MCN
Mobile Cloud Networking

SEVENTH FRAMEWORK PROGRAMME

FUTURE COMMUNICATION ARCHITECTURE FOR MOBILE CLOUD SERVICES
Acronym: MobileCloud Networking Project No: 318109
Integrated Project FP7-ICT-2011-8 Duration: 2012/11/01-2015/09/30

MCN
Mobile Cloud Networking

D4.3: Algorithms and Mechanisms for the Mobile Network Cloud

Type	Report
Deliverable No:	D4.3
Workpackage:	WP4
Leading partner:	PTInS
Author(s):	Carlos Parada (Editor) List of Authors overleaf
Dissemination level:	Public (PU)
Status:	Final
Date:	07 November 2014
Version:	1.0

List of Authors (in alphabetical order):

Aiko Pras	UTwente
Alexandru Russu	Fraunhofer
Anna Pizzinat	ORANGE
André Gomes	UniBe
Carlos Marques	PTInS
Carlos Parada	PTInS (editor)
Dragos Vingarzan	Fraunhofer
Eryk Schiller	UniBe
Imad Aad	UniBe
Islam Alyafawi	UniBe
Lucian Suciu	Orange
Lucio Studer Ferreira	INOV
Luis M. Correia	INOV
Luuk Hendriks	UTwente
Malla Reddy Sama	Orange
Miguel Santos	PTInS
Marco Liebsch	NEC
Marius Corici	Fraunhofer
Morteza Karimzadeh	UTwente
Oliver Keil	Fraunhofer
Ricardo de O. Schmidt	UTwente
Simone Ruffino	TI
Sina Khatibi	INOV
Tarik Taleb	NEC
Torsten Braun	UniBe
Valentin Vlad	TUB
Zarrar Yousaf	NEC
Zhongliang Zhao	UniBe

Reviewers (in alphabetical order):

David Palma	ONE
Paolo Crosta	Italtel

Versioning and contribution history

Version	Description	Contributors
0.1 (27 May 2014)	Initial list of expected contributions of the multiple partners	Carlos Parada
0.2 (05 Jun 2014)	Initial draft for the Table of Contents (ToC)	Carlos Parada
0.3 (20 Aug 2014)	Initial contributions for Network Function Placement (section 2)	Carlos Parada
0.3 (30 Sep 2014)	Sections assignment (to be discussed) and Executive Summary, Introduction, Conclusions	Carlos Parada
0.4 (3 Oct 2014)	Added content of chapter 5 on RAN	Lucio Studer Ferreira
0.5 (3 Oct. 2014)	Adding the prediction (mobility and BW prediction) and DMM parts	Morteza karimzadeh
0.6 (06.10.2014)	Added a placeholder and abstract of our contribution related to cost analysis of initial deployment of EPCaaS in a DC	Zarrar Yousaf,
1.0 (30.10.2014)	Final delivery (after revision)	Carlos Parada

Table of Contents

Table of Figures

Table of Tables

List of Acronyms

1:1	Mapping of 3GPP EPC components each to one VM
1:N	Mapping of 3GPP EPC components each to multiple VMs
3GPP	3rd Generation Partnership Project
AAA	Authentication, Authorization Au
ANDSF	Access Network Discovery and Selection Function
AP	Access Point
APN	Access Point Name
CC	Cloud Controller
CCNx	Content Centric Networking
CDF	Cumulative Density Function
CPU	Central Processing Unit
CRRM	Common Radio Resource Management
CTRL	Control
DC	Datacenter
DM	Device Management
DMM	Distributed Mobility Management
DNS	Domain Name Server
DRA	Diameter Router Agent
EEU	Enterprise End User
EMS	Element Management System
EPS	Evolved Packet System
eNB	Evolved NodeB
EPC	Evolved Packet Core
EPCaaS	EPC as a Service
FE	Front End
FQDN	Fully Qualified Domain Name
GRE	Generic Routing Encapsulation
GTP	GPRS Tunnelling System
GTP-C	GTP-Control
GTP-U	GTP-User Plane
HSS	Home Subscriber Server
HTTP	Hyper Text Transmission Protocol
IMS	IP Multimedia Subsystem
IP	Internet Protocol
IP CAN	IP Carrier Access Network
IRP	Ip-raft Request Processor
ITG	Interface Template Graph
LMA	Local Mobility Anchor
LP	Linear Programming
LRRM	Local Radio Resource Management
M-STCP	Mobile Stream Control Transmission Protocol
MaaS	Monitoring as a Service
MAC	Medium Access Control
MCN	Mobile Cloud Networking
MCNSP	Mobile Cloud Networking Service Provider
MME	Mobility Management Entity
MOBaaS	Mobility and Bandwidth prediction as a Service
MVNO	Mobile Virtual Network Operator
N:1	Mapping of all 3GPP EPC components to a single VM

N:2	Mapping of all 3GPP EPC components to two VMs
NAS	Non-Access Stratum
NAT	Network Address Translation
NF	Network Function
NFS	Network File System
NMS	Network Management System
OFC	OpenFlow Control
OFP	OpenFlow Protocol
OFS	OpenFlow Switch
OMS	Operational Management System
OS	Operating System
PCEF	Policy and Charging Enforcement Function
PCF	Policy Control Function
PCRF	Policy and Charging Rules Function
PDF	Probability Density Function
PDN	Packet Data Network
P-GW	Packet Data Network Gateway
P-GW-C	Packet Data Network – Control
P-GW-U	Packet Data Network – User Plane
PoP	Point of Presence
QoE	Quality of Experience
QoS	Quality of Service
RAN	Radio Access Network
RANaaS	RAN as a Service
RAT	Radio Access Technology
RCBaaS	Rating Charging and Billing as a Service
RRM	Radio Resource Management
RRU	Radio Resource Unit
SDB	Subscriber Data Base
SDK	Software Development Kit
SDN	Software Defined Networks
SGW	Serving Gateway
SGW-C	Serving Gateway – Control
SGW-U	Serving Gateway – User Plane
SHM	Shared Memory
SINR	Signal-to-Interference-plus-Noise Ratio
SLA	Service Layer Agreement
SLB	Signalling Load Balancer
SIG	Service Instance Graph
SLF	Service Locator Function
SPR	Subscriber Profile Repository
STG	Service Template Graph
SM	Service Manager
SO	Service Orchestrator
SO-D	SO-Decision
SO-E	SO-Execution
SW	Switch
TPS	Transactions Per Second
TEID	Tunnel Endpoint Identifier
UE	User Equipment
V-RAN	Virtual Radio Access Network
VC	VideoConferencing
VM	Virtual Machine

vNF	virtualized Network Function
vNFC	virtualized Network Function Component
VNI	Virtual Network Interface
VNO	Virtual Network Operator
VoIP	Voice over IP
VRRM	Virtual Radio Resource Management
W	Worker
WP	Work Package
XaaS	X as a Service

Executive Summary

This Deliverable 4.3 (D4.3) presents the main algorithms and mechanisms used on the Mobile Cloud Networking (MCN) project, in particular for services covered by the WP4/T3.5: EPCaaS, ANDSFaaS, MOBaaS and RANaaS. This is the achievement of the project regarding the strategies that a service must follow in order to perform an efficient and flexible operation of the service. Those algorithms and mechanisms will be implemented and evaluated in the project in D4.4, D4.5, D6.3 and D6.4.

The Deliverable focuses on strategies to perform an efficient placement of the service components, in order to fit the required resources and use the most efficient locations, considering both, the initial instantiation and runtime migrations. It also covers Prediction algorithms, which are an important tool to prepare services to adapt to foreseen future conditions, by predicting the location of users and the bandwidth utilization. Distributed Mobility Management mechanisms are also covered, by establishing new logics to forward traffic among the network in a flatter and more efficient way than it is done today. RAN aspects are also covered, aiming to increase the management efficiency of the spectrum as well as the isolation of multiple tenants.

1 Introduction

This section introduces the reader to the motivations, objectives and scope of this document.

1.1 Motivation, Objectives and Scope

The objective of this Deliverable 4.3 (D4.3) is to present and help the reader to understand the algorithms behind the implementation of the EPC, ANDSF and RAN services, provided under the "XaaS" model. The mechanisms and algorithms here presented are aligned with the overall MCN architecture described in [MCN-D2.2] and the EPC- and RAN-specific issues, described in D4.1 [MCN-D4.1], D3.1 [MCN-D3.1] and D3.2 [MCN-D3.2].

The overall MCN architecture [MCN-D2.2] defined the Service Orchestrator (SO) as the element responsible to manage the whole lifecycle of any service, from the initial deployment of resources in the cloud (deployment) and respective configuration (provisioning), going through the runtime execution, which comprises the entire period when the service executes, ending with the service termination and disposal of all associated resources. To implement the decisions and actions to be performed at any moment in time, a clear and supported set of policies must be researched, in order to be sure that customers receive the expected quality of experience (QoE), spending the minimum resources and using them as efficiently as possible.

To trigger the different policies and decisions, monitoring data is a fundamental input. Monitoring provides a picture about how the service is running, by delivering valuable KPIs. Those figures indicate whether the service is providing an acceptable quality of experience, whether the allocated resources are enough, or whether the location of workloads is adequate. However, monitoring has a major limitation: it only provides this picture after the reality occurs, not before. Although in many cases this is enough for problem detection in early stages and take actions in time; in other cases, detection comes too late or the action takes too long.

Prediction mechanisms complement monitoring on this aspect, by predicting some events, which due to its characteristics, can be forecasted based on historical data or by other means. Examples are periodic events related to some periods of the day or the week, which occur due to the repetition of some facts (e.g. high bandwidth in residential areas during the night, or high number of requests during the weekend). For these cases, services can react in advance, e.g. by scaling-out platforms and moving resources to these spots, avoiding any turbulence. For good prediction mechanisms, related to location or bandwidth, specific algorithms must be researched and benchmarked, in order to have reliable and helpful predictions.

The current RAN/EPC services, according to 3GPP specifications, follow a mobility management mechanism common to previous generations of mobile core networks. However, evolutions to this model have been raised in the last years, in particular the DMM (Distributed Mobility Management). DMM intends to simplify the mobility management, flattening the network and easing the re-routing of data packets every time a node moves along the geography and gets attached to different segments. The study and simulation of the different mechanism and algorithms associated to DMM is a relevant work, which should be evaluated and benchmarked to evolve on further deployments.

Closely related to the EPC service is always the RAN (Radio Access Network). RAN provides access to the mobile core network through a licensed spectrum allocated to traditional mobile operators. As licensed spectrum is a very limited and expensive resource, it is of high interest the capability to share this asset, by using a clear multi-tenancy concept. The RANaaS service shares the same physical infrastructure, and is served in an on-demand and elastic environment, according to SLAs previously defined. The final purpose is to provide to customers (typically, MVNOs) the perception that they are the only users of the access, fully isolated from (with no impact on) the other customers. The achievement of this goal is not easy and requires a rigorous and complex set of management mechanisms and algorithms. They were shortly presented in deliverable D3.2, being described and evaluated in the current deliverable. In particular, LTE RAN "cloudification" issues are studied, where a base station is split into remote radio-head and software-based digital unit (BBUs). The feasibility of BBUs running on a virtualized environment on the cloud is evaluated. On the other side, the aggregation of BBUs in pools must be studied, its dimensioning being exemplified taking into consideration existing infrastructure and respecting system requirements. Finally, a virtualisation mechanism of radio resources is also proposed, enabling to serve VNOs by a wireless capacity composed of a shared physical multi-RAT infrastructure either cellular or supported by WLAN traffic offloading capabilities.

1.2 Structure of the Document

Section 2 presents algorithms and mechanism related to the network function placement of services (EPCaaS and ANDSFaaS), both regarding initial deployment as well as runtime migration. Section 3 focuses on prediction algorithms, especially devoted to user mobility prediction. Section **4** introduces the concept of DMM, exploring some mechanisms through simulation and approaching the integration with the EPC. Section 5 goes deeper into the RAN, exploring some algorithms to provide on-demand, elastic and isolated next generation RANaaS service. Finally, section 6 concludes and raises some spotlights to the future work.

2 Network Function Placement

Traditionally, in the telecom world, the network and services equipment that support the operations is very static. In the deployment process, a set of infrastructure resources is deployed in strategic places, previously planned. The capacity and locations of nodes are dimensioned considering the figures provided at design time, usually based on busy hour load and/or predictions for future demand, in order to accommodate the performance growth.

When operators dimension their networks and services, they do not consider the average for regular days, but they target peak periods, like Christmas or New Year. Depending of networks and services, this means an increase of 100% or even more regarding average. This makes operators investments very inefficient, increasing the overall CAPEX costs. Ideally, operators would only pay for this capacity when needed (e.g. peak days, Christmas, New Year, etc.) and not all the time. That would save a lot of resources.

Usually, operators perform major upgrades every three to five years. Changes in networks and services require substantial effort and investments and efforts, and are manually performed by high-skilled technicians. Many times, those operations require service outages (this is the main cause for service downtimes in operators – 99.999%, five 9s) and are susceptible to misconfigurations, increasing failure risks.

The high flexibility and dynamicity of the cloud paradigm can leverage the telecom business when compared to the traditional design and deployment of networks and services. This way, operators no longer need to dimension the traffic demand for the next years, investing on capacity now (CAPEX).

The cloud paradigm is able to address this topic, by allowing operators to dynamically increase their resources, and therefore the capacity of the network, only on peak periods, when this extra capacity is actually needed. During off-peak periods, the level of resources consumed could be made available for other purposes, thereby extracting more value from them.

The cloud paradigm can also address other topics, namely regarding the location of network elements. Using the cloud, network elements can be easily deployed in the most convenient places, according to the traffic amount of different regions, the distance between customers and network elements, or the most suitable (cheap, available) datacentre/cloud provider.

Using the cloud, the redesign of the network will take place with shorter periodicity (e.g. weekly, daily, hourly). This decision can be based on monitoring and short-term predictions. Redesign decisions can increase the resources allocated to a given node or function, or migration of determined nodes and functions to a given region (datacentre).

The MCN Deliverable 4.1, [MCN-D4.1], in section 3, described some cloud enabling technologies that can leverage current network and service models, raising some aspects that can be considered to take advantage of cloud principles and capabilities. This section intends to describe in more detail the algorithms that can be used to dynamically improve the efficiency of EPC deployments.

2.1 Network Function Initial Placement

Initial placement algorithms consider all policies used initially for the EPC service placement. This is the best guess that the Service Orchestrator (SO) can do at this time, considering that no monitoring and load information is yet available. The EPCaaS service can be dynamically adjusted later on during runtime. Section 2.2 describes the algorithms that can be used during service lifetime. The algorithms here described are aligned with the description in [MCN-D4.1], section 3.2.

This section considers algorithms for the EPC service as a whole, comprising by the MME, S-GW, P-GW and HSS components, and using the N:2 architectural approach; and the ANDSF service, comprising an (optional) EPC component (responsible for sending discovery and attachments policies to the terminals), and using the 1:N architectural approach. This section also analyses the cost impact of initial deployment of 1:N based EPCaaS in a Datacentre Infrastructure (DCI). The cost of deployment is measured in terms of the utilization of DC infrastructure resources such as computation and networking.

The initial deployment of virtual network infrastructure (VNI) in a Datacentre (DC) is a complex task with multi-dimensional aspects as compared to deploying a single virtual network function and or virtual application function. The challenge is to ensure that the deployment strategy should not only meet the inter-functional constraints between the multiple virtual network function components (VNFCs) constituting the VNI while ensuring service, performance and operational integrity, but it should also ensure optimal utilization of the underlying resource of the DC infrastructure such as computation and networking. In this section we analyse the cost incurred by two constraint-based heuristically applied initial VNI deployment strategies with reference to virtual mobile network infrastructure providing EPCaaS while taking into consideration functional and administrative constraints. The cost of deployment is measured in terms of the utilization of DC infrastructure resources such as computation and networking.

2.1.1 General Considerations

Some preliminary considerations must be done, before tackling the issue of the initial and runtime placement of EPCaaS SICs within a real operator's network. In fact, assumptions must be made on the physical structure of the network topology, as well as the DCs topology and interconnection.

Usually, the Network domain and the IT domain are managed by different entities within an Operator and there can be cases in which DCs, traditionally owned by IT departments, evolved separately from the rest of the Network. As a consequence, DCs could be physically deployed in different sites than the main Network Point-Of-Presence sites (POP) and they can be interconnected with dedicated links, even physically separated from the operator's backbone network. DCs and POPs can be interconnected in this case, but generally speaking they could be two separate networks, also operated by different company departments. This situation is depicted in Figure 1. The DCs of this type are usually large, secure, highly powered sites.

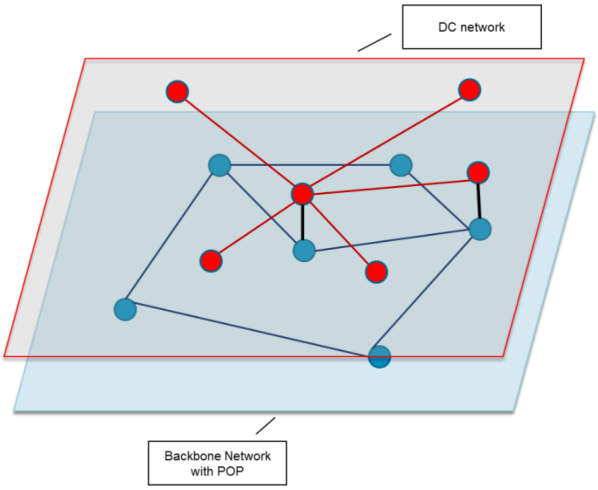


Figure 1 - DC and POP networks physically separated

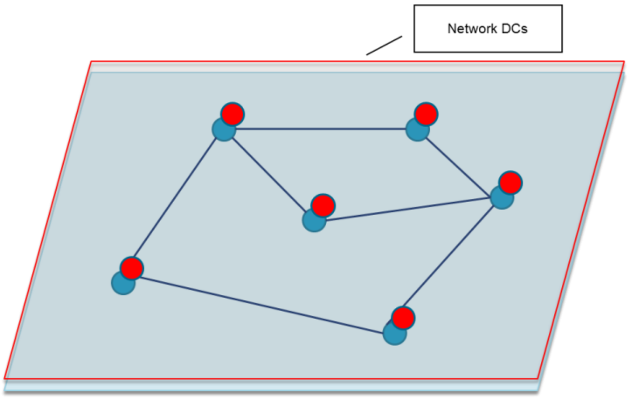


Figure 2 - DC and POP networks integrated - Network DCs

In this deliverable instead, we assume that the two above networks are integrated and DC and POPs are actually co-located, thus not requiring any special connectivity between POPs and DCs (and between DCs), other than the links interconnecting the POPs. This assumption significantly simplifies placement algorithms of EPCaaS SICs across the DCs, mainly because of the geographical distribution of the RAN: in fact, the eNBs are currently connected only to the operators' POPs; if the DCs are not located at the same sites, additional latency would be added, to reach the "closest" DC, where needed EPCaaS are placed (e.g. SGW-U)[1].

We assume also that the co-located DCs are generally smaller, compared to the large, IT-like DCs we described above, since they will be tailored to host "only" EPCaaS SICs. We can call these integrated sites "Network DC", to emphasize the difference with the tradition IT DC. The result network topology is exemplified in Figure 2: DC and POP are actually one entity.

Moreover, it can be envisaged that the Network DCs will be differentiated according to

- Geographical location: DCs in the main cities will have generally larger capacity, in terms of compute/network/storage to cope with the higher traffic and Transactions-per-second (TPS) generated in highly populated areas, while peripheral DCs will have smaller capacity. This is one of the main big advantages brought by the Cloud: the number of compute/network/storage entities can smoothly follow the required capacity.
- Functionalities: centralized, bigger DCs are better suited to host control plane service components, e.g. HSS, PCRF, AAA, DNSs, IMS control, or services like DPI and Firewalling that handle user plane, not to process it but to enforce policies or gather statistics; peripheral DCs are instead better suited to host user plane latency-sensitive services, like CDNs, IMS media-plane.

[1] Here, "closest" means the DC that has less number of hops towards the eNBs. It is worth noting that, in a situation like the one depicted in Figure 1, the DC closest to the majority of the eNBs would be the central one. This means that all EPCaaS would be placed, in the central DC, thus overloading it and not using the remaining DCs efficiently.

The resulting reference network topology, on which the placement analysis will be done, is depicted in Figure 3.

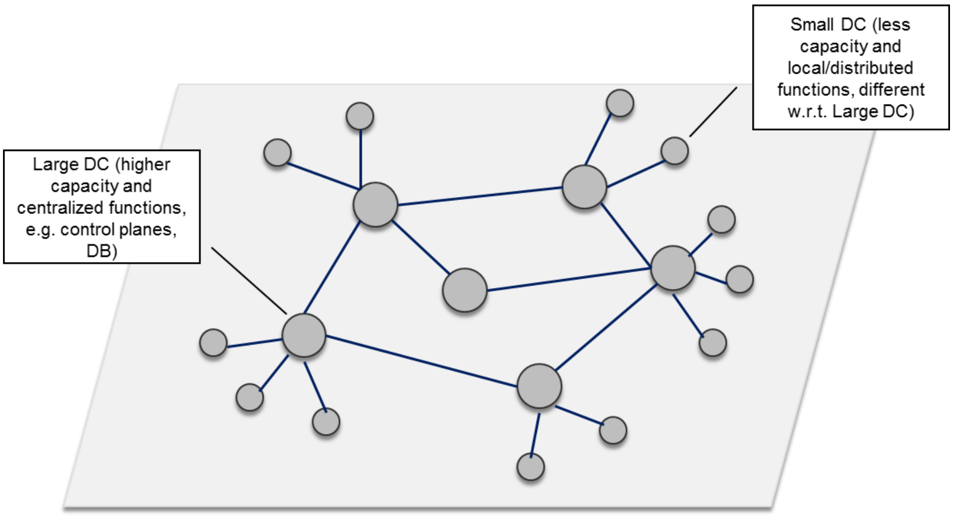


Figure 3 - Network DCs target geographical topology – Example

Summarizing, we assume that in every POP, sufficient compute/network/storage resources (i.e. atomic services, in the terminology of MCN) are available to host EPCaaS SICs. In the next sections, we will analyse placement logics.

2.1.2 EPCaaS Initial Placement Algorithm

The Initial Placement Algorithm (IPA) run by the EPCaaS SO must take the decision on where initially to place the EPCaaS SICs. In this deliverable we'll focus on the N:2 architectural option for the EPCaaS as described in [MCN-D4.2]. As a reminder, the N:2 architecture is described in section 7.

No runtime metric is available at the time of the initial placement and the algorithm can work only with static constraints: topology, capacity of the Network DC and, in general operator's policies. Hence, the input data for IPA are:

- Number, location and type (primary/secondary) of POPs, i.e. Network DC.
- Capacity of each DC, in terms of compute/storage resources. Compute can be expressed in terms of CPU or GHz, storage in terms of TBs available.
- Network constraints (if any).
- Number of SICs for each type to be deployed in the network. We assume this is an input, because it is a figure that usually comes from a priori analysis with the Vendor of the EPCaaS solution and with the Planning department that has updated figures on the capacity, in terms of # of users and traffic load that must be supported.
- Size of the to-be-deployed SICs, in terms of vCPU and vRAM; this is also assumed to be an input, provided by interacting with the Vendor.
- Policy constraints (e.g. required by SLAs or marketing decisions).

The number and type of SICs can be parameters, passed to the SM of EPCaaS, while the other data can be read by the SO at the deployment time, since they are "structural" data. Note that, if the solution has been carefully characterized before the deployment, the SM can be simply passed the total number of user and/or the amount of control/user plane traffic the EPCaaS must handle (similarly to what proposed in section 2.1.4 for the ANDSF SIC). In this case, the SO is responsible to decide number and type of SIC to be deployed.

The following factors must be taken into account by IPA logic to take the decisions:

- **Latency**, which in turn implies:
  - Impacts on the idle-to-active signalling procedures, needed to manage the idle state of the UE. In particular, the procedures SERVICE REQUEST and PAGING, which are used to change the UE state from ECM-IDLE to ECM-CONNECTED, which impacts the latency perceived by the user before being able to transmit data;
  - Impacts on handover latency (i.e. performance of mobility management), in particular inter-eNodeB handover (with or MME and/or SGW relocation).
  - Latency on data traffic, which impacts the UE throughput.

  In section 7, it is included a more detailed analysis of the idle-to-active latency and the e2e data traffic latency, and it is explained how these two latencies influences the location of the EPCaaS SICs. Summarizing, user and control plane SICs should be placed either both distributed or both centralized. Moreover, the location of the user plane SICs depends on the location of the egress point to other PDNs (i.e. towards the Internet or towards other Operator's services)
- **Complexity** in the management: i.e. there must be a hard limit on the number and type of deployable entities, to not complicate the management of those entities. Every SIC, in fact must be provisioned by the SO itself and monitored by the MaaS.

2.1.2.1 Logic for IPA

This section describes an IPA, which can be used to deploy an instance of EPCaaS on a Cloud, managed by Openstack (this is the current assumption of the MCN project).

The IPA logic is summarized Table 1 and is run by the SO, when the SM calls the "deploy" action

Table 1 - Example of an IPA

Input data
- Number of MME/SGW-C/P-GW-C SIC (control SIC, or C-SIC) - Number of SGW-U/P-GW-U SIC (user plane SIC, or U-SIC)[2]
Logic

[2] We assume here the number of control plane and user plane SICs is the same (i.e. un C-SIC and one U-SIC per POP)

```
-   Get N-DCs capabilities, querying Openstack
        o   Calculate aggregate capacity for each Network DC (N-DC)
        o   Get current usage of resources for each N-DC
-   Get the list of all N-DC marked as "primary/large" in the input data[3]
-   Get the list of all N-DC marked as "secondary/small" in the input data
-   For each C-SIC in C-SIC list:
        o   If the list of primary N-DC is not empty
                ▪   For each primary N-DC:
                        •   If the N-DC has sufficient free resources,
                                o   associate it to the C-SIC i
                                o   Remove C-SIC from the C-SIC list
                        •   Else, go to the next N-DC
        o   Else
                ▪   For each secondary N-DC:
                        •   If the N-DC has sufficient free resources,
                                o   associate it to the C-SIC i
                                o   Remove C-SIC from the C-SIC list
                        •   Else, go to the next N-DC
-   If C-SIC list is NOT empty
        o   Exit("Mapping not possible with the current usage level")
-   For each N-DC in primary N-DC list
        o   If N-DC hosts a C-SIC
                ▪   Associate a U-SIC to the N-DC
-   For each N-DC in secondary N-DC list
        o   If N-DC hosts a C-SIC
                ▪   Associate a U-SIC to the N-DC
-   Associate HSS and DNS to a primary N-DC
-   Deploy C-SIC and U-SIC in their respective N-DC
```

2.1.3 EPC Cost Analysis of initial deployment strategies in a DC

A virtualized mobile core network, such as an EPCaaS, is composed of multiple Service Instances (SI) that are interconnected by standardized interfaces forming a Virtual Network Function (VNF) graph. For truly resilient and elastic performance, SIs are decomposed into multiple Service Instance Components (SICs), where each SIC is hosted on a single VM. The optimized initial deployment of these SICs, that constitutes a virtualized mobile network infrastructure, is a challenging task owing to the intra-functional dependencies and constraints between the various SICs.

In this section we analyse the cost incurred by two constraints-based, heuristically applied, initial VNF-graph deployment strategies with reference to virtualized mobile network infrastructure providing EPCaaS, while taking into consideration functional and administrative constraints. The cost of deployment is measured in terms of the utilization of Datacentre Infrastructure (DCI) resources such as computation and networking.

For the analysis and evaluation, we have adopted a simplistic architecture of a virtualized Evolved Packet Core (vEPC) network as a reference for our evaluation [TE-524262]. The main objective is to compare the two constraint-based heuristic approaches of initial deployment of vEPC SICs over a Datacentre Infrastructure (DCI) and analyse the impact of the two deployment strategies on the cost of deployment. These two reference deployment strategies are referred to as Vertical Serial Deployment (VSD) and Horizontal Serial Deployment (HSD) strategies respectively.

[3] The list can be ordered, according to operator's preferences. E.g. it can list POPs acting as egress point towards the Internet before the other ones. Or, this list can be built according commercial agreements between the EPCaaS EEU and the MCNSP: it can contain only a subset of the whole

In this section, a conceptual overview of EPCaaS with reference to vEPC networks will be provided. It will also be described the evaluation framework and evaluation method that includes the modelling of the DCI and vEPC system. This will be followed by the description of the two proposed deployment strategies namely VSD and HSD followed by detailed performance analysis. EPCaaS: Conceptual and functional overview

The objective of EPCaaS is to virtualise the Evolved Packet Core (EPC) infrastructure in order to render the advantages of the cloud system to mobile network operators. This is done by instantiating the EPC system's functional entities (i.e., SIs) such as MME, S-GW, P-GW etc. on VMs over Commercial Off-the-Shelf (COTS) servers instead of specialized mission specific custom tuned expensive hardware platform. To provide the service and design concept of EPCaaS we have adopted a simplistic architecture of a vEPC network [TE-524262] as a reference use case, an overview of which is depicted in Figure 4, where the virtualized functional elements of MME and S/P-GW are referred to as vMME and vS/P-GW respectively.

Four possible architectural reference models for EPCaaS have been specified in [MCN-D4.1] based on how the functional entities are mapped on the VMs, and they are:

1. 1:1 mapping; where each EPC functional entity is implemented on a separate VM.
2. 1:N mapping; where each EPC functional entity is decomposed into sub-functional entities and each sub entity is implemented on a separate VM.
3. N:1 mapping; where the complete EPC system is implemented on a single VM
4. N:2 mapping; it's the same as N:1 except that it combines the C-plane (CP), U-plane (UP) and database services of the EPC onto three separate interconnected VMs.

The vEPC system falls in the category of 1:N mapping where the respective SIs of the vMME and vS/P-GW functions are decomposed into separate CP and UP SICs in order to render enhanced agility and elasticity in view of different traffic and application types.

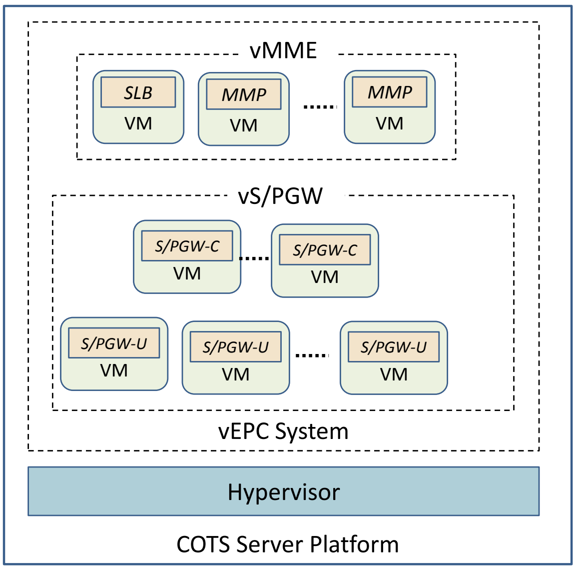


Figure 4 - Functional overview of the vEPC system

Thus the vS/P-GW is divided into two SICs namely vS/P-GW-C and vS/P-GW-U; with the former SIC processing the CP load and the latter processing the UP load. Similarly the vMME functionality is embedded in the combination of a Signalling Load Balancer (SLB) and Mobility Management Processor (MMP) SICs, where the MMP performs the processing task of the MME. The combination of SLB and MMP will allow the scaling of the vMME by using the SLB and by adding/deleting MMPs. Each SIC (i.e., SLB, MMP, vS/P-GW-C and vS/P-GW-U) is realized on a separate VM and the interconnectivity between these SICs based on standard interfaces (see Figure 6).

2.1.3.1 Evaluation framework and methodology

For our cost analysis, we have developed an evaluation framework in C++ that is composed of

1. A DCI model,
2. A vEPC system model and,
3. A deployment model.

For a specific CP/UP input traffic profile, the vEPC system model determines the required number of SICs and their respective resource requirements that will support the incident traffic load. The deployment model, based on a specific deployment strategy, will then deploys the respective SICs on the servers of the underlying DCI model while taking into account the resource requirement of individual SICs and the vEPC system internal bounds and constraints. The framework then computes and determines the cost incurred by the respective deployment strategy in terms of DC networking and compute resource consumption for the incident load profile. Our evaluation framework can be scaled to any size DC and to any size vEPC system depending on the load on the operator's network.

The overview of the DCI model, the vEPC system model and the deployment strategies are discussed in the following subsections.

2.1.3.1.1 Datacentre Infrastructure (DCI) model

For our analysis we have modelled the traditional hierarchical 3-tier DC architecture composed of (i) the core layer, (ii) the aggregation layer, and (iii) the access layer [Cisco-DC]. At the lowest level is the access layer, which contains pools of servers housed in racks, where each server is connected to one (or two for redundancy) top-of-rack (TOR) L2 switch. Each of the TOR switch is in turn connected to one (or two for redundancy) high capacity L2/L3 switch at aggregation layer. The aggregation switches are then connected to the top-level switches/routers forming the core layer. Such a fat-tree topology can be scaled up by scaling up each individual switch.

Figure 5 illustrates the DCI topology that has been modelled, where the dotted lines indicate redundant links thereby connected to the redundant/backup node. For the analysis we do not consider failure scenarios and hence the redundant links/nodes are not utilized. The access layer is modelled as an m x n matrix where (m) is the number of racks and (n) is the number of servers per rack. For our analysis we consider a homogenous access system where all racks are of the same size and all servers are of the same configuration and form-factor. The servers are modelled having x number of CPU cores, and xGbps aggregate network bandwidth. On the other hand the switches/routers are modelled considering xGbps aggregate bandwidths.

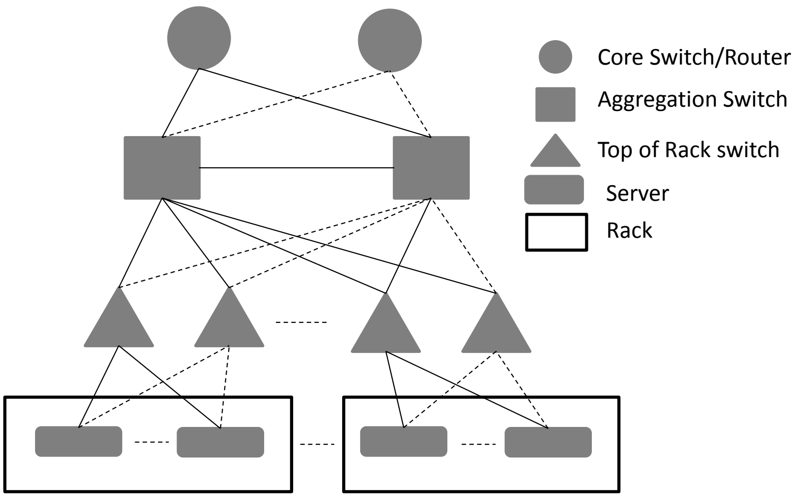


Figure 5 - Three-Layer Date Centre Infrastructure Model

2.1.3.1.2 vEPC System Model

The vEPC system is modelled by characterizing the individual SICs (SLB, MMP, SP-GW-C and SP-GW-U) in terms of the CP/UP load that they process. The model also captures the interfaces between the different relevant SICs as depicted in Figure 6.

Figure 6 illustrates the interconnected SICs constituting the vEPC network with relevant interfaces. The model is able to determine not only the number of relevant SICs required to handle a particular input CP/UP load, but it will also determine the resource requirement of individual SICs in terms of CPU cores and network bandwidth. This information is then used to analyse the deployment cost of the vEPC system in a DCI, thereby enabling the operators to dimension the resources of their respective DCI for specific load conditions and service requirements. This model is expected to provide insight into the resource requirement of every SIC, and thus the size of the overall vEPC system, in response to external inputs. The load models for vMME and vSP-GW are derived with reference to Figure 6, and summarized below.

