

# Experimental Assessment of Degradation-triggered Reconfiguration in Optically Interconnected Cloud-RAN

Lluís Gifre<sup>1</sup>, Marc Ruiz<sup>1</sup>, Alberto Castro<sup>2</sup>, Roberto Proietti<sup>2</sup>, S. J. B. Yoo<sup>2</sup>, and Luis Velasco<sup>1\*</sup>

<sup>1</sup>Optical Communications Group (GCO), Universitat Politècnica de Catalunya (UPC), Barcelona, Spain

<sup>2</sup>Department of Electrical and Computer Engineering, University of California (UCDavis), Davis, USA.

\*e-mail: lvelasco@ac.upc.edu

**Abstract:** C-RAN control architecture is proposed and experimentally assessed. A local controller in each location receives latency, jitter, and BER monitoring data from local BBUs and detects a CPRI degradation, which triggers a mobile network reconfiguration.

© 2017 Optical Society of America

**OCIS codes:** (060.4250) Networks; (060.4256) Networks, network optimization

## 1. Introduction

The C-RAN architecture has demonstrated remarkable benefits and cost reductions [1]. As a result, there is an increasing interest to develop networking and control/management architectures to automate operation and re-configuration of optically interconnected mobile networks based C-RAN [2]. In C-RAN, the front haul network carries signals from RRHs in remote sites to BBUs in Central Offices (CO) using the CPRI protocol [3]. Moreover, to support handover, among others, coordination among neighboring RRH is needed, and X2 interfaces between BBUs in remote COs are required. Finally, S1 links towards the core location CO (hosting MME and S-GW) need also to be set-up over the back haul network. Among these elements, CPRI is the one that imposes the most stringent performance requirements, including: *i*) 100 $\mu$ s of one way delay; *ii*) 65ns of maximum jitter; and *iii*) 10<sup>-12</sup> of maximum bit error rate (BER).

In this paper, we propose a hierarchical C-RAN control architecture that includes CO controllers with a centralized C-RAN controller on top. The CO controllers collate monitoring data from the BBUs and packet nodes in a CO and apply data analytics to detect failures and degradation. When a failure or degradation is detected, the CO controller reports the C-RAN controller that can re-configure the mobile network.

## 2. C-RAN control and reconfiguration

Aiming at supporting C-RAN self-management, we propose the architecture in Fig. 1, where a C-RAN controller controls C-RAN resources, including BBU pools in COs, and activates and deactivates resources to adapt the mobile network to the actual load. A CO controller in each CO is responsible for handling (de)allocation of BBUs in its local pool and its interconnection with the front haul and back haul networks using a local packet node. Finally, the C-RAN controller can request connectivity for the front haul and back haul networks using the NBI offered by the metro and core network controllers, respectively. We assume that both, the metro and the core network are based on the optical technology (DWDM or EON).

Regarding monitoring, two sources of data in the COs are defined: *i*) received power and BER in the optical connections, measured by the optical transponders, and *ii*) latency, jitter, and BER for the CPRI protocol, measured by the BBUs. Monitoring data is sent to the CO controller that can make local decisions and to issue notification reports to the C-RAN controller in case of detecting faults or degradations. In the event of a failure or degradation, the C-RAN controller can reconfigure the mobile network by, e.g., allocating BBUs in a different CO and setting-up new optical connections in the front haul and back haul networks.

For illustrative purposes, Fig. 2 presents an example of C-RAN reconfiguration. When CO #1 controller detects poor CPRI protocol performance in the monitoring data received from one of the BBUs (e.g., excessive latency) it notifies the C-RAN controller, since no local action can be performed. The C-RAN controller can decide to allocate a new BBU in a different CO (CO #2) and to re-configure both the front haul and the back haul networks by creating new connections; unused resources can be eventually released.

