



# Generalized Software Defined Network Platform for Radio Access Networks

Aravinthan Gopalasingham, Laurent Roullet, Nessrine Trabelsi, Chung Shue  
Chen, Abdelkrim Hebbar, Erick Bizouarn

► **To cite this version:**

Aravinthan Gopalasingham, Laurent Roullet, Nessrine Trabelsi, Chung Shue Chen, Abdelkrim Hebbar, et al.. Generalized Software Defined Network Platform for Radio Access Networks. IEEE Consumer Communications and Networking Conference (CCNC), Jan 2016, Las Vegas, United States. 2016. <hal-01216869>

**HAL Id: hal-01216869**

**<https://hal.inria.fr/hal-01216869>**

Submitted on 17 Oct 2015

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Generalized Software Defined Network Platform for Radio Access Networks

Aravinthan Gopalasingham, Laurent Roulet, Nessrine Trabelsi, Chung Shue Chen  
Abdelkrim Hebbar, Erick Bizouarn

Alcatel-Lucent Bell-Labs, Route de Villejust, 91620 Nozay, France  
INRIA-Alcatel-Lucent Bell Labs Joint Research Center

Emails: {gopalasingham.aravinthan, laurent.roulet, cs.chen, abdelkrim.hebbar, erick.bizouarn}@alcatel-lucent.com  
Email: nessrine.trabelsi@inria.fr

**Abstract**—Software defined networking (SDN) is growing rapidly in telecommunications due to its capability to efficiently manage end-to-end networks by decoupling control plane and data plane. The scalability and flexibility can bring benefits to network management and maintenance. However, it has yet to happen for wireless networks since there is a lack of practical solutions for migrating Radio Access Networks (RAN) to software defined wireless networks. Here, we design a wireless SDN controller based on OpenDayLight (ODL). We show how our framework can support typical use case and Self-Optimized Network algorithm. The design is highly flexible and can be leveraged for future wireless networks such as C-RAN and 5G.

**Index Terms**—SDN, RAN, LTE/5G, OpenDaylight, optimization.

## I. INTRODUCTION

Wireless data traffic has been growing very rapidly during the last few years due to the dramatic increase of wireless devices, online contents, and new applications. It is predicted that there will be tremendous increase in the number of connected devices by 2020 and mobile broadband network operators will face the challenge of delivering up to 1GB/user/day [1]. This huge demand will require significant network capacity increase and more importantly to adapt dynamically to various traffics and user patterns.

The introduction of LTE HetNet is the well known solution for increasing network capacity through higher spectrum spatial reuse by deploying low power small cells (SC) mixed with macro cells (MC). It is expected that 5G will be deployed as heterogeneous network, consisting of various radio access technologies. Besides, there will be a combination of macro and small cells of various size [2]. This would generally bring additional management and operational complexity.

3GPP Self-Organizing Network (SON) introduced in LTE is known for simplifying network operations and optimizing radio access network (RAN) through automated mechanism [3]. For example, SON supports low frequency method (e.g., self-configuration at power up, per day self-organization) and also high frequency method (e.g., distributed interference coordination via X2 interface).

Software defined networking (SDN) is the emerging technology for designing, building and managing networks, in decoupling control plane and data plane to better optimize each other. In fixed networks, telecom operators and vendors have already started to align their network management systems (NMS) with SDN paradigm: SDN provides operational

advantages like high flexibility and programmability, enables operators to have a complete control over the network from a centralized vantage point, and facilitates operators to easily deploy various applications, services and fine-tuned network policies. Though there is significant research for aligning SDN with RAN (see, e.g., [4]–[6]), currently none of the mainstream SDN controllers provides support for 4G RAN. This is disappointing since SDN controller could be both a potential alternate for 3GPP NMS as well as a nice tool for supporting advanced SON features. SDN would be also beneficial to future 5G since SDN can control 5G system from end to end, from access network to core network.

Inspired by SDN and OpenDayLight (ODL) [7], we propose in this paper a SDN platform for RAN with network abstraction. We define a new south bound (SB) protocol called “Radio Net Flow” using 3GPP Stream Control Transmission Protocol (SCTP). We build a prototype using ODL controller and emulated base stations (BSs). We validate our framework with Enhanced inter-cell interference coordination (eICIC) deployed as a north bound (NB) application.

## II. ARCHITECTURE AND IMPLEMENTATION

### A. Architecture Overview

Our architecture closely follows the SDN paradigm by providing global network view using a logically centralized controller. It is illustrated by Fig. 1. Since the focus of the architecture is to provide vendor-agnostic North Bound (NB) APIs for supporting SON algorithms and optimization, the controller has to provide abstraction of network Key Performance Indicators (KPI) and configuration parameters. The RAN applications would use the controller’s NB APIs to:

- discover existing network elements and topology,
- retrieve RAN measurements and configuration parameters, and
- re-configure RAN using optimal parameters.

