5G PPP

European Commission | Horizon 2020 European Union funding for Research & Innovation

5G-XHaul

Dynamically Reconfigurable Optical-Wireless Backhaul/Fronthaul with Cognitive Control Plane for Small Cells and Cloud-RANs

D3.3 5G-XHaul algorithms and services Design and Evaluation

This project has received funding from the European Union's Framework Programme Horizon 2020 for research, technological development and demonstration

Advanced 5G Network Infrastructure for the Future Internet

Project Start Date: July 1st, 2015 | Duration: 36 months

H2020-ICT-2014-2 671551 | 14th August 2018 – Final Version

Project co-funded by the European Commission

Under the H2020 programme

Dissemination Level: Public

Grant Agreement Number:	671551
Project Name:	Dynamically Reconfigurable Optical-Wireless Backhaul/Fronthaul with Cognitive Control Plane for Small Cells and Cloud-RANs
Project Acronym:	5G-XHaul
Document Number:	**D3.3**
Document Title:	5G-XHaul algorithms and services Design and Evaluation
Version:	1.0
Delivery Date:	30th June 2018 (**14th August 2018**)
Responsible:	**IHP**
Editor(s):	Jesús Gutiérrez (**IHP**)
Authors:	Meysam Goodarzi (**IHP**), Nebojsa Maletic (**IHP**), Vladica Sark (**IHP**), Daniel Camps-Mur (**i2CAT**), Eduard García-Villegas (**i2CAT**/**UPC**), Najeeb Ul Hassan (**HWDU**), Emmanouil Pateromichelakis (**HWDU**), Artur Hecker (**HWDU**), Jim Zou (**ADVA**), Stefan Zimmermann (**ADVA**), Kostas Choumas (**UTH**), Paris Flegkas (**UTH**), Ioanna Mesogiti (**COS**), Konstantinos Filis (**COS**), Thierno Diallo (**UNIVBRIS-HPN**), Rafael Cantó (**TID**), Nicolás Ángel Serrano Linares (**TID**).
Keywords:	End-to-End Scalable Control plane, SDN, mobility prediction, Service Level Agreement, multitenancy, virtualisation, QoS, Traffic Engineering
Status:	Final
Dissemination Level	**Public** / Confidential
Project URL:	http://www.5g-xhaul-project.eu/

Version History

Rev. N	Description	Author	Date
0.0	First draft with ToC	Jesús Gutiérrez (**IHP**)	06/02/2018
0.1	Included draft sections 2.4 and 3.5	Meysam Goodarzi, Vladica Sark, Jesús Gutiérrez (**IHP**)	15/03/2018
0.2	Included draft sections 2.1 and 3.2	Daniel Camps-Mur, Eduard García-Villegas (**i2CAT**/**UPC**)	15/03/2018
0.3	Included draft sections 2.3 and 3.4	Emmanouil Pateromichelakis (**HWDU**)	13/04/2018
0.31	Included draft sections 2.2 and 2.1.2	Kostas Choumas, Paris Flegkas (**UTH**)	16/04/2018
0.4	Draft including previous contributions	Jesús Gutiérrez (**IHP**)	27/04/2018
0.5	Added a new contribution in Section 2.3 Contribution from COS/TID on SLAs	Artur Hecker (**HWDU**) Ioanna Mesogiti (**COS**)	02/05/2018 07/05/2018
0.6	Added contribution from **ADVA**, **TID**, revised Section 2.3 (**HWDU**)	Jim Zou (**ADVA**), Rafael Cantó (**TID**), Artur Hecker (**HWDU**), Najeeb Ul Hassan (**HWDU**), Jesús Gutiérrez (**IHP**)	15/05/2018
0.6	Added revised contribution to Section 3.3 and Section 3.5	Ioanna Mesogiti (**COS**), Elina Theodoropoulou (**COS**), Konstantinos Filis (**COS**), George Lyberopoulos (**COS**), Rafael Cantó (**TID**), Nicolás Ángel Serrano (**TID**)	18/05/2018
0.7	Revision of Section 3.5 and draft compilation	Meysam Goodarzi (**IHP**), Jesús Gutiérrez (**IHP**)	22/05/2018
0.8	Full initial review and restructuring of the document	Daniel Camps-Mur (**I2CAT**), Jesús Gutiérrez (**IHP**)	24/05/2018
0.9	IHP revision, COS revision	Meysam Goodarzi (**IHP**), Ioanna Mesogiti (**COS**)	04/06/2018
0.95	Section 5 on KPIs, Revised Section 2.4 and 3.3, Revised Section 2.2 and Section 2.1.2, Revised Section 4.1	Daniel Camps-Mur (**i2CAT**), Emmanouil Pateromichelakis (**HWDU**), Kostas Choumas (**UTH**), Dimitris Giatsios (**UTH**), Nicolás Ángel Serrano (**TID**)	06-07/06/2018
0.96	Final contribution to Section 2.5	Stefan Zimmermann (**ADVA**)	19/07/2018
0.98	Section 3.5.2 with additional results	Vladica Sark (**IHP**), Nebojsa Maletic (**IHP**)	07/08/2018
0.99	Section 2.1 with additional results	Daniel Camps-Mur (**i2CAT**)	09/08/2018
1.0	Final Document Review, Version for submission	Jesús Gutiérrez (**IHP**)	14/08/2018

Table of Contents

List of Figures

List of Tables

Executive Summary

This document constitutes deliverable D3.3 from the 5G-XHaul project, and reports on the work carried out during the third year of the project within WP3. It provides a clearer insight on which of the developments carried out in WP3 have been implemented towards the 5G-XHaul final demonstration activities, and nails down the different research activities from the different partners.

The structure and concepts initially defined in deliverable D3.1 and, subsequently, extended in deliverable D3.2, are in this document strengthened through i) a detailed specification and demonstration of the end-to-end SDN control-plane hierachy providing connectivity across wireless and optical technology domains; and ii) the exploitation of RAN information, including user mobility and cell loads, to optimise the transport network.

More specifically, we first continued the development of southbound connectors and agents to allow communication between Level-0 controllers and the 5G-XHaul wireless and optical network elements. Second, we developed the higher levels of the 5G-XHaul hierarchical control plane. Third, we evaluated techniques to localize user device with sub-meter precision using a single anchor point, combining mmWave and Sub-6 radios, and we define mobility prediction models that estimate the future movement of groups of users to feed network planning and management tools. Finally, we demonstrate a multi-domain connectivity service between two control plane areas located at NITOS and i2CAT, hence fully validating the 5G-XHaul hierarchical control plane.

1 Introduction

The 5G-XHaul control plane was introduced in deliverable D3.1 [1], where the main architectural building blocks were presented, along with several control plane mechanisms which operate at various parts of this architecture. Deliverable D3.2 [2] went in depth on the definition of the 5G-XHaul control-plane architecture, with the definition of interfaces and mechanisms that enhance the scalability of an SDN-based control plane in transport networks. As well, various Traffic Engineering (TE) mechanisms tailored to wireless transport areas, including millimetre wave (mmWave) and Sub-6 technologies were included and we have further defined the southbound interfaces between the 5G-XHaul network elements and their corresponding controllers. Finally, we have delved deeper into mechanisms that enable a tighter intergration between the Mobile and the Transport Network.

Figure 1-1 represents an architectural view of the work in WP3, as the control plane building blocks developed in 5G-XHaul. It depicts in orange the aforementioned contributions belonging to deliverables D3.1 and D3.2, which are refined and extended in this deliverable. The remaining blocks and entities related to complement the previous work in network management and Service Level Agreement (SLA) definition are as well highlighted. Altogether, they will provide an operator with the means to manage in an unified manner the heterogeneous 5G-XHaul data plane (shown below as WP4 work).

The final contributions within WP3 target the following aspects:

1) development of the higher levels of the 5G-XHaul hierarchical control plane together with the demonstration of a multi-domain connectivity service between two control plane areas.
2) design of a new protocol to provide transactional network updates to traditional SDN southbound protocols. Final development of southbound connectors and agents to allow communication between Level-0 controllers;
3) exploitation of RAN information, whether it is by localising user nodes, or predicting user mobility and cell loads, to optimise the transport network;
4) definition of the 5G-XHaul SLAs.

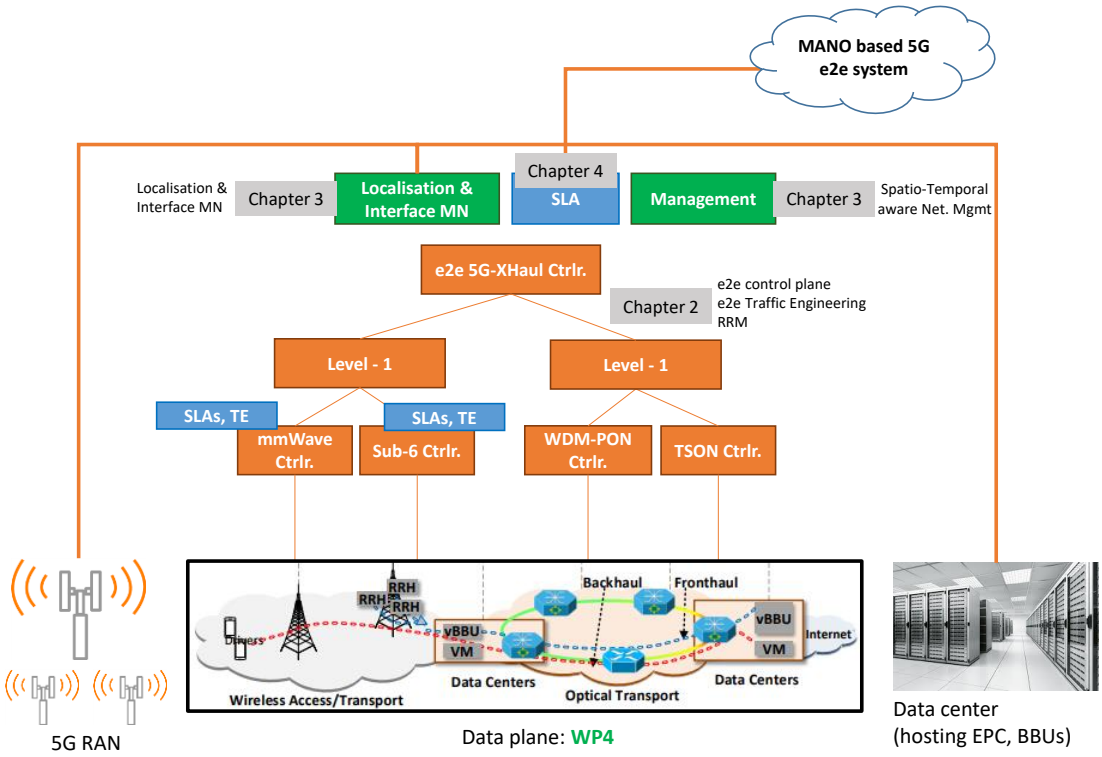


Figure 1-1: Mapping of the remaining work in WP3 to the sections in this deliverable.

