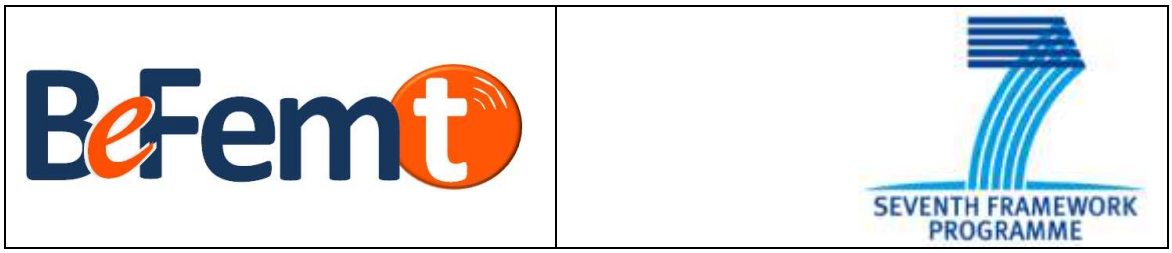

INFSO-ICT-248523 BeFEMTO

D6.2

Integration of selected algorithms into platforms & interfaces finalization

Contractual Date of Delivery to the CEC:	M24
Actual Date of Delivery to the CEC:	M24
Author(s):	P. Arozarena, L. Cucala, P. Dini, D. Dutoit, A. Felber, J. Fitzpatrick, R. García, F. Géhéniau, B. Iribarne, M. Lalam, M. López, J. Mangues, D. Marandin, S. Mayrargue, J. Nin, J. Núñez, M. Palmowski, M. Requena, S. Rouco, F. Zdarsky
Participant(s):	CEA, CTTC, mimoOn, NEC, SC, TID, TTI
Workpackage:	WP6: Integration and Proof of Concepts
Estimated person months:	80
Security:	**PU**
Nature:	**R**
Version:	1.0
Total number of pages:	81

Abstract:

This deliverable describes and summarizes in details how key algorithms and technologies that BeFEMTO Work Package 6 is going to demonstrate were both developed and integrated jointly into the different platforms and testbeds.

Keyword list:

Testbed, Algorithm Down-Selection, Algorithm Integration, Specification, Interface

Executive Summary

This deliverable describes and summarizes in details how key selected algorithms and technologies were both developed and integrated jointly into the different platforms and testbeds, provided within Workpackage (WP) 6.

Among all the concepts and algorithms developed within BeFEMTO, a particular attention was given to the demonstration of the following technologies within BeFEMTO various testbeds.

- In Testbed 1, the **SON principle** relying on graph colouring and dynamic frequency reuse for a ubiquitous access will be evaluated in real-time.
- In Testbed 2, **distributed routing and traffic offload** within an all-wireless network of femtocells will be investigated through the evaluation of dynamic backpressure routing algorithm. Testbed 2 will also demonstrate the benefit of the **Local Femto Gateway** concept introduced in BeFEMTO to deal with **traffic breakout, routing or control** within a network of femtocells using the Iuh-tap functionality (dissection of Iuh messages on the fly).
- In Testbed 3, **subscriber authentication** based on a removable UICC card will be investigated, allowing one user to have access wherever he wants to all its services/content delivered through different radio interfaces provided by the multi-radio FemtoNode.

To this end, this deliverable gives a clear view on the algorithms and the testbed set-up supporting the previous demo scenarios. Key building block integration and the interface finalization are addressed to ensure a consistent joint integration before the final validation phase.

Authors

Partner	Name	Phone / Fax / e-mail
CEA		
	Denis Dutoit	Phone : +33 4 38 78 01 63
		e-mail : denis.dutoit@cea.fr
	Sylvie Mayrargue	Phone : +33 4 38 78 62 42
		e-mail : sylvie.mayrargue@cea.fr
CTTC		
	Josep Mangues	Phone : +34 93 645 29 00
		Fax : +34 93 645 29 01
		e-mail : josep.mangues@cttc.cat
	Paolo Dini	e-mail : paolo.dini@cttc.cat
	Jaume Nin	e-mail : jaume.nin@cttc.cat
	José Núñez	e-mail : jose.nunez@cttc.cat
	Manuel Requena	e-mail : manuel.requena@cttc.cat
mimoOn		
	Dimitry Marandin	Phone : +49 (0)203 306 4537
		e-mail : dimitri.marandin@mimoon.de
	Manuel Palmowski	e-mail : manuel.palmowski@mimoon.de
NEC		
	John Fitzpatrick	Phone : +49 6221 4342 231
		e-mail : johnfitzpat@ieee.org
	Frank Zdarsky	Phone : +49 6221 4342 142
		e-mail : frank.zdarsky@neclab.eu
	Arne Felber	Phone : +49 6221 4342 159
		e-mail : arne.felber@neclab.eu
SC		
	Massinissa Lalam	Phone : +33 1 57 61 13 41
		Fax : +33 1 57 61 39 09
		e-mail : massinissa.lalam@sagemcom.com
	Frédéric Géhéniau	Phone : +33 1 57 61 16 79
		Fax : +33 1 57 61 39 09
		e-mail : frederic.geheniau@sagemcom.com
TID		
	Borja Iribarne	e-mail : e.ch1@tid.es
	Santiago Rouco	Phone : +34 91 312 95 75
		e-mail : srouco@tid.es
	Luis Cucala	Phone : +34 91 312 87 99
		e-mail : lcucala@tid.es
	Pablo Arozarena	Phone : +34 91 483 28 66
		e-mail : pabloa@tid.es
TTI		
	Mariano López	Phone : +34 94 229 12 12
		Fax : +34 94 227 01 39
		e-mail : mlopez@ttinorte.es
	Rocío García	Phone : +34 94 229 12 12
		Fax : +34 94 227 01 39
		e-mail : mrgarcia@ttinorte.es

Table of Contents

List of Figures

List of Tables

List of Acronyms and Abbreviations

Acronym	Definition
µs	micro-second
3G	3rd Generation
3GPP	3rd Generation Partnership Project
4G	4th Generation
ADC	Analog-Digital Converter
ADSL	Asymmetric Digital Subscriber Line
AGC	Automatic Gain Control
AKA	Authentication and Key Agreement
AM	Acknowledge Mode
AMBR	Aggregated Maximum Bit Rate
AMF	Access Management Function
API	Application Programming Interface
APN	Access Point Name
A-RACF	Access-Resource and Admission Control Function
ARF	Access Relay Function
ARQ	Automatic Repeat reQuest
BB	Baseband
BeFEMTO	Broadband evolved FEMTO networks
BER	Bit Error Rate
BGF	Border Gateway Service
BSSID	Basic Service Set Identifier
BW	Bandwidth
CBR	Constant Bit Rate
CFO	Carrier Frequency Offset
CLF	Connectivity session Location Function
CN	Core Network
CNG	Customer Network Gateway
CPE	Customer Premises Equipment
CPU	Central Processing Unit
CQI	Channel Quality Indicator
CRC	Cyclic Redundancy Check
CS	Circuit Switch
CSI	Channel State Information
CSMA/CA	Carrier Sense Multiple Access/Collision Avoidance
DAC	Digital-Analog Converter
dB	deciBel (referenced to watt)
dBc	deciBel (respect to carrier)
dBm	deciBel (referenced to milliwatt)
dBp-p	deciBel peak to peak
DL	Downlink
DMA	Direct Memory Access
DSL	Digital Subscriber Line
DSLAM	Digital Subscriber Line Access Multiplexer
DSP	Digital Signal Processor
EAP	Extensible Authentication Protocol
EMMA	EXTREME Measurement Architecture
EPC	Evolved Packet Core
Eth	Ethernet
ETSI	European Telecommunications Standards Institute
E-UTRA	Evolved Universal Terrestrial Radio Access
E-UTRAN	Evolved UTRAN
FAP	Femto Access Point
FDD	Frequency Division Duplex
FFT	Fast Fourier Transform
FIFO	First In First Out

FP	Framework Programme
FPGA	Field Programmable Gate Arrays
GB-DFR	Graph theory-Based Dynamic Frequency Reuse
GBR	Guaranteed Bit Rate
GHz	Giga Hertz
GND	Ground
GSM	Global System for Mobile Communications
GTP	GPRS Tunnelling Protocol
GTP-U	GPRS protocol tunnelling user plan
H(e)NB	Home (evolved) Node B
HAL	Hardware Abstraction Layer
HARQ	Hybrid ARQ
HMS	HNB Management System
HNBAP	Home Node B Application Part
HPA	High Power Amplifier
HSDPA	High Speed Downlink Packet Access
HSPA	High Speed Packet Access
HTTP	HyperText Transfer Protocol
HW	Hardware
Hz	Hertz
I/Q	In-phase/Quadrature
ICT	Information and Communications Technology
IEEE	Institute of Electrical and Electronics Engineers
IF	Intermediate Frequency
IFFT	Inverse FFT
IMSI	International Mobile Subscriber Identity
IO	Input Output
IP	Internet Protocol
KBB	Key Building Blocks
kHz	Kilohertz
KPI	Key Performance Indicator
LA	Link Adaptation
LAN	Local Area Network
LCH	Logical CHannel
LFGW	Local Femtocell GateWay
LNA	Low Noise Amplifier
LO	Local Oscillator
LTE	Long Term Evolution
LTE-A	Long Term Evolution - Advanced
MAC	Medium Access Control
MBR	Maximum Bit Rate
MCN	Mobile Core Network
MCS	Modulation and Coding Scheme
MGEN	Multi-Generator
MHz	Mega Hertz
MIMO	Multiple Input Multiple Output
MMSE	Minimum Mean Square Operator
mm	millimetre
MNO	Mobile Network Operator
MRFN	Multi-Radio FemtoNode
ms	millisecond
mV	milliVolt
NACF	Network Access Configuration Function
NAS	Non Access Stratum
NASS	Network Attachment Subsystem
NAT	Network Address Translation
NGN	Next Generation Networks
NoF	Network of Femtocells
ns	nanosecond

OAM	Operation And Maintenance
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency Division Multiple Access
OLT	Optical Line Termination
ONT	Optical Network Terminator
OP1dB	Output 1dB Compression Point
OS	Operating System
PC	Personal Computer
PDCP	Packet Data Convergence Protocol
PDU	Protocol Data Unit
PF	Proportional Fair
PHY	Physical Layer
PLL	Phase-Locked Loop
PoC	Proof of Concept
ppm	Part per million
PRB	Physical Resource Block
PS	Packet switched, Primary Subband
PUSCH	Physical Uplink Shared CHannel
QAM	Quadrature Amplitude Modulation
QCI	QoS Class Identifier
QoS	Quality of Service
QPSK	Quadrature Phase Shift Keying
RA	Resource Allocation
RAB	Radio Access Bearer
RACF	Resource Admission Control Function
RACH	Random Access Channel
RACS	Resource Admission Control Subsystem
RAM	Random Access Memory
RANAP	Radio Access Network Application Part
RAT	Radio Access Technology
RB	Resource Block
RCEF	Resource Control Enforcement Function
RF	Radio Frequency
RFID	Radio Frequency IDentification
RG	Residential Gateway
RSSI	Radio Signal Strength Indicator
RLC	Radio Link Control
ROHC	Robust Header Compression
RRC	Radio Resource Control
RRM	Radio Resource Management
RTP	Real-time Transport Protocol
RUA	RANAP user adaptation
Rx	Receiver
SCTP	Stream Control Transmission Protocol
SDU	Service Data Unit
SINR	Signal to Interference plus Noise Ratio
SON	Self-Organized Network
SPDF	Serving Policy Decision Function
SRAM	Static Random Access Memory
SS	Secondary Subband
SSL	Secure Socket Layer
SU-MIMO	Single User MIMO
SW	Software
TB	Transport Block
TBS	Transport Block Size
TCP/IP	Transmission Control Protocol / Internet Protocol
TDD	Time Division Duplex
TIMSI	Temporary IMSI
TISPAN	Telecommunications and Internet converged Services and Protocols for

	Advanced Networking
TTI	Transmission Time Interval
TTL	Transistor-Transistor Logic
Tx	Transmitter
UAAF	User Access Authorisation Function
UDP	User Data Protocol
UE	User Equipment
UICC	Universal Integrated Circuit Card
UL	Uplink
UMTS	Universal Mobile Telecommunications System
UP	User Plane
USB	Universal Serial Bus
UTRAN	UMTS Terrestrial Radio Access Network
V	Volt
VCO	Voltage Controlled Oscillator
VCXO	Voltage Controlled Crystal Oscillator
VoIP	Voice Over IP
Vpp	Volt peak to peak
Wi-Fi	Wireless Fidelity
WLAN	Wireless Local Area Network
WMR	Wireless Mesh Router
WP	Work Package
WPA	Wi-Fi Protected Access

1. Introduction

This deliverable presents the integration and interface specification of selected algorithms to be demonstrated through the various testbeds provided within WP6. These algorithms have been developed by the other BeFEMTO Work Packages (WP3, WP4, WP5) and close Work Package interaction was ensured to set-up a meaningful, and relevant down-selection process accordingly to both the real world limitations of the platforms and interest of algorithms. Each section of this deliverable deals with one specific testbed [1].

As a common basis for each section, the algorithms (down-selected within the WP6) are briefly recalled for each testbed with a description of the demo scenario envisaged at this stage of the project. Specific key performance indicators are given to assess the success of the demo. Key building blocks integration and interface specifications are given to ensure a coherent comprehension and integration of these algorithms amongst the involved partners.

More specifically:

- In Testbed 1, **self-organisation network (SON)** will be demonstrated using graph colouring associated with dynamic frequency reuse.
- In Testbed 2, **distributed routing algorithm** will be tested within a network of femtocells wirelessly connected to each other. The BeFEMTO **Local Femto Gateway Iuh-Tap capability** (dissection of Iuh messages on the fly) will also be evaluated.
- In Testbed 3, **authentication** of one femtocell subscriber will be implemented through the use of a removable UICC card to insert in order to allow an authentication procedure regardless of the geographical location.
- In Testbed 4, **fault diagnosis analysis** was supposed[1] to be evaluated in an enterprise networked femtocells context, using the BeFEMTO Local Femto Gateway to host diagnosis components.

[1] Due to internal reassignment which occurred during the second year of the project within one partner, this testbed had to be discontinued. Despite noticeable progress presented here, the content is still incomplete and should be regarded as work in progress.

2. Testbed1, Standalone LTE Radio

Testbed 1 is composed of the three (3) main parts of a femtocell that implement an air interface equivalent to the Uu interface [1]:

- the Radio Frequency (RF) Front End,
- the Physical Layer (PHY),
- the Protocol Stack (layer 2 and 3, which include scheduler and RRM).

Testbed 1 provides the possibility to show different LTE-A radio technology algorithms; its architecture is depicted in Figure 2-1.

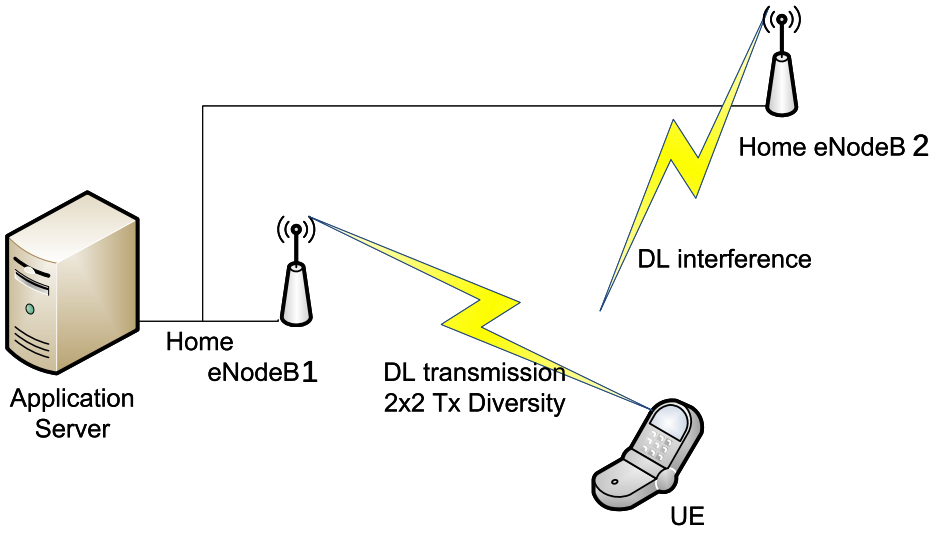

Figure 2-1: Testbed1 architecture

The algorithm down-selected by Testbed 1 partners will be first presented. This algorithm was proposed in the context of [3]. The reason for this down-selection is given in Appendix A, together with the list of some algorithms derived from the work of WP3/WP4. Description of the main testbed building blocks and their interfaces will then be given. Finally, some milestones will be presented through Testbed 1 work plan ensuring a synchronisation between the different partners.

2.1 Extended Graph-Based Dynamic Frequency Reuse

2.1.1 Short Description

In this scheme developed in [2] and [3], graph theory is used to mitigate interference by performing a dynamic partitioning of the frequency resources, here subbands. Indeed, using graph theory terminology, HeNBs are identified to "nodes" while "edges" connect HeNBs that interfere with each other. By assuming each colour is a different subband, a graph colouring algorithm will assign subbands so that two HeNBs connected with edges should not use the same subband. This straightforward application of graph colouring leads to a uniform partitioning of resources, where each HeNB receives the same number of subbands.

This is suboptimal when considering that all HeNBs do not experience the same interference conditions. HeNB facing less interference could be assigned more subbands and vice-versa, which would lead to an increase in the network resource efficiency. Such an algorithm, coined "Graph theory-based dynamic frequency reuse" (GB-DFR) has been presented in [2]. In [3] GB-DFR has been further enhanced with the objective of increasing the cell-edge capacity whilst maintaining high subband utilization. Two classes of subbands have been defined, primary subbands (PSs) and secondary subbands (SSs). PSs are assigned by a central controller similar to GB-DFR to protect cell-edge UEs facing high interference.

However, according to their location, or equivalently their perceived interference condition, users may require less protection, which is typically the case of .cell centre users. This situation can be exploited to increase frequency reuse. Indeed, a HeNB whose PS is dedicated to central users only can release the

constraints on this PS, by allowing neighbouring cells to use it. In this case, the neighbouring HeNB can only use this additional subband for its own central users, since a cell edge user would be highly interfered by the PS of the original cell. In this case, the additional subband is "secondary" in the sense that in contrast to PSs, it cannot be used at cell-edges. Consequently, the usage of the PSs boosts cell-edge capacity, whereas the SSs increase the spatial reuse of resources especially for multi-user deployments

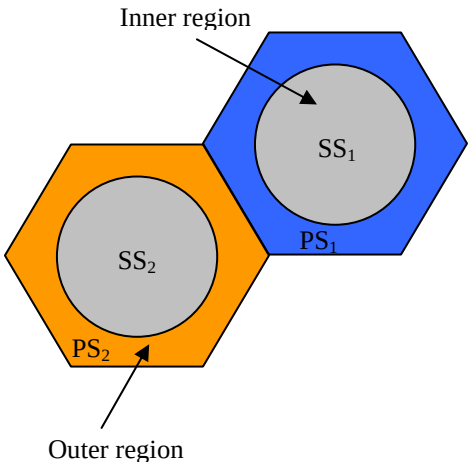

Figure 2-2: PS (resp. SS) allocation in outer (resp. inner) part of a cell

2.1.2 Demo Scenario and KPIs

- Testbed1 is composed of two HeNBs and one UE, as indicated in Figure 2-1, the UE being connected to one of the HeNB, while the second HeNB creates interference. It is assumed that GB-DFR has been run offline, leading to assigning PS1 to HeNB1 and PS2 to HeNB2.
- At the beginning of the communication, UE is at cell edge of its serving cell HeNB1, thus highly interfered by HeNB2. The fact that the UE is located at cell-edge is assessed by measuring PS1 and PS2 SINRs (SINR1 and SINR2 respectively) and comparing them with predefined thresholds. SINR1 is high enough to ensure requested QoS, but lower than the threshold which defines the border between cell-edge and cell centre. SINR2 is lower than the threshold for QoS guarantee.
- Then, the UE moves towards cell centre, as shown in Figure 2-3. The fact that the UE belongs to cell centre is assessed by SINR1, which increases above a certain threshold. When detecting such a situation, HeNB1 assigns subband 2 to the UE as SS. Success is assessed by ensuring UE QoS is maintained.
- **KPIs:** In practice, we will run a video, and it will be shown that the quality is good with both subbands assignments. In addition, we will display the modulation constellation in both cases. We will measure the bit error rate (BER) and throughput.

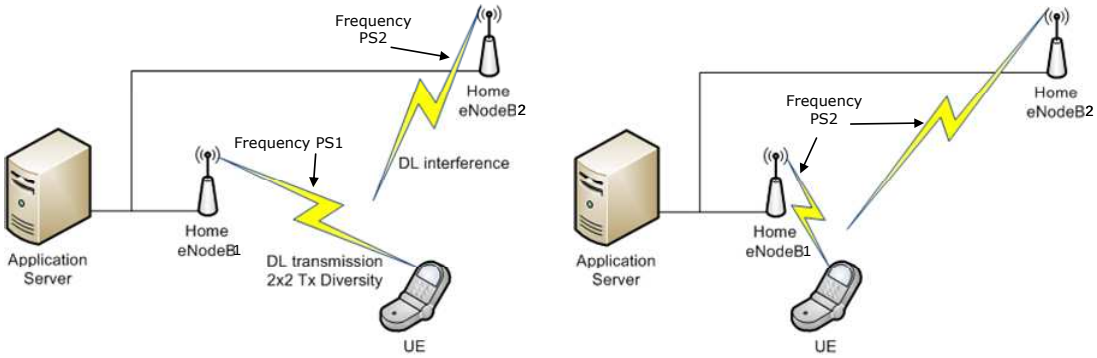

Figure 2-3 FUE served by HeNB1 moves from cell edge (left), to cell centre (right)

2.2 Key Building Blocks and Integration Specification

2.2.1 RF Front-end

2.2.1.1 Short Description

The RF front-end block is composed in Testbed 1 by two transmitters, for (H)eNB1 and (H)eNB2, and one receiver, for UE. These blocks receive as an input the data sent by the Magali platform (transmitter case) and deliver the data sent over the air to the Magali receiver (receiver case). The operation frequency band of this part is from 2620 to 2690MHz (DL of Band 7).

2.2.1.2 Integration

A short overview of transmitter and receiver is given below, including their main and features.

2.2.1.2.1 Transmitter

The RF front-end transmitter is described in the next block diagram.

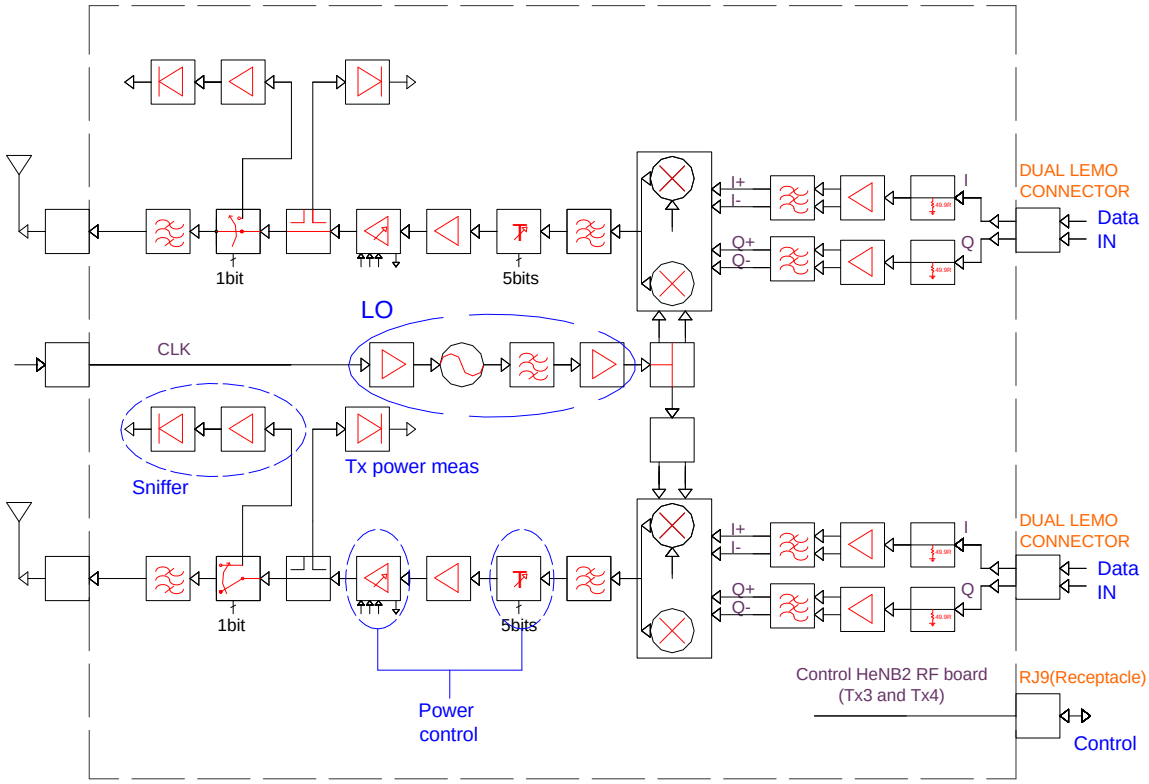

Figure 2-4: Transmitter block diagram

The transmitter takes the input signal (data) from Magali platform and converts it to an RF signal, tuned in 2620-2690 MHz band, by means of a synthesized local oscillator in the same band (direct conversion is employed). In each HeNB, the transmitter is duplicated in order to enable 2x2 MIMO scheme. Its architecture conception has taken into account some functions enabling the interference algorithms.

These are:

- **Power control**: by introducing one digital attenuator which has 31dB attenuation range (1dB step) and one variable gain amplifier (with 20dB of range).
- **RSSI sniffer**: for measuring the surrounding environment. This includes a switch (for switching the Tx to Rx mode), an amplifier stage and one power detector.
- **Tx power measurement**: in order to monitor the transmitted output power. It is composed by a coupler, and a power detector.
- **LO frequency**: by means of the synthesized local oscillator which allows the frequency change within 2620-2690MHz band in 300 kHz steps.

All of these parameters (and the LO frequency) are controlled by the Magali platform.

A control card is included in the transmitter chain, in order to translate the commands of Magali to the RF stages. The control parameters in this chain are summarised in Table 2-1 below.

Table 2-1: HeNB Transmitter control parameters

HeNB Tx					
PARAMETER NAME	**FROM**	**TO**	**DESCRIPTION**	**TYPE**	**RF INVOLVED**
HeNB-mode	Magali Board	RF Board (HeNB)	HeNB mode of operation: "*Normal transmission* (default)" or "*Sniffer mode*"	Digital TTL command	**Switch** Vctrl = +5.0V/0V (Normal Tx mode/Sniffer mode) **HPA** Normal Tx mode (high gain)/Sniffer mode (shutdown)
PotTx-HeNB	Magali Board	RF Board (HeNB)	Desired Transmitter Power of the HeNB	Digital TTL command	**Digital Attenuator** 0 to 31dB of attenuation (1dB steps)→ 5bits **HPA** 0 to 20dB attenuation→1bit (*High gain mode/bypass mode*)
Meas-PotTx-HeNB	RF Board	Magali Board	Measured transmitted power of HeNB2	Digital TTL command	**Tx power Detector**
Meas-sniffer-HeNB	RF Board	Magali Board	Measured sniffer power received in HeNB	Digital TTL command	**Sniffer power detector**
LO-frequency	Magali Board	RF Board (HeNB)	Local Oscillator Frequency HeNB	Digital TTL command	**PLL frequency synthesizer**

Main features of each individual transmitter chain are summarized in Table 2-2 below.

Table 2-2: Transmitter characteristics

Parameter	Value
Output Frequency	2620-2690MHz
OP1dB	+24.5dBm
Gain (dB)	15.5dB
Nominal Output Power	+7dBm
Attenuation Range (Tx Power control)	From 0 to 51dB: from 0 to 31dB (1dB step) from 31 to 51dB (20dB step)
OL Phase Noise	-65dBc/Hz @ 10Hz -77dBc/Hz @ 100Hz -77dBc/Hz @ 1kHz -102dBc/Hz @ 10kHz -121dBc/Hz @ 100kHz
Sniffer Sensibility	-71dBm to -17dBm

2.2.1.2.2 Receiver

The scheme followed by the receiver for implementing the UE is depicted below in the Figure 2-5 .

