Real-time Emulation Methodologies for Centralized Radio Access Networks

USE OF OPENAIRINTERFACE IN RESEARCH AND PROTOTYPING

Luis Felipe Ariza Vesga
Universidad Nacional de Colombia
Raymond Knopp
EURECOM

Nokia Bell Labs, Murray Hill, New Jersey - June 26, 2019

Agenda

Introduction

- There is a trade-off between network simulations, network emulators and real test-beds :
  1. **A network simulator** has good scalability and reproducibility.
  2. **A network emulator** has good applicability and captures 3GPP standard-compliant environments.
  3. **A test-bed** has good applicability but reproducibility issues in multi-vendor scenarios.
- Optimizing software functions and simulating the multipath channel in terms of a frequency domain representation, we decrease the signal processing complexity in a **software-only environment.**

Objectives

- Increase the scalability of real-time synthetic networks (Multiple Remote Radio Units and User Ends) in a software-only environment.
- Prototype 4G and 5G rapid proof-of-concept designs before launching to the market.
- Hybridize real-time synthetic network components, and Radio Frequency (RF) hardware for complex scenarios.

Architecture

- We extracted Primary and Secondary Synchronization Signals (PSS and SSS) information from the eNB and assigned to the UE (frame_type, cell_id).
- PBCH is decoded from the rxdataF at the UE (*initial_synch_freq*() function).

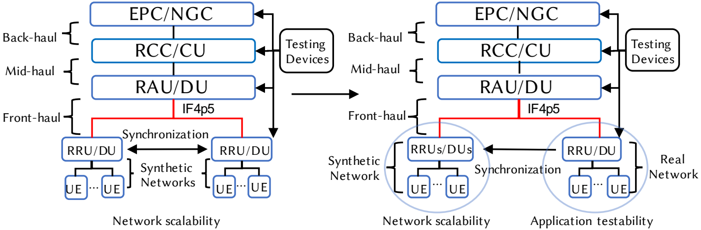

FIGURE – Architecture.

Functional split IF4p5

We created new methodologies for C-RANs, where I/Q signals are exchanged in the frequency domain (Split 7.1).

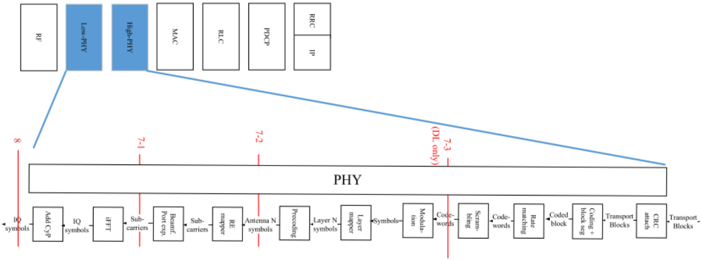

FIGURE – IF4p5 functional split at the transmitter [1]

Single Instruction Multiple Data

We implemented new SSE and AVX2 optimized functions of the multipath channel to improve the emulation speed.

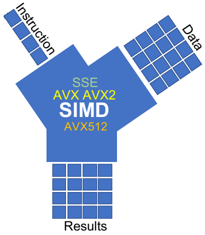

FIGURE – Single Instructions Multiple Data.

File	Function
abstraction.c	init_freq_channel_AVX_float
	freq_channel_AVX_float
	init_freq_channel_prach_AVX_float
	freq_channel_prach_AVX_float
	sincos256_ps
	log256_ps
	exp256_ps
multipath_channel.c	multipath_channel_freq_AVX_float
	multipath_channel_prach_freq_AVX_float
rangen_double.c	nfix_AVX, boxmuller_AVX_float
	SHR3_AVX, UNI_AVX, NOR_AVX
channel_sim.c	do_DL_sig_freq
	do_UL_sig_freq
	do_UL_sig_prach_freq
dac.c	dac_fixed_gain_AVX_float
	dac_fixed_gain_prach_AVX_float
rf.c	rf_rx_simple_freq_AVX_float
adc.c	adc_AVX_float
	adc_prach_AVX_float

TABLE – New AVX2 optimized functions of the multipath channel.

Function	Time Domain (s)	Frequency Domain(s)	Gain
`Downlink multipath channel`	45.58	9.774	4.66
`Uplink multipath channel`	46.048	11.356	4.06
`Downlink DAC`	19.303	13.815	1.4
`Uplink DAC`	19.487	13.99	1.39
Downlink receiver rf	**500.288**	**37.062**	**13.49**
Uplink receiver rf	**494.876**	**36.867**	**13.42**
`Downlink ADC`	18.52	2.304	8.04
`Uplink ADC`	18.489	2.122	8.71

TABLE – Average computation times in time and frequency domains. C-RAN architecture, 5 MHz of bandwidth, 10000 frames, AWGN channel model, and 5 MB of iperf traffic.

Gaussian random number generators

Gaussian random number generators (*rf_rx_simple_freq*) are employed to simulate the noise at the receiver.