The document at hand further develops the previous concepts, while including exhaustive performance evaluations for the various mechanisms proposed.

This deliverable is organised as follows:

Chapter 2 presents the activities related to the end-to-end control plane definition across technologies. It refines the interface definition between controllers, and presents mechanisms to both cluster and place controllers in a distributed hierarchical control plane, and to provide transactional update properties to distributed hierarchical control planes. Work on Traffic Engineering (TE) and south-bound interfaces are included, which complement those included in previous deliverables. Finally, we evaluate a multi-domain connectivity service between two control plane areas located at NITOS and i2CAT, hence fully validating the 5G-XHaul hierarchical control plane

Chapter 3 extends the cognitive capability of the control plane, with work on estimation/prediction and tools to acquire relevant information from the RAN, which can be used by the control plane, to optimise the transport network accordingly.

Chapter 4 provides set of mechanisms which to the higher levels of the 5G-XHaul control plane hierarchy. Concretely we define the SLAs for different services, and we present the SLAs the 5G-XHaul solution enables.

Chapter 5 puts forward how the innovations presented in this deliverable impact some of the 5G-PPP KPIs, fitting in this context those related to the control plane.

Finally, chapter 6 draws the conclusions of the deliverable.

It is worth noting that some of the mechanisms described in this deliverable have been extended through several peer-reviewed publications, which are properly referenced throughout the document.

2 E2E optical/wireless control plane activities

After the initial definition of the 5G-XHaul hierarchical control plane architecture in deliverable D3.1 [1], and a detailed description of mechanisms and algorithms focusing on the lower levels of the control hierarchy in deliverable D3.2 [2], in this section we introduce a set of mechanisms that relate to the higher levels of the 5G-XHaul control plane hierarchy, and we propose some additional mechanisms related to the lower levels that complement those captured in deliverable D3.2 [2].

In particular, this chapter presents the following five contributions:

- Section 2.1 provides a detailed description and evaluation of the 5G-XHaul e2e control plane, including the interface between L1 and L0 controllers, and between L1 and Top controllers, both based on the Control Orchestration Protocol (COP), as well as the interface between the Top controller and the ETN and IATN network elements.
- Sections 2.2 and 2.3 describe respectively a mechanism to cluster and place controllers in a distributed hierarchical control plane, and a mechanism to provide transactional update properties to distributed hierarchical control planes. These mechanisms contribute to reliability and scalability of hierarchical SDN controller planes as it is the case in 5G-XHaul.
- Section 2.4 describes a Traffic Engineering (TE) mechanism for mmWave control plane areas providing joint access and backhaul Radio Resource Management (RRM). This mechanism can be implemented in a L0 controller for mmWave control plane area, and complements previous work in deliverable D3.2 [1] on joint access and backhaul for Sub-6 control plane areas.
- Finally, Section 2.5 provides the complete definition and evaluation of the NETCONF agent for the WDM-PON Optical Line Terminal (OLT) node developed in 5G-XHaul. This is an SDN south-bound interface that complements the wireless and the Time Shared Optical Network (TSON) SDN interfaces already described in deliverables D3.1 [1] and D3.2 [2].

2.1 5G-XHaul E2E optical/wireless control plane definition and evaluation

2.1.1 Consolidated 5G-XHaul hierarchical control plane

Figure 2-1 depicts a refinement of the 5G-XHaul hierarchical control plane originally introduced in deliverables D3.1 [1] and D3.2 [2]. In previous deliverables though, our focus was on the design of the technology specific L0-controllers, whereas in this section we present a detailed design of the higher layers (L1 and Top controllers), and report on a proof of concept that we have developed to validate the operation of the 5G-XHaul control plane.

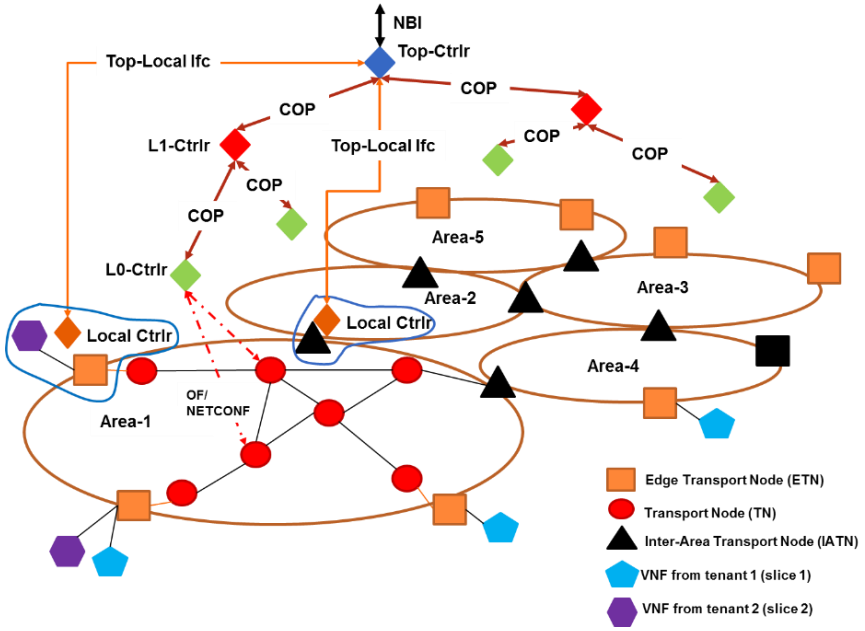


Figure 2-1: Final hierarchical 5G-XHaul control plane design.

As introduced in deliverable D3.1, the 5G-XHaul architecture is composed of three main functions in the data-plane, namely the Transport Nodes (TNs), depicted as red circles in Figure 2-1; the Edge Transport Nodes (ETNs), depicted as orange squares in Figure 2-1; and the Inter-Area Transport Nodes (IATNs), depicted as black triangles in Figure 2-1. The TNs act as regular tenant-agnostic transport nodes, and could be instantiated by different technologies, such as wireless or TSON. The ETNs would typically be implemented as a software datapath in a network hypervisor, and hold all tenant related state thus enabling virtualisation/multi-tenancy over the 5G-XHaul transport network. Finally, the IATNs stitch pre-defined connections between different control plane areas, where each area is under the control of a technology specific L0-controller.

The main principle adopted in the design of the higher layers of the 5G-XHaul control plane is the separation of responsibilities between the L1-controller and the Top-controller, whereby the Top-controller interfaces with the ETNs, e.g. in order to provision a new Virtual Network Function (VNF); and with the IATNs in order to stitch domains. On the other hand the L1-controller's job is to act as an aggregator of L0-controllers, thus interacting only with these controllers, which end up programming the TN nodes of each domain.

The relevant interfaces in the operation of the higher layers of the 5G-XHaul control plane are:

- The Top-Local interface, which enables the Top controller to interact with a Local controller that manages ETNs and IATNs. The local controller is collocated with the ETN or IATN, as indicated by the circle in Figure 2-1. The Top-Local interface is described in detail in section 2.1.2.
- The interface between the Top and the L1-controller, and the interface between the L1- and L0-controllers, which are both implemented using the Control Orchestration Protocol (COP) introduced in deliverable D3.2 [1]. These interfaces and the role of the L1-controller are described in detail in section 2.1.3.

2.1.2 Top controller and Top-Local interface definition

The Top-local interfaces are different for the local controllers hosted at ETNs to those hosted at IATNs, as these types of nodes are responsible for different functionalities. Both of them, however, are Representational State transfer (REST)-based interfaces.

In the case of ETNs, the basic resources made available to the Top controller are called *Virtual Interfaces,* as an abstraction that covers both interfaces directly connected to a VNF, and interfaces of a virtual datapath (vDP) in a tenant's slice topology. These are uniquely identified by the L2 segment ID (L2SID) where they are attached and the corresponding MAC address, scoped at the specific L2SID. Each virtual interface is hosted at one and only one ETN at a given time (migrations are of course allowed), and this information is a property of the virtual interface resource. The other type of managed resource are tunnels, also called paths, whose source node is the particular ETN, and destination is some remote ETN. These are identified by the (area-specific) path ID attached to packets of that path leaving the ETN, which in general is a VLAN ID field - for instance, the B-VLAN field value in case of a Provider Backbone Bridging (PBB) implementation. They feature at least one associated value, indicating the destination ETN, but they can also optionally feature information about the QoS guarantees provided by that path, thus allowing the controller to forward packets of special requirements flows accordingly.

The REST operations allowed on these two types of resources are summarised in the bullets below. They are self-explanatory in general. We just make two notes here. First, the PUT operations can be used both to add a new resource and to update the data of an existing resource (like in the case of VNF migration for example). Second, and most important, the virtual interface resources maintained at each ETN local controller are not restricted to the ones hosted in this particular ETN. They include all virtual interfaces attached to an L2 segment featuring at least one endpoint at this ETN. This is required to make sure that the ETN knows *where* to forward packets intended for virtual interfaces not hosted locally.

- *PUT /vifaces/l2sids/{l2sid}/mac_addresses/{mac_address}*
  - *Data: {ETN ID}*
- *DELETE /vifaces/l2sids/{l2sid}/mac_addresses/{mac_address}*
- *GET /tunnels*
- *PUT /tunnels/{path_id}*
  - *Data: {destination ETN ID, possibly QoS guarantees information}*
- *DELETE /tunnels/{path_id}*
- *GET /tunnels*

In the case of IATNs, the resources managed by the Top controller are tunnel (path) mappings between areas attached to the IATN. The tunnels are organised by incoming IATN port, since in our design each IATN port corresponds to an area. They are uniquely identified by this port, the incoming Path ID, and the associated properties are the outgoing Path ID and the outgoing IATN port. Remember that these translations are required to keep full independence of different areas (domain stitching functionality). So, in fact the IATN Local Agent is a simple thin layer acting as a proxy between the Top-Controller and the IATN datapath. The REST operations supported by it are summarised below. Again, the PUT operation can be used to add a new resource or update its data. Note that, while an update operation is sufficient to change the outgoing port or outgoing path ID, in order to change the incoming port or path ID it is required to delete the old resource and add a new one.

