Coordinated scheduling in a Virtual-RAN prototype with OpenAirInterface

Niccolò Iardella, Giovanni Nardini,
Giovanni Stea, Antonio Virdis
Dip. Ing. dell'Informazione, University of Pisa, Italy
niccolo.iardella@unifi.it, g.nardini@ing.unipi.it
giovanni.stea@unipi.it, a.virdis@iet.unipi.it

Francesco Mauro, Dario Sabella, Marco Caretti,
Gian Michele Dell'Aera
Telecom Italia Lab., Turin, Italy
{francesco.mauro, dario.sabella}@telecomitalia.it
{marco.caretti, gianmichele.dellaera}@telecomitalia.it

***Abstract*— The virtualized Radio Access Network (V-RAN) is a key technology for 5G networks. In this paper we present a live prototype of Virtual RAN implementing a Coordinated Scheduling algorithm enforced by a centralized coordinator. The 5G proof of concept, devised to improve the usage of radio resource and efficiency, is realized by exploiting open-source software to fully virtualize the LTE eNodeBs, and accommodates commercial terminals. We implemented two coordination algorithms: a simple static one for testing purposes, and a dynamic one appeared in [1]. Preliminary results show that coordination actually isolates the eNodeBs, reducing inter-cell interference.**

Keywords—5G, OpenAirInterface, CoMP, Coordinated Scheduling, Prototype

I. Introduction

The incoming fifth generation cellular networks will need to provide new services with strong and heterogeneous requirements, that will be reflected in the necessity for operators to deploy flexible and efficient hardware (HW) and software (SW) infrastructure. Virtual-RAN is a new architectural paradigm, where a reconfigurable general-purpose processing (GPP) hardware is used to pool a significant number (~hundreds) of cells. The baseband resources in principle can be located in the same or different physical locations (the "cloud"). Operators recently dedicated an increased attention to V-RAN, especially because it abates the cost to manage, maintain, and expand the RAN. In this work we describe a live V-RAN prototype where a central coordinator realizes Coordinated Scheduling in a CoMP (Coordinated MultiPoint) setting, to assess the impact of coordination on various metrics (throughput, energy efficiency, interference, etc.). We report some preliminary performance results. The above work is carried out in the framework of the Flex5Gware EU project [2].

II. Description of the Prototype

The V-RAN prototype, outlined in Fig. 1, consists in three general-purpose machines, each one hosting one virtual BBU (vBBU) realized with a customized version of the open source OpenAirInterface (OAI) framework [3], enhanced with our modifications [4-5]. Each vBBU is connected to a RRH (Radio Remote Head), which is implemented using Ettus USRP B210 boards. UEs are commercial Huawei E392u-12 dongles or smartphones. Another machine hosts a software *coordinator*, which communicates with the vBBUs. The coordinator runs a Coordinated Multi-Point/Coordinated Scheduling (CoMP/CS) algorithm: it prepares a *frame partitioning (FP)*, i.e. a list Resource Block (RB) that each cell (vBBU) is allowed to use; the FP is sent to the coordinated vBBUs, which then apply it. The coordinator can run two algorithms: a *static* one, that shares the available RBs proportionally according to the number of coordinated nodes, and the *dynamic* one presented in [1], which we briefly recall. The static algorithm has been implemented in a preliminary version, in order to initially concentrate the attention on the testing of the latter. In this first version, vBBUs only communicate their *presence* to the coordinator, which shares the same number of RBs among all present nodes, thus implementing a crude mutual exclusion approach. In the dynamic algorithm, the coordinator receives *requests* by the vBBUs, as a numbers of RBs where each node requests neither, either or both the other nodes to be muted. If requests are mutually incompatible (i.e., there are not enough RBs to satisfy all of them), then the coordinator curbs them proportionally until they are, and then sends back the new FP to the nodes. Nodes then proceed to schedule their UEs in the RBs that they are granted in the FP, also knowing who else is going to use them (hence what interference to expect). As shown below, the software architecture of the prototype allows one to plug in other algorithms with minor and localized modifications.

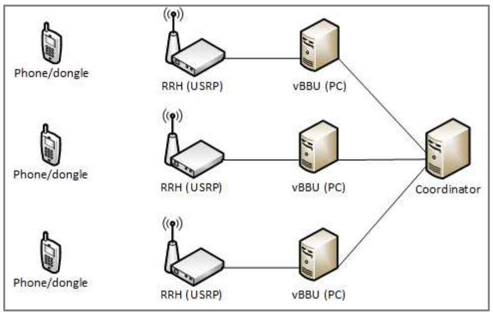

Fig. 1. Prototype overview

In the prototype, we designed the coordinator to be asynchronous w.r.t. the nodes. While the nodes perform scheduling once per TTI (i.e., once per millisecond), the coordinator can work (i.e., produce a new FP) on a different timescale, which allows different algorithms to be plugged in, with increasing complexity. This is achieved by implementing the different phases of the coordinator – the part that receives scheduling requests, the part that sends responses and the part that runs the algorithm – with different threads that run periodically at configurable periods. Receiving and scheduling threads are decoupled by a moving average accumulation buffer. The above architecture is shown in Fig. 2. The coordinator and vBBUs communicate via UDP sockets. When a request from a new node appears, the coordinator reacts accordingly by creating data structures dedicated to it, whereas inactive nodes are purged after a timeout. Moving-average buffers ensure tolerance to occasional losses of node requests, whereas nodes apply the last received FP, hence losing one will only cause temporary inconsistencies.