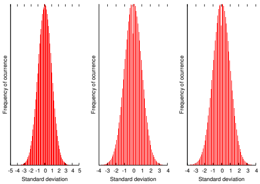

FIGURE – Ziggurat method to generate Gaussian random numbers.

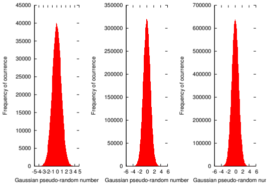

FIGURE – Box-Muller method to generate Gaussian random numbers.

Generator	Samples	Chi-Square	Computation time (ns/samples)
Box-Muller		9.99e+05	290
SSE Box-Muller	1e+06	9.99e+05	74.5
AVX2 Box-Muller		9.75e+05	37.4
Ziggurat		1e+06	220
SSE Ziggurat	1e+06	1e+06	78
AVX2 Ziggurat		1e+06	39

TABLE – Chi-Square and average computation time metrics for Box-Muller and Ziggurat methods.

Physical slot structure

In the frequency domain analysis the Cyclic Prefix is not implemented. Inter-symbol interference is not avoided. We change the *time_stamp* in *eNB_trx_read* and *UE_trx_read* functions.

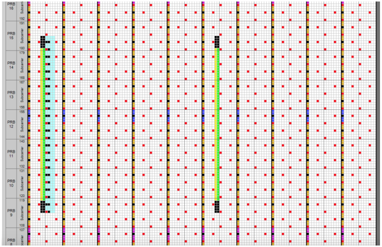

FIGURE – LTE resource blocks allocation (25 PRBs / 300 subcarrier indices / 512 FFT size) [2].

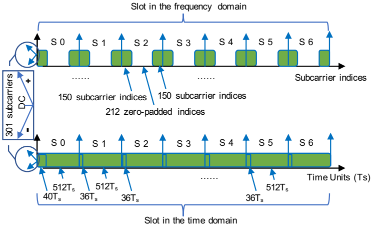

FIGURE – Physical slot structure. $T_s = \frac{1}{512*15000} = 130{:}21ns$

Time_stamp(time)=samples_per_tti=(40+512+6*36+6*512)*2 = 7680

Time_stamp(frequency)=symbols_per_tti (14) * ofdm_symbol_size (512)=7168.

Channel Frequency Response

We create *slot_fep_freq* and *lte_dl_channel_estimation_freq* functions to exclude dfts and to add an offset to the rxdataF vector.

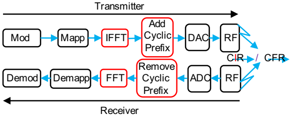

FIGURE – Orthogonal Frequency Division Multiplexing (OFDM) chain.

Tapped Delay Line (TDL) model [3] :

$h(m)=\sum_{l=0}^{N_p-1} \mathbf{a}[l] sinc(m \quad F_s(l + \quad) \quad _d \quad 0{:}5F_s \quad _{max})$

$H[k]=\sum_{i=0}^{N_p-1} \mathbf{a}[l](j \sin(2 \quad \frac{k}{N} m_l) \quad \cos(2 \quad \frac{k}{N} m_l))$

$m_l = F_s(l + \quad) \quad _d + 0{:}5F_s \quad _{max}; \quad _d = \frac{_{max}}{N_p}$

N = sampling rate,N_p = channel path number, F_s = sampling frequency, = real number to ensure $h(m)$ envelope continuity, $_{max}$ = maximum allowable delay in the channel, **a** is the channel state vector,

m = samples in the time domain, k = samples in the frequency domain.

$$r(m)=s(m)*h(m)+n(m) \quad (1)$$

$$R(k)=S(k).H(k)+N(k) \quad (2)$$

Synthetic Network Scalability

- Scalability is enabled changing the scheduler behaviour per component carrier (CC).
- Each IF4p5 link has a physical Ethernet connection.

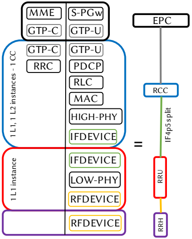

FIGURE – Simple Synthetic Network.

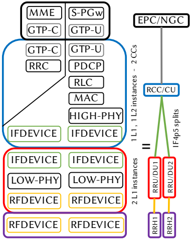

FIGURE – Synthetic Network Scalability.

phy_procedures_eNB_TX :~~if (eNB->CC_id == 0)~~

! *eNB_dlsch_ulsch_scheduler* (*Mod_id*; :::; CC_id)

! ! *schedule_ue_spec* (*module_idP*; :::; CC_id)

! ! ! *dlsch_scheduler_pre_processor* (*module_idP*; :::; CC_id)

! ! ! ! *store_dlsch_buffer* (*module_idP*; :::; CC_id)

! ! ! ! *assign_rbs_required* (*module_idP*; :::; CC_id)

! ! ! ! *sort_UEs* (*module_idP*; :::; CC_id)

! ! ! ! *store_dlsch_buffer* (*module_idP*; :::; CC_id)

Parameter	Single-UE Value / USRP B200mini-i
Band	7
Transmitter gain	90 dB
Receiver gain	120 dB
Transmitter power	15 dBm
Working Mode	FDD
Cyclic Prefix	Normal
Interface compression	A-law
System Bandwidth	5 MHz
Transmission Mode	1 SISO
Multipath Channel Model	AWGN / Rayleigh 1

TABLE – Network emulation parameters for the C-RAN.