- *PUT /tunnels/ports/{incoming_port}/path_ids/{incoming_path_id}*
  - *Data: {outgoing path id, outgoing port)*
- *DELETE /tunnels/ports/{incoming_port}/path_ids/{incoming_path_id}*
- *GET /tunnels*

Based on the interaction with the local agents (controllers) of ETNs and IATNs just described, the Top-Controller is responsible for performing a number of operations. The first one is deployment of VNFs and vDP interfaces at specific ETNs, thereby instantiating a given tenant slice virtual topology. Another operation is to provide the functionality of simple migration of virtual interfaces from one ETN to another. The important thing to realize is that the Top-Controller makes sure that each ETN is kept up to date about the location of all virtual interface addresses it might need to send packets to. This requires a global view of the location of all virtual interfaces, and could not be addressed locally. Yet another operation is informing the local controllers of edge elements (ETNs, IATNs) of all path IDs they should be aware about. This comes as the subsequent step after an initial step of actual path establishment, which involves the interaction between the Top-Controller and L1-Controllers. The latter interaction is analysed in the following subsection (2.1.3).

2.1.3 L1 controller and COP interface definitions

The following two main services are offered by the L1-controller towards the Top-controller:

- A topology dissemination service, able to retrieve topologies from individual L0-controllers and aggregate them into an e2e topology.
- A path provisioning service, able to receive a path provisioning request involving nodes located in different domains (i.e. control plane areas) and resolve it into separate path requests for each of the involved domains.

In addition, the L0-controller offers the same two services towards the L1-controller:

- A topology dissemination service, where the L0-controller exports the topology specific to its domain (i.e. control plane area) in COP format. We note that the L0-controller might choose to report only a summarised version of its topology, where the only critical information to be exposed to the L1-controller are the nodes connecting to an ETN or an IATN.
- A path provisioning service, whereby the L0-controller receives a request to connect to TNs under its control, and the L0 control responds with the corresponding label switched path identifiers.

To implement these two services we use the Control Orchestration Protocol (COP), originally defined in the Strauss project [3]. COP is a Representational State Transfer (REST) based protocol, which defines a set of data models to allow REST endpoints to offer network related services. In particular, COP defines the following data models that implement a topology and connectivity provisioning service:

- COP *service-topology*. Available at:
  https://github.com/ict-strauss/COP/blob/master/yang/yang-cop/service-topology.yang
- COP *service-call*. Available at:
  https://github.com/ict-strauss/COP/blob/master/yang/yang-cop/service-call.yang

Figure 2-2 illustrates the COP service-topology data model, which consists of a list of nodes, representing network devices, and edges, representing network links. In COP nodes embed multiple edge-ends, representing a port or an interface, and edges refer to the nodes at each side of the link.

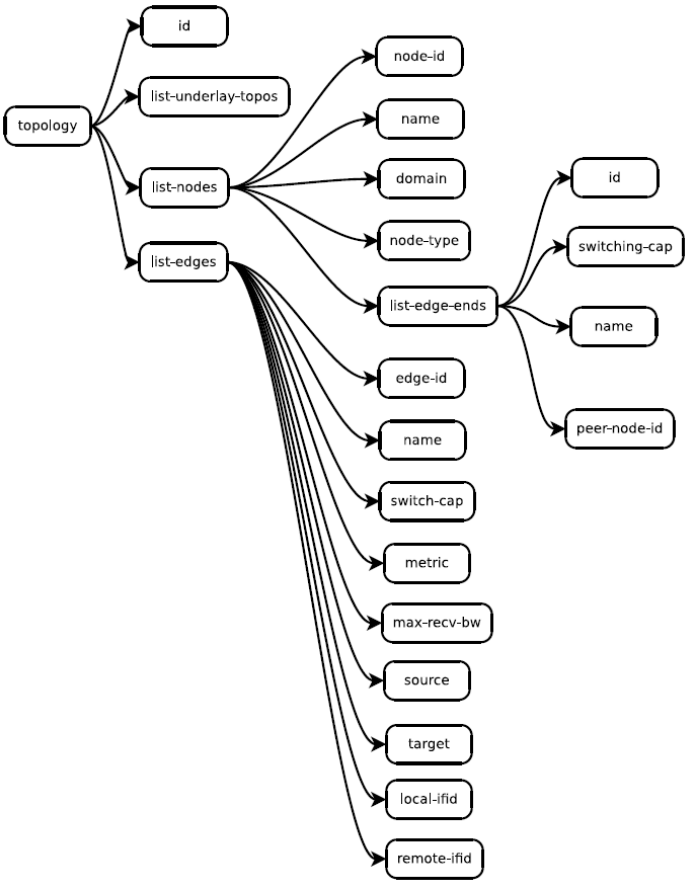


Figure 2-2. COP service-topology.

There is no direct mapping between the data models defined in COP and all the functionality defined in 5G-XHaul, for example introducing the notion of ETNs, IATNs and TNs. Therefore we need to make some design decisions in order to map 5G-XHaul functions into the existing COP data-models, which we prefer to maintain unmodified for forward compatibility. In this regard, our main design assumption is that since the L1-controller only interacts with the L0-controllers, not ETNs or IATNs, the COP topology will only report TNs in its list of nodes. However, the Top-controller needs to be able to resolve an ETN or IATN into a TN in order to issue a path request to the L1-controller. Since the mapping between ETN/IATNs and TNs is expected to be something fairly static, we opt to manually provision the L1-controller with this information. In particular, we enable an additional REST endpoint in the L1-controller that allows to specify information about IATNs and ETNs connecting to one of the L0 domains under the control of this L1-controller. Then, the L1-controller exposes the IATN/ETN information as a COP edge. This information is sufficient for the Top-controller to match ETN/IATNs with TNs and issue a path request.

In particular, the L1-controller maps 5G-XHaul functions into COP topology objects in the following manner:

- Node: *TN_switchIdA*
- Regular edge: *TN_switchIdA:PortA->TN_switchIdB:PortB*
- IATN edge: *IATN_switchIdA:PortA->IATN_switchIdB:PortB*
- ETN edge: *ETN_switchId:Port*

To feed the COP topology module in the L1-controller we developed a COP server for OpenDayLight (ODL) that obtains topology information from the Model-Driven Service Abstraction Layer (MD-SAL), and publishes it through COP. The MD-SAL is automatically populated with topology information if ODL is connected to OpenFlow switches. The COP server would then run in an L0-controller, and allow the L1-controller to poll for topology. In our current implementation the L1-controller periodically polls the L0 controller for topology updates. A more efficient publish/subscribe interface is left for future work.

The 5G-XHaul L1-controller supports the following operations:

- *Add L0-controller*: Provisions a new L0-controller under the control of this L1-controller. Upon adding a new L0-controller the L1-controller automatically connects to it and retrieves its individual topology.
- *Add IATN*: Registers an IATN node that connects to a TN in one of the L0 domains under the control of this L1-controller. The corresponding L0-controller must have been added before into the L1-controller.
- *Add ETN*: Equivalent operation than Add IATN.
- *Get Topology*: Retrieves the aggregated topology retrieved from all the provisioned L0-controllers, following the COP data model and 5G-XHaul mapping rules described in the previous paragraphs.
- *Add Service call*: Provisions an end-to-end connectivity service between two ETN topology endpoints.

To illustrate in more detail the service call provisioning mechanism we include Figure 2-3, where we can see two domains A and B, respectively connecting ETN A and ETN Z, and an IATN interconnecting both domains. We also observe in the figure the concrete TNs composing each domain, where as previously described only the nodes connecting to the ETN and IATN need to be included, while the rest could be abstracted for simplicity. Finally, we illustrate with a yellow and blue lines two unidirectional connections. The yellow arrow signals a unidirectional connection between ETN Z and ETN A, which is identified by VLAN 210 in Domain B, and by VLAN 110 in Domain A. Each of these connections has been provisioned by the corresponding L0-controller, and the IATN is in charge of stitching the different connections in each domain. The unidirectional connection between ETN A and ETN Z is depicted with the blue arrows corresponding to VLANs 100 and 200 in Domain A and B respectively. Notice that the 5G-XHaul architecture is flexible to accommodate various connectivity services, and even to different services in different domains (e.g. VLANs in one domain, and MPLS tags in another domain). Without the lack of generality, hereafter we adapt our description and our proof of concept to the 5G-XHaul unidirectional VLAN connectivity service, which was the one more readily available in our prototypes.

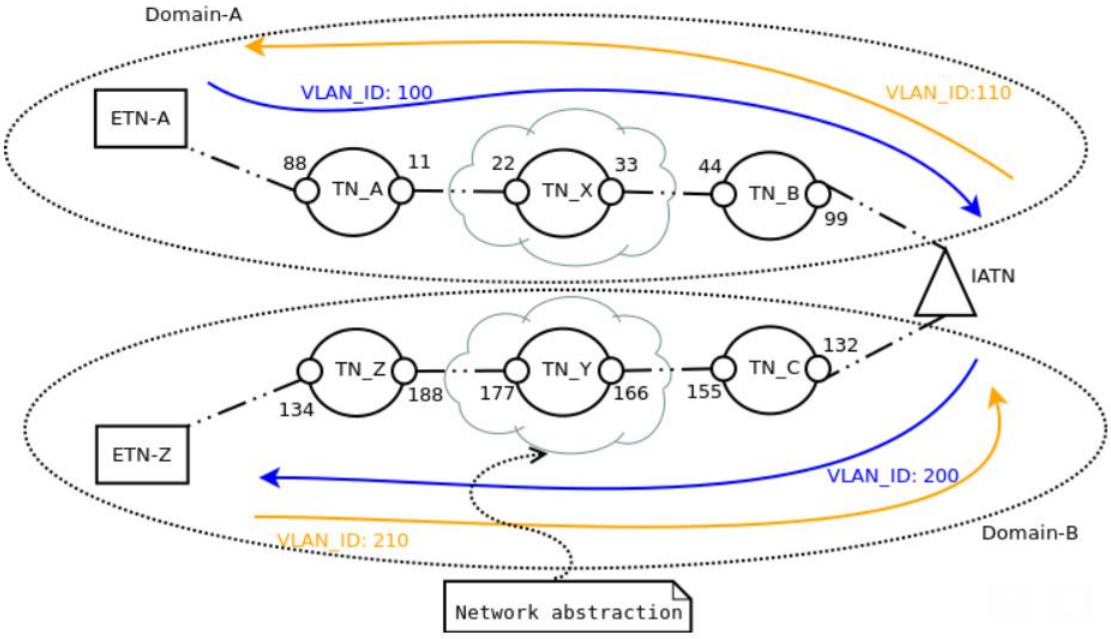


Figure 2-3: Example topology to illustrate operation of the COP service call.