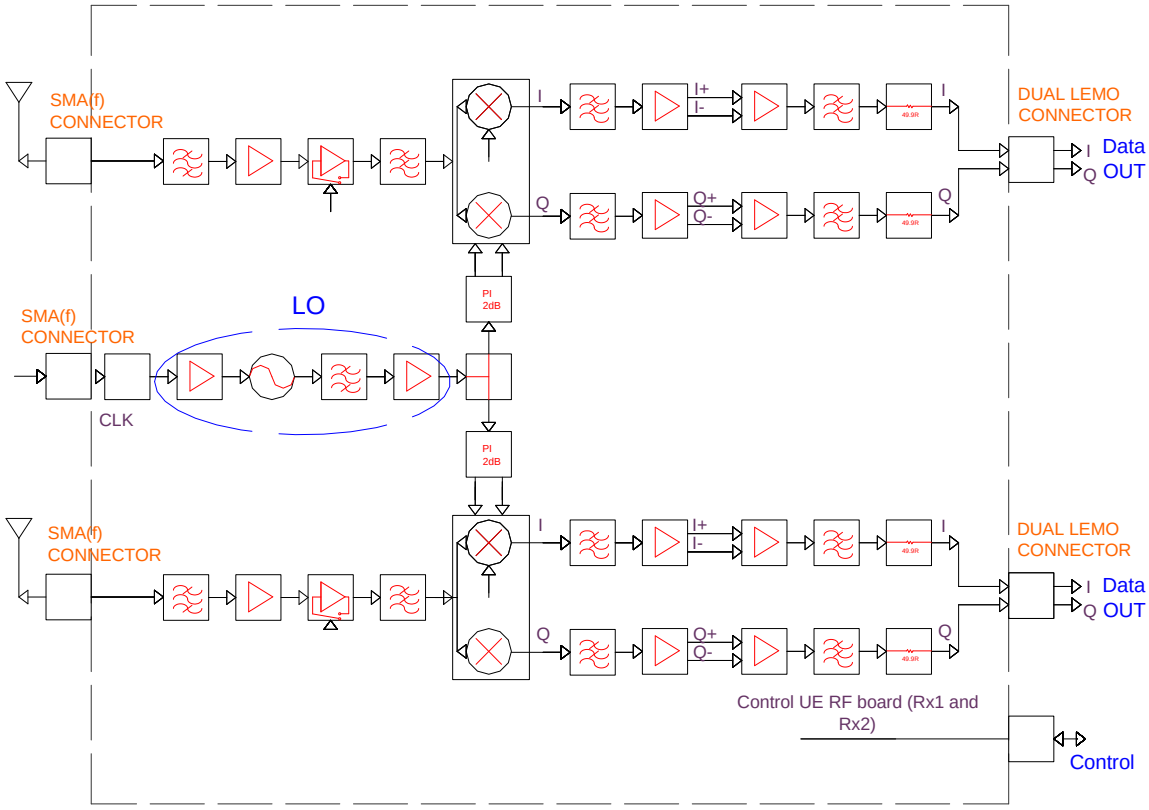

Figure 2-5: Receiver block diagram

This module takes the signal which is received over the air and converts it to baseband (using the same local oscillator scheme than in the transmitter case) and delivers it to Magali. Like in the transmitter case, two identical chains are developed in order to enable 2x2 MIMO scheme. It also incorporates a control board, which translates the commands of Magali platform for the RF stages. In this case, the parameter to control in the receiver is the input frequency, through a synthesized local oscillator.

Table 2-3 shows a summary with the parameters to control in the receiver part of the RF front-end.

Table 2-3: Control parameters of UE Receiver

UE Receiver					
PARAMETER NAME	**FROM**	**TO**	**DESCRIPTION**	**TYPE**	**RF INVOLVED**
Byp-LNA2-UE	Magali Board	RF Board	LNA “Normal operation (default)” or “Bypass mode”	Digital TTL command	**LNA2** Vcrtl = +5.0V/0V (normal operation/bypass mode)
LO-freq-UE	Magali Board	RF Board	Local Oscillator Frequency UE	Digital TTL command	**PLL frequency synthesizer**

Finally, main features of the UE receiver are shown in the Table 2-4.

Table 2-4: Receiver main features

Parameter	Value
Noise Figure	3.1dB
Sensibility @ BW=20MHz, SNR=0dB	-97dBm
LO Phase Noise	-65dBc/Hz @ 10Hz -77dBc/Hz @ 100Hz -77dBc/Hz @ 1kHz -102dBc/Hz @ 10kHz -121dBc/Hz @ 100kHz
Gain (dB)	28dB (RF fixed) + 39.5dB to -30.0dB (demodulator AGC) + 7.2V/V BB amplification

2.2.1.3 Interfaces

The RF front-end interacts with PHY (Magali Platform) which controls its main parameters in order to implement the above presented algorithms. Figure 2-6 shows a block diagram with the main connections between these two platforms.

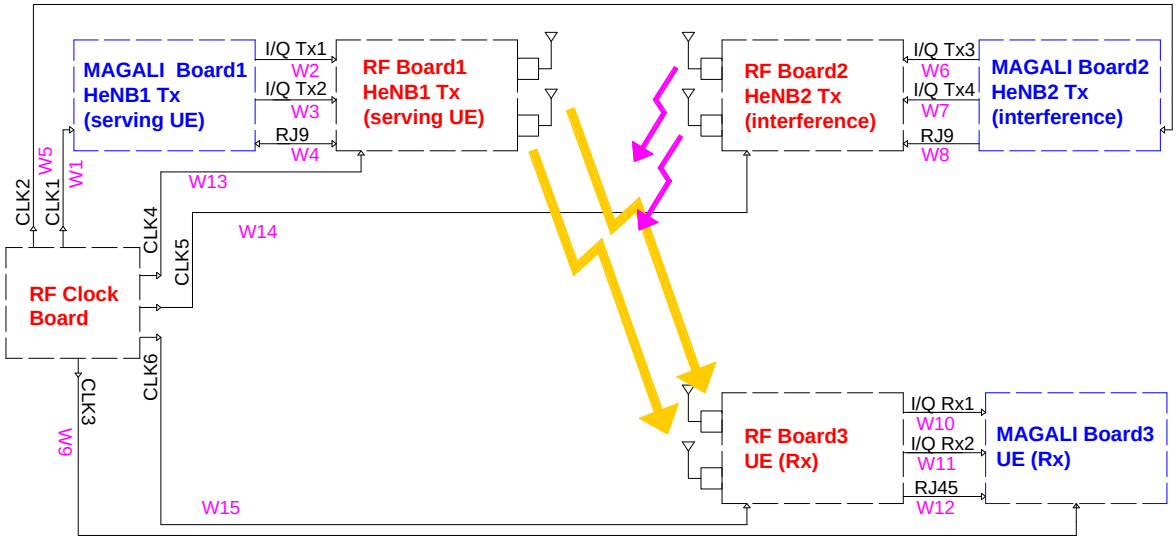

Figure 2-6: Scheme of connections between RF front-end and PHY

A more detailed description is given in the following tables (Table 2-5, Table 2-6, Table 2-7), which describe all the connections between RF front-end and Magali platform for (H)eNB1, (H)eNB2 and UE, respectively.

Table 2-5: HeNB1 interfaces description

HeNB1					
NAME	**FROM**	**TO**	**DESCRIPTION**	**TYPE**	**NATURE AND RANGE**
W1	RF clock board	Magali Board1 (HeNB1: Tx1 and Tx2)	HeNB1-CLK1 CABLE	Dual LEMO (m) - Dual LEMO (m)	Pin A: 10MHz, +0dBm +-3dB Pin B: No signal
W2	Magali Board1 (HeNB1: Tx1 and Tx2)	RF Board1 (HeNB1 serving UE)	HeNB1-TX1 CABLE	Dual LEMO (m) - Dual LEMO (m)	Pin A: 500mVpp (default) Pin B: 500mVpp (default)
W3	Magali Board1 (HeNB1: Tx1 and Tx2)	RF Board1 (HeNB1 serving UE)	HeNB1-TX2 CABLE	Dual LEMO (m) - Dual LEMO (m)	Pin A: 500mVpp (default) Pin B: 500mVpp (default)
W4	Magali Board1 (HeNB1: Tx1 and Tx2)	RF Board1 (HeNB1 serving UE)	HeNB1-RJ9 CABLE	4P4C(plug) - 4P4C(plug)	Pin 1: RS232 out BB - in RF RS232 standard (+/-4V) Pin 2: GND Pin 3: RS232 out RF - in BB RS232 standard (+/-4V) Pin 4: GND
W13	RF clock board	RF Board1 (HeNB1 serving UE)	HeNB1-CLK4 CABLE	SMA(m) - SMA (m)	Reference (10MHz) +3dBm to +4dBm

Table 2-6: HeNB2 interfaces description

HeNB2					
NAME	**FROM**	**TO**	**DESCRIPTION**	**TYPE**	**NATURE AND RANGE**
W5	RF clock board	Magali Board2 (HeNB2: Tx3 and Tx4)	HeNB2-CLK2 CABLE	Dual LEMO (m) - Dual LEMO (m)	Pin A: 10MHz, +0dBm +-3dB Pin B: No signal
W6	Magali Board2 (HeNB2: Tx3 and Tx4)	RF Board2 (HeNB2 interference)	HeNB2-TX3 CABLE	Dual LEMO (m) - Dual LEMO (m)	Pin A: 500mVpp (default) Pin B: 500mVpp (default)
W7	Magali Board2 (HeNB2: Tx3 and Tx4)	RF Board2 (HeNB2 interference)	HeNB2-TX4 CABLE	Dual LEMO (m) - Dual LEMO (m)	Pin A: 500mVpp (default) Pin B: 500mVpp (default)
W8	Magali Board2 (HeNB2: Tx3 and Tx4)	RF Board2 (HeNB2 interference)	HeNB2-RJ9 CABLE	4P4C(plug) - 4P4C(plug)	Pin 1: RS232 out BB - in RF RS232 standard (+/-4V) Pin 2: GND Pin 3: RS232 out RF - in BB RS232 standard (+/-4V) Pin 4: GND
W14	RF clock board	RF Board2 (HeNB2 interference)	HeNB2-CLK5 CABLE	SMA(m) - SMA (m)	Reference (10MHz) +3dBm to +4dBm

Table 2-7: UE interfaces description

UE					
NAME	**FROM**	**TO**	**DESCRIPTION**	**TYPE**	**NATURE AND RANGE**
W9	RF clock board	Magali Board3 (UE: Rx1 and Rx2)	UE-CLK3 CABLE	Dual LEMO (m) - Dual LEMO (m)	Pin A: No signal Pin B: 10MHz, +0dBm +-3dB
W10	RF Board3 (UE Rx)	Magali Board3 (UE: Rx1 and Rx2)	UE-RX1 CABLE	Dual LEMO (m) - Dual LEMO (m)	Pin A: +1,3Vpp (+650mV / -650 mV). Pin B: +1,3Vpp (+650mV / -650 mV).
W11	RF Board3 (UE Rx)	Magali Board3 (UE: Rx1 and Rx2)	UE-RX2 CABLE	Dual LEMO (m) - Dual LEMO (m)	Pin A: +1,3Vpp (+650mV / -650 mV). Pin B: +1,3Vpp (+650mV / -650 mV).
W12	Magali Board3 (UE: Rx1 and Rx2)	RF Board3 (UE Rx)	UE-RJ45 CABLE	8P8C(plug) - 8P8C(plug)	Pin 1: RS232 out BB - in RF RS232 standard (+/-4V) Pin 2: GND Pin 3: RS232 out RF - in BB RS232 standard (+/-4V) Pin 4: GND Pin 5: Not used Pin 6: GND Pin 7: Not used Pin 8: GND
W15	RF clock board	RF Board3 (UE Rx)	UE-CLK6 CABLE	SMA(m) - SMA (m)	Reference (10MHz) +3dBm to +4dBm

2.2.2 PHY layer

Testbed1 PHY layer is composed of two transmitters, for (H)eNB1 and (H)eNB2, and one receiver, for UE. The receiver block implemented on a Magali platform receives as an input the data from the RF front-end and the transmitter block implemented on another Magali platform delivers the data to the RF front-end to be sent over the air.

Only downlink scheme is implemented over the air. The PHY blocks incorporate several operating modes as well as Channel State Information (CSI) estimation and feedback which enable the algorithm described in section 2.1.

2.2.2.1 Downlink Technical Specifications

The downlink transmission scheme implemented on Magali platform is based on OFDMA access technology and uses FDD duplex mode (uplink and downlink are operated on two different frequency carriers). As described in 3GPP [4] the frame structure is made of 20 timeslots; each of the 20 slots is composed of 7 consecutive OFDM symbols. Each user uses 2 consecutive slots, equivalent to 1 Transmission Time Interval (TTI), to transmit its data. Each slot pair associated to one TTI is called a sub-frame.

A first difference with 3GPP appears in the frame structure used in the Magali platform as illustrated in Figure 2-7. Indeed, 2 extra symbols are inserted in this frame for cell search and synchronization

processes of mobile terminals. These symbols are positioned in the middle of the frame (after 10 slots) and at the very end of the frame.

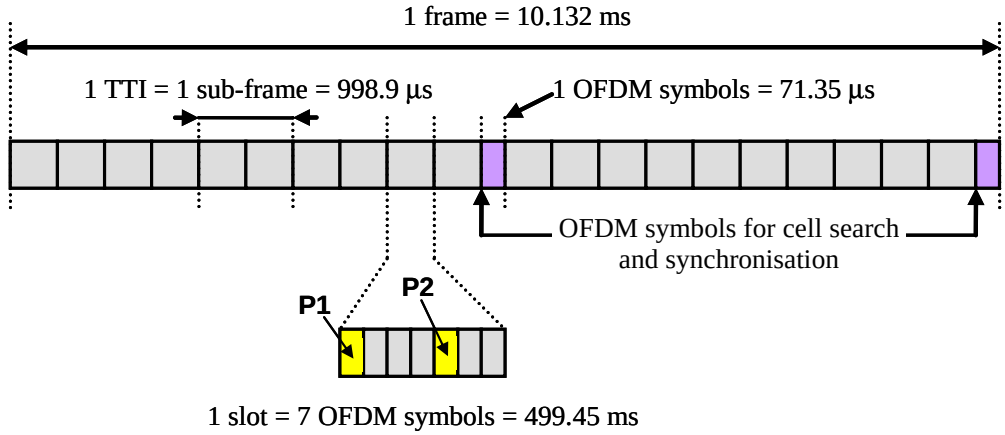

Figure 2-7: Downlink prototype frame structure

This downlink transmission scheme is based on OFDM modulation with 15 kHz sub-carrier spacing for all bandwidths. The various channel bandwidths are obtained with varying the number of sub-carriers. In order to cope with channel time dispersion, a cyclic prefix is inserted at the beginning of each OFDM symbol.

A second difference with 3GPP concerns cyclic prefix duration. In 3GPP, it is specified that 6 out of 7 symbols have a cyclic prefix duration equal to 4.69 µs, while the 7-th symbol cyclic prefix duration equal to 5.61 µs. For sake of simplification, all symbols have here the same cyclic prefix duration, namely 4.69 µs. As a result, a sub-frame lasts 998.9 µs instead of 1 ms as in 3GPP, while a frame lasts 10,132 ms instead of 10 ms as in 3GPP.

Table 2-8 lists the downlink prototype system parameters. Table 2-9 and Table 2-10 give more details on system parameters associated to the two channel bandwidths selected for the prototype: 10 MHz for pilots and data OFDM symbols. Table 2-11 gives the reachable data rates of downlink prototype operating modes. Each mode corresponds to a different number of Physical Resource Blocks (PRB). A PRB is composed of 12 adjacent subcarriers for one slot duration [1].

Table 2-8: Downlink prototype common system parameters

OFDM symbol duration (useful)	T	66.66 µs
Cyclic prefix duration	T_{IG}	4.69 µs
Total OFDM symbol duration	T_0	71.35 µs
Modulation speed	$1/T_0$	14.014 kHz
Inter-carrier spacing	$1/T$	15 kHz
Slot duration	T_{slot}	0.499 ms
Slot rhythm	$1/T_{slot}$	2 kHz
Number of OFDM symbols per slot	$N_{symbols}$	7
Frame duration	T_{frame}	9.989 ms
Frame rhythm	$1/T_{frame}$	100.11 Hz
Number of slots per frame	N_{slot}	20

Table 2-9: System parameters for 1.25 MHz channel bandwidth (pilot and data symbols)

Channel bandwidth	B_C	1.25 MHz
Sampling clock	T_S	0.52 µs
Sampling frequency	$1/T_S$	1.92 MHz
Number of samples per sub-frame	N_S	959
Frame size	N_{FFT}	128
Guard interval size	N_{IG}	9
OFDM symbol size	N_0	137
Number of sub-carriers null / modulated	N_C	1 / 72

Occupied bandwidth	B_0	1.095 MHz

Table 2-10: System parameters for 10 MHz channel bandwidth (pilot and data symbols)

Channel bandwidth	B_C	10 MHz
Sampling clock	T_S	65.1 ns
Sampling frequency	$1/ T_S$	15.36 MHz
Number of samples per sub-frame	N_S	7672
Frame size	N_{FFT}	1024
Guard interval size	N_{IG}	72
OFDM symbol size	N_0	1096
Number of sub-carriers null / modulated	N_C	1 / 600
Occupied bandwidth	B_0	9.105 MHz

Table 2-11: System capacity of DL operating modes

Modes	**DL capacity per UE**
(1) QPSK, 1/2, 1 PRB B_{info} = 120 bits	0.12 Mbits/s
(3) 16-QAM, 2/3, 2 PRBs B_{info} = 640 bits	0.64 Mbits/s
(5) 64-QAM, 3/4, 10 PRBs B_{info} = 5400 bits	5.4 Mbits/s

For this downlink prototype, a 2x2 MIMO scheme based on Alamouti transmit diversity has been chosen, which means that one codeword is transmitted on two antennas. This scheme has been identified as a relevant scenario that increases the robustness of the transmission by making use of the MIMO configuration, while allowing for low implementation complexity.

2.2.2.2 Platform Hardware Architecture

As shown in Figure 2-8, the baseband platform is basically composed of three main parts: the Magali chip, a Xilinx Virtex-5 FPGA and versatile interfaces. The Magali chip is used to speed up computations of high-performance functions and to support basic functions of a signal processing chain. The FPGA is used for advanced functions which are not supported by the Magali chip. Three different output interfaces are available on the board. A host interface proposes several different links with a PC, a simple RS232 link for FPGA control, a USB interface and two Ethernet interfaces (one at 1 Gbits/s and the other one at 100 Mbits/s) for board configuration and debug/monitoring, and. The Rocket IO interface can be used for connection of the board with another high-speed board. RF interfaces are also available.

2.2.2.3 Tx Interface with RF front-end

The Magali board includes two dual DAC AD9745 that convert the Tx I/Q 12-bit data (one converter per antenna) to analogue signals. These I/Q analogue signals are available on dual Lemo connectors to be used by the TX RF board.

The Magali board receives the reference clock 10 MHz from the Tx RF clock board through the input n°1 of the dual Lemo connector reserved for clocks. The 122.88 MHz VCXO of Magali board is enslaved to this external 10 MHz reference clock. The 122.88 MHz clock is divided by 3072 to obtain the first 40 kHz clock. The 10 MHz clock is divided by 250 to obtain the second 40 kHz clock. These two 40 kHz clocks, common divider between the two initial clocks, are phased together using the FPGA resources (phase comparator) and an operational amplifier (loop filter).

The control of Tx RF board is done by the FPGA through a serial RS-232 link. On Magali board, the RS-232 line is controlled by the RS-232 transceiver MAX3380 which is connected to a RJ9 female connector.

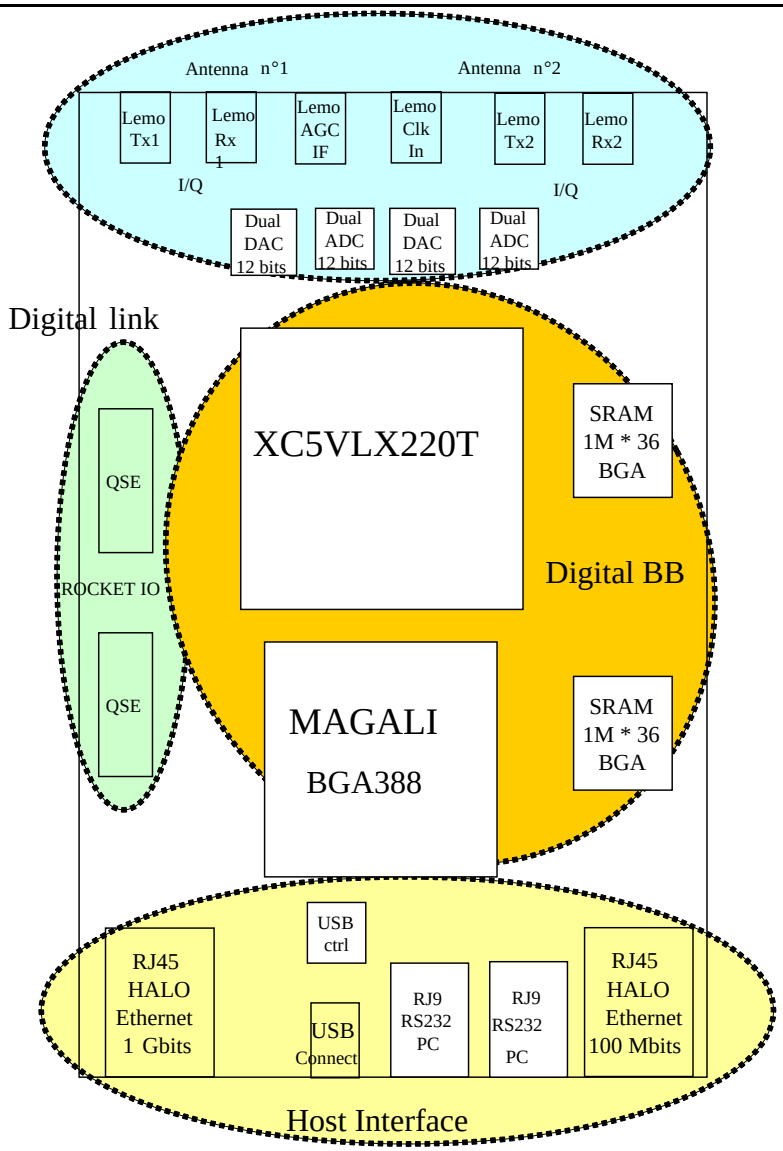

Figure 2-8: Simplified view of the baseband platform

2.2.2.4 Rx Interface with RF front-end

The interface with the Rx RF board is done through two female Lemo connectors. The analogue low-IF (38.4MHz) data provided by the Rx RF board are converted to digital 12 bits signals by two AD9627 ADCs. The Magali board includes two dual AD9627ADCs and two dual Lemo connectors which can be used to receive I/Q analogue signals instead of the low-IF signal.

Similarly to the interface with the Tx RF board, the Magali board receives the reference clock 10 MHz from the RF clock board through the input n°2 of the dual Lemo connector.

The control of Rx RF board is done by the Virtex-5 FPGA through a serial RS232 link. On Magali board, the RS232 lines are controlled by the RS232 transceiver MAX3380, which is connected to a RJ45 female connector. The automatic gain control of the RF board is done by a quad serial DAC AD5318 controlled by the FPGA. The RF AGCs of both channels (RF1_AGC and RF2_AGC) are provided to the Rx RF board, through the RJ45 female connector. The IF AGCs of both channels (IF1_AGC and IF2_AGC) are provided to the Rx RF board, through a dual female Lemo connector.

2.2.2.5 Downlink Receiver Description

The receiver has been implemented on Magali board with several operating modes. Testbed 1 implementation is based on a 2x2 MIMO scheme based on Alamouti transmit diversity. The downlink receiver chain is described in Figure 2-9.

After OFDM demodulation, information and reference symbols are separated on each antenna branch for further processing. The estimation of the residual Carrier Frequency Offset (CFO) is based on the complex correlation between the replicas of a same pilot symbol for each receive antenna in the frequency domain. Both data and pilot symbols are compensated in frequency prior to channel estimation. Reference symbols are fed through the MIMO channel estimation module which correlates the observed pilot symbols with the reference sequences. A Wiener filter is applied in the frequency domain for each Tx/Rx pair in order to interpolate the channel coefficients over the whole bandwidth.

To further improve performance, channel coefficients are linearly interpolated in the time domain between each pair of OFDM reference symbols. The two streams of information symbols and the corresponding channel coefficients are fed through the MIMO Minimum Mean Square Error (MMSE) decoder that produces one SISO data stream.

After MIMO decoding, the receiver implements the dual functions of the transmitter (see below). In a first step, data is processed by a soft M-ary symbol de-mapping module that produces a metric associated to each bit carried by the M-ary symbols. The flow of soft bits is then fed through a Turbo decoder. Decoded bits are finally de-scrambled and sent to the MAC layer.

The receiver has been implemented on Magali board with several operating modes. Testbed1 implementation is based on a 2x2 MIMO scheme based on Alamouti transmit diversity.

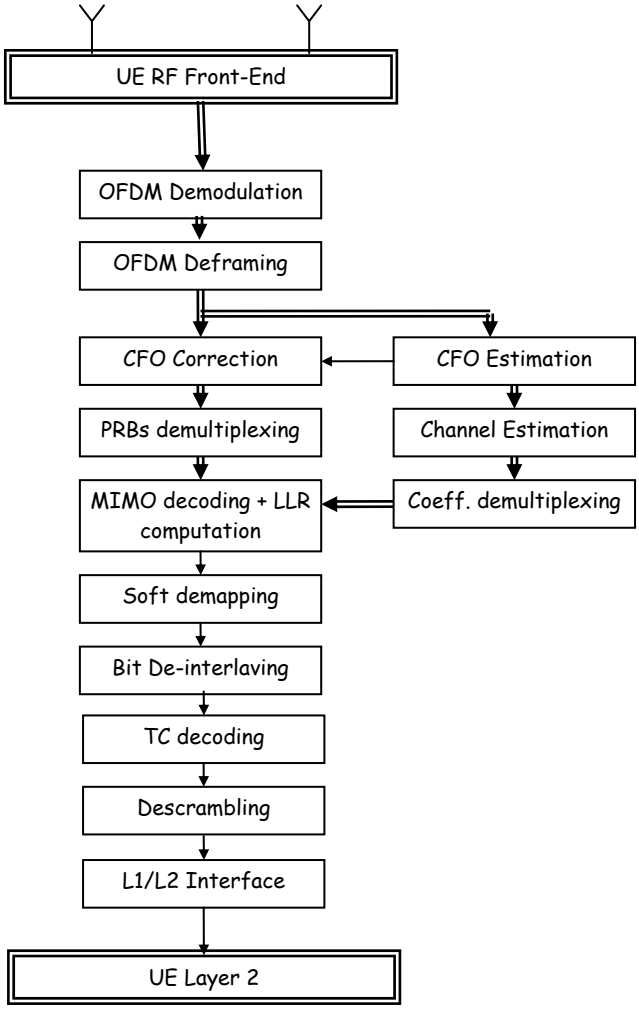

Figure 2-9: Functional description of the DL receiver

The main risk for this implementation step is the lack of hardware maturity encountered in the chip, which required software workarounds. One impact of these workarounds is the lack of real time processing due to some readjustment in the sequence of the signal processing steps. Performance measurements have shown that the non-optimized porting of the full application requires 13ms per TTI processing instead of the targeted real-time processing in 1ms. Following some analysis, we have seen that we can reach at best 3ms per TTI. But the 3ms processing time would have required too many runtime software optimizations, which is not in line with the testbed objective. We have decided to

process one TTI in 10 ms which represents a good trade-off between needed performances for the demonstrator and the engineering time needed to optimize the runtime software.

2.2.2.6 Downlink Transmitter Description

The downlink transmitter is described in Figure 2-10. Upon reception of the Transport Block (TB) from the MAC layer, TB are segmented if needed into a number of equal length blocks with size equal to one of the basic Transport Block Size (TBS). Information bits are then scrambled with a pseudo-random sequence to avoid detrimental effect of long sequences '0' or '1' potentially produced by the source code. In the prototype, blocks of information bits are encoded using a duo binary Turbo Code with mother code rate equal to 1/2.

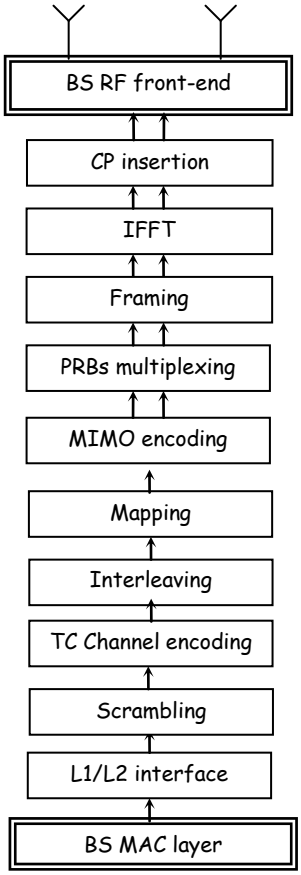

Figure 2-10: Functional description of the DL transmitter

The demonstrator supports three basic target code rates: 1/2, 2/3 and 3/4. The desired channel coding rate is obtained by puncturing. After channel encoding, channel interleaving is performed over the entire sub-frame to benefit from time and frequency diversity gain at the channel decoding level. The resulting bits are converted into M-ary symbols according to the modulation order (mapping). To simplify the implementation, the prototype supports 3 MCSs between the robust (QPSK, 1/2) mode to the spectrally efficient (64QAM, 3/4) mode with the intermediate (16QAM, 2/3) mode in-between.

The M-ary symbols resulting of the mapping are then fed through the MIMO encoder (Alamouti scheme). The transmitter then performs the so-called PRB mapping that allocates the MIMO encoded symbols to the resource blocks (RBs) defined by the time/frequency grid on each sub-frame. The framing module then multiplexes reference symbols with data. It also inserts for each of the OFDMA symbols the null values that modulate sub-carriers at the edges. The result of the framing is processed by the OFDM modulation module to form the sampled version of each OFDM symbol. For each antenna, this module performs an Inverse Fast Fourier Transform (IFFT) and the guard interval (the cyclic prefix) insertion.