Radio Net Flow (RNF) is the SB protocol introduced in the ODL controller as a RAN configuration protocol for (i) collecting radio network measurements and configurations, and (ii) re-configuring parameters of those radio elements. We implemented RNF using SCTP since it is the standard transport layer protocol in mobile network architecture, which can facilitate the integration of our SDN controller with

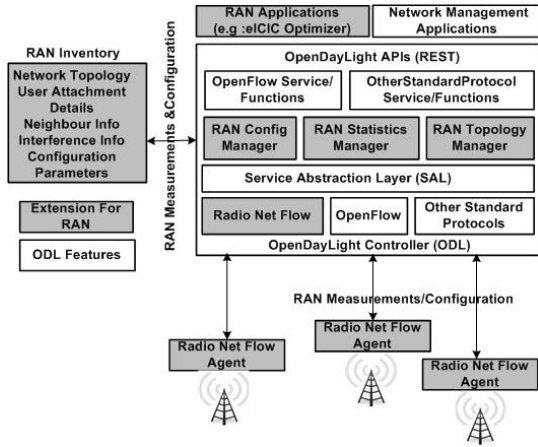


Fig. 1: SDN platform for RAN architecture.

standard RAN [8]. We developed the corresponding protocol agent called Radio Net Flow Agent (RNFA). RNFA can be integrated into any eNodeB as a communication interface to SDN controller. We introduced three service modules in the Service Abstraction Layer (SAL) of ODL: the RAN Configuration Manager, the RAN Statistics Manager, and the RAN Topology Manager. All these service modules interface with the RNF SB protocol using specific libraries. Through RNF, they can collect data from the RAN and also send configurations to the RAN.

Since the SAL imposes transient data storage, we introduced RAN Inventory that is an external database to maintain long term network state history. We used No-SQL database for performance considerations. During operations, RAN service modules update the RAN Inventory. On the other hand, NB applications access the RAN Inventory for information retrieval and settings via NB REST (HTTP RESTful compliant) APIs. The RAN Inventory is illustrated by Fig. 2. It consists of the following information for NB application to deploy optimization:

- Network Topology: A complete record of network topology contains all the macro and small cells connected to the controller, together with their unique IDs and neighboring cell information. The RAN Topology Manager generates such a graph in the logical table format. The record remains static until there is a change in the network topology (e.g., BS addition/removal/failure).
- User Attachment and Interference Information: The RAN Statistic Manager is responsible for populating this information in the RAN Inventory. It is updated via RNF protocol once a user performs handover or enters/exits the network. Interference information refers to the interference situation experienced by each user, indicating the user ID, its BS, interfering BSs, and the level of interference. It is sent to the controller periodically.
- Configuration Parameters: The RAN Configuration Manager handles configuration parameters in the RAN Inventory. NB applications store optimized parameters for each BS. The RAN Configuration Manager sends these

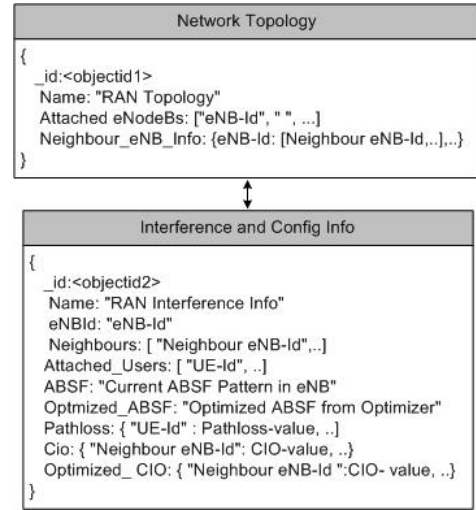


Fig. 2: RAN Inventory in MongoDB for eICIC optimization.

parameters periodically to the base stations.

### B. SDN Use Cases for RAN

Our SDN platform is prototyped to support RAN optimization. As shown in Table I, there are three potential use cases of SDN for RAN: radio optimization, network optimization, and load balancing/virtualization. Regarding RAN, we can categorize it into two major classes in terms of coordination requirement: Centralized RAN (C-RAN) and Distributed RAN (D-RAN). The SDN platform can provide dynamic control over fronthaul transport network and also mobile backhaul network to allocate available capacity, control latency, and meet QoS requirements. Besides, the network abstraction using ODL with multiple SB protocols can be used to adjust transport network resources (flow control technique based on OpenFlow [9] to control bandwidth and QoS assignment) according to the network requirements, e.g., to tackle temporary traffic peak and provide dynamic capacity.