The first step in provisioning the connectivity service is a POST request from the Top-controller, where it specifies as endpoints ETN-A and ETN-Z, and specifies the type of connectivity service[1], as well as a QoS class. This is illustrated in Figure 2-4.

[1] A new transport layer type was defined in COP to indicate the 5G-XHaul VLAN service

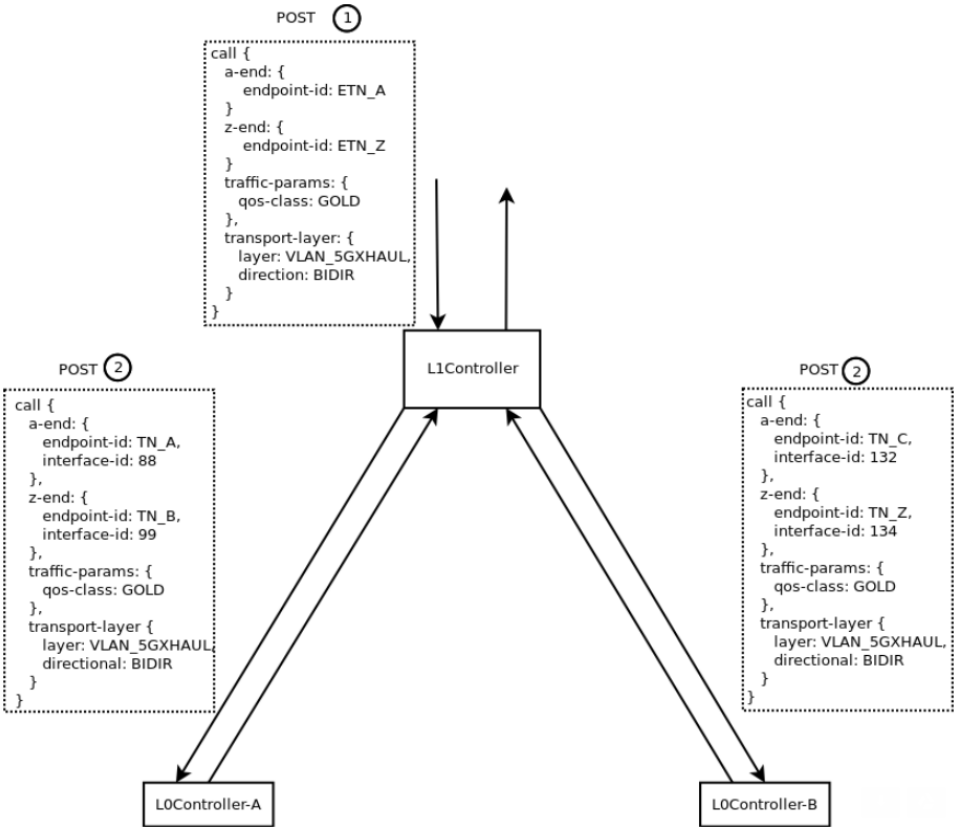


Figure 2-4. Service call request from ETN controller and split calls towards L0-controllers.

Upon receiving the POST service-call message from the Top-controller, the L1-controller first inspects the domains where each ETN is located, through its pre-provisioned information. After finding the domains where each ETN is located, it derives the TN connecting to each of the ETNs, and discovers the IATN connecting the two domains. Notice that in a more general case where the two domains may not connect directly, the L1-controller would have to resolve an end to end path at the domain level to connect the two domains. Once the L1-controller has resolved the involved IATNs, it can also resolve the TNs that connect to each IATN from the end to end topology delivered by the L1-controller. With this, the L1-controller is now in position to issue two separate POST service-call requests to each of the involved L0-controller, where the request endpoints are adapted to the involved TNs in each domain. The transport layer service and QoS parameters are reused from the original Top-controller request.

Figure 2-5 illustrates how after receiving the service-call, each L0-controller computes the actual path connecting the endpoints in its domain. Following the unidirectional VLAN 5G-XHaul service, each L0 controller provisions two connections between the endpoints, one for each direction, and a different path identified (VLAN), is assigned to each connection.

In the last step described in Figure 2-6, the L1 controller returns the POST service-call message from the Top controller, with a *200 OK*, and in the body includes all the information about the provisioned connections. The important point to notice is that L1-controller replaces the endpoints returned by the L0-controllers with the corresponding ETN or IATN endpoints, which are what the Top-controller understands. With the VLAN information included in the label object of each connection, the Top-controller is now in possession of the information that it needs to program the datapaths of the involved ETNs and IATNs.

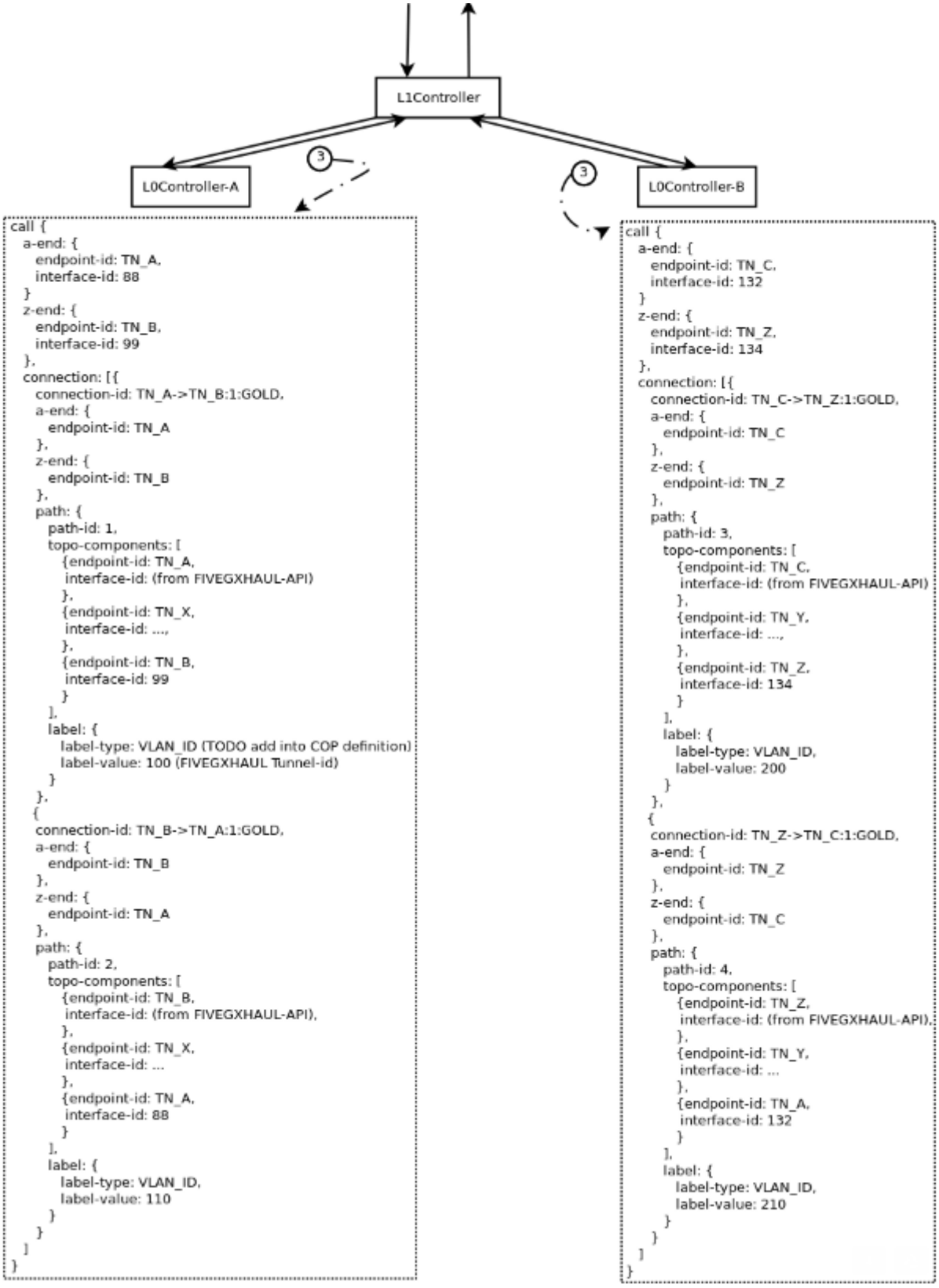


Figure 2-5. Response from L0-controllers with actual path allocations and path IDs.