FIGURE – The C-RAN is composed of 3 PCs for the EPC, the RCC, and RRUs. The USRP, antennas and the COTS UE are employed for validation.

FIGURE – Fronthaul and backhaul network segments for 3RRUs and 3 UEs.

Downlink and uplink maximum user throughputs have errors of 4.5% and 8.4% compared with speci cations respectively.

	MCS	Time	Frequency	USRP B200mini-i	TS 36.213 [4]
Downlink	28	17.5 Mb/s	17.5 Mb/s	17.5 Mb/s	18.336 Mb/s
Uplink	18	8.43 Mb/s	8.43 Mb/s	8.43 Mb/s	9.144 Mb/s

TABLE – Maximum user throughput (5 MHz of Bandwidth, AWGN channel model).

BLER for both domains are pretty similar, however there is still issues compared with reference [5].

FIGURE – Downlink Block Error Rate for different MCSs (5 MHz of Bandwidth, 5000 subframes, transmission mode 1, and Rayleigh channel model (1 tap)).

We accomplished a gain of almost one order of magnitude for both, uplink and downlink multipath channels.

FIGURE – Average computation time (Time domain).

Channel function	Time Domain (s)	Frequency Domain(s)	Gain
Uplink Channel	596.232	72.758	8.19
Downlink Channel	596.833	78.811	7.57
Uplink PRACH Channel	n/a	219.202	n/a

TABLE – Average computation times in time and frequency domains. C-RAN architecture, 5 MHz of bandwidth, 10000 frames, AWGN channel model, and 5 MB of iperf downlink traf c.

FIGURE – Average computation time (Frequency domain).

FIGURE – Average computation time benchmark.

RCCs	CCs/RRUs	UEs	Time Domain s	Frequency Domain s	B200mini-i s
		1	1193.065	151.66	0.0651
1	1	2	2262.5	330.972	N/A
		3	3614.066	523.208	N/A
		1	1280.8	162.25	N/A
1	2	2	2215.87	358.2	N/A
		3	N/A	622.164	N/A
		1	N/A	148.072	N/A
1	3	2	N/A	397.556	N/A
		3	N/A	546.334	N/A
.	.	.	Non-Real-time zone	.	.

TABLE – Average computation times of the multipath channel.

Real-time emulations can be improved for complex scenarios using AVX512 instructions and more threads.

We implemented the static coordinated scheduling. The RCC works as the coordinator using 1 scheduler and multiple CCs.

RCCs	CCs/RRUs	UEs	Time Domain s	Frequency Domain s
1	2	2	N/A	335.2
1	3	3	N/A	455.23
.	.	.	Non-Real-time zone	.

TABLE – Average computation times of the multipath channel. Time Domain does achieve synchronization.

FIGURE – Static Coordinated Scheduling. Subframes 0 and 5 are used for common channels.

FIGURE – Xforms (eNB->UE). Even subframes of UE->...thread[1].rxdataF and uplink traf c.

FIGURE – Real-time emulation methodologies for Centralized Radio Access Networks.

- We successfully implemented affordable real-time emulation methodologies in the frequency domain for C-RANs as a prototyping framework to rapid proof-of-concept and time-to-market designs in a software-only environment.
- We reduced near 10-fold the average computation time of the multipath channel in the frequency domain compared to the time domain. The cost in time we need to pay is related to the additional uplink PRACH channel.
- We improved the applicability and the scalability fro CRANs on top of OpenAirInterface.
- Our proposal allows real-time 3GPP standard-compliant C-RANs emulations for downlink and uplink transmissions.

Gitlab branches : large_scale_simulations for RRUs + UEs, and master_large_scale_emulations for the RCC and multiple CCs.

Several videos related to the extensions in the frequency domain.

- L. M. P. Larsen, A. Checko, and H. L. Christiansen, "A survey of the functional splits proposed for 5g mobile crosshaul networks," *IEEE Communications Surveys Tutorials*, vol. 21, no. 1, pp. 146–172, Firstquarter 2019.
- LG. (2019) Lte resource grid. [Online]. Available : http://niviuk.free.fr/lte_resource_grid.html
- F. Kaltenberger, "Low-complexity real-time signal processing for wireless communications," Ph.D. dissertation, Vienna University of Technology, Vienna, Austria, May 2007.
- 3GPP, "Evolved universal terrestrial radio access (e-utra) :physical layer procedures," 3rd Generation Partnertship Project (3GPP), Technical Specificacion (TS)

36.213, 10 2018, version 15.3.0. [Online]. Available : https://goo.gl/C8ePeU

EURECOM. (2011) Openairinterface dl simulation results. [Online]. Available : https://cordis.europa.eu/docs/projects/cnect/8/248268/080/deliverables/001-D33AppendixOpenAirInterfacelinklevelsimulationresults.pdf