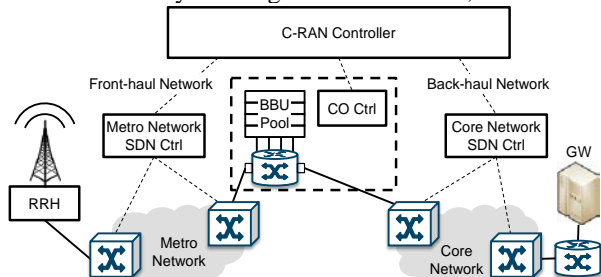


Fig. 1. C-RAN-based mobile network

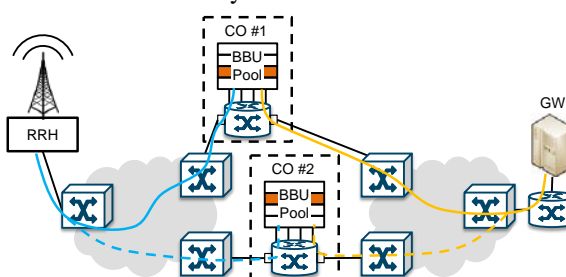


Fig. 2. Example of C-RAN reconfiguration

### 3. Proposed C-RAN control architecture and reconfiguration workflow

To implement the required reconfiguration, we propose the C-RAN control architecture in Fig. 3. BBU and packet nodes are under the CO controller and use the IPFIX protocol [4] to send monitoring data. Monitoring data is collated in a repository in the CO controller and it is periodically sent to the C-RAN controller. Local monitoring data is used to detect failures and degradations. RESTCONF [5] is used to configure monitoring parameters in the CO controller and to send asynchronous notifications to the C-RAN controller (e.g., after a threshold violation).

Two IPFIX templates have been defined to transport monitoring data reported by the packet nodes and the BBUs located in each CO. Table 1 presents the proposed template to monitor optical transponders in packet nodes. Among the defined fields, the template includes the identifier of the transponder being monitored and received power and BER in the monitored period. Regarding BBU monitoring, Table 2 presents the proposed template to monitor CPRI performance. The template includes the BBU and remote RRH identifiers, the average latency, jitter, and BER measured in the last period.

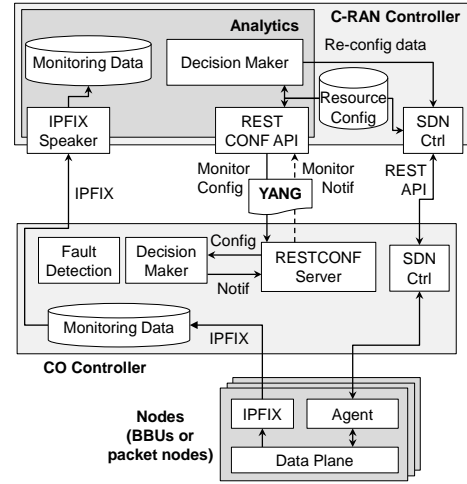


Fig. 3. Proposed C-RAN control architecture

Table 1. Proposed IPFIX template for BER monitoring

Field Name	Description	Units
observationDomainId	Id of the exporting device (Packet Node in the Central Office)	---
Timestamp	Timestamp of the sample	UTC since epoch
portId	Id of the transponder being monitored.	---
rxPower	Received optical power for this port.	dbm
ber	BER since the previous report for this port.	---

Table 2. Proposed IPFIX template for CPRI protocol monitoring

Field Name	Description	Units
observationDomainId	Id of the exporting device (BBU in the Central Office)	---
Timestamp	Timestamp of the sample	UTC since epoch
observationPointId	Id of the remote RRH the CPRI interface is connected to.	---
latencyMicroseconds	Average latency in the last period for this BBU.	μs
jitterNanoseconds	Average jitter in the last period for this BBU.	ns
ber	BER since the previous report (if any) for this port.	---

The configuration of the BBUs requires defining acceptable CPRI protocol performance values. To that end, we developed a YANG data model to include the thresholds for latency, jitter, and BER, as well as the monitoring period used to send monitoring data to the C-RAN controller. The configuration of the network requires defining the acceptable BER values for each port on the network devices, so we defined another YANG model that includes the threshold for the BER. Finally, the CO controller issues threshold violation notifications to the C-RAN controller following a third YANG data model that specifies the value that exceeds the threshold.