Since the controller has the knowledge of RAN elements, it can trigger handover of mobile users in particular cell-edge users to lightly loaded neighboring BSs for more radio resource and throughput enhancement. This is a typical load balancing scenario that can be easily achieved by using SDN controller with dynamic resource allocation in fast recovery/response during failure and re-configuration among BBUs. In the following, we describe two use cases of our design.

1) *SDN for eICIC*: eICIC introduced in 3GPP-LTE is to conduct a time repartition of the physical resource blocks (PRBs) such that during some time slots called Almost Blank Subframe (ABS), macro cells will mute and do not transmit any data traffic. This allows to mitigate strong inter-cell interference that may severely degrade cell-edge user performance. Besides, in HetNets, due to the large difference in the transmit power between macro and small cells, user-cell association should be conducted in a more intelligent manner but not simply be decided due to the strongest received signal. In order to favor user association to small cell against macro cell, Cell Individual Offset (CIO) can be employed.

TABLE I: SDN platform for RAN use cases

	D-RAN	C-RAN
Radio Optimization	eICIC/ICIC (More generally dynamic/fast SON)	CoMP (e.g., Coordinated scheduling, dynamic point selection [11])
Network Optimization	Optimization of mobile backhaul network [12]	Fronthaul network optimization [13]
Load Balancing/Virtualization	Load balancing between base stations [5]	Creation, configuration and load balancing between vBBUs [14]

Our eICIC optimizer is based on a 2-tier model composed of macro and small cells that perform dynamic user-cell association and transmission scheduling optimization, which is to determine optimal CIO and ABS settings for each BS. We use a potential game best-response approach [10] for network utility maximization. During each optimization iteration at randomly chosen BS, we choose the setting that maximizes a localized potential function, which involves the BS and its neighboring cells. To facilitate the above optimization, each UE reports to its serving BS their experienced channel quality, signal level and interference power. These are sent via the SDN controller to the logically centralized eICIC optimizer.

Through iterations, the global utility of the network improves gradually and converges to an optimal setup. Note that here we can plug different optimization methods (e.g., heuristic or strict global optimization) as applications to the SDN controller, depending on system criteria and cost.

Note that this 2-tier model permits the separation of control decisions between the SDN controller and the local operation of eNBs, so as to facilitate flexible and dynamic network management and optimization. The parameters that would influence the neighboring eNBs need some central coordination in the controller, shown in Fig. 3. Here, CIO re-configuration can trigger handovers and maintain a load balanced network. Determining ABS ratio can coordinate the use of PRBs and mitigate inter-cell interference.

2) *Dynamic Point Selection*: LTE contains several coordinated multipoint transmission techniques (CoMP) in performing dynamic coordination of transmission and reception among BSs. It aims to enhance system overall performance, resource utilization efficiency, and user service quality (e.g., cell-edge user throughput). However, CoMP requires close coordination between geographically separated eNBs. The main techniques of CoMP are coordinated scheduling/beamforming (CS/CB), dynamic point selection (DPS), and joint transmission (JT). For example, DPS can occur given that there is coordination among BSs so UE can select the best cell in a very fast time scale or even on a subframe basis. This would require a communication and instruction of from which cell the network should direct its data to and also transmission synchronization. SDN is the choice for DPS such that one can maintain a fast switching and dynamic backhauling between the eNBs and benefit from the diversity gain when choosing the best serving point dynamically, e.g., according to channel conditions.

In our proposed architecture, CS/CB and DPS can be implemented as NB applications where they get the required input parameters such as neighbor cell list (NCL), radio resource

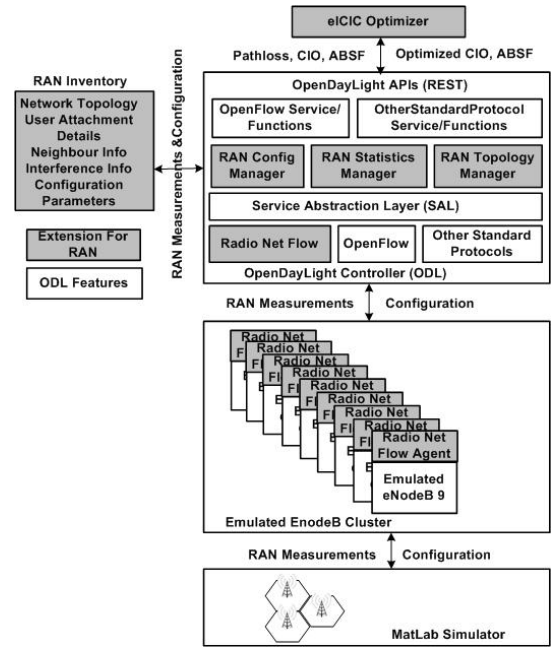


Fig. 3: Prototype for performance validation.