2.2.3 Protocol Stack

2.2.3.1 Description

The (H)eNB protocol stack is a LTE FDD Release 8 compliant protocol stack [5]-[8], consisting of MAC, RLC, PDCP, RRC and common services and support functions as shown in Figure 2-11. For the

integration within BeFEMTO, a scheduler is provided that is integrated into the stack according to the agreed FemtoForum Scheduler API. The architecture of the protocol stack is based on the layering concept, where each layer provides services to its upper layer through well defined interfaces. In addition a UE protocol stack is used as a counterpart to the eNB protocol stack (which will not be described here in more detail). The functionality of each module shown in Figure 2-11 is described below.

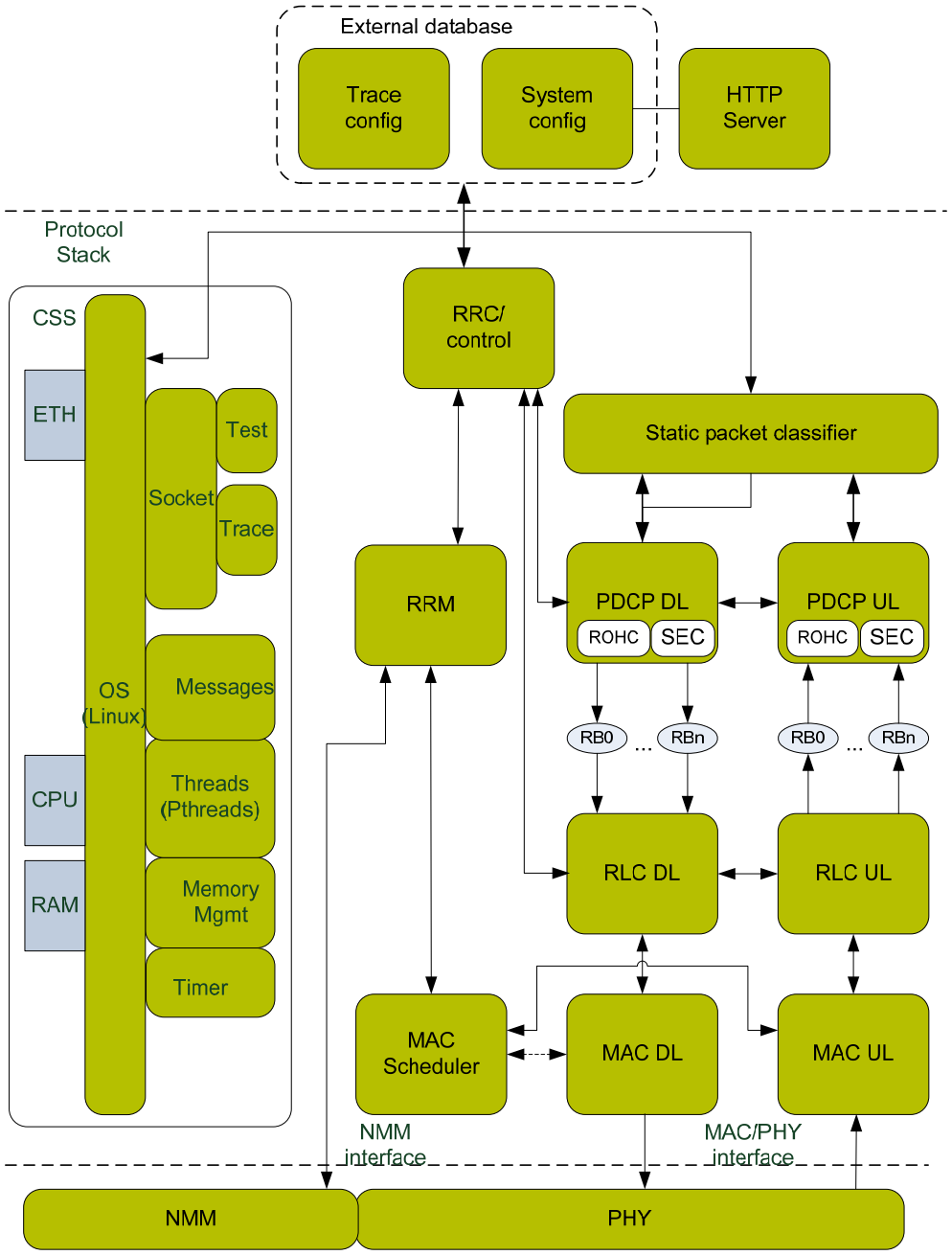

Figure 2-11: Protocol stack overview

- **Radio Resource Control/Radio Resource Mangagement (RRC/RRM)**
  - Broadcast of system information
  - Paging
  - Admission and Load Control
  - Load Balancing
  - Connection Mobility (Handover)
    - UE measurement reporting and control of the reporting for inter-cell and inter-RAT mobility
    - UE Context Transfer
  - RRC Connection
  - (Signalling) Radio Bearer control (setup/ release/ maintenance)

- PDCP, RLC, MAC configuration
- Measurement Configuration
- Encoding and Decodes of RRC messages
- Security functions including key management
- QoS management functions
- UE measurement reporting and control of the reporting
- NAS direct message transfer to/from NAS from/to UE
- Allocation of temporary identifiers between UE and E-UTRAN

- **Packet Data Convergence Protocol (PDCP)**
  - Maintenance of Sequence Numbers
  - Header (De-)Compression (ROHC)
  - Integrity/Ciphering
  - Transfer of upper layer data from/to RRC and S-GW to/from the RLC layer
  - SDU enqueuing
  - QoS queue management
  - Timer-based SDU Discard
  - Handover
    - In-sequence delivery of upper layer PDUs for RLC AM
    - Duplicate detection of lower layer SDUs for RLC AM
    - Retransmission of PDCP SDUs for RLC AM

- **Radio Link Control (RLC)**
  - Data transfer services to/from the upper layers in the acknowledged (AM), unacknowledged, and transparent modes
  - For AM, Automatic Repeat Request (ARQ) is used for retransmissions and error correction
  - In sequence delivery of upper layer PDUs except at handover
  - Duplicate detection
  - Segmentation and concatenation of SDUs. RLC uses dynamic PDU size to build each PDU according to the requested size
  - Each PDU can have multiple SDUs and segmentation of SDUs is supported for an optimal fit without padding
  - Re-segmentation of PDUs which need to be retransmitted
  - Protocol error detection and recovery
  - RLC re-establishment
  - SDU Reassembly and PDU Reordering of uplink PDUs

- **Medium Access Control (MAC)**
  - Mapping between logical channels and transport channels
  - Multiplexing/de-multiplexing of RLC PDUs belonging to one or different radio bearers into/from transport blocks (TB) delivered to/from the physical layer on transport channels
  - Traffic volume measurement reporting
  - Error correction through HARQ
  - Padding

- **MAC Scheduler**
  - Dynamic scheduling of UEs according to QoS parameters, HARQ and channel measurements
  - Priority handling between logical channels of a single UE
  - Transport format selection, MIMO mode selection

- **Packet Classifier**
  - Maps IP packets to/from bearer to/from IP address and port

- **Database**
  - Contains the system configuration, which is used by RRC to populate the system information messages and the radio bearer setup messages

Framework Blocks hide everything which is not related to the logical (protocol stack) view and consists of an OS abstraction layer and additional functionalities used by all layers. The framework allows for a flexible system partitioning depending on the HW architectures, number of cores, etc. This is achieved by **Messages** and **Threads** blocks, which are responsible for handling the inter-module communication in a transparent manner and for mapping modules on one or multiple threads. The different deployment options are shown in Figure 2-12. The interfaces are loosely coupled in so far that the actual routing of the messages to the entities can be configured at run-time, allowing redirection of primitives to the other modules or test systems. The **Tracing** block handles logging of selected layer-specific system and protocol errors, as well as logging of selected events and changes in layer-specific context. The **Memory** block handles all aspects related to memory management, i.e. sharing, encoding/decoding of PDUs, etc. The **Socket** module is responsible for all IP interface related functionality and is used by Tracing module and by the TNL module.

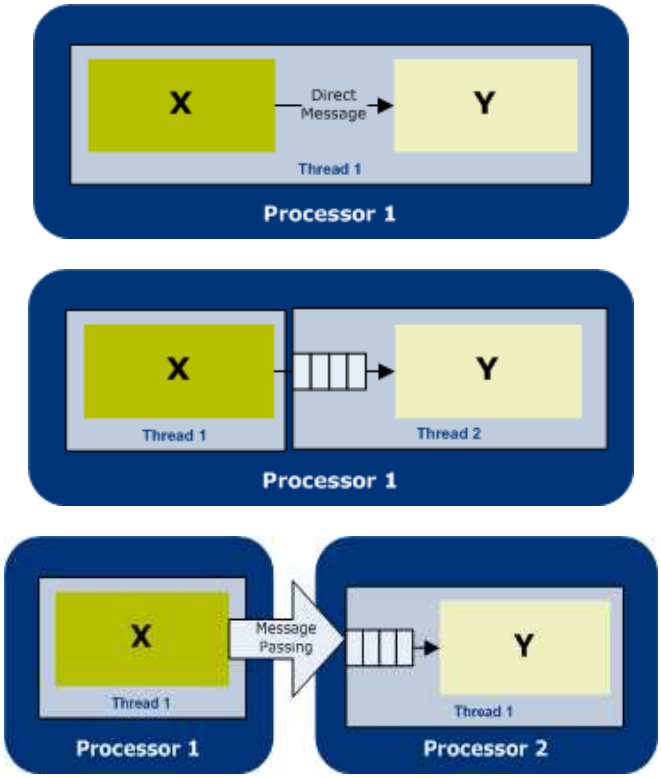

Figure 2-12: Protocol stack framework

The **HTTP server/web interface** allows configuration of the HeNB by modifying the database via a web or command line interface.

2.2.3.2 Dynamic View of Protocol Stack

This section explains the information and data flow of the protocol stack. The CPU time allocation of the TTI based operation is shown as well as the signalling and scheduling message sequence. In Figure 2-13, the basic TTI-based operation is shown for a system with one CPU when DMA is used for transferring the user data to and from PHY. The length of the blocks shows the relative allocation of each block within one millisecond.

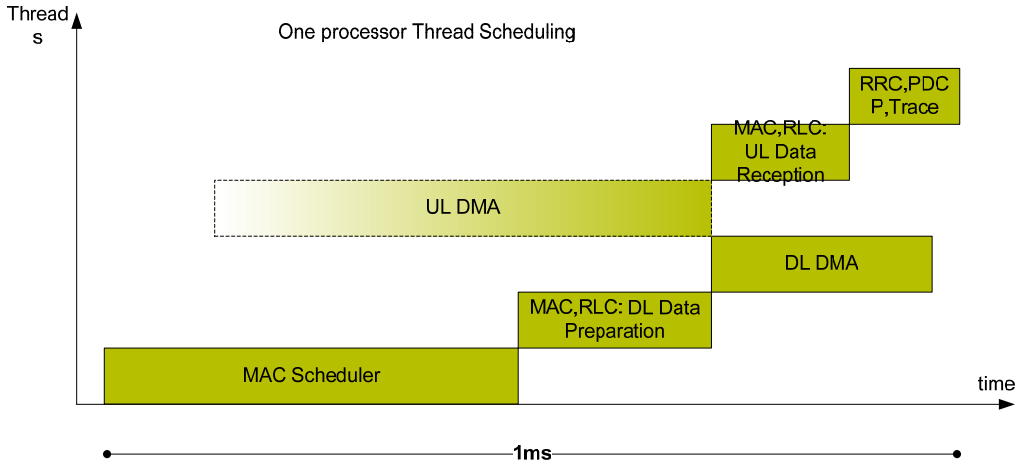

Figure 2-13: TTI execution partitioning

2.2.3.2.1 Signalling

The message sequence chart in Figure 2-14 shows the supported signalling in RRC, when no evolved packet core (EPC) network is available.

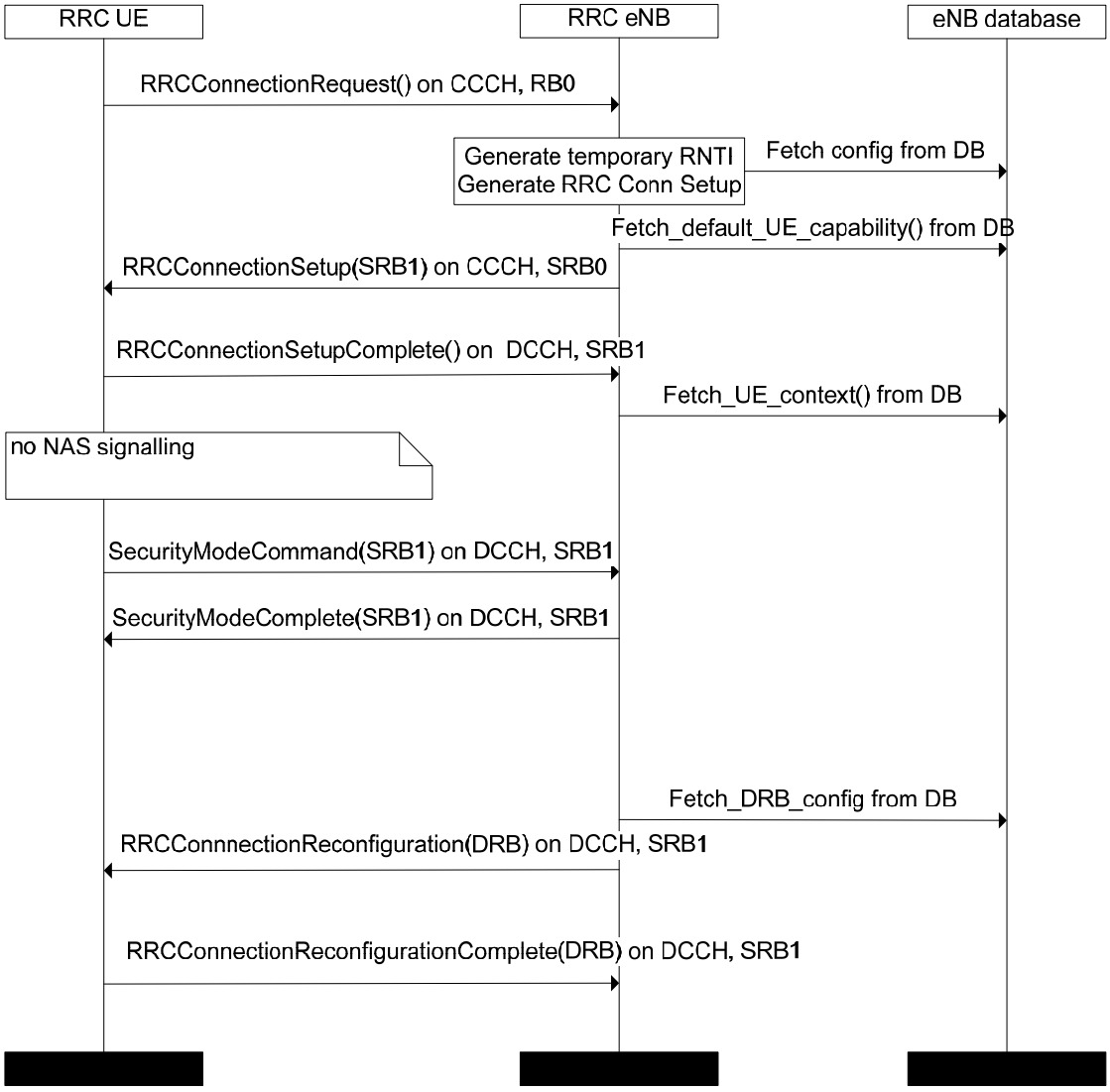

Figure 2-14: Signalling flow

The UE triggers the random access procedure after selecting the HeNB as serving cell. The RRC Connection Establishment procedure is then executed after the RACH preamble has been detected. Dummy ciphering and integrity protection is activated and then the RRC Connection Reconfiguration procedure is executed to setup the data radio bearer for transferring user data. During this initial setup procedure parameters which are needed to be configured in the UE and are normally either received via S1 interface from the core network or via OAM interface are fetched from the configuration database.

2.2.3.2.2 Scheduling

After the initial bearer setup user data scheduling is done every millisecond. The scheduler, as shown in Figure 2-15 receives information from different layers like RLC, MAC, PHY and RRC in order to make its scheduling decision. It will receive RLC queue status information, QoS parameters, CSI for frequency selective scheduling, HARQ feedback and semi-static configuration parameters like channel state feedback periodicity, etc.

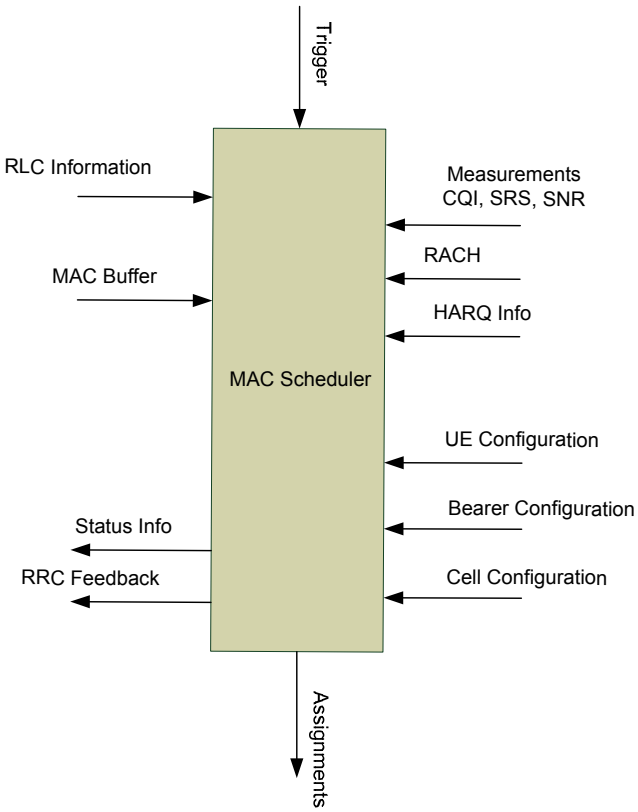

Figure 2-15: Scheduler overview

One example message sequence of how the scheduler is triggered is shown in Figure 2-16. It also shows the interworking with the user-plane and the PHY. Whenever data arrives at RLC the buffer status is updated in the scheduler using the SCHED_DL_RLC_BUFFER_REQ message. The message includes information about the queue size and the head of line delay of the queue. When the PHY has received a new channel quality indicator (CQI) from the UE this information is passed via the MAC to the scheduler using the SCHED_DL_CQI_INFO_REQ. This message includes the information standardized in LTE for CSI feedback. The scheduler will then be triggered to make it scheduling decision by receiving a SCHED_DL_TRIGGER_REQ. Based on the updated queue status, CQI and other information the scheduler makes it scheduling decision, allocating resource blocks for the selected UEs and setting the transport block size. This transport block size will then be used to build the transport block to be transmitted in this TTI.

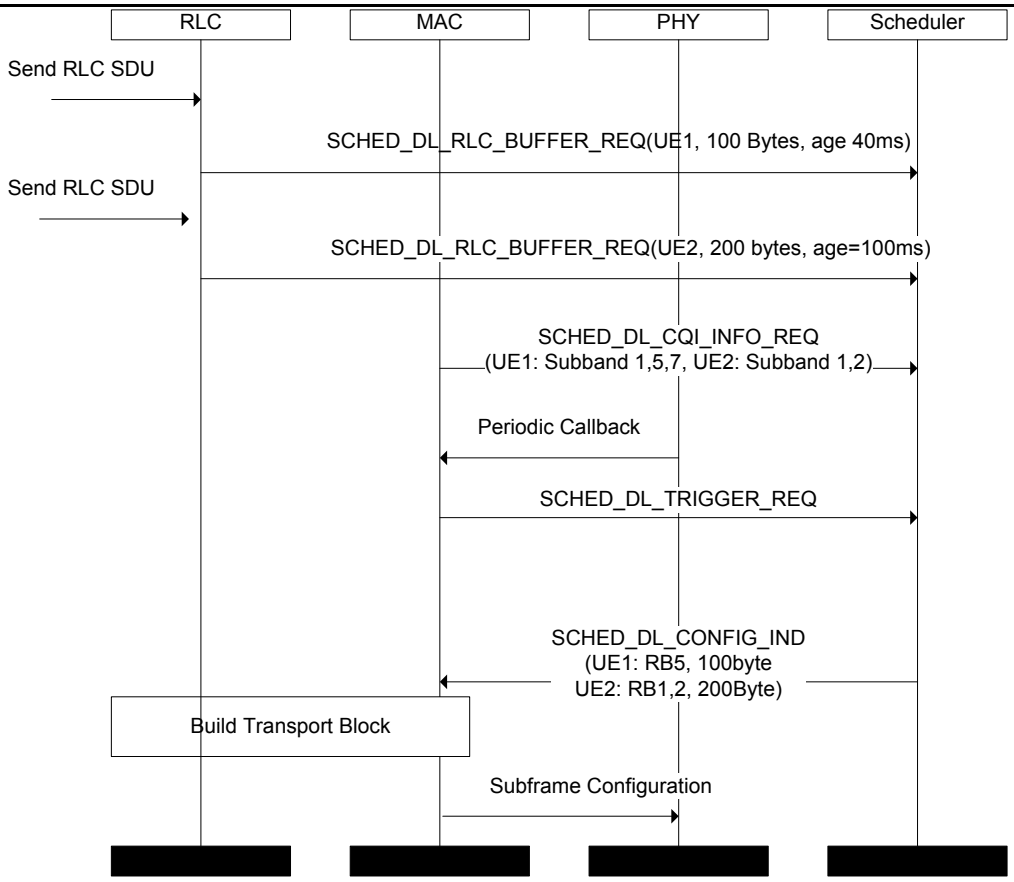

Figure 2-16: Downlink scheduling

2.2.3.3 Main Features Available

The main features of the scheduler are described below.

2.2.3.3.1 Frequency Selective Scheduling

The scheduler supports time/frequency-domain channel-dependent scheduling, that is, selecting the transmission configuration and related parameters depending on the instantaneous downlink/uplink channel conditions. Such scheduling, for example, in case of frequency-selective fading, significantly improves system performance in terms of achievable UE throughputs. It is a key feature of LTE.

2.2.3.3.2 UE Class Support

The scheduling algorithm respects the UE maximum data rates taking the UE categories into account. It does not assign a data rate to the UE exceeding UE's capabilities.

2.2.3.3.3 Distributed VRB Allocation

In this LTE feature, consecutive virtual resource blocks (VRBs) are not mapped to PRBs that are consecutive in the frequency domain. Distributed VRB allocation is beneficial in following cases:

- For low date rate services such as VoIP as the control signalling associated with frequency-selective scheduling results in relative high overhead.
- At high mobility scenarios (high UE speed) as it is difficult to perform accurate channel-dependent scheduling.

2.2.3.3.4 QoS Support

Although channel-dependent scheduling increases system capacity and spectral efficiency, it is not realized at the cost of UEs with unfavourable channel conditions. A new advanced proportional fair (PF) scheduling algorithm considers explicitly the UEs' QoS requirements in its operation. Additionally, the algorithm makes its scheduling decision not only based on the past UE average throughput but uses also the prediction of required scheduling rate for UE taking past scheduled UE throughput and QoS requirements of UE's logical channels (LCH) into account. The scheduler is configured to ensure that

traffic belonging to a bearer with a certain standardized QCI receives the appropriate QoS treatment. The standardized QoS characteristics from 3GPP TS 23.203 [9] (Table 6.1.7) are supported.

Table 2-12: Standardized QCI characteristics

QCI	Resource Type	Priority	Packet Delay Budget	Packet Error Loss Rate	Example Services
1	GBR	2	100 ms	10^{-2}	Conversational Voice
2		4	150 ms	10^{-3}	Conversational Video (Live Streaming)
3		3	50 ms	10^{-3}	Real Time Gaming
4		5	300 ms	10^{-6}	Non-Conversational Video (Buffered Streaming)
5	Non-GBR	1	100 ms	10^{-6}	IMS Signaling
6		6	300 ms	10^{-6}	Video (Buffered Streaming) TCP-based (e.g., www, e-mail, chat, ftp, p2p file sharing, progressive video, etc.)
7		7	100 ms	10^{-3}	Voice, Video (Live Streaming) Interactive Gaming
8		8	300 ms	10^{-6}	Video (Buffered Streaming) TCP-based (e.g., www, e-mail, chat, ftp, p2p file sharing, progressive video, etc.)
9		9			

For Guaranteed Bit Rate (GBR) bearer, additionally Maximum Bit Rate (MBR) parameter may be configured to limit the bit rate assigned to a GBR bearer. Any traffic in excess of the MBR is discarded by a rate shaping function.

In addition to the QoS parameters associated with each GBR bearer, the Aggregate Maximum Bit Rate (AMBR) parameters that are associated with non-GBR bearers may be configured. They limit a total bit rate that a scheduler is allowed to assign for all non-GBR bearers. Two variants of the AMBR are supported: the APN-AMBR and the UE-AMBR.

2.2.3.3.5 MIMO Support

Spatial multiplexing based on Multiple Input Multiple Output (MIMO) technology sends one or two independently transport blocks to the UE on spatial streams. The scheduler dynamically switches between transmission of one and two transport blocks and assigns transmission parameters of streams depending on the CQI, Rank Indicator, and precoding matrix selection reported from the UE. Link adaptation (LA) and HARQ for the two streams are decoupled. Transmission of two parallel streams is supported only in downlink direction to one UE (SU-MIMO).

2.2.3.3.6 Link Adaptation

Link adaptation is a fundamental feature of the radio interface for the efficient data transport. In the link adaptation, data rate is dynamically changed for each UE by selecting the Modulation and Coding Scheme (MCS) based on radio channel conditions so that overall network capacity and coverage are improved. In the downlink direction, the scheduler decides on MCS according to the Channel Quality Indicator (CQI) reported from UEs. The CQI is an indication of the data rate, which can be supported by the downlink channel, with regard to SINR and the characteristics of receiver at the UE.

2.2.3.3.7 MAC Multiplexing

Multiplexing of logical channels (LCHs) depends on the priorities between different bearers. Radio Resource Control (RRC) signalling has a higher priority than streaming data, which in turn has higher priority than a background file transfer. QoS requirements affect the multiplexing of different logical channels. The multiplexing algorithm firstly gives priority to MAC Control Elements (CEs), RLC Status PDUs and pending RLC retransmissions. After that LCHs are multiplexed in decreasing order of priority, but the multiplexing amount of data from each LCH is limited firstly to the LCH QoS requirements. The remaining space is multiplexed by round-robin fashion guaranteeing service of the best-effort traffic.

2.2.3.3.8 HARQ Retransmissions

The scheduler is responsible for allocating resources to UEs with pending HARQ retransmissions. HARQ retransmissions have highest priority in time domain, to ensure lowest latency. In frequency domain, retransmissions are scheduled taking into account the HARQ combining gain.

2.2.3.3.9 Adaptive and Non-Adaptive Uplink HARQ Retransmissions

The scheduler implements the possibility to operate the uplink hybrid-ARQ in a synchronous, adaptive mode, where the RB allocation and MCS scheme for the HARQ retransmissions are changed. Although non-adaptive HARQ retransmissions are usually used in uplink to reduce overhead of downlink control signalling, the scheduler uses adaptive uplink retransmissions when needed to avoid fragmenting the uplink frequency resource and to avoid collisions with RACH resources.

2.2.3.3.10 Interframe and Intraframe Uplink Hopping

The scheduler supports two types of uplink frequency hopping defined for PUSCH:

- Subband-based hopping according to cell-specific hopping/mirroring patterns.
- Hopping based on explicit hopping information in the scheduling grant.

Selection between hopping types is performed dynamically by setting single bit in the uplink scheduling grant.

2.2.3.3.11 Supported Release 8 Transmission Modes

Transmission mode/schemes (transmit diversity, open and closed-loop spatial multiplexing, or beam forming) is configured semi-statically via RRC signalling. The different transmission modes imply different configurations of the required UE reports and thereby the scheduler operation takes transmission modes into account.

2.2.3.3.12 Flexible Parameterization

Finding one optimal Scheduler parameter setting for all possible deployment scenarios is not possible. Optimum Scheduler behaviour usually depends on scenario constraints such as the desired fairness level, the current cell load and other cell-individual and provider-specific boundary conditions. Irrespective of the default parameter settings of the scheduler, operators or system vendors shall be able to configure the behaviour of the scheduler.

2.2.3.3.13 KPI Monitoring

Scheduler performance tracking is carried out using the key indicators: cell throughput, UE throughput, RB resource usage.