Two workflows have been defined. The first workflow (Fig. 4) starts when the C-RAN controller issues requests to configure the monitoring parameters in a CO, using a message following the defined YANG data model (messages 1 and 2 in Fig. 4). Next, BBUs and packet nodes start issuing periodic IPFIX reports to the CO controller (message 3), which aggregates monitoring data and forwards it to the C-RAN controller (message 4).

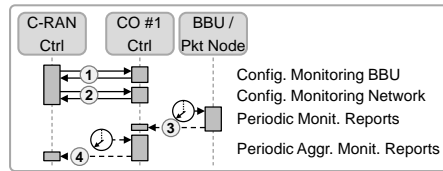


Fig. 4. Proposed workflow for monitoring configuration

The re-configuration workflow (Fig. 5), starts when the CO controller receives a monitoring report that exceeds the configured threshold (message 5 in Fig. 5). The CO controller immediately issues a notification message (following the defined YANG data model) to the C-RAN controller (message 6) specifying the node that exceeded the

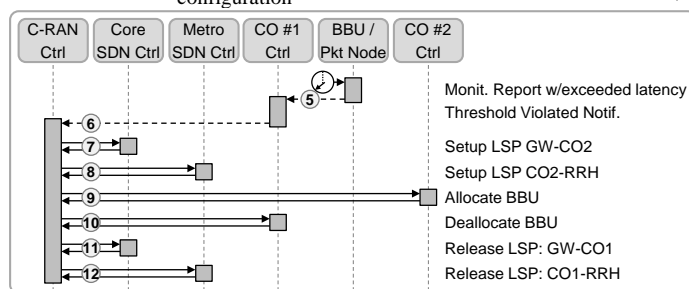


Fig. 5. Proposed workflow for reconfiguration

threshold, as well as the measured value.

Upon the reception of the notification, the C-RAN controller decides to connect the remote RRH to a BBU in a different CO to improve CPRI performance. To that end, it issues requests to the Core and Metro SDN controllers to set up LSPs connecting GW with CO#2 (message 7) and CO#2 with the RRH (message 8), allocates and configures a new BBU in CO#2's BBU pool (message 9),

and requests the deallocation of the current resources BBU to CO#1 (messages 10-12). The operation can be done in a *make-before-break* manner to avoid service disruption.

**4. Experimental assessment**

The experimental assessment was carried out in UPC’s SYNERGY test-bed. We reproduced the example in section 2, including a C-RAN controller, two CO controllers and the Core and Metro SDN controllers, and implemented the workflows proposed in section 3. The C-RAN controller and the module to emulate CO controllers were developed in Python. The SDN controllers were implemented as Ryu v4.6 applications using a REST API interface to accept requests from the C-RAN controller and OpenFlow v1.3 protocol to interact with the data plane. The data plane was based on OpenVSwitch v2.5 switches deployed on Mininet v2.2.

The CO controller implements a RESTCONF server that uses messages encoded using the defined YANG data models for receiving monitoring configuration and issuing asynchronous notifications to the C-RAN controller. Likewise, the CO controller implements an IPFIX interface to collect monitoring data from its packet nodes and BBUs and periodically reports aggregated information to the C-RAN controller.

The messages corresponding to the configuration workflow are listed in Fig. 6; the messages are labeled following the workflow in Fig. 4. Fig. 7 shows the detail for the configuration message (message 1), where BBU#10 is configured to issue reports with a period of 5 minutes and the CPRI thresholds are set to  $10^{-8}$  for the BER, 100  $\mu$ s for the latency and 65 ns for the jitter. Besides, Fig. 8 illustrates the details of configuring the optical transponders in the packet nodes; in particular, a period length of 5 minutes between reports is configured, as well as a maximum optical BER of  $10^{-8}$ .

The periodic monitoring reports of CPRI and transponder’s BER (message 4) are shown in Fig. 9.

