

Software-based Implementation of Dual Connectivity for LTE

Carlos Pupiales, Ilker Demirkol
 Department of Network Engineering
 Universitat Politècnica de Catalunya
 Barcelona, Spain

carlos.pupiales@upc.edu, ilker.demirkol@entel.upc.edu

Abstract—One of the major challenges of current mobile networks is to increase the per-user data rate without significant infrastructure cost and using the already existing physical resources, i.e., the bandwidth. In this sense, Dual Connectivity emerges as a promising solution for LTE and future 5G to increase the throughput with minor changes in current systems. This work presents the implementation of the U-Plane of Dual Connectivity for LTE using the open source software Open Air Interface with commodity hardware and the impact of this technology on UDP and TCP performance.

Index Terms—Dual Connectivity, LTE, Open Air Interface

I. INTRODUCTION

Dual Connectivity (DC) is a 3GPP solution, on the purpose of which is to achieve higher data rates by aggregating traffic from multiple base stations, where the UE can simultaneously transmit and receive data on multiple carriers from a Master eNB (MeNB) and a Secondary eNB (SeNB). Dual Connectivity was initially proposed in Release 12 [1] to work in an environment with macro and small cells for LTE and recently it was included in the non-standalone version of 5G-NR in Release 15 [2]. In this work, we present an implementation of Dual Connectivity solution for option 3C using the open source software platform Open Air Interface (OAI) [3]. The developed implementation is evaluated through physical experiments using Software-defined Radio (SDR) and Generic Purpose Processor (GPP) elements, focusing on the performance of TCP and UDP. Finally, the validated implementation is offered to the community at [4] and it is expected to offer a base for further R&D studies on this topic.

II. IMPLEMENTATION

The general architecture implemented for this work is explained in [1] and shown in Fig. 1. Considering this, the solution focuses on enabling the split bearer at PDCP layer at MeNB and its forwarding through RLC layer at SeNB. Nevertheless, OAI does not allow an unique UE with two protocol stacks connected to MeNB and SeNB simultaneously. Hence, it was necessary to create a solution, where two UE instances communicating with MeNB and SeNB, mUE and sUE, respectively are interconnected to represent the ideal UE needed for DC. Moreover, this work implements the functionality of DC from the U-Plane perspective and it works for downlink only. For C-Plane, we use the basic functionality

of X2 interface, X2 Setup Procedure, to interconnect MeNB and SeNB.

A. DC Implementation for eNB

For split bearer, data are split at PDCP layer in MeNB and then forwarded to mUE, through Uu interface, and to RLC layer at SeNB through X2U interface. This new X2U interface handles the transfer of PDUs from PDCP layer at MeNB to RLC at SeNB and the communication between MeNB and SeNB using UDP, the GTP encapsulation required by the 3GPP specification is left as future work. We implement a simple flow control logic at PDCP for split bearer, which consists of sending PDUs alternately through MeNB or SeNB depending on the PDU sequence number.

B. DC Implementation for UE

Since we use two UE instances to represent the DC UE, these UEs must be interconnected so that we can combine the PDUs sent through MeNB and SeNB at the PDCP layer of mUE. The interconnection is done through a newly defined UE-DC interface which allows to send/receive data from RLC layer of sUE to PDCP layer of mUE, to retrieve the UE context at mUE for the PDUs coming from SeNB, and to manage the communication between mUE and sUE using UDP.

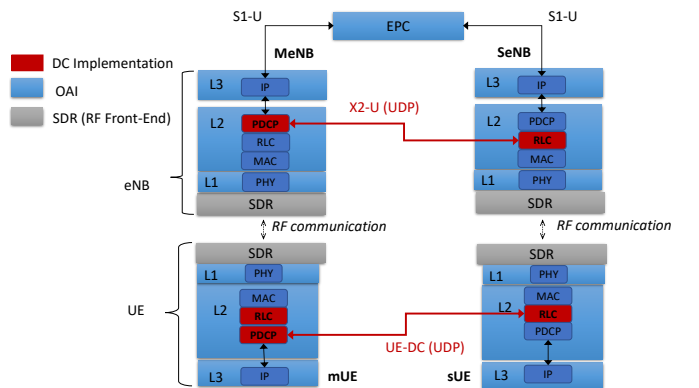


Fig. 1. Architecture for DC U-Plane Implementation.

III. PERFORMANCE EVALUATION

Performance is evaluated in two approaches: one without DC and another with DC. One performance metric for this

work is the achieved DL throughput of UDP/TCP traffic using 5 MHz of bandwidth. The other metric is the packet round-trip-delay measured for PDUs of 56 and 1400 bytes going via MeNB and SeNB in order to evaluate the impairments that X2 interface and split bearer procedure add to the performance of DC. The implementation is validated in the ORBIT testbed [5] where we used five hosts (two UEs, two eNBs, and one EPC), four SDRs, one Giga-Ethernet switch, and RF attenuators/cables. UEs are connected to eNBs using RF cables of 3.65 m approximately to reduce interferences and losses that wireless propagation may introduce to the system and affect the expected results. Table 1 shows the configuration parameters used in the testbed.