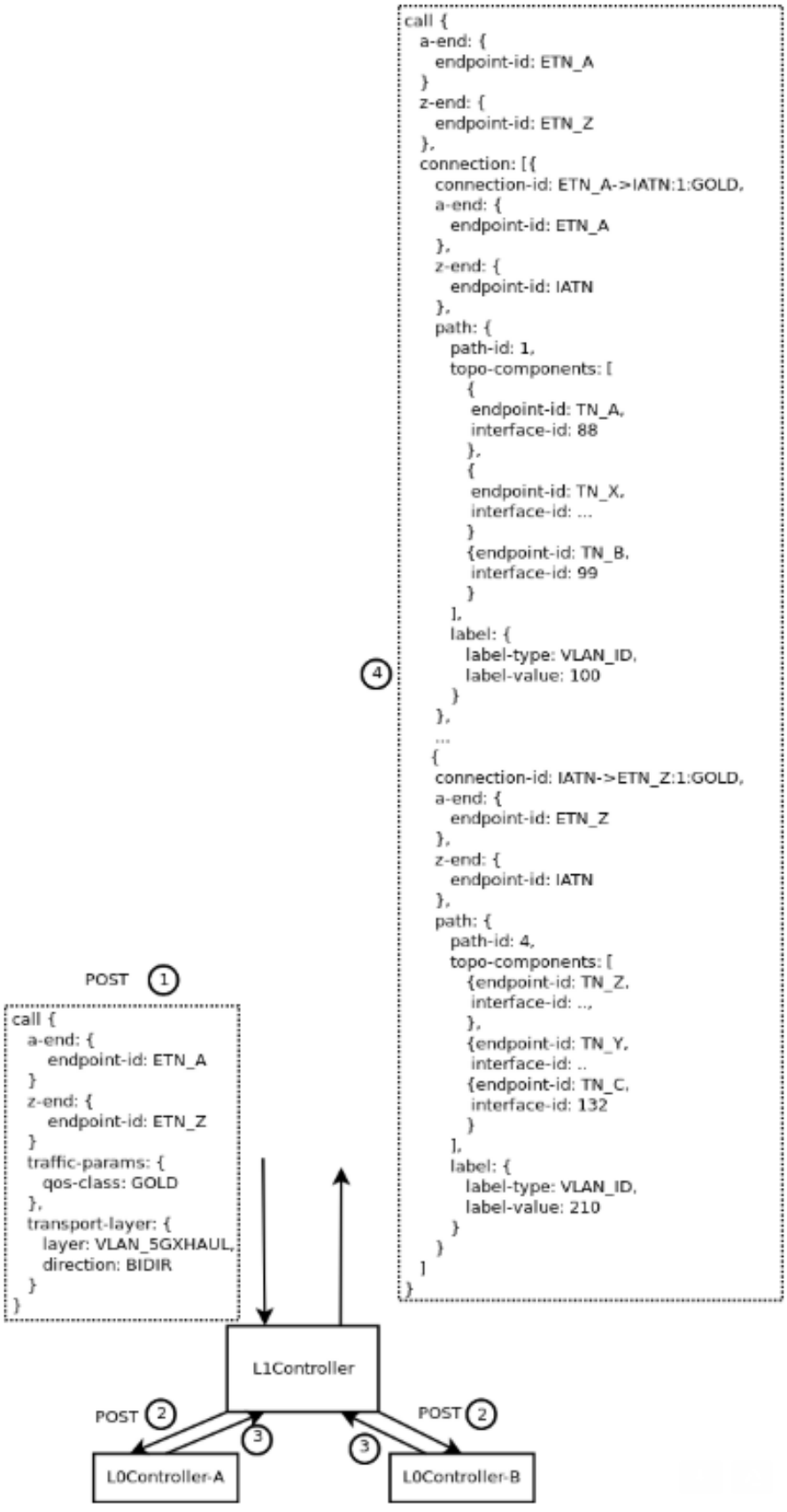


Figure 2-6. Final reply from L1-controller to ETN controller.

2.1.4 Validation of the 5G-XHaul hierarchical end to end control plane

In this section we evaluate the performance of the 5G-XHaul hierarchical control plane looking at the overall time required to provision an end-to-end connection involving multiple control plane areas. To carry out this evaluation we have prepared the testbed described in Figure 2-7. End-to-end control plane for evaluation.. Our testbed consists of a wireless control plane area featuring two Sub-6 Transport Nodes (TNs) located in NITOS, and an Ethernet based control plane area featuring three TNs, hosted by i2CAT. The two control plane areas are connected using at Layer 2 using an Ethernet over GRE tunnel (EoGRE) over the Internet. The Inter-Area Transport Node (IATN) is connected at the NITOS side of the EoGRE tunnel. Notice that the EoGRE tunnel affects the data-plane performance achievable in this setup, but we consider this acceptable since the focus of our evaluation is on the control plane. The 5G-XHaul control plane is implemented in a set of Virtual Machines (VMs), including the two L0-controllers, the L1-controller and the Top controller. The L0-controller and the L1-controller VMs are hosted in the i2CAT cloud, whereas the Top controller VM is hosted in NITOS. Finally, we have one ETN, along with its associated VM, connected in each control plane area (NITOS and i2CAT), which we will use to establish the end-to-end connection. ETN1 and the IATN communicate with the Top controller directly in NITOS, while ETN2 communicates with the Top controller using an OpenVPN tunnel over the Internet.

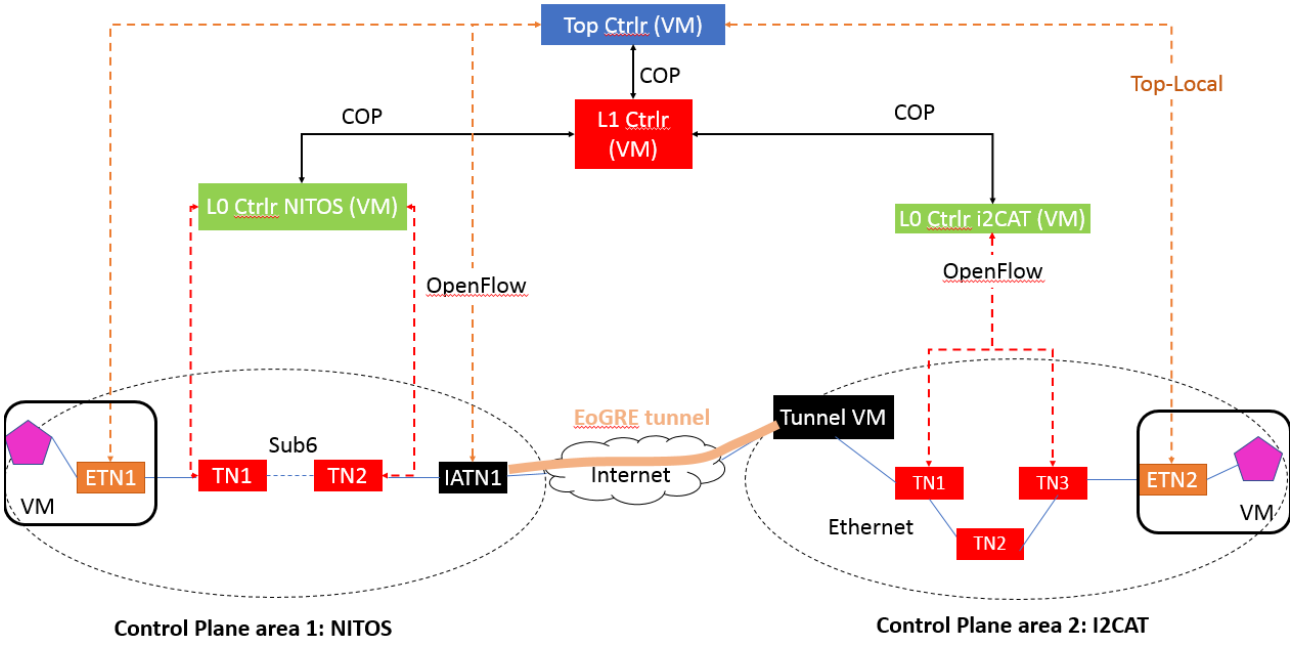


Figure 2-7. End-to-end control plane for evaluation.

Using the previous testbed we carry out a set of measurements to establish the overall time required to establish an end-to-end service connection. In particular, we measure: i) the time required by the Top controller to recover the aggregate network topology delivered by the L1-controller, and ii) the time, as measured in the Top controller since it requests an end-to-end service call between ETN1 and ETN2, until the L1-controller confirms that the end to end path is established, and iii) the overall service-call provisioning time, including the time the Top controller takes to program the ETNs and the IATNs. We execute 100 different tests and report the resulting CDFs in Figure 2-8 and Figure 2-9.

We can see in Figure 2-8 how the completion times for both the COP *service-topology* (left graph) and the COP *service-call* (right graph) primitives take around 200 ms. Notice that both of these primitives require the L1-controller to partition the end-to-end request into two individual requests for each L0-controller, in one case to retrieve the topology and in the other case to instantiate a pair of unidirectional label switched paths. The execution time of the *service-call* requires a longer completion time because in this case the L0 controllers also need to program the TNs in each control plane area. Finally, it is worth noticing that the provisioning times reported in this experiment represent a worst case scenario, because the L2 and L1 controllers, as well as the L0 controller and the TNs in the NITOS domain, are physically located in different data centres (DCs) – NITOS and i2CAT – with an average internet round trip latency that we measure to be around 50 milliseconds. This

internet latency therefore is added to the reported provisioning times, and would not be there for example in the case of a 5G-XHaul infrastructure provider that could host all the control plane elements in the same DC.

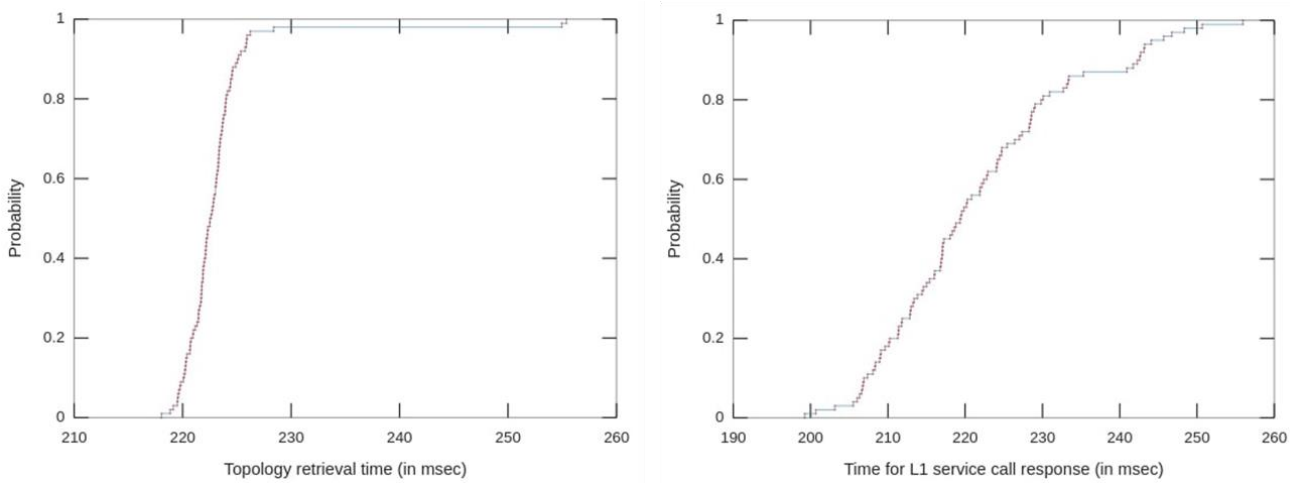


Figure 2-8: CDF of SDN provisioning times over the 5G-XHaul heterogeneous network.