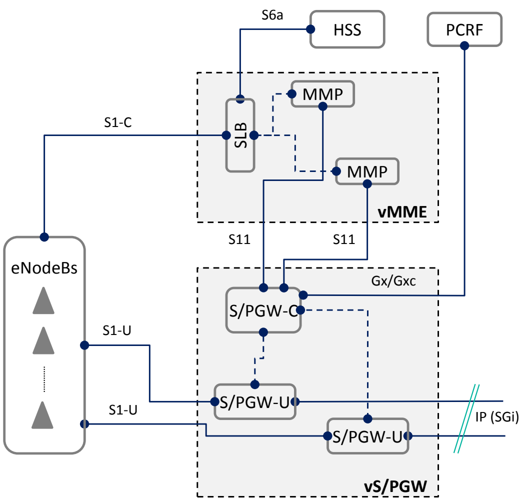


Figure 6 - Interfaces between SICs of the vEPC System

Load model for vMME:

As stated, the vMME is composed of the SLB and MMP virtual instances. The load from the eNBs is balanced by the SLB amongst multiple MMP instances. We assume the SLB to be balancing the load between the MMPs based on an equal weight round-robin manner. The total S1C load on a single SLB instance (i) from eNBs is equal to the aggregate S1C load from the eNBs associated with the SLB and represented as:

$$L_{S1C}^{SLBi} = \sum_{n=1}^{N_{eNB}^{SLBi}} L_{S1C}^{eNBn} \quad .... (1)$$

The load on a single MMP instance (i) is given as:

$$L_{S1C}^{MMPi} = \left\lfloor \frac{L_{S1C,total}^{SLBi}}{N_{MMP}^{SLBi}} \right\rfloor .... (2)$$

Where $L_{S1C,total}^{SLBi}$ is the total CP load on an SLB, for example from the associated eNBs, and N_{MMP}^{SLBi} is the number of MMPs that a single SLB SIC can serve. The ratio between the number of eNBs per SLB and the number of MMPs per SLB is dependent on the load balancing capability of the SLB as well as the maximum load that an MMP can handle i.e., $L_{S1C,\max}^{MMPi}$

Load model for vS/P-GW:

The vS/P-GW is modelled by characterizing the S/P-GW-C and S/P-GW-U SICs in terms of the CP and UP load that the respective SICs process. For a single instance of S/P-GW-C, the total CP load is from the MMPs and the PCRF (see Figure 3) and is represented as:

$$L_{S11,total}^{SPGWCi} = \alpha_{S11} * \sum_{i=1}^{N_{MMP}} L_{S11,total}^{MMPi} + L_{PCRF} \ldots (3)$$

Where the first term of above equation is α = (0; 1] times the total S11 load from the N number of MMPs incident on the S/P-GW-C and the second terms is the CP load from PCRF. The total load processed by the S/P-GW-U is the CP load from the S/P-GW-C and the UP load from the eNBs and is represented as:

$$L_{total}^{SPGWUi} = L_{S11}^{SPGWUi} + L_{S1U,total}^{SPGWUi} \ldots (4)$$

Where

$$L_{S11}^{SPGWUi} = \beta_{S11} * L_{S11,total}^{SPGWci}\, ;\ 0 \le \beta \le 1 \mid \sum_{m=1}^{N_{eNB}} \beta_m \ \ldots\ (5)$$

And

$$L_{S1U,total}^{SPGWUi} = \sum_{i=1}^{N_{eNB}} L_{S1U}^{eNBi} \ldots (6)$$

Deployment Strategies

The following are the two constraint-based and heuristically derived deployment strategies namely:

1. Vertical Serial Deployment (VSD) strategy
2. Horizontal serial deployment (HSD) strategy.

Both the strategies deploy the SICs serially such that the vMME SICs (i.e., SLBs and MMPs) are deployed first followed by the vS/P-GW SICs (i.e., S/P-GW-C and S/P-GWU). In the VSD, the SICs are deployed from top to bottom on servers of one rack, and then moving on to the servers in the next rack when no more resources are available in the previous rack. In HSD, the SICs are deployed on the first available server in rack, then moving on to the next available server on the next rack, and so on till all the SICs are deployed. In other words, considering the access layer as an (*m* x *n)* matrix, in VSD the SICs are deployed column-wise, whereas in HSD the SICs are deployed row-wise.

While deploying, the VSD/HSD deployment strategy will take into account the (anti)affinity between respective SICs, system reliability, server resources in terms of available CPU cores, and the network resources such as the capacity of the network interfaces on the servers and of the links in the DCI.

The following constraints are also taken into consideration during deployment:

1. For reliability, a single server may not have more than one instance of the S/P-GW-U belonging to the same logical vS/P-GW.
2. Each time a SIC is instantiated, the associated standby SIC will also be instantiated.
3. A single server shall not host the active and standby instance of a particular SIC.
4. A SIC is deployed only if the server has the required CPU cores required by the target SIC.

In both VSD and HSD, any server that may not have the resources for a particular SIC or offers anti-affinity with any of the previously installed SIC is skipped over. For our analysis we assume that the servers are all dedicated for vEPC system deployment and no other third party services are running on them.

2.1.3.2 Performance Evaluation

In order to compare and analyse the cost impact of the VSD and HSD deployment strategies on the DCI compute and networking resources, we perform experiments on our evaluation framework using CP load (Ccp) and UP load (Cup) values based on conservative estimates during a busy hour period. According to [D-Nowoswiat] an MME can experience a sustained signalling load of 500-800 messages per UE during busy hour, and up to 1500 messages per UE per hour under adverse conditions. Furthermore according to [T-Parker], the chattiest applications can generate 2400 signalling events per hour. Based on these observations, we assume a tri-cell eNB having 30 users per cell that generate the bulk of traffic events. For our scenario, we assume 5% of users generating 2400 ev/hr, 25% producing 800 ev/hr while 70% producing 500 ev/hr during busy hour. Thus during busy hour a vEPC system will encounter 60300 ev/hr from a single eNB. Based on the incident load, the vEPC system model will compute the required number of SICs, which is then deployed by the respective deployment strategy (i.e., HSD and VSD) on the DCI model in view of the constraints and affinity between the relevant SICs. The access layer of the DCI is modelled as a 4 x 45 matrix and all the 180 servers have 16 cores each.

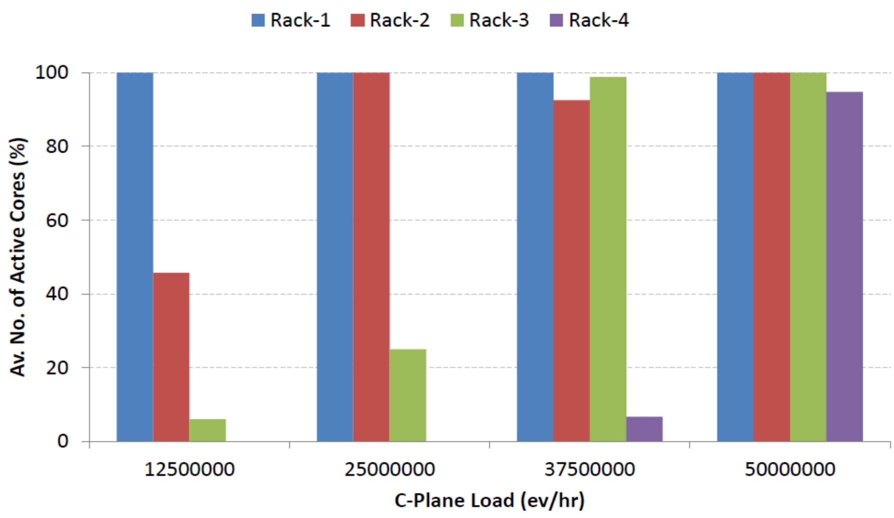
For simplicity we assume all SICs are assigned 4 CPU cores during deployment. Our evaluation framework also provides the standby SICs based on 1+N redundancy but since we are not considering failure scenario therefore we will not consider the standby SICs and corresponding links for throughput calculations. Besides, we also ignore the LPCRF during calculations. The rest of the parameters used in our simulation framework are listed in Table 2.

The performance of two deployment strategies (i.e., VSD and HSD) are measured with respect to the average number of active cores utilized per rack (Figure 7) and the average throughput per rack (Figure 8) for four reference *Ccp* values.

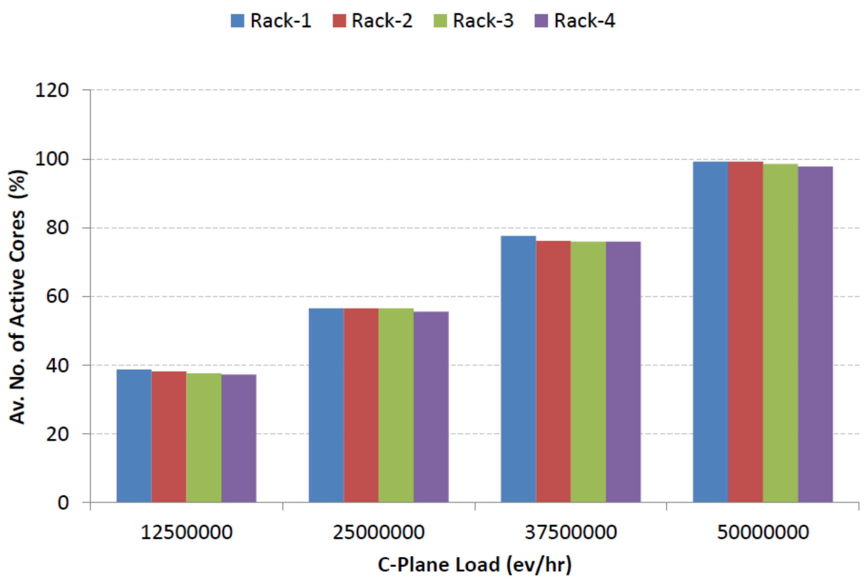
Table 2: Simulation Parameters

Parameter	***Notation***	***Value***
Total CPU Cores per SIC	N_{core}^{SIC}	4
eNB per SLB	N_{eNB}^{SLBi}	100
Number of S/P-GW-U per SP-GW-C	N_{SPGWU}^{SPGWCi}	6
MaximumS1C load per MMP	$L_{S1C,max}^{MMPi}$	500,000 ev/hr
Maximum S11 load per SP-GW-C	$L_{S11,max}^{SPGWCi}$	1,000,000 ev/hr
Maximum S11 load per SP-GW-U	$L_{S11,max}^{SPGWUi}$	166666.7 ev/hr
$\alpha_{S11};\beta_{S11}$		0.5
CP load demand (ev/hr)	C_{cp}	$x = L_{S1C,max}^{MMPi}$ *Where x = 0.25; 0.50; 0.75; 1.0*
Number of eNBs, where each value corresponds to the respective value of C_{cp}	N_{eNB}	[1500, 2000, 2500, 3000]
Average CP packet size (in bytes)		192
Av. number of messages per CP event		6
UP load demand (Gbps)	C_{up}	[64, 128, 256, 512]
UP Packet size (in bytes)		512

As evident, the deployment strategy has a marked and substantial effect on the distribution of the number of active cores on a per rack basis. With VSD (Figure 7(a)), 100 % of all the cores (and hence all the servers) are utilized in Rack-1, while the cores in other racks become active sequentially with increasing load. As a result, there are load conditions where a rack (and hence the servers in it) may remain completely inactive and un-utilized. For example, the servers in Rack-4 remain un-utilized for $C_{cp} = 12.5x10^6$ *ev/hr* and $C_{cp} = 25x10^6$ *ev/hr*. This will cause un-even load distribution over the access links and hence the ToR switches, where one link or switch may become overloaded, while the others may remain un/under-utilized. This is evident from Figure 8(a) where the load is unevenly distributed amongst servers in the four racks. For example, for $C_{cp} = 50x10^6$ *ev/hr* all the load is on servers on rack 3 and 4, whereas racks 1 and 2 have no load on them.

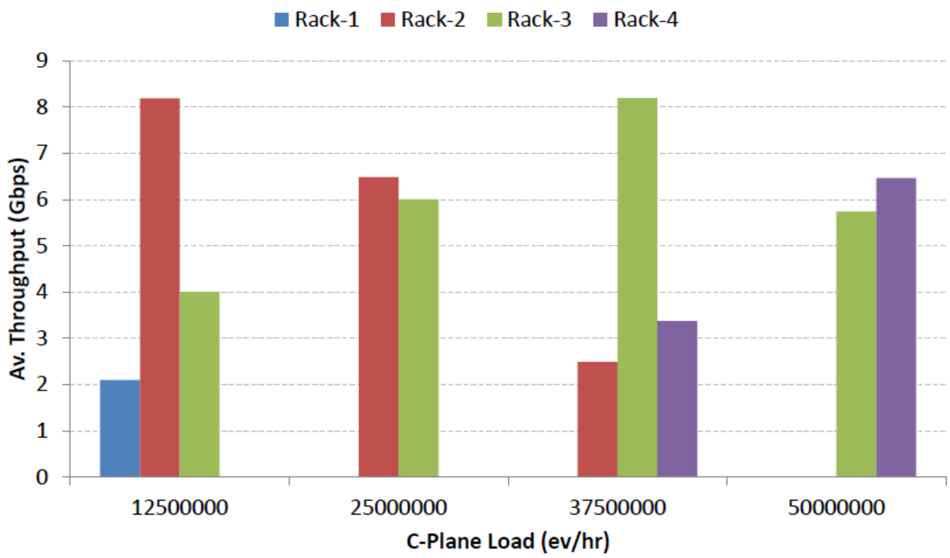


(a) VSD

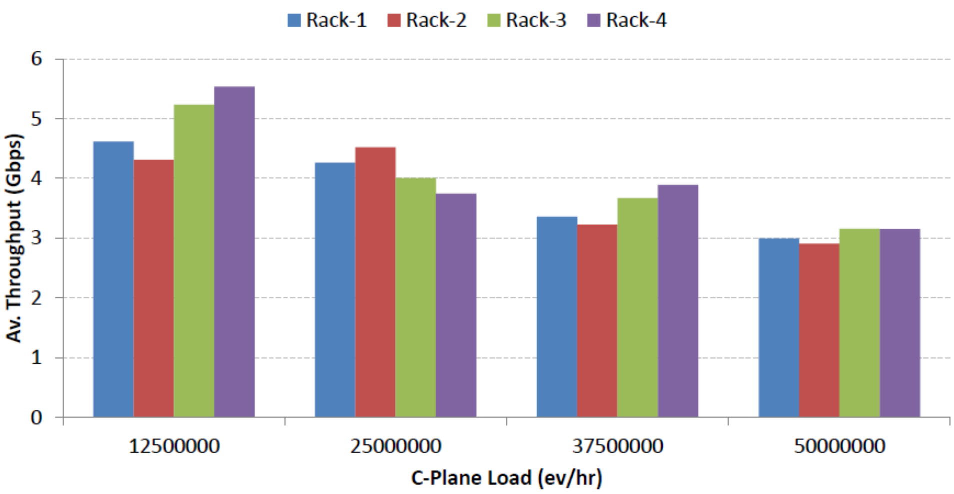


(b) HSD

Figure 7 - Average number of active cores per rack



(a) VSD



(b) HSD

Figure 8 - Average throughput per active server for CP load = 512Gbps

In contrast to VSD, the HSD deploys the SICs evenly across the racks resulting in even and optimum utilization of the compute and network resources for all load conditions. This can be observed from Figure 7(b) where for all load conditions, the SICs are deployed evenly across the racks and thus the CPU core assignment is even. This will also ensure even distribution of load over the access links and hence the ToR switches as evident from Figure 8(b). Thus in contrast to VSD, the HSD strategy results in the optimal utilization of the DCI resources without overloading any particular set of servers and the resources scale evenly with increase in input load.

As evident from the results, for specific load profile the total number of active servers and active cores are the same for both HSD and VSD. However, HSD delivers the best performance in terms of even distribution of load over all the servers, access links and hence the ToR switches. In fact, for higher load profiles HSD will result in reduced average throughput per active servers as the load is evenly distributed across all racks while the number of active servers increases. In contrast to HSD, VSD is not efficient as it will cause uneven distribution of SICs and hence load on the servers. This may make specific racks, and hence the servers in them and associated links to be 100% utilized while some other racks with servers may remain completely underutilized. The scalable evaluation framework developed for the analysis can be used by the operators to appropriately dimension their respective DCI to meet the expected peak traffic demands while optimizing the utilization of available resources.

2.1.4 ANDSF Initial Placement Algorithm

This sections assumes that the reader is familiar with the architecture and concepts described in [MCN-D4.2], sections 2.3 and 5.7, regarding the implementation of the ANDSF, and with the 1:N approach described in [MCN-D4.1], section 2.8.1.2.

This section documents the set of algorithms that can be applied by the Service orchestrator (SO), in order to initially provision an ANDSF instance. As this is the initial provisioning, there is no runtime information available, such as monitoring (coming from the MaaS). For this reason, information about prediction must be used as an initial guess, as described in detail in [MCN-D4.1], section 2.3.1.

The following should be the main metrics to consider for prediction:

- **Infrastructure KPIs**
  - *Predicted CPU Utilization*
  - *Predicted Memory RAM Utilization*
  - *Predicted Storage Utilization*
  - *Predicted Network Utilization*

However, note that in practice it is hard to predict those values. For this reason, the main objective should be to benchmark the platform for a common set of hardware types and, based on key predicted/contracted service performance parameters (KPIs), define a set of elements (VMs, storage) that will map into the initial deployment figures (the best possible guess).

e.g. Prediction/Contract: 1000 Requests/second -> 1-LB;1-WS;3-IRP;1-DB

Note that this is just an initial placement and the service will have time to adjust to the actual requirements, based on the reception of monitoring (MaaS). This adjustment follows the mechanisms and algorithms described in section 2.2.2.

- **Service KPIs**
  - Contracted (Max) Number of Requests
  - Predicted Number of Requests
  - *Predicted Number of Requests LB (depend on the first two)*
  - *Predicted Number of Requests WS (depend on the first two)*
  - *Predicted Number of Requests IRP (depend on the first two)*
  - *Predicted Number of Requests DB (depend on the first two)*

The maximum contracted number of requests and the predicted number of request are main service metrics to be considered. This means the total number of requests that the service receives. Note that based on this number, it can be easily calculated the number of request received by each tier.

The total number of requests received is the same as the number of requests received by the LB, since this tier receives all the traffic. For WS and IRP tiers, the total number of requests is the same as LB. Therefore, the number of requests per component will depend on the number of components, since they will be split (round robin) among them. For the DB tier, it must be multiplied by 2, since for each service request are required 2 accesses to the database. In short, that means that based on the predicted service requests all other KPIs can be achieved.

Other information should also be considered:

- Location
  - DC Locations
  - DC Costs
  - DC Hardware/Software Requirements
  - User Locations
  - Service Locations (only for Data Plane components)
- Bandwidth and Traffic
  - Inter-DC Bandwidth (total/per-tenant)
  - Predicted Inter-DC Traffic (total/per-tenant)
  - Intra-DC Bandwidth (total/per-tenant)
  - Predicted Intra-DC Traffic (total/per-tenant)

In the particular case of the ANDSF services, all tiers that compose a single instance should be in the same DC. There are no advantages on splitting a single instance. In case the ANDSF should cover a large area, a better option is to create multiple instances and balance them.

The following sections describe the major algorithms that should be implemented.

2.1.4.1 Contracted/Predicted Nº Requests

This algorithm intends to perform an initial placement of the ANDSF, based on information regarding the total number of request per second that the service expects to receive. This information may come from two different sources: contracts and predictions (or any combination of them). Contracts define the service level agreement (SLA) that a customer signed with the service provider. It is expected that the contract states the maximum allowed (eventually the average) number of requests per seconds. A fraction of this value can be a very good candidate for the initial best guess. Prediction mechanisms can also be used to perform a reasonable forecast of the number of requests per second. Prediction can also take into consideration the historical load on different periods of time, adjusting the values to the different hours of the day or days of week. For this reason, a good approach could be the combination of both sources.

Although this first value is important to be as approximate as possible, after this initial placement, the service will be able to adjust dynamically during runtime as described in section 2.2.2.

Since the target number of requests is found based on contracts and/or prediction, algorithms has to define how many components of each tier are required (considering a particular VM template). This mapping should be based on intervals and a correspondent set of components. The following table shows an example of mapping (nº requests, components list).

Table 3 – Initial placement mapping (nº requests, components list).

Name	Mapping (nº requests, components list)
Description	Maps the expected number of requests per second into a list of ANDSF components, indicating the number of components per tier: LB, WS, IRP and DB.
Example	Nº Requests <= 500 - 1 LB (LB) - 1 WS (FE) - 1 IRP (Worker) - 1 Mongo (DB) 500 < Nº Requests <= 1000 - 1 LB (LB) - 1 WS (FE) - 2 IRP (Worker) - 1 Mongo (DB) 1000 < Nº Requests <= 1500 - 1 LB (LB) - 1 WS (FE) - 3 IRP (Worker) - 1 Mongo (DB) 1500 < Nº Requests <= 2000 - 1 LB (LB) - 2 WS (FE) - 4 IRP (Worker) - 1 Mongo (DB) Nº Requests > 2000 - 1 LB (LB) - 3 WS (FE) - 5 IRP (Worker) - 1 Mongo (DB)

This algorithm can be represented using a generic rules engine as follows:

```
andsf_req_1 = 500
andsf_req_2 = 1000
andsf_req_3 = 1500
andsf_req_4 = 2000

# Rule-001: Interval #1
rule "Rule-001-ANDSF-Init":
        agenda-group andsf_init
        when:
                $Prediction := Prediction(n_req <= andsf_req_1)
        then:
                insert Decision("init",{'lb':1,'ws':1,'irp':1,'db':1})
                halt

# Rule-002: Interval #2
rule "Rule-002-ANDSF-Init":
        agenda-group andsf_init
        when:
                $Prediction   :=   Prediction((n_req   >   andsf_req_1)   and
(n_req <= andsf_req_2)
        then:
                insert Decision("init",{'lb':1,'ws':1,'irp':2,'db':1})
                halt

# Rule-003: Interval #3
```

```
rule "Rule-003-ANDSF-Init":
        agenda-group andsf_init
        when:
                $Prediction   :=   Prediction((n_req   >   andsf_req_2)   and
(n_req <= andsf_req_3)
        then:
                insert Decision("init",{'lb':1,'ws':1,'irp':3,'db':1})
                halt

# Rule-004: Interval #4
rule "Rule-004-ANDSF-Init":
        agenda-group andsf_init
        when:
                $Prediction   :=   Prediction((n_req   >   andsf_req_3)   and
(n_req <= andsf_req_4)
        then:
                insert Decision("init",{'lb':1,'ws':2,'irp':4,'db':1})
                halt

# Rule-005: Interval #5
rule "Rule-005-ANDSF-Init":
        agenda-group andsf_init
        when:
                $Prediction := Prediction(n_req > andsf_req_4)
        then:
                insert Decision("init",{'lb':1,'ws':3,'irp':5,'db':1})
                halt
```

2.1.4.2 Other Service-based scaling

Although the number of service requests is the most representative KPI of the required performance of the ANDSF service, other service metrics can be used to map into a set of virtual resources. Examples of this are the total number of subscribers provisioned, or the number of applicable rules.

In the first case, it has a direct impact on the storage required to allocate to the DB component. The DB (supporting the SPR - Subscriber Profile Repository) stores the profile of subscribers, which allows the ANDSF to provide a per-subscriber provisioning of network discovery and selection rules. In the second case, the impact is reduced since the number of rules is expected to be reduced (tens, maximum hundreds) and the storage requirements will not be significant anyway.

In both cases, the process is similar to the mapping exposed for the number of requests. The only difference is that in this case the type of resources to be requested should be more related to storage than computation (VMs).

2.2 Network Function Runtime Migration

Runtime migration algorithms consider all policies used during the EPC service lifetime. This is the set of action that the Service Orchestrator (SO) can do as a result of the service feedback, namely regarding monitoring or prediction. Section 2.1 describes the algorithms that can be used for an initial service deployment. The algorithms here described are aligned with the description in [MCN-D4.1], section 3.3.

This sections considers algorithms for the EPC service as a whole, comprising by the MME, S-GW, P-GW and HSS components, and using the N:2 architectural approach; and the ANDSF service, comprising an (optional) EPC component (responsible for sending discovery and attachments policies to the terminals), and using the 1:N architectural approach.

2.2.1 Network Functions Placement Decision Framework for Highly Distributed Cloud Infrastructures

2.2.1.1 Problem Statement

With the proliferation of datacentres and with the aggregation within a common ecosystem of multiple heterogeneous cloud infrastructures pertaining to various tenants, a novel network infrastructure is emerging with different cloud nodes at different network locations. Across this entire distributed infrastructure, at edge, medium and macro levels, a set of virtual network functions and underlying virtual networks will be deployed.

The main use case scenario, assumed by this research work, is the one where the deployed virtual services are medium in regard to the infrastructure size i.e. the virtual service should be deployed on more than one of the datacentres and in less than the complete infrastructure, specifically addressing the emerging market of enterprise and private virtual networks and highly distributed networks.

> Note: For those cases where the virtual service size is large, it is assumed that there will be a need for the virtual service in all the network areas, thus all the datacentres will include at least a minimal support for that functionality (such as one VM) and elastically scale when needed.

> Note: For small virtual service size architectures, where the virtual service size is small in comparison to the infrastructure), all the network functions will be placed in all, or a single, Macro DC for size efficiency reasons.

For anchoring the further development of the network functions placement decision, a simplified network model is assumed, as depicted in Figure 9. The model consists of:

- Macro datacentres – centralized datacentres, considered as default deployment nodes following the considerations of the TI section;
- Micro datacentres – regional datacentres, which can serve a specific region, however, do not cover the complete mobility of the subscribers i.e. a better micro datacentre may be selected during the course of a day for a considerable amount of subscribers between 1% and 20%;
- Edge datacentres – base station or street cabinet datacentres which serve a very small area with the assumption that a very large amount of connected devices will be best served by a large number of edge datacentres in the course of a day between 50% and 100% depending on the number of wireless connected fixed devices in the system.

The framework here described is based on a live operator mobility and service usage trace to determine when it is best to place functionality closer to the edge, within the micro or the edge datacentres, considering the following main parameters:

- Load/Delay – An edge node will relief the network from part of the load and at the same time will impact the end-to-end service delay as a result from removing one link from the data path compared to the centralized solution
- Mobility – For a mobile device served by an edge node there is a larger chance of requiring handover procedures within the network, depending on the device characteristics.
- Management – the complexity of management is highly increased with the distribution of the datacentres, especially in the number of required connections, heterogeneity and communication [Cast96].

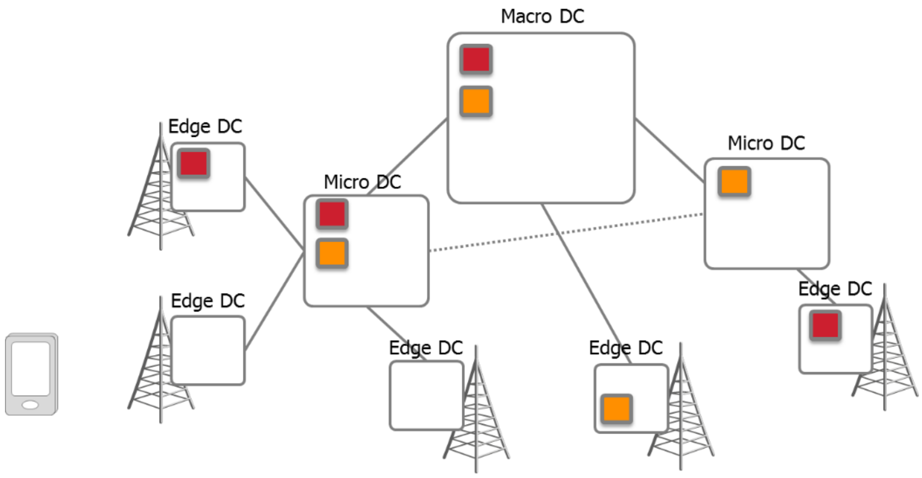


Figure 9 - Network Model Considered

The framework mainly addresses the placement of EPC components in the form of software on top of the different datacentres. However, the same logical framework can be used for other services.

As a generic use case, it offers the mechanisms for a light dimensioning process addressing the specific medium size virtual services.

2.2.1.2 Edge Function Placement Decision

When making a knowledgeable edge function placement decision, two parameters enable the reduction of complexity. First, a new edge node takes over functionality from a next level core node. Through this limitation, only local decisions have to be taken, reducing the complexity of the overall system to only a specific area. The function placement is independent of the momentary needs of the system. It is assumed that a less optimized and highly stable system can be achieved by using the next level core node. Thus, there is no stringent need of optimization in order to maintain the continuity service.

For adding a new node at the edge of the network, two decisions have to be made. First determine whether a new edge network is suitable for the specific virtual network infrastructure. Secondly, whether the node is efficient in supporting the roaming devices across the specific operator area e.g. that the obtained optimization is significantly larger than the management costs and the operational costs resulting mainly from the handover. In the following sub-sections, the framework for the two decisions is described.

2.2.1.3 Sampling a Specific Network Area

The most important piece of information is the current placement of the end-user subscribers. A representation at the appropriate level of granularity is a values vector, including in each field the momentary usage of a cell in the specific network area.

$$S_x = [a_{x1}, \ a_{x2}, \ a_{x3}, \ a_{x4}, \ a_{x5}, \ \dots, \ a_{xn}]$$ where n is the number of cells

Since the subscriber hand-over is only confined to the neighbourhood cells, different clusterisation and partial computations can be considered.

A cluster can be represented by a set of cells. These cells are represented as a single value in the state vector. They do not have individual datacentre connectivity. This is especially important for macro- to micro-datacentre functionality transfer, as it allows a lower size vector.

Also, for partially localized computations, a cluster may be represented by all the other cells, which are not in the scope of the specific decision. The granularity decision lies in the infrastructure provider since it is the one that is aware of the datacentre placement.

Additionally, as there is no need for individual subscriber information, not for providing the necessary tilt for a network function decision, the values in the state vector can be normalized to a number of subscribers e.g. x50, x100, x200, x1000, etc. as to reduce the impact of low movement variations. This allows to scale the system up to millions of subscriber devices through the major reduction in number of states.

Through this information a momentary view on the network is achieved. Additionally to the momentary load view, mobility has to be accounted for. For this reason, a complete state machine can be created. The specific states have to be accounted with different sampling rates . As there is no need for a very short delay procedure, the sampling may happen even at the level of minutes, thus reducing the information flow.

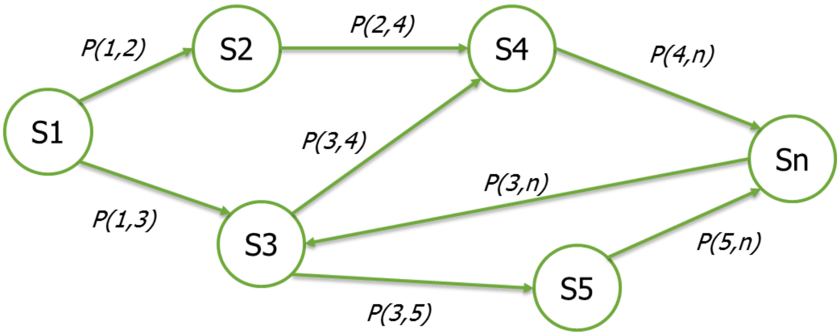


Figure 10 - State Machine

Based on a long-term sampling (e.g. 1 month) a transition graph can be generated and different probabilities can be associated with different transitions. The main reason for such an operation is to discover correlated areas and clusters. There are several parameters that make such a computation possible:

1. Human mobility takes place in clusters i.e. within the same city. Thus the sample can be easily cut into smaller size vectors. Additionally, as previously discovered [Bara08], [Bara10].
2. A complex system can be simplified starting by considering larger clusters and then moving the focus on the further detail, dependent on the need. [BakP95].

Using basic data mining algorithms, different maximums on top of a different set of data (i.e. set of cells in a specific network area) can be evaluated through the different states. The purpose of this procedure is to determine whether there is a very large probability that a set of cells in a specific area is occupied for a long duration of time, making it a potential candidate for a new edge network function.

Note: a separation between control and data plane can be considered. Based on the specific subscriber dimensioning a statistic load distribution may be assumed. For example, from 100 subscribers, in average 10 are active and 1 is consuming a very high level of resources. Alternatively, a computation of load on a specific area can be computed at the anchor point (e.g. P-GW or DPI), thus simplifying load computations. This is considered a too costly optimization compared to the use of traffic engineering and statistical approximation on the usage of subscribers. However, such a claim cannot be substantiated without real system data.

2.2.1.4 Sampling the subscribers in the area

To completely assess the opportunity of a new edge node, the subscribers in the specific area should be analysed. For the most pre-eminent location, as determined in the previous step, the individual subscribers have to be identified.

The subscriber devices can be classified into different clusters depending on the camping time and on the specific mobility pattern i.e. the direction of the handovers. Especially a large number of fixed or limited mobility devices would require the deployment of an edge node. Additionally, devices that do not change the network location often – e.g. camp in a specific network area – require a low number of handovers/day are considered an incentive for edge networking.

On the other side, the users that execute a handover to a specific area only to be immediately followed by another handover outside of it are considered unsuitable for edge network function support due to the large number of procedures to be executed.

$$Ux = [U1\ (u11, \dots u1n(1)),\quad U2\ (u21, \dots u2n(2))\quad ,\ \dots ,\quad Um\ (um1, \dots umn(m))\ ]$$

where m is the number of clusters and n(x) is the number of subscribers in cluster x

As there is no perfect fit between multiple subscribers, a specific error rate should be considered for a cluster of subscribers e.g. a subscriber pertains to a low mobility cluster if it hands over out of a specific network area more than one hour after handing over in the specific network area. The higher the granularity of a cluster is larger the complexity. For reducing the complexity, the subscriber sampling should be executed at the largest group possible (e.g. users that spend 20min/1h/2h/… Nh in the specific area) and with a very large sampling. Additionally, subscribers which have very high speed should be removed from the sampling all together.

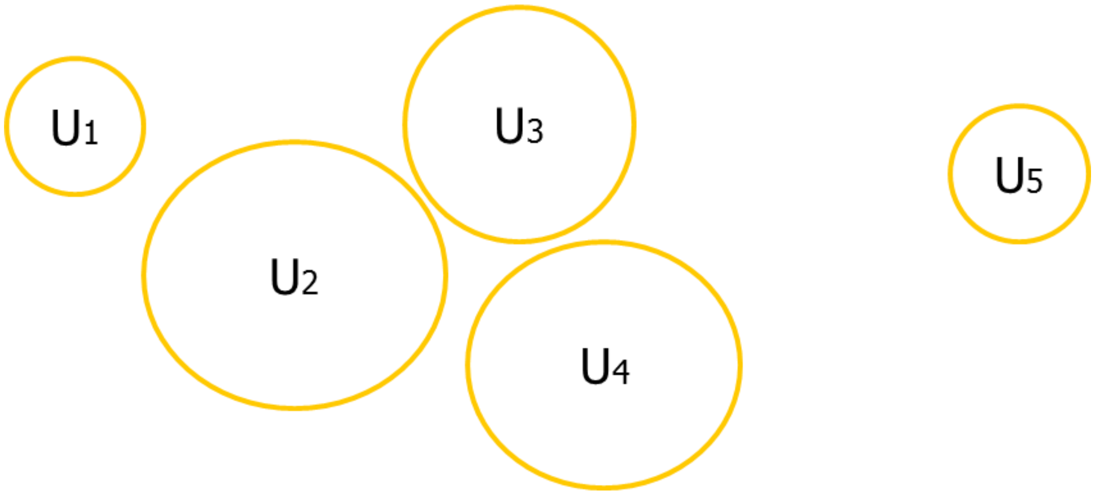


Figure 11 - Subscribers Cluster

2.2.1.5 Considerations on the Learning Process

With the evolution of the network, a different set of load and subscriber's state can be induced including, the gradual or steep increase of connected devices or the difference on the achieved mobility. For this, the load and the subscriber clustering should be continuously analysed and the edge network functions placement should be appropriately adopted. Additionally, situational states should be derived, forcing specific end-user subscriber clustering. A generic state relationship is defined in Figure 12.

Regarding **Load States (L1, L2, …)**, there are two distinct possibilities of change: the probability of the transition between one state and the next one may change in time and the apparition of new states. The probability may change due to external factors such as changes in the human situational context (e.g. a new office building, etc.) as well as due to the changes in the usage of the virtual service (e.g. an additional service supporting a specific number of users, etc.). The new states appear with the adoption of the services by a larger number of subscribers and by changes in the human behaviour in regard to the offered virtual services (e.g. changing from initial enthusiastic usage to habitual usage). These state changes can be observed at the network level and do not require an advanced understanding of the situational context of the subscribers.

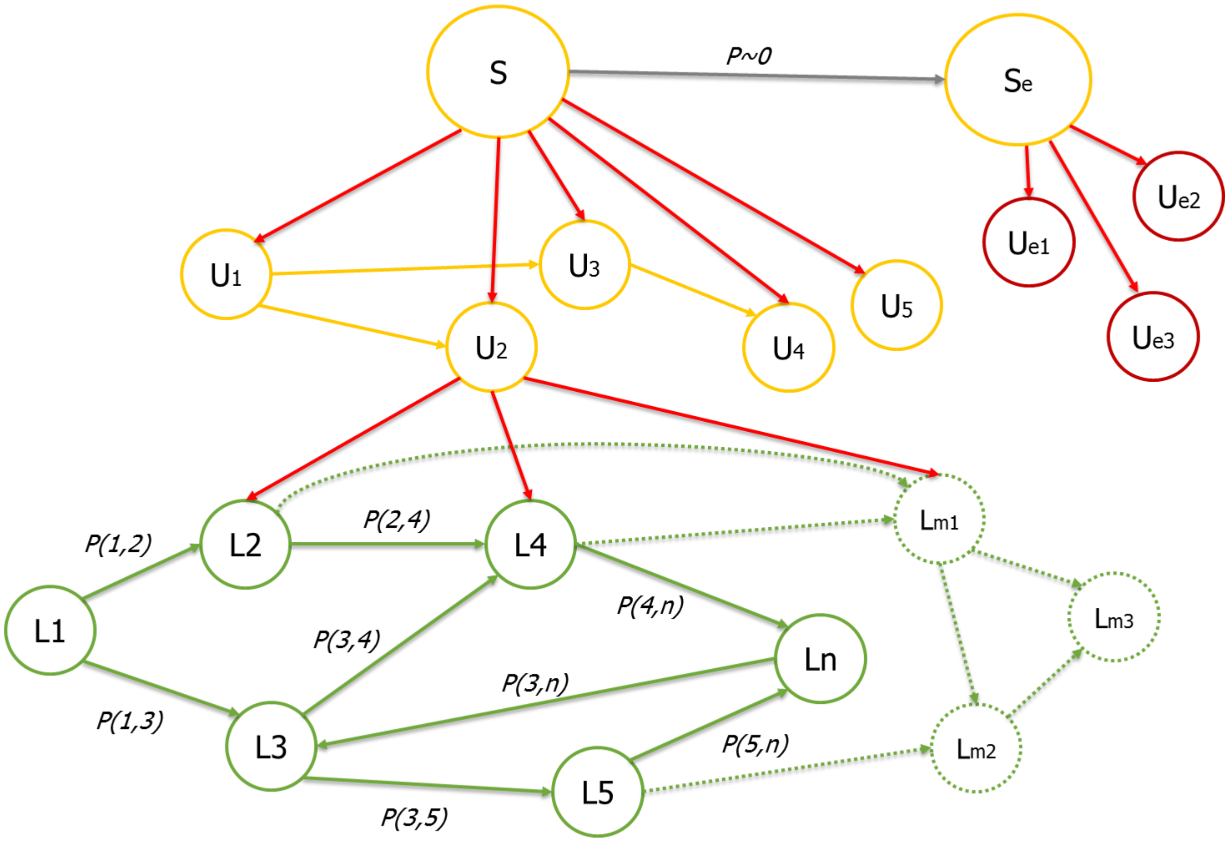


Figure 12 – Learning Process and equivalence of states

The **User Clusters (U1, U2, …)** represent a means of expressing traces based on users on top of the load state machine. For example Cluster U1 of users contributes to the specific state distribution of [L2, L4, Ln]. It is expected that the user clusters will be as complex to the load states as single flow processing is to the routing process. Thus, such clustering should be executed only in the specific network areas where it has the most meaning.