TABLE I
TESTBED CONFIGURATION PARAMETERS

	Parameter	Value
PC	CPU	Intel Core i7-4770 @ 3.4GHZ / 16 GB RAM
	OS	Ubuntu 16.04.1 with 4.15.0-52-lowlatency kernel
LTE	Duplex Mode	FDD
	Bandwidth	5 MHz
	DL Frequency	2.68 GHz for MeNB 2.63 GHz for SeNB

A. Packet Latency

Latency for downlink is measured for packets going from EPC to mUE using the *ping* tool. Table 2 summarizes the latency obtained for the proposed scenarios after sending 20 packets of 56 bytes and 1400 bytes each. The additional delay that the PDUs going through SeNB experience is caused by the processing time added at X2U and UE-DC interfaces which also might cause that PDUs arrive out-of-order at mUE.

TABLE II
PDU PACKET LATENCY SUMMARY

Parameter	LTE without DC		LTE with DC	
	56 bytes	1400 bytes	56 bytes	1400 bytes
DELAY_MIN (ms)	17.631	18.687	18.824	18.763
DELAY_AVG (ms)	30.065	29.851	28.807	30.24
DELAY_MAX (ms)	42.896	57.71	45.684	45.912

B. Throughput

Throughput that UE may reach depends mainly on channel conditions, hardware capabilities, and OAI software reliability/stability, so results may differ for different setups. For numerical comparison, we refer to the practical values achieved using the base implementation of OAI eNB with a category 3/4 COTS UE detailed in [5]. Additionally, the analysis is done using the *iperf* tool for both TCP and UDP during a session of 30 seconds where the complete results are depicted in Fig. 2 and summarized as follows:

- For LTE no DC, we got 16.9 and 16.3 Mbps for UDP and TCP traffic, respectively, which is indeed in the range of the expected results. Thus, these results represent a baseline to compare with the DC scenario.

- For LTE DC, we reached 33.9 and 12.4 Mbps for UDP and TCP traffic, respectively. For the former, this value shows that we indeed doubled the throughput reached in the no DC implementation. Even though the throughput increases, packets arrive out-of-order to mUE due to the delay introduced by the split bearer procedure. On the other hand, the low throughput for TCP traffic represents an acceptable performance considering that due to the implemented flow control logic, half of the sent PDUs take a longer path to reach the mUE causing longer delays, out-of-order arrival, and rejection/retransmission than the other half that affect the TCP performance.

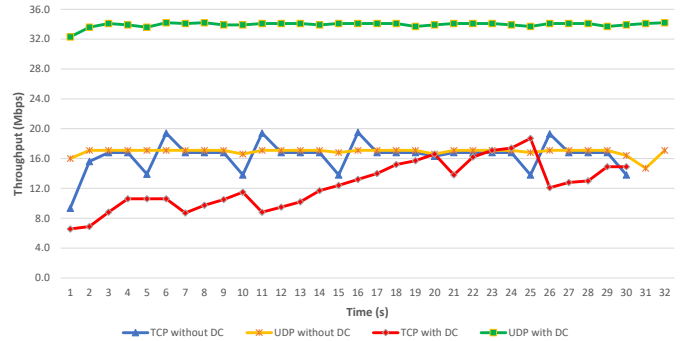


Fig. 2. Downlink Throughput for Legacy LTE and LTE DC.

IV. CONCLUSIONS

In this work, we presented a Dual Connectivity implementation for U-Plane based on scenario 3C using Open Air Interface and commodity hardware. Evaluations show a considerable improvement in UDP throughput; on the other hand, TCP throughput is affected by delays added by the split bearer procedure causing a performance degradation that can be mitigated using PDU reordering at PDCP layer and a more efficient flow control logic. Our current work focuses on the implementation of a reordering algorithm at PDCP to improve the worse performance of TCP against UDP.

ACKNOWLEDGMENT

This work has been supported by the European Project Dual Connectivity Solution for ORCA (DALI) as part of the Open Call 2 of H2020-ICT-2016-2017 (ORCA) project.

REFERENCES

- [1] 3GPP TR36.842, "Study on Small Cell Enhancements for E-UTRA and E-UTRAN - Higher Layer Aspects," v. 12.0.0, Dec. 2013
- [2] 3GPP TS 37.340 v15.0.0, 3rd Generation Partnership Project; NR; Technical Specification Group Radio Access Network; Multi-connectivity; Overall description; Stage-2 (Release 15), Jan. 2018.
- [3] N. Nikaein, M.K. Marina, S. Manickam, A. Dawson, R. Knopp, and C. Bonnet, "OpenAirInterface: a flexible platform for 5G research," ACM SIGCOMM Computer Communication Review, vol. 44, no. 5, pp. 33-38, Oct. 2014.
- [4] DALI GitHub, <https://github.com/ni/Orca-Dali-oai>
- [5] Orbit Homepage, <https://www.orbit-lab.org/>
- [6] EURECOM, The Basics of OpenAirInterface, Accessed on Sep. 3, 2019. [Online]. Available: <https://www.openairinterface.org>