2.3 Work Plan

Aiming completion by the end of the project, Testbed1's integration will go through the following tasks:

Task Id	Start	End	Task Description
T1	May-11	Dec-11	RF Development
T2	May-11	Jan-12	PHY Development
T3	May-11	Dec-11	Stack Development
T4	May-11	Jan-12	Algorithm Selection and Implementation
T5	Dec-11	Apr-12	RF and PHY Integration
T6	Dec-11	Apr-12	Protocol Stack and PHY Integration
T7	Mar-12	Apr-12	System Integration
T8	May-12	Jun-12	System Validation

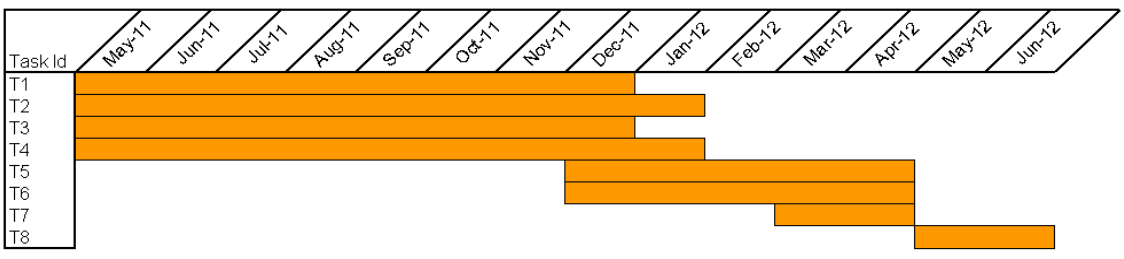

Figure 2-17: Testbed 1 integration work plan

3. Testbed 2, Networked Femtocells

The networked femtocells testbed is devoted to experimentally demonstrate the feasibility of the network of femtocells theme coined by BeFEMTO [1]. Its goal is to demonstrate routing and traffic offload concepts developed in WP5. This entails integrating a 3GPP core network and a local network interconnecting the femtocells deployed in a certain area. One of the key enablers of this proof-of-concept is the Iuh interface [10], [11]. The 3G femtocells (UMTS/HSPA) integrated into Testbed 2 supports full compliant Iuh interface with the core network. The underlying transport network of this local network may be all-wireless, hence forming a wireless multi-hop network, or wired (e.g., the LAN of an enterprise). In this section, we describe the selected schemes for which a proof-of-concept has been built upon Testbed 2.

3.1 Distributed Routing Algorithm

A fundamental building block for the performance of an all-wireless network of femtocells is the underlying routing scheme, since it will eventually determine the quality experienced by the end-users. This is the reason why we set ourselves to evaluate by simulation (in WP5) and experimentally (in WP6) the performance of such a relevant building block. Mainly targeting scalability, we proposed a distributed routing approach (see subsection 3.1.1) that exploits queue length information of neighbouring nodes and geographic information to make routing decisions.

The proposed solution is able to distribute resource consumption amongst all HeNBs that belong to the Network of Femtocells (NoF) whenever it is required. A more detailed description, and also an extensive ns-3 simulation evaluating the resulting routing protocol can be found in [12]. In contrast to topology-based studied in [1] the solution proposed adapts to the properties of an all-wireless NoF (e.g., cheap infrastructure, unreliable medium and high degree of self-organization). It can be considered a mesh-based approach in the sense that multiple paths are available between HeNBs and the EPC [13]. Nevertheless, the proposed solution goes beyond previous mesh-based approaches exploiting queue length and geographic information in a totally distributed manner.

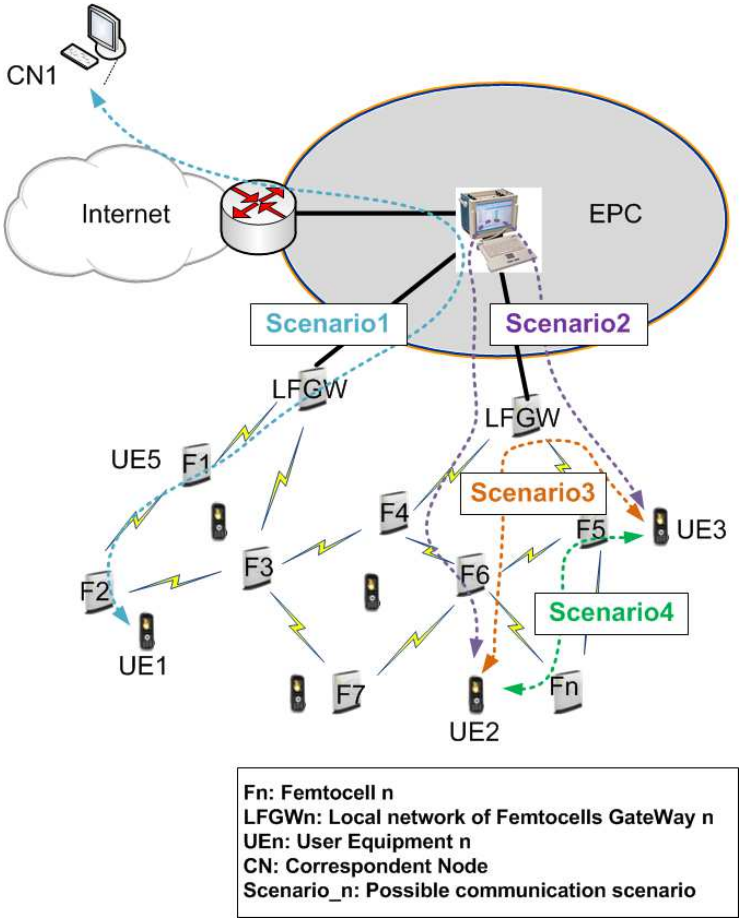

Figure 3-1: All-wireless NoF

More specifically, in our proposal HNBs are equipped with an additional wireless interface (e.g., Wi-Fi) in order to create a wireless mesh network amongst them acting as multihop wireless backhaul to reach the EPC. The wireless mesh network acts as a low-cost backhaul for HeNB interconnections (see F_1 to F_n in Figure 3-1) to which UEs can attach. Additionally, we introduce a mandatory element in the network called the Local network of Femtocells GateWay (LFGW) which acts as traffic concentrator coming from the HeNBs.

3.1.1 Short Description

The distributed routing algorithm selects the neighbour providing the best trade-off between minimizing the Euclidean distance to the destination and maintaining the network stable (i.e., the number of packets in the queue of every femtocell in the network is upper bounded). See [12] for an explanation and solution of the stochastic network optimization problem in order to get the proposed distributed routing algorithm.

When taking forwarding decisions at a given node femtocell *i*, it selects the neighbouring femtocell *j* that maximizes the weight W_{ij} between both femtocells. This weight is calculated as the difference of physical queue lengths minus the difference of an appropriately weighted penalty function, as shown in the following equation:

$$W_{ij} = \Delta Q_{ij} - V \cdot penalty$$

where

$$\Delta Q_{ij} = Q_i - Q_j$$

is the differential of physical queue lengths between femtocell *i* and femtocell *j*. The penalty function is introduced to not merely aim for network stabilisation but also for optimization of another parameter, such as delay. Moreover, the penalty function is weighted by a non-negative control parameter $V \geq 0$. This parameter weights the penalty function when calculating the weight of each pair (local femtocell *i*, neighbour femtocell *j*).

In our case, the penalty function is a function of the difference between the Euclidean distance between femtocell *j* and destination *d* and the Euclidean distance between femtocell *i* and destination *d*. Let *penalty(i, j, d)* denote the penalty function computed as the cost to traverse the link between femtocell *i* and femtocell *j* to reach destination femtocell *d*. In particular, the cost to traverse a link (*i, j*) depends on the distance of a certain femtocell to the intended destination. In the proof-of-concept testbed, the following penalty (or cost) function *penalty(i, j ,d)* is defined as:

$$penalty(i,j,d) = \begin{cases} 1 & \text{if } i \text{ is farther than } j \text{ from } d \text{ in terms of euclidean distance} \\ -1 & \text{if } i \text{ is closer to } d \text{ than } j \text{ in terms of euclidean distance} \end{cases}$$

Therefore, *V* has the responsibility of weighting the cost function with respect to the differential of physical queue lengths between femtocells *i* and *j*.

If there is no neighbouring femtocell *j* whose weight W_{ij} computation with node *i* is bigger than 0, the Data Queue Management procedures of the block in node *i* are not executed. This means that there will be no dequeueing event in node *i* routing data queue. Therefore, in this case, the routing algorithm considers that it is even better for the all-wireless network of femtocells not to forward the data packet.

3.1.2 Demo Scenario and KPIs

3.1.2.1 Distributed Routing/Load Balancing for All-Wireless Networks of Femtocells

- A practical (i.e., implementable) version of dynamic backpressure routing will be deployed in a 12-node network of femtocells (see Figure 3-2) in which the Iuh logical interface will physically traverse a multi-hop wireless WLAN transport network.
- The network will be loaded with traffic coming from UEs running the Multi-Generator (MGEN) traffic generator. It may also be loaded with VoIP or other types of clients (e.g., FTP to test the performance of TCP traffic). We will demonstrate how resource consumption is distributed throughout the network as a function of the load in the network.
- **KPIs**: Aggregated throughput at the gateway, per-flow throughput, and packet delivery ratio will be measured. Other parameters may also be measured depending on interest for each particular setup (e.g., packet reordering, end-to-end delay).

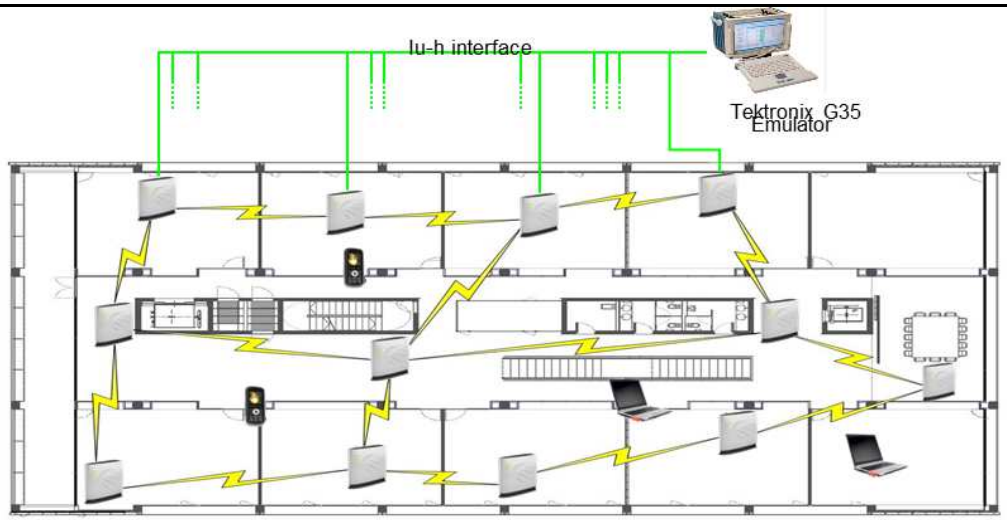

Figure 3-2: 12-node network of femtocells

3.1.3 Key Building Blocks and Integration Specification

The first subsection describes the main key building blocks, and the second subsection outlines the integration work carried out to implement the distributed routing protocol described in section 3.1.1.

3.1.3.1 Key Building Blocks

The following is a list of the key building blocks required for the demo scenario described above.

- **Routing Module** (Distributed): Provides the control logic, state machine, and interface for the distributed routing module.
    - *Data queue management*. Storage of packets waiting for forwarding decision to be taken. FIFO scheduling will be implemented. It may optionally implement various packet scheduling strategies.
    - *Neighbour management*. Determination of neighbouring nodes reachable in one wireless hop (direct communication). Generation/processing of HELLO messages containing queue backlog and coordinates. Maintenance of neighbour table.
    - *Next-hop determination*. Per-packet determination of best next-hop based on routing algorithm.
    - *Auxiliary blocks*. Availability of coordinates is assumed. They will be manually configured at each node. Location management statically configured by means of a table available at each node. This table maps IP addresses into physical coordinates.
- **Routing Algorithm** (Distributed): Implementation of the location-aware, backpressure-based distributed routing algorithm.
    - Determination of best next-hop based on queue backlog difference, where queue means the sum of the real queue and virtual queue. Virtual queue length calculated as a function of distance to the destination.

Furthermore, and benefiting from the transport network just explained, there is also the need for:

- **Iuh Interface**: Implementation of an Iuh compliant interface to enable interoperability between partners equipment.

The following subsection provides a brief description of the key enablers for the deployment of the algorithm in the testbed and their main characteristics. After that, we explain how they are used to implement the building blocks of the distributed routing algorithm.

3.1.3.2 Required Enablers for the Integration

In [12] we provide an extensive simulation of the distributed routing algorithm. In this section we describe the required implementation enablers in order to evaluate the distributed routing protocol in the all-wireless NoF testbed.

3.1.3.2.1 ns-3 Emulation Framework

We use the emulation framework provided by ns-3 in order to test the routing protocol in a proof-of-concept testbed. Basically, it allows executing ns-3 simulation stacks over real physical devices. To enable these features, the ns-3 emulation framework provides ns-3 emulated network devices that look like a usual ns-3 simulated device (i.e., EmuNetDevice) from the top, which will provide the interface to the real physical network underneath using a packet (raw) socket (see Figure 3-3). In this way, the ns-3 stack is able to send and receive packets over real devices.

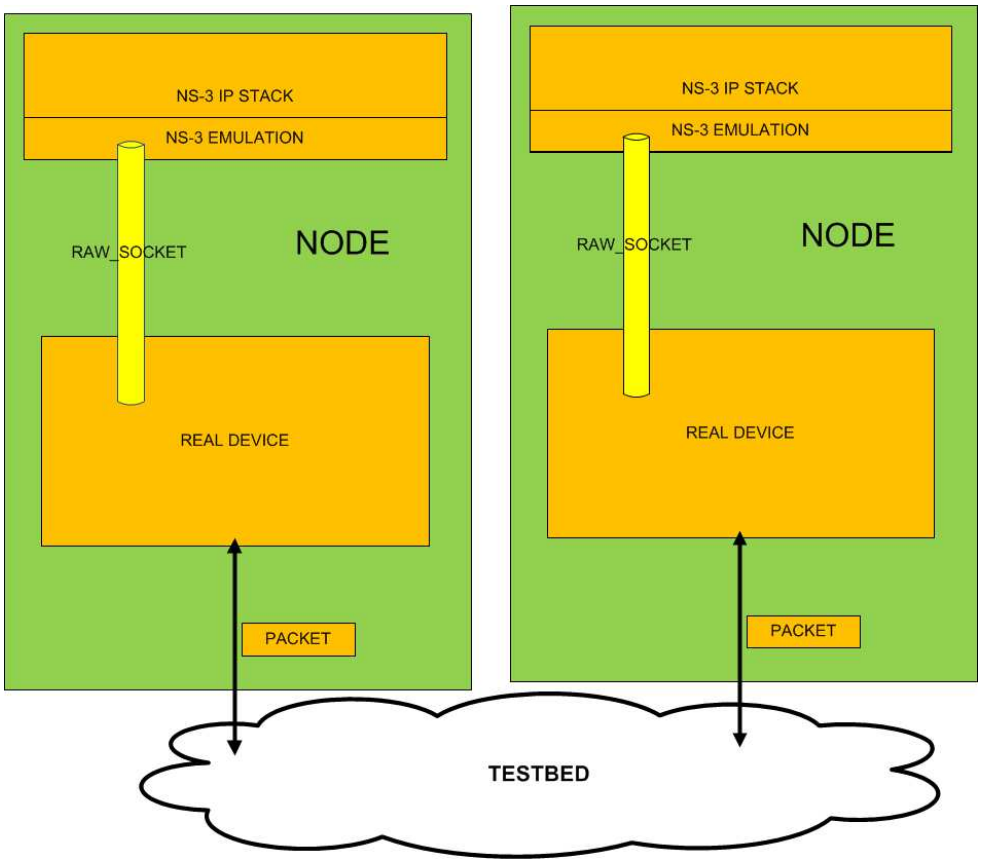

Figure 3-3: ns-3 emulation framework

To distinguish between Linux traffic that might be generated concurrently and ns-3 simulation traffic, the EmuNetDevice implements MAC spoofing. To work properly, the real physical device associated to the EmuNetDevice must be up and configured in promiscuous mode. As a result, the packets generated with the EmuNetDevice and sent over the physical device, have assigned a MAC address different from the MAC belonging to the real Linux device. On the other hand, the ns-3 emulation framework requires a real-time scheduler, given that physical real devices (synchronized with real physical clocks) generate events (e.g., packet reception) in the ns-3 stack. In order to receive and process packets the real devices must work in promiscuous mode.

3.1.3.2.2 /sys Subsystem

The /sys subsystem (see Figure 3-4) is a kernel memory region storing information about what is happening in the kernel space. It can be used as a way of announcing events to the user space, given that user processes can have access to this region. We are using this virtual file system to store the hardware queue length of a real femtocell. To do this, a new /sys entry was implemented to have access to the hardware queue length values (see Figure 3-4). Then, the ns-3 user space program reads an entry in this /sys entry whenever it has to send a HELLO broadcast message.

In other words, for efficiency purposes, the current hardware queue length is read when a new HELLO broadcast message has to be sent. However, the hardware queue might fill up between the transmissions of two consecutive HELLO broadcast messages. To announce hardware queue full/empty events we use Netlink sockets.

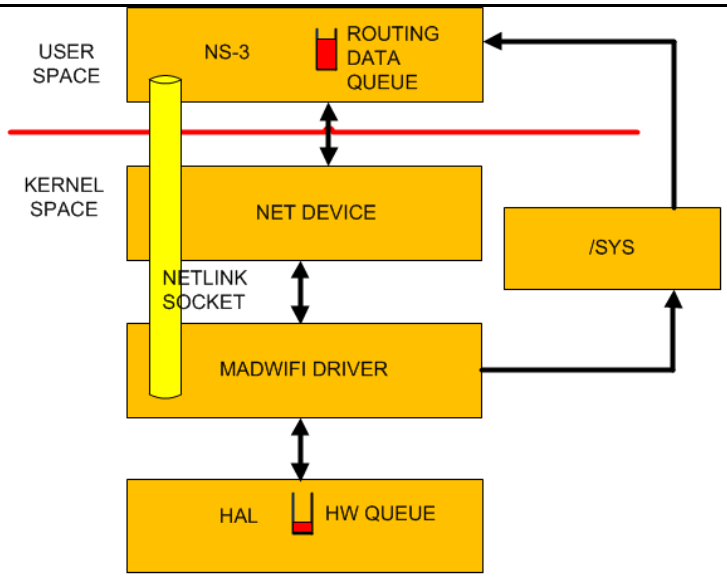

Figure 3-4: User/kernel interaction through the /sys subsystem

3.1.3.2.3 Netlink Sockets

Linux Netlink Sockets allow for bidirectional interaction between user space and kernel space modes in an efficient manner. It consists of a standard sockets-based interface for user processes and an internal kernel API for kernel space.

Netlink sockets are used in our implementation to allow for the efficient interaction between real physical queues (those in the Atheros Wi-Fi card managed by the Atheros madWi-Fi linux driver as depicted in Figure 3-4) and the ns-3 daemon running in user space.

More specifically, we want to transfer information between the routing data queue building block in the routing algorithm, and the Linux module able to access the physical queue information. A direct communication channel is then established between the ns-3 user space and the hardware queue. Specifically, we are interested in the event that the hardware queue is full/empty so that no more data packets are sent to the physical queue.

3.1.3.3 Integration in the Key Building Blocks

3.1.3.3.1 Neighbour Management

The neighbour management building block is in charge of determining which femtocells can be reached by means of direct communication (i.e., without having to cross any intermediate femtocell). A detailed explanation of the building blocks and their inter-relation can be found in BeFEMTO D6.1 [1]. Essentially, this building block requires the periodical transmission and reception of HELLO broadcast messages. In our proof-of-concept testbed, the ns-3 emulation framework would be in charge of providing the transmission and reception of HELLO broadcast messages. This is done through the EmuNetDevice, which provides ns-3 of an abstraction of the real wireless physical device. Specifically, ns-3 implements an independent thread (from the one running the ns-3 routing code) in order to send and receive packet over the EmuNedNetDevice, given the synchronous nature of the sockets.

On the other hand, the Neighbour Management building block requires the announcement of the number of packets stored in a femtocell. In the testbed, the location of packets may be shared between the kernel space in the hardware queue length, and in the user space (in the ns-3 emulation stack). In order to include the number of potential data packets accumulated in the hardware buffers, the Neighbour Management building block exploits the /sys filesystem, which will store the hardware queue size.

3.1.3.3.2 Data Queue Management

The Data Queue Management building block determines the scheduling carried out for incoming data packets. It is subdivided into two components, namely a packet scheduler policy and a queue-based structure. Both the packet scheduler and the queue-based structure are provided by the ns-3 emulation framework. Additionally, the queue-based structure also needs the hardware queue length to maintain the total queue length of a femtocell (i.e., hardware queue length plus routing queue length). The hardware

queue length is obtained from an entry written by a kernel module in a file in the /sys subsystem, as summarized in 3.1.3.2.2.

3.1.3.3.3 Next-Hop Determination

The Next-Hop Determination building block is in charge of computing the most appropriate next-hop on a per-packet basis. This building block requires of the ns-3 emulation framework, the Netlink sockets, and the /sys subsystem. The ns-3 emulation framework is required for the implementation of the algorithm in the testbed, in order to send/receive ns-3 simulation packet over the real proof-of-concept all-wireless network of femtocells. Netlink sockets (described in 3.1.3.2.3) are used for transmission regulation (i.e., scheduling) purposes between the HELLO emission intervals.

Recall that a femtocell will use a file /sys subsystem to read hardware queue length when it has to send a broadcast HELLO message. When knowing the hardware queue length, a femtocell will autonomously decide whether it can transmit data packets or not. However, between this period and the next HELLO message interval the hardware queue length might potentially change of state (e.g., hardware queue length might fill up due to an increment of traffic load). Therefore, the ns-3 emulation framework requires a way to detect these potential changes of state between the HELLO transmission intervals.

Specifically, the kernel will provide a signal to the ns-3 user space whenever it can not accept more data packets. This happens when the hardware queue is full. Furthermore, it can also provide a signal to the ns-3 user space to restart the injection of packets from the ns-3 user space to the kernel space. This can happen after detecting that the femtocell hardware queue is empty. On the other hand, given that the routing protocol also decides whether to send a data packet to the HW queue or not (i.e., scheduling) it might happen that packets get trapped at data queues in ns-3 user space. This occurs in the case in which the routing protocol decides to do not send packets to the HW queue because of congestion in the neighbour set. Therefore, these packets remain blocked at the routing data queue. In order to avoid this situation, the routing data queue is polled while it has still data packets trapped at routing data queues so that packets can be sent to its corresponding HW queue.

3.1.4 All-Wireless Network of Femtocells Specific Setup

With regards to the setup of experiments within the all-wireless NoF testbed, we highlight the following features.

3.1.4.1 Node Configuration

For the validation of the distributed routing protocol, just one single Wi-Fi interface is configured in every node. On top of this Wi-Fi interface we associate an instance of an ns-3 stack, by means of the EmuNetDevice. We associate to every interface a different MAC destination address from that corresponding in the physical device. All the Wi-Fi interfaces are configured to the band of 5 Ghz, and they have assigned the same channel. We are using 5 Ghz band in order to avoid external interference from the usual Wi-Fi LAN service of the CTTC building, and neighbouring buildings so that the experiments can be carried out without taking into account external interference sources.

3.1.4.1.1 On the Spoofed MAC Destination Address

The madWi-Fi drivers allow one to create multiple virtual wireless interfaces, which you can configure independently. This MAC addresses associated to these new virtual wireless interfaces must have a specific format. Precisely, the MAC address must satisfy the restrictions imposed by the mask assigned to the variable Basic Service Set Identifier (BSSID) mask in the implementation of the madWi-Fi driver. The BSSID mask specifies the common bits that a MAC destination address must match with the BSSID in order to process a receiving packet.

Therefore, as illustrated in Figure 3-5, the MAC destination address specified by the ns3 EmuNetDevice must also satisfy this format. Otherwise, packets received by the madWi-Fi driver with a MAC address not following BSSID mask restrictions are discarded. Thus, they cannot be captured by the RAW socket in the ns-3 class defined by the EmuNetDevice. In our case, the mask allows to specify a different MAC address to the ns-3 emulated device by changing the more significant bits of the usual MAC identifier. Therefore, a MAC was statically assigned to the ns3 EmuNetDevice by changing first more significant bits of the MAC address assigned to the ath0 device.

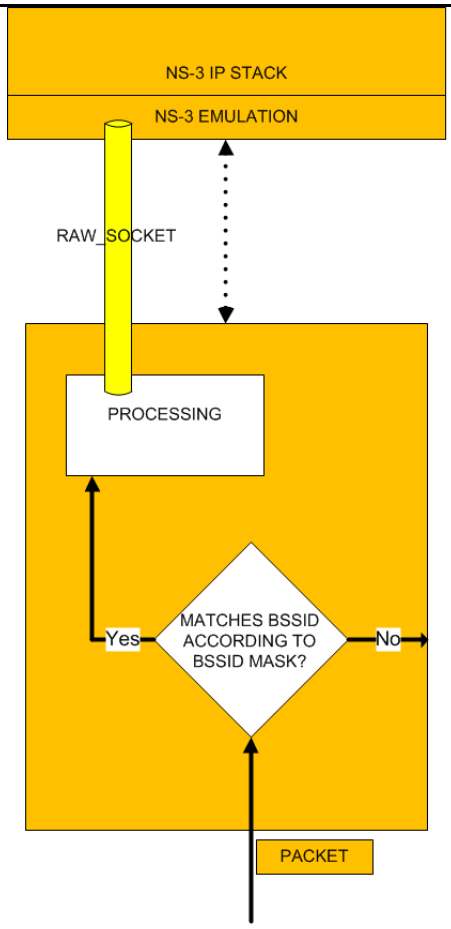

Figure 3-5: Packet reception with ns-3 emulator

3.1.4.1.2 Multihop Routing in a Confined Space

HELLO messages determine which nodes can be reached by means of direct communication (i.e., the neighbours). In order to evaluate the distributed routing protocol, a primary aspect is its behaviour under multihop topologies. However, the whole testbed is deployed over an area of around 1000 square meters. After a set of experiments performed, we observed in practice that HELLO broadcast messages were practically visible between all nodes in such confined space. The reason for this is that HELLO broadcast messages are usually transmitted at minimum data rate allowed by the Wi-Fi card (in our case 6Mbps). This leads into a high probability of decoding the received HELLO messages no matter the sender node under evaluation. In order to force multihop in this reduced area, we introduced a modification in the Linux madWi-Fi driver so that one can modify the rate at which broadcast HELLO messages can be transmitted. Hence, in order to evaluate multihop routing topologies HELLO messages are sent at the maximum allowed physical data rate (i.e., 54Mbps) so that several hops can be forced between the nodes deployed at the first floor. We have observed that under this configuration every node has at least two visible nodes and no more than four nodes with a high probability. Attenuators are also used for this purpose. In this way we can evaluate routing paths that have up to 4 hops of length.

In Figure 3-6 we illustrate the resulting connectivity patterns (with yellow lines) from the neighbour connectivity point of view after sending HELLO broadcast messages at a physical rate of 54Mbps. In order to facilitate multihop connectivity, we also exploited building obstacles (e.g., concrete walls or the elevator in the first floor of the building). An important point is that neighbour connectivity patterns may also be affected depending on additional obstacles during working hours (e.g., people working). In order to having more static link connectivity patterns, we carried out the experiments in the testbed at night. On the other hand, data packets are transmitted using an autorate algorithm. Specifically, the SampleRate algorithm is used. The initial rate is the maximum allowed physical data rate (i.e., 54Mbps). The number of retransmission is set to three. After these three retransmissions, the minimum physical data rate is used (i.e., 6Mbps). Additionally, other autorate algorithms which also exploit intermediate rates might be evaluated.

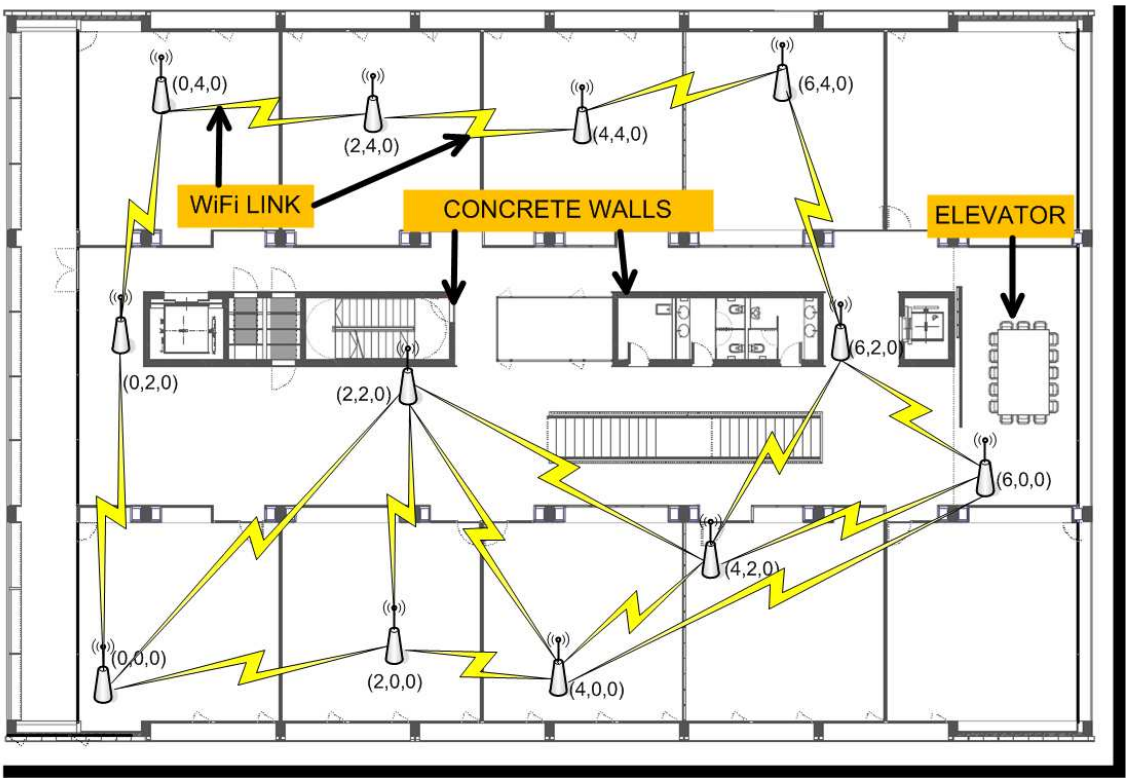

Figure 3-6: Single radio single channel wireless backhaul

3.1.4.1.3 On Coordinate Assignment

Recall that the routing protocol needs location information in order to compute the weights of every link. To do so, every node requires to have assigned some geographical coordinates. For the proof-of-concept, coordinates are statically assigned so that the network models a 4x3 grid. Note that according to the way the routing protocol computes the weights of the links, the specific value of the coordinates is not the most important value but the relative distance between two nodes to any other potential node. As can be shown in Figure 3-6 the coordinates might not reflect the physical geographic position with respect to the other nodes due to obstacles (such as brick walls or the elevator of the building). For instance, node whose coordinate assigned is position (6, 0, 0) should physically have assigned a bigger y coordinate taking at look at its neighbour in position (4, 2, 0). In contraposition, the coordinate assignment is such that it tries to respect the link connectivity patterns observed during the execution of the experiments.