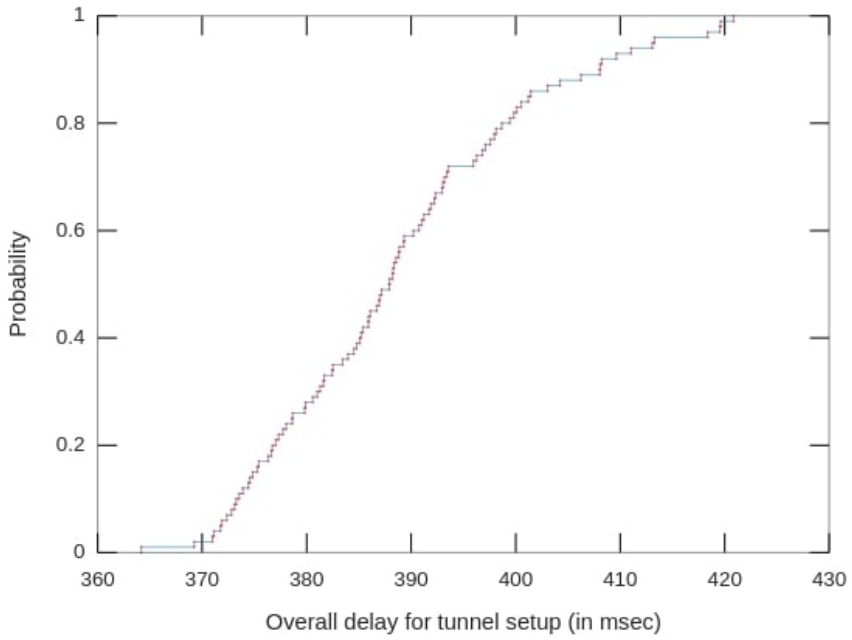


Figure 2-9: Overall service-call provisioning time.

After analysing the delays incurred by the L1 controller we report in Figure 2-9, the overall delay incurred by the 5G-XHaul control plane to establish a connection between ETN1 and ETN2 in our testbed, across the NITOS and i2CAT control plane areas. In addition to the L1 service-call provisioning time included in the right part of Figure 2-8, Figure 2-9 includes the time required by the Top controller to bind ETN1, ETN2, and the IATN to the label switched paths provisioned by the L1 and L0 controllers. The aforementioned internet latency, also affects the communication between the Top controller, hosted in NITOS, and the ETN2 hosted at i2CAT, however we observed an overall connection establishment time below 500 milliseconds, which is negligible compared to service provisioning time in current IP networks, which often require manual intervention.

Finally, we verify the data-plane connectivity between the two domains by reporting two Wireshark captures in the two interfaces of the IATN function located in NITOS. The left graph in Figure 2-10 depicts a packet capture of an ICMP packet in the NITOS control plane area before entering the IATN. We see that following the 5G-XHaul virtualisation mechanisms, this packet comes encapsulated with an outer Ethernet MAC header with a VLAN tag used to signal the label switched path within NITOS; in this case VLAN=30 is used as Path-

ID. In addition, the inner MAC header has an additional VLAN used as Virtual Layer 2 Segment ID, which will be used in ETN2 to deliver the packets to the proper VM. The right graph in Figure 2-10: Dataplane packet captures when entering and leaving the IATN in NITOS depicts the same ICMP packet when leaving the IATN, where we can see how the IATN has modified the outer VLAN tag to VLAN=40, which is the identifier of the label switched path established in the i2CAT control plane area, which will deliver packets to ETN2.

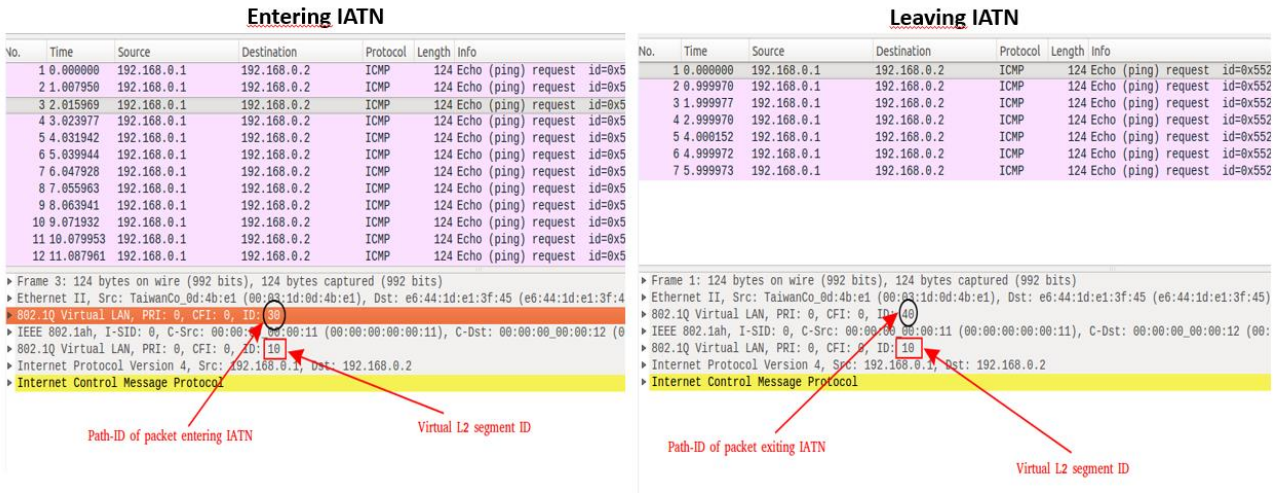


Figure 2-10: Dataplane packet captures when entering and leaving the IATN in NITOS.

2.2 Controller placement and TN assignment

As we have already shown in deliverable D3.1 [1], the control plane is hierarchical and layered. However, each layer consists of clusters of synchronised controllers, apart with other individual controllers, instead of using only the latter ones. The controllers of each cluster share a common view of a subset of the SDN controllers or devices, let's say entities from now on, belonging to the lower level. The existence of clusters is not depicted in the hierarchical structure, because each cluster is abstractly presented as a single controller responsible for all the underlying entities. The rationale of using controller clusters instead of single controllers is their support for extra robustness and efficiency. In this way, there are no single points of failure and the controllers are closer to the underlying entities. However, there is a hidden cost of using clusters, which is the extra control traffic required for their synchronisation.

For example, each TN (or L0) controller could be either single entity, or a cluster of controllers, responsible for the TNs of a particular transport domain. The placement of the controllers belonging to a cluster is not a trivial task. They should be as close as possible to the majority of the underlying TNs, hence we choose as location of the controllers the location of some of the TNs. Moreover, we assume that only one controller will be placed at each TN, since the majority of the TNs are network equipment with limited CPU processing power, not feasible for running efficiently multiple controllers. The scenario of collocated DCs with the equipment, available for hosting more than one controllers, is not feasible in the use cases we investigate in 5G-XHaul, since the network equipment consists mostly of wireless devices deployed in the city centre. The investigation of the optimal number of controllers is a challenge, since the utilisation of multiple controllers enables their distribution and moves them closer to the TNs. However, their synchronisation requires individual communication channels between them, which increase exponentially with the number of the controllers. Thus, we have to balance between the inter-controller and controller to TN traffic, focusing on the minimisation of the total required bandwidth for the control traffic and dealing appropriately with the effects of the contradictory tendencies of either centralising or distributing the control plane. The centralised control plane refers to the utilisation of as few as possible controllers, while the distributed control plane leverages on multiple controllers.

In this section we present the 5G-XHaul solution of the TN (or L0) controller placement problem. We model the control plane of the Areas, presented in Figure 2-1, that is in-band with the data plane, for resolving the optimal controller placement and TN assignment. We formulate the control plane as an undirected connected graph G = (S, L), where S represents the set of TNs and L represents the set of network links. Let $S = |S|$ and $L = |L|$ be the number of TNs and the number of the links respectively. We assume that the path connecting the couple of TNs s1, s2 $\in$ S is p(s1, s2) and the number of links included in this path is |p(s1, s2)|. Finally, C $\subseteq$ S is the subset of the TNs that have collocated controllers, and $c^s \in C$ is the controller that TN s $\in$ S is

assigned to. Vector c = ($c^s \in C$: $s \in S$) describes a controller placement and assignment, where each vector ' s coordinate maps to a TN $s \in S$ and the vector's value indicates the corresponding controller $c^s \in C$.

On the one hand, we aim at minimising the required bandwidth for the controller to TN (Ctr-TN) traffic from all network links, noted as

$$B^S \triangleq \sum_{s \in \mathcal{S}} \sum_{l \in p(s,c_s)} b^s = \sum_{s \in \mathcal{S}} w^s b^s,$$

where b^s is the bandwidth required for the Ctr-TN traffic exchanged between TN s and controller c^s, while $w^s = |p(s, cs)|$ is the length of the path connecting s to c^s. The decrease of B^S depends on the decrease of the weights $w = (w^s : s \in S)$, which happens when there are multiple distributed controllers close to the switches.

On the other hand, we aim at minimising the bandwidth requirements for the inter-controller (Ctr-Ctr) traffic from all network links, noted as

$$\begin{aligned} B^C &\triangleq \sum_{c_1 \in \mathcal{C}} \sum_{c_2 \in \mathcal{C}-\{c_1\}} \sum_{l \in p(c_1,c_2)} b^{(c_1,c_2)} \\ &= \sum_{(c_1,c_2) \in \mathcal{C}^2} w^{(c_1,c_2)} b^{(c_1,c_2)}, \end{aligned}$$

where $b(c_1,c_2)$ is the bandwidth required for the Ctr-Ctr traffic sent from controller c_1 to controller c_2, while $w(c_1,c_2) = |p(c_1, c_2)|$ is the length of the path connecting the two controllers c_1 and c_2. The reduction of B^C depends on the placement of fewer and more centralised closer controllers.