Similarly to the load states, also the cluster states are evolving through migration of subscribers between different clusters. As the human behaviour, and of their devices is rather predictable, the migration between different clusters should happen rarely and for a large number of subscribers. Thus, for immediate network function optimizations there is no stringent need for this information.

A mapping between clusters of users and load states enables the creation of a **situational context** (S), which includes as much information as needed on the human mobility while not deterring scalability. While sampling the network for a specific duration, with no major or exceptional human situational events, (i.e. without including winter holidays or world football championships) a specific dynamic dimensioning of the network is achieved. The situational context can be easily split into multiple states, depending on the time of the day, day of the week or other situational contexts that may be acquired through human analysis (e.g. football games, motorway accidents, etc.).

Starting from the default situational context, alternative ones can be imagined, through human intervention and analysis of the specific network functions, such as emergency or disaster cases (Se). As already analysed in disease spreading studies, new user clustering is achieved based on the initial context before the emergency state (e.g. if users are at home or at the work place). When an extreme event occurs, a different set of load states apply, enabling the automatic adaptation of the networks functions. Even though the probability of these events is very low, due to their importance, it is outmost necessary to prepare a high availability in advance.

2.2.1.6 Summary, Conclusions and Next Steps

This section presented, on a high level, a framework for supporting dynamic network function placement addressing especially medium size virtual network infrastructures running on top of a virtualized framework.

The framework here presented comes to replace the dimensioning process which currently is executed for large Telco networks with a lightweight reactive mechanism which starts from a default state – having each network function replicated in each of the macro- datacentres in the area of interest – and ends up with a distributed architecture which has part of the load handled by micro- or edge network datacentres.

Three main sources of information are considered:

- the load of the network infrastructure/per cell/per time which allows the evaluation of the basic opportunity to place one network function at a specific network location;
- the user clustering information which groups users with the same behaviour either from a time and/or from a network location perspective;
- the situational context (either default, exceptional and sampled or exceptional and foreseen).

As there are currently no foreseen means to access user clustering information from a Telco operator network, the here presented framework is designed to provide an initial setup for the further development of an exemplary evaluation (which will be included in D4.5) of such states in an example environment and a computation of the specific algorithm complexity.

The main role is to provide a suitable realistic set of use cases to be later implemented as part of distributed virtual infrastructure test components, which will allow for the accounting of complex situational cases and to increase the trust in distributed virtualized infrastructure products.

Furthermore, the results will contribute as input for the benchmarking tool developed for virtual EPCs by Fraunhofer FOKUS and TU Berlin and to provide a fundament for the different mobility and QoS traces which will be later experimented with.

2.2.2 ANDSF Runtime Migration

This sections assumes that the reader is familiar with the architecture and concepts described in [MCN-D4.2], sections 2.3 and 5.7, regarding the implementation of the ANDSF, and with the 1:N approach described in [MCN-D4.1], section 2.8.1.2.

This section documents the set of algorithms and policy rules that can be applied by the Service orchestrator (SO), in order to scale the ANDSF, based on monitoring data coming from the MaaS during runtime. According to a set of KPIs, described in detail in [MCN-D4.1], section 3.3.1, some rules will be applied.

The following are the main metrics to consider:

- **Infrastructure KPIs**
    - CPU Utilization

- Memory RAM Utilization
- Storage Utilization
- Network Utilization

- **Service KPIs**
  - Number of Requests
  - Number of Requests LB
  - Number of Requests WS
  - Number of Requests IRP
  - Number of Requests DB

Other information to be considered:

- **Location**
  - DC Locations
  - DC Costs
  - DC Hardware/Software Requirements
  - User Locations
  - Service Locations (only for Data Plane components)
- **Bandwidth and Traffic**
  - Inter-DC Bandwidth (total/per-tenant)
  - Inter-DC Traffic (total/per-tenant)
  - Intra-DC Bandwidth (total/per-tenant)
  - Intra-DC Traffic (total/per-tenant)

In the particular case of the ANDSF services, all tiers that compose a single instance should be in the same DC. There are no advantages on splitting a single instance. In case the ANDSF should cover a large area, a better option is to create multiple instances and balance them.

The following sections describe the major algorithms that should be implemented.

2.2.2.1 CPU-based scaling

This algorithm intends to scale-in or scale-out the ANDSF worker tier (IRP), based on CPU average utilization measurements. Based on two thresholds, maximum and minimum, the SO scales-out or -in the number of workers, respectively. The scale-out takes place when the CPU utilization is above the maximum threshold, while the scale-in is performed when the CPU utilization is below the minimum threshold.

Note 1: The threshold values should be defined properly in order to avoid platforms to keep continuously scaling-in and -out. The minimum number of workers is always 1.

The following tables show in detail the two different scaling operations that can be taken.

Table 4 - CPU-based scaling-out.

Name	**CPU-based scaling-out**
Description	Anytime the average CPU utilization of SIC Workers raises above a certain level (e.g. 70%), the SO instantiates one more unit (VM), in order to cope with the lack of computing resources. *Note: During a (guard) period, no other scaling operation is allowed.*
Example	Before, the ANDSF is composed by: - 1 LB (LB) 20% CPU - 1 WS (FE) 20% CPU - **1** IRP (Worker) **80%** CPU - 1 Mongo (DB) 30% CPU After, the ANDSF will be composed by: - 1 LB (LB) 20% CPU - 1 WS (FE) 20% CPU - **2** IRP (Worker) **40%** + **40%** CPU - 1 Mongo (DB) 30% CPU

Note: Of course this is a theoretical exercise; in reality, the effect will not be exactly like this.

Table 5 - CPU-based scaling-in.

Name	**CPU-based scaling-in**
Description	Anytime the average CPU utilization of SIC Workers falls below a certain level (e.g. 30%), the SO instantiates one more unit (VM), in order to cope with the lack of computing resources. *Note: During a (guard) period, no other scaling operation is allowed.*
Example	Before, the ANDSF is composed by: - 1 LB (LB) 20% CPU - 1 WS (FE) 20% CPU - **2** IRP (Worker) **25%** + **25%** CPU - 1 Mongo (DB) 30% CPU After, the ANDSF will be composed by: - 1 LB (LB) 20% CPU - 1 WS (FE) 20% CPU - **1** IRP (Worker) **50%** CPU - 1 Mongo (DB) 30% CPU

Note: Of course this is a theoretical exercise; in reality, the effect will not be exactly like this.

This algorithm can be represented using a generic rules engine as follows:

```
andsf_cpu_threshold_min = 30
andsf_cpu_threshold_max = 70

# Rule-001: scale-out CPU threshold
rule "Rule-001-ANDSF-CPU-Scale-out":
        agenda-group maas_andsf_cpu
        when:
                $MaaS := MaaS( cpu_avg > andsf_cpu_threshold_max)
        then:
                insert Decision("scale-out",{'irp':1})
                halt

# Rule-002: scale-in CPU threshold
rule "Rule-002-ANDSF-CPU-Scale-in":
        agenda-group maas_andsf_cpu
        when:
                $MaaS := MaaS( cpu_avg < andsf_cpu_threshold_min )
        then:
                insert Decision("scale-in", {irp':1})
                halt
```

2.2.2.2 Memory-based scaling

This algorithm intends to scale-in or scale-out the ANDSF worker tier (IRP), based on Memory average occupation measurements. Based on two thresholds, maximum and minimum, the SO scales-out or -in the number of workers, respectively. The scale-out takes place when the Memory occupation is above the maximum threshold, while the scale-in is performed when the Memory occupation is below the minimum threshold.

Note 1: The threshold values should be defined properly in order to avoid platforms to keep continuously scaling-in and -out. The minimum number of workers is always 1.

The following tables show in detail the two different scaling operations that can be taken.

Table 6 - Memory-based scaling-out.

Name	**Memory-based scaling-out**
Description	Anytime the average Memory occupation of SIC Workers raises above a certain level (e.g. 70%), the SO instantiates one more unit (VM), in order to cope with the lack of memory resources. *Note: During a (guard) period, no other scaling operation is allowed.*
Example	Before, the ANDSF is composed by: - 1 LB (LB) 20% Memory - 1 WS (FE) 20% Memory - **1** IRP (Worker) **80%** Memory - 1 Mongo (DB) 30% Memory After, the ANDSF will be composed by: - 1 LB (LB) 20% Memory - 1 WS (FE) 20% Memory - **2** IRP (Worker) **40% + 40%** Memory - 1 Mongo (DB) 30% Memory

Note: Of course this is a theoretical exercise; in reality, the effect will not be exactly like this.

Table 7 - Memory-based scaling-in.

Name	**Memory-based scaling-in**
Description	Anytime the average Memory utilization of SIC Workers falls below a certain level (e.g. 30%), the SO instantiates one more unit (VM), in order to cope with the lack of computing resources. *Note: During a (guard) period, no other scaling operation is allowed.*
Example	Before, the ANDSF is composed by: - 1 LB (LB) 20% Memory - 1 WS (FE) 20% Memory - **2** IRP (Worker) **25% + 25%** Memory - 1 Mongo (DB) 30% Memory After, the ANDSF will be composed by: - 1 LB (LB) 20% Memory - 1 WS (FE) 20% Memory - **1** IRP (Worker) **50%** Memory - 1 Mongo (DB) 30% Memory

Note: Of course this is a theoretical exercise; in reality, the effect will not be exactly like this.

This algorithm can be represented using a generic rules engine as follows:

```
andsf_memory_threshold_min = 30
andsf_memory_threshold_max = 70

# Rule-001: scale-out Memory threshold
rule "Rule-001-ANDSF-Memory-Scale-out":
      agenda-group maas_andsf_memory
```



```
                when:
                        $MaaS := MaaS( memory_avg > andsf_memory_threshold_max)
                then:
                        insert Decision("scale-out",{'irp':1})
                        halt

        # Rule-002: scale-in Memory threshold
        rule "Rule-002-ANDSF-Memory-Scale-in":
                agenda-group maas_andsf_memory
                when:
                        $MaaS := MaaS( memory_avg < andsf_memory_threshold_min )
                then:
                        insert Decision("scale-in", {'irp':1})
                        halt
```

2.2.2.3 Network (NIC) Utilization

This algorithm intends to scale-in or scale-out the ANDSF worker tier (IRP), based on the network (NIC – Network Interface Controller) utilization measurements. Based on two thresholds, maximum and minimum, the SO scales-out or -in the number of workers, respectively. The scale-out takes place when the bandwidth is above the maximum threshold, while the scale-in is performed when the bandwidth is below the minimum threshold.

Similar examples can be defined as in previous sections and similar rules policies can be built.

2.2.2.4 Storage Occupation

This algorithm intends to scale-in and out the ANDSF database tier (DB) based on the Storage occupation measurements. Based on two thresholds, maximum and minimum, the SO scales-out or -in the block disk associated to the DB VM in order to storage space.

2.2.2.5 Nº of Requests-based scaling

This algorithm intends to scale-in or scale-out the ANDSF worker tier (IRP), based on the number of requests received from the ANDSF Service measurements. Based on two thresholds, maximum and minimum, the SO scales-out or -in the number of workers, respectively.

Similar examples can be defined as in previous sections and similar rules policies can be built.

2.2.2.6 Other Service-based scaling

Similar as for the number of requests, many other ANDSF service specific metrics can be used to trigger scaling procedures. Examples are the total number of provisioned subscribers, the serving bytes (amount of content served), or the serving time (time elapsed to serve a particular request), among others.

Here we presented infrastructure and service metrics as triggers for scaling procedures. It is pertinent to ask why we should scale based on service parameters, since what really matters are resources. For example, in case the number of requests rises above a certain level, CPU, memory or other basic resources will suffer it and, at the end, it will be noticed at infrastructural level.

In fact this is true, but not always. In some cases, metrics related to SLAs like the serving time does not relate linearly to infrastructural metrics. For example, it is hard to predict at what CPU utilization level the serving time overtakes a given threshold (SLA). On the other hand, even if benchmarking work is performed to find the CPU utilization for a given threshold, this can be true for a given hardware (DC) but not in another (DC), what makes the exercise useless. For this reason, both metrics should always be considered.

2.3 Evaluation of state and signalling reduction for SDN-based EPC procedures

This section explains the state comparison in the user plane establishment (i.e. Attach procedure) and handover procedures in both 3GPP LTE/EPC and SDN-based EPC architectures. We have considered two different deployment scenarios of SDN-based EPC architecture: (i) two-tier scenario (ii) three-tier scenario. In the two-tier deployment scenario, all the control entities such as SDN Controller, new MME and P-GW-C are running in single location (i.e. the new MME and P-GW-C run as modules on top of the SDN In the three-tier deployment scenario, the control entities run in different locations, i.e. the new MME and P-GW-C run in one location and the SDN-Controller runs in another location (for instance, the new MME and P-GW-C run in a central office and the SDN-Controller runs in a regional office, as shown in Fig. 13).

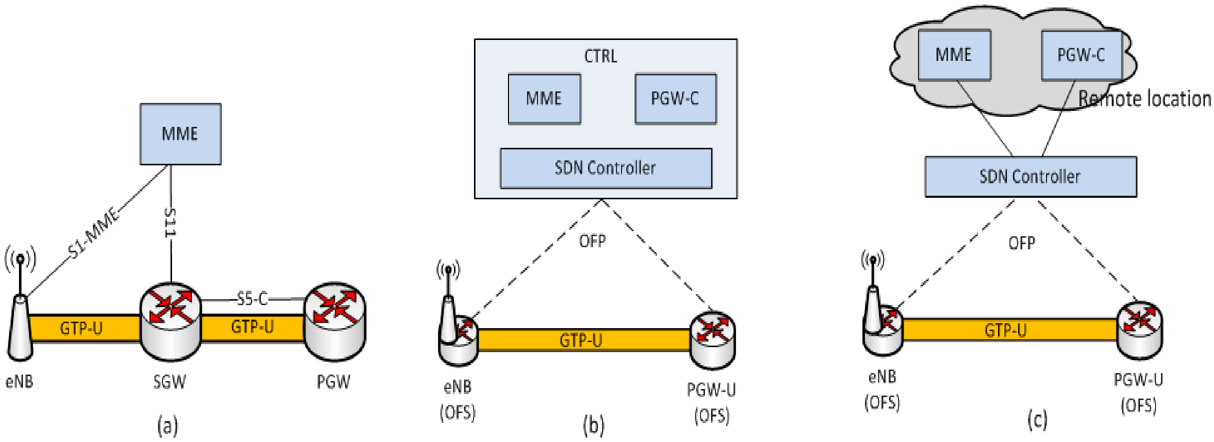


Figure 13 - (a) 3GPP LTE/EPC architecture. (b) Two-tier deployment SDN-based EPC architecture. (c) Three-tier deployment SDN-based EPC architecture.

In the existing 3GPP LTE/EPC, the Attach Procedure is used to establish the UE user plane bearer for a first-time connected UE. To establish the user plane, 10 control messages are exchanged between the UE and network entities (without counting the authentication messages). We are using an OpenEPC test-bed to calculate the control messages size including all the messages' information (i.e. identifiers in the message like IMSI, S1AP IDs.). Specifically, we are using the OpenEPC Release 3, which is a reference implementation of 3GPP's EPC Release 10 (see [OpenEPC]). We have calculated the size for each context information such as eNB UE S1AP ID, NAS PDU (includes IMSI, APN address), Tracking Area List (TAI), EUTRAN-CGI (E-UTRAN Cell Global Identifier) and RRC establishment cause. Thus, the total Attach Request message size (including MME IP and SCTP headers) is 144 Bytes. Similarly, we have calculated the size for the remaining 9 messages from the Attach Procedure, and the total size (including IP and SCTP/UDP headers) is 1128 Bytes.

In our proposed SDN-based EPC (in both 2-tier and 3-tier scenarios), one of the goal is to reduce (or to optimize) the context in each control message and also in each network entity. Significantly, the reduction of context in the message will reduce the message size, which improves the overall network performance (i.e. even if the number of control messages increases when compared to standard 3GPP EPC). Because in the proposed SDN-based EPC (in both 2-tier and 3-tier scenarios), the legacy SGW functions are merged in P-GW-C, the entire legacy SGW context and all messages exchanged between SGW and other network entities (MME, P-GW-C, eNB) are removed. On the other hand, in our proposal new context information is introduced (such as Data-Path ID between switches and controller and Buffer ID between the eNB and Controller instead of pair of UE S1-AP IDs).

2.3.1 Bearer establishment in three-tier scenario

As we explained earlier, in 3-tier deployment scenario, the control applications (SDN Controller, MME and P-GW-C) are deployed in different locations. In this scenario, the SDN Controller needs to transfer the UE request to the MME, which needs an extra overhead (i.e. IP and TCP header) when compared to standard 3GPP LTE/EPC. In addition, all control applications (SDN Controller, MME and P-GW-C) are maintaining the unique Application ID allocated by the operator during the configuration; these application IDs are included in each control message between the entities.

In 3-Tier scenario, we have taken into consideration all control messages needed to establish the user plane for a UE. Each message information is taken from the standard 3GPP and we removed the unnecessary context information (such as pair of UE S1AP IDs), which is not needed, in our proposal. In addition, we added the new context information that is required in our proposal (such as OpenFlow related information e.g. Buffer ID, Table ID, Reason, and Application ID used by the Controller on its northbound interface). For instance, the Attach Request message context information between eNB and new MME (via SDN-Controller) is the same like in the standard 3GPP message, but the S1AP header is replaced with OpenFlow header (i.e. NAS PDU, TAI, ECGI and RRC Cause are still required). Thus, the Attach Request message between eNB and new MME is 305 Bytes. In fact, the Attach Request message size is increased when compared with the standard 3GPP Attach Request; this is due to the extra overheads and also to the number of messages, two messages in our approach instead of one message in 3GPP needed between eNB and new MME (i.e. eNB to SDN-ctrl and SDN-ctrl-MM).

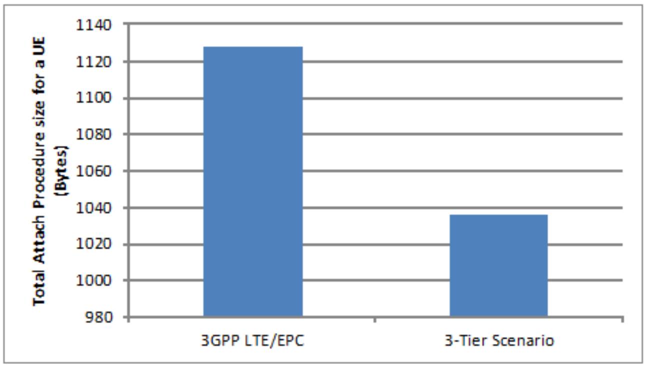


Figure 14 - 3-Tier deployment scenario Attach Procedure savings compared with 3GPP LTE/EPC

Therefore, in 3-tier scenario, a total of 8 messages are exchanged to establish the user plane while 10 messages are needed in the 3GPP case. In total, approximately 8% of Bytes are reduced in Attach Procedure with compared to 3GPP Attach Procedure as shown in Fig.14; this savings are due to the removal of SGW and its interfaces (S1, S5, and S11) and their exchanged information (such as F-TEIDs).

2.3.2 Bearer establishment in two-tier scenario

In two-tier deployment scenario, all the control entities such as SDN Controller, new MME and P-GW-C are running in a single location. Contrary to the 3-Tier scenario, the new MME and P-GW-C applications run on top of the SDN-Controller as applications and are connected through local interface (like Java interface, RPC, etc.), which reduces the extra overheads compared with the 3-Tier scenario. This solution falls in the N:2 category, for which the OpenEPC implementation is described in section 7.1 (Appendix 1).

The Attach Procedure in 2-Tier scenario is similar to the one from the 3-Tier scenario; thus, all the message contexts are not changed but in this scenario there are no extra overheads between the Controller and MME/P-GW-C (i.e. IP and TCP headers). All the control messages are process locally within the Controller that maintains local interfaces between MME and P-GW-C. In addition, the entities (i.e. Controller, MME and P-GW-C) are using a local database and they update the UE related information locally. In the 2-Tier scenario, to establish the UE user plane a total of 8 control messages are exchanged between the UE and the control entities. Therefore, due to local exchanges, the total savings in Attach Procedure are approximately 18% with compared to 3-Tier scenario and approximately 28% with 3GPP Attach Procedure as shown in Fig.15. These additional savings are due to the fact that we are maintaining the local interface and processing the control message within the Controller.

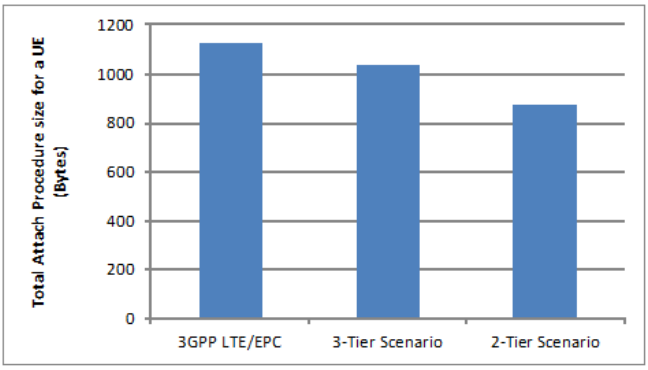


Figure 15 - 2-Tier deployment scenario Attach Procedure savings with compared 3-Tier deployment and 3GPP LTE/EPC

2.3.3 Handover in two-tier scenario

During the handover (intra-mobility) between eNBs, the established tunnels (uplink and downlink) need to be changed and to be re-established again as UE moves from one eNB to other. This is the case in the LTE/EPC network but also in all other tunnelling technology (like GRE, PMIP). However, in our proposed approach, the uplink tunnels are unique and will not be changed until the UE changes the P-GW. In addition, all control entities (i.e. SDN Ctrl, MME and P-GW-C) are placed in a single location (see Fig.13b); the intra mobility will be simpler because the SDN controller can more easily communicate with MME and P-GW-C. To maintain the unique tunnels, the P-GW-C application on top of controller allocates unique tunnels within the serving are of that P-GW-C.

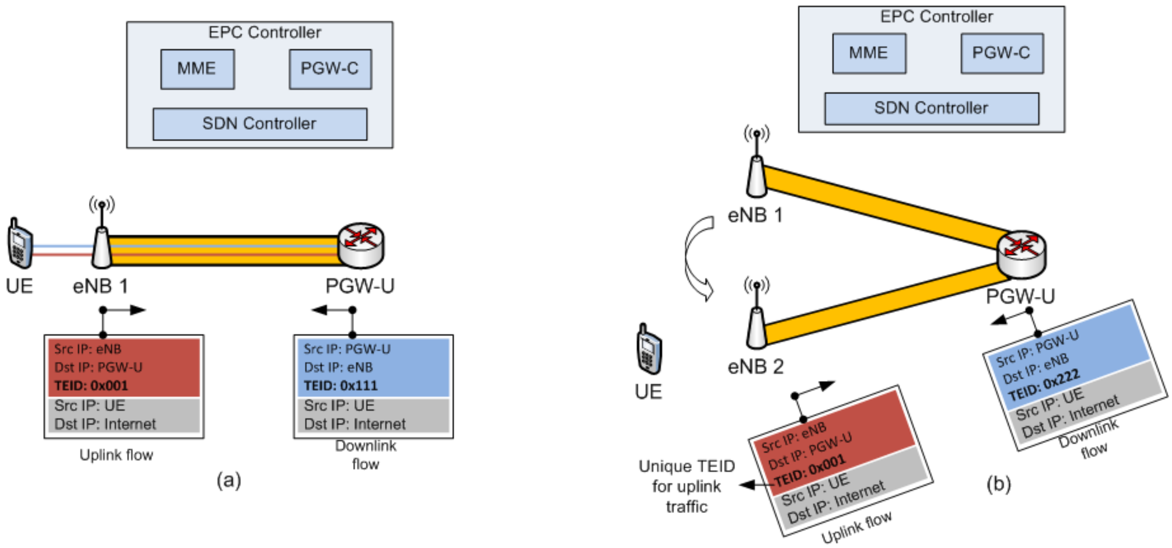


Figure 16 - (a) UE flow without mobility (b) Unique uplink tunnel during mobility.

For example, after successful initial attach process, the uplink and downlink GTP tunnels are established in the user plane (as shown in Fig. 16a) and these tunnel IDs are stored in UE table in the SDN controller database.

If a UE moves from eNB1 to eNB2, the uplink tunnels (TEID: 0x001) are not changed (see fig. 16b) and the downlink tunnels (TEID: 0x222) are modified in the P-GW-U as allocated by the new eNB2. By doing this, the network resources are re-used and also the user plane resources and establish time will be optimized.

On the other hand, in the control plane, when the UE moves from the eNB1 to eNB2, the eNB2 notifies the SDN controller about the UE movement. The SDN controller extracts the UE user plane tunnelling identifiers from the UE table in its database. As we said earlier, because the uplink tunnels are unique, the SDN controller uses the already allocated tunnels (during initial attachment) and it sends this tunnel identifier to the new eNB (i.e. eNB2). In addition, the controller also updates the flow entries in the P-GW-U with eNB2 IP and tunnel IDs. The exact calculations and the gain obtained (total size of the messages) will be given in a subsequent document.

2.3.4 Handover in three-tier scenario

The intra-mobility between eNBs in 3-tier scenario is also similar to 2-tier scenario, the only difference is the SDN controller and EPC control applications (MME and P-GW-C) are placed in different locations. Thus, when the UE handovers from one eNB to other eNB (eNB1 to eNB2), the SDN controller needs to exchange the messages with the EPC control applications. Significantly, these exchanges increases the latency compared to the 2-tier scenario. To avoid frequent communication between the MME and P-GW-C during mobility, the SDN Controller maintains the UE identifiers in the database allocated during the initial attachment. When a UE moves between eNBs, the controller look-ups the UE table in its database and extracts the identifiers such as TEIDs and sends the TEIDs to newly attached eNB. In addition, the controller updates the UE table with new eNB TEIDs and updates the new UE location in the MME. For instance, when a UE moves to eNB1 to eNB2 (shown in Fig. 17b), the eNB2 request the controller for the UE data plane routing rules. Because the UE is already attached to the network, the SDN controller extracts the UE identifiers from its database and it sends the uplink TEID and P-GW-U address to the eNB2. In addition, the controller also updates the

UE flow entry in the P-GW-U with newly attached eNB2 IP and TEID. After these updates, the controller updates also the UE location in the MME.

The exact calculations and the gain obtained (total size of the messages) will be given in a subsequent document.

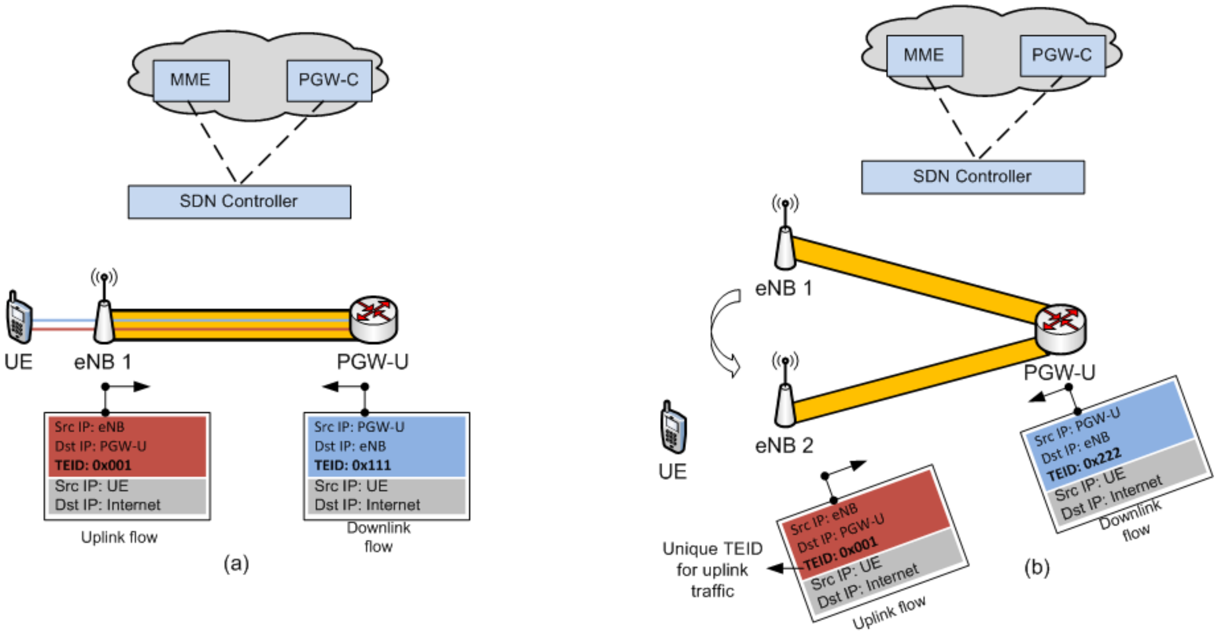


Figure 17 - (a) 3-teir scenario without the mobility (b) during mobility by maintaining uplink tunnels.

3 Prediction

Predicting mobile users' locations in a future moment of time is essential for a wide range of mobile applications, including location-based services, mobile access control, mobile multimedia QoS provision, as well as the resource management for mobile communication and storage. In the context of the MCN project, different MCN services might need the location prediction to optimize network performance. For example, ICNaaS could optimize the distribution of content according to the user mobility prediction results, such that the user content will be available/stored in the location he/she will visit in the future. In addition to user mobility prediction, estimation of used and available bandwidth will also bring benefits to mobile telecommunication applications. For example, service provider could "intelligently" allocate more bandwidth in a certain area at a future time, given the knowledge that the bandwidth demands are higher than in the normal case.

MOBaaS (Mobility and Bandwidth Availability Prediction as a Service) is a MCN support service that provides the prediction information regarding (1) the estimated location of an individual/a group of end-users in a future moment of time and (2) the bandwidth used/available at a certain location in a future moment of time. The generated information by MOBaaS can be used by any MCN service in order to generate triggers needed for self-adaptation procedures e.g., optimal run-time configuration, scale-up and scale-down of service instance components, optimal network function placement, or to optimize network resource management.

3.1 Mobility Prediction

In the mobile cellular networks, a mobility prediction mechanism mainly focuses on the movement of mobile user(s) relative to a particular area, such as a cell. There are various kinds of approaches in solving mobility prediction problems. Most often the proposed mechanisms are unique and have been developed to solve a specific or particular type of problem. Most of the available algorithms are based on information about the location, movement patterns, velocity, neural network-based prediction, etc. [MCN-D4.1]. One of the intuitive methods to determine future behaviour of a mobile user to predict his/her mobility pattern could be attempting to trace, measure or capture some regularity of the user's mobility. This scheme is based on movement history of the mobile nodes. In this deliverable, we describe our proposed mobility prediction algorithm, which utilizes user's movement history, his/her current location (cell) and time (day/hour/minute) to estimate location (cell) of the user in a future moment of time. In the future steps, output of this algorithm can be integrated with other user related information, such as his/her device signal strength, to perform more precise estimation. The output of mobility prediction algorithm (so far) could be used by ICNaaS and EPCaaS. As shown in Figure 18 ICNaaS utilizes user mobility prediction information to decide on content relocation. Based on the network topology design and implementation, content may be relocated on the CCNx routers/repositories, which are placed on the eNodeBs and/or the S/P-GWs. When a user moves from one cell (source eNodeB) to the other cell (target eNodeB), which is served by the same S/P-GWs, the content may be relocated between the CCNx routers/repositories implemented in the eNodeBs. If the source and target eNodeB are not served by the same S/P-GWs, the content may be moved between the CCNx routers/repositories, which are placed in the target S/P-GW.

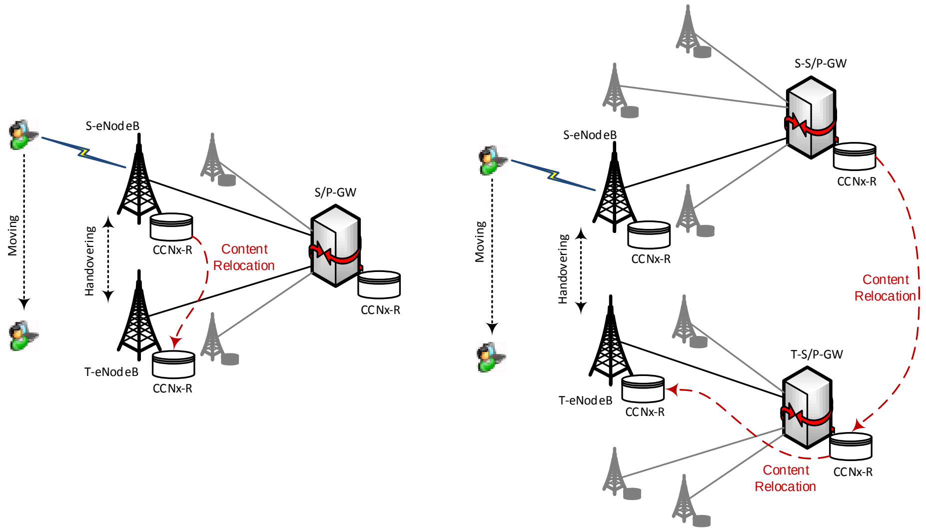


Figure 18 - Decision on content relocation in ICNaaS based on MOBaaS output

In the EPCaaS, user mobility prediction information could be used to optimize placement of the EPC components in the cloud, as well as to dynamically adapt those components while taking into account multiple factors such number of users, traffic loads, etc. The user mobility prediction information is a key element in DMM as well.

As it's shown in Figure 19, DMM controller can use this information to redirect the user's traffic when he/she changes its S/P-GW.

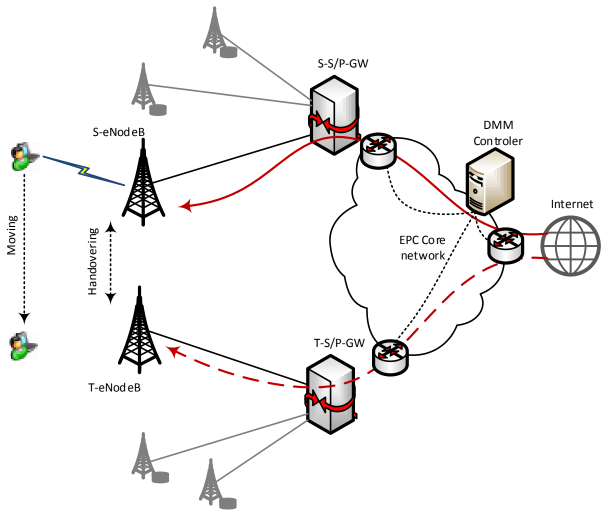


Figure 19 - Traffic redirection in EPCaaS based on MOBaaS output

3.1.1 Mobility Prediction algorithm

The trajectory of a mobile user and the locations that he/she visits can reflect the tastes, lifestyle and shape of user's social relationships. Many studies which have carried out on the mobile user behaviour, show that most often a mobile user has tendency to regularly behave the same way. The traced trajectories of the mobile users show a high degree of temporal and spatial regularity. For instance most of the people catch recurrent behaviours, such as going from home to the office and leaving it every days of a week at the same hours and more likely using the same path to return. Such regularity in the mobile user's behaviour could be considered as its profile and utilized to perform the estimation of the places the user will visit in the future.

This section presents a mobility prediction approach, which utilizes the current location of a mobile user, his/her time (day/hour/minute) and his/her trajectories information traced in the past, to predict the next location (cell) that may be visited by the user. The spatial granularity of this algorithm is at the cell level, which means the prediction algorithm outputs the possible future cells of users. The temporal granularity of this algorithm is dependent on the application requirements. For example, the consumers of MOBaaS, such as ICNaaS or EPCaaS, may request MOBaaS to provide the user future locations in next 1-minute, 10 minutes, half an hour, 1 hour, or etc.