3.1.4.2 Experiment Launch and Exploitation

In this subsection we outline the launching of an experiment, and also the collection of metrics in order to obtain network performance metrics.

3.1.4.2.1 Launching of an Experiment

Experiment automation benefits from the capabilities of the EXTREME Testbed. We use the EMMA tool to describe the execution of an experiment. Experiments are launched from a central server which is directly connected to the testbed via Ethernet through several L2 switches. This central server is in charge of executing a tool called EMMA. EMMA describes the execution of an experiment in the testbed. The configuration of the tool is based on an XML file that describes the execution of an experiment. EMMA allows repeating the same experiment as many times as specified, hence allowing to get statistical results of the same experiment execution. Moreover, EMMA allows running several experiments with a different input parameter specified. Eventually, EMMA executes an 'ssh' command to the selected remote nodes. In every selected node, this script will launch ns-3 emulation with a set of pre-defined parameters, and specifying if the node is going to behave as a sender, receiver, or merely forwarder of traffic. Senders and receivers of nodes running ns-3 emulation can also act as routers/forwarders of data traffic. Other parameters are specified (e.g., the *V* parameter, the emulation time, or the traffic intensity in the case the node is a sender).

3.1.4.2.2 Gathering of Results

Exploiting EMMA features, we remotely specify a Wi-Fi virtual device in monitor mode associated to the physical device used to send/receive packet in every remote node. In this way, a 'pcap' file is generated at every node in the testbed. Furthermore, the central server gathers results that correspond to statistics belonging to the nodes in the network. Precisely, one of the metrics collected is the data queue length. By data queue length we mean the queue length of the HW queue and the queue length belonging to the ns3 routing data queue. Furthermore, the number of data packets lost due to a queue drop, and the number of packets sent to the Wi-Fi HW queue are monitored. All these node metrics are periodically stored in their respective nodes.

3.1.4.3 Preliminary Results

In order to see how reliable are the results obtained by the evaluation of the routing protocol exploiting the ns-3 emulation framework, we carried out two types of measurements. One type of measurements is focused on evaluating the reliability of the ns-3 emulation framework with regards to throughput. The second type of experiments is devoted to evaluate the main feature of the routing protocol, namely the variable degree of load balancing a data flow can experience as a function of the load in the network. Thus, this section evaluates the implementation of the routing protocol in the ns-3 emulation framework (see [12] for a more exhaustive evaluation using ns-3 network simulator) over the testbed previously described.

3.1.4.3.1 Throughput Results

The first experiment corresponds to the measure of the throughput attained by a node which is at one hop distance from the node injecting traffic to the NoF. In this case, the distributed routing protocol which is configured with a high V (i.e., V=1000) merely has to take a single routing decision. In other words, the degree of load balancing was configured so that the shortest path in terms of number of hops was the only route taken by data packets. Precisely, we used as source, the node in position (4, 4, 0) and as destination node in position (6, 4, 0) (see Figure 3-6).

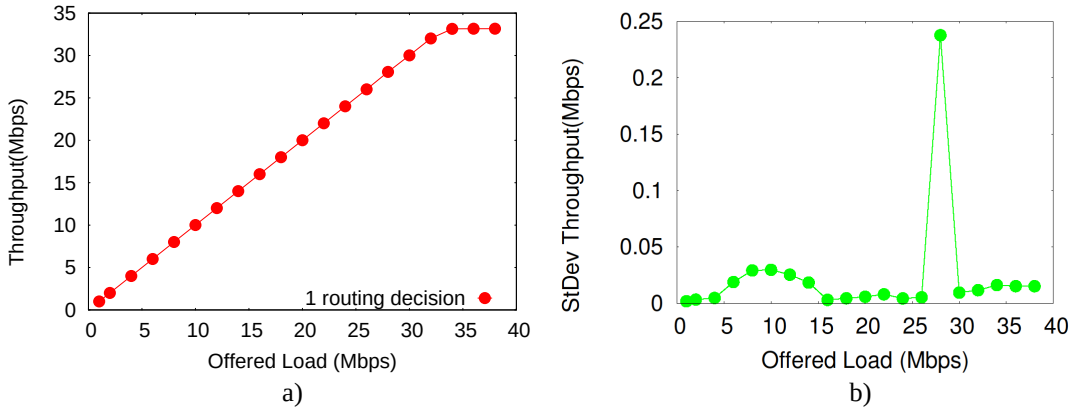

Figure 3-7: Achieved throughput a) Boxplot and b) Standard Deviation for 1hop

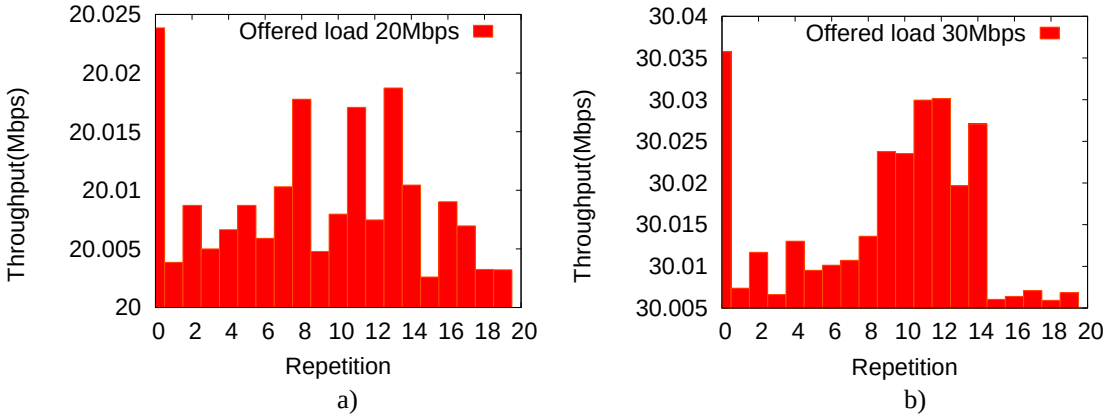

Figure 3-8: Achieved throughput per repetition in case a) Offered Load=20Mbps and b) Offered Load=30Mbps

Figure 3-8 shows the throughput reached at the receiver for 20 replications of the experiment varying the offered load from 1Mbps up to 36Mbps. In each case, a single node injected a unidirectional Constant Bit Rate (CBR) traffic of a different intensity (traffic intensity ranging between 1 and 36Mbps) generated from the ns-3 application. As it can be shown in Figure 3-7a, throughput results match the expected throughput that the Wi-Fi Atheros card would attain if the ns-3 emulation framework was not used (around 2900 packets of with maximum packet size). Recall that in this case, the *V* parameter is set up to a value so that the distributed backpressure routing protocol transmits directly to the receiver, and the receiver was located at 1-hop distance from the source. On the other hand, standard deviation (see Figure 3-7b) is shown in order to appreciate the slight variation experienced by the experiment.

Additionally, in order to see the slight throughput variability we show the cases in which the offered load is of 20Mbps (Figure 3-8a), and in the second case the offered load is of 30Mbps (Figure 3-8b). Interestingly, one main observation from these result is that in terms of throughput, the ns-3 emulation framework do not suppose a bottleneck with respect to the results that one would obtain using directly the Linux kernel stack (i.e., without the intervention of the RAW SOCKET entity to send/receive packet from user space to kernel space as Figure 3-4 depicts). Therefore, we can conclude that, in terms of packet generation and packet reception, the ns-3 emulation framework does not introduce any throughput degradation when saturating a Wi-Fi card configured at 54Mbps link rate.

The second experiment corresponds to the throughput measured attained by a node which is at a distance of two hops from the node injecting traffic to the NoF. Precisely, we used a source node in position (2, 4, 0) and a destination node in position (6, 4, 0) (see Figure 3-6). As in the previous experiment, the degree of load balancing was configured so that the shortest path in terms of number of hops was the only route taken by data packets. This experiment is composed of 18 replications. For each replication, several cases were tested. In each case, a single node injected a unidirectional CBR traffic of a different intensity (traffic intensity ranging between 1 and 18Mbps).

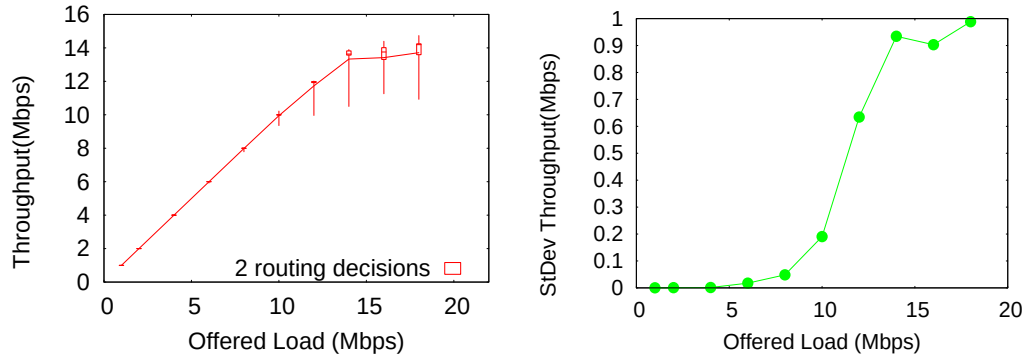

Figure 3-9 Achieved Throughput Results a) Boxplot and b) Standard Deviation for two hops

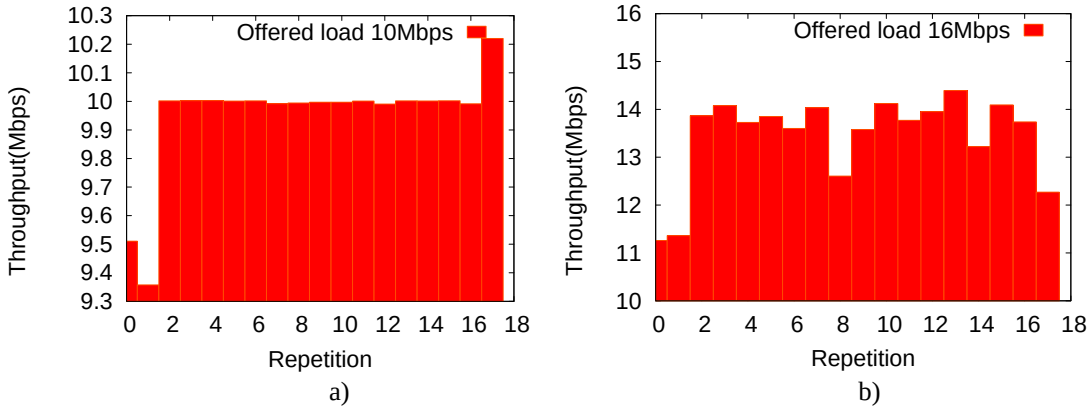

Figure 3-10 Achieved Throughput per repetition in case a) Offered Load=10Mbps and b) Offered Load=16Mbps

Throughput results (see Figure 3-9a for an illustration of boxplot results, and Figure 3-9b depicting standard deviation of the 18 replications) show an increase of variability (up to 1Mbps) with respect to the previous experiment. This is due to the fact of the increase in terms of contention experienced by the nodes in the Wi-Fi medium acting as first hop with the node acting as second hop. This contention

coordinated by the CSMA/CA algorithm can be unfair, especially when dealing with high input rates. This effect can be observed in figures representing the standard Deviation (see Figure 3-9b) which shows the increase of variability (up to 1Mbps), especially as the injected traffic intensity is increased.

3.1.4.3.2 Load Balancing Results

As explained in subsection 3.1.1, one of the key aspects characterizing the implementation of the routing protocol previously described is its capacity to do load balancing by means of exploiting the *V* parameter. In order to validate the load balancing features of the distributed routing protocol, we have carried out two experiments in which by means of modifying the degree of load balancing (indicated by the *V* parameter) we evaluate the behaviour of the routing protocol. For both cases under evaluation we inject a CBR unidirectional flow of 8Mbps from node in position (0, 4, 0) using UDP within the ns-3 application to the node in position (6, 4, 0). The UDP packets injected are of the maximum size allowed. We measure the throughput received by both of the available neighbours (nodes in positions (0, 2, 0) and (2, 4, 0)) to the transmitter and the evolution of the queue length of the transmitter and its set of neighbours during time. In the first case, we evaluate the routing protocol behaviour with a $V = 0$. This corresponds to a routing strategy in which data packets do not have any more direction to reach the destination than that specified by their node queue lengths. We evaluate the behaviour in terms of packets transmitted at every neighbour in the first hop (see Figure 3-11), and queue length (see Figure 3-12).

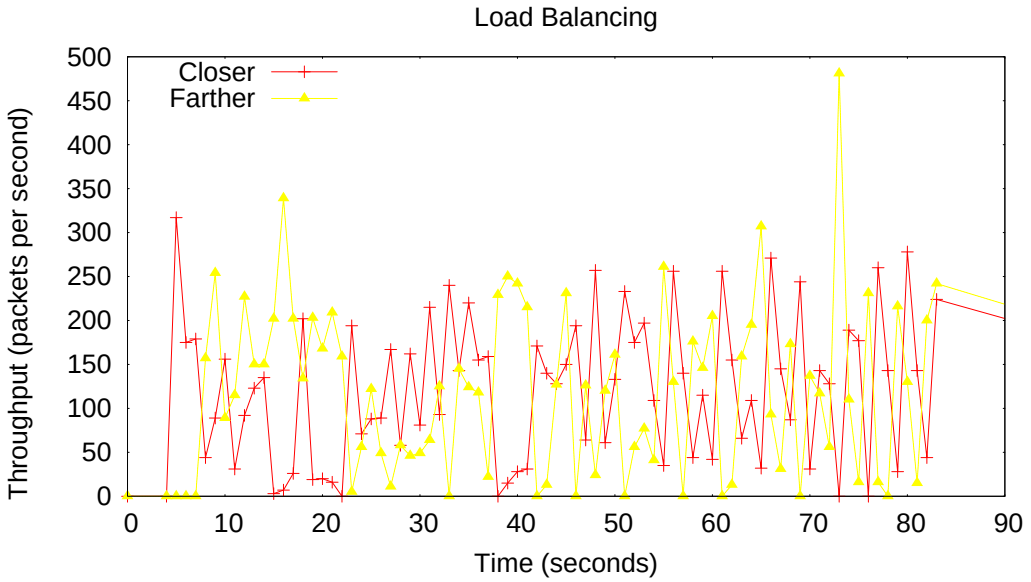

Figure 3-11: Packets received by neighbours V = 0

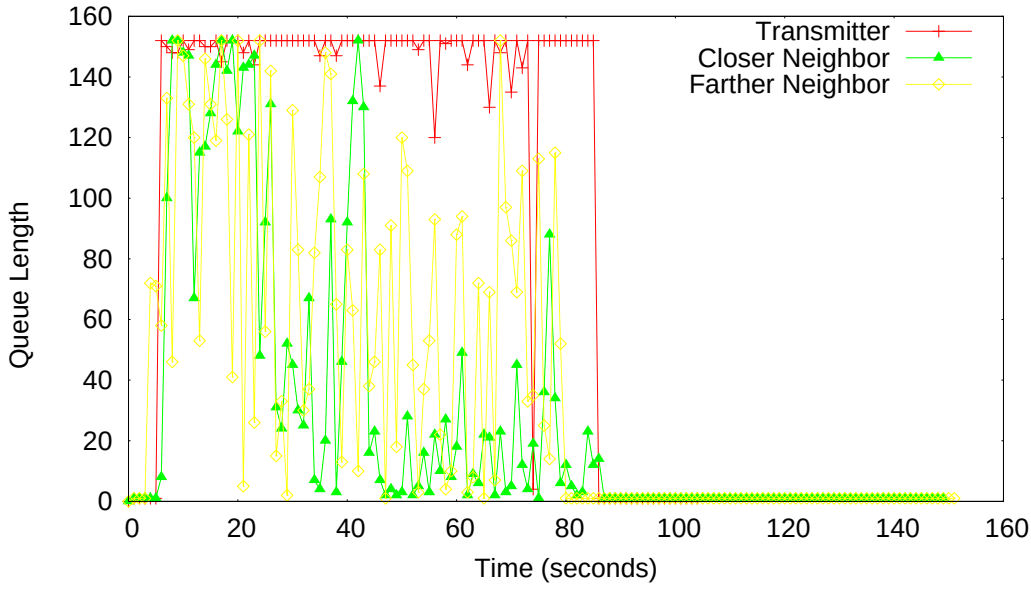

Figure 3-12: Data queue length evolution V = 0

As it can be shown in Figure 3-11, traffic sent by the node generating traffic is equally shared by the two neighbours available. In fact, choosing a V = 0 is equivalent to the original backpressure algorithm based on minimizing the Lyapunov drift alone. The node which is farther from the destination (i.e., the yellow one in Figure 3-11) is able to send packets to both the transmitter and the closer neighbour. Hence, routing loops might be generated, though in this specific case is unlikely given that the node generating traffic has usually the higher data queue length.

Interestingly, Figure 3-12 illustrates the decreasing queue backlog gradient generated towards a single destination (usually generated by a backpressure algorithm), given that the closer neighbour is the one with lower data queue length. With V = 0 until a decreasing data queue backlog is generated packet might arrive randomly or not to the destination that experiences a lower data queue length. On the other hand, as Figure 3-12 illustrates, at the end of the experiment all data queue lengths evolve to a queue length of 0 packets indicating that there is no more traffic to be serviced by the network which validates the appropriate interaction established between the HW queue (kernel level) and the ns-3 routing data queue using Netlink Sockets and the /sys subsystem (see Figure 3-4). In this case, there is a high variation in the queuing levels of the three nodes under evaluation due to the fact of taking transmission decisions based on the length of the data queues.

In the second case, we use a $V = 100$. This means that the penalty function indicating relative closeness to the destination is contributing to the routing control decisions. As it can be shown by Figure 3-13 no packets are transmitted to the neighbour that is farther to the destination. With regards to the evolution of the data queue length, it is clear than in the case of $V = 0$ both neighbours experience queuing. Interestingly, Figure 3-14 illustrates that the neighbour farther from the destination tends to experience more queuing than the neighbour closer to the destination. In contrast, in this case with $V = 100$ just the neighbour closer to the destination experiences queuing. This is due to the fact of the less degree contention experienced by the CSMA/CA, and that the input rate injected to the network was feasible (i.e., 8Mbps).

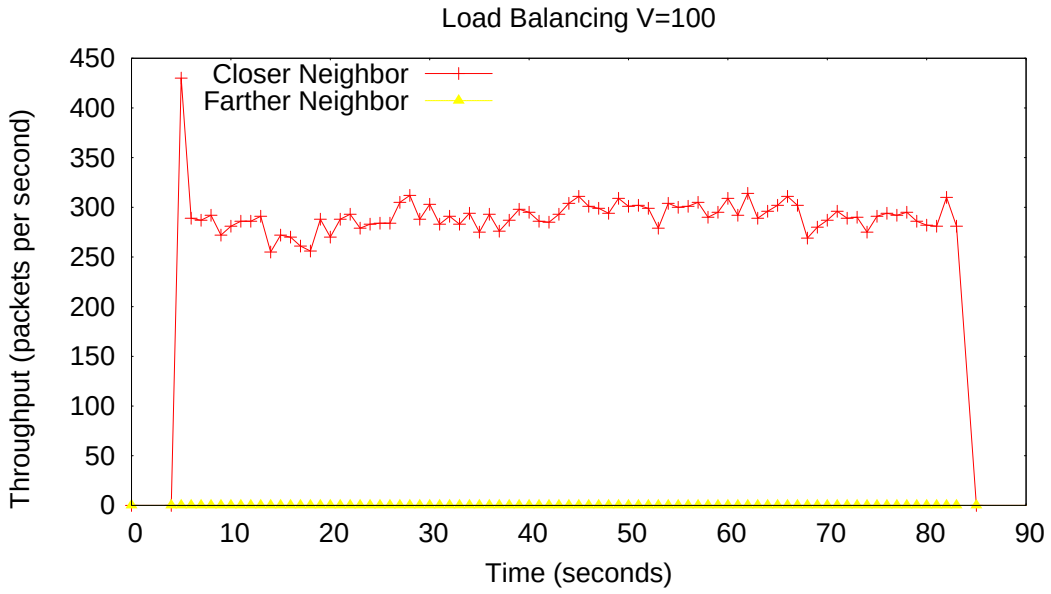

Figure 3-13: Packets received by neighbours V = 100

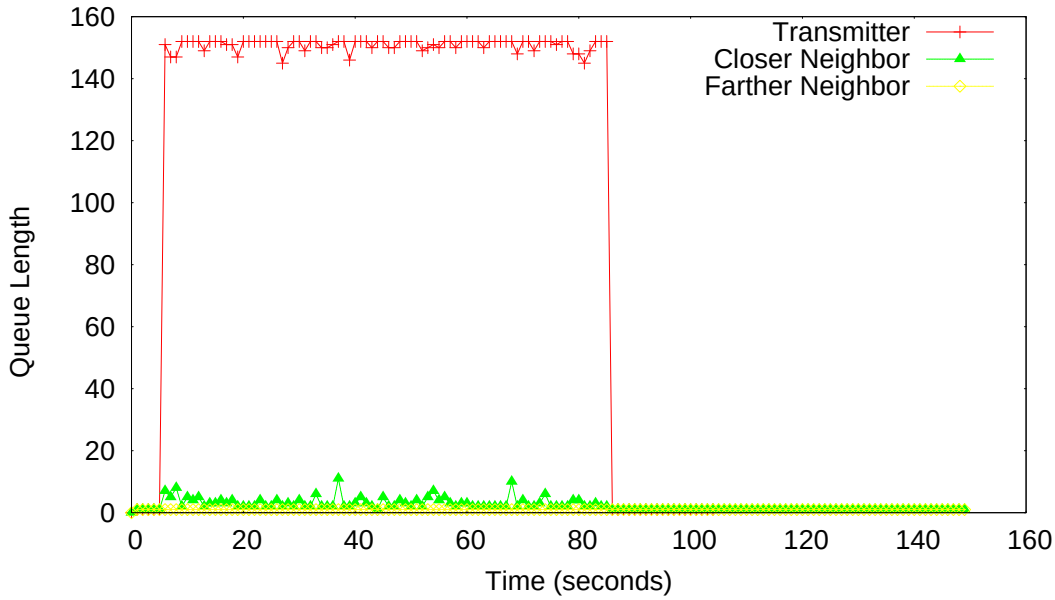

Figure 3-14: Data queue length evolution V = 100

We conjecture from these results of the importance of choosing an appropriate *V* parameter in order to minimize the queuing level at the nodes deployed in the wireless backhaul. Note that in this case the number of nodes sharing the medium and sending a significant number of data packets is reduced. In this case, the component indicated by the penalty function cause more transmission opportunities in the node closer to the destination compared with the previous case. This leads to a lower level of queuing in the closer neighbours and a more constant queuing level in the transmitter than in the case in which $V = 0$.

3.1.4.4 Concluding Remarks

We explained the particularities and experiences of the migration of a routing protocol implementation within the ns-3 simulator to an all-wireless NoF testbed. Additionally, we evaluated the resulting implementation in order to validate the feasibility of using ns-3 as a unique tool for simulation and experimentation with multi-hop wireless networks. On the one hand, we observed that ns-3 emulation framework does not introduce any impairment in terms of packet generation and reception rates. On the other hand, we evaluated the main feature of the routing protocol, that is, the role of the V parameter in the computation of the link weights. On the one hand, a low V parameter might tend to increase queuing of packets in the network though the input rate injected might be feasible without queue drops. In contraposition, a high value of the V parameter could be efficient for light injected rate is using the routing policy as a shortest-path strategy. Based on the initial evaluation carried out over the networked femtocells proof-of-concept testbed, and the potential of ns-3 simulator in order to re-use a network protocol implementation, we believe that the approach followed is the appropriate one. The last part of the project will be devoted to extensively evaluate the performance of the distributed routing protocol over this framework.

3.1.5 Work Plan

The realisation of the presented testbed can be divided in two main activities: the construction of the network of femtocells and the integration and validation of the backpressure routing proof-of-concept on top of this network. The following Gantt charts summarize the tasks to assemble the network of femtocells (Figure 3-15) and the distributed routing testbed (Figure 3-16).

Task Id	Start	End	Task Description
T1	Jun-10	Aug-10	Identification and aquisition of luh enabled femtocells
T2	Aug-10	Dec-10	Upgrade and configuration of the Emulated Core Network (ECN)
T3	Dec-10	Feb-11	Integration of a femtocell and the ECN
T4	Feb-11	Apr-11	Integration of a UE and the ECN
T5	May-11	Jun-11	Integration and performance analysis of 1UE + 1femtocell + ECN through wired backbone
T6	Jul-11	Oct-11	Integration and performance analysis of 12UEs + 12femtocells + ECN through wired backbone
T7	Jul-11	Sep-11	Integration and performance analysis of 1UE + 1femtocell + ECN through wireless backbone
T8	Nov-11	Dec-11	Integration and performance analysis of 12UEs + 12femtocells + ECN through wireless backbone

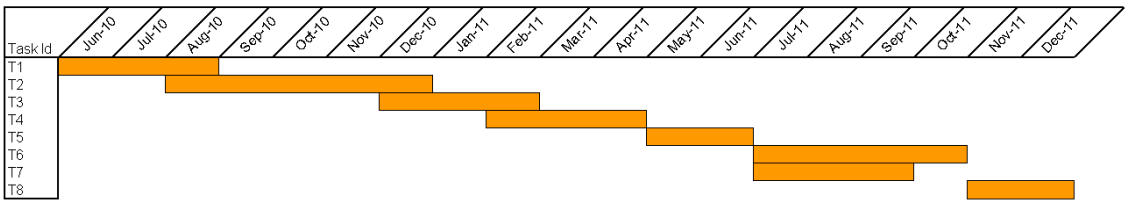

Figure 3-15: Testbed 2 (network of femtocells) work plan

Task Id	Start	End	Task Description
T1	May-11	Jul-11	Integration of dynamic backpressure routing with Ns3-emulation framework
T2	Aug-11	Oct-11	Integration of dynamic backpressure routing with netlink sockets and /proc subsystem
T3	Sep-11	Jun-12	Evaluation of the dynamic backpressure routing proof-of-concept
T4	Jan-12	Jun-12	Dynamic backpressure routing demo preparation

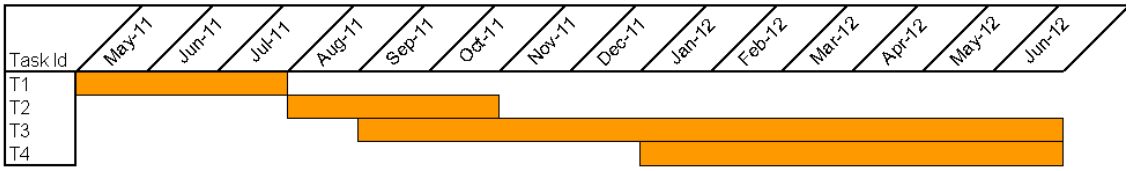

Figure 3-16: Testbed 2 (distributed routing) integration work plan

3.2 Iuh-Tap

The communication between a femtocell and the mobile core network (MCN) is carried over the local enterprise network and the Internet. This makes it absolutely essential to secure the traffic from the femtocell until the gateway of the MCN. It is implemented typically as IPsec tunnel between the femtocell and a special secure gateway. Over the secure connection the whole traffic of the femtocells flows to the operator. Therefore, it is impossible to apply routing optimisation or traffic offloading, without breaking the bound of trust.

To avoid the unsecured communication over the Internet, the enterprise network needs a secure gateway as well. This is called the Local Femtocell Gateway (LFGW). Between these two secure gateways the traffic is tunnelled over IPsec. Inside the enterprise network the connection between the femtocell and LFGW could be secured with IPsec, depending on security demands. The principle mode of operation for the LFGW is described in [14].

This structure allows the LFGW to tap into the communication of the femtocells and can moreover also be an interconnection between different MCNs inside the enterprise network. The traffic of the femtocells cuts into two different types:

- **Control plane**: To exchange control information about connected UE's and status updates.
- **User plane**: The traffic from the connected UE's (voice, IP traffic …).