We present what is the optimal controller placement C^* and assignment $c^* = (c^*_s \in C^* : s \in S)$ that minimises the total bandwidth required for the control traffic

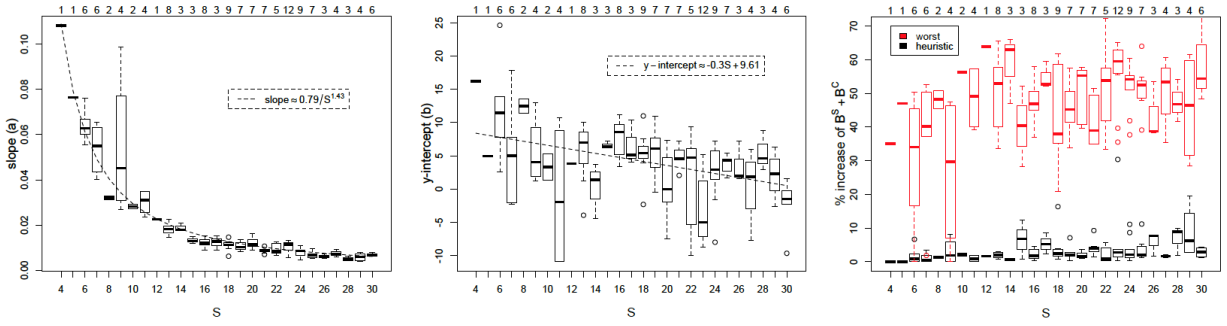
$$\begin{aligned} \mathbf{c^*} &\triangleq \arg\min_{\mathbf{c}}(B^S + B^C) = \\ &\arg\min_{\mathbf{c}}\Big(\sum_{s \in \mathcal{S}} w^s b^s + \sum_{(c_1,c_2) \in \mathcal{C}^2} w^{(c_1,c_2)} b^{(c_1,c_2)}\Big). \end{aligned}$$

After long experimentation in the NITOS testbed [10], with multiple network topologies given by the Internet Topology Zoo collection [11], we observed the dependencies of the optimal solution, as it given by R and CPLEX using Integer Quadratic Programming (IQP). In general, IQP is NP-hard, and as a result, we expect that the problem's time complexity is not polynomial to the network size S. This is also verified through our extensive IQP solving for the tested network topologies, which lasts for exponential time as the network size increases. Hence, we present a heuristic for retrieving quickly near to optimal solutions. For the examples above, we assume that all network switches have almost the same magnitude of flows f and each switch sends β^s traffic to its controller for each flow. Thus, $b^s = f\beta^s$, $\forall s \in S$. Moreover, we define as β^c the bandwidth required for the traffic sent from one controller to another, for each switch assigned to the first controller ($b^{(c1,c2)} = \sum_{s \in S:cs=c1} \beta^c$, $\forall (c_1, c_2) \in C^2 : c_1 \neq c_2$). In particular, we find that there is a dependency of the optimal number $|C^*|$ of controllers on the network size *S*, as well as the fraction β^s / β^c.

$$\begin{aligned} a = 0.79/S^{1.43} \;\&\; b = -0.3S + 9.61, \\ C^H = \big(a(f\beta^s/\beta^c) + b\big)S. \end{aligned}$$

The fraction $f\beta^s / \beta^c$ is analogous to the fraction of the Ctr-TN to Ctr-Ctr traffic, since β^s and β^c give the order of magnitude of the corresponding traffic.

After answering the question how many controllers will be placed, that is CH according to our heuristic approach, then we answer where these controllers should be placed, using the betweenness centrality metric. The CH most central switches will be the locations of the used controllers. Figure 2-11 illustrates evaluation results of the presented heuristic approach. The first two subfigures show how we concluded to the formulas, giving the a and b variables, while the third subfigure shows the gaps between the performances of the optimal placement, our heuristic placement and the worst placement. In particular, the black boxes illustrate the increase of the control traffic of the heuristic placement, comparing to the optimal placement, while the red boxes illustrate the corresponding increase of the worst placement

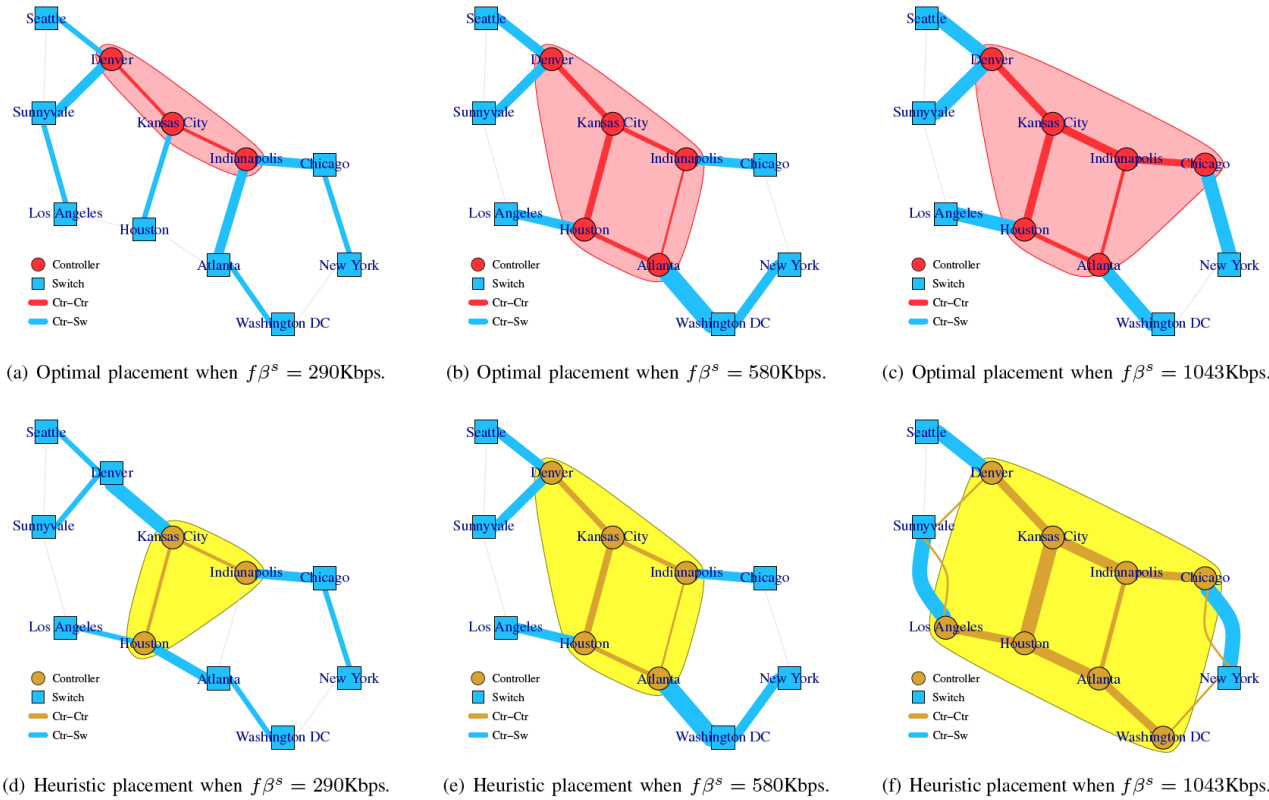


(a) slope (a) for each network size S. (b) y-intercept (b) for each network size S. (c) % increase of $B^S + B^C$ for each net. size S.

Slope (a), y-intercept (b) and the % increase of $B^S + B^C$ for various networks of size $S = 4, 5, \ldots, 30$. The number of the used network topologies of a particular size S is given on the top axis of each of the three plots. The dashed lines in the first two plots illustrate the relationship between the medians of the corresponding boxplots and the network size S.

Figure 2-11: The results of the evaluation of the proposed heuristic placement.

Figure 2-12 illustrates the application of our heuristic controller placement in an Area of the exemplary *Eixample* neighbourhood in Barcelona [5]. In one of the Areas in this example, the TN network topology is the "Abilene" network topology of the Internet Topology Zoo collection. The figure shows which are the optimal and heuristic controller placements for different network conditions.



(a) Optimal placement when $f\beta^s = 290$Kbps. (b) Optimal placement when $f\beta^s = 580$Kbps. (c) Optimal placement when $f\beta^s = 1043$Kbps.

(d) Heuristic placement when $f\beta^s = 290$Kbps. (e) Heuristic placement when $f\beta^s = 580$Kbps. (f) Heuristic placement when $f\beta^s = 1043$Kbps.

Visualization of the optimal and heuristic controller placements on the "Abilene" network topology for $\beta^c = 41$Kbps and various $f\beta^s = 290$, 580 or 1043Kbps. The colour of each link indicates whether it is used by *Ctr-Ctr* or *Ctr-Sw* traffic. The weights of the links are proportional to the bandwidth requirements of all the *Ctr-Ctr* and *Ctr-Sw* connections using them.

Figure 2-12: Comparison of the heuristic and the optimal controller placement.

2.3 Transactional Network Update Support

2.3.1 Motivation and Problem Statement

As discussed in previous sections, the SDN control plane in 5G-XHaul is distributed (hierarchical, layered). Besides, every cluster controller, even though being a single logical entity, could be, in fact, a set of controller elements running on a DC, possibly dynamically adjusted. There are two basic effects which can be derived from these observations:

1. The switches at the edge of each defined area, i.e. in 5G-XHaul terms, e.g. of a reliability cluster or ingress/egress TNs of a control plane area, are logically controlled by both their cluster/control plane area controller and by the higher layer controller (e.g. L1-Controller), which sees them as a port of a "big switch", for example.
2. The distributed, possibly adaptive, cluster controller instantiations, might lead to a situation, where a switch within the cluster is assigned to several controller elements at once.

To support this, in OpenFlow SDN, a switch is already allowed to have several transport layer connections to different controllers (interesting for case 2, possibly for case 1 as well). However, due to the limitations of current OpenFlow SDN (v1.5), only one of these connections can be marked as Master controller, thereby allowed to both write and read the switch flow tables. All other controllers can only read the status. It means that any other controller, if interested in changing the flow table state of a switch, needs to contact the master controller, e.g. using interfaces as we defined them in 5G-XHaul (COP). Similarly, in case of a failure (problem on the control path or in the serving end node, e.g. due to rescaling or hot migration, etc.), the switch needs to renegotiate a master controller. Both situations lead to an increase of latency in runtime and, possibly, to service interruptions. More generally, the need for synchronisation between controller elements limits the scalability of the solution, as it spoils the positive effects of decentralisation.

To improve both runtime latency and system resilience, ideally, several controller elements should be able to control the switch. As observed in SDN-related research [56][57], this would allow generally simpler and more robust control plane deployments and would also contribute to a better scalability of the overall solution. For instance, load balancing and failovers would be much easier to implement in this situation. On the downside, however, having several master controllers leads to obvious conflicts: what if both controllers decide to change the very same flow rule in two different ways?