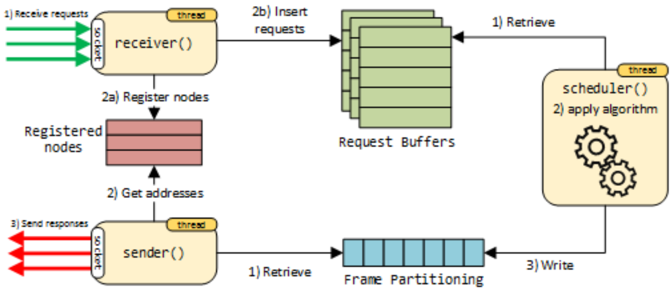

Fig. 2. Architecture of the CoMP/CS Coordinator

III. First Results and Evaluation

In the preliminary test of the prototype we used the coordination system described above together with OpenAirInterface system emulation – hence, eNBs and air interface have been emulated in software. We tested the Coordinator interfaces and operations in a scenario with one emulated eNB and one associated UE. We also implemented simple eNB stubs that generate requests and receive responses, to test the coordinator also under scenarios unachievable in practice with a live prototype. We used the built-in OAI traffic generator on the emulated node to send CBR downlink (DL) traffic from the node to the UEs and evaluate the throughput when coordination is applied. The results (Fig. 3) indicate that when the mutual exclusion algorithm is applied – therefore, a lower amount of RBs are assigned to each node by the coordinator – the DL throughput between the eNB and the UE is proportionally reduced, which is the expected outcome. In the emulated scenario, inter-cell interference and its effects over reported CQIs could not be evaluated.
As a second testing phase we introduced hardware devices, implementing eNBs with OAI BBUs and USRP boards, and UEs with commercial dongles and smartphones. As expected, the usage of hardware RF equipment introduced additional complexity. In particular, we were able to set up a scenario with 2 OAI+USRP nodes and 2 users, putting each node and the associated UE in a shielded box. This scenario allows us to check the interworking among coordinator and vBBUs. Fig. 4 shows that eNB1 and eNB2 without the coordinator allocates RBs with no restrictions. Fig. 5 instead, shows that when coordinator is enabled each eNB uses only the portion of bandwidth it is allowed to. As a next step, we plan to analyze inter-cell interference and its effect on CQIs. Without the shielded box, in fact, the signal fluctuation due to propagation characteristics introduces instability and synchronization problems between UEs and the eNB. This behavior is also due to UEs' attempts to probe the neighbor cells for measurement and handover opportunities, a feature still not supported by OAI. To circumvent these problems, the next tests will be performed in a controlled environment by using wired connections, attenuators and splitters to emulate inter-cell interference.

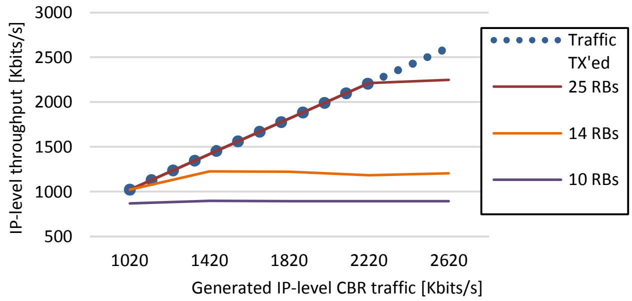

Fig. 3. DL throughput experienced by UEs vs. offered load, when different amounts of RBs are assigned to the serving eNB by the coordinator.

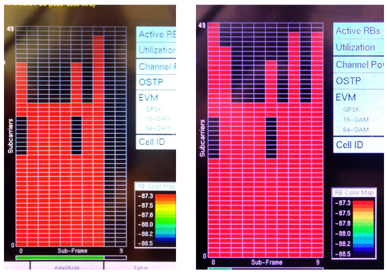

Fig. 4. DL allocation on two uncoordinated eNBs, measured with a spectrum analyzer. Both eNB allocate almost all the RBs on the same subcarriers.

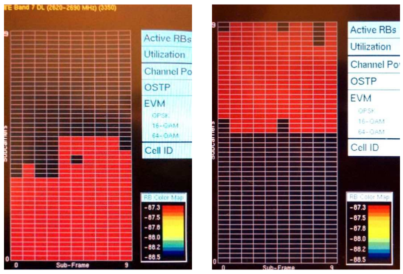

Fig. 5. DL allocation on two *coordinated* eNBs. Both eNBs allocate roughly half the available RBs, on different subcarriers.

Acknowledgments

The subject matter of this paper includes description of results of a joint research project carried out by Telecom Italia and the University of Pisa. Telecom Italia reserves all proprietary rights in any process, procedure, algorithm, article of manufacture, or other result of said project herein described. This work was partially supported by the European Commission in the framework of the H2020-ICT-2014-2 project Flex5Gware (grant agreement no.671563).

References

[1] G. Nardini, *et. al*, "Practical large-scale coordinated scheduling in LTE-Advanced networks", Wireless Networks, 22:(1), pp. 11-31, 2016

[2] Flex5Gware website: http://www.flex5gware.eu (accessed Mar. 2016)

[3] R. Wang *et al.* "OpenAirInterface - An effective emulation platform for LTE and LTE-Advanced". Proc. ICUFN 2014, Shanghai, Ch., pp. 127–132.

[4] N. Iardella, *et al.*, "Statistically sound experiments with OpenAirInterface Cloud-RAN prototypes", Proc. of CLEEN 2016, Grenoble, FR, May 2016

[5] A. Virdis, N. Iardella, G. Stea, D. Sabella, "Performance analysis of OpenAirInterface system emulation", Proc. of PMECT 2015, Rome, IT, August 26, 2015