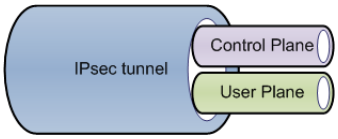

Figure 3-17: Transport scheme of femtocell traffic

For the effort of an additional system inside an enterprise network, the LFGW allows routing optimisation and centralized traffic offloading before the traffic reaches the MCN. Moreover it is a connector between the MCN and the enterprise network to observe the link quality. The benefit has to be evaluated by simulation [12] and experimentally in the testbed (using 3G femtocells).

3.2.1 Short Description

The Iuh-Tap is located on the LFGW. As introduced, this is in worst case security scenario the only point inside the enterprise network, where the traffic of the femtocell is unencrypted. This allows tapping into the Iuh interface on types: the control and the user plane.

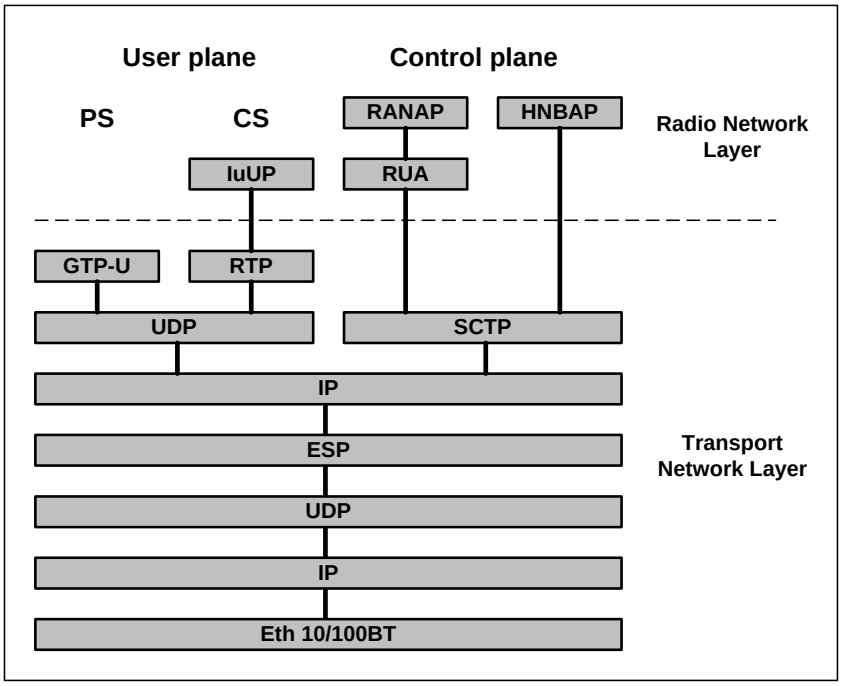

Figure 3-18: Iuh protocol stack

The control plane is transported over SCTP and contains different types of protocols to exchange event information between the femtocell and the MCN. Typical HNBAP events are registrations of femtocells and UE's. The RANAP event contains bearer information of the user plane with QoS and bandwith parameters. The Iuh-Tap observes the HNBAP and RANAP signalling and generates events for other modules. They previously need to subscribe to such events and will be notified, if it occurs. This external event interface is implemented as RESTful interface and allows message exchange over HTTP.

On the user plane, it is capable of re-routing traffic passing through the Iuh-Tap, for example to implement a traffic offloading function or a local connection between two UE's.

Moreover it generates traffic statistics and offers the possibilty to create HNBAP messages for management towards the Mobile network Operator (MNO) or to the femtocell.

3.2.2 Demo Scenario and KPIs

This section provides an overview of the demo scenarios envisaged at this stage of the project. In particular it highlights the use cases and functionality that will be demonstrated.

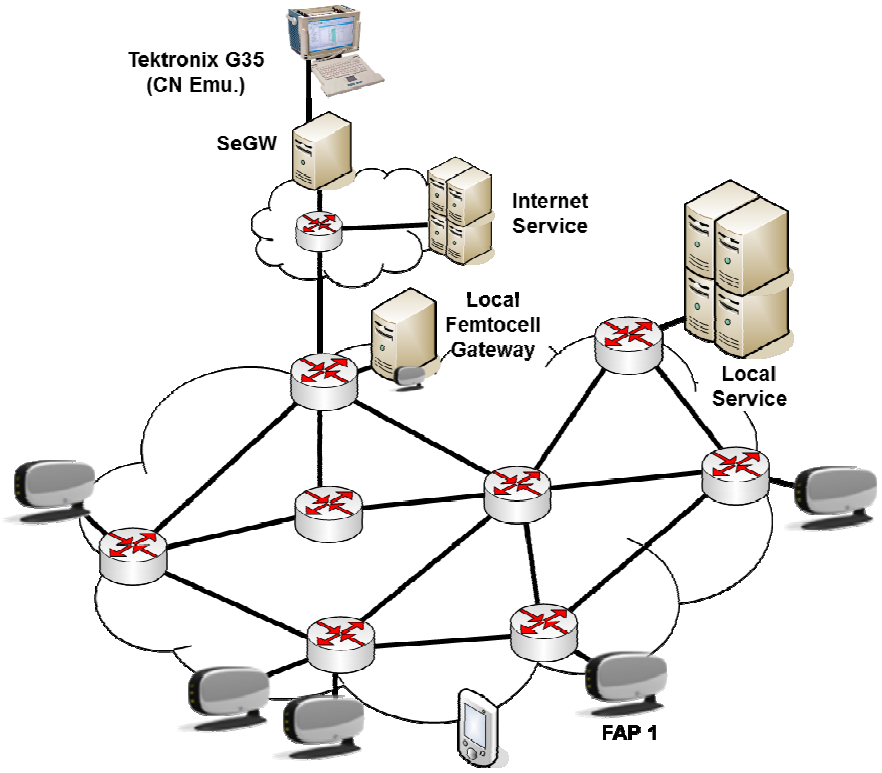

Figure 3-19: Assumed testbed setup

The assumed test-bed for the Iuh-Tap (Figure 3-19) consists of the LFGW which has an IPsec tunnel to the secure gateway of the simulated MCN. Moreover allows the LFGW direct connections to the Internet. The simulated enterprise network is a meshed network of openFlow switches and has several femtocells, connected to different ports and switches of the meshed network. The openFlow switches have a controller, which is also located on the LFGW.

3.2.2.1 Traffic Breakout

- The first demo (Figure 3-20) will show the traffic breakout of an Internet connection. A connected UE establishes a streaming session. Without breakout capabilities of the LFGW flows the traffic of the stream through the MCN and is detectable on the G35. After the traffic breakout is active for the UE, the LFGW stripes of the data from the user plane and send it directly to the Internet. The responses have to filter out by the LFGW and extended with a GTP header. This is transparent for the UE and the MCN did not spend resources for this traffic.

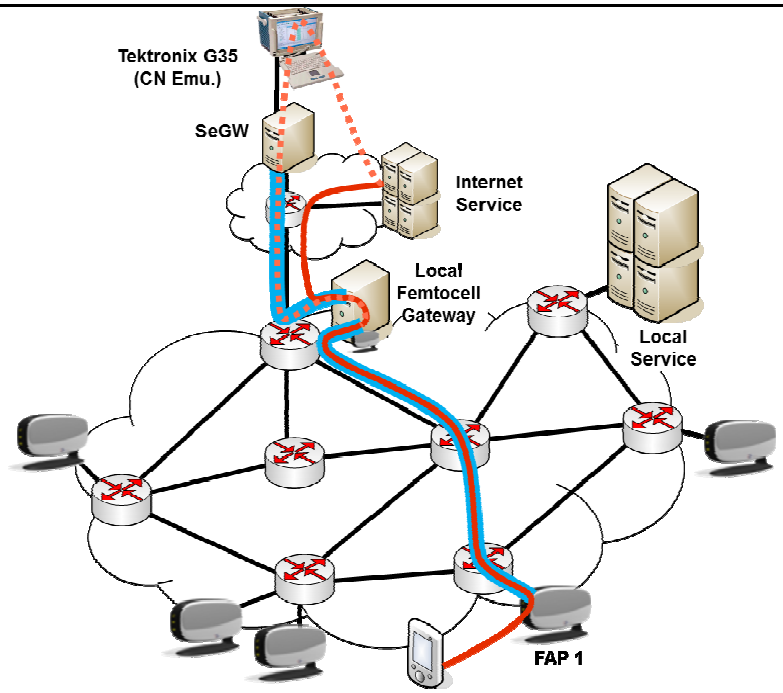

Figure 3-20: Demo 1, Traffic Breakout

- **KPIs**: The Iuh-Tab acts as an enabler for other demo functionality and is not meant to be production quality and performance optimized. For the purpose of the test-bed, the primary KPI is whether it correctly implements the required functionality or not. For analysis reasons the throughput, latency and system load could be measured.

3.2.2.2 Traffic Routing

- The second demo (Figure 3-21) will show the traffic routing between two UE's. Usually it is not possible to establish a data connection between two UE's without traffic routing over the MCN. The LFGW can exchange the user plane information of the traffic and allows a local routing. The traffic of this connection will never pass the MCN (see also Section 3.2.4.1).

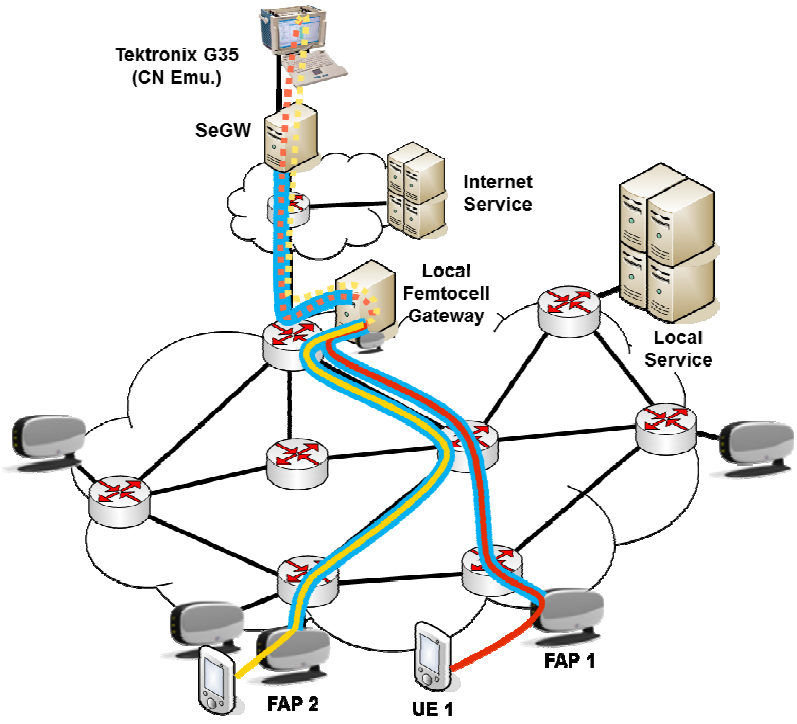

Figure 3-21: Demo 2, Traffic Routing

- **KPIs**: The Iuh-Tab acts as an enabler for other demo functionality and is not meant to be production quality and performance optimized. For the purpose of the test-bed, the primary KPI is whether it correctly implements the required functionality or not. For analysis reasons the throughput, latency and system load could be measured.

3.2.2.3 Traffic Control

- The third demo (Figure 3-22) will show the traffic control inside the enterprise network. The openFlow technology allows transferring connection flows to avoid the overload of a link inside the network. For a Mobile Operator the guarantee of link capacities is important for the QoS. To show this, the UE will establish a connection to an Internet service. This connection will be affected by cross traffic. The openFlow controller detects the critical bandwidth of the link and tries to transfer the flow to another link.

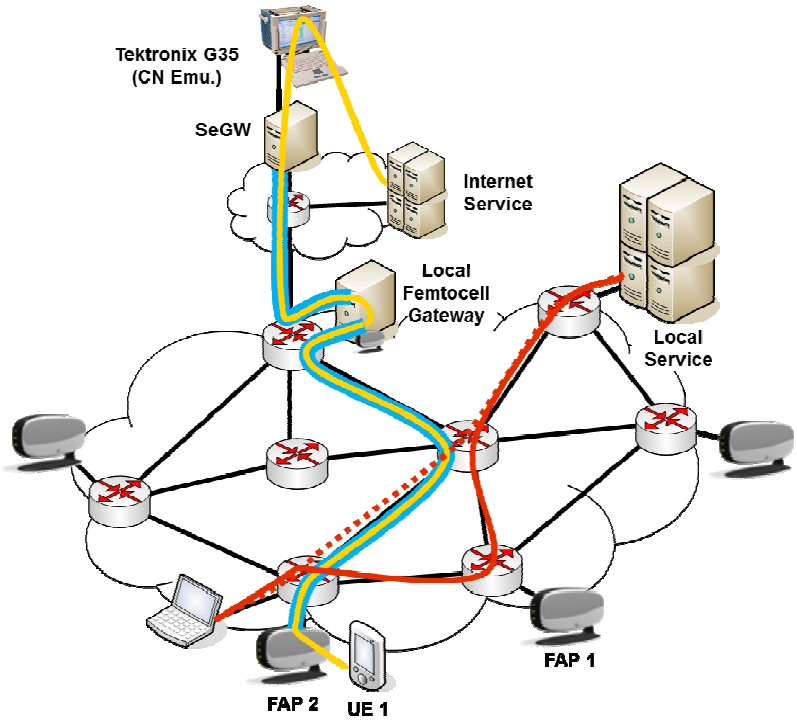

Figure 3-22: Demo 3, Traffic Control

- **KPIs**: The Iuh-Tab acts as an enabler for other demo functionality and is not meant to be production quality and performance optimized. For the purpose of the test-bed, the primary KPI is whether it correctly implements the required functionality or not. For analysis reasons the throughput, latency and system load could be measured. Moreover the link capacity and the structure of the meshed network could be analysed.

3.2.3 Key Building Blocks and Integration Specification

3.2.3.1 Key Building Blocks

The following is a list of the key enabling building blocks required for the demo scenarios described above.

- The **Iuh-Tap framework**, which is the basic infrastructure needed to insert into the Iuh interface. Contains, e.g., functions to proxy and intercept SCTP connections and GTP tunnels on the control and data planes, respectively.
- The **HNBAP/RANAP Dissector** sits on top of the Iuh-Tap framework and processes passing signalling messages. It also supports an interface on which other modules can register to receive event notifications and usage statistics. For example, such modules can be the Traffic Management Function and Fault Diagnosis.
- The **Traffic Re-routing Function** sits on the data plane and is capable of re-routing incoming user traffic from the femtocells, the core network and the Traffic Offload Function (i.e. from the

local IP network) and forward it, potentially on a different bearer, to the core network, the femtocells or the Traffic Offload Function.

- The **Traffic Offload Function** works closely together with the Traffic Re-routing Function and performs the IP address allocation for the user traffic within the local IP network and performs a network address translation on the traffic if necessary.
- The **openFlow meshed Network** is the basic infrastructure to allow traffic control inside the enterprise network. It contains several openFlow switches (virtual or real hardware) and enables advanced network functionality.
- The **openFlow Controller** is the management part for the openFlow network and is also located on the LFGW. The main function of the controller is to hold the actual state of the network, integrate centralised routing algorithm and configuration of the network.
- The **Traffic Management Function** controls both the Traffic Re-routing Function and the Traffic Offload Function, based on the events received from the HNBAP/RANAP Dissector and operator policies. Moreover it could interact with the openFlow Controller to monitor the link capacities of the network.

These building blocks and their inputs and outputs to other building blocks are summarised in the Table 3-1 below.

Table 3-1: Key Building Blocks of the LFGW

Key Building Block	Functionalities	Input	Output
Iuh-Tap Framework	The infrastructure needed to insert into the Iuh interface between the femtocells and the core network, e.g. to proxy the SCTP connections on the control plane.		
HNBAP/RANAP Dissector	Observes passing HNBAP/RANAP signalling messages and generates events and statistics.	HNBAP/RANAP signalling messages	Signalling events and statistics
Traffic Re-routing Function	Sits on the data forwarding path and potentially re-routes traffic between bearers or to/from Traffic Offload Function	IP traffic from femtocells, core network and/or Traffic Offload Function; control messages from the Traffic Management Function	IP traffic to femtocells, core network and/or Traffic Offload Function; traffic statistics to Traffic Management Function
Traffic Offload Function	Allocates a local IP address for offload traffic and performs NATting.	IP traffic from local network or Traffic Re-routing Function; control messages from the Traffic Management Function	IP traffic to local network or Traffic Re-routing Function
openFlow Network	The infrastructure to allow advanced network functionality.		
openFlow Controller	Observes the network and configure the openFlow switches.	control messages from the Traffic Management Function	Link events and statistics
Traffic Management Function	The control logic behind Traffic Re-routing and Offloading.	Events and statistics from HNBAP/RANAP dissector; traffic statistics from Traffic Re-routing Function; Operator policies	Control messages to Traffic Re-routing and Traffic Offload Functions

The following interfaces have thus to be implemented:

- Interface between the HNBAP/RANAP Dissector in the Iuh-Tap and the Traffic Management Function.
- Interface between the Traffic Management Function and the Traffic Re-routing Function.
- Interface between the Traffic Management Function and the Traffic Offload Function.
- Interface between the Traffic Management Function and the openFlow Controller.

All above-mentioned Interfaces are integrated as RESTful interface and follows the "publish-subscribe" model. This allows the configuration and monitoring of the Key Building Blocks over a web browser.

3.2.3.2 Key Building Blocks Integration into Modules

The table below (Table 3-2) provides a high level overview of the building blocks required for each demo. It should be noted that the complexity of the implemented building block will be dependent on the final demo scenario which is chosen.

Table 3-2: Building Blocks Vs Demos

Building Block	Demo 1	Demo 2	Demo 3
Iuh-Tap Framework	X	X	
HNBAP/RANAP Dissector	X	X	
Traffic Re-routing Function	X	X	
Traffic Offload Function	X		
openFlow network			X
openFlow Controller			X
Traffic Management Function	X	X	X

The main building blocks from the table above are combined to modules, depending on their function. Therefore, from the software architecture point of view the Iuh-Tap consist of four modules (Figure 3-23). The Iuh-Tap Framework and the HNBAP/RANAP dissector are combined to the *C plane Tab module*. Both traffic functions are implemented in the *U plane module*.

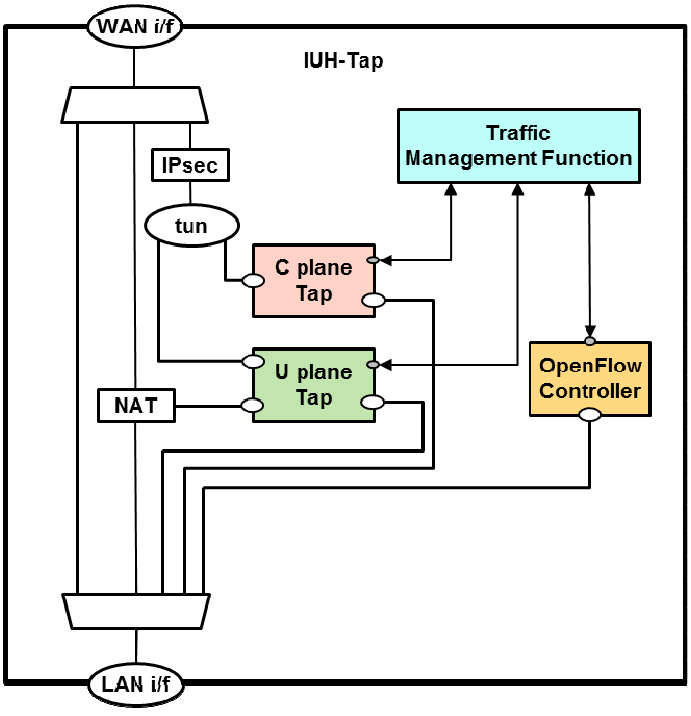

Figure 3-23: Architecture of the Iuh-Tap

3.2.4 Traffic Offload and Local Routing

One of the functionalities of the Iuh-Tap is the traffic offloading. The management of the traffic offloading can be transparent to the operations of the mobile core network and the femtocells. To achieve this, new functionalities are added to the LFGW.

On the control plane, the HNB-GW and the femtocell send regular signalling messages through the SCTP connections established between them. The LFGW acts as a simple proxy between them. On the user plane, both the HNB-GW and the femtocell, follow the regular procedures, i.e., the HNB-GW establishes the GTP tunnels with the femtocells. The GTP tunnel can be intercepted at intermediate nodes (Wireless Mesh Routers (WMRs) or LFGW) for local routing and/or data offloading. Depending on where the traffic offload module is located, the traffic offloading process can be centralized or distributed.

3.2.4.1 Centralized Traffic Offloading

When the Traffic Offloading module is located in the LFGW, it can be used for data offloading or centralized routing. In this case, the LFGW is the only element offloading data from the femtocells.

For each femtocell of the NoF, the LFGW breaks the original GTP tunnel between the HNB-GW and the femtocell. The LFGW manages the two *parts* of the tunnel: there is one *part* between the HNB-GW and the LFGW. This part of the tunnel is only used for external traffic but it is not used for traffic local to the NoF. If there is no external traffic, the LFGW maintains this part to avoid that the HNB-GW closes the tunnel with the femtocell. There is another *part* between the LFGW and the femtocell. This part is local to the NoF. The mechanisms implemented in the LFGW are transparent from the point of view of the mobile core network and the femtocells. They still see the same original tunnels.

When two UEs in the NoF send data between them, the traffic goes through the LFGW. It forwards the local UE data between the two local parts of the GTP tunnels. If the traffic is for an external user from the NoF, the traffic is forwarded towards the HNB-GW. The LFGW maintains the original GTP tunnel with the HNB-GW (with keepalive packets, if needed) so that resources are not released in the mobile core network.

The following figure (Figure 3-24) shows one data connection between two UEs attached to the NoF. This data connection results in two GTP tunnels (green and blue lines) between the UEs and the mobile core network. The LFGW keeps the original GTP tunnels with the HNB-GW (dashed lines) and forwards the data between the two local GTP tunnels (solid lines).

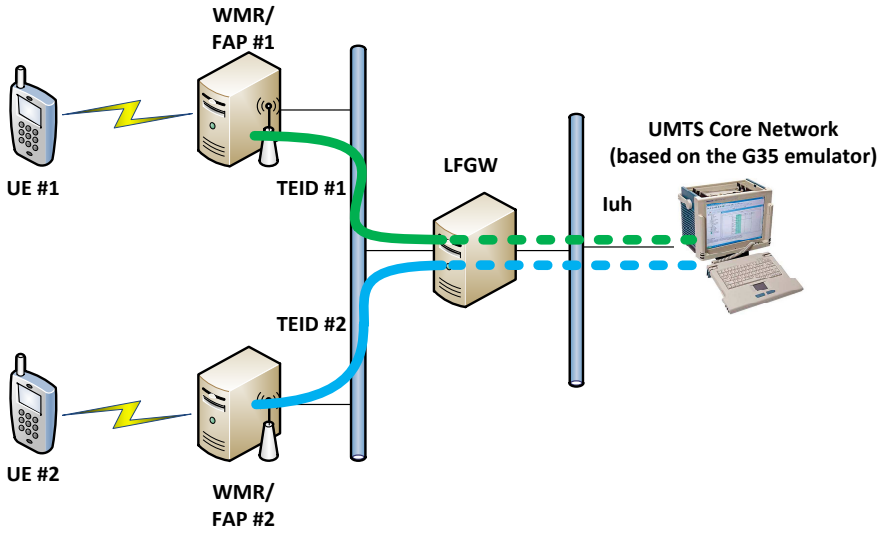

Figure 3-24: Traffic offloading centralized in the LFGW

3.2.4.2 Distributed Traffic Offloading

When the Traffic Offloading module is deployed at each femtocell of the NoF, it can be used for local routing or local data offloading. In this case, each femtocell can offload data from the UEs attached to it.

For each femtocell of the NoF, the WMR (that is part of the wireless femtocell) breaks the original GTP tunnel between the HNB-GW and the femtocell. The WMR manages the two parts of the tunnel: there is one part between the HNB-GW and the WMR. This part of the tunnel is only used for external traffic but it is not used for traffic local to the NoF. If there is no external traffic, the WMR maintains this part to

avoid that the HNB-GW closes the tunnel with the femtocell. There is another part between the WMR and the femtocell. This part is local to the wireless femtocell (i.e., between the WMR and the femtocell). The mechanisms implemented in the WMR are transparent to the mobile core network and the femtocells. They still see the same original tunnels.

When the UEs in the NoF send data between them, the traffic crosses the WMR. It sends the local UE data to a remote WMR through the NoF using the distributed backpressure routing or any other routing solution but the data do not pass through the LFGW. Only if the traffic is for an external user, then the traffic is forwarded towards the LFGW and the HNB-GW. The WMRs maintain the original GTP tunnel with the HNB-GW (with keepalive packets, if needed).

The following figure (Figure 3-25) shows one data connection between two UEs attached to the NoF. This data connection results in two GTP tunnels (green and blue lines) between the UEs and the mobile core network. The WMRs keep the original GTP tunnels with the HNB-GW (dashed lines) and forwards data to the NoF using the distributed backpressure routing (orange line). When the destination WMR receives the data, if forwards the data to the GTP tunnel (solid line) established with the local femtocell.

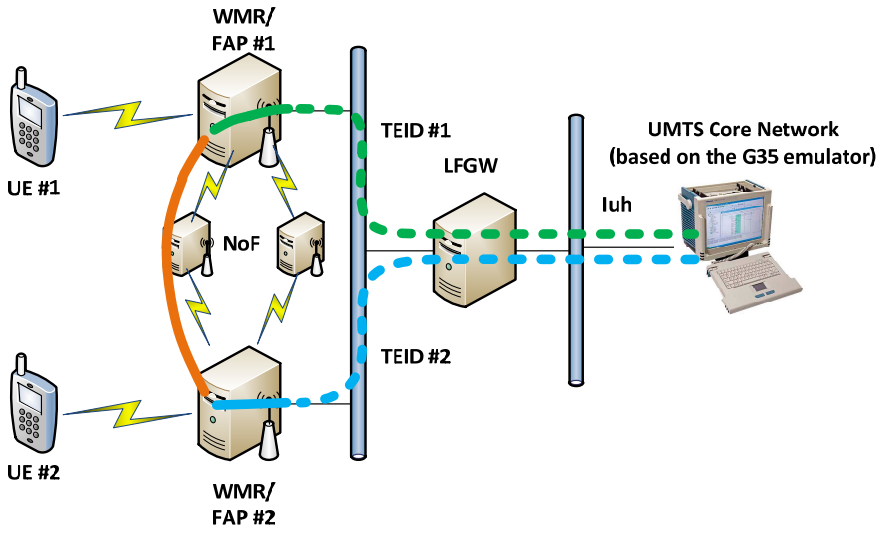

Figure 3-25: Traffic offloading distributed between the WMRs

3.2.4.3 Implementation of the Traffic Offloading module

The following figure (Figure 3-26) shows a block diagram with the implementation sub-modules of the Traffic Offloading:

- The *Traffic Management* sub-module captures and filters the GTP traffic in the LFGW. The data packets between the source and destination UE are encapsulated in GTP tunnels between the femtocells and the HNB-GW. The GTP traffic goes from the *source* femtocell to the HNB-GW and from the HNB-GW to the *destination* femtocell. The source UE is attached to the *source* femtocell and the destination UE is attached to the *destination* femtocell. The HNB-GW receives GTP packets from the *source* femtocell (from the GTP tunnel identified by the Tunnel Endpoint Identifier $TEID_{UPLINK}$) and forwards them to the *destination* femtocell (through the GTP tunnel identified by the Tunnel Endpoint Identifier $TEID_{DOWNLINK}$).
- The *TEID database* stores the mapping between the pair of source and destination UE of the data flow and the information about the GTP tunnels ($TEID_{UPLINK}$ and $TEID_{DOWNLINK}$) used for the data flow in uplink, i.e., the GTP tunnel between the *source* femtocell and the HNB-GW and in downlink, i.e., the GTP tunnel between the HNB-GW and the *destination* femtocell.
- The *Tunnels Management* sub-module learns the information from the uplink GTP tunnel and the downlink GTP tunnel and populates the *TEID database*. Moreover, it *breaks* the uplink GTP tunnel and *creates* the downlink GTP tunnel, i.e., when it receives the GTP packets from the uplink GTP tunnel, if the source and destination UE mapping is stored in the *TEID DB*, then it de-encapsulates the user data and generates the new downlink GTP packets. If the mapping is still not known, the GTP packet continues towards the HNB-GW. Finally, it keeps the *original* uplink and downlink GTP tunnels established with the HNB-GW.

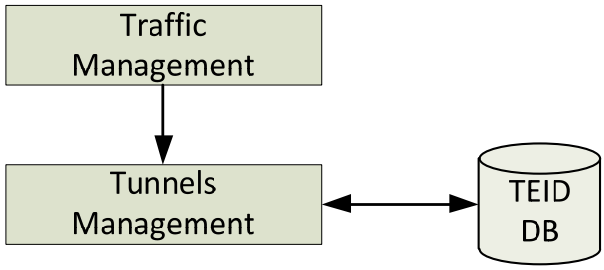

Figure 3-26: Implementation sub-modules of the Traffic Offloading

3.2.5 Work Plan

The following Gantt charts (Figure 3-27) summarize the tasks to set-up the Iuh-Tap demonstration within Testbed 2.

