Disjoint Embb and URLLC Scheduling in Cloud Native RAN Deployments

Introduction

Network slicing requires a RAN system to support Embb, URLLC and MMTC UEs with different SLA requirements, of which URLCC has the most stringent, followed by the Embb and the MMTC flows. These QoS requirements have been identified in 3GPP specification 38.xxx[ ]. The various scheduling proposals to handle URLLC and Embb flows with different QoS requirements in the RAN have been made under the assumption that a single schedular manages these two kinds of flows. This is true when the PDCP, RLC, MAC and the PHY layers are integrated as a single monolithic implementation as realized in legacy eNodeB or gNodeB implementations. However, this design is changing towards an O-RAN’s cloud native and disaggregated deployment model. Here, eNodeB/gNodeB is functionally realized as two network functions, central unit (CU) and the distributed unit (DU). The CU with its control and user plane identified as CU-CP and CU-UP includes the RRC and the SDAP/PDCP functions respectively, and the DU has the RLC, MAC and High-PHY functions. This dissaggration of the eNodeB/gNodeB enables more flexibility in deployment and realize them as virtual network functions (VNF) over commodity hardware achieving the software-hardware de-coupling objective. The CU’s interface to the DU has been standardized by 3GPP and realized as the F1-C and F1-U for the control and user plane respectively. The Lower-PHY functions are realized in the O-RU and it interfaces with the DU over the O-RAN standardized Option split 7.2 interface[O-FH specification].

This paper offers another approach to schedule URLLC and Embb flows using disjoint scheduler residing in two different DUs but hosted on the same physical server. Having the two DU dedicated to serve Embb and URLLC users is practical and efficient considering performance benefits, traffic isolation and simiplicity of schedular implementation itself. The two DUs communication can be over high-speed local networking interface between the PODs or containers (if the choice of implementation is to realize them as two containers in the same POD) depending on the choice of implementation. The goal of this paper is not to delve into system level design details of realizing this, rather the goal is to propose and study a algorithm which operates in as part of the URLLC scehdular’s logic. The objective of this algorithm is to aggregate the traffic staticts from the Embb DUs such as its buffer state and its UE traffic arrival rate and use it to determine the PRBs that needs to be allocated to the URLLC and the Embb schedulers to be used in the future set of TTIs.

Next section will go into the details of the discjoint schedular design. Then we propose the algorithm for traffic and PRB estimation. We then evaluate this algorithm and compare its performance when joint scheduling is used.

Disjoint Schedular Design