the RRH is critical [3, 4]. TODO: mettre les citations sur la latence dans la 5G, archi ngreen et concurents E2E à reprendre de nos précédents articles The aim of the article is to propose simple methods to deterministically manage the periodic C-RAN traffic. In a previous work [5], the authors have studied this problem for a star shaped network. Here the load due to the CRAN traffic is small which makes the problem easier. However, we add several new difficulties: the messages from RRHs are fragmented, there are other traffics which must not be to impacted and the broadcast and select policy of the ring makes reservation of a specific time for emission somewhat wasteful.

In Sec. 2, we model the optical ring and the traffic flow. In Sec. 3, we study the latency when using stochastic multiplexing to manage packets insertions on the ring, with or without priority for C-RAN packets. In Sec. 4, we propose deterministic ways to manage C-RAN packets without buffers and thus with a minimal latency compared to stochastic multiplexing. The best method to do so also improves the latency of best effort packets. TODO: Je sais pas si il faut parler de SDN, qui est un keyword a la mode mais peut etre que l'architecture ngreen nous propose, demander a chiaroni, Yann : je pense qu'il faut parler de SDN, reprendre peut être des choses de notre article précédent sur Cloud RAN

This work was developed around the ANR N-GREEN project. The authors thanks all the Nokia Bell Labs team for their collaboration.

2 Model of C-RAN traffic over an optical ring

N-GREEN Optical ring The unidirectional optical ring is represented by an oriented cycle. The vertices of the cycle represents the nodes of the ring, where traffic arrives. The edges (u,v) of the cycle have an integer weight $\omega(u,v)$ which represents the time to transmit a unit of information from u to v. We denote by RS (ringsize) the length of the cycle, that is the sum of $\omega(u,v)$ for all arcs (u,v). Each arc (u,v) can be seen as a sequence of $\omega(u,v)$ **containers**, of capacity C, expressed in Bytes. The containers are filled by the nodes. The node u can put some data in the first container of the arc (u,v) *only if it is free*. The dynamic of the network is simple: at each unit of time, data contained in a container go to the next. The ring follows a **broadcast and select with emission release policy**. When a container is filled by some node u, it is freed when it comes back at u after a whole round trip. TODO: on ne parle pas assez de la periode je trouve

C-RAN traffic There are n RRHs attached to the ring, the RRH i being attached to the vertex u_i (several antennas can be attached to the same vertex). The antenna are linked to the nodes of the ring through an electronic interface of bit rate R Bps. The ring has a larger rate of $F \times R$ Bps. The integer F is called the **acceleration factor** between the electronic and the optical domains. A node aggregates the data received on the electronic interface during F unit of times to create a packet of size C which completely fills one container of the ring. An RRH emits a C-RAN traffic during a time ET (emission time) in each period P, which consists in a packet of size C each F unit of times.

The data of the RRH i arrives at node u at a time m_i called the **offset**. The offset can be determined by the designer of the system and can be different for each RRH but must remain the same over all periods. We assume that all BBUs are contained in the same data-center attached to the node v. The data from u is routed to its BBU at node v through the ring and arrives at time $m_i + \omega(u,v)$ if it has been inserted in the ring upon arrival. Then after some computation time (which is supposed to be zero to simplify the model), an answer is sent back from the BBU to the RRH. The same quantity of data is emitted by each BBU or RRH during any period. The **latency** of a message is the time it waits in a node before being put in a free container on the ring. The aim of our study is to minimize the latency of the C-RAN traffic, both from the RRHs and the BBUs. In fact in Sec.4 we propose a deterministic mechanism with zero latency. The only downside is that other data used the optical ring and that their latency may be increased. We shortly describe the nature of this additional traffic in the next subsection.

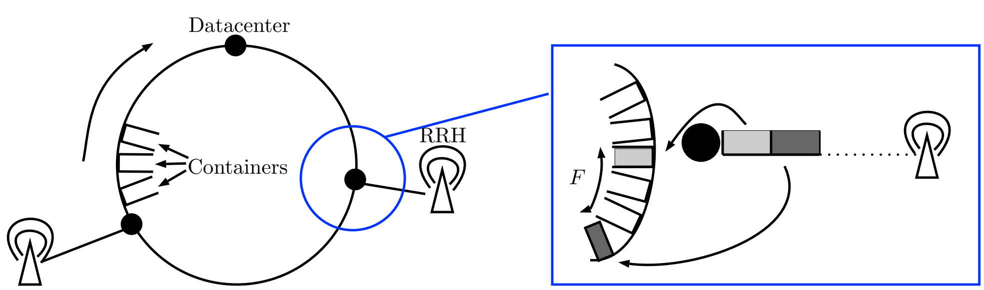

Figure 1: Opto-electronic interface.