Task Id	Start	End	Task Description
T1	May-11	May-11	Set up of the local testbed with LFGW and Sagemcom Femtocells
T2	May-11	May-11	Set up of tunnel to CTTC testbed and emulator
T3	May-11	Jun-11	Development of Iuh-Tap Framework
T4	May-11	Jul-11	Development of HNBAP Dissector
T5	Aug-11	Aug-11	Integration test Iuh-Tap Framework and HNBAP Dissector
T6	Aug-11	Aug-11	Development of basic Traffic Management Function
T7	Sep-11	Sep-11	Integration test HNBAP Dissector with Traffic Management Function
T8	Oct-11	Oct-11	Development of Traffic Re-routing Function
T9	Oct-11	Oct-11	Development of Traffic Offloading Function
T10	Nov-11	Dec-11	Integration of Traffic Re-routing Function and Traffic Offloading Function with Traffic Management Function and Testing
T11	Jan-12	Mar-12	Iterative refinement
T12	Apr-12	Jun-12	Integration into Testbed 2

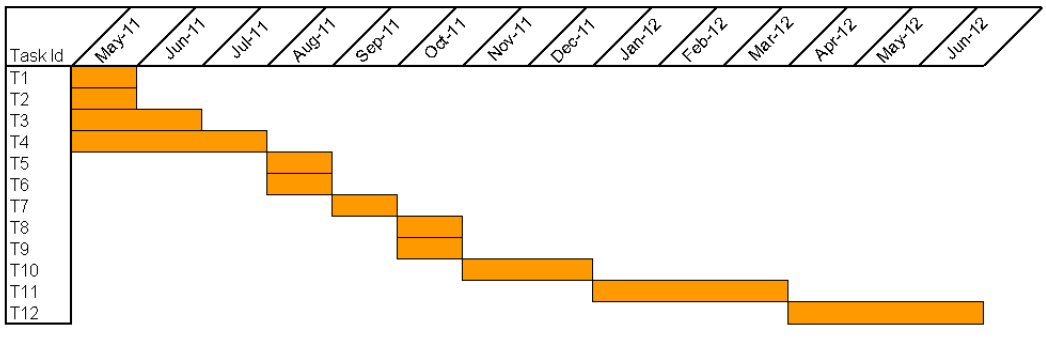

Figure 3-27: Testbed 2 (Iuh-Tap) integration work plan

4. Testbed 3, Multi-Radio FemtoNode Authentication in the Fixed Access Network

The main objective of Testbed 3 (Figure 4-1) is to validate and demonstrate the authentication of the FemtoNode subscriber by inserting a removable Universal Integrated Circuit Card (UICC) card in order to allow an authentication procedure regardless the geographical location and the access network physical infrastructure. New functionalities that can be provided by a Multi-Radio FemtoNode will also be investigated. Moreover, the self-configuration of the access network elements will be demonstrated too.

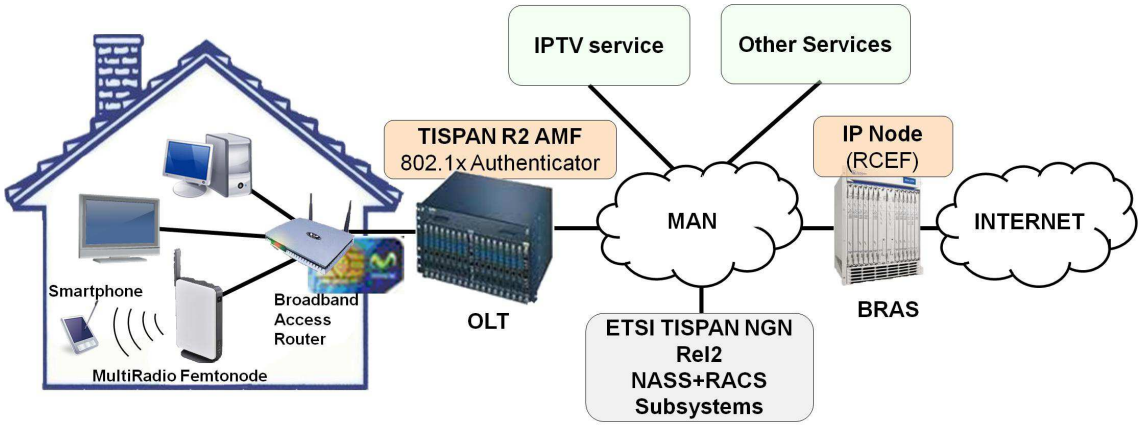

Figure 4-1: Testbed 3 architecture

4.1 EAP-AKA over 802.1X Algorithm

4.1.1 Short Description

EAP-AKA over 802.1X is the selected algorithm to authenticate the subscriber to the fixed backhaul and configure it in real time. The criteria for choosing this solution involve the following advantages:

- It is highly probable that the femtocell node will provide the software and hardware elements required to run EAP-AKA protocol stack.
- The IEEE 802.1X protocol, which is the standard for EAP authentication over Ethernet, is currently quite popular due to its intensive usage in Wi-Fi (since it is the core of WPA authentication) and counts with enough support in access network equipments: commercial DSLAMS or Optical Line Terminations (OLTs) from various manufacturers.

The algorithm developed covers the following situations:

- Initial attachment: the user inserts the UICC card for first time.
- Full Re-authentication: IMSI is used and fresh authentication vectors are generated to avoid security threads.
- Fast Re-authentication: TIMSI is used to avoid overloading the network and detect whether or not the UICC card is removed.

The subscriber authentication algorithm is fully described in BeFEMTO D5.1, section 5.1 [13].

4.1.2 Demo Scenario and KPIs

Testbed 3 will implement two different demo scenarios, each one holding different scenarios.

4.1.2.1 Secure Loose-Coupled Authentication of the Femtocell Subscriber

- In this demo scenario, the home access network must be configured with the services subscribed by the user. Two possible situations have been distinguished:
    - Recognition of a valid/invalid card by the Network. The network must be able to recognize if a valid card is inserted in the card reader.

- Recognition of different subscribers from the same home. The network must be able to differentiate between users and their subscribed services.

- **KPIs**: No KPIs are needed for the "Secure Loose-Coupled Authentication of the Femtocell Subscriber" use case. The **key message** would be that the flexibility and dynamism of this kind of authentication enables the slogan "My home moves with me".

4.1.2.2 Multi-Radio FemtoNode as enabler of a new generation of services

- A demonstration is proposed in which the Multi-Radio FemtoNode (MRFN) radio interfaces, 3.5G, Wi-Fi, RFID, ZigBee and Bluetooth, will be jointly used. More specifically, user identification based on RFID and home surveillance will be demonstrated.

    - User identification (by means of RFID) and home services personalization. The multi-radio FemtoNode will identify the customer by means of RFID tag in the user terminal. It will send a welcome message to the user terminal by means of a Bluetooth connection, and it will upload the user photograph, stored in the terminal to a digital frame, by means of Bluetooth connection.

    - Home surveillance from the user terminal. The user terminal is connected in remote or local to the multi-radio FemtoNode by means of 3.5G (or Wi-Fi), and through the MRFN he will access a home web server, to check home alarms and status of sensor nodes and also actuate on some sensors. Then using his mobile, the user can drive a small robotic vehicle (ROVIO) that transmits video in real time. The video and the robotic vehicle are connected to the MRFN via a Wi-Fi link.

- **KPIs**: The **key message** would be "the Multi-Radio FemtoNode is the enabler of new generation of advanced services".

4.2 Key Building Blocks and Integration Specification

4.2.1 Key Building Blocks

For the authentication part, three Key Building Blocks have been identified:

- **Customer Network Gateway (CNG)**. Provides the functionalities of an Optical Network Terminator (ONT), Broadband Access Router or Residential Gateway (RG) and a Multi-Radio FemtoNode. As currently there is neither a Broadband Access Router nor femtocell equipment supporting UICC cards insertion and management, an external pluggable USB Smart Card Reader will be used. This device will be plugged into the PC simulating the femtocell (since the Broadband Access Router does not offer either USB connections or UICC cards).

- **Network Attachment Sub-System (NASS)**. For testing the authentication, only the minimum and essential set of entities and functionalities from the NASS needed for a proof of concept will be implemented. This functionalities are:

    - Network Access Configuration Function (NACF).

    - Access Management Function (AMF).

    - User Access Authorization Function (UAAF).

    - Connectivity session Location and repository Function (CLF).

- **Resource and Admission Control Subsystem (RACS)**. For testing the authentication, only the minimum and essential set of entities and functionalities from the RACS needed for a proof of concept will be implemented. This functionalities are grouped into policy decision points and policy enforcement points and are the following:

    - Access - Resource Admission Control Function (A-RACF).

    - Service-based Policy Decision Function (SPDF).

    - Resource Control Enforcement Function (RCEF).

    - IP Edge Node (BRAS).

    - Access Node (OLT).

Table 4-1 hereafter summarises their functionalities and input/output.

Table 4-1: Authentication KBBs

Key building Block	Functionalities	Input	Output
Customer Network Gateway	• ONT • Broadband Access Router • PC simulating multi-radio FemtoNode • External pluggable Smart Card reader	The subscriber introduces the UICC card into the Broadband Access Router	The authentication procedure is triggered
NASS subsystem	• CLF • NACF • UAAF • AMF	NASS receives from CNG 802.1x authentication packets	- As a consequence of a successful authentication procedure, NACF provides the CNG with a public IP address. - As a consequence of a successful authentication procedure, CLF conveys to the RACS the policies to be enforced in the RCEF for this subscription.
RACS subsystem	• A-RACF • RCEF • SPDF • Access Node • IP Edge Node	RACS receives from the NASS the policies to be enforced for a subscription in the RCEF	A-RACF commands the enforcement of the policies in the RCEF

The following is the list of the key building blocks required for the MRFN demo scenario, in which multiple radio interfaces will be used.

- **Modem router ADSL**, to which the MRFN is connected.
- **MRFN** (A Linux PC with Linux Ubuntu 10.04) with following interfaces: 6 USB ports, in which are connected: HSPA, RFID, Bluetooth, ZigBee and UWB radio interfaces, 2 Ethernet ports, to which is connected a Wi-Fi Access Point.
- Drivers and applications to manage and bridge the aforementioned radio interfaces.

4.2.2 Interfaces

The main interfaces and responsibilities involved in the architecture are described below.

4.2.2.1 User Equipment Authentication Interfaces

The equipment deployed at a customer premise (Figure 4-2) is composed of a User Equipment (UE) and the Customer Network Gateway (CNG). In TID's testbed the CNG is implemented by the PC simulating the FemtoNode functionalities, the Optical Network Unit (ONT), and the Broadband Access Router.

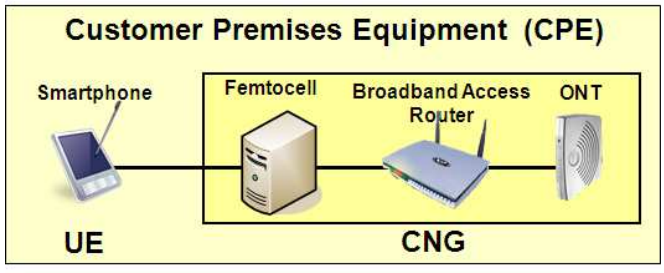

Figure 4-2: Customer Premise Equipment

- **Reference point CNG-AMF**. This reference point enables the FemtoNode to initiate requests for IP address allocation and to trigger the authentication of the FemtoNode Subscriber, in order

to access the network. These requests are received by the AMF entity implemented in the OLT. This reference point will support several interfaces:

  - One of them, to trigger access network authentication and IP address allocation. This interface will not be implemented following ETSI TISPAN definition but the proprietary interface provided by the OLT vendor.
  - The second one, to support exchange of 802.1X messages between the Femto Node and the OLT (Access Relay Function (ARF) + AMF).

- **Reference point CNG-UAAF**. This reference point enables the FemtoNode to exchange EAP messages with the UAAF for authentication purposes. As stated previously, it will implement the EAP protocol.

4.2.2.2 Network Attachment Sub-System

The Network Attachment Sub-System (NASS) (Figure 4-3) provides registration at access level and initialization of User Equipment (UE) for accessing to the TISPAN Next Generation Networks (NGN) services. The NASS provides network level identification and authentication, manages the IP address space of the Access Network and authenticates access sessions. The NASS also announces the contact point of the TISPAN NGN Service/Applications Subsystems to the UE.

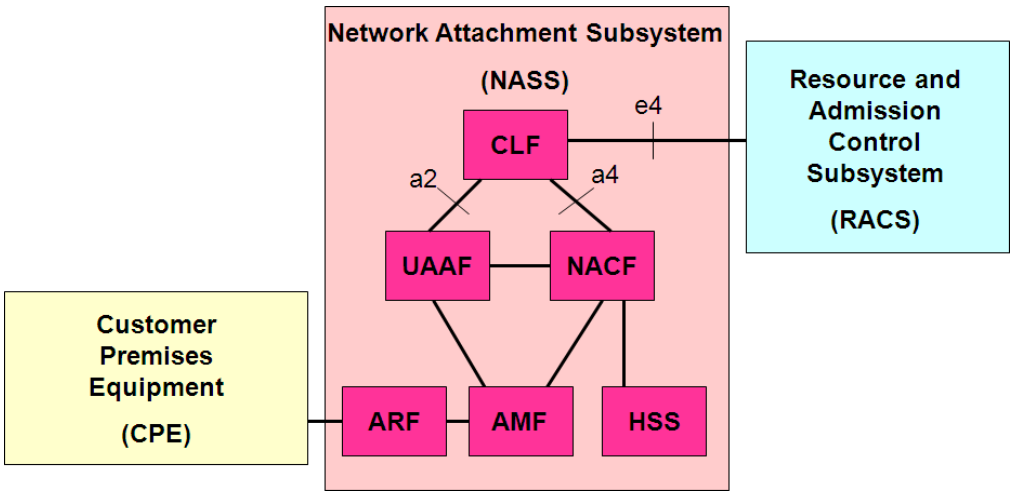

Figure 4-3: NASS architecture

The main responsibilities of the interfaces to be implemented in this testbed are the following:

- **Reference point NACF-CLF (a2)**. This reference point allows the NACF to register in the CLF the association between the allocated IP address of the NASS User Identity and the related location information (IP edge ID, Line ID). The a2 interface is based on the Diameter protocol.
- **Reference point UAFF-CLF (a4)**. This reference point allows the CLF to register the association between the NASS User Identity and the NASS User preferences regarding the privacy of location information provided by the UAAF. Reference point a4 is also used to register NASS User network profile information (QoS profile). The CLF may retrieve the NASS User network profile from the UAFF. Interface a4 is based on the Diameter protocol.
- **Reference point NACF-AMF (a1)**. This reference point allows the AMF to request the NACF for the allocation of an IP address to user equipment as well as other network configuration parameters. This interface will not be implemented following the ETSI TISPAN definition but the proprietary interface provided by the OLT vendor.
- **Reference point UAAF-AMF (a3)**. This reference point allows the AMF to request the UAAF for NASS User authentication and network subscription checking. This interface will not be implemented following the ETSI TISPAN definition but the proprietary interface provided by the OLT vendor.
- **Reference point CLF-RACF (e4)**. This reference point is used to pass the association between the Globally Unique Address and/or NASS User ID on the one hand, and the Access Identifier (logical or physical) on the other hand, from the CLF to the RACS. This allows RACS to determine the amount of available network resources. The e4 reference point may also be used to

pass QOS profile information and initial gate settings from the CLF to the RACS. This allows RACS to take them into account when processing resource allocation requests. e4 interface is based on the Diameter protocol.

4.2.2.3 Resource and Admission Control Sub-System

RACS (Figure 4-4) is the NGN Subsystem responsible for policy control, resource reservation and admission control. In addition, it also supports core Border Gateway Services (BGS) including Network Address Translation (NAT) mechanisms.

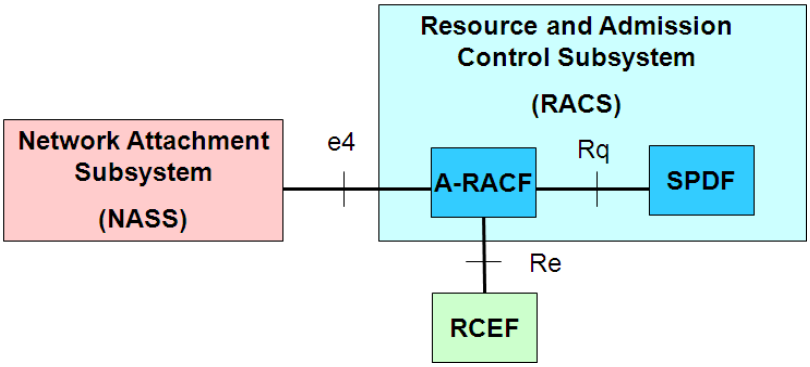

Figure 4-4: RACS architecture

The main responsibilities of the interfaces to be implemented in this testbed are the following:

- **Reference point SPDF-A-RACF (Rq)**. The Rq reference point is used for QoS resource reservation information exchange between the SPDF and the A-RACF. Via the Rq reference point the SPDF issues requests for resources in the access and aggregation networks, indicating IP QoS characteristics. Rq interface is based on the Diameter protocol.
- **Reference point A-RACF-NASS (e4)**. The definition of this interface is already described in the NASS interface description in the previous section.
- **Reference point A-RACF-RCEF (Re)**. The Re reference point is used for controlling the L2/L3 traffic policies performed in the transport plane, as requested by the resource management mechanisms, i.e. gating, packet marking, traffic policing and mid-session updates functionalities. Re interface is based on the Diameter protocol

4.2.3 Relation to Demo Scenarios

The table below (Table 4-2) summarizes the information and provides a high level overview of the building blocks required for each demo.

Table 4-2: KBB relation to the scenarios

Scenario	Key building block	Minimum Functionality
Secure, Loose-Coupled Authentication of the Femtocell Subscriber	Customer Network Gateway	ONT Residential Gateway PC simulating FemtoNode External pluggable Smart Card reader
	NASS subsystem	CLF NACF UAAF AMF
	RACS subsystem	A-RACF RCEF SPDF Access Node IP Edge Node
Demonstration of advanced services enabled by the MRFN	MRFN equipped with different radio interfaces	MRFN equipped with different radio interfaces

4.3 Work Plan

Testbed 3 will be the result of the work performed by two groups of TID. Thus, Testbed 3 will comprise the developments performed with regard to two solutions.

- Subscriber authentication solution
- Multi-Radio FemtoNode solution

These two solutions will demonstrate the physical parameter independence for the subscriber authentication as well as the real time configuration of the access network for the services demanded by the subscriber, allowing the multi-radio FemtoNode to provide such services to the end user throughout a wide variety of radio interfaces.

To this end, a work plan has been defined to ensure the proper coordination for both groups and to establish the due dates in which these solutions will be delivered and integrated within a single testbed. Three tasks have been identified until Testbed 3 will be ready for a demonstration. More precisely, the Table 4-3 schedules the completion of Testbed 3 (three tasks identified), while Figure 4-5 summarizes the work plan in the form of a Gantt chart.

Task Id	Start	End	Task Description
T1	May-11	Sep-11	Development and finalisation of each KBB
T1.1.1	May-11	May-11	CPE
T1.1.2	May-11	May-11	NASS
T1.1.3	May-11	May-11	RACS
T1.2.1	May-11	May-11	Femto connection to xDSL router through PC
T1.2.2	May-11	Jun-11	Radio interfaces integration in PC
T1.2.3	May-11	Sep-11	multi-radio SW
T2	Oct-11	Nov-11	Integration of each solution
T2.1	Jan-00	Nov-11	Subscriber Authentification Solution
T2.2	Oct-11	Nov-11	Multi-radio node
T3	Dec-11	Jan-12	Integration of the system within Testbed 3

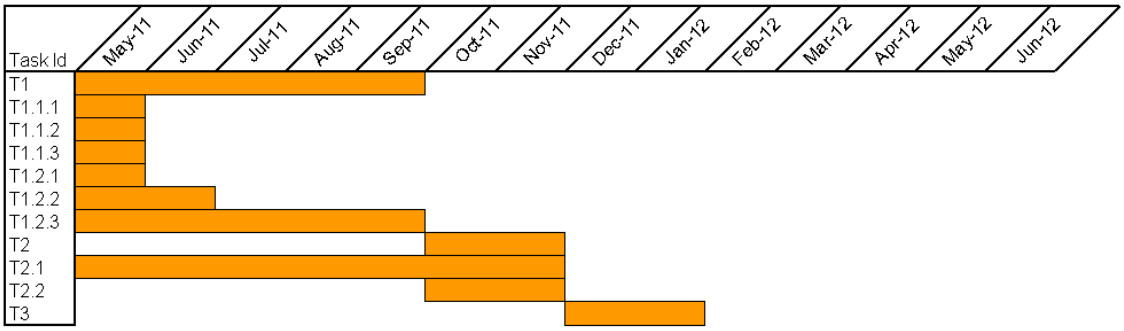

Figure 4-5: Testbed 3 integration work plan

		Task 1: Development and finalisation of each KBB		Task 2: Integration of each solution		Task 3: Integration of the system within Testbed 3	
Solution	**KBB**	**Development Status**	**Due Date**	**Challenge**	**Due Date**	**Challenge**	**Due Date**
Subscriber authentication solution	CPE	EAP/AKA 802.1x software to read cards (XSupplicant)	31.05.2011	Integration of all subsystems with the OLT. The solution must achieve: • Read the UICC card. • Authenticate the Subscriber. • Enable services contracted by the subscriber.	30.11.2011	UICC card reader will be integrated within the multi-radio FemtoNode. **Proof of Concept:** It will be shown how services are available only when the UICC card is inserted within the Multi-Radio FemtoNode.	31.01.2012
	NASS	Authentication procedure among the UAAF and CLF	31.05.2011				
	RACS	Policy enforcement in RCEF according to A-RACF rules	31.05.2011				
Multi-Radio femtonode solution	Femto connection to xDSL router through PC	Standard femto connected to the mobile core network through the PC and ADSL router	31.05.2011	Inter-operation of different radio interfaces, including the FemtoNode, showing the provision of some demo services that involve 2 or more simultaneous radio interfaces	30.11.2011		
	Radio interfaces integration in PC	Checking simultaneous operation of radio plug-ins in the PC	30.06.2011				
	Multi-radio SW	Developing proof-of-concept services supported on multi-radio	30.09.2011				

Table 4-3: Work Plan of Testbed 3

5. Testbed 4, Automatic Fault Diagnosis in Enterprise Networked Femtocells

The goal of Testbed 4 was to validate the Fault Diagnosis functionality developed in BeFEMTO [12][13]. This functionality is intended to find the most likely cause of service errors in the enterprise networked femtocells scenario and relies on the availability of a LFGW to host diagnosis components.

5.1 Automatic Fault in Enterprise Networked Femtocells

By means of its fault diagnosis capabilities, the BeFEMTO system proposes to locally find the root cause of service affecting problems without the intervention of the femtocell management system (HMS). To do so, fault diagnosis functionality will be deployed in the BeFEMTO LFGW and will gather relevant status information from several sources.

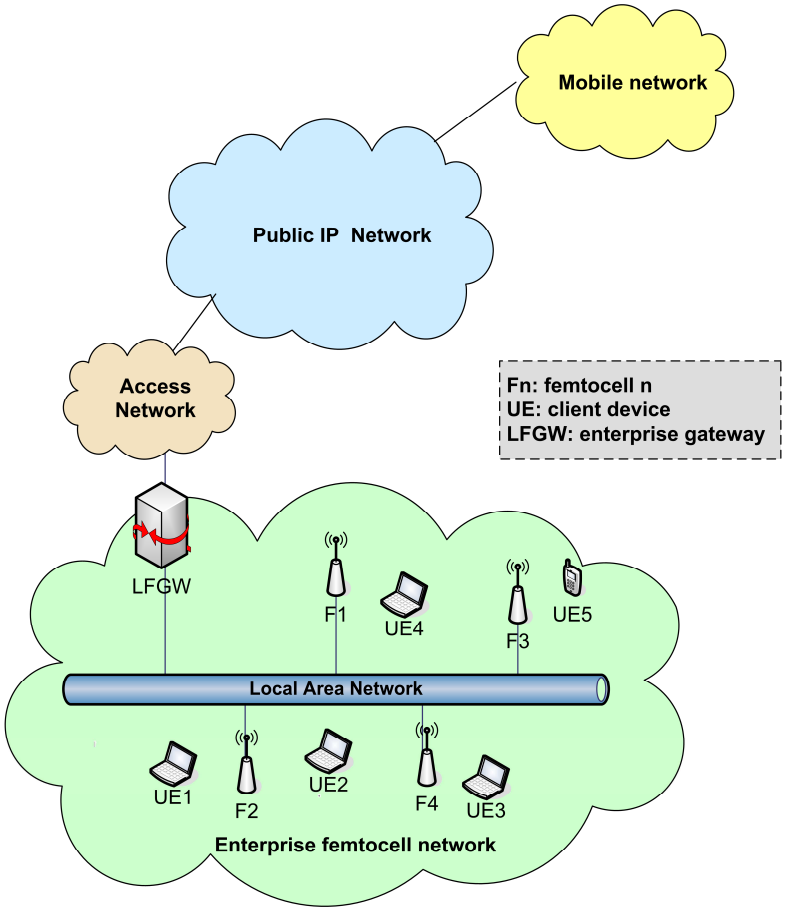

Figure 5-1: Testbed 4 architecture

The scenario selected focuses on QoS degradation when a user is consuming a video in his mobile device under the coverage of an enterprise femtocell network.

5.1.1 Short Description

Fault Diagnosis is designed as a recursive algorithm that performs Bayesian inference with available information until it reaches a given predefined confidence level. Otherwise, it tries to gather additional information by performing relevant tests or requesting a belief to diagnosis agents in other network domain. From all the set of possible tests that can be performed, the one with the largest difference between value and cost is chosen first. The value and cost parameters are described below:

- **Value of the test**. A test has a large value if the result of that test implies a significant increase in the confidence of the diagnostic.
- **Cost of the test.** It represents the cost in terms of resources used, time consumed to perform such a test, price, etc.

In short, the diagnosis procedure works as follows:

1. It adds to the BN all stored observations and beliefs related to it.
2. It performs Bayesian inference
3. It checks if the confidence in the diagnosis is high enough. If so, it stops the diagnosis. Otherwise it goes to 4.
4. It selects the most appropriate action (test, request observation, request belief or request diagnosis). Depending on the action to be done:
    - If the given action is to perform a test, it performs it and returns to 1.
    - If the given action is to request an observation, it sends the request, and waits a maximum time for an answer. When receiving a proper answer, it goes to 5.
    - If the given action is to request a belief, it sends the request, and waits a maximum time for an answer. When receiving a proper answer, it goes to 5.
    - If the given action is a request for a diagnosis, it waits for the answer a maximum time - already passed time since the start of the diagnosis. Once it receives the answer, it goes to 5.
    - If no more actions are found, it finishes the diagnosis procedure. This can happen because it cannot get more evidences or beliefs, or because the time to perform the diagnosis has been exceeded.
5. It waits for an answer. When receiving one, should evaluate the response to continue the diagnosis, going to 1 again.

5.1.2 Demo Scenario and KPIs

- This testbed relates to the use case “Femto for Enterprise” and aims to cover the requirements associated to Zero-configuration and self-management of femtocells. In particular, it focuses on automatic fault diagnosis, addressing problems that may span several network domains. In particular, the chosen service error failure consists on QoS degradation perceived by a user accessing a video through his UE when he’s in the local femtocell network. As a result of service error diagnosis, the BeFEMTO system should be able to determine the cause of the QoS degradation, as well as the network domain responsible for the fault.
- In addition to finding the root cause, propagation of fault information between different network domains may be needed. For that, 3GPP technical report part on alarm requirement [15] should be revisited for the Femto Network scenario. Care must be taken that only relevant fault information is propagated according to the policies in place.
- **KPIs**: The evaluation results will address the following KPIs:
    - Percentage of problems locally diagnosed inside the femtocell network without involving resources from other domains.
    - Percentage of conclusive diagnosis.
    - Average time to diagnose a problem.
    - Average number of test agents involved per type of diagnosis error.
    - Alarm reduction propagation to the macro network.

5.2 Key Building Blocks and Integration Specification

5.2.1 Fault diagnosis framework

Fault Diagnosis relies on the KOWGAR framework developed by Telefónica I+D. This framework is built on top of the Multi Agent Platform JADE and provides generic fault diagnosis functionality which needs to be adapted to the scenario at hand. So far, the framework has been used in several scenarios, both in research and commercial projects.

The main adaptations needed in this framework for BeFEMTO purposes are the modelling of the diagnosis knowledge, by means of a Bayesian Network, and the development of interfaces with the main

sources of status information. Further, specific test agents may need to be developed to meet BeFEMTO needs.

Therefore, the following interfaces have to be implemented in BeFEMTO (Table 5-1):

- Interface between Fault Diagnosis and the femtocells. This interface will be based on CLI and will allow Fault Diagnosis retrieving relevant status information from each of the femtocells that make up the Enterprise Network. Alarms and performance counters are provided by the femtocell. The selection of the ones relevant to the use case still needs to be done.
- Interface between Fault Diagnosis and the Iuh-Tap. This component dissects Iuh traffic passing through the LFGW and gathers relevant information on the status of the enterprise femtocell network. Based on the events detected, it can further propagate this information to other BeFEMTO components, as would be the case of Fault Diagnosis.
- Specific tests to check connectivity status between different network domains and other issues which can have a negative impact on video quality may need to be integrated in some of the KOWGAR observation agents.