Fig. 10 presents the list of messages for the reconfiguration workflow. When an IPFIX report exceeding a configured

Time	Src Addr	Dst Addr	Info
0.003	CRANCtrl	CO1	POST /restconf/api/data/co/bbu HTTP/1.1
0.008	CRANCtrl	CO1	HTTP/1.1 200 OK (application/yang.data+json)
0.011	CRANCtrl	CO1	POST /restconf/api/data/co/network HTTP/1.1
300.383	RT1	CO1	IPFIX flow ( 60 bytes) Obs-Domain-ID=251
300.438	BBU10	CO1	IPFIX flow ( 52 bytes) Obs-Domain-ID= 10
300.733	CO1	CRANCtrl	IPFIX flow (168 bytes) Obs-Domain-ID= 1

Fig. 6. Message list for configuration workflow

Fig. 7. BBU monitoring configuration

Fig. 8. Packet nodes monitoring configuration

Fig. 9. Periodic BER report

threshold is received in the CO controller (message 5 in Fig. 10), it throws an asynchronous notification to the C-RAN controller (message 6) specifying the node where that condition was detected and the value exceeding the threshold. Then, the C-RAN controller decides to reconfigure the mobile network and to connect the remote RRH to a new BBU in a different CO (messages 7-12).

**5. Conclusions**

A C-RAN control architecture has been defined. It defines a CO controller that collates monitoring data from the local packet nodes and BBUs and detects degradations. When a degradation is detected, a notification is sent to the centralized C-RAN controller that can decide to reconfigure the mobile network by allocating new resources and releasing the previously used ones. YANG data models and IPFIX templates have been defined to reconfigure and report the monitored values, respectively. Monitoring configuration and network reconfiguration workflows have been proposed. The proposed architecture and workflows have been experimentally assessed in our SYNERGY test-bed.

**References**

[1] A. Asensio et al., “Study of the Centralization Level of Optical Network-Supported Cloud RAN,” in Proc. ONDM, 2016.  
 [2] M. Raza et al., “Demonstration of Dynamic Resource Sharing Benefits in an Optical C-RAN,” JOCN, 2016.  
 [3] Common Public Radio Interface (CPRI). www.cpri.info/  
 [4] B. Claise et al., “Specification of the IPFIX Protocol,” IETF RFC 7011, 2013.  
 [5] A. Bierman et al., “RESTCONF Protocol,” IETF draft, work in progress, 2016.

Time	Src Addr	Dst Addr	Info
600.132	BBU10	CO1	IPFIX flow ( 52 bytes) Obs-Domain-ID= 10
600.911	CO1	CRANCtrl	HTTP/1.1 Event Stream (text/event-stream)
601.210	CRANCtrl	Core-SDNC	POST /tunnel/1/LSP-GW-CO12 HTTP/1.1
601.214	Core-SDNC	CRANCtrl	HTTP/1.1 200 OK (application/json)
601.220	CRANCtrl	Metro-SDNC	POST /tunnel/2/LSP-CO12-ANT HTTP/1.1
601.224	Metro-SDNC	CRANCtrl	HTTP/1.1 200 OK (application/json)
601.256	CRANCtrl	CO2	POST /restconf/api/data/co/bbu HTTP/1.1
601.269	CO2	CRANCtrl	HTTP/1.1 200 OK (application/yang.data+json)
601.308	CRANCtrl	CO1	DELETE /restconf/api/data/co/bbu[id=10]
601.313	CO1	CRANCtrl	HTTP/1.1 200 OK (application/yang.data+json)
601.316	CRANCtrl	Core-SDNC	DELETE /tunnel/1/LSP-GW-CO11 HTTP/1.1
601.319	Core-SDNC	CRANCtrl	HTTP/1.1 200 OK (application/json)
601.324	CRANCtrl	Metro-SDNC	DELETE /tunnel/2/LSP-CO11-ANT HTTP/1.1
601.327	Metro-SDNC	CRANCtrl	HTTP/1.1 200 OK (application/json)

Fig. 10. Message list for the reconfiguration workflow