TODO: Améliorer le dessin: un seul schéma avec les notations et horizontal pour pas perdre de place. Éventuellement faire une représentation du ring et en zoom faire l'interface

TODO: Décrire ici les degrés de liberté -¿ quand ça part + la latence

Best effort traffic The optical ring supports other traffics, corresponding to the internet flow. We call this traffic **Best Effort** (BE), since it is less critical and one can increase the latency of BE messages to improve the latency of C-RAN messages. A contention buffer is filled by a batch arrival process of BE data. Then according to some parameters (fill rate and maximum wait time) a container is sent on the ring from the contention buffer. The arrival of BE messages is modeled by a temporal law that gives the distribution of times between two arrivals of a container of BE messages in a node. This distribution is obtained by choosing optimal parameters for the contention buffer [].

3 Performance evaluation of the N-GREEN optical ring

One first study the latency of the C-RAN and BE traffics when the ring follows an opportunistic insertion policy: when a node wants to send some data, it uses the first free container available on the ring. If a node can not send a container, because there is no free slots or several containers must be sent at the same time, the containers surplus are buffered in an insertion buffer. Two different method to manage this insertion buffer are proposed here. The FIFO rule is compared to another method that uses two insertion buffers: one for the containers with BE, and another for the containers with C-RAN. If there is a free container on the ring, the node wants to empty the C-RAN insertion buffer first. Fig. 2 gives the cumulative distribution of both C-RAN and BE traffics latencies for the FIFO rule and the "C-RAN first" rule. On this experiment, the offsets of the RRH are randomly drawn in the period. The parameters of the ring are given by Fig. 3. Those results are computed over 100 instances of optical rings that have ran during $10,000,000$ unit of time.

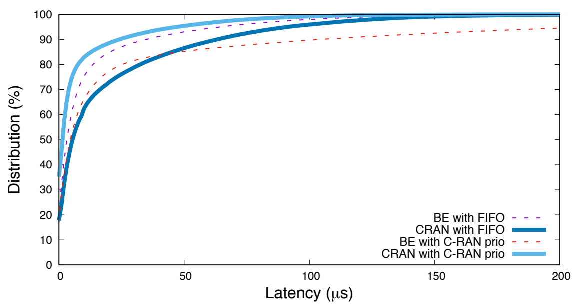

Figure 2: Performances of FIFO compared to C-RAN first.

Acceleration factor F	10
Bit rate of an electronic interface R	10 Gbps
Container size C	$12,500$ B
Slot duration $C/F \times R$	1μ s
Optical ring bit rate $F \times R$	100 Gbps
Length of the ring RS	100
Emission time ET	500
Period P	$1,000$
Number of RRH	5
Number of nodes n	5

Figure 3: Parameters experiments, based on N-GREEN.

Unsurprisingly, the latency of the C-RAN traffic is better when we prioritize the C-RAN messages, while the BE traffic is penalized. Nevertheless, there is still 10% of the C-RAN traffic with a latency higher than 50μs.

4 Deterministic approach for 0 latency on C-RAN

Some studies have already been made in order to set the offsets of C-RAN traffic for a different topology [5]. It appears that this problem is not hard in the N-GREEN optical ring. Therefore, one must try to set the offsets for C-RAN traffic such that the BE-traffic latency be not impacted too much.

In order to ensure 0 latency for C-RAN, the container needed to send a C-RAN message from the RRH i must be free at time m_i. Consequently the node reserve the written access this container at time $m_i - RS$. Thus, during the last round trip of the container before m_i, the others nodes can only free the reserved containers.

Since the computation time of the BBU is supposed to be zero (or a multiple of the period), setting the offsets m_i for all RRH i also sets the time $m_i + \omega(u,v)$, called **BBU offset**. Because all the BBU are located in the same datacenter, setting the BBU-offsets without coliisions directly avoid the collision between the C-RAN. A **macro slot** is a sequence of F slots. A RRH can emit at most once every macro slot but requires two free containers (one for the uplink message, and one for the downlink one) to ensure zero latency. A **pair** is a sequence of two consecutive slots in the macro slot, the messages of two RRH assigned to different pair can not collide.