Table 5-1: Fault diagnosis KBBs

Key building Block	Functionalities	Input	Output
Fault Diagnosis framework	Automatic root cause detection.	• Femtocell status information. • Iuh-Tap events. • Connectivity information.	Set of main hypothesis together with their probabilities.
Interface between Fault Diagnosis and the femtocells	Propagating relevant information about the status of a femtocell.		Status information.
Interface between Fault Diagnosis and the Iuh-Tap	Propagating relevant status information and events detected by the Iuh-Tap.		Status information.
Specific test agents	Execute tests to get additional evidence.		Result of the test.

5.2.2 Alarms and Performance Counters Feedback by the 3G Femtocell

To support the fault diagnosis activity, the femtocell reports various alarms and performance counters. Since real implementation of Testbed 4 did not support TR-069 [16], which is the native way for the femtocell to report those variables to the OAM using the specific femtocell data model [17], the use of an XML format was chosen such that alarms and performance counters can be retrieved by the LFGW. Table 5-2 and Table 5-3 show the alarm and performance counters available at the LFGW, respectively.

Table 5-2: Alarms reported by the femtocell

Alarm Internal ID	Severity	Event type	Probable cause	Description
Generic Alarms				
ERROR_SOFTWARE_PROCESSING	MAJOR	Processing error alarm	Software Error	Software error
ERROR_LAN_COMMUNICATIONS	MAJOR	Processing error alarm	LAN error	No LAN communication available
ERROR_DNS_QUERY	MAJOR	Processing error alarm	LAN error	No response fro DNS Query
ERROR_NETWORK_CLOCK_SYNC	MAJOR	Processing error alarm	LAN error	HNB could not get clock synchronization
ERROR_HNB_AUTHENTICATION	MAJOR	Processing error alarm	Invalid parameter	HNB could not authenticate
ERROR_HMS_ASSOCIATION	MAJOR	Processing error alarm	LAN error	HNB could not associate with HMS
ERROR_HNBGW_ASSOCIATION	MAJOR	Processing error alarm	LAN error	HNB could not regster with HNB_GW
ERROR_LOSS_OF_SYNCRONIZATION	MAJOR	Processing error alarm	Clock synchronization problem	Loss of clock synchronization
ERROR_FIRMWARE_UPGRADE	MAJOR	Processing error alarm	Software Download Failure	Firmwar efails to upgrade
ERROR_LOCATION_CHANGE	Warning	Environmental Alarm	External Equipment Failure	HBN detects a location change during REM
SON Alarms				
SON_DETECT_HIGH_INTERFERENCE	MAJOR	Quality Of Service Alarm	Performance Degraded	HNB detects high level of radio interference
SON_CONFIGURATION_ERROR	MAJOR	Processing error alarm	LAN error	HNB detects configuration error and can not become operatonal
SON_STATE_CONTROL_ERROR	MAJOR	Processing error alarm	LAN error	HNB detects an internal state control error
SON_NO_SUITABLE_PSC	MAJOR	Quality Of Service Alarm	Performance Degraded	No free PSC available
SON_NO_SUITABLE_LAC	MAJOR	Quality Of Service Alarm	Performance Degraded	No free LAC/RAC available
SON_NWLPHY_CONTROL_ERROR	MAJOR	Equipment	Equipment failure	HNB unable to control network listening system
SON_RADIO_CONTROL_ERROR	MAJOR	Equipment	Equipment failure	HNB unable to control radio system
SON_ASN1_DECODE_ERROR	minor	Processing error alarm	Software Error	ASN1 decode error
SON_TLV_DECODE_ERROR	minor	Processing error alarm	Software Error	TLV decode error
SON_SYSTEMCALL_ERROR	minor	Processing error alarm	Software Error	System call error

Table 5-3: Performance counters reported by the femtocell

Name	Type	Description
Performance counters relating to Real Time Transport using RTP		
LostRcvPackets	uint32	The number of Lost RTP packets in reception
LostFarEndPackets	uint33	The number of Far End Lost RTP packets
Overruns	uint32	Total number of times the receive jitter buffer has overrun
Underruns	uint32	Total number of times the receive jitter buffer has underrun for a CS-domain RAB
MeanRTT	uint32	The mean Round Trip Time in microseconds as computed by the source
MaxRTT	uint32	The maximum Round Trip Time in microseconds as computed by the source
MeanReceiveJitter	uint32	The mean receive jitter in microseconds as computed by the source
MaxReceiveJitter	uint32	The maximum receive jitter in microseconds as computed by the source
MeanFarEndJitter	uint32	The mean far end jitter in microseconds as computed by the source
MaxFarEndJitter	uint32	The maximum far end jitter in microseconds as computed by the source
Performance counters relating to RTP		
SentPackets	unsigned int	The number of sent RTP packets
RcvPackets	unsigned int	The number of received RTP packets
BytesSent	unsigned int	The number of sent RTP bytes
BytesReceived	unsigned int	The number of received RTP bytes
Performance counters relating to RAB establishment		
RABSuccEstabCS	unsigned int	The number of successsful establishments of CS RAB
RABFailEstabCS	unsigned int	The number of failed establishments of CS RAB
RABSuccEstabPS	unsigned int	The number of successsful establishments of PS RAB
RABFailEstabPS	unsigned int	The number of failed establishments of PS RAB
FailHO	unsigned int	The number of successsful handover
SuccHO	unsigned int	The number of failed handover
Performance counters relating to SCTP		
InCtrlChunks	unsigned int	The number of input control chunks
OutCtrlChunks	unsigned int	The number of output control chunks
InDataChunks	unsigned int	The number of input data chunks
OutDataChunks	unsigned int	The number of output data chunks
Performance counters relating to Iuh access		
ueRegIuhSuc	unsigned int	The number of UE successful registration
ueRegIuhFail	unsigned int	The number of UE failed registration
Performance counters relating to Nas access		
locUpdateSuc	unsigned int	The number of UE successful location update
locUpdateFail	unsigned int	The number of UE failed location update
attachSuc	unsigned int	The number of successful attach procedure
attachFail	unsigned int	The number of failed attach procedure
rouUpdateSuc	unsigned int	The number of successful routing area update
rouUpdateFail	unsigned int	The number of failed routing area update

5.3 Work Plan

Due to internal reassignment which occurred during the second year of the project within one partner, this testbed had to be discontinued. Despite noticeable progress presented here, the content is still incomplete and should be regarded as work in progress.

6. Conclusions

This deliverable has presented the down-selected algorithms (with associated scenarios) that are going to be demonstrated by BeFEMTO Work Package 6 (WP6) at the end of the project. Key building blocks integration and interface specifications were given to ensure a coherent comprehension and integration of these algorithms amongst the involved partners. These algorithms reflect a small part of all the innovative work carried out during the BeFEMTO project by the other WPs.

More specifically:

- Testbed 1 will demonstrate **self-organisation network (SON)** using graph colouring associated with dynamic frequency reuse.
- Testbed 2 will test **distributed routing algorithm** within a network of femtocells wirelessly connected to each other. The BeFEMTO **Local Femto Gateway** Iuh-Tap capability (dissection of Iuh messages on the fly) will also be evaluated.
- Testbed 3 will implement a **seamless authentication** of one femtocell subscriber through the use of a removable UICC card in order to provide new/enriched services procedure regardless of the geographical location of the user.
- Testbed 4 should have evaluated fault diagnosis in an enterprise networked femtocells but due to internal resource reassignment it has to be discontinued.

The remaining work to ensure a successful demonstration by the end of the project of the three testbeds has been carefully planned among the involved partners (Gantt charts provided).

7. Appendix A: Testbed 1 Down-Selection Procedure

This appendix gives an overview of the currently defined algorithms from WP3 [18] and WP4 [2][3]. These algorithms are evaluated for their applicability based on the information template defined in Table 7-1. This is used for facilitating comparability. It should however be noted that this is a preliminary selection based on the current available results and that, very likely, the algorithms will be improved further, and/or more detailed results will be provided in the coming project months' that might thus slightly alter the selection.

Table 7-1: Algorithm Description Template

Algorithm Name	The name of the technique/algorithm
Description	High-level description of the algorithm
Direction	Please specify to which transmission direction the algorithm is applicable: UL, DL, both
Channels	Control, Data (if unknown select data)
Coordination	Specifies the coordination aspect of the algorithm: standalone, coordination (centralized, decentralized/distributed)
Node Impact	Specifies which nodes the algorithm impacts, i.e. Macro eNB, Home eNB, UE, centralized controller and the number of nodes necessary
Update Frequency	How often is the algorithm supposed to be executed, i.e. every TTI, every radio frame, every 100ms, once (static), at startup, etc.
Inputs	- The inputs needed for the algorithm.
Outputs	- The parameters controlled by the algorithm.
Key Building Blocks affected	Defines which blocks need to be changed to implement the algorithm and what needs to be changed with respect to the algorithm.
Implementation Complexity	Gives an estimate of the expected implementation complexity of the algorithm for each key building block

The following algorithms have been so far presented in WP3 and WP4.

WP3 based on internal report IR3.2 [16]:

- Interference Avoidance
  - Static Fractional Frequency Reuse Schemes (IR3.2, §3.1.2.2)
  - Downlink Power Control (IR3.2, §3.1.2.1.2.6)
  - Uplink (IR3.2, §3.1.3)
- Multi Operator Indoor Band Sharing (IR3.2, §3.2)
- TDD at UL FDD Bands (IR3.2, §3.3 and §4.1.4)
- RRM Scheduling Algorithm for Self-Organizing Femtocells (IR3.2, §4.1.2)
- Resource Allocation with Opportunistic Spectrum Reuse (IR3.2, §4.1.3)

WP4 based on deliverable D4.1 [2] and D4.3 [3]:

- Graph-Based Dynamic Frequency Reuse (D4.1, §3.1.3.3)
- Radio Context Aware Learning Mechanism (D4.1, §2.3.4)
- Base Station Coordinated Beam Selection (D4.1, §3.1.3.4)
- Decentralized Femto Base Station (HeNB) Coordination for Downlink Minimum Power Beamforming (D4.1, §3.1.3.4)
- Decentralized Q-learning (D4.1, §3.2.3.1)
- QoS Provisioning for Femtocell Networks (D4.1, §3.2.3.3)
- Radio Context Aware Learning Mechanisms (D4.1, §2.3)
- Extended Graph-Based Dynamic Frequency Reuse (D4.3, §4.4.3)

As the focus of radio testbed is on the standalone scenario, only a selected number of these WP4 algorithms will be analyzed in more detail.

Based on the algorithm overview template given in Table 7-1 the algorithms are analyzed with respect to which key building blocks are affected and what would be the implementation complexity for each key building block taking into account the workings of the algorithm, the LTE standard compliance and its required input and outputs.

7.1.1 Interference Avoidance: Static Fractional Frequency Reuse Schemes

Table 7-2: Interference Avoidance: Static Fractional Frequency Reuse Schemes

Algorithm Name	Interference Avoidance: Static Fractional Frequency Reuse Schemes
Description	The algorithm is an extension of the soft frequency reuse scheme which uses 8 subbands for the cell-centre region and one subband for the cell-centre and cell-edge region (using higher power) in the configured downlink bandwidth. It is extended to 3 virtual frequency reuse tiers per cell each using different power. The algorithm consists of 3 substeps running at different timescales. At deployment the RBs and power are assigned to each eNB and HeNB. Based on the location of the femto user the corresponding resource blocks (subbands) are selected for the user. Every TTI the user is scheduled in the subband.
Direction	UL/DL
Channels	Data
Coordination	Standalone
Node Impact	eNB, HeNB, [FM]UE
Update Frequency	Static, Event-based (every CQI report), every TTI
Inputs	Static: - Number of RBs - Number of HeNB per eNB Every CQI report: - Location estimate of UE (SINR- or RSRP-based) Every TTI: - Correct subband is selected for UE
Outputs	- Resource allocation for each eNB and HeNB - Subband(s) per UE - Resource allocation per UE
KBB affected	Protocol Stack (RRM): Configure subbands and CQI threshold Protocol Stack (MAC Scheduler): select subband according to CQI and schedule UE in subband. PHY: make CQI channel measurements and report to (H)eNB, transmit different RBs with different power values.
Implementation Complexity	Protocol Stack (RRM): Low Protocol Stack (MAC Scheduler): Low PHY: Medium

7.1.2 Interference Avoidance Downlink Power Control

Table 7-3: Interference Avoidance: Downlink Power Control

Algorithm Name	Interference Avoidance Downlink Power Control
Description	One proposed solution is downlink power control which controls the transmit power of the femtocell in order to mitigate interference to the macro user, while ensuring coverage for the femto user. It is based on the macro inter-site distance and the pathloss estimate from eNB to HeNB.
Direction	DL
Channels	Control/Data
Coordination	Standalone
Node Impact	HeNB
Update Frequency	Static/Periodic
Inputs	- Pathloss to eNB

	- HeNB Coverage Area Factor - Alpha coefficient
Outputs	- Transmit Power
KBB affected	Protocol Stack (RRM): Receive pathloss and calculate transmit power Sniffer: PHY receiver does pathloss measurements PHY: applies configured transmit power RF: applies configured transmit power
Implementation Complexity	Protocol Stack: Middle Sniffer: Middle PHY: Low RF: Low

7.1.3 Interference Avoidance Uplink

Table 7-4: Interference Avoidance Uplink

Algorithm Name	Interference Avoidance Uplink
Description	This coordination mechanism aims to diminish the experienced co-channel interference levels by reducing the number of simultaneous transmissions taking place in nearby femtocells. The interference level per TTI is reduced by rescheduling conflicting transmissions of surrounding femtocell sites to non-overlapping resource allocations. This is done by executing a coordination mechanism in a sequence following a distance-based priority so as to reschedule for each HeNB that has detected high interference. Though the exact algorithm describing how this is done is not described yet.
Direction	UL
Channels	Data
Coordination	Distributed
Node Impact	HeNB
Update Frequency	Event-based, Periodic
Inputs	- Measured interference - Interference threshold
Outputs	- Resource allocation
KBB affected	Protocol Stack (MAC Scheduler): Execute coordination algorithm and make resource allocation Protocol Stack (X2): A X2-like protocol needs to be implemented PHY: Needs to provide uplink interference measurements
Implementation Complexity	Protocol Stack (X2): Middle Protocol Stack (MAC Scheduler): Middle PHY: Middle

7.1.4 Multi-Operator Indoor Band Sharing

Table 7-5: Multi-Operators Indoor Band Sharing

Algorithm Name	Multi Operator Indoor Band Sharing
Description	HeNB of multiple operators use the same spectrum which is different from the operator specific macro spectrum
Direction	UL
Channels	Control/Data
Coordination	Standalone
Node Impact	N/A
Update Frequency	Static
Inputs	- Type of eNB (Macro or Home)
Outputs	- Carrier frequency
KBB affected	Protocol Stack, PHY, RF
Implementation Complexity	Low

7.1.5 TDD overlay within UL FDD Band

Table 7-6: TDD Overlay within UL FDD Band

Algorithm Name	TDD overlay within UL FDD Band
Description	Interference avoidance is achieved by using optimal resource allocation, which consists of deciding which user to allocate, in which time slot and which operation to perform (UL or DL). This is achieved by building a local conflict graph based on CQI/RSRP measurements of the FUEs.
Direction	DL/UL
Channels	Data
Coordination	Standalone
Node Impact	eNB, HeNB, FUE
Update Frequency	Event-based (every CQI report)
Inputs	- CQI report
Outputs	- scheduled FUE and their resource allocation per HeNB
KBB affected	Protocol Stack (MAC Scheduler): create graph and use it to do resource assignment PHY: make CQI channel measurements and report to (H)eNB
Implementation Complexity	Protocol Stack: High PHY: Low

7.1.6 RRM Scheduling Algorithm for Self-Organizing Femtocells

Table 7-7: RRM Scheduling algorithm for Self-Organizing Femtocells

Algorithm Name	RRM Scheduling Algorithm for Self-Organizing Femtocells
Description	This algorithm exploits the fact that in a femtocell context, due to the low number of served users, a large amount of spectrum per user is available. Each scheduled user uses available resource blocks to repeat (typically twice) his data. Its total transmit power is then divided by the number of used RBs. In this way, the transmitted power per RB decreases, thus mitigating interference.
Direction	DL
Channels	Data
Coordination	Standalone
Node Impact	HeNB, FUE
Update Frequency	Every TTI, Event-based (every CQI report)
Inputs	- CQI of FUE
Outputs	- Power per RB - Allocated RBs - Modulation and Code rate
KBB affected	Protocol Stack (MAC Scheduler): Adapt resource allocation to reduce power PHY: Make rate matching more flexible to allow this specific repetition of data. Make CQI channel measurements and report to (H)eNB
Implementation Complexity	Protocol Stack: Low PHY: Middle

7.1.7 Resource Allocation with Opportunistic Spectrum Reuse

Table 7-8: Resource Allocation with Opportunistic Spectrum Reuse

Algorithm Name	Resource Allocation with Opportunistic Spectrum Reuse
Description	The algorithm exploits the MUE to HeNB channel conditions and the knowledge of the RB assignment of the MUE in order to allow for opportunistic RB reuse at HeNB for scheduling
Direction	DL (UL for measurements)
Channels	Data
Coordination	Standalone
Node Impact	HeNB, MUE, eNB
Update Frequency	Every TTI

Inputs	- eNB RB Assignments - MUE Identifiers - MUE-HeNB Channel
Outputs	- Reusable Resource Blocks
KBB affected	Protocol Stack (MAC Scheduler): take into account MUE RB assignment and the channel quality when scheduling FUE. Protocol Stack (X2): A X2-like protocol needs to be implemented PHY: make MUE-HeNB channel measurements
Implementation Complexity	Protocol Stack (MAC Scheduler): High Protocol Stack (X2): Middle PHY: High

7.1.8 Base Station Coordinated Beam Selection (BSCBS)

Table 7-9: Base Station Coordinated Beam Selection

Algorithm Name	Base Station Coordinated Beam Selection
Description	This algorithm involves a message exchange between macro-user and interfering Femto Base station, followed by several message exchanges between interfering Femto Base station and macro base station. The first message contains information about those precoding matrices that, if not used at an interfering cell, would result in reduction of interference at the UE. Indeed, the goal is to coordinate selection of beams at neighbouring cells to avoid beam 'collisions'. Note that it may well happen that this "codebook restriction" means no transmission at all.
Direction	DL
Channels	Data
Coordination	Distributed
Node Impact	HeNB, eNB, UE
Update Frequency	Event-based (Depends upon CQI reporting period)
Inputs	- Restriction Request from UE (codebooks) - Restriction Grant/Reject from neighbour cell (codebooks, utility) - Utility metric of scheduled UEs
Outputs	- scheduled UEs and their precoding
KBB affected	Protocol Stack (MAC Scheduler): computation of a utility gain at each cell as to gain of the codebook restriction Protocol Stack (X2): X2-like connection needs to be implemented in order to exchange precoding coordination messages. PHY: Make CQI and precoding matrix channel measurements and report to (H)eNB. Apply codebook-based precoding for transmit and receive
Implementation Complexity	Protocol Stack (MAC Scheduler): High Protocol Stack (X2): Middle PHY: High

7.1.9 Decentralized Femto Base Station (HeNB) Coordination for Downlink Minimum Power Beamforming

Table 7-10: Decentralized Femto Base Station Coordination for DL Minimum Power Beamforming

Algorithm Name	Decentralized Femto Base Station (HeNB) Coordination for Downlink Minimum Power Beamforming
Description	The system optimization objective is to minimize the total transmitted power of coordinated HeNBs subject to fixed cross-tier interference constraints, and femto-UE specific SINR constraints using a complex iterative optimization algorithm which requires information exchange between eNBs many iterations
Direction	DL
Channels	Data
Coordination	Distributed
Node Impact	HeNB, UE
Update Frequency	Event-based

Inputs	- Interference level from neighbour cell (or interference at each UE)
Outputs	- Beamformers
KBB affected	Protocol Stack (MAC Scheduler): iterative optimization algorithm Protocol Stack (X2): A X2-like protocol needs to be implemented PHY: Apply beamforming. Make interference measurements
Implementation Complexity	Protocol Stack: High PHY: Very High

7.1.10 Graph-Based Dynamic Frequency Reuse (GB-DFR)

Table 7-11: Graph-based Dynamic Frequency Reuse

Algorithm Name	Graph-Based Dynamic Frequency Reuse
Description	A centralized controller builds an interference graph based on information received from the UE and assigns subbands to femto cells, so as to minimize inter cell interference.
Direction	DL
Channels	Data
Coordination	Centralized
Node Impact	HeNB, UE and centralized controller
Update Frequency	Event-based (every measurement report)
Inputs	- Cell IDs of the interfering HeNBs - Total number of subbands - Minimum number of subbands assigned to each HeNB (s_{min}), - SINR threshold, γ_{th}, used by UEs to determine interfering HeNBs - UE Measurement report (RSRP)
Outputs	- Subbands that will be used by each HeNB
KBB affected	Protocol Stack (RRM): Build graph of nodes. Determine subbands to be used by scheduler PHY: make neighbour cell measurements
Implementation Complexity	Protocol Stack: Middle PHY: High

7.1.11 Decentralized Q-learning algorithm)

Table 7-12: Decentralized Q-learning algorithm

Algorithm Name	Decentralized Q-learning algorithm
Description	Multiple agents try to optimize their transmit power per resource block by repeatedly interacting (selecting a transmit power per RB) with the environment in such a way that their performances are maximized, while trying to protect the MUE from interference.
Direction	DL
Channels	Data
Coordination	Distributed
Node Impact	HeNB, eNB, MUE
Update Frequency	Event-based (every CQI report)
Inputs	- MUE Interference Indicator (true, false) per RB via X2 based on CQI report
Outputs	- Power allocation per RB
KBB affected	Protocol Stack (RRM): Implementation of Q-learning algorithm Protocol Stack (X2): A X2-like protocol needs to be implemented for exchange of interference indicator PHY: make CQI channel measurements and report to eNB, support of power control per RB
Implementation Complexity	Protocol Stack (RRM): High Protocol Stack (X2): Middle PHY: Middle

7.1.12 QoS Provisioning for Femtocell Networks

Table 7-13: QoS Provisioning for Femtocell Networks

Algorithm Name	QoS provisioning for femtocell networks
Description	By using game theory the HeNB transmission is dynamically updating the resource block allocation and the power level of the RB
Direction	DL
Channels	Data
Coordination	Distributed
Node Impact	HeNB, eNB, MUE
Update Frequency	Event-based (every CQI report)
Inputs	- MUE SINR threshold - SINR of MUE per RB
Outputs	- Power allocation per RB
KBB affected	Protocol Stack (RRM): Implementation of stochastic game algorithm Protocol Stack (X2): A X2-like protocol needs to be implemented for exchange of MUE SINR PHY: make SINR measurements and report to eNB
Implementation Complexity	Protocol Stack (RRM): Very high Protocol Stack (X2): Middle PHY: Middle

7.1.13 Radio Context Aware Learning Mechanism

Table 7-14: Radio Context Aware Learning Mechanism

Algorithm Name	Radio Context Aware Learning Mechanism
Description	The algorithm tries to find a HeNB policy that maximizes the achievable rate of the FUE selecting specific RBs for transmission while trying to keep the MUE QoS requirements.
Direction	DL
Channels	Data
Coordination	Distributed
Node Impact	HeNB, eNB, MUE
Update Frequency	Event-based (every CQI report)
Inputs	- SINR of MUE and FUE per RB
Outputs	- RB allocation
KBB affected	Protocol Stack (RRM): Implementation of learning algorithm Protocol Stack (X2): A X2-like protocol needs to be implemented for exchange of MUE SINR PHY: make SINR measurements and report to eNB
Implementation Complexity	Protocol Stack (RRM): High Protocol Stack (X2): Middle PHY: Middle

7.1.14 Extended GB-DFR

Table 7-15: Extended GB-DFR

Algorithm Name	Extended Graph-Based Dynamic Frequency Reuse
Description	A centralized controller builds an interference graph based on information received from the UE and assigns subbands to femto cells, so as to minimize inter cell interference. In order to improve spectrum efficiency, cell edge users are assigned the above mentioned subbands, while cell centre users, more protected are allowed to reuse more spectrum, under certain conditions
Direction	DL
Channels	Data
Coordination	Centralized
Node Impact	HeNB, UE and centralized controller
Update Frequency	Event-based (every measurement report)

Inputs	- Cell IDs of the interfering HeNBs - Total number of subbands - Minimum number of subbands assigned to each HeNB (smin), - SINR threshold, γth, used by UEs to determine interfering HeNBs - UE Measurement report (RSRP)
Outputs	- Subbands that will be used by each HeNB
KBB affected	Protocol Stack (RRM): Build graph of nodes. Determine subbands to be used by scheduler PHY: make neighbour cell measurements
Implementation Complexity	Protocol Stack: Middle PHY: High (can be simplified, see below)

7.1.15 Conclusions

From the above presented algorithms a selection was done based on what we think provides a good cross section of the BeFEMTO algorithms. The selected algorithm “Extended GB-DFR”, while being a promising method to manage interference within the boundaries of the LTE-A standard, is feasible to implement in terms of complexity and capabilities in a real setup within the BeFEMTO project.

The reason why we did not select the above described other algorithms is essentially due to the limitations of the PHY part of Testbed1:

- Testbed1 cannot implement power control digitally. Power control could be implemented via the RF part, but this would have added a risk w.r.t. the testbed integration plan, thus it was discarded, at least for the time being. This discards algorithms 7.1.1, 7.1.11 and 7.1.12.
- Testbed1 does not have a sniffer, which discards algorithm 7.1.2.
- Testbed 1 does not have UL, which discards algorithms 7.1.3, 7.1.4, 7.1.5 and 7.1.7.
- Testbed1 is currently unable to repeat the same data on 2RBs, in addition on the transmitter side Maximum Ratio Combining is not implemented to date. So, this discards algorithm 7.1.6.
- Testbed1 does not implement beamforming, which discards algorithms 7.1.8 and 7.1.9.
- Algorithm 7.1.13 makes sense only with at least 2 HeNBs and 2 FUEs, while Testbed1 have only one FUE.

Algorithm 7.1.10 (GB-DFR) assumes a context with more than two HeNBs, and applies a complex graph theory algorithm to determine a frequency portioning, where each HeNB is allocated a certain number of subbands. From a demonstration point of view, there is no point to choose this algorithm with two HeNBs, since in this symmetrical situation, making the frequency allocation orthogonal is quite simple and does not require a complex algorithm.

In contrast, Algorithm 7.1.15 (extended GB-DFR), though assuming also a context of more than 2 HeNBs, separates cell centre from cell edge users, applying the results of GB-DFR to the cell-edge users, while leaving more flexibility for the cell centre users. For instance, GB-DFR will assign PS1 to HeNB1 and PS2 to HeNB2. However, HeNB2 does not have any cell edge users to serve, so that HeNB1 is entitled to use PS2 for its cell centre users. More details are given in section 2.1.1. Even though the implementation of GB-DFR is out of the scope of our demonstration, we can still demonstrate an interesting result, which is that the use of PS2 by HeNB2 has only a slight impact on FUE1, when it uses PS2 in the cell centre of its serving cell HeNB1.

8. References

[1] BeFEMTO D6.1, “Selection of scenarios for proof of concept testbeds and specifications for key building blocks functionalities and interfaces”, ICT 248523 FP7 BeFEMTO project, December 2010

[2] BeFEMTO D4.1 “Preliminary SON Enabling & Multi-Cell RRM Techniques for Networked Femtocells”, ICT 248523 FP7 BeFEMTO project, December 2010

[3] BeFEMTO D4.3, “Multi-cell RRM and self-optimisation for networked, fixed relay and mobile femtocells”, ICT 248523 FP7 BeFEMTO project, December. 2011

[4] 3GPP TS 36.211, “Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Channels and Modulation (Release 8)”

[5] 3GPP TS 36.321, “Evolved Universal Terrestrial Radio Access (E-UTRA); Medium Access Control (MAC) protocol specification (Release 8)”

[6] 3GPP TS 36.322, “Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Link Control (RLC) protocol specification (Release 8)”

[7] 3GPP TS 36.323, “Evolved Universal Terrestrial Radio Access (E-UTRA); Packet Data Convergence Protocol (PDCP) specification (Release 8)”

[8] 3GPP TS 36.331, “Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification (Release 8)”

[9] 3GPP TS 23.203, “Policy and charging control architecture (Release 8)”

[10] 3GPP TS 25.468, “UTRAN Iuh Interface RANAP User Adaption (RUA) signalling”

[11] 3GPP TS 25.469, “UTRAN Iuh interface Home Node B (HNB) Application Part (HNBAP) signalling”

[12] BeFEMTO D5.2, “Femtocells Access Control, Networking, Mobility, and Management mechanisms”, ICT 248523 FP7 BeFEMTO project, December 2011

[13] BeFEMTO D5.1, “Femtocells Access Control, Networking, Mobility, and Management concepts”, ICT 248523 FP7 BeFEMTO project, December 2010

[14] BeFEMTO D2.2, “The BeFEMTO System Architecture”, ICT 248523 FP7 BeFEMTO project, December 2011

[15] 3GPP TR 32.821 “Study of Self-Organizing Networks (SON) related Operations, Administration and Maintenance (OAM) for Home Node B (HNB) (Release 9)”

[16] Broadband Forum TR-069, “CPE WAN Management Protocol v1.1”

[17] Broadband Forum TR-196, “Femto Access Point Service Data Model issue 1”

[18] BeFEMTO IR3.2, “Radio Access Specs and promising techniques for indoor standalone femtocells”, ICT 248523 FP7 BeFEMTO project, December 2010