The Mobility Management Entity (MME) is one of the key components in the Evolved Pack Core (EPC) for LTE, which is responsible for initiating paging and authentication of the mobile device. It also keeps location information at a Tracking Area level for each user, and it is involved in choosing the right gateway during the initial registration process. It provides mobility session management and supports subscriber authentication, roaming and handovers to other networks as well. By monitoring and tracing the MME's information, it would be possible to drive the user's movement historical information and accordingly its trajectories.

The parameters and variables used in the proposed mobility prediction are as follows;

- $u_i \in U$: u_i ,represents a user with ID *i*
- $c_j \in C$:c_j ,represents a cell with ID *j*
- *S :* It represents the time step at which the user's location(cell) has been traced by the MME *(for instance; S=1, means that the location has been traced in each minute. This value depends on the granularity of input data from MaaS.)*
- T_n , $n=\{0,..,N\}$: *n* represents the time step of a day and T_n represents the real time in a day *(for instance; if S=1 and n=60 in this case* T_n *=01:00 AM, and if S=5 and n=60 in this case* T_n *=05:00 AM)*

$$N = \frac{24 * 60}{S}$$

- D_l , $l=\{1,..,7\}$: d_l , represents the day of week *(for instance; l=1, means the day is Saturday and l=7 the day is Friday)*

We use the Dynamical Bayesian Network (DBN) [Etter13], to model the mobility prediction algorithm The rationale behind using DBN is that, the next visited cell by a user depends on: (1) It current location (cell), (2) the movement time, and (3) the day that user is in the movement. The proposed DBN model is illustrated in Figure 20.

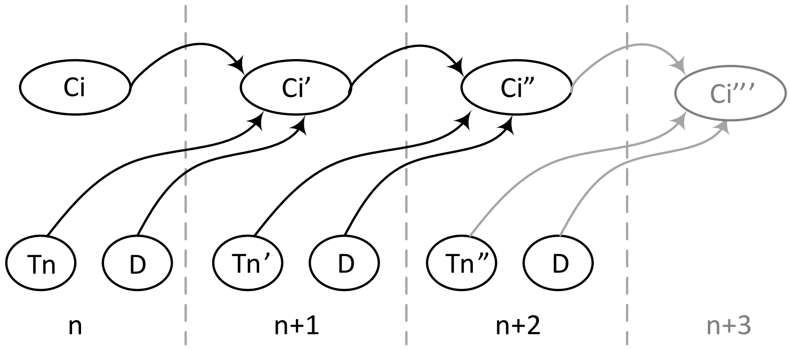


Figure 20 - DBN to model the mobility prediction algorithm

As it is shown in Figure 20, the conditional distribution of the next visited location (cell) by a user comprises the location and time dependent distributions.

$$P(\, c(n+1) = c_{i'} | c(n) = c_i \,, T(n) = T_n \,, D = D_l \,)$$

We can separate the time and location dependency as follows;

$$\begin{aligned} P(\, c(n+1) = c_{i'} | c(n) = c_i \,, T(n) = T_n, D = D_l \,) \\ = \beta \times P(\, c(n+1) = c_{i'} | c(n) = c_i \,) + (1-\beta) \\ \times P(\, c(n+1) = c_{i'} | T(n) = T_n \,, D = D_l \,) \end{aligned}$$

$$\beta \in [0,1]$$

The first part of this equation represents the location dependency, which is simply a first order Markov Chain (M.C.) that encodes the frequency (probability) of transitions between locations (cells). In order to drive the Markov Chain transition matrix, we calculate the probability of moving from $(c(n) = c_i)\ to\ (c(n+1) = c_{i'})$, for each individual user, by counting the number of transitions from one cell to another in the traced files. This calculation has been done separately for workdays and weekends, because the user’s behaviours and its moving profiles may be different in these parts of a week. By performing this, in the transition matrix, we may encounter some situations in which the probability of moving from one cell to its two neighbours cells have the same amount, as shown on the Figure 21.

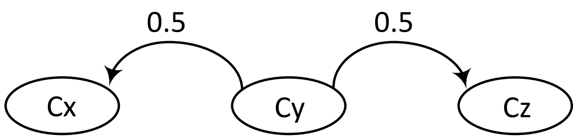


Figure 21 - The same probability of transition from one cell to two its neighbour cells

This situation could be interpreted in such way that, the cell C_y placed in the route of a user, which he/she pass it from ($C_x \rightarrow C_y \rightarrow C_z$) in the direction from home to work and cross it from ($Cz \rightarrow C_y \rightarrow C_x$) when he/she is in the return path. To solve this uncertainty such that the algorithm is able to select one of these two cells (C_x *or* C_z), we integrate the time step (the time that a user visits or is in a cell) and the cell ID to drive the Markov Chain states. In this case each state comprises the location (cell ID) and the time (or time step) in which user is in it, see Figure 22.

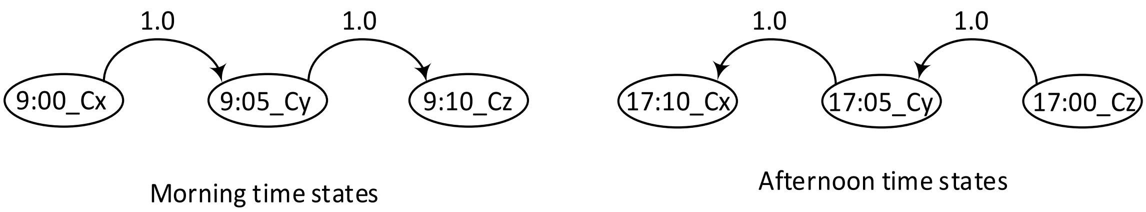


Figure 22 - Integration of the time step and the cell ID to drive the Markov chain states.

As it is shown in Figure 23, by utilizing the M.C. states in this way, we can drive the probability transition matrix which somehow also comprises (describes) the direction of the user based in his/her movement time in a day. Creating the M.C. states in this way may increase number of states. This issue could be solved by grouping the states that have the same cell IDs, which means that during some period of time the user is still in one cell and its coverage cell in not changed. This situation is depicted in Figure 23.

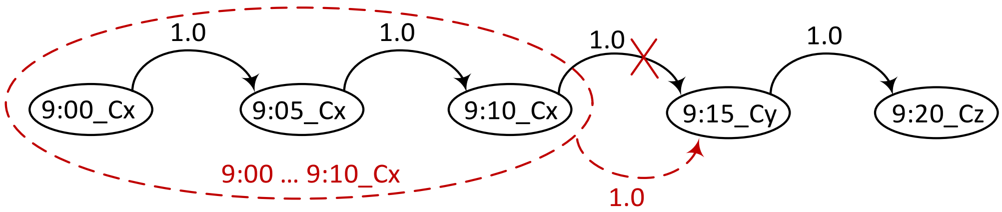


Figure 23 - Grouping the states with the same cell ID to lessen number of states

By integrating the time step and the cell ID, the DBN model for mobility prediction algorithm could be modified as follows.

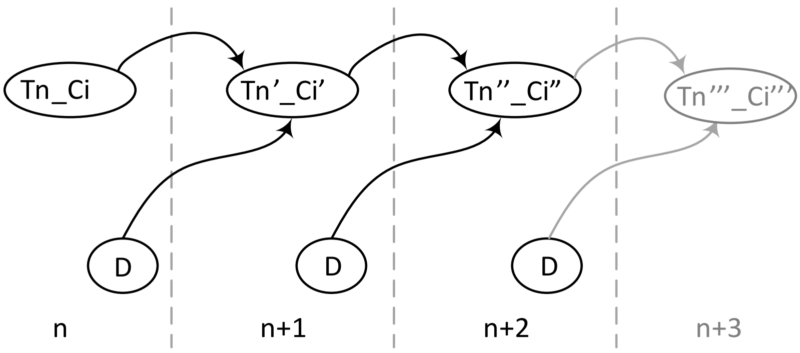


Figure 24 - Modifying the DBN for mobility prediction algorithm

$$P(\,c(n+1) = c_{i'} | c(n) = c_i\,, T(n) = T_n\,, D = D_l) \\ = \beta \times P\big(\,c(n+1) = c_{i'}\,\big|\,c(n) = c_i\,, T(n) = T_n\big) + (1-\beta) \\ \times P(\,c(n+1) = c_{i'} |\, D = D_l\,)$$

$$\beta \in [0,1]$$

In this case the second part of equation represents the day dependency, which expresses the probability of transition to the next locations (cells) in a given day. Using the available traced files for each user, this probability could be driven by averaging the number of transition to a specific cell in each day of week. Similar to first part (M.C.), the probability calculation has been done separately for workdays and weekends, because of the different user's behaviours and its moving profiles in these parts of a week.

For each user, we separate the available traced data into two parts, the learning data set (A) and testing data set (B), as illustrated in Figure 25.

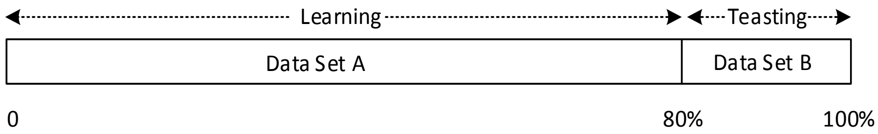


Figure 25 - Dividing the available traced data to learning dataset and testing data set

Data Set A, is the 80% of user's traced data and utilized in the learning phase. This data set is used to drive the M.C. states and to calculate its probability transition matrices ($P\big(\, c(n+1) = c_{i'} \,\big|\, c(n) = c_i \,, T(n) = T_n \big)$) . It also used to calculate the probability of transition to the next locations (cells) in each specific day ($P(\, c(n+1) = c_{i'} |\; D = D_l \,)$). Data set B is the rest 20% of traced data, which is used to test and evaluate the proposed algorithm. In the above mentioned equation, $\beta \in [0,1]$ is a parameter that governs the contribution of each distribution. Data set A is utilized to drive the appropriate β and define the portion of each distribution as well. The value of β is selected in such a way to maximize the $P(\, c(n+1) = c_{i'} | c(n) = c_i \,, T(n) = T_n \,, D = D_l)$ or minimize the prediction error.

3.1.2 User Movement Traces

To do the prediction, MOBaaS needs significant amount of historical user movement information, which should be provided by another MCN supporting service such as Monitoring as a Service (MaaS). These input data will be used as the training samples by MOBaaS prediction algorithms to learn the user movement patterns and find out the regularities of user mobility. When MOBaaS is connected with MaaS, all the necessary information used for prediction should be provided by MaaS on-demand. However, in this deliverable, due to the limitations of the current MaaS implementation and the missing of the implementation of the interface connecting MaaS and MOBaaS, we cannot have any data (user movement trace data monitored from MME or other EPC components) from MaaS as input to the prediction algorithms. Therefore we use the pre-collected trace files as the artificial inputs to our prediction algorithms.

The performance of MOBaaS in the mobility prediction algorithms depends on the quality of the mobility traces used for training the system. To this end we choose the data provided by Nokia (https://www.idiap.ch/dataset/mdc) for academic research. The data collection was performed under the Mobile Data Challenge (MDC), which is a larger-scale research initiative aimed at generating innovations around smartphone-based research, as well as community-based evaluation of related mobile data analysis methodologies [Laurila12]. The data collected a wide range of contextual data, including GPS and Cell IDs, from approximately 200 users over 2 years.

The data is granted upon signing legal agreements, mainly aiming at protecting the privacy of the participants, and illegal copying. We extracted the mobility traces and trained/run our mobility prediction algorithms taking these aspects into consideration.

Since the mobility prediction algorithm is based on the historical information of user locations, GPS location data is an essential input to the prediction algorithm. Meanwhile, a certain user is always within the coverage range of one or multiple eNodeB. Therefore, to provide the cell-level location prediction, it's not necessary to have the exact GPS locations of users, since multiple users might be covered by one eNodeB. Instead, the knowledge of which cell is currently covering the user and providing the connectivity is enough. Based on this, we use cell ID of the cell, which is currently covering the user to represent the location of the user. In this way, the mobility prediction algorithm takes the following inputs:

<UserID, Timestamp, CellID >

This indicates that at a certain time, a certain user is covered by a certain eNodeB (cell). We provide this information by extracting the necessary items from the original trace files.

3.1.3 Preliminary result of simulation

Figure 26, represents the format of data that the mobility prediction algorithm gets in input in the request message (sent by MCN consumers) and provides in output in the answer message.

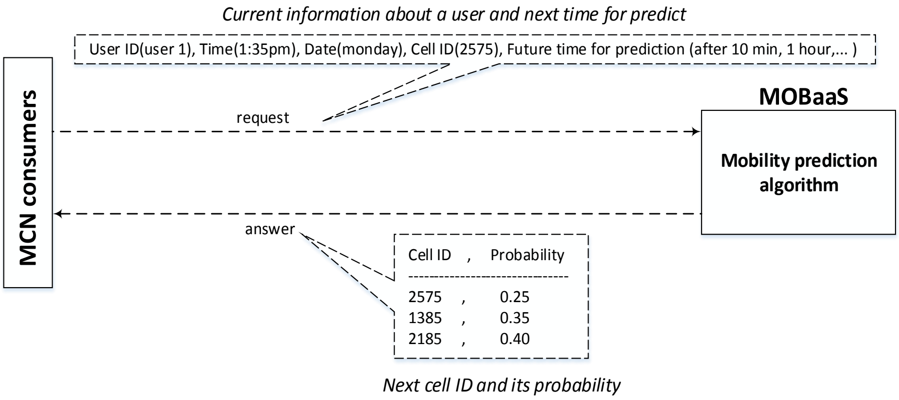


Figure 26 - Mobility prediction algorithm input/output data format

In the request message:

- *User ID :* represents the ID of a user *(Integrated Circuit Card ID(ICCID) could be used as the user's ID)*
- *Time :* shows the current time that user is served by an specific Cell
- *Date :* represent the date of week
- *Cell ID :* indicates the ID of a cell which currently serves the user
- *Future time for prediction ;* indicates the time on which a MCN consumers desires to know the location of the user

In the answer message:

- *Cell ID :* represents the ID of a cell which may serve the user at a future time *(indicated by the "Future time for prediction" in the request message)*
- *Probability ;* indicates the amount of probability that a specific cell may serve a use at a future time

In the current version of mobility prediction, output in the answer message could be provided in a text file or in a graph, as shown in Figure 27. It depicts the next location (showing by the Cell ID) that a user may pass it and its corresponding probability. The result in this deliverable is only based on the first part of proposed model $(\mathrm{P}\big(\, \mathrm{c}(\mathrm{n}+1) = \mathrm{c}_{\mathrm{i}'} \,\big|\, \mathrm{c}(\mathrm{n}) = \mathrm{c}_{\mathrm{i}}\, , \mathrm{T}(\mathrm{n}) = \mathrm{T}_{\mathrm{n}}\big))$. The results based on the whole model, and its performance evaluation will be presented on future deliverables. The calculation to estimate for the number of users that may pass from a specific cell will be provided in the next deliverables as well.

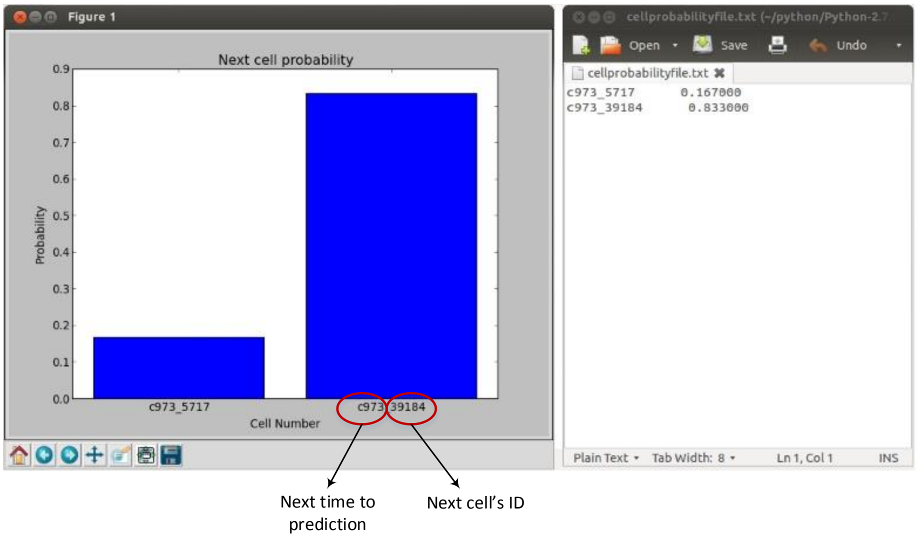


Figure 27 - Output of Mobility prediction algorithm (If S=1, the '973' represents that time is 16:13)

3.2 Bandwidth Prediction

Within the MCN project, allocation and dimensioning of bandwidth is of great importance. As in any dynamic cloud infrastructure, requirements for bandwidth are subject to change due to (re) allocation of resources, e.g. new components that are added to increase computing capacity. But in the context of MCN, specifically in MOBaaS, we are interested in handling the challenge of dynamic bandwidth requirements caused by end users. With mobile devices (e.g. smartphones, tablets) being able to replace tasks that required full-fledged computers in the past, these mobile devices need the connectivity and bandwidth that are readily available on conventional wired networks. This great diversity in mobile application uses results in demanding and dynamic bandwidth usages. The ability to offer a reliable service in this aspect means one should be able to predict the bandwidth use based on user behaviour (in terms of capacity requirements). With that information, cloud services can be tailored according to bandwidth availability, preventing services from allocating network resources while none are available and consequently impairing end-user experiences.

3.2.1 Bandwidth algorithm

Performing bandwidth prediction is based on validated algorithms from [Rica12] and on the works described in [Hans06] and [Aiko09] which use the classic IP level 5-tuple as input data. This 5-tuple is consisting of:

<*source ip; source port; destination ip; destination port; protocol*>

This can be obtained in several ways. The algorithms that are at the base of our bandwidth prediction approach use the information obtained from NetFlow data, but research showed that OpenFlow is able to supply the same information. This enables us to do bandwidth prediction on every part of the network that is forwarding packets based on OpenFlow, i.e. every routing node in the MCN structure. The algorithm needs a brief historical input in order to compute valid results. This means that an initial input of 15 minutes of measurement data is required. After the first 15 minutes, the algorithm updates its estimation every minute with fresh data. It is assumed that the user will maintain similar behaviour in terms of bandwidth usage in the next minutes, so the estimation provided by the algorithm can be used as a prediction value within the MCN context.

Obtaining the information from OpenFlow switches can be operated in both a passive and an active mode. The passive approach is based on per-flow statistics in *OFPT_FLOW_REMOVED* messages, which are exported to the OpenFlow controller when a flow entry has passed either its *soft* or *hard timeout.*

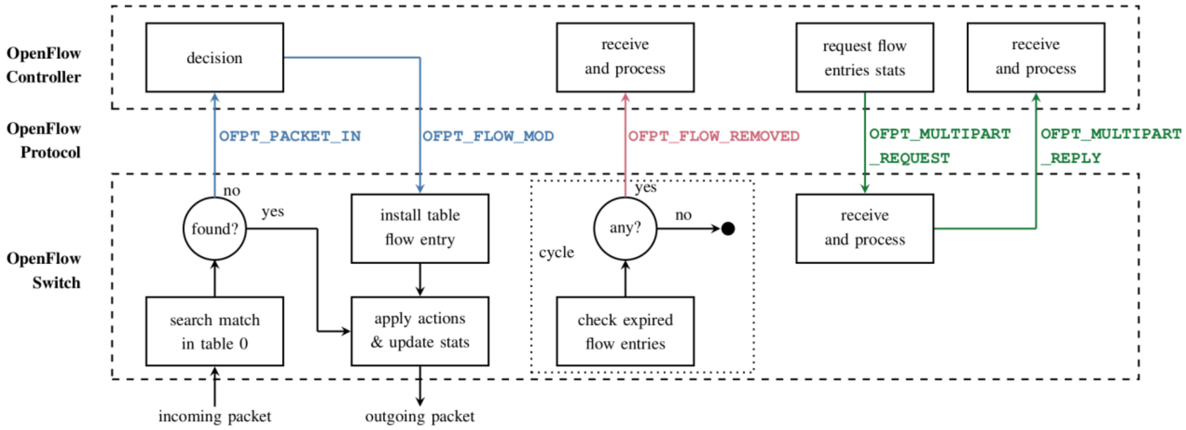


Figure 28 - Exchanging the messages between the OpenFlow switch and OpenFlow controller in the bandwidth estimation procedure

Figure 28, visualizes the exchange of messages between the OpenFlow switch and OpenFlow controller necessary to obtain the information required for the bandwidth estimation. The blue cycle shows the start of the metering, where a new flow entry is installed in the flow table based on a *PACKET_IN* event, followed by a *FLOW_MOD* installing the new flow entry in the flow table. Once the flow has expired, as shown in the red cycle, the flow information is exported by means of the *FLOW_REMOVED* message. This is the passive part. Finally the green cycle depicts the active approach. The active approach does not rely on flow expiration, but is based on *OFPMP_FLOW* requests (as specific type of OFPT_MULTIPART_REQUEST) querying specific flow tables within the OpenFlow switch. Resulting *OFPT_MULTIPART_REPLY* messages can be interpreted similarly to *OFPT_FLOW_REMOVED* messages in the passive approach.

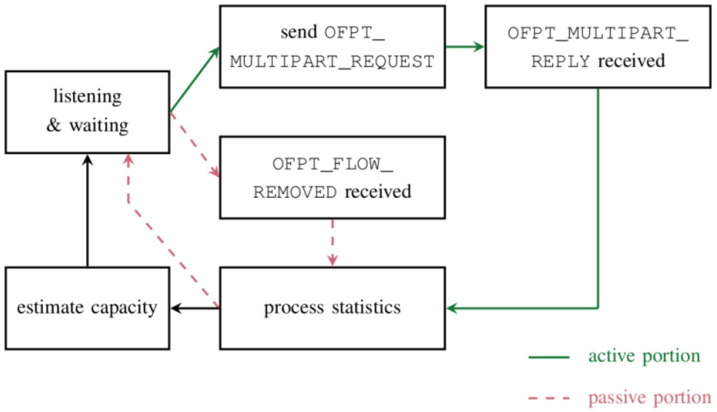


Figure 29 - The combination of the passive and active approaches in the bandwidth estimation procedure

Figure 29, shows a combination of the passive and active approaches from the perspective of the OpenFlow controller, i.e. the place where all the required information to perform bandwidth estimation is gathered. Analogue to Figure 28, the red cycle depicts the passive portion of the approach, waiting for *FLOW_REMOVED* messages containing the information on expired flows. The green cycle shows the active approach based on actively sending out requests for information. By requesting the entire flow table, flow information about all the active flow entries is sent from the OpenFlow switch to the OpenFlow controller, so the application running on the controller can use it in a way similar to the flow information returned in the FLOW_REMOVED messages.

In the MCN infrastructure, the information described above should be provided by MaaS. Using that as input, the bandwidth prediction part of MOBaaS is able to supply the MCN consumers with results describing capacity on links within the cloud infrastructure. The output of bandwidth prediction expresses the available capacity on a certain network link. This means that an a priori knowledge is required in addition to the aforementioned algorithm, namely the total link capacity of the link for which the traffic is dimensioned. Subtracting the bandwidth derived by the algorithm from the total capacity yields the available capacity. Finally, this information could be integrated with the output of the mobility prediction algorithm, estimating the total number of the users in a specific location and time, to predict the available bandwidth in a certain link in the MCN architecture.

The algorithm is based on the following probability of inequality (Eq 3.2.1) by which satisfaction should result in a certain capacity *C*, expressing the required bandwidth.

$$\mathbb{P}\, A\,(T)/T \geq C \leq \varepsilon \qquad \text{Eq 3.2.1}$$

- *A(T)*: total amount of traffic arriving in intervals of length *T*
- *T*: timescale
- *C*: constant denoting a certain traffic rate
- *ε*: probability that the traffic rate of C is exceeded by *A(T)/T*

Assumptions for this inequality are:

- *A(T)* is normally distributed
- *A(T)* is stationary

Based on the inequality and the assumptions listed above, for a certain time interval T and 'error rate' ε, we get

$$C(T,\varepsilon) = \rho + \frac{1}{T}\sqrt{-2\log(\varepsilon)\cdot \upsilon(T)} \qquad \text{Eq 3.2.2}$$

where:

- ρ: mean traffic rate
- $\upsilon(T)$: variance of $A(T)$

Added to the mean traffic rate (ρ) is a margin based on the variance of the amount of traffic that arrived in a certain time-interval T, i.e. $A(T)$.

The scalability of using flow-based information as input to the algorithm comes at the cost of information loss, caused by aggregation. Specifically, packet inter-arrival times are lost in the metering/aggregation process of a flow-setup, while the mathematical expressions above require (the variance) of $A(T)$. In order to overcome this loss, an estimation of $A(T)$ and $\upsilon(T)$ is used based on the assumption that bytes are distributed uniformly over the duration of a flow.

For a certain time interval t_i

$$t_i = [iT, (i+1)T] \qquad \text{Eq 3.2.3}$$

the set of flow records F that accounts for the number of bytes in that time interval can be expressed as

$$K_i = \{f_j \in F: s_j < (i+1)T \wedge e_j \geq iT\} \qquad \text{Eq 3.2.4}$$

where

- f_i: flow from set of flow records F
- s_i: start time of the flow
- e_i: end time of the flow

For all the flow records in set K_i, an estimation a_i of the total number of transferred bytes can be expressed as

$$a_i = \frac{\sum_{f\,j \in K_i} max(min((i+1)T, e_j) - max(iT, s_j), 1)}{max(e_j - s_j, 1)} \cdot b_j \qquad \text{Eq 3.2.5}$$

where

- b_j: number of bytes transferred in the flow

Finally, the variance *υ(T)* can be computed using

$$v_a = \frac{1}{m-1}\sum_{i=1}^{m}(a_i - \rho T)^2 \qquad \text{Eq 3.2.6}$$

where

- *ρ:* observed traffic rate
- *m:* number of measured intervals

3.2.2 Performance evaluations

As shown in [Rica12], the algorithm is able to yield accurate results on several timescales. Evaluation within MCN context can only be performed based on historical traces as the current implementation of MaaS is not yet able to provide the necessary input data.

Performance of bandwidth estimation is directly influenced by, as expressed in the second mathematical expression in this section, the chosen time interval and the error rate. As the algorithm is already validated, evaluation of the approach should result in the time interval and error rate suitable for the MCN infrastructure. Assumptions are that the 1-second time interval yields a performance that is acceptable and deemed appropriate, although a greater time interval might still be sufficient while less resource demanding. The right error rate can be considered more subjective and SLA-dependent, though the 'amount of errors', i.e. the amount of time the bandwidth estimated is insufficient for the actual bandwidth demand, is a comprehensive metric in terms of performance evaluation. For the transition from the theoretical to the practical domain, the observed error rate can be used to adjust the time interval (directly affecting resource requirements) for proper functioning in different production environments.

Based on the input traces, or in the future, on input from MaaS, the performance will be evaluated by measuring the metrics indicated in the following table:

Table 8 – Performance evaluation table for the BW prediction in the next step

Input	Ground truth capacity requirement	Estimated capacity requirement	Percentage of time with wrong estimations
Trace 1			
Trace 2			
Trace n			

The capacity requirement and estimation will be expressed in typical comprehensive graphs (time series) of required bandwidth, while the share of wrongly estimated capacities will be calculated from the raw data/results and expressed in a percentage.

4 Distributed Mobility Management

Recently, Telecommunication networks (e.g, 3G and 4G cellular networks) have increasingly become the major access method to the Internet and data services. Accordingly, many networks currently experience a rapid growth in the number of mobile subscribers and wireless data traffic. Over the last few years, wireless operators' networks have rapidly turned into full IP-based networks for both voice and data, thus stimulating an underlying IP mobility support. Mobility management refers to a set of mechanisms to keep ongoing-sessions continuity while a mobile user changes his/her physical channel, access network or communication protocol. Real-time IP multimedia applications such as Video Conferencing (VC), Voice over IP (VoIP), Game net, download/upload of large size files (particularly in cloud computing environment) are examples of such notably demanded applications in mobile network environments. In these network supporting IP mobility and seamless session continuity is a necessity for the users changing their mobility anchor points during inter-operator and intra-operator/technology handovers.

Various mobility management mechanisms may employ different layers of the OSI protocol stack to handle their functionalities [Akyi04]. In the physical layer, mobility management carries out the "detach" and "attach" operations to different access points during handover. In the network layer, mobility support means to deal with the change in the sub-network. Mobility support in this layer may be based on routing (used e.g., in Cellular IP [Valk99]) or mapping (used e.g., in Mobile IP (MIP) [John04] and Proxy Mobile IP (PMIP) [Gund08]). In the transport layer, mobility management focuses on keeping the on-going TCP connection, though IP address is changing (used e.g., in Mobile Stream Control Transmission Protocol (M-SCTP) [Rieg07]). In the application layer, a particular design is considered to tackle mobility issues for each application type (used e.g., in the Session Initiation Protocol (SIP) [Rose02]), or a middleware may be implemented between applications of two nodes to manage mobility, such as WiSwitch [Gior05]. The network layer based scheme is the most popular one offering transparent mobility support to all kinds of applications. MIP [John04], PMIP [Gund08], and 3GPP mobility management [3GPP-Release-8], are examples of such a scheme. Most of these solutions rely on a centralized mobility management entity which is in charge of both control and data planes [John04], [Gund08],[3GPP-Release-8]. Centralized mobility management inclines to several restrictions such as centralized management of one/several hierarchical tunnels for each Mobile Node (MN), data processing overhead to perform encapsulation/de-capsulation functions during tunnelling updates, network bottleneck, single point of failure and non-optimal routing (particularly when MN and correspondent node (CN) are close to each other but are both far from the mobility anchor point) [Bert09],[Chan11],[Boko11]. Centrally managed IP mobility in the current mobile Internet is not scalable enough to efficiently deal with demands raised by ever-growing number of mobile users of new generation of applications seeking for IP mobility. Over the last few years, operators and research communities are investigating alternative mobility solutions that exploit a flatter system in which the mobility anchors are distributed and located at the edge of the access network. In this approach, denoted as Distributed Mobility Management (DMM), only the data plane (partially DMM) or both the data and control planes (fully DMM) may be handled by the distributed anchor points. In this regard as discussed in [KariCLOSER14] Software Defined Networking (SDN)/OpenFlow approach outperforms existing solutions. SDN [ONF13] has emerged as a new paradigm offering a logically centralized control model which detaches control and data planes, thus enabling direct programming of the control plane and abstracting the underlying infrastructure for applications and services. SDN makes networks programmable, manageable and adaptable to a wider extent, which is ideally suited for highly scalable mobile wireless networks. OpenFlow [OpenFlow 1.3.0], as the most common communication protocol used in SDN approaches, can significantly facilitate traffic management by accessing the control plane and the data plane of switches and routers over the network (e.g., the Internet architecture). Capabilities offered by OpenFlow would be as an enabler to improve IP mobility management, such that each traffic path could be traced from an Internet Ingress node (e.g., Internet PoPs) to an Egress node (e.g., router at the edge of access network) as a separate flow, and traffic could be redirected to a new mobility anchor point without any IP address translation or modification. Consequently, it eliminates the need for IP and GPRS Tunnelling Protocol (GTP) tunnelling respectively in wireless and cellular networks demanded for mobility management and diminishes data processing and traffic overhead in a large scale as a result of optimum encapsulations/de-capsulation, handover and location signalling. Hereby it brings more scalability with the increasing number of MNs. The OpenFlow-enabled switches and routers comprise a set of actions that give the possibility to modify the transiting packet headers belonging to a specific flow. The ability of dynamic configuration of flow tables in switches and routers via OpenFlow Controller (OC), as well, makes the OpenFlow-based SDN architecture as a promising approach which can notably enhance mobility management in terms of scalability and network resource utilization in the virtualized LTE system [KariAIMS2014].

4.1 DMM Functional Framework and Architecture

In the context of the IETF DMM working group, [Mark14] defines a functional framework for DMM and introduces a set of Functional Entities (FEs) which are required to support IP address continuity in a network with distributed mobility anchors. IP address continuity is a property of a network to seamlessly hand over an ongoing session from a source mobility anchor to a destination one during the lifetime of the session. In this deliverable the DMM functional framework specified in [Mark14], is used and applied for cloud based LTE systems as follows.

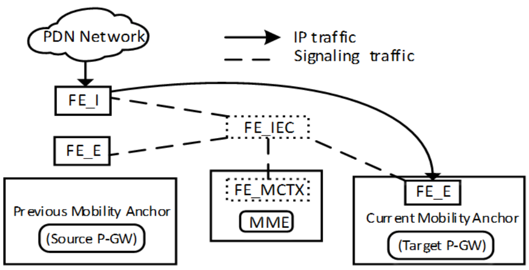


Figure 30 - The DMM Architecture

FE_MCTX (Functional Entity Mobility Context Transfer) provides mobility context information to the control function (FE_IEC) to allow forwarding of packets to the UE's currently used mobility anchor. This function could be co-located with the MME entity which provides mobility context information to other EPC elements (e.g., P-GW). FE_I (Functional Entity Ingress) establishes the forwarding of the UE's data packets to the appropriate Egress function (FE_E). FE_E (Functional Entity Egress) is a function that receives downlink packets forwarded by the DMM Ingress (FE_I) function and terminates the data forwarding plane. FE_IEC (Functional Entity for Ingress/Egress Control) is a function to establish, update and remove policies in the DMM Ingress (FE_I) and Egress (FE_E) functions to allow forwarding of UE's data packets towards the currently used mobility anchor (target P-GW). UE's mobility context information is delivered to this function by the FE_MCTX function (MME) upon triggering of the specific handover procedure.

4.2 Proposed DMM Architecture in a virtualized LTE system

Figure 30, depicts the proposed DMM architecture. The FE_MCTX function is co-located with the MME, resulting in the need of a direct signaling path between the MME and the entity where the Ingress/Egress Control (FE_IEC) functions is located. Alternatively the FE_IEC function can also be co-located with the MME, which then will need to be extended to support signaling towards the entities that implement the Ingress and Egress functionalities. The two Egress functions FE_E have been placed in different positions. This is done in order to show the possibilities of co-locating the FE_E with the P-GW, or of locating the FE_E close to the P-GW (e.g., one hop away). When the serving FE_E is located on an entity placed closer to the mobility anchor point (e.g., next hop for the source P-GW), specific forwarding techniques has to be used to deliver the traffic to the mobility anchor point. After completing a handover procedure with mobility anchor relocation, the destination IP address (or prefix in case of IPv6) of the downlink packets is topologically incorrect. Therefore, the DMM Ingress function needs to redirect the downlink traffic towards the UE that kept its previous IP address active upon movement. Downlink traffic forwarding by the DMM Ingress function can, for example, be accomplished by an IP tunnel to the Egress function, address translation to a routable IP address or other means. In order to avoid encapsulations/de-capsulation processing overhead during the commonly deployed IP tunnelling (and tunnel updating) to deliver the UE's downlink traffic to the currently used mobility anchor, alternative forwarding techniques are introduced. The source address carried by the uplink packets is also topologically incorrect. For the uplink packets to be assumed as routable, IP routers of the mobility domain must not apply filtering according to the source addresses. Since IP address continuity is a problem concerning downlink traffic, this report focuses mainly on this particular case. In the current 3GPP's LTE standard, IP address continuity for users that change their EPC mobility anchor point is not supported. This means that an ongoing flow cannot survive the change of the user mobility anchor in the current LTE network system. Certain modifications are therefore required in the protocol, particularly on both X2 and S1 handover procedures. Figure 31, shows the DMM solution's signalling procedures over X2-handover. More detail can be found in [MCN-D4.2].