availability, backhauling information (data traffic and also CSI/control traffic) and transmission power setting from the RAN Inventory via REST APIs. Since in ODL, the NB applications can use multiple SB protocols for producing the global abstraction of LTE networks, CS/CB and DPS can easily use our RNF protocol (built on SCTP) for abstracting RAN and use OpenFlow protocol for backhaul network configuration to achieve CoMP and enhanced QoS.

### III. VALIDATION OF THE FRAMEWORK

To justify the proposed SDN and system design, we tested the first use case: the eICIC optimizer over LTE HetNet delivering optimized CIO and ABS values. We use a Matlab based LTE-compliant simulator developed by TU-Wien to emulate the cellular networks. It generates macro and small cells, mobile users, radio propagation and pathloss, and 3GPP-LTE setup. The network is iterated per Transmission Time Interval (TTI), which generates time varying link quality and traffics, and each eNB runs real-time resource scheduling.

At some point, the MATLAB simulator (shown in Fig. 3) connects with the eNBs emulators via UDP messages to exchange measurements and information. The IP addresses and ports of the emulated eNBs are known by the simulator. An initial connection is done to send the information of network topology (including a list of all cells in the cluster, their neighboring eNBs, the maximum transmission power of each cell). During optimization iteration, the Matlab simulator creates UDP socket and sends any updated messages for each eNB: the CIO values, the ABS patterns, the attachment of mobiles users, and the channel conditions (by the CQI from each UE to its serving cell and neighboring cells). All these messages are then sent via SCTP from the eNBs to the ODL to be stored at the RAN Inventory. The eICIC optimizer therefore sends requests using REST APIs to retrieve the information

TABLE II: eICIC simulation parameters

Radio	10MHz LTE Release 8 SISO, 3GPP TS36.942 -recommended pathloss and shadow fading models
Topology of eNBs	Hexagonal, a macro (MC) has 3 sectors & 6 SCs Antenna (MC/SC): Kathrein/omnidirectional Max Power (MC/SC): 40W/1W
Mobile users	25 UEs per MC, 10 UEs per SC, total 135 UEs, No mobility, Full buffer model
Optimization values	CIO choices = [0, 10, -10] ABS choices = [0%, 20%, 40%]

needed to conduct optimization iterations. The parameters of the tested LTE HetNet for simulation are given in Table II.

The optimization delivers the best CIO values for each cell. Before optimization, the CIO value at each eNB is by default 0. After eICIC optimization, the CIO value at a small cell can be increased to favor attaching to small cell. Meanwhile, the ABS ratio of macro cell is increased to reduce its interference to neighboring small cells in some PRBs. This is to offload UEs to small cells and improve radio resource utilization efficiency with interference management. Via SDN controller, at each TTI, new CIO and ABS values are sent to the different eNBs to update the user attachment and inter-cell interference coordination and tune the system dynamically.

#### IV. DISCUSSIONS & CONCLUSION

In the performance validation of the framework, we used a 16-core server for running the SDN controller, RAN Inventory and eICIC optimizer. We also measure the latency between the control node and BSs. Fig. 4 shows the result against the number of BSs activated. For 30 BSs, it takes about 2.7 ms for completely updating the database for each cycle of updates from all the BSs to reach the RAN Inventory and about 1.5 ms for sending back the re-configuration parameters from the SDN controller to reach all the BSs after eICIC optimization. In the validation scenario, we have time budget of 50 ms for applying optimal parameters to the BSs, so our SDN design can satisfy the time requirement for eICIC optimization.

As shown in Fig. 4, the latency increases steadily when the number of BSs increases. This indicates that the proposed architecture is scalable. Note that when adding a new BS to the controller, the latency is increased by some nanoseconds, which is mainly due to the time taken in updating the centralized database and querying the database for re-configuration parameters to the BSs. Indeed, it is possible to improve the performance for stringent time budget optimization algorithms by adding more hardware resources at the server for running the controller and database or applying techniques such as selecting proper schema and indexes for the applications and using high-performance storage. This could be a future work.