Interestingly, this problem is not limited to a distributed control plane, like in 5G-XHaul. A similar effect can also be observed in a small scale, single physical controller setup, if the controller executes several independently developed control applications. Fundamentally speaking, the conflict here leads to an inconsistent policy deployment and is related to the problem of concurrency [58][59]. To avoid this class of problems, similar to what is being done in distributed data base management systems (DBMS), SDN has to support *isolation*, e.g. by implementing some form of serialisation of requests.

Another important problem space in SDN becomes visible when one considers a network update encompassing not one but several switches. A typical example could be an e2e path (re-)allocation, e.g. in case of mobility, TE or new sessions. Another example could be a distributed gate keeper, updating several edge switches to allow some traffic according to the current situation. Independently of the concurrency, what could easily happen here is that some clusters/switches could accept, while others might reject the new policy. Just like with the previous problem class, this would lead to an overall inconsistent policy update, e.g. a partial path setup is in principle useless; filtering rules installed but in a part of concerned edges are a security problem, etc. To avoid such situations, the SDN has to support *atomicity* of network-wide updates: whatever the scale, a network-wide update should either succeed completely or fail completely, i.e. without leaving any partial configuration.

In summary, it is the concurrency of such network-wide updates, to be made atomic first, which has to be resolved. In other words, the SDN should support ACID transactional properties, i.e. atomicity, consistency, isolation and durability of critical network policy updates.

2.3.2 State of the Art and Related Work

Originally, SDN was proposed as an extension of a campus network. The updates used to be hardly more than an initial configuration means. In particular, the applications were not critical. In the context of 5G-XHaul this is different. The system using the proposed 5G-XHaul control plane is actually the 5G control plane, and many of its typical tasks each would happen in runtime and would appear as critical. In the SDN research, only

consistency of policy updates has so far received a considerable attention. Concurrency is considered a problem, since it was introduced in [58], yet without technical details. A solution relying on time synchronisation of all switches was proposed in [59]. Integrated in OpenFlow v1.5, it comes with a relatively high burden of time synchronisation of network devices, particularly difficult during network bootstrapping. More problematically, the solution so far only time-serializes individual switch updates, i.e. it does not provide nor does it consider network-wide atomicity. Therefore, even with this solution, concurrent network-wide updates are not always treated correctly.

2.3.3 Our Proposal

We analysed the overall situation and proposed a new, clean architecture extension to support transactional updates in SDN. We considered different implementation options, both in the sense of solution generality and ease of implementation. The resulting architecture is shown in Figure 2-13. The corresponding implementation for the FloodLight controller and OpenvSwitch is available for download [55][2].

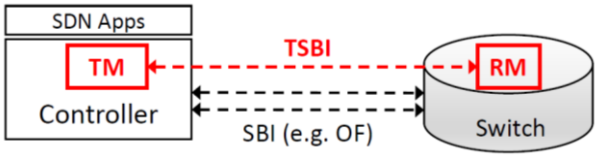


Figure 2-13: Transactional SDN architecture extension.

As shown in Figure 2-13, an extension of the southbound (SBI) interface, which we refer to as Transactional SBI (TSBI) is required. We show that TSBI can be achieved without any changes to the OpenFlow protocol. Our implementation can use OpenFlow v1.4 and higher without any protocol changes, yet with a slightly different local interpretation of the existing messages.

Since the implementation of the Resource Manager (RM) turns out to be relatively simple, we put the RM on the switch and the Transaction Manager (TM) on the controller. This choice has an advantage of being suitable for both single, decentralised and fully distributed controller setups. Equipped with a local RM, the switch now supports several master controllers without conflicts. Hence, it is possible to execute one control application on say controller 1 and another control application on controller 2 such that both controllers can work on an arbitrary, possibly overlapping, set of switches. This greatly simplifies control plane scalability (e.g. it removes many constraints from the controller placement, it allows seamless elasticity, etc.). This design also is suitable for hierarchical setups, both in the sense that it efficiently supports non-strict hierarchies, and because it can be used at any layer. For instance, the RM could be put on the cluster controller, while the TM could be put on next higher layer controllers. The pattern could be repeated on each layer, on which concurrency and wide scope are expected.

Our TM uses 2-Phase Commit (2PC) as an atomic commit protocol (ACP) to coordinate RMs. Our RM currently uses strong strict two-phase locking (SS2PL), which is known to be the most pessimistic scheduler, therefore providing the highest isolation levels. Note that weaker guarantees could be supported by having different schedulers integrated into the RMs, like it is usually the case with DBMS. Our choice of the most general and most pessimistic scheduler is on purpose: while it supports the highest isolation class, it is suitable for all critical applications; at the same time, the evaluation of this scheduler can be understood as a worst-case base line: by using multi-level isolation, a better performance can be achieved in practice as a trade-off against weaker guarantees. The DBMS research has established that this combination of ACP and scheduler permits to extend the local schedules to the system level. To achieve durability, we disallow “self-expiring” flow rules, and instead require an additional transactional “flow removal cycle”. This guarantees that no flow rules disappear, while the network is modified as a whole for both addition and deletion. All in all, we achieve atomicity, isolation and durability. Our method can be easily integrated with any previously proposed consistency methods.

For a more detailed description and analysis we refer an interested reader to our published work [54].

[2] The implementation can be downloaded from https://github.com/OVSacid/ACIDRepo (May 2018)

2.3.4 Performance Evaluation of the Proposed Solution

Here, we present some details of our performance evaluation. The most urgent question that arises with regard to the transactional support in SDN is the question of performance impact. Specifically, our implementation uses 2PC/SS2PL, which therefore involves strict locking of all switches involved in the update for the duration of the update. It means that with the growing number of switches involved, while these switches would not be available for other updates, they would stay locked longer. Therefore, one could reasonably argue that the system could starve itself, meaning that the throughput of (successful) network updates could decrease sharply at some scale, while the latency of individual updates might increase, especially with the rate of incoming requests.

We used our implementation [55] in FloodLight/Mininet to first measure real values. For this, we created simple linear topologies involving *d* switches and allocated paths over them end to end. We then measured the lock time distributions (CDF) and got the results as presented in the boxplot in Figure 2-14 (the box waist is the median, the limits of the box are quartiles (25^{th}/75^{th} quantiles) and the black bars represent the upper and lower whiskers). As can be seen, even for *d*=10 switches involved in one atomic update, the overall lock median stays under 1 ms. (Note that presuming close-to-shortest path allocations for an end-to-end transport, *d* roughly corresponds to the network diameter. In real-world telecom networks, it scales logarithmically with the network size; for instance, the Internet has a diameter substantially lower than *d*=10.)

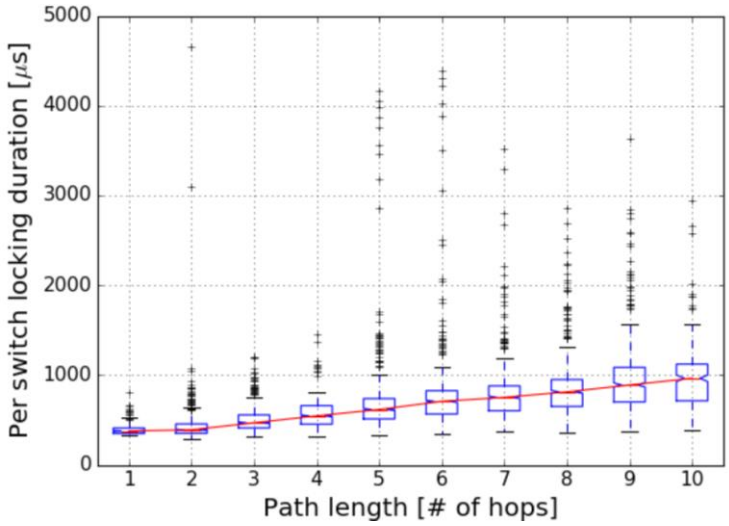


Figure 2-14: Lock duration per switch as measured on the implementation (Floodlight/Mininet).

We use these measured distributions and feed them into a simulation, in which we allocate paths between two randomly chosen access ports of a telecommunications network. We use hierarchical network topologies of two sizes (medium and large) from [60] and [61]. These represent typical telecommunication backhaul networks (access, aggregation, and core). The medium (in parenthesis: large) topology has 160 access (360), 64 aggregation (144) and 8 core (12) switches. In the illustrations below, the medium topology is referred to as *k*=8, and the large one is referred to as *k*=12, where k is parameter for topology generation. The medium network has a diameter *d*=7, and the bigger network has a diameter *d*=8. Note that we on purpose use topologies badly suitable for locking, as they feature only few core network switches, hence quite often locked in the process and blocking other paths. Other, more balanced networks should exhibit better values.

In Figure 2-15, we show how the success rate drops with the increasing arrival rate of requests. As the drop per se was to expect, we implemented and tested a simple back-off mechanism, which reschedules the request after a random backoff interval up to 3 times. The results with backoff (less than 5% of drops at 1000 requests/sec) make us believe that transactional SDN can be implemented in real-world telecom backhauls.

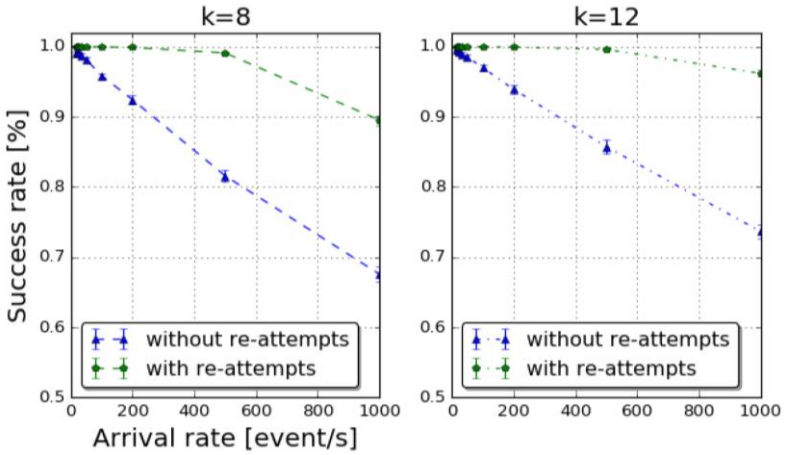


Figure 