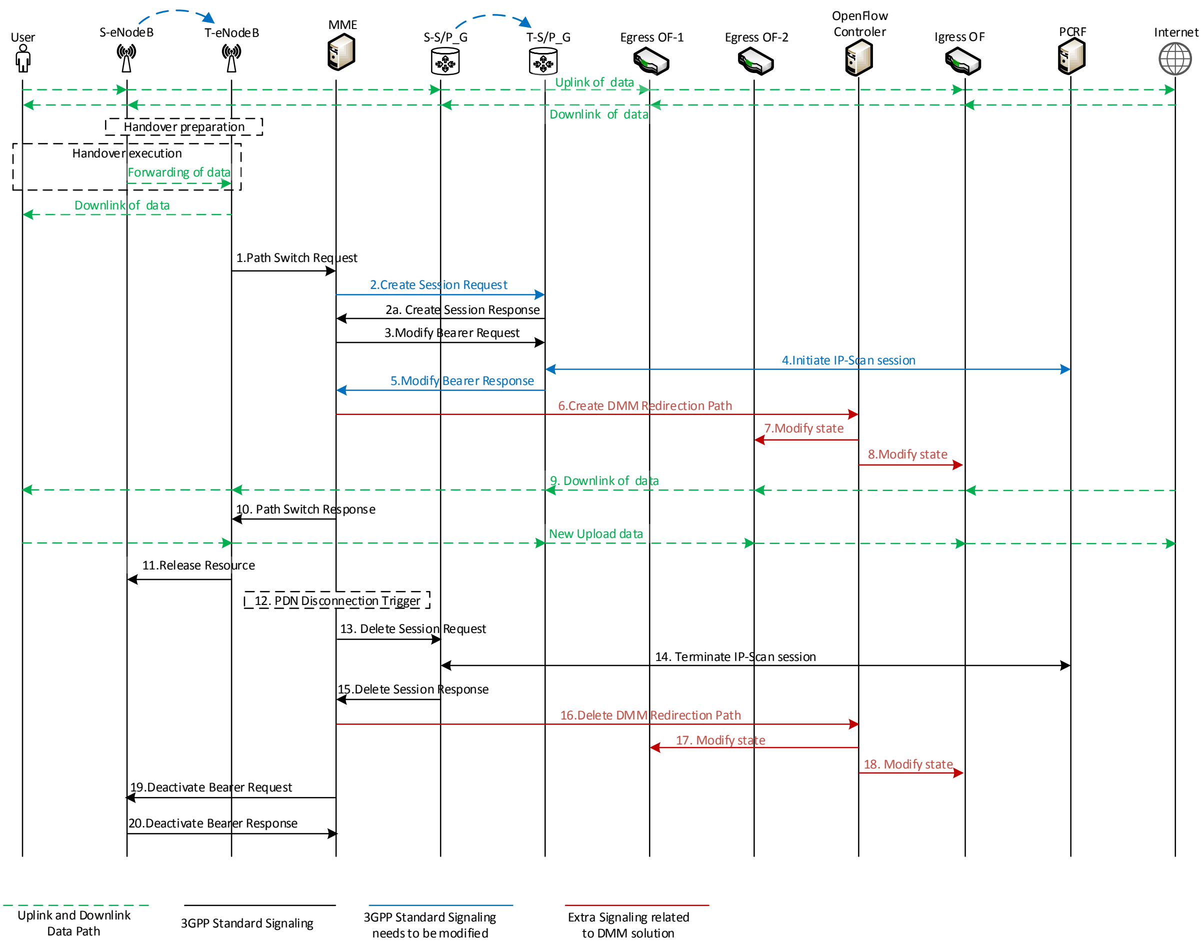


Figure 31 - DMM solution's signalling procedures over X2-handover

The signalling procedures are as follows;

1. The target eNodeB sends a "*Path Switch Request"* message to MME to inform that the UE has changed cell, including the target cell ID and the list of EPS bearers to be switched.
2. The MME sends a "*Create Session Request"* (*EPS bearer context(s), …*) message per PDN connection to the target S-/P-GW.

2a.The target S/P-GW sends a "*Create Session Response"* (*S/P-GW addresses and uplink TEID(s) for user plane*) message back to the MME. The MME starts a timer, to be used in step 15.

3. The MME sends a "*Modify Bearer Request"* (*eNodeB address and TEID allocated at the target eNodeB for downlink traffic on S1-U for the accepted EPS bearers*) message to the target S/P-GW for each accepted PDN connection.
4. Target S/P-GW executes a PCEF *"Initiated IP-CAN Session Modification"* procedure with the PCRF, to report the new IP-CAN type.
5. The S/P-GW creates a new entry in its EPS bearer context table. The new entry allows the S/P-GW to route user plane PDUs between the E-UTRAN and the Packet Data Network. The target S/P-GW sends a *"Modify Bearer Response message"* to the MME. The message is a response to a message sent at step 3.

6. The MME sends a *"Create DMM Redirection Path"* message by making flag field *(to either 01, 10 or 11) value*) and setting the *current_address* field as the IP address extracted from the *"Modify Bearer Response"* message received from the target S/P-GW. By looking in the UE Context of the UE object, the MME can retrieve the IP address belonging to flow(s) kept active by the UE after handover. This address is set as the *previous_address* field. If more than one active IP addresses are found in the UE Context, the MME duplicates this message created so far and set the *previous_address* field accordingly. Each message is then encapsulated in a TCP/IP packet using the information retrieved from the OpenFlow Controller Info table and then forwarded to the OpenFlow Controller(s).

7&8. A different *"Modify-State"* message (*using Set-Field and Output*) is created for the Egress switch and Ingress switches. In the Egress *Set-Field* action is defined in a way that reverses the modification performed on the data packet's headers by the Ingress switch.

9. Forwarding path is created through the transport network and data is downloaded to the user.

10. The MME confirms the "*Path Switch Request"* message with the "*Path Switch Response"* message (*S/P-GW addresses and uplink TEID(s) for user plane*). The target eNodeB starts using the new S/P GW address(es) and TEID(s) for forwarding subsequent uplink packets.

11. By sending "*Release Resource"* message the target eNodeB informs success of the handover to source NodeB and triggers the release of resources.

12. The MME "*trigger PDN disconnection"* procedure to source S/P-GW. Bearers including the default bearer of this PDN shall be deleted during this procedure. The procedure also allows the MME to initiate the release of the source S/P-GW PDN connection.

13. The EPS bearers in the source S/P-GW for the particular PDN connection are deactivated by the MME by sending "*Delete Session Request"* to the source S-/P-GW. This message includes an indication that all bearers belonging to that PDN connection shall be released.

14. The source S/P-GW employs the PCEF "*IP-CAN Session Termination"* procedure to indicate to the PCRF that the IP-CAN session is released if PCRF is applied in the network.

15. The S/P-GW acknowledges with "*Delete Session Response".*

16. The MME sends a "*Delete DMM Redirection Path"* message by making flag field (*to 00*). *Current_address* and *previous_address* messages are extracted from the default bearer indicated "*trigger PDN disconnection"* procedure. This message is then encapsulated in a TCP/IP packet using the information retrieved from the OpenFlow Controller Info table and then forwarded to the OpenFlow Controller(s).

17&18. OpenFlow Controller sends two sets of "*Modify-State messages"* to the Ingress switches and the Egress switch respectively to reset the setting applied in 7 & 8.

19. The MME shall delete the corresponding contexts of the PDN connection locally. It initiates the deactivation of all Bearers associated with the PDN connection to the source eNodeB by sending the "*Deactivate Bearer Request"* message.

20. The source eNodeB sends an "*Deactivate Bearer Response"* as an acknowledgement of the deactivation to the MME.

4.2.1 SDN/OpenFlow Based DMM Solution

The capabilities offered by OpenFlow, as the most common communication protocol used in SDN approach [OpenFlow 1.3.0], would be as an enabler to access the forwarding plane of OpenFlow-enabled switches over the network and reconfigure it according to the needs of applications and network services. The flow-tables in OpenFlow-capable switches can be remotely programmed via the OpenFlow Controller (OC), such that each traffic path could be traced from an Ingress node to an

Egress node as a separate flow and traffic could be redirected to a new mobility anchor point without any IP address translation or modification. OpenFlow-enabled switches contain a set of actions that could be applied to flow-specific packets providing further capabilities. The *Set-Field* action is the most relevant one for our purpose, giving to OpenFlow switches the possibility to modify packets' and frames' headers. A combination of *Set-Field* and *Output* actions (specifying the output interface) could be utilized to support dynamic per-flow redirection. Both flow-tables and action lists modifications are accomplished by the OC via a dedicated secure connection with each switch [OpenFlow 1.3.0]. Two main different DMM solutions can be deployed using the set of features provided by OpenFlow as Full OpenFlow and Partial OpenFlow. In the first approach, all switches in the operator's transport network are OpenFlow-enabled and no modifications to incoming packets are needed for traffic redirection. For the Partial OpenFlow solution, a hybrid network model could by applied, in which the OpenFlow switches are placed at the downlink Ingress (FE_I) and Egress (FE_E) nodes, while IP routers are used in the core network. In this approach, traffic redirection on the core part is based on currently deployed layer 3 routing mechanisms. The choice of the approach depends on the operator's investment roadmap, given the fact that switching to a full OpenFlow-enabled network can be a large economic investment. In [Valt13], both solutions have been addressed. In this deliverable, only the Partial OpenFlow based approach is considered in the following.

4.2.2 Partial OpenFlow based DMM

Figure 32, shows the Partial OpenFlow based DMM approach topology based on the DMM architecture presented in Section 4.1.1. In this topology the OpenFlow-enabled Ingress (FE_I) and Egress (FE_E) switches are managed by the same OC (acting as the FE_IEC). When a UE attaches to the access network for the first time, it gets an IP address allocated from its serving P-GW, which is the mobility anchor for flow(s) initiated by the UE when attached to it. This IP address is used to forward traffic only outside of the operator's transport network (e.g., in the EPC, in the Internet, etc.). To provide IP address continuity when a UE changes its mobility anchor point, while keeping some flows active, inside the operator's transport network a different IP address will be used to route traffic to the current UE's P-GW. This address can be the SGi address of the current UE's mobility anchor point. Ingress and Egress OpenFlow switches will take care of replacing the original destination IP address with the SGi address using the *Set-Field* action. The *Output* action is needed in both Ingress and Egress OpenFlow switches to forward incoming downlink traffic to the transport network and EPC network, respectively.

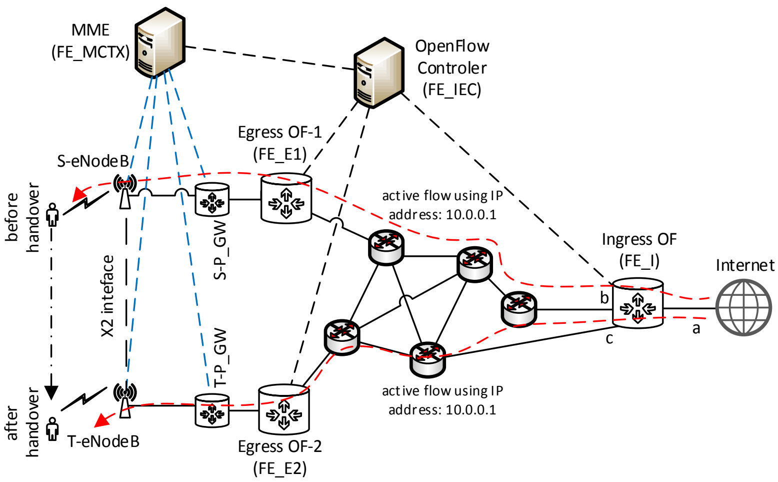


Figure 32 - Partial OpenFlow based DMM (before and after handover)

Figure 32, represents the downlink path for the flow before and after that UE has been handed over from the source eNodeB to the target eNodeB (X2 capabilities between eNodeBs can be used to forward user's downlink traffic during the handover procedure, if available). Assume that the UE was previously attached to the source P-GW and had initiated a flow using an IP address allocated from the P-GW's address pool (for instance 10.0.0.1). When it is handed over to the target P-GW and wishes to keep the previously initiated flow active, Set-Field instructions need to be present into the action set of both Ingress OF(OpenFlow) and Egress OF-2 switches (to replace the destination IP address (10.0.0.1) with the target P-GW's SGi address in the Ingress OF and to do the contrariwise replacement in Egress OF-2). Output actions are also needed in both Ingress and Egress switches. In the Ingress OF, it will forward incoming downlink traffic belonging to flow via interface c. In the Egress OF-2, the incoming downlink traffic will be forwarded via the interface that connects the switch to the target P-GW. In this way, one IP address can be used in the transport network to route traffic belonging to multiple flows anchored at a target P-GW leading to a less complex and more scalable flows management. Flow matching in the Egress switch is then based on the combination of the source IP address with the transport protocol ports.

5 Radio Access Network

MCN aims at enabling the end-to-end provision of mobile networks to Enterprise End Users (EEUs), being the access part provided through the Radio Access Network as a Service (RANaaS). The RANaaS architecture, presented in D2.2 [BoAd13], will enable an efficient usage of datacentre resources, through the dynamic deployment of the required resources. In deliverable D3.2 [MCN-D3.2] several research topics were presented within the scope of RANaaS:

- Profiling of radio and cloud resources of a software-based base-station, so called Base Band Unit (BBU).
- Dimensioning and location of pools of BBUs.
- Virtual radio resource management.

In the present chapter, algorithms, mechanisms and preliminary results on these ongoing works are presented.

5.1 Profiling of BBU radio and cloud resources

In this section, we present EURECOM Centralized RAN code evaluation. An overview of the C-RAN concept is available in Annex 8.1, which enables the concept of RANaaS, presented in D3.1 [MCN-D3.1] and D3.2 [MCN-D3.2]. In our set-up, we study C-RAN in which signal processing runs in the cloud on virtual machines (VM) equipped with typical General Purpose Processors (GPP). Our results are based on the Open Air Interface (OAI) LTE-Release-8 software emulator. This approach provides a precise method for the estimation of required deployment configurations of processing resources to efficiently handle momentary traffic load.

5.1.1 Cloud Platform for LTE RAN

Virtualization is the process to let one physical machine acting as several logical machines using a layer of software called a hypervisor. Current Infrastructure-as-a-Service (IaaS) platforms provide customers with processing, storage, network and other fundamental services to run variant range of applications. This utilizes the cost per virtual machine hour, saves floor space, electricity and cooling consumption. This section describes virtualization for mobile operators, their software and processing requirements.

Most of the heavy computational processing requirements in LTE RAN are in BBUs, which depends directly on the number of UE per eNB. The C-RAN architecture presents different challenges, specifically related to the real-time processing of the baseband signals, which has strict requirements for processing deadlines that must be met regardless the system load. The highest demand on signal processing deadlines comes from the Hybrid Automatic Repeat Request (HARQ) protocol. HARQ is a retransmission protocol between eNB and UE stated that the reception status of any received packet has to be reported to the transmitter. In LTE networks using FDD, the HARQ Round Trip Time (RTT) equals 8 ms. The transmission time of a LTE subframe $T_{subframe}$ is 1 ms and the processing time of eNB and UE is 3 ms each. This means that each packet received at subframe k has to be acknowledged, i.e., ACK or NACK, at subframe k+4 in both eNB and UE sides. The eNB processing includes (i) received packet decoding, (ii) ACK/NACK creation, and (iii) ACK/NACK subframe encoding. The authors in [Cost13] propose to relate the HARQ acknowledgement directly to SINR. The ACK/NACK creation and subframe encoding becomes a standalone algorithm running in parallel to the decoding process. Hence, the HARQ processing bottleneck can be limited to the packet decoding time. Moreover, in C-

RAN architecture, the maximum separation distance between RRHs and BBU pool should be considered into delay calculation.

OpenStack is a cloud management system that orchestrates different components such as compute, storage, and networking to control and manage the execution of virtual machines (VMs) on the server pool.

5.1.2 OpenAirInterface

OpenAirInterface (OAI) emulation platform is an open-source software from EURECOM that follows the LTE 3GPP Release-8 standards. The platform is developed by EURECOM. OAI provides a complete wireless protocol stack containing the PHY, MAC, RLC, PDCP, and RRC layers as well as a Non-Access-Stratum (NAS) driver for IPv4/IPv6 interconnection with network Services. The objective of this software platform is to fill the gap between dedicated-hardware and software-based implementations of different LTE network entities. Regarding the PHY layer, OAI implements the signal processing chains for OFDMA (Downlink) and SC-FDMA (Uplink) as expressed in Annex 8.1.3. OAI uses optimized C code for Intel architectures (using MMX/SSE3/SSE4 intrinsic) which can be used for efficient numerical performance evaluation. We are using OAI as a benchmark for profiling the processing time of the LTE PHY layer given different load configurations, expressed I nterms of number of Physical Resource Blocks (PPRB) and Modulation and Coding Schemes (MCS) on cloud. More specifically, we are using two benchmark tools called dlsim, which emulates the PDSCH downlink channel, and ulsim, which emulates the PUSCH uplink channel. Both tools are designed to emulate the behaviour of eNB and UE transceivers over an emulated wireless medium. Moreover, the processing time of each signal-processing module in the downlink and uplink communication is calculated using time counters at the beginning and end of each module. OAI uses an instruction, available in all x86 and x64 processors, which counts the number of CPU clocks since reset. The processing time $T_{process}$ of a certain module is calculated by dividing the difference of the CPU clocks by the CPU frequency.

5.1.3 Evaluation

In this section, we draw our results that should be considered while designing and deploying LTE-FDD C-RAN.

5.1.3.1 Processing Times

As mentioned before, once an eNB receive a subframe, it has to decode it, build Ack/Nack subframe and encode the new subframe in 3 ms. In this section, we study the decoding processing time using the receiver part of OAI ulsim as illustrated in Figure 33, and the encoding processing time using the transmitter part of OAI dlsim as illustrated in Figure 34. In each figure, we plot the overall processing time for different modulation and coding schemes (MCS: 0, 9, 10, 16, 17, 2, and 27), different system bandwidth (5, 10, and 20 MHz), and on different machine environments (dedicated GPP, KVM, and cloud (ZHAW Openstack testbed [ZHAW14]). Every machine has 2GB RAM and 2.4GHz CPU.

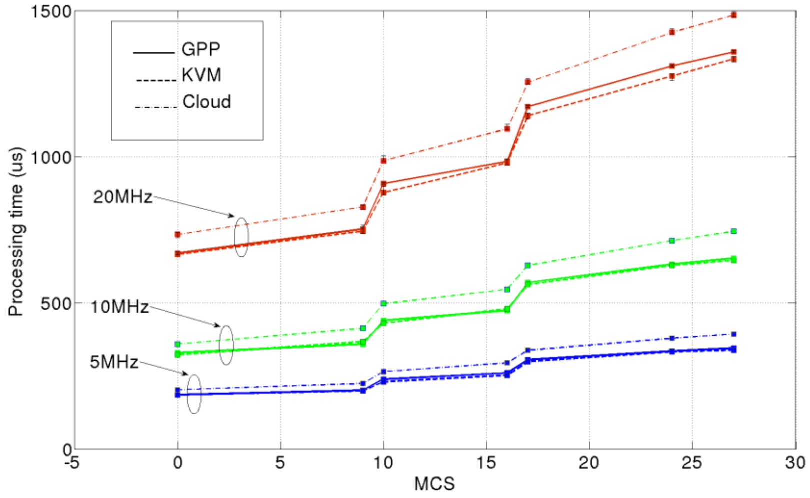


Figure 33 - Processing time requirement for dlsim-Tx

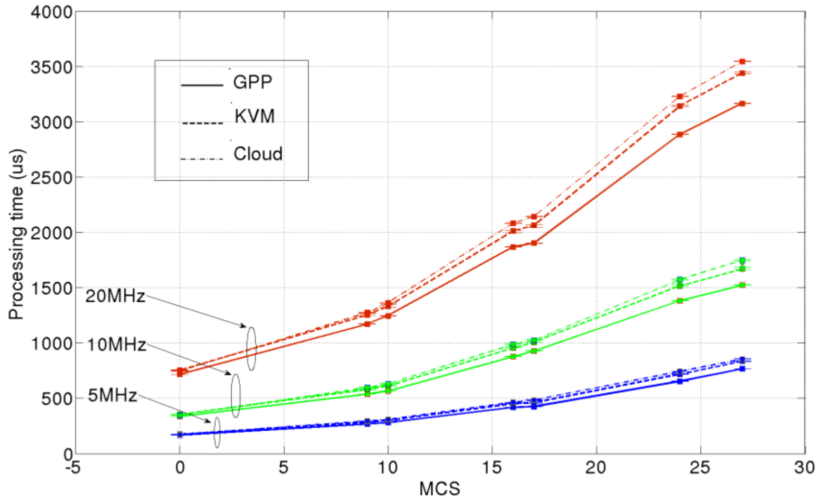


Figure 34 - Processing time requirements for ulsim-Rx

These figures allow us to provide the following observations:

- The decoding and encoding processing time for LTE subframes increases with the increase of MCS given 25, 50, and 100 PRB for 5, 10, and 20 MHz bandwidths. On average, the decoding time is twice as long as the encoding time.
- Given that all machines have the same RAM size and CPU frequency we see that the median values for the cloud VM show lower performance (more processing time) than KVM VM and GPP machine. This can be explained by the natural implementation of cloud resource sharing mechanisms.
- According to OAI implementation, subframes that require more than 2 ms for decoding will be negatively acknowledged. Thus, all MCS requiring total decoding time greater than 2 ms cannot be supported by these machines specifications.

5.1.3.2 Required CPU frequency

After we studied the behaviour of processing time given different system bandwidths and MCSs, it is important, from the system design point of view, to know the minimum required CPU frequency that supports certain SLA agreement, e.g., data rate. Figure 35 illustrates the eNB overall processing time (decoding SC-FDMA subframe and encoding OFDMA subframe) given different CPU frequencies (1.6, 2.0, 2.4, 2.8, 3.3 GHz). Note that we consider the worst case scenario defined by the LTE standards, which stated that UE transmits PUSCH subframe with MCS equals 27 (UEs of category 5) and the eNB transmits the Ack/Nack using PDSCH channel (note that encoding of PHSCH is faster than PDSCH). In order to perform experiments at different CPU frequencies, we used *cpupower* tool to limit the available CPU clock cycles and utilize workload configurations (e.g., *userdemand*) [Pall06] .

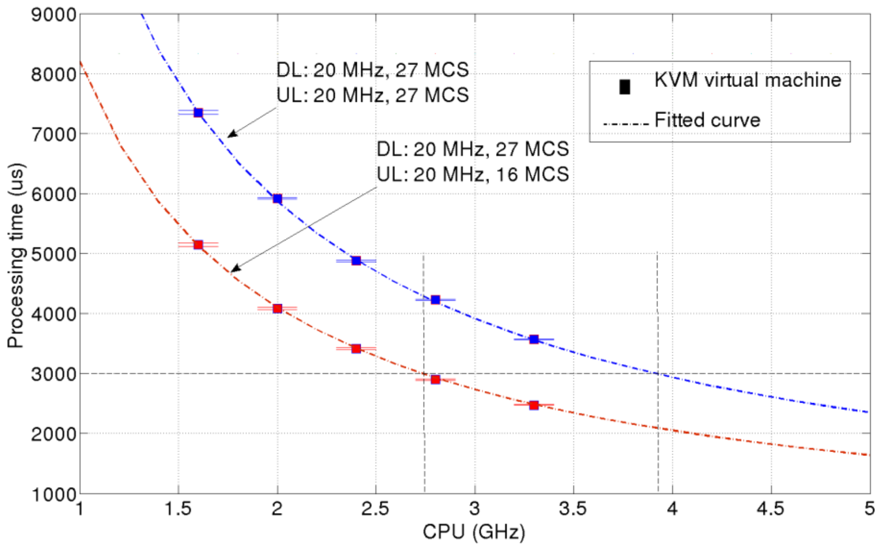


Figure 35 - Overall processing time requirements at the eNB side

From Figure 35, we observe the exponential behaviour of the processing time versus CPU frequency. We used curve fitting from MATLAB to model the processing time $T_{subframe}$ for various Intel based processor frequencies, expressed by the following equation

$$T_{subframe} = \frac{1174}{CPU\ (GHz)}$$

From the equation above, cloud operators would require at least VMs with 4 GHz based CPU to support the aforementioned scenario.

5.1.3.3 Processing Time Distribution

During our experiments, we observed higher variations in cloud and KVM environment than dedicated GPP. In Figure 36, we modelled the distribution of the overall signal processing on cloud VM using MATLAB. From our observations, the processing time distributions are mostly skewed to the right. The best distribution fit for the given raw data of MCS equals 27 and 5, 10, and 20 MHz is *lognormal* distribution, which meets the findings in [Bake09]. Moreover, we see that the more processing time an operation requires, the more variations (Inter-Quantile Range IQR) it has. This result affects directly the system efficiency (data rate) and dropped packet rate. Figure 37 shows explicitly the number of subframes exceeding a decoding processing time of 2 ms given that the total number of subframes equals 10000. The figure contains data from Cloud, KVM, and GPP machines. Each subplot in Figure 37 has MCS and bandwidth input variables. We can see clearly that non-dedicated machines cannot guarantee exact CPU frequency at any instance but on average. This affects services that have strict processing deadlines, such as gaming applications, and should be "cloudified".

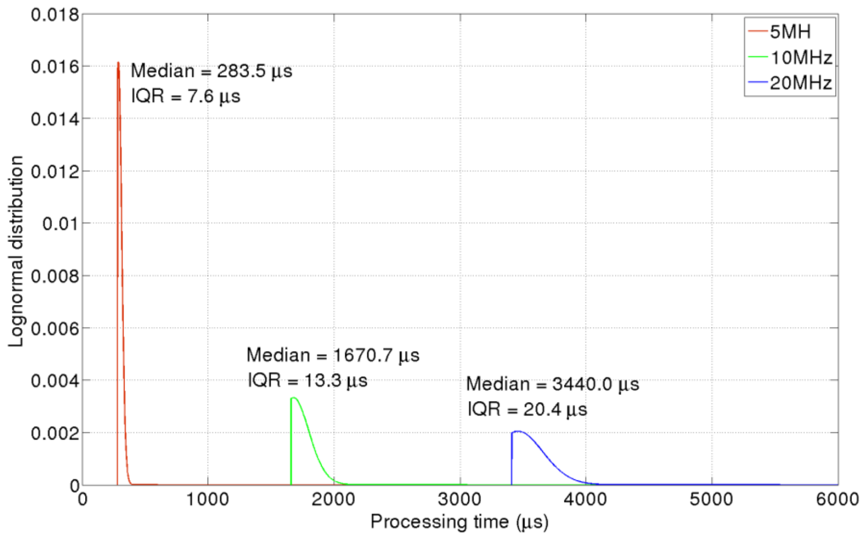


Figure 36 - Processing time distribution at the eNB

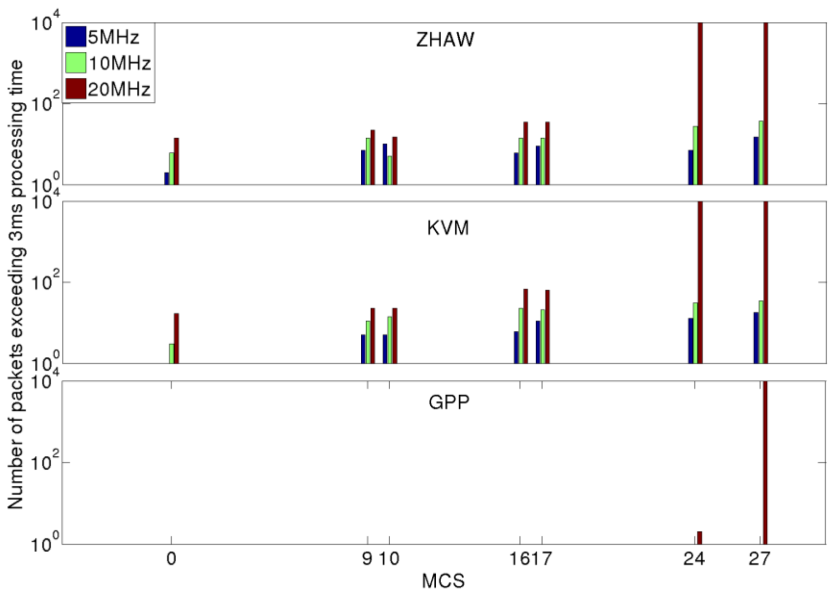


Figure 37 - number of packets exceeding 3ms deadline

5.2 BBU pool dimensioning

In C-RAN, BBU pools located on a datacentre are connected to Remote Radio Heads (RRHs) via fibre. This poses several requirements in terms of capacity of fibre links as well as processing in the datacentre to satisfy maximum latencies for processing subframes. The state of the art and challenges of dimensioning BBU pools have been introduced in D3.1 [MCN-D3.1], while fronthaul solutions have been described in D3.2 [MCN-D3.2]. On the basis of these elements and considering existing fronthaul requirements, a dimensioning example based on a real network configuration is presented in this section.

The exercise of defining and dimensioning BBU pools (also called hotels) is done for a scenario consisting of the French Brittany region, with a typical mix of dense urban, semi-rural and rural areas, as shown in Figure 38. This region covers about 30000 km2 with 3.2 million inhabitants.

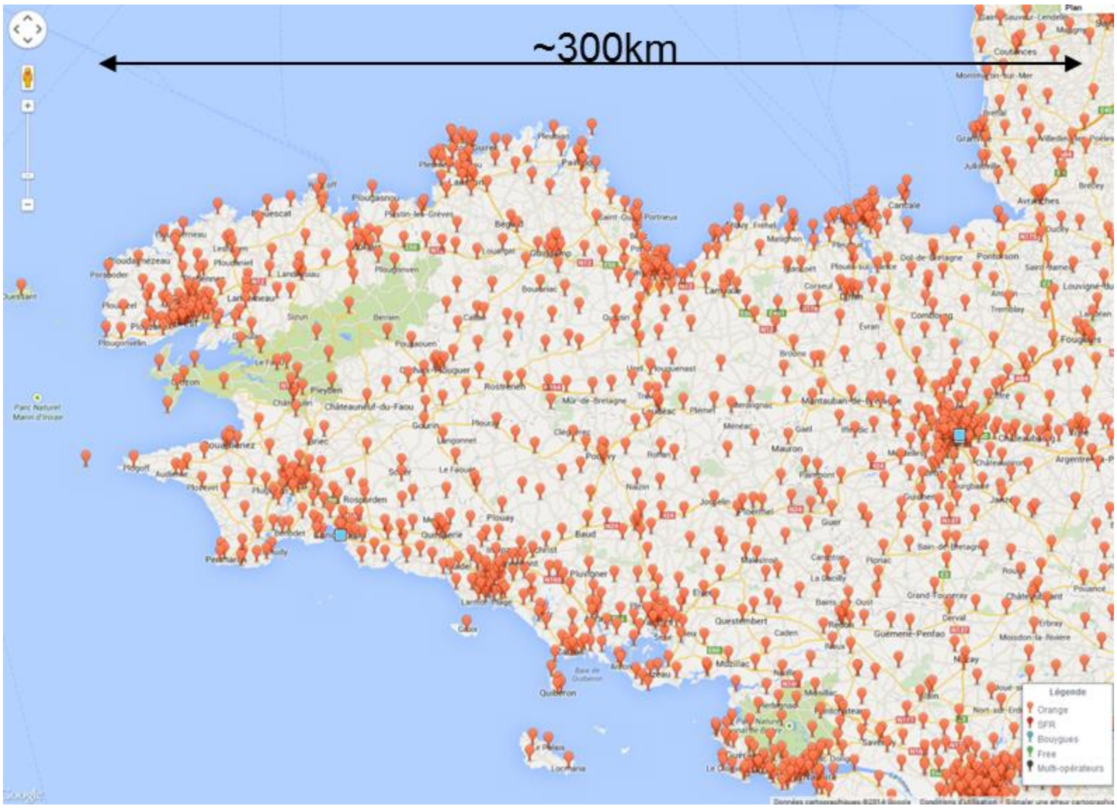


Figure 38 – Antenna site locations in French Brittany.

The network model used for this scenario is based on public information on antenna site location provided by Agence Nationale Fréquence Radio (ANFR) and on statistical data for Central Office (CO) location extracted by Orange internal data. The considered area comprises nearly 900 antenna sites. Most of them present multi Radio Access Technology (RAT) (2G, 3G and LTE) operating in various carriers. Moreover, they have three sectors. With these assumptions, the goal of the study is to find optimal BBU hotel locations keeping into account for:

- Existing fixed infrastructure.
- Fronthaul requirements.

A tool has been developed to visualize on a map the antenna site locations as well as the existing fixed infrastructure, i.e., Central Offices (CO).

In France, a big part of existing antenna sites is already connected to a central office by optical fibre and in general a pair of fibre is available. In particular, in urban areas, more than 90% of antenna sites are connected to the CO by optical fibre. As a consequence, in this analysis we have assumed that a fibre pair is available at each antenna site.

We assume that BBU pools will be naturally located in existing COs and co-located with fixed network equipment. This choice is also in the direction of a *first level* of fixed-mobile structural convergence, as proposed in FP7 COMBO project [COMBO].

The fixed network COs have a hierarchical structure of 3 levels:

- first level Central Offices;
- Main Central Offices;
- Core Central Offices.

In the considered scenario, depicted in Figure 39, there are nearly 900 first level Central Offices (green points in the figure below), 15 Main central offices (red points) and 2 Core Central Offices (blue square).

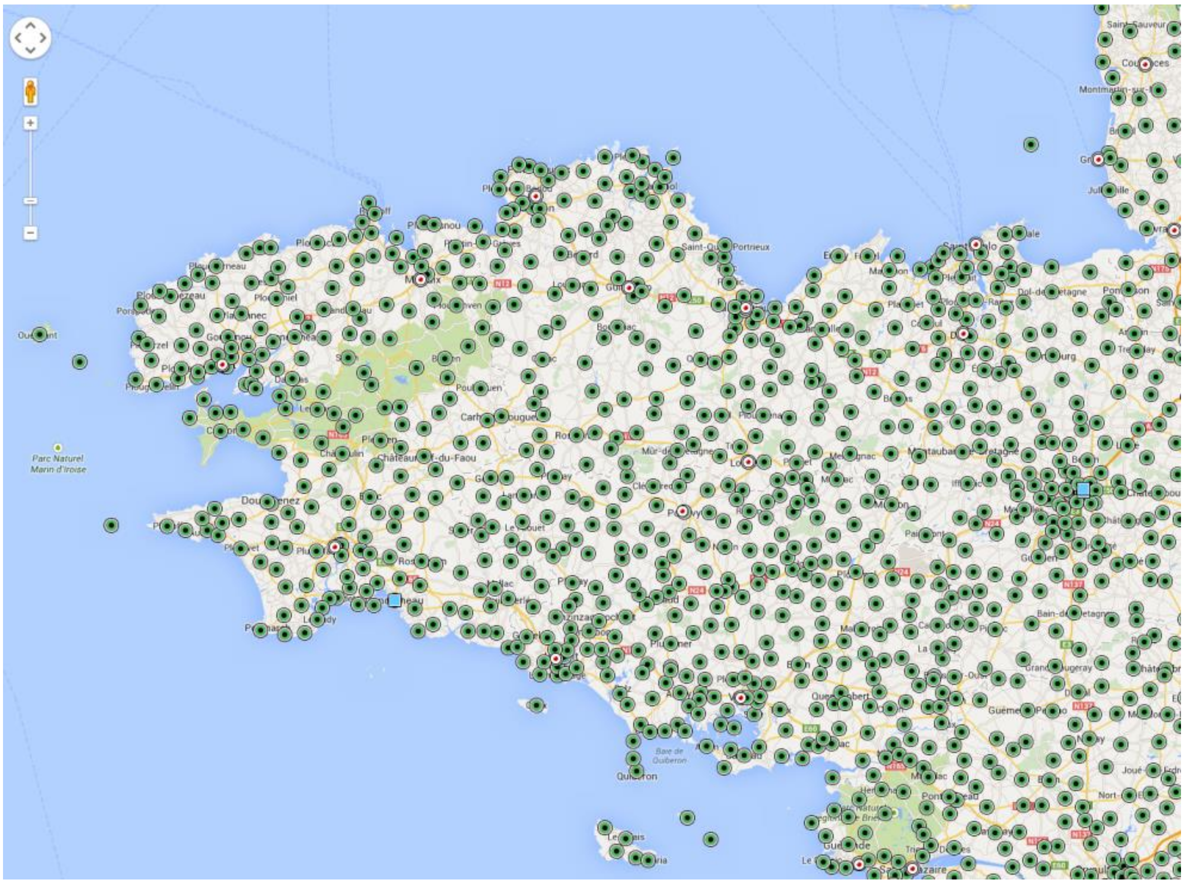


Figure 39 – Location of central offices (green), main central offices (red) and core central offices (blue) in French Brittany.

Concerning the fronthaul requirements, we consider the implementation of a passive Coarse Wavelength Division Multiplexing solution that allows transmitting on a fibre-pair up to 18 (Common Public Radio Interface) CPRI links (corresponding to 18 RRHs). This fits well with the maximum of 15 RRH for each antenna site.

The other fronthaul requirement to be taken into account for BBU pool dimensioning is the maximum allowed round trip latency. This value is still under discussion, which results in maximal RRH-BBU distances. Legacy eNodeBs (already installed) can support up to 15km fibre length between RRH and BBU, whereas NGMN requirement is to support up to 50km.

To find optimal BBU locations, the tool has been enriched with the approximate coverage area of each antenna site. Moreover, it is possible to select a BBU location, fix a radius and determine the antenna sites that fall in such coverage.

Figure 40 shows an example of antenna sites in a radius of 15km from a Core Central Office in one of the big cities of Brittany. As a general comment, a RRH-BBU distance of 15km is enough considering one BBU location for each dense urban area. However, this is not possible for less dense areas (e.g., rural ones), where the BBU-RRH distance can be larger than the maximum distance. This would require having extra BBU locations.