In the proposed design, we have deployed three different entities in the control node: the controller, the database, and the application. The motivation of using centralized database as an inventory rather than using XML-based yang data model in ODL is to generate historical data as an input for data analytic tools and applications, which can facilitate big data analytics and anticipatory network management. The implementation of REST APIs for NB applications is to enable the deployment of

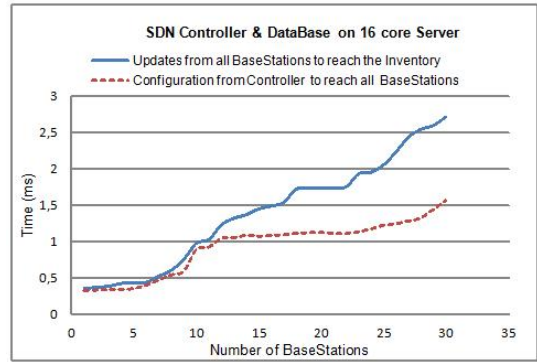


Fig. 4: Latency in the SDN Architecture.

RAN applications in remote server which can be developed in various programming languages. The SB protocol plug-in RNF is implemented using SCTP since SCTP has unique feature of creating multiple and parallel message streams between two end points which makes SCTP robust in comparison to TCP and UDP. Though our SDN platform is primarily developed for supporting LTE SON methods, it can be further extended to support different use cases in Cloud RAN and future 5G.

#### ACKNOWLEDGEMENT

The research leading to these results has received funding from European Unions 7th Framework Programme under grant no. 318115, Connectivity management for eneRgy Optimised Wireless Dense networks (CROWD), <http://www.ict-crowd.eu>.

#### REFERENCES

- [1] Nokia Networks APT700 White Paper. [Online] [http://networks.nokia.com/sites/sites/default/files/document/nokia\\_apt700\\_white\\_paper.pdf](http://networks.nokia.com/sites/sites/default/files/document/nokia_apt700_white_paper.pdf).
- [2] Nokia Government Relations Policy Paper: 5th Generation of Communication Networks. [Online] [http://networks.nokia.com/system/files/document/nokia\\_government\\_relations\\_policy\\_paper\\_on\\_5g\\_usa\\_version.pdf](http://networks.nokia.com/system/files/document/nokia_government_relations_policy_paper_on_5g_usa_version.pdf).
- [3] C. S. Chen, F. Baccelli, and L. Roullet, "Joint optimization of radio resources in small and macro cell networks," in *IEEE VTC*, 2011.
- [4] M. Bansal, J. Mehlman, S. Katti, and P. Levis, "OpenRadio: a programmable wireless dataplane," in *ACM HotSDN*, 2012, pp. 109–114.
- [5] A. Gudipati, D. Perry, L. E. Li, and S. Katti, "SoftRAN: software defined radio access network," in *ACM HotSDN*, 2013, pp. 25–30.
- [6] K.-K. Yap, M. Kobayashi, R. Sherwood, T.-Y. Huang, M. Chan, N. Handigol, and N. McKeown, "OpenRoads: empowering research in mobile networks," *SIGCOMM Comput. Commun. Rev.*, Jan. 2010.
- [7] OpenDayLight. <https://www.opendaylight.org>.
- [8] L. Ong and J. Yoakum, "An introduction to the stream control transmission protocol (SCTP)," Internet RFC 3286, May 2002.
- [9] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "OpenFlow: Enabling innovation in campus networks," *SIGCOMM Comput. Commun. Rev.*, 2008.
- [10] C. Singh and C. S. Chen, "Distributed downlink resource allocation in cellular networks through spatial adaptive play," in *25th International Teletraffic Congress (ITC)*, 2013.
- [11] R. Agrawal, A. Bedekar, R. Gupta, S. Kalyanasundaram, H. Kroener, and B. Natarajan, "Dynamic point selection for LTE-advanced: algorithms and performance," in *IEEE WCNC*, 2014, pp. 1392–1397.
- [12] H. Ali-Ahmad, C. Cicconetti, A. de la Oliva, M. Draxler, R. Gupta, V. Mancuso, L. Roullet, and V. Sciancalepore, "CROWD: an SDN approach for DenseNets," in *EWSDN*, Oct. 2013, pp. 25–31.
- [13] Nokia Solutions and Networks: SDN in Mobile Broadband Networks. [Online] [http://networks.nokia.com/sites/sites/default/files/document/015\\_sdn\\_impact\\_on\\_mbb\\_networks.pdf](http://networks.nokia.com/sites/sites/default/files/document/015_sdn_impact_on_mbb_networks.pdf).
- [14] M. Yang, Y. Li, D. Jin, L. Su, S. Ma, and L. Zeng, "OpenRAN: a software-defined RAN architecture via virtualization," in *ACM SIGCOMM*, 2013, pp. 549–550.