Proposition 1. *If several RRH shares a pair, it is possible to set the BBU-offsets such that there is no collisions between the messages if $P \geq n \times ET + RS$.*

Proof. One show that proposition by induction. In this proof, the model is simplified in order to study the behavior of one pair, thus, both of the RRH emit some containers every slots, during ET slots in the period P. **Base case:** Consider $P = 2.ET + RS$ and two RRH each connected to the nodes A and B. We assume that A starts to emit a time 1 during ET slots. The containers of A arrive in B during the interval $[\omega(A,B);\omega(A,B)+ET[$. Then, B can immediately emits its containers at time $\omega(A,B)+ET$. Those containers from B arrives at A at during $[\omega(A,B)+ET+\omega(B,A);\omega(A,B)+ET+\omega(B,A)+ET] = [ET+RS;2.ET+RS] = [ET+RS;P]$. Then, A can emit in the next period, without collisions. Thus, for $n=2$, it is possible to assign two RRH on the same pair if $P = 2\times ET + RS$.

Induction step: A ring can support n RRH with a period of size $P = n\times ET + RS$. One must show that if there is one more RRH on the ring, the period must be at least $P = (n+1)\times ET + RS$.

Take a sequence of container of size P on a ring with n RRH. One can observe that this sequence is the same at every nodes, shifted of the length of the routes between the nodes. For instance, with two nodes A and B, the sequence in A is the same that the sequence in B shifted by $\omega(A,B)$. In addition if the sequence is increased by ET free containers, the periodic emissions of all others nodes are not impacted, there is just a new interval in the sequence to carry the traffic of a new RRH. Thus a period $P = (n+1)\times ET + RS$ is enough to carry the traffic of $n+1$ RRH. □

For the purpose of decreasing the BE traffic latency, it is interesting to regularly ensure some free pairs for the BE messages. To this end, the pairs must be filled with as much RRH as possible. Then, the free pairs are uniformly apportioned into the macro slot. Nevertheless, this is not always enough to ensure a good BE traffic latency. Indeed, with the previous parameters, with $ET = \frac{P}{2}$, $F = 10$ and $n = 5$, a pair is given to each antenna, and if all the RRH starts to emit in the same macro slots, a long sequence of reserved container will highly impact the latency of the BE traffic. The solution to this issue is to uniformly distribute the BBU-offset of the first RRH of a pair on the period. In this situation, the number of free pairs in a macro slot is balanced on the period, that allows a more regular BE traffic. Fig. 4 is a compariason between the cumulative distribution of the BE traffic latency with the FIFO rule or with the reservation algorithm proposed here. Since the previous parameters were too restrictive to put several RRH on the same pair, the parameters ET, and n have been changed to respectively 200 and 12, to keep the same expected load. The

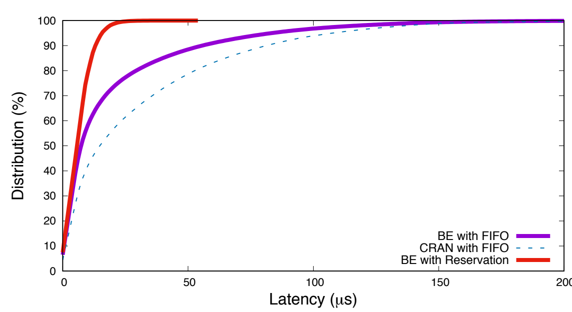

Figure 4: Impact of the reservation of the traffics.

performances are excellent, since the C-RAN traffic have 0 latency and the BE traffic have a better latency with reservation than with the FIFO rule. Indeed, our algorithm balanced the load of the C-RAN traffic over the period , that helps the BE traffic to not aggregate and create additional latency.

References

[1] A. Pizzinat, P. Chanclou, F. Saliou, and T. Diallo, "Things you should know about fronthaul," *Journal of Lightwave Technology*, vol. 33, no. 5, pp. 1077–1083, 2015.

[2] I. s. D. Chiaroni, "Network energy: Problematic and solutions towards sustainable ict," *International Commission of Optics (ICO 24)*, August 2018.

[3] Y. Bouguen, E. Hardouin, A. Maloberti, and F.-X. Wolff, *LTE et les réseaux 4G*. Editions Eyrolles, 2012.

[4] 3GPP, *3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Service requirements for the 5G system;*. Stage 1 (Release 16).

[5] B. Dominique, G. Maël, and S. Yann, "Deterministic scheduling of periodic messages for cloud ran," *arXiv preprint arXiv:1801.07029*, 2018.