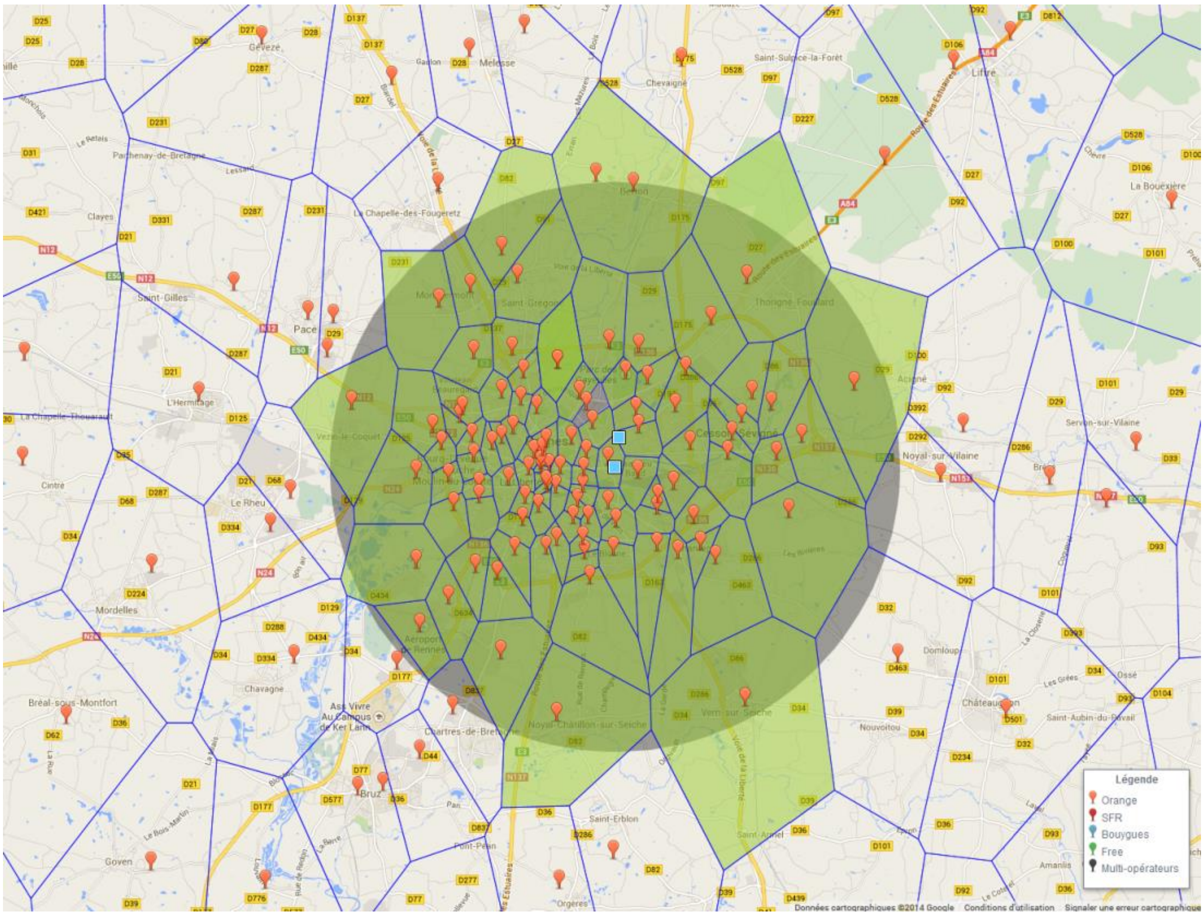


Figure 40 – Antenna sites locations (in orange) in a radius of 15km from a Core Central Office (in blue) in one of the big cities of Brittany.

Simulations have been performed with different values of fronthaul maximum distances in order to find optimal BBU locations, without increasing too much their number.

As a result of the proposed mechanism, Figure 41 shows BBU pool dimensioning results for the scenario under analysis. A good compromise is obtained considering BBU locations in the Core Central Offices and in the Main Central Offices. As shown in Table 9, under these assumptions, more than 90% cell sites are covered by 14 BBU pool locations. Each BBU pool corresponds to several tens of antenna sites up to more than a hundred in dense urban areas. This seems a good dimensioning also for pooling gains. Moreover, in urban areas, cell densification with small cells will even increase pooling gains.

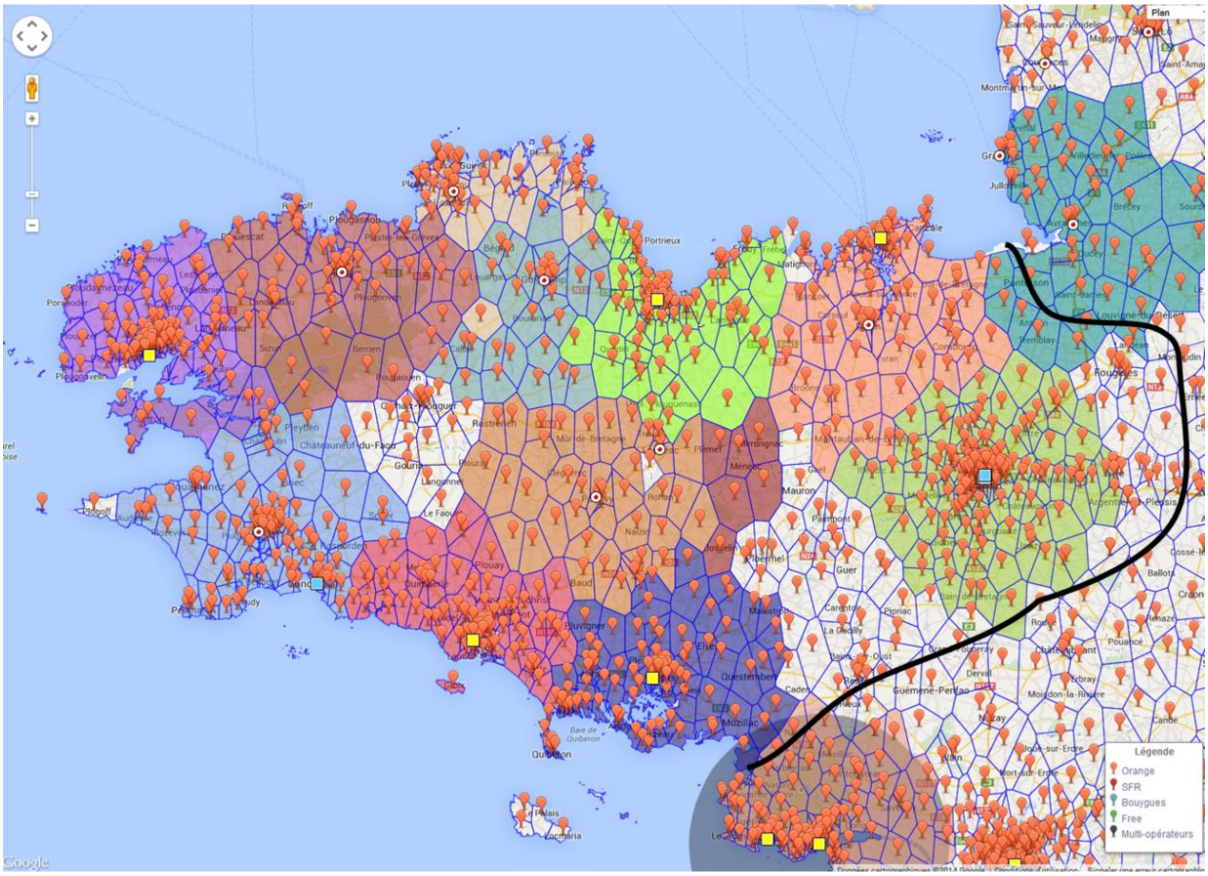


Figure 41 – BBU pool locations in the core and main central offices in French Brittany.

The 10% cell sites that are not reached are all in rural areas. The following options can be considered: either add at least 4 new BBU pools, in order to guarantee the maximal BBU-RRH distance, or to extend the fronthaul maximum distance to 60km. The feasibility of this last option has to be further studied, but a preliminary analysis would be in a positive direction as Coordinated Multipoint implementation is not critical in rural areas.

Table 9 – Number of radio sites per BBU pool hotel.

BBU hotel	Radio sites
1	194
2	100
3	51
4	39
5	31
6	77
7	45
8	91
9	98
10	88
11	93
12	53
13	6
14	131

5.3 Virtual Radio Resource Management

As referred to in D4.1 [Tale13], the interaction between RANaaS and EPCaaS is of high importance. As identified in D3.1 and D3.2 as well as later introduced in D2.2, Management of Virtual Radio Resources is an ongoing research activity within RANaaS. The virtualization concept of radio resources that provides virtual wireless links is presented in Annex 8.2. In this section, an overview of concepts, algorithms, and first results are presented. As the main contribution, VRRM offers and guarantees Service Level Agreements (SLAs) to multiple EEUs tenants using a typical physical infrastructure.

5.3.1 Analytical Model for VRRM

5.3.1.1 Radio Resource Management hierarchy in V-RAN

The goal of Virtual Radio Resource Management (VRRM) is to meet the instances' objectives while optimizing the resource usage and other additional objectives (e.g., guaranteeing fairness among multiple instances). Wireless links are always subject to fading and interference, and their performance is not certain. As a matter of fact, management of resources in order to achieve the aforementioned objectives is an elaborated procedure. The importance and complexity of VRRM is the reason to count it as a milestone in realization of end-to-end virtual radio network. Management of virtual radio resources, as depicted in Figure 42, is hierarchal and consists of four key actors: VNOs, VRRM, Common Radio Resource Management (CRRM), and local RRMs.

At the highest level of this hierarchy, we deal with VNOs, which do not own any RAN infrastructure and have to ask for wireless channel of certain capacity to appropriately serve subscribers. When the capacity is granted to a VNO, it can admit UEs requesting the service. VRRM makes sure that the virtual wireless links carrying traffic of the service can meet the QoS requirements as long as the service is active. In addition to the QoS requirements of every service session, VNOs request for a level of service quality from VRRM in terms of guaranteed capacity per service. The SLAs can be summarized as three basic types of contracts:

- **Guaranteed** in which VNOs are guaranteed with a minimum as well as a maximum level of service (e.g., data rate) to be served on-demand. The operator choosing a guaranteed contract receives guaranteed capacity regardless the network situation; its subscribers also experience a good quality of service.
- **Best effort with minimum guarantees** in which VNOs are provided with a minimum level of service. A request for better services (e.g., higher data rates) is served in the best effort manner. The VNO does not invest as much as an operator with a guaranteed contract, but it can guarantee a minimum QoS to its subscribers. Since the VNO is served in the best effort manner, it is expected to be charged comparatively less than a guaranteed VNO. Subscribers of a VNO of this type experience acceptable network performance, but not as good as subscribers of a guaranteed VNO. Nevertheless, this comes at a relatively lower price, since the virtual operator pays less for the RAN.
- **Best effort** in which VNOs are served in the pure best effort manner. This contract type has the lowest cost, however, the operators and consequently, their subscriber may suffer from a low service quality at rush hours.

VRRM, the highest resource manager, is in charge of translating VNOs' requirements and SLAs into the set of policies for lower layers. In other words, the key role of VRRM is to map virtual radio resources onto physical ones and to issue policies managing virtual resources. VRRM does not handle physical radio resources. Nevertheless, reports and monitoring information (e.g., estimated remained capacity) received from CRRM enable it to improve the policies. It should be stated that in a virtual network, in addition to radio resources, there also exist processing capacities and fibre resources, which should be commonly managed. The scope of this framework puts a particular focus, however, only on management of radio resources. CRRM and Local RRMs are usual resource managers in heterogeneous access networks [PeGS09]. The former manages resources of different Radio Access Technologies (RATs), while the latter are liable to the optimization of radio resource usage in a single access technology.

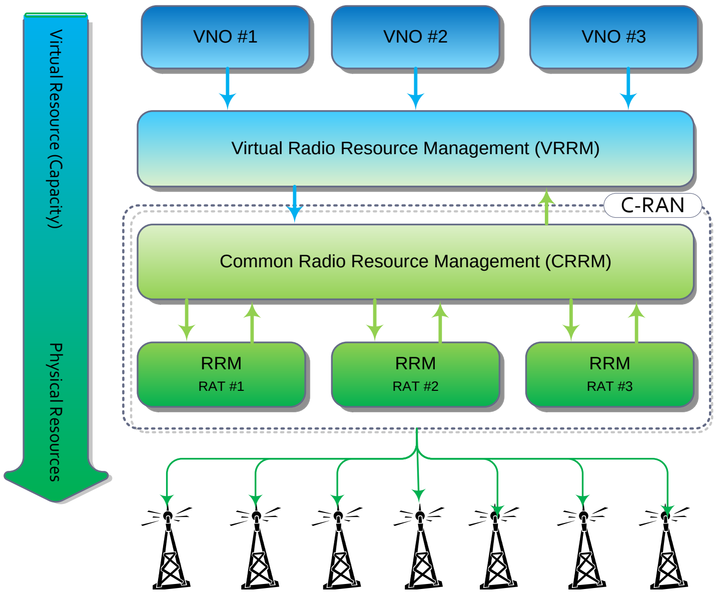


Figure 42 – Radio Resource Management hierarchy in V-RAN.

In the following, an analytical model for the management of virtual radio resources is described comprehensively. In the first step, a realistic mapping between available radio resources and the network capacity is considered. Modelling of virtual radio resource management, as well as any other management problem, contains two key parts that is estimation and allocation of available resources. Having an estimation of the available capacity, the allocation of data rates to different services of the VNOs could be later optimized.

5.3.1.2 Estimation of Available Virtual Radio Resource

The goal of the estimation of available virtual radio resources is to obtain a relation as the Probability Density Function (PDF) from the set of available resources to the total network capacity. In general, the applicable data rate provided through the assignment of a radio resource unit (e.g. time slot, radio block or code block) to a terminal, varies between the minimum and maximum data rata based on various parameters such as RAT, modulation, coding schemes, etc.. For a given configuration (e.g., certain modulation), the data rate of a single Radio Resource Unit (RRU) is a function of a channel quality, Signal-to-Interference-plus-Noise Ratio (SINR), varies between the minimum and maximum levels as follows:

$$R_{b_{RAT_i[\mathrm{Mbps}]}}(\rho_{in}) \in \left[0, R_{b_{RAT_i[\mathrm{Mbps}]}}^{max}\right], \tag{1}$$

where:

- $R_{b_{RAT_i}}$: the data rate of a single RRU of i-th RAT,
- ρ_{in}: input SINR,
- $R_{b_{RAT_i}}^{max}$: the maximum data rate of a single RRU of i-th RAT.

In [Jaci09], based on real traces, the data rate as a function of input SINR (and the other way around) for various access technologies have been studied. In the next step, for the sake of simplicity, these functions, have been approximated through an equivalent quantic polynomial. Therefore, the SINR as a function of the data rate goes as follows:

$$\rho_{in_{[\mathrm{dB}]}}\left(R_{b_{RAT_i}}\right) = \sum_{k=0}^{5} a_{k_{\left[\frac{\mathrm{dB}}{\mathrm{Mbps}^k}\right]}} R_{b_{RAT_i[\mathrm{Mbps}]}}^{k}, \tag{2}$$

where:

- a_k: coefficients of the polynomial approximation of SINR as a function of data in each RAT.

Based on [DGBA12], the PDF of $R_{b_{RAT_i}}$, can be expressed as follows:

$$p_{R_b}\left(R_{b_{RAT_i[\mathrm{Mbps}]}}\right) = \frac{\frac{0.2}{\alpha_p}\ln(10)\left(\sum_{k=1}^{5} k a_k \left(R_{b_{RAT_i}}\right)^{k-1}\right) e^{-\frac{0.2}{\alpha_p}\ln(10)\sum_{k=0}^{5} a_k \left(R_{b_{RAT_i}}\right)^k}}{e^{-\frac{0.2}{\alpha_p}\ln(10) a_0} - e^{-\frac{0.2}{\alpha_p}\ln(10)\sum_{k=0}^{5} a_k (R_{b_{RAT_i}}^{max})^k}}, \tag{3}$$

where:

- α_p: the path loss exponent where $\alpha_p \geq 2$.

The next step is to extend the model and estimate the overall performance of a single RAT. The total data rate from a single RAT pool is:

$$R_{b_{tot}}^{RAT_i} = \sum_{n=1}^{N_{RRU}^{RAT_i}} R_{b_n}^{RAT_i}, \tag{4}$$

where:

- $N_{RRU}^{RAT_i}$: the number of Radio Resources Unite (RRUs) in *i*-th RAT,
- $R_{b_{tot}}^{RAT_i}$: the data rate from a *i*-th RAT pool,
- $R_{b_n}^{RAT_i}$: the data rate from *n*-th RRU of *i*-th RAT.

Based on the assumption that the channels are independent, random variables of the achieved data rates R_{bi} are also independent. The PDF of the total data rate is also equal to the sum of all the random variables. Let $p_{R_{bi}}\left(R_{b_n}^{RAT_i}\right)$ be PDF of *i*-th radio resource unit, based on [PaPi02], $p_{R_b}\left(R_{b_{tot}}^{RAT_i}\right)$ is equal to the convolution of all the RRUs' PDFs.

In currently deployed heterogeneous access networks, the resource pools of RATs can be aggregated under the supervision of CRRM. The total data rate from all the RATs is the sum of the total data rate.

$$R_{b\,[\mathrm{Mbps}]}^{CRRM} = \sum_{i=1}^{N_{RAT}} R_{b_{tot}[\mathrm{Mbps}]}^{RAT_i} \tag{5}$$

Having a realistic estimation on the total network capacity, a fraction of the capacity is assigned to VNO services according to the SLAs and priorities.

5.3.1.1 Resource Allocation

After estimating available resources, VRRM has an estimation of the total network capacity, which is used in the allocation procedure. The goal of the allocation procedure is to increase total network throughput while also considering the priority of different services and taking into account the SLAs of various VNOs and other constraints. Due to this fact, the objective function for VRRM $f^{v}_{\mathbf{R_b}}$, which is the total weighted network data rate in Mbps, can be written as:

$$f^{v}_{\mathbf{R_b}}(\mathbf{R_b}) = W^{cell} f^{cell}_{\mathbf{R_b}}(\mathbf{R^{cell}_b}) + W^{WLAN} f^{WLAN}_{\mathbf{R_b}}(\mathbf{R^{WLAN}_b}) - W^{f} f^{f}_{R_b}(\mathbf{R^{f}_b}), \tag{6}$$

where:

- $\mathbf{R_b}$: the vector of served data rates, which is:
- $f^{cell}_{\mathbf{R_b}}$: the objective function for cellular RATs,
- W^{cell}: the weight of allocated capacity from cellular RATs,
- $\mathbf{R^{cell}_b}$: the vector of served data rates from cellular network which is:
- W^{WLAN}: the weight for allocated capacity from Access Points (APs),
- f^{WLAN}_{v}: the objective function for APs,
- $\mathbf{R}^{WLAN}_{\mathbf{b}}$: the vector of served data rates from APs, which is:
- $f^{f}_{\mathbf{R_b}}$: the fairness function,
- $W^{f} \in [0,1]$ is the fairness weight in the objective function, which indicates how much focus should be put on the fair allocation,
- $\mathbf{R^{f}_b}$: the vector of intermediate fairness variable.

The objective function for cellular RATs addressed in (6) is given by:

$$f^{cell}_{\mathbf{R_b}}(\mathbf{R_b}) = \sum_{i=1}^{N_{VNO}} \sum_{j=1}^{N_{srv}} W^{Srv}_{ji} R^{cell}_{b_{ji}} \text{[Mbps]}, \tag{7}$$

where:

- N_{VNO}: the number of VNOs served by this VRRM,
- N_{srv}: the number of services for each VNO,
- W^{Srv}_{ji}: the weight of a service unit described by VRRM through the data rate of service j and VNO i, where $W^{Srv}_{ji} \in [0,1]$,

The weights in Eq. (7) are used to prioritize the allocation of data rates to different services of various VNOs. The choice of these weights are to be based on SLAs between VNOs and VRRM, while the sum of the weights is equal to the unit (i.e., they are normalized weights). The Services of VNOs, which are classified as background traffic, for instance, have the lowest weight in Eq. (7). It is desired that the service with relatively higher serving weights, receive higher data rate than services with lower serving weights.

The network architecture and throughput dependencies are the two reasons, based on which the cellular access network and Wi-Fi are considered separately in this work. From the network architectural point of view, Access Points (APs) are connected to the network core in a different way than cellular RATs. The cellular networks (i.e., GSM, UMTS, and LTE), considered as 3GPP trusted networks, are connected to S-GW, while the Wi-Fi network may be considered either as trusted or untrusted non-3GPP networks. Architecturally, it connects directly to P-GW, instead of S-GW, which is the classical interface of cellular RANs.

In addition, throughput of a Wi-Fi network depends on the collision rate and channel quality. It can be stated that throughput in cellular networks is almost independent of the number of connected end-users. Granted the fact that the increase of the number of end-user leads to an increment of the interference level, the network throughput still depends only on SINR. In the contention-based networks such as Wi-Fi, however, the increment of connected end-users not only decreases SINR, but also increases conflict rates. Terminals have to retransmit their packets whenever a conflict occurs. We can observe that taking the retransmission of packets alone into account (without even considering the back-off time interval after each conflict) lowers the network throughput. Thus, the objective function has to minimize the number of connected users to APs. On the ground of this discussion, the objective function for WLAN is:

$$f_{\mathbf{R_b}}^{WLAN}\left(\mathbf{R_b^{WLAN}}\right) = \sum_{i=1}^{N_{VNO}} \sum_{j=1}^{N_{srv}} W_{ji}^{Srv} R_{b_{ji}[\text{Mbps}]}^{WLAN} + W^{SRb} \sum_{i=1}^{N_{VNO}} \sum_{j=1}^{N_{srv}} \frac{\overline{R_{b_j[\text{Mbps}]}}}{\overline{R_{b[\text{Mbps}]}^{max}}} R_{b_{ji}[\text{Mbps}]}^{WLAN} \tag{8}$$

where:

- W^{SRb}: the weight for a given session average data rate, where $W^{SRb} \in [0, 1]$.
- $R_{b_{ji}}^{WLAN}$: the allocated capacity for service *j* of the VNO *i* in the Wi-Fi network,
- $\overline{R_{b_j}}$: the average data rate for service *j*,
- $\overline{R_b^{max}}$: the maximum average data rate among all the network services (i.e., video streaming).

In (8), the W^{SRb} is a weight used to control the average session data rate upon allocation. The average data rate of services in the allocation procedure with higher values of this weight has relatively higher effect than the same situation with lower weights. Obviously, assigning zero to this weight, completely eliminates the average data rate effect and reduces the AP objective function addressed in (8) to cellular one presented in (7).

Fairness, is yet another objective in the allocation procedure. It is desired that services are served according to their serving weights where services with higher serving weights are to be allocated with a comparatively bigger portion of resources. It is neither fair nor desirable to not allocate (or allocate the minimum possible) capacity to services with lower serving weights. The ideal case is when the normalized data rate (i.e., data rate dived by the serving weight) of all the services, and consequently the normalized average, has the same value. This can be expressed as:

$$\frac{R_{b_{ji}\,[\text{Mbps}]}^{Srv}}{W_{ji}^{Srv}} - \frac{1}{N_{VNO} N_{srv}} \sum_{i=1}^{N_{VNO}} \sum_{j=1}^{N_{srv}} \frac{R_{b_{ji}\,[\text{Mbps}]}^{Srv}}{W_{ji}^{Usg}} = 0 \tag{9}$$

Nevertheless, the resource efficiency and the fair allocations are two conflicting goals. The increment of one of them leads to the decrement of the other one. Hence, instead of having the fairest allocation (i.e., the deviation of the normalized data rate from the normalized average is zero), the minimization of the total deviation from the normalized average is used, given by:

$$\min_{R_{b_{ji}}^{Srv}} \left\{ \sum_{i=1}^{N_{VNO}} \sum_{j=1}^{N_{srv}} \left| \frac{R_{b_{ji}\,[\mathrm{Mbps}]}^{Srv}}{W_{ji}^{Srv}} - \frac{1}{N_{VNO} N_{srv}} \sum_{i=1}^{N_{VNO}} \sum_{j=1}^{N_{srv}} \frac{R_{b_{ji}\,[\mathrm{Mbps}]}^{Srv}}{W_{ji}^{Srv}} \right| \right\} \tag{10}$$

where:

- $f_{\mathbf{R_b}}^{fr}$: fairness objective function,

It is worth noting that the fairness for services with minimum guaranteed data rates, applies only to the amount exceeding the minimum guaranteed level.

In order to achieve Linear Programming (LP), (10) has to change as follows:

$$\min_{R_{b_{ji}}^{Srv}, R_{b_{ji}}^{f}} \left\{ \sum_{i=1}^{N_{VNO}} \sum_{j=1}^{N_{srv}} R_{b_{ji}[\mathrm{Mbps}]}^{f} \right\}$$

$$s.t. \begin{cases} \dfrac{R_{b_{ji}\,[\mathrm{Mbps}]}^{Srv}}{W_{ji}^{Srv}} - \dfrac{1}{N_{VNO} N_{srv}} \displaystyle\sum_{i=1}^{N_{VNO}} \sum_{j=1}^{N_{srv}} \dfrac{R_{b_{ji}\,[\mathrm{Mbps}]}^{Srv}}{W_{ji}^{Srv}} \leq R_{b_{ji}[\mathrm{Mbps}]}^{f} \\ \dfrac{-R_{b_{ji}\,[\mathrm{Mbps}]}^{Srv}}{W_{ji}^{Srv}} + \dfrac{1}{N_{VNO} N_{srv}} \displaystyle\sum_{i=1}^{N_{VNO}} \sum_{j=1}^{N_{srv}} \dfrac{R_{b_{ji}\,[\mathrm{Mbps}]}^{Srv}}{W_{ji}^{Srv}} \leq R_{b_{ji}[\mathrm{Mbps}]}^{f} \end{cases} \tag{11}$$

where:

- $R_{b_{ji}}^{f}$ is an intermediate variable used to simplify the problem.

As the network capacity increases, the summation of weighted terms in (7) increases. Therefore, to combine it with (11), the two goals of optimization, the fairness intermediate variable, R_{ji}^{f} has to adopt to the network's capacity as in the objective function. This is given by:

$$f_{\mathbf{R_b}}^{fr}(\mathbf{R_b^f}) = \sum_{i=1}^{N_{VNO}} \sum_{j=1}^{N_{srv}} \left(\frac{R_{b\,[\mathrm{Mbps}]}^{CRRM}}{\overline{R_{b[\mathrm{Mbps}]}^{min}}} R_{b_{ji}[\mathrm{Mbps}]}^{f} \right) \tag{12}$$

where:

- $\overline{R_b^{min}}$ is minimum average data rate among all the network services (i.e., VoIP).

The division of network capacity by the minimum average data rate of services gives the maximum possible number of users in the network with given network capacity and service set. By multiplying the fairness variable by the maximum number of users, the balance of these two objectives (i.e., the network throughput and the fairness) can be achieved.

In addition, there are more constraints for VRRM to allocate data rates to various services, which should not be violated. The very fundamental constraint is the total network capacity estimated in last section. The sum of the entirely assigned data rates to all services should always be smaller than the total estimated capacity of the network given by:

$$\sum_{i=1}^{N_{VNO}} \sum_{j=1}^{N_{srv}} R_{b_{ji}\,[\mathrm{Mbps}]}^{Srv} \leq R_{b\,[\mathrm{Mbps}]}^{CRRM} \tag{13}$$

The offered data rate to the guaranteed and the best effort with minimum guaranteed services imposes the next constraints. The allocated data rate related to these services have to be higher than minimum guaranteed level (for guaranteed and best effort with minimum guaranteed) and lower than maximum guaranteed (for guaranteed services only):

$$R_{b_{ji}[\mathrm{Mbps}]}^{Min} \leq R_{b_{ji}\,[\mathrm{Mbps}]}^{Srv} \leq R_{b_{ji}[\mathrm{Mbps}]}^{Max} \tag{14}$$

where:

- $R_{b_{ji}}^{Min}$: the minimum guaranteed data rate for service *j* of the VNO *i*,
- $R_{b_{ji}}^{Max}$: the maximum guaranteed data rate for service *j* of the VNO *i*.

In (11), the allocated data rate for a specific service is defined as:

$$R_{b_{ji}\,[\mathrm{Mbps}]}^{Srv} = R_{b_{ji}\,[\mathrm{Mbps}]}^{cell} + R_{b_{ji}\,[\mathrm{Mbps}]}^{WLAN} \tag{15}$$

Due to changes in the physical infrastructure, users' channel, etc., in practice, there are situations in which the resources are not enough powerful to meet the guaranteed capacity and the allocation optimization, as addressed in previous sections, is no longer feasible. A simple approach in these cases is to relax the constraints by introduction of violation (also known as slack) variables. The new objective function contains the objective function of the original problem plus the penalty for violations. In the case of VRRM, the relaxed optimization problem can be considered by adding a violation parameter to (14) given by:

$$\begin{aligned} &R_{b_{ji}[\mathrm{Mbps}]}^{Min} \leq R_{b_{ji}\,[\mathrm{Mbps}]}^{Srv} + \Delta R_{b_{ji}\,[\mathrm{Mbps}]}^{v} \\ &\Delta R_{b_{ji}\,[\mathrm{Mbps}]}^{v} \geq 0 \end{aligned} \tag{16}$$

where

- $\Delta R_{b_{ji}}^{v}$: a violation variable for minimum guaranteed data rate for service *j* of the VNO *i*.

By introducing the violation parameter, the former infeasible optimization problem turns into a feasible one. The optimal solution maximizes the objective function and minimizes the average constraints violations. The average constraints violation is defined as follows:

$$\Delta\overline{R_{b}^{v}}_{[\mathrm{Mbps}]} = \frac{1}{N_{VNO} N_{srv}} \sum_{i=1}^{N_{VNO}} \sum_{j=1}^{N_{srv}} W_{ji}^{v}\, \Delta R_{b_{ji}\,[\mathrm{Mbps}]}^{v} \tag{17}$$

where

- $\Delta\overline{R_{b}^{v}}$: average constraint violation,

W_{ji}^{v}: weight of violating minimum guaranteed data rate service *j* of the VNO *i* where $W_{ji}^{v} \in [0,1]$.

The objective function presented in (6) also has to be altered. The new objective function, the relaxed one, has to contain the minimization of violations in addition to the maximization of former objectives. Although the average constraint violation has direct relation with the allocated data rate to services where the increment of one leads to decrement of the other one, it does not have the same relation with fairness. It can be claimed that the maximization of fairness and minimization of constraints violations are independent. In better words, the fairness variable is weighted as it is presented in (13) to compensate with summation of weighted data rate of various services. Therefore, minimizing the average constraint violation in addition to maximizing the fairness and the network's weighted throughput have to consider together as follows:

$$f_{\mathbf{R_b}}^{v}(\mathbf{R_b}) = W^{WLAN} f_{\mathbf{R_b}}^{WLAN}(\mathbf{R_b^{WLAN}}) + W^{cell} f_{\mathbf{R_b}}^{cell}(\mathbf{R_b^{cell}}) - f_{R_b^v}^{vi}(\Delta\overline{R_b^v}) - W^f f_{R_b}^{f}(\mathbf{R_b^f}) \tag{18}$$

where:

- $f_{R_b^v}^{vi}$: the constraint violation function:

$$f_{R_b^v}^{vi}(\Delta\overline{R_b^v}) = \frac{R_{b\,[\mathrm{Mbps}]}^{CRRM}}{R_{b[\mathrm{Mbps}]}^{min}} \Delta\overline{R_{b\,[\mathrm{Mbps}]}^{v}} \tag{19}$$

The definition of fairness, however, in a congestion situation is not the same. The fairness objective in the normal case is to have the same normalized data rate for all the services. As a reminder, when the network face against congestion, there are not enough resources to serve all the services with the minimum acceptable data rates. Therefore, not only all the best effort services are not allocated with any capacity, some violation is also introduced to the guaranteed data rates. Obviously, the violation data rates for the best effort services are always zero. Consequently, (11) is altered in the following way:

$$\begin{cases} \dfrac{W_{ji}^{v}\Delta R_{b_{ji}\,[\mathrm{Mbps}]}^{v}}{W_{ji}^{Usg}} - \dfrac{1}{N_{VNO}N_{srv}} \displaystyle\sum_{i=1}^{N_{VNO}} \sum_{j=1}^{N_{srv}} \dfrac{W_{ji}^{v}\Delta R_{b_{ji}\,[\mathrm{Mbps}]}^{v}}{W_{ji}^{Usg}} \leq R_{b_{ji}[\mathrm{Mbps}]}^{f} \\ -\dfrac{W_{ji}^{v}\Delta R_{b_{ji}\,[\mathrm{Mbps}]}^{v}}{W_{ji}^{Usg}} + \dfrac{1}{N_{VNO}N_{srv}} \displaystyle\sum_{i=1}^{N_{VNO}} \sum_{j=1}^{N_{srv}} \dfrac{W_{ji}^{v}\Delta R_{b_{ji}\,[\mathrm{Mbps}]}^{v}}{W_{ji}^{Usg}} \leq R_{b_{ji}[\mathrm{Mbps}]}^{f} \end{cases} \tag{20}$$

5.3.2 Partial VRRM

The former section presented the modelling of virtual radio resource management addressing the objectives in form of optimisation problem. Since the network status and constraints varies during the time, this problem is an uncertain non-convex optimisation problem. However, there are techniques to solve these kinds of problems; "Partial VRRM" among various approaches is one of the simplest one. In this approach, the optimisation problem is tackled by breaking the main problem into multiple sub-problems. In the case of VRRM, time axis is divided into decision windows and the objective function is maximised in each of these intervals, independently. However, it is worth noting that decisions in each interval affect directly the network state and the outcome of a policy at a certain point is dependent of the decisions and states in former intervals. Hence, the optimal solution has to take this dependency of decisions also into consideration. On the ground of this discussion, the output of partial VRRM may be only a local minimum and not the global one. Nevertheless, the partial VRRM is a simple solution, which can be used as the starting step and reference point. Outcome of the more sophisticated approaches, then, can be compared to these results.

Figure 43 illustrates the decision window of VRRM, CRRM, and LRRMs. The VRRM decision window contains multiple CRRM ones during which CRRM applies the decided policy set. On the next decision window of VRRM, after multiple network stages, the VRRM updates network situation and makes the new decision for the next time interval.

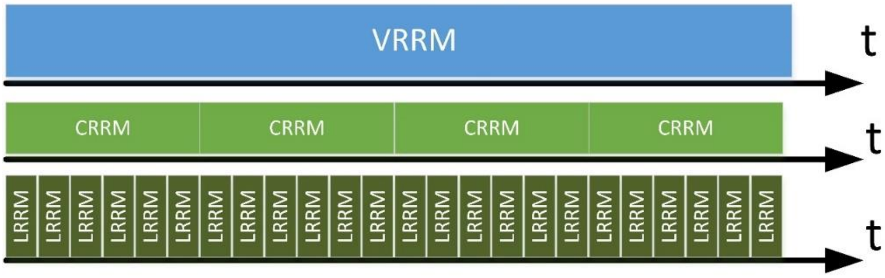


Figure 43 – Decision window of VRRM and CRRM.

Consider the objective function presented in previous sections, it can be considered as summation of objective functions over time intervals in partial VRRM as follows:

$$f^{v}_{\mathbf{R_b}}(\mathbf{R_b}) = \sum_{t_i=0}^{n_t} f^{v}_{\mathbf{R_b}}(\mathbf{R_b}[t_i]) \tag{21}$$

where:

- n_t: number decision window,
- $\mathbf{R_b}[t_i]$: allocated data rate vector at time interval t_i given by:
- $R^{Srv}_{b_{ji}}[t_i]$: Served (allocated) data rate for service *j* of the VNO *i* at the time frame t_i,

The goal is to maximise the objective function for all running time. Considering objective function in each time interval independent of the other time intervals, it can be written:

$$\max_{\mathbf{R_b}} f^{v}_{\mathbf{R_b}}(\mathbf{R_b}) \Leftrightarrow \sum_{t_i=0}^{n} \max_{\mathbf{R_b}[t_i]} f^{v}_{\mathbf{R_b}}(\mathbf{R_b}[t_i]) \tag{22}$$

As the VRRM starts to work, it has to gather information about serving VNOs plus their services and SLAs. This stage is referred as “Network Initiation” in the Figure 44. Estimating total systems (V-RAN) capacity is the next step. VRRM uses the information about number of available RRUs in different RATs and estimate the PDF of (5). The allocation of the available capacity to VNOs services is done by solving Linear Programming (LP) problem. In each decision window, the network status has to be update. The network status at time interval t_i contains the following items:

- Remained network capacity, $R^{CRRM}_{b\,[\text{Mbps}]}[t_i]$, which is the data rate achievable from unassigned RRUs.
- Freed service data rate, $R^{fr}_{b_{ji}\,[\text{Mbps}]}[t_i]$, which is referred to RRUs freed by termination of service *j* of VNO *i* in time interval t_i. It should be reminded that this value is considered from the estimation of data rate based on number of freed RRUs. The actual data rate offered to the subscribers may not be the same value.
- On use service data rate, $R^{Us}_{b_{ji}\,[\text{Mbps}]}[t_i]$, which is referred to RRUs exhausted by service *j* of VNO *i* in time interval t_i. Like the freed service data rate, this value also is achieved based on the estimation of data rate of used RRUs.
- Actual freed service data rate, $R^{Afr}_{b_{ji}\,[\text{Mbps}]}[t_i]$, which is referred to actual data rate of service *j* of VNO *i* terminated in this time interval.

- Actual on used service data rate, $R^{AUs}_{b_{ji}\,[\mathrm{Mbps}]}[t_i]$, which is the actual data rate exhausted by ongoing service *j* of VNO *i*.

According to these definition the allocated data rate to for each service in each decision is:

$$R^{Srv}_{b_{ji}}[t_{i+1}] = R^{Srv}_{b_{ji}}[t_i] + R^{fr}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] - R^{Us}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] \tag{23}$$

Using the updated information of network status, the new network capacity in time interval *i*+1 is defined as follows:

$$R^{CRRM}_{b\,[\mathrm{Mbps}]}[t_{i+1}] = R^{CRRM}_{b\,[\mathrm{Mbps}]}[t_i] + \sum_{i=1}^{N_{VNO}} \sum_{j=1}^{N_{srv}} \left(R^{fr}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] - R^{Us}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] \right) \tag{24}$$

Relatively, the minimum and maximum guaranteed data rate on update as follows:

$$R^{Min}_{b_{ji}[\mathrm{Mbps}]}[t_{i+1}] = R^{Min}_{b_{ji}[\mathrm{Mbps}]}[t_i] + R^{Afr}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] - R^{AUs}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] \geq 0 \tag{25}$$

$$R^{Max}_{b_{ji}[\mathrm{Mbps}]}[t_{i+1}] = R^{Max}_{b_{ji}[\mathrm{Mbps}]}[t_i] + R^{Afr}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] - R^{AUs}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] \geq 0 \tag{26}$$

The maximum and minimum guaranteed data rate, obviously, cannot be less than zero. A service with zero (or negative) minimum guaranteed data is going to be served in best effort manner and the one with zero maximum guaranteed data is no longer going to be served. Using this technique, VRRM algorithm repeatedly optimises the objective function presented in (6) as follows:

$$\max_{\mathbf{R}^{\mathrm{cell}}_{\mathbf{b}}[t_i],\mathbf{R}^{\mathbf{WLAN}}_{\mathbf{b}}[t_i],\mathbf{R}^{f}_{\mathbf{b}}[t_i]} \left\{ W^{WLAN} WLAN\left(\mathbf{R}^{\mathbf{WLAN}}_{\mathbf{b}}[t_i]\right) + W^{cell} f^{cell}_{\mathbf{R_b}}\left(\mathbf{R}^{\mathbf{cell}}_{\mathbf{b}}[t_i]\right) - W^{f} f^{f}_{R_b}\left(\mathbf{R}^{\mathbf{f}}_{\mathbf{b}}[t_i]\right) \right\}$$

$$s.t.\begin{cases} R^{Srv}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] = R^{cell}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] + R^{WLAN}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] \\ 1^{T}\mathbf{R_b}[t_i] \leq R^{CRRM}_{b\,[\mathrm{Mbps}]}[t_i] \\ R^{Min}_{b_{ji}[\mathrm{Mbps}]}[t_i] \leq R^{Srv}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] \\ R^{Srv}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] \leq R^{Max}_{b_{ji}[\mathrm{Mbps}]}[t_i] \\ \dfrac{R^{Srv}_{b_{ji}\,[\mathrm{Mbps}]}[t_i]}{W^{Srv}_{ji}} - \dfrac{1}{N_{VNO}N_{srv}} \displaystyle\sum_{i=1}^{N_{VNO}} \sum_{j=1}^{N_{srv}} \dfrac{R^{Srv}_{b_{ji}\,[\mathrm{Mbps}]}[t_i]}{W^{Srv}_{ji}} \leq R^{f}_{b_{ji}[\mathrm{Mbps}]}[t_i] \\ \dfrac{-R^{Srv}_{b_{ji}\,[\mathrm{Mbps}]}[t_i]}{W^{Srv}_{ji}} + \dfrac{1}{N_{VNO}N_{srv}} \displaystyle\sum_{i=1}^{N_{VNO}} \sum_{j=1}^{N_{srv}} \dfrac{R^{Srv}_{b_{ji}\,[\mathrm{Mbps}]}[t_i]}{W^{Srv}_{ji}} \leq R^{f}_{b_{ji}[\mathrm{Mbps}]}[t_i] \end{cases} \tag{27}$$

For the resource shortage situations, the objective function in the optimisation is:

$$\begin{aligned} \max_{\mathbf{R}^{\mathrm{cell}}_{\mathbf{b}}[t_i],\mathbf{R}^{\mathbf{WLAN}}_{\mathbf{b}}[t_i],\mathbf{R}^{f}_{\mathbf{b}}[t_i],\Delta\mathbf{R}^{\mathbf{v}}_{\mathbf{b}}[t_i]} & \left\{ W^{WLAN} f^{WLAN}_{\mathbf{R_b}}\left(\mathbf{R}^{\mathbf{WLAN}}_{\mathbf{b}}[t_i]\right) + W^{cell} f^{cell}_{\mathbf{R_b}}\left(\mathbf{R}^{\mathbf{cell}}_{\mathbf{b}}[t_i]\right) - f^{vi}_{R^{v}_{b}}\left(\Delta \overline{R^{v}_{b}}\right) \right. \\ & \left. - W^{f} f^{f}_{R_b}\left(\mathbf{R}^{\boldsymbol{f}}_{\mathbf{b}}[t_i]\right) \right\} \end{aligned} \tag{28}$$

$$s.t: \begin{cases} R^{Srv}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] = R^{cell}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] + R^{AP}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] \\ 1^T \boldsymbol{R_b}[t_i] \leq R^{CRRM}_{b\,[\mathrm{Mbps}]}[t_i] \\ R^{Min}_{b_{ji}[\mathrm{Mbps}]}[t_i] \leq R^{Srv}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] \\ R^{Srv}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] \leq R^{Max}_{b_{ji}[\mathrm{Mbps}]}[t_i] \\ 0 \leq \Delta R^{v}_{b_{ji}} \\ W^{v}_{ji} \Delta R^{v}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] - \frac{1}{N_{VNO} N_{srv}} \sum_{i=1}^{N_{VNO}} \sum_{j=1}^{N_{srv}} W^{v}_{ji} \Delta R^{v}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] \leq R^{f}_{b_{ji}[\mathrm{Mbps}]}[t_i] \\ -W^{v}_{ji} \Delta R^{v}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] + \frac{1}{N_{VNO} N_{srv}} \sum_{i=1}^{N_{VNO}} \sum_{j=1}^{N_{srv}} W^{v}_{ji} \Delta R^{v}_{b_{ji}\,[\mathrm{Mbps}]}[t_i] \leq R^{f}_{b_{ji}[\mathrm{Mbps}]}[t_i] \end{cases}$$

In the provisioning phase, system is in initial status and VRRM outputs leads to set of policies for system runtime. In this situation, VRRM divides the total network capacity among the services using (6) when all RRUs are available and there is no -used data rate. As the requests arrive and network starts to work, system states (e.g. remaining RRUs and served data rate) will change according to (24), (25), (26), and (27). The algorithms update the network capacity estimation and guaranteed data rates. Then it solves again the optimisation problem of (6). The outcome is sent to CRRM, as a policy update. Figure 44 presents partial VRRM flowchart in provisioning and runtime of virtual RAN.

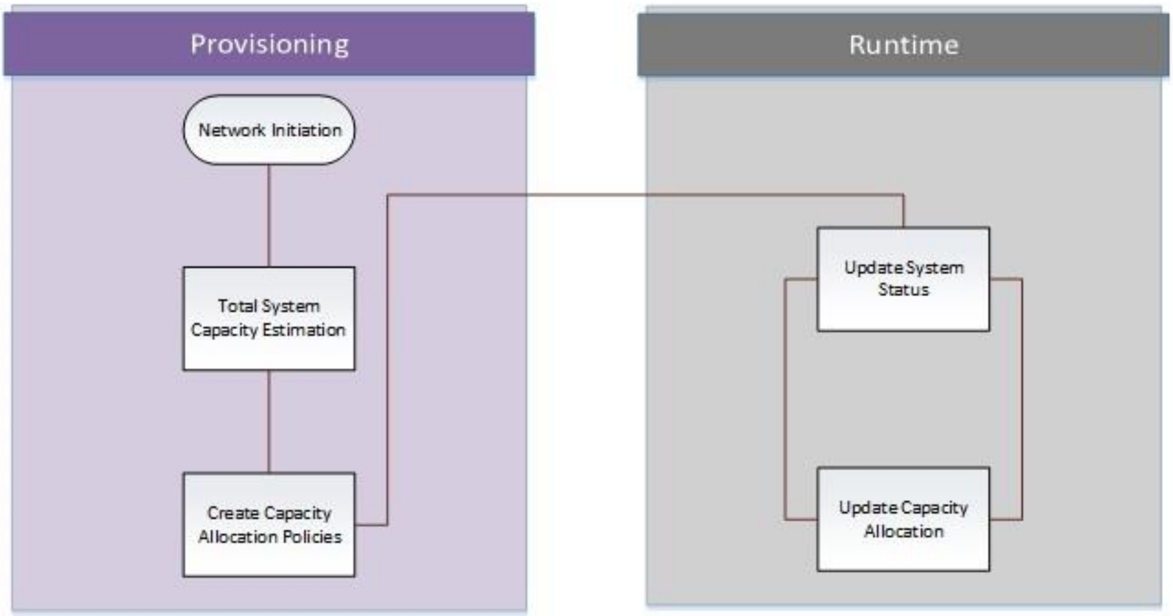


Figure 44 – Partial VRRM flowchart.

5.3.3 Reference Scenarios – Urban Hotspot

The performance of VRRM is simulated and evaluated in an urban hotspot scenario. This section describes this reference scenario, afterwards other variation of scenarios (e.g., different number of user) are considered based on the reference one. The key parts of this scenario are RATs, subscribers, services, and VNOs.

Regarding to the RATs, both cellular RAN and Wi-Fi APs are considered in this scenario. The RAN is based on a set of RRHs capable of supporting multiple RATs, which are OFDMA (e.g. LTE), CDMA (e.g. UMTS), and FDMA/TDMA (e.g. GSM). Although flexibility of these RRHs offers various cell layouts, the configuration illustrated in Figure 45, is used.

- The OFDMA RAT assumption is based on the 100 MHz LTE-Advanced. The cells of this RAT are the smallest ones with the radius of 400 meters. Each cell has 500 RRUs, which can be assigned for traffic bearers.
- The configurations for CDMA cells are chosen according to UMTS (HSPA+) working on 2.1 GHz. Each cell of this RAT with the radius of 1.2 km has 3 carriers and each carrier has 16 codes. Only 45 codes out of all 48 codes in each cell can be assigned to users' traffic.
- The biggest cell size with the radius of 1.6 km is configured for FDMA/TDMA. Based on GSM900, each cell has 10 frequency channels and each channel has 8 timeslots. It is assumed that 75 timeslots out total 80 available timeslots in each cell can be used for to users' traffic.

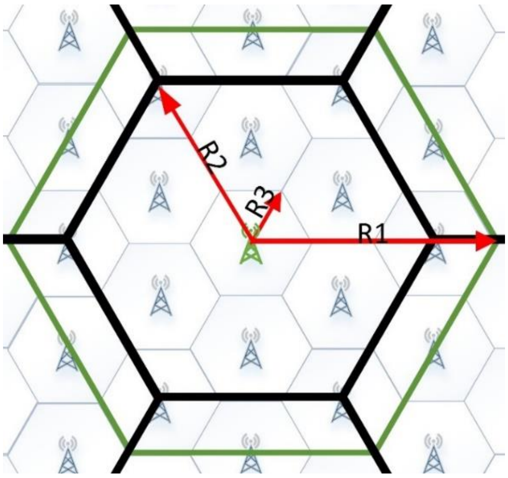


Figure 45 – Network Cell Layout (R1=1.6 km, R2=1.2 km, R3=0.4km)

In addition to cellular access network, in this scenario full coverage of Wi-Fi (OFDM) is provided by means of IEEE802.11ac standard APs, configured to work 80 MHz channel bandwidth. It is assumed that each access point covers a cell with radius 80 meters and they are facilitated with beamforming and MU-MIMO technology to support up to eight spatial streaming. Due to European Union rules and regulations, there are only five available channels for 80 MHz APs [Cisc12]. In contrast to former RATs, the Wi-Fi APs use the same set of links for upload and download streams. To achieve coherency among various RATs, the total throughput of APs is equally divided between downlink and uplink. Therefore, in Table 10, where coverage information is summarised, the number of RRUs in each Wi-Fi cell is indicated as half of total number of available channels. This table also presents the maximum data rate for each RAT in downlink. In this scenario, it is assumed the APs are only deployed on the OFDM base station and there is not full coverage.

Table 10 – Different RAT cell radius (based on [Cisc12]).

RAT	Number Cells	Cell Radius [km]	System	Downlink			
				Number of RRU/Cell	$N_{RRU}^{RAT_i}$	$R_{b_{RAT_i[\text{Mbps}]}}^{max}$	$R_{b_{tot[\text{Gbps}]}}^{RAT_i}$
OFDM	16	0.08	Wi-Fi	2.5	40	1300.0	50.78
OFDMA	16	0.4	LTE	500	8000	0.7	5.47
CDMA	1.7	1.2	UMTS	45	80	43.0	3.36
TDMA	1	1.6	GSM	75	75	0.05833	4.37

The terminals of subscribers can be smartphone, tablets, laptops, and M2M devices. According to [ERIC2013] and [Cisc13b], the average subscribers' traffic is related to the type of terminals they use as it is presented in Table 11. Using this table, the VNOs' contracted capacity can be estimated based on number of subscribers. In this reference scenario, it is assumed that all the UEs are smart phones. In other variation of reference scenario, other mixture of user terminal is going to be considered.

Table 11 – Average Mobile Network Connection Speed (extracted from [Cisc13a])

Terminal	Average Connection Speed [Mbps]
Smartphones	6.375
Tablets	11.39
Laptops	14
M2M terminals	1

The service set in this scenario is assumed based on Figure 46. The volume share of services, which is presented in outer circle in the figure, is the percentage of the specific service traffic volume from total operator's traffic. The service penetration, however, is the percentage of active subscribers using that service. This charts clearly present that the video streaming holds the highest volume whereas VoIP is the most requested service. These charts may be altered when different type of terminals mixture (and consequently service profile) is chosen. Table 12 presents more details on these services and their QoS requirements.

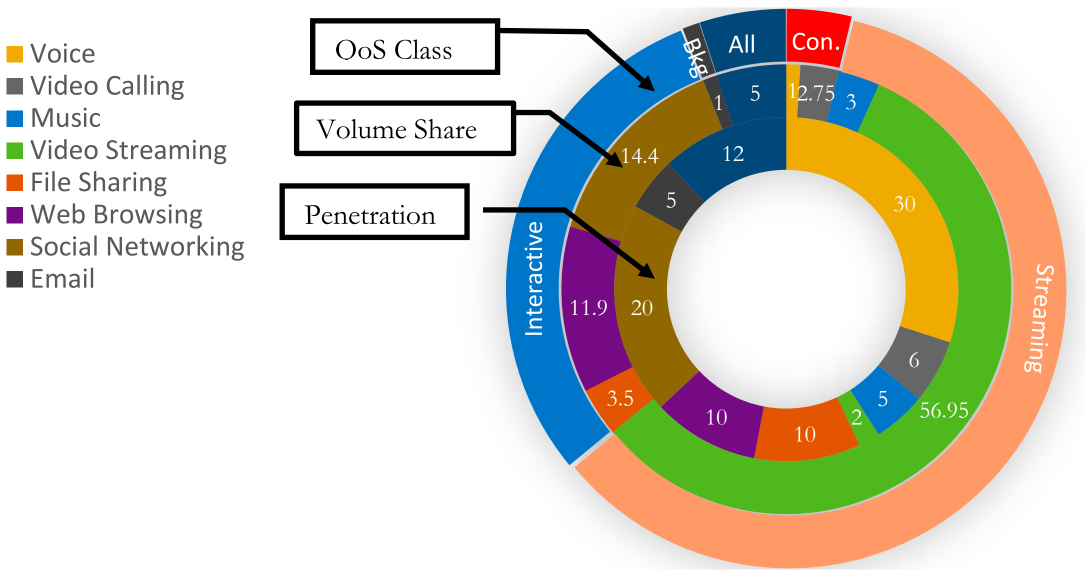


Figure 46 – Various service volume share and penetration.

Three VNOs are assumed to operate in this area, VNO GB, BG, and BE. All of these VNOs offer the same set of services to their users. These services and their volume share of an operator traffic are listed in Table 12. The order of service weights based on general service classes are conversational (0.4), streaming (0.3), interactive best effort (0.2), and best effort (0.05). To not compromise the objective function for having higher fairness, the fairness weight, W^f, is heuristically chosen to be equal to the lowest serving weight (0.05).

Table 12 – Services characteristics for multi-service VNOs based on [Robe04].

Service	Service Class	Max Data Rate [kbps]			Size[kB]
		Min.	Average	Max.	
Voice	Conversational	5.3	12.2	64	–

Music		Streaming	16	64	160	–
File Sharing		Interactive	384	1024	–	2042
Web Browsing		Interactive	30.5	500	–	180
Social Networking		Interactive	24	384	–	45
Email		Background	10	100	–	300
M2M	Smart Meters	Background	–	200	–	2.5
	e-Health	Interactive	–	200	–	5611.52
	ITS	Conversational	–	200	–	0.06
	Surveillance	Streaming	64	200	384	5.5
Mobile Video	Video Calling	Conversational	64	384	2048	–
	Video streaming	Streaming	500	5120	13000	–

Each VNO have 300 subscribers where the average data rate for each of them is 6.375 Mbps [Cisc13c]. Hence, the contracted data rate for whole operator is 1912.5 Mbps and each service receives a portion based on the volume percentage in Table 12. The SLAs of these VNOs are summarised as follows.

- VNO GB, the allocated data rates for services are guaranteed to be between 50% and 100% of the service data rate.
- VNO BG, has best effort with minimum 25% of service data rate guaranteed SLA.
- VNO BE, is served in best effort manner.

5.3.4 Results

To estimate total data rate of the network described in last section, PDF of different RATs as it is presented in (5) and (6), is calculated. By means of implementation of numeric convolution, the PDF and CDF of data rate for total network are achieved. Besides, to have a comparison between virtualisation of radio resources with RAN sharing concept, CDF of total network for RAN sharing and V-RAN approach is illustrated in Figure 47.

In the V-RAN, all resources are aggregated where in the RAN sharing approach, each operator has one third of resources and total network data rate is three times that of a single operator. It can be seen that it is more probable to serve all the three operators when the resources are aggregated than when the resources are divided. For instance, for 50% of time (when CDF is equal to 0.5) the total network capacity in V-RAN is 1.76 Gbps where RAN sharing offers 1.74 Gbps. The highest difference can be seen where CDF is equal to 0.1. In this case, the relative data rate for V-RAN is 1.68 Gbps where RAN sharing offers only 1656 Mbps.

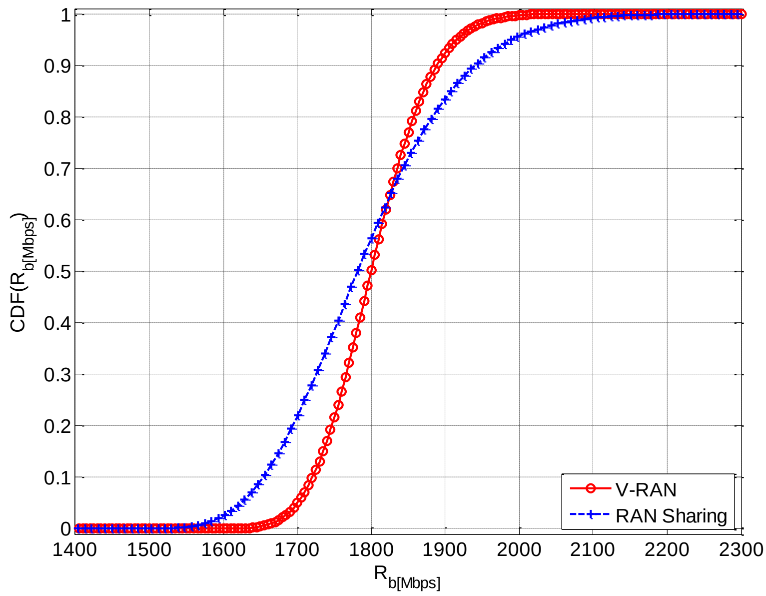


Figure 47 – CDF of network capacity for V-RAN and RAN sharing

In the next step, allocation of resources is done for the case where CDF is equal to 0.5. The VNO GB received the biggest portion (59%) of resources since it is the VNO with the guaranteed SLA.VNO BE, the VNO to be best effort manner, is allocated only 7% of capacity. The rest of resources (34%) are assigned to VNO BG so that it is served better than the minimum guaranteed.

The optimisation problem stated in (27), when there is only cellular network available, is solved by MATLAB linear programming problem solver (i.e., linprog function) [Math14]. The results for various services of different VNOs are listed in Table 13. As it can be seen, VRRM manages to meet the guaranteed level of services. Based on serving weight, as it was expected, the conversational services (e.g. VoIP and Video call) have the highest data rate while background services (e.g. Email and Smart Meter) have the lowest data rate. The M2M services present this issue better since they all have the same volume percentage but different serving weight. Services of VNO GB and BG have minimum guaranteed level so the assigned data rates cannot go below a certain level. This is the reason for having high data rate assigned to video streaming in these two VNOs. In general, it can easily be seen that the same service in the VNO GB and BG received more capacity in comparison to VNO BE.

Table 13 – Data rate allocated to each service

Services	$R_{b_{ji}}^{Srv}$ [Mbps]		
	VNO GB	VNO BG	VNO BE
VoIP	19.12	21.43	16.65
Music	40.30	25.96	11.62
File Sharing	41.21	24.48	7.74
Web Browsing	121.54	64.64	7.74
Social Networking	145.44	93.33	7.74
Email	11. 50	6.72	1.94
M2M-SM	15.08	8.51	1.94
M2M-eH	20.89	14.32	7.74
M2M-ITS	26.30	23.22	16.65
M2M-SV	24.76	18.12	11.62

Video Streaming	556.24	283.92	11.62
Video Call	43.08	29.78	16.65

Considering the traffic offloading, the capacity of cellular network is computed using (5). Then, the same method is used to obtain Cumulative Distribution Function (CDF) of the extension of capacity by implementing APs. The results are plotted in Figure 48 for three common path loss exponents. It is obvious that increase of attenuation when the collisions effects are neglected leads to increment of capacity. According to the numeric results depicted in Figure 47, a capacity of 1.76 Gbps is achieved in 50% of the time, for a network without WLAN. When considering WLAN, the capacity is increased up to 6.8 Gbps, when $\alpha_p$is 3.8.

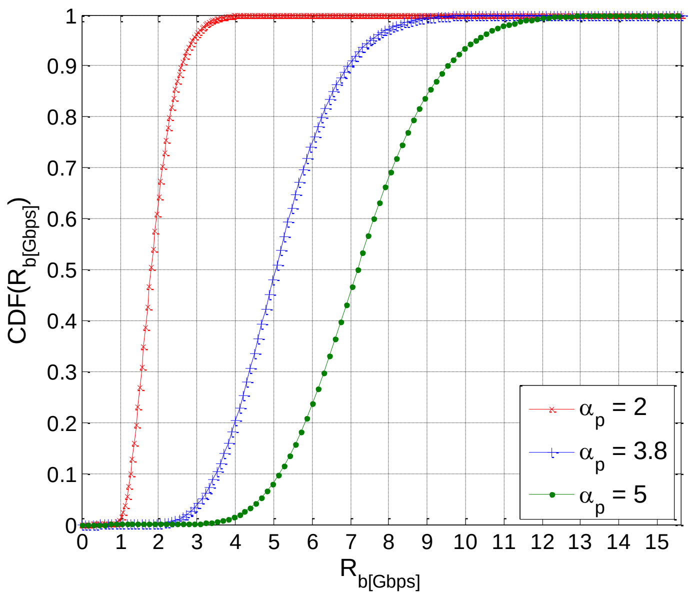


Figure 48 – CDF of offered capacity by APs.

In the offloading scenario, the number of subscribers is increased to 500 per VNO. Table 14 presents the allocated data rate to each service from cellular network and APs. According to this table, conversational services (VoIP, video call, and M2M – ITS), which are services with the highest serving weights received relatively the highest data rates too. Streaming class including Music, M2M – SV, and video streaming are placed in the second order. The lowest served services are the services of background class including email and M2M – SM. In addition, by considering the capacities from cellular network and APs, it is obvious that the services with the bigger average data rate are allocated more from APs comparing to cellular network and vice versa. For instance, video streaming with the highest data rate is allocates about 6.5 times more than the cellular ones. One the other hand, email with a low data rate is allocated more from cellular network. However, VoIP is not following the same rule since it has very high serving weight. So it has been allocated comparatively with high data rates from both networks. But the capacity allocated to the video streaming, the service with high data rate per session, is mainly from Wi-Fi APs.

Table 14 – Allocated capacity to the services.

Service	$R_{b[\text{Mbps}]}^{cell}$	$R_{b[\text{Mbps}]}^{WLAN}$	**Total**	**Total Allocated Data Rate [Mbps]**		
				VNO GB	**VNO BG**	**VNO BE**
VoIP	170.44	613.58	784.02	31.87	393.43	358.72
Music	148.30	257.60	405.90	95.62	167.09	143.18

FTP	136.35	194.01	330.37	111.56	123.35	95.46
Web Browsing	166.83	404.02	570.85	285.11	190.28	95.46
Social Networking	175.35	483.16	658.51	324.96	238.10	95.46
Email	54.15	33.42	87.57	31.88	31.38	23.86
M2M - SM	61.18	41.34	102.51	43.83	34.82	23.86
M2M - her	112.64	133.06	245.70	43.83	106.41	95.46
M2M - ITS	176.48	622.19	798.67	43.83	396.12	358.72
M2M - SV	129.44	211.71	341.15	43.83	154.14	143.18
Video Streaming	276.19	1514.91	1791.11	1050.89	597.04	143.18
Video Call	192.27	659.99	852.26	87.54	406.00	358.72
Total	1799.62	5169.00	6968.62	2194.74	2838.61	1935.27

In the same table, the data rates are addressed for each service of each VNO. It is apparent from this table that most of services of VNO GB are served with the maximum guaranteed data rate. For instance, VoIP and Music in this service is guaranteed to be served with relatively maximum of 31.87 Mbps and 95.62 Mbps and they have been allocated the same amount. The main difference between this VNO and VNO BG is the service maximum data rate. Since the services of VNO BG have no upper limit, while the VNO GB ones have, the total allocated data rate to the former one is higher than the latter one. Obviously, the offered capacity to VNO BG in resource shortage situation is going to be smaller than VNO GB.

6 Conclusions and Further Work

This Deliverable 4.3 (D4.3) described the algorithms and mechanisms used under the scope of services EPCaaS, ANDSFaaS, MOBaaS and RANaaS. Those mechanisms and algorithms rule the way the management of those services will be performed, especially considering how services are deployed and managed during runtime, taking advantage of the cloud principles, increasing efficiency and reducing costs. The implementation, to be documented in D4.4, will consider those algorithms as the basis.

Section 2 described several network function placement algorithms and mechanism, both for initial placements as well as for runtime migrations. Different strategies were approached to manage the EPC and ANDSF services, providing algorithm to perform an efficient use of resources, by growing and shrinking resources when necessary (elasticity). For the particular case of the EPC initial placements, a detailed cost analysis was performed, providing important clues about the rules that an EPC placement should follow. For the runtime management case, the placement on highly distributed clouds was analysed, while it was performed an evaluation of the state and signalling reduction for SDN-based EPC procedures, devising a new model, more efficient, to be addressed.

Section 3 proposed algorithms that can be used in the MOBaaS to estimate the next location of an individual/a group of end-users and the bandwidth used/available at a certain location in a future moment of time. The performance of each prediction algorithm and the precision of the derived data, are completely affected by a specific series of data that it gets as input data. In this deliverable, the proposed approaches are based on the information and the monitored metrics that could/will be provided, so far, by the MaaS as the inputs. However, the MOBaaS's software architecture has been designed in a modular way and based on the new inputs that may be available or will be provided by MaaS in future steps, it would be possible to upgrade/replace the new designed solutions in the prediction algorithm parts in MOBaaS. Because currently the chain of MCN components are completed and not integrated yet and MaaS cannot provide the required input information for the prediction algorithms, we use the available offline traced data to evaluate the proposed approaches. In the future steps the algorithms will be tested and evaluated with the real monitored data by MaaS.

Section 4 introduced a SDN/OpenFlow based DMM approach, denoted as Partial OpenFlow based DMM that can be applied in virtualized LTE systems. In particular, it mainly studied whether the proposed DMM solution is able to support the continuity of a session that is handed over from a source P-GW to a target P-GW in a seamless way. The simulation experiments show that when the X2 path is used for support of the inter P-GW handover, the sessions can be handed over seamlessly to the target P-GW, by using the proposed approach, for a duration less than 150 ms. The best results in terms of latency in configuring traffic redirection in the DMM transport network were obtained when the OC has been positioned in the middle of the core network and the distance between Ingress and Egress OpenFlow switches was lower than or equal to 4 hops. The experiments also show that, in the case of migration of the virtual P-GW from one source datacentre to a target datacentre, a function to predict the mobility of UEs need to be present in the network in order to setup DMM traffic redirection prior to trigger of the handover procedure. Implementation a prototype of the proposed architecture in OpenStack virtualization test bed as a supplementary validation will be our next research in this part.

In Section 5, several mechanisms and algorithms are evaluated for RAN. Firstly, important issues related to the LTE RAN "cloudification" were studied. The Open Air Interface LTE base station emulator was used as a benchmarking tool on different machine environments (dedicated GPP, KVM, and Cloud) and several conclusions were drawn related to the proper system bandwidth and MCS to be used for a specific Intel CPU. The processing time as a function of CPU frequency was modelled. It was found the minimum required Intel CPU frequency to support the worst-case scenario for HARQ process to 4 GHz. Secondly, a BBU pool dimensioning example based on a real network configuration in French Brittany region was provided. Optimal BBU hotel locations were identified considering existing fixed infrastructure central offices. Fronthaul requirements were also considered. Under these assumptions, it was concluded that more than 90% of the 900 multi-RAT antenna sites are covered by 14 BBU pool locations. Finally, as a third mechanism, and as a solution to extreme mobile data demand, it was proposed to use traffic offloading and virtualisation of radio resources together. By means of virtualisation of radio resources, it is possible to share the physical infrastructure either cellular or WLANs among multiple VNOs. The performance of the proposed model was evaluated for a practical scenario. By adding an AP to each OFDMA cell, the network capacity increased up to 2.8 times. Three VNOs with three different SLAs were considered in the scenario.

In the next months, it is expected a significant evolution in the implementation issues, completing the work that has been initiated in D2.2 [MCN-D2.2], D3.1 [MCN-D3.1], and D4.1 [MCN-D4.1]. Deliverables D3.4/D4.4, scheduled for M26, will describe the final software components and evaluate (briefly) the performance. Further D6.3 and D6.4, to be released until the end of the project, will complement this work with a careful evaluation of those services, as well as all aspects related to the integration with other services, in order to ultimately build the so-called e2e (end-to-end) services.

In short, this D4.3 has just provided the basis for the further stages of development, according to the plans. So, for this point of view, we are on the right track to achieve the challenging objectives that we proposed ourselves in the beginning of the project, which represent a significant move forward to what is the state-of-the-art in this area today.

References

(by order of appearance)

[MCN-D2.2] MCN D2.2, "Overall Architecture Definition, Release 1", European Commission, EU FP7 Mobile Cloud Networking public deliverable, "First Mobile Network Cloud Software Components" 2013;

[MCN-D3.1] MCN D3.1, "Infrastructure Management Foundations – Specifications & Design for Mobile Cloud framework", European Commission, EU FP7 Mobile Cloud Networking public deliverable, November 2013;

[MCN-D3.2] MCN D3.2, "Infrastructure Management Foundations – Components First Release", European Commission, EU FP7 Mobile Cloud Networking public deliverable, November 2013;

[MCN-D4.1] MCN D4.1, "Mobile Network Cloud Component Design", European Commission, EU FP7 Mobile Cloud Networking public deliverable, November 2013;

[MCN-D4.2] MCN D4.2, "First Mobile Network Cloud Software Components", European Commission, EU FP7 Mobile Cloud Networking public deliverable, April 2014;

[D-Nowoswiat] David Nowoswiat, "Managing LTE Core Network Signaling Traffic". Internet: http://www2.alcatel-lucent.com/techzine/managing-lte-core-networksignaling-traffic/, [Sept. 26, 2014]

[TE-524262] White Paper TE-524262, "NEC Virtualized Evolved Packet Core - vEPC". Internet: http://www.nec.com/en/global/solutions/nsp/nfv/doc/vEPC_WP.pdf, [Oct. 09, 2014]

[Cisco-DC] "Cisco datacenter infrastructure 2.5."

[Akyi04] Akyildiz, I.F., Xie, J., Mohanty, S. "A Survey of Mobility Management in NextGeneration All-IP-Based Wireless Systems", IEEE Wireless Communications, 16–28 , 2004.

[Valk99] Valko, A.G., "Cellular IP, A New Approach to Internet Host Mobility", ACM SIGCOMM Computer Communication Review, 1999.

[John04] Johnson, D., Perkins, C., Arkko, J., "Mobility Support for IPv6" , IETF RFC 3775 , 2004.

[Gund08] Gundavelli, S., Chowdhury, K., Devarapalli, V., Patil, B., Leung, K., et al., "Proxy Mobile IPv6", IETF RFC 5213, June 2008.

[Rieg07] Riegel, M., Tuexen, M., "Mobile SCTP", IETF draftriegeltuexen-mobile-sctp-09.txt, 2007.

[Rose02] Rosemberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Sparks, R., Handley,A., Schooler, E., "SIP: Session Initiation Protocol", IETF RFC 3261, 2002.

[Gior05] Giordano, S., Lenzarini, D., Puiatti, A., Vanini, S., "WiSwitch: Seamless Handover between Multi-provider Networks", Proceedings of Second Annual Conference on Wireless On-demand Network Systems and Services (WONS 2005), pp. 224–235, 2005.

[3GPP-Release-8] 3GPP Technical Specification 29.060, "General Packet Radio Service (GPRS); GPRS Tunnelling Protocol (GTP) across the Gn and Gp interface ", Release 8, http://www.3gpp.org.

[John04] Johnson, D., Perkins, C., Arkko, J., et al.," Mobility support in IPv6", IETF RFC 3775 , June 2004.

[BakP95] Bak, P., and Paczuski, M., "Complexity, contingency, and criticality". Proc Natl Acad Sci U S A. 92 (15): 6689–6696. 1995;

[Bara08] González, M.C., Hidalgo, C.A., Barabasi, A.-L.: Understanding individual human mobility patterns. Nature Journal 435 (June 2008), http://www.nature.com

[Bara10] Song, C., Qu, Z., Blumm, N., Barabasi, A.-L.: Limits of Predictability in Human Mobility. Science Maganzie 327 (February 2010), http://www.sciencemag.org

[Cast96] John L. Casti, “Would-Be Worlds: How the New Science of Simulation Is Breaking the Complexity Barrier: How Simulation Is Changing the Frontiers of Science”, September 1996;

[Bert09] Bertin, P., Bonjour, S., Bonnin, J.-M., “Distributed or centralized mobility?”, IEEE Global Telecommunications Conference, GLOBECOM 2009, p. 16., 2009.

[Chan11] Chan, H.A., Yokota, H., Xie, J., Seite, P., Liu, D., “Distributed and Dynamic Mobility Management in Mobile Internet”, Current Approaches and Issues. Journal of Communications 6, 2011.

[Boko11] Bokor, L., Faigl, Z., Imre, S., “Flat architectures: Towards scalable future internet mobility”, Domingue, J., et al. (eds.) Future Internet Assembly,. LNCS, vol. 6656, pp. 35–50. Springer, Heidelberg , 2011.

[KariCLOSER14] Karimzadeh, M., Valtulina, L., Karagiannis, G., “Applying SDN/OpenFlow in Virtualized LTE to Support Distributed Mobility Management(DMM)”, 4th International Conference on Cloud Computing and Services Science , 2014.

[ONF13] “ONF official website”, https://www.opennetworking.org/ ,visited in December 2013.

[OpenFlow 1.3.0] “The OpenFlow Switch Specification”, Version 1.3.0, http://archive.openflow.org,visited in December 2013.

[KariAIMS2014] Karimzadeh, M., Sperotto, A., Pras, A., ”Software Defined Networking to Improve. Mobility Management Performance”, Autonomous Infrastructure, Management and Security, AIMS 2014.

[Mark14] M., Liebsch, P., Seite, G., Karagiannis, S., Gundavelli, "Distributed Mobility Management- Framework & Analysis", IETF Internet draft (work in progress), February 2014.

[Valt13] L. Valtulina, "Seamless Distributed Mobility Management (DMM) Solution In Cloud Based LTE Systems", M.Sc thesis, University of Twente, the Netherlands, November 2013, (visited in March 2014), http://www.utwente.nl/ewi/dacs/assignments/completed/master/reports/report-luca-valtulina.pdf

[SANS] “Latency and QoS for Voice over IP - SANS Institute”, http://www.sans .org/reading-room/whitepapers/voip/latency-qos-voiceip-1349?show=latency-qos-voice-ip-1349&cat=voip.

[Khan09] Farooq Khan, "LTE for 4G mobile broadband: air interface technologies and performance", Cambridge University Press, 2009.

[Amma11] D. Ammar, T. Begin, I. Guerin-Lassous, “A new tool for generating realistic internet traffic in ns-3”, Proceedings of the 4th International ICST Conference on Simulation Tools and Techniques, pp. 81–83. 2011.

[Valt GLOBECOM 2014] L. Valtulina, M. Karimzadeh, G. Karagiannis, G. Heijenk, A. Pras, "Performance Evaluation of a SDN/OpenFlow-Based Distributed Mobility Management (DMM) Approach in Virtualized LTE Systems", GLOBECOM 2014

[Rica12] R. de O. Schmidt, A. Sperotto, R. Sadre, and A. Pras, "Towards Bandwidth Estimation using Flow Measurements," in Proceedings of the 6th IFIP WG 6.6 International Conference on Autonomous Infrastructure, Management, and Security (AIMS), 2012, pp. 127–138

[Hans06] H. van den Berg, M. Mandjes, R. van de Meent, A. Pras, F. Roijers, and P. Venemans, "QoS-aware bandwidth provisioning for IP network links," Elsevier Computer Networks, vol. 50, no. 5, pp. 631–647, 2006.

[Aiko09] A. Pras, L. J. M. Nieuwenhuis, R. van de Meent, and M. R. H. Mandjes, "Dimensioning Network Links: A New Look at Equivalent Bandwidth," IEEE Network, vol. 23, no. 2, pp. 5–10, 2009.

[BeKM13] Bejarano, O., Knightly, E. W., and Minyoung, P., "IEEE 802.11ac: from channelization to multi-user MIMO", IEEE Communications Magazine, Vol. 51, No. 10, Oct. 2013, pp. 84-90.

[BoAd13] Bohnert, T. M. and A. Edmonds (Eds.), "Overall Architecture Definition, Release 1", Deliverable D2.2, MobileCloud Networking Project, Oct., 2013 (http://www.mobile-cloud-networking.eu/).

[Cisc12] Cisco Systems, 802.11ac: The Fifth Generation of Wi-Fi, Technical White Paper, Cisco, San Jose, CA, USA, Aug. 2012 (http://www.cisco.com/en/US/prod/collateral/wireless/ps5678/ps11983/white_paper_c11-713103.pdf).

[Cisc13a] Cisco Systems, Global Mobile Data Traffic Forecast Update, 2012 - 2017, from Visual Network Index (VNI) White Paper, Cisco Systems, California, USA, Feb. 2013 (http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-520862.pdf).

[Cisc13b] Cisco Systems, The Zettabyte Era - Trends and Analysis, from Visual Network Index (VNI) White Paper, Cisco Systems, California, USA, May 2013 (http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/VNI_Hyperconnectivity_WP.pdf).

[Cisc13c] Cisco Systems, Cisco VNI Adoption Forcast, 2012 - 2017, from Visual Network Index (VNI) White Paper, Cisco Systems, California, USA, May 2013

[DGBA12] Dhillon, H. S., Ganti, R. K., Baccelli, F., and Andrews, J. G., "Modelling and Analysis of K-Tier Downlink Heterogeneous Cellular Networks", IEEE Journal on Selected Areas in Communications, Vol. 30, No. 3, Apr. 2012, pp. 550-560.

[ERIC2013] ERICSSON, ERICSSON Mobility Report, ON THE PULSE OF THE NETWORKED SOCIETY, ERICSSON, Stockholm, Sweden, Nov. 2013 (http://www.ericsson.com/res/docs/2013/ericsson-mobility-report-november-2013.pdf).

[Jaci09] Jacinto, N. M. d. S., "Performance Gains Evaluation from UMTS/HSPA+ to LTE at the Radio Network Level," Master of Science, Department of Electrical and Computer Engineering, Instituto Superior Técnico - Universidad Técnica de Lisboa, 2009

[KhCo14] Khatibi, S. and Correia, L. M., “Modelling of Virtual Radio Resource Management for Cellular Heterogeneous Access Networks”, in Proc. of PIMRC'14 - IEEE 25th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications, Washington, DC, USA, Sep. 2014.

[KZJK12] Khan, A., Zugenmaier, A., Jurca, D., and Kellerer, W., “Network virtualization: a hypervisor for the Internet?”, IEEE Communications Magazine, Vol. 50, No. 1, Jan. 2012, pp. 136-143.

[Math14] http://www.mathworks.com/, Feb. 2014.

[PaPi02] Papoulis, A. and Pillai, S. U., "Probability, random variables, and stochastic processes". McGraw-Hill, New York, USA, 2002.

[PeGS09] Pérez-Romero, J., Gelabert, X., and Sallent, O., “Radio Resource Management for Heterogeneous Wireless Access,” in Hossain, E., (Ed.) Heterogeneous Wireless Access Networks, Springer US New York, NY, US, 2009, pp. 1-33.

[Robe04] Roberts, J. W., “Internet traffic, QoS, and pricing”, Proceedings of the IEEE, Vol. 92, No. 9, Sept. 2004, pp. 1389-1399.

[Tale13] Taleb, T. (Ed.) "Mobile Network Cloud Component Design", Deliverable D4.1, MobileCloud Networking Project, Nov., 2013 (http://www.mobile-cloud-networking.eu/).

[T-Parker] Tammy Parker, ``Chatty smartphones stressing networks''. Internet: http://www.fiercewireless.com/tech/story/chatty-smartphones-stressing-networks/2012-04-27, [Sept. 26, 2014]

[Cost13] R. X. Costa-Pérez, J. Swetina, T. Guo, R. Mahindra und S. Rangarajan, «Radio Access Network Virtualization for Future Mobile Carrier Networks,» IEEE Communications Magazine, 2013.

[CHIN13] C. M. R. Institute, «C-RAN White Paper: The Road Towards Green RAN.,» [Online]. Available: http://labs.chinamobile.com/cran, 2013.

[SAIL12] «Scalable and Adaptive Internet Solutions (SAIL). EU FP7 Official Website.,» [Online]. Available: http://www.sail-project.eu., 2012.

[MCN14] «Mobile Cloud Networking project (FP7-ICT-318109),» [Online]. Available: http://www.mobile-cloud-networking.eu., 2014.

[Wübb14] D. Wübben, P. Rost, B. J., M. Lalam, V. Savin, M. Gorgoglione, A. Dekorsy und G. Fettweis, «Benefits and Impact of Cloud Computing on 5G Signal Processing,» Special Issue "The 5G Revolution" of the IEEE Signal Processing Magazine, 2014.

[Chan13] P. Chanclou, A. Pizziant und e. al., «Optical Fiber Solution for Mobile Fronthaul to Achieve Cloud Radio Access Network,» Future Network and Mobile Summit (FutureNetworkSummit), 2013.

[Ugal09] A. S. Ugal, «Hard Real Time Linux using Xenomai on Intel Multi-Core Processors,» Intel Corporation, 2009.

[Bera08] G. Berardinelli, L. A. Ruiz de Temino, R. M. Frattasi S. und P. Mogensen, «OFDMA vs. SC-FDMA: Performance Comparison in Local Area IMT-A Scenarios,» IEEE Wireless Communications, 2008.

[ZHAW14] «Testbed, ZHAW Openstack,» [Online]. Available: http://blog.zhaw.ch/icclab/, 2014.

[Pall06] V. Pallipadi und A. Starikovskiy, «The ondemand governor: past, present and future,» in Proceedings of Linux Symposium, 2006.

[Bake09] K. R. Baker und D. Trietsch, Principles of Sequencing and Scheduling: Appendix A: Practical Processing Time Distributions, John Wiley & Sons, Inc., 2009.

[OpenEPC] OpenEPC Release 3, http://www.openepc.net/_docs/OpenEPC_DINlang_Rel3_WEB.pdf

[COMBO] European FP7 COMBO project, www.ict-combo.eu.

7 Annex 1 – EPCaaS considerations

7.1 EPCaaS N:2 architecture

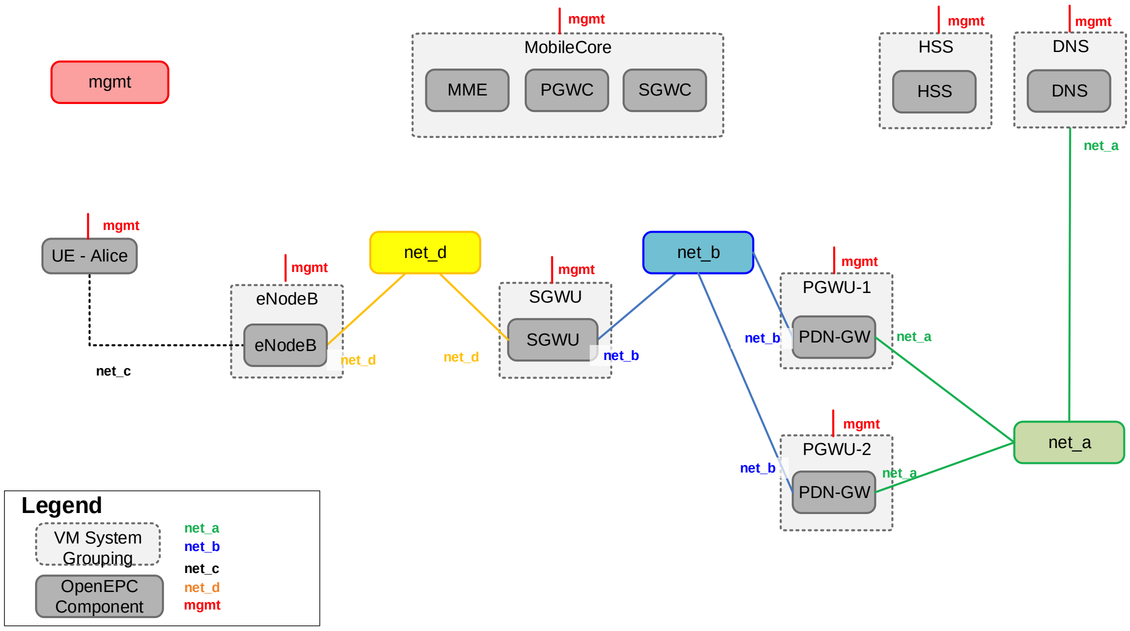


Figure 49 - N:2 EPCaaS option, as implemented by OpenEPC

Figure 49, taken from [MCN D4.2], depicts the N:2 architecture. It includes the following components:

- DNS VM – it is a key piece of the EPC, it handles 3GPP discovery and selection procedures of the other OpenEPC components;
- HSS VM – it stores the 3GPP user profile repository and credentials.
- SGW-U and P-GW-U VMs – include the data plane part of the SGW and P-GW; they are implemented as OpenFlow Switches. They can also collapse in a single VM, which is therefore connected to the RAN on one side (S1-U) and to a PDN on the other side (SGi)
- A single control VM that include the MME, SGW-C and the P-GW-C functionality. Currently this functionality is represented by a set of three different processes running in parallel and communicating via GTP-C using internal sockets. This VM includes also an OpenFlow Controller, which acts as a translation layer between GTP control messages and OpenFlow messages, used to configure forwarding rules on the switches.

The peculiarities of this architecture can be summarized as follows:

- All control entities (MME, SGW-C, P-GW-C SICs) can be hosted on the same Virtual Machine: the 3GPP S11 and S5 interfaces are internal to the VM, thus effectively lowering latencies for session control procedures.
- Control and User Plan SICs can scale independently: e.g. more P-GW-U SICs can be added if the user traffic growths, keeping the same number of MME SICs.

- Multiple SGW-U, P-GW-U SICs can be transparently controlled by one SGW/P-GW-C, in a load balancing mechanism. Since only SGW/P-GW-C FQDN is needed for selection procedures, this simplifies the DNS provisioning. This is one of the benefit brought in by using OpenFlow, since it automates the process of adding a new switches to an already running network, controlled by the OpenFlow Controller in the MME/SGW-C/P-GW-C VM.

7.2 Impacts of the EPCaaS SICs location on the latency of UE idle-to-active signalling procedure

This analysis refers to 3GPP EPC (i.e. 1:1 architecture), but can be extended to cover also N:2 case: in fact, referring to the following pictures, the EPCaaS N:2 “Control node” would contain “MME” together with the SGW-C, P-GW-C component of the “EPS GW”.

Service Request

The delay of the Service Request procedure, depicted in Figure 50 strictly depends from the location of EPC entities in the network, as show in Figure 51.

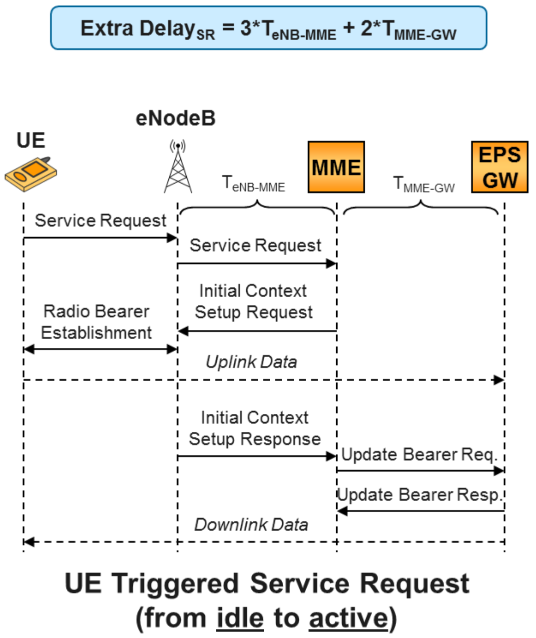


Figure 50 – Service Request procedure.

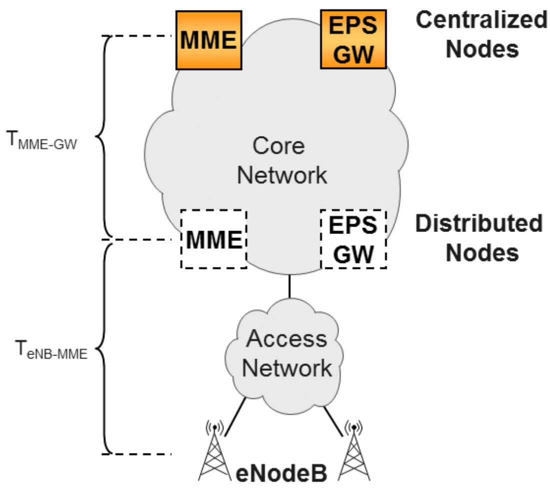


Figure 51 – Location of EPC entities.

As can be seen in the above pictures, if the MME and EPS GW are collocated, the T_MME_GW is close to 0. More precisely, it can also be assumed that the delay for the OpenFlow signalling between SGW/P-GW-C and SGW/P-GW-U is 0, if they are collocated.

Paging

Paging is actually a Service Request procedure, triggered from the Core Network. It means that the extra delay is the one already measured for Service Request, plus the delay of an additional message to reach the UE, as shown in Figure 52.

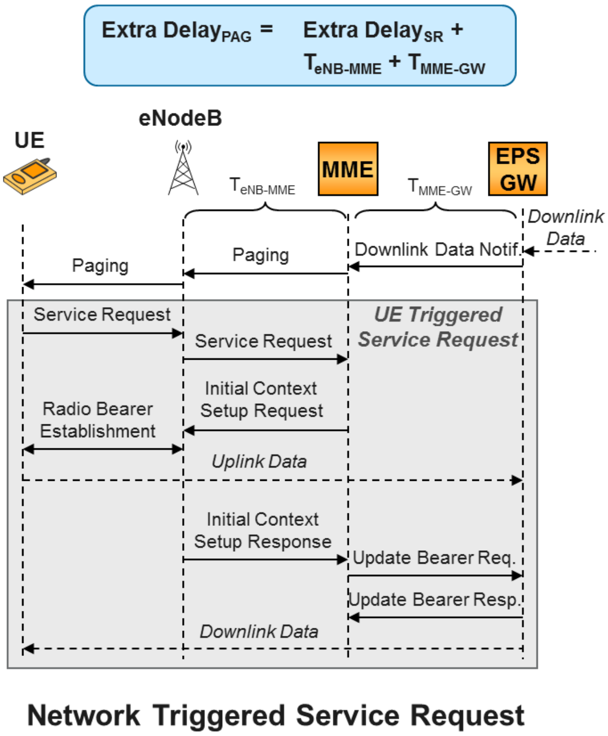
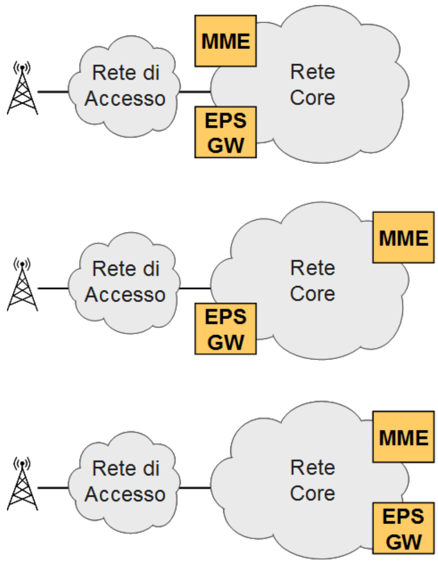


Figure 52 - Paging message sequence and latency.

Having defined the components for the delay in the two procedures, we can now calculate the impact on the total delay for idle-to-active procedures, brought by the different placement scenarios: as can be seen in Figure 53, the total signalling delay is impacted by the relative location of CP end UP entities.



	Service Request	Paging
Extra Delay$_D$	$D_{SR} = 3{*}T_{accesso}$	$D_{PAG} = 4{*}T_{accesso}$

	Service Request	Paging
Extra Delay$_M$	$D_{SR} + 5{*}T_{core}$	$D_{PAG} + 7{*}T_{core}$

	Service Request	Paging
Extra Delay$_C$	$D_{SR} + 3{*}T_{core}$	$D_{PAG} + 4{*}T_{core}$

Figure 53 - Extra delay in different network setups (1:1 architecture)

Having in mind the N:2 architecture, it should be clear that the only two options that can be considered are the first one (totally distributed) and the third one (totally centralized), although the latter introduces more core network round-trips. It can be still a viable option, considering the fact that the latency of a modern core network is usually very low, in the 5-7 ms range, even less. No real measurements can be included in this deliverable, but only estimates: the total extra delay can go from the 10-15 ms range, in the first scenario to the 30-45 ms range in the third scenario.

These considerations cannot be used standalone, to decide the placement of UP and CP entities, because also latency on data traffic should be analysed. The next section covers this aspect.

7.3 Impacts of the EPCaaS SIC location on the end-to-end latency on UE data traffic

The location of EPCaaS UP entities (i.e. SGW-U, P-GW-U[4]) has impacts of course on the e2e latency experienced by IEU application traffic, both directed towards the Internet and both directed to another IEU, possibly locally connected.

From Figure 54, it can be observed that the delay can be defined as follows:

$$\mathbf{Delay_{UE\text{-}PDN} \sim T_{RAN} + T_{eNB\text{-}edge} + T_{edge\text{-}GW} + T_{GW\text{-}PDN}}$$

$$\mathbf{Delay_{UE\text{-}UE} \sim 2^*(T_{RAN} + T_{eNB\text{-}edge} + T_{edge\text{-}GW})}$$

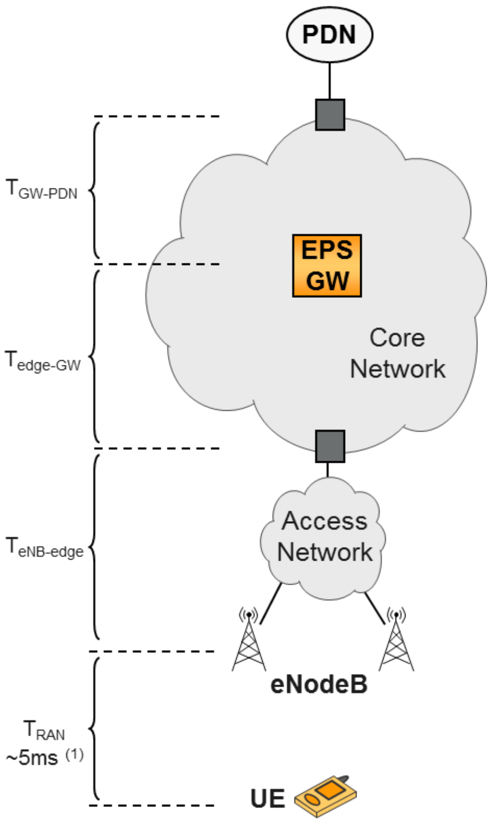


Figure 54 - Analysis of the component of user plane latency

The $\mathbf{T_{RAN}}$ component is highly dependent on how the RAN is deployed in the area of coverage of the user and cannot be reasonably predicted beforehand, but a good reference value is the requirement set by 3GPP TR 25.912, i.e. max. 5ms.

It can be concluded that:

- Placing the UP entities, i.e. OpenFlow switches in the N:2, towards the edge, i.e. in the peripheral DCs, can provide better performance on the mobile-to-mobile traffic locally terminated (you save 2*($T_{edge\text{-}GW}$ + $T_{GW\text{-}PDN}$), i.e. you save twice the Core traversal latency). Moreover, in a scenario where DCs are deployed across a geographical area, there can be services provided by the operator that are distributed as well. In this case, users can get better experience, since they are locally accessing those services.
- For mobile-to-Internet traffic, performance improvements strictly depend on the relative location of UP node and egress point to the internet (the connection to the external PDN in the picture). If this egress point is one and centralized, it makes no difference placing the UP at the edge or at a central site. If multiple distributed egress points are available, it can be advantageous to place the UP in the peripheral DCs as well.

---

[4] Figure 54 depicts an "EPS GW", because it covers the 1:1 EPCaaS architecture. In the N:2 architecture, the EPS GW shall be considered as the user plane component of the SGW/P-GW, namely SGW-U/P-GW-U. As explained in sec 7.1, they are implemented as OpenFlow switches.

7.4 Conclusions

As explained in sec. 7.2, the location of CP and UP entities in EPCaaS has of course impacts on the idle-to-active procedure (which has in turn direct impact on the quality perceived by the user), but within modern networks, the extra delays are anyway very low, varying from the 10-15 range to 30-45 ms. The difference can be considered negligible in practically every case.

Regarding the data traffic, as explained in sec. 7.3, a distributed location of the UP entities could bring to an optimal usage of the underlying transport network and achieve improvements in the e2e delay experienced for mobile-to-mobile traffic terminated locally. The benefits in the case of mobile-to-Internet traffic (which is the vast majority of the mobile traffic today) cannot be spotted so easily, since they strictly depend on the relative position of the CP entities and the egress router towards the Internet or of the service point-of-delivery. If the latter are centralized, no significant improvement can be envisaged.

Surely, if both CP and UP entities are distributed in all the peripheral DCs, the higher number of entities that has to be managed (monitored, provisioned, upgraded, operated) poses additional requirements on the OSS platform of the operator, which will manage an higher number of entities.

8 Annex 2 – RAN considerations

8.1 C-RAN

8.1.1 Description of C-RAN

In networks with C-RAN, BBU processing resources are no longer maintained alongside with a BTS at a remote location. The BBU and RRH are separated; the RRH remains at the previous location, while the BBU migrates to a data-centre. The role of the data-centre is to host a large number of BBUs by providing them with computing power [Wübb14]. Every BBU is connected through, for example, a Common Public Radio Interface (CPRI) via a dedicated link such as an optical fibre to its RRH. Such a very fast link of low delay is required as the BBU processes the most critical physical (PHY) and link layers (MAC) of the LTE standard, which impose hard signal processing deadlines. Notice that the LTE Frequency Division Duplex (FDD) in is extremely deadline critical, because the BBU has to quickly accomplish signal processing at the subframe level. The BBU has to receive and decode a subframe coming from the RRH, as well as, assemble and deliver another subframe back to the RRH within a hard deadline of 3 ms (c.f., Figure 55).

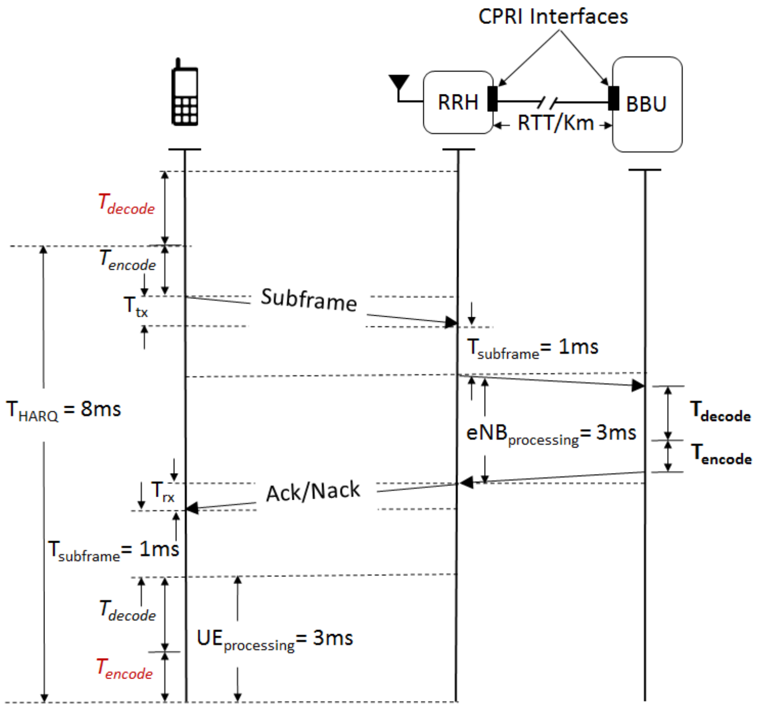


Figure 55 - HARQ process in LTE RAN

Typically, the length of the BBU-RRH link does not exceed 15 km to prevent from high round-trip-delays. Assuming, that the speed of light in fibre is equal to 200 m/us, the round-trip-delay is already at the level of 150 us. According to [Chan13], the round trip propagation time between RRH and BBU is up to 700 us for LTE and 400 us for LTE-Advanced. Notice also that a Real Time (RT) Operating System such as for example RTLinux is required at the BBU side to provide appropriate timing and to introduce a small interrupt-response time of around 50-100 us, which further increases the total round-trip delay [Ugal09], leaving the BBU with only around 2.2ms–2.5 ms for signal processing in the data-centre.

The C-RAN design may offer many benefits to mobile operators related to scalability, energy saving and resources optimization. Other advantages include signal processing performed on general purpose hardware such as typical Graphic Processing Units (GPUs) accessible for example from Amazon (Amazon GPU Instances) and CPUs; therefore no expensive specialized signal processing hardware is required. In this appliance, general-purpose equipment is far less effective than a dedicated device utilizing Digital Signal Processors (DSPs) or Field-Programmable Gate Arrays (FPGAs), but its price is significantly cheaper. Hence, it should be possible to allocate huge computational resources necessary for PHY and MAC processing (possibly, it could be at even supported by currently existing data-centres) for a reasonable price. This new C-RAN processing paradigm for RAN implies that signal processing becomes software-based. It could be for example implemented in CUDA for NVIDIA based environments or in C, C++ for Intel i386, x64 based architectures. This solves the problem of previously mentioned expensive RAN upgrades, as in the future, the problem will reduce to a software upgrade in the data-centre. Every RAN upgrade will probably require negotiating new Service Level Agreements (SLAs) with data-centres to receive more computing power in order to meet the requirements of upcoming communication standards.

8.1.2 Virtualization Layers of LTE RAN

The Virtualization of C-RAN to which we refer as V-RAN is an interesting concept especially for cloud providers. As illustrated in Figure 56, the software-based BBU should include the following layers: i) LTE Physical (PHY) layer with all symbol-level processing, ii) Medium Access Control (MAC) which supports wideband multiuser scheduling and Hybrid automatic repeat request (HARQ) retransmission protocol, Radio Link Control (RLC), and Packet Data Convergence Protocol (PDCP) layer, and iii) Radio Resource Control (RRC) layer. The GPP advantage is that it allows BBU pools to serve multi-vendors, multi-access technologies from a single hardware platform and allocate processing resources dynamically according to the requested load. Moreover, the software-based BBU can be executed on a dedicated machine or even on the top of virtualized cloud platforms, such as CloudSigma, sharing computational and storage resources among different VMs. With increasing coverage of RRHs by BBU pools, fewer pools are required and more efficient optimization of BBU resources could be performed. However, this trend also increases the BBU-RRH round-trip delay and requires faster processors at the BBU pool.

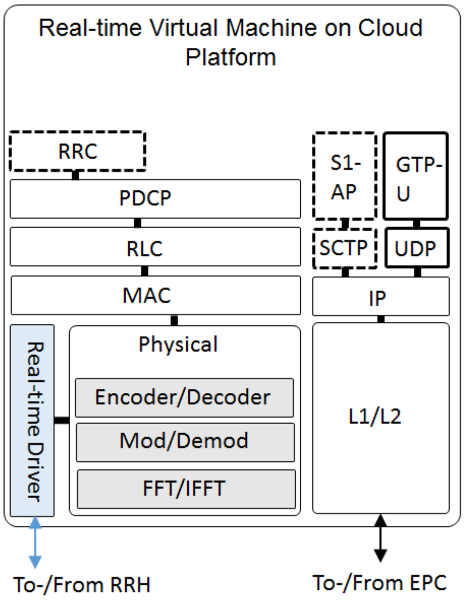


Figure 56 - Software based eNB architecture

8.1.3 LTE RAN PHY Layer

The smallest amount of data that can be identified in LTE physical layer is called Resource Element (RE). Each RE is made out of one LTE symbol and one subcarrier modulated by some amount of bits as illustrated in Figure 57.

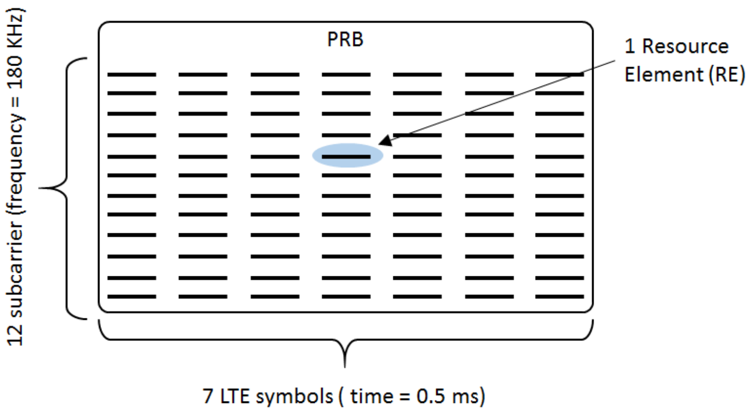


Figure 57 - PRB in time and frequency domains

The amount of bits that can be carried by a single RE depends on the Modulation and Coding Scheme (MCS). UEs report their channel quality, expressed by Channel Quality Indicator (CQI), as an input to the LTE network to decide on the MCS. While the modulation defines the number of bits handled per a symbol, coding rate describes the amount of redundant information used for protecting the data. According to the 3GPP standards, MCS index ranges between 0 and 28. It supports QPSK, 16QAM and 64QAM with different coding rates. The smallest chunk of data transmitted by the LTE eNB is called Physical Resource Block (PRB). Each PRB consists of 12 subcarriers in the frequency domain and 6 or 7 LTE symbols in the time domain. Notice that 6 symbols are used in case of the long Cycling Prefix (CP) insertion. Long CP is used to improve inter-symbol interference [Dahl11].

8.1.4 Channel Bandwidth and PRB

One of the key parameters having influence on the channel capacity in wireless systems is the communication bandwidth. LTE supports communication channels of 1.4 MHz, 3 MHz, 5 MHz, 10 MHz, 15 MHz and 20 MHz, allowing for 6, 15, 25, 50, 75 and 100 PRBs, respectively. Despite the bandwidth allocated to PRBs, there is a remaining spectrum used to guard subcarriers and cannot be used for any transmissions in uplink or downlink.

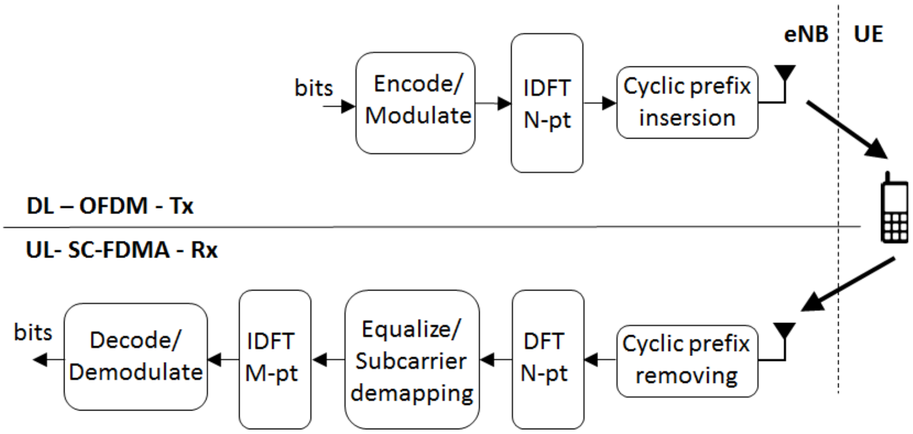


Figure 58 - LTE eNB signal processing in uplink and downlink

8.1.5 Downlink/Uplink Processing chains

The LTE technology uses Orthogonal Frequency-Division Multiple Access (OFDMA) for downlink communication. OFDMA systems have higher resistance to multipath communication in comparison to single-carrier systems. The Peak to Average Power Ratio (PAPR) of OFDM systems grows with the increasing number of subcarrier, i.e., the size of IFFT module N, in Figure 58.

OFDMA is not recommended in uplink communication as it puts higher demands on the power of the amplifier at the UE. The limited battery resources at the UEs led 3GPP to adopt a hybrid access scheme known as Single-Carrier Frequency-Division Multiple Access (SC-FDMA). SC-FDMA combines together low PAPR of a single-carrier system and resilience to multipath propagation provided by OFDMA systems [Bera08]. The system is implemented by additional Discrete Fourier Transform (DFT) and inverse DFT (IDFT) modules at the transmitter and receiver sides, respectively (c.f., Figure 58).

8.2 V-RAN

As referred in D4.1 [Tale13], the interaction between RANaaS and EPCaaS is of main importance. As identified in D3.1 and D3.2, One of the ongoing research within RANaaS is the Management of Virtual Radio Resources, as introduced in D2.2. In the present section an overview of the concept, algorithms and first results is presented. As main added value, VRRM enables to offer and guarantee Service Level Agreements (SLAs) to multiple EEUs tenants using a common physical infrastructure.

One of the key characteristics of RANaaS is the capability to provide multi-tenancy. The tenants, which are Virtual Network Operators (VNOs), are served elastic, on-demand and simultaneously over the same physical infrastructure according to their SLAs. In addition, it is desired to share the limited radio resources (i.e., spectrum) while providing them isolation, ease of use (i.e., network element abstraction), and multi-RAT (Radio Access Technique) support. It is worth noting that the isolation among various Service Instances (SIs) means each SI can have different configuration while any change in one of them does not impose any effect on the other ones. However, the realisation of this concept is not a trivial task.

Hence, the concept of virtualisation of radio resources to realise virtual wireless links is proposed. It is also referred as the fundamental step toward implementing Connectivity-as-a-Service. The novelty of this concept comparing to RAN sharing (i.e., the similar approach) can be summarised as network element abstraction, isolation among virtual instances, and ability to support multi-RAT (Radio Access Technology). The better differentiation of these two concepts can be addressed using the analogy presented in [KZJK12], where a process on an Operating System (OS) is presented as equivalent of a session on a network. As it is depicted in Figure 59 the Virtual RAN (V-RAN) and the Virtual Machine (VM) can be claimed to be realisation of corresponding concept while RAN-sharing is equivalent of multi-tasking in OSs.

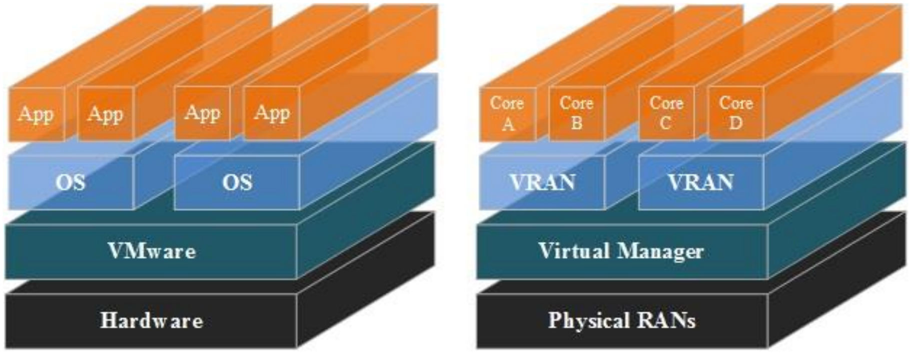


Figure 59 – Comparison of V-RAN and VM [KhCo14].

In a virtualisation approach, in contrast to network sharing, the physical infrastructures are not transparent to the VNOs. The operators have an abstract vision of physical elements. To be more specific, in V-RAN, the operator requests from virtual resource manager a set of virtual wireless links with certain capacity and Quality of Service (QoS) requirement to send/receive data to/from its subscribers. In return, it is served by virtual links and it does not involve in details of channel assignment, power control and so on.

Isolation, the second feature offered by virtualisation, enables offering services with different protocols, algorithms, and requirements over the same physical infrastructure. Since the two virtual entity of a virtual network are completely isolated from each other, the problem of one of them does not affect the other one. This fact leads to lower system down time comparing to shared network. Variation of VNO traffic does not affect the quality of service offered to the other operator.

At last but not at least, ability to support multi-RAT is another difference between these two concepts. Wi-Fi traffic offloading, as an example for this matter, cannot be realised for multiple VNOs operating over the same physical infrastructure since there are not enough available channel (i.e., only one 160MHz channel for IEEE 802.11ac [BeKM13] to allocate to each VNO separately. The only practical solution is to apply the radio resource virtualisation to realise Wi-Fi virtual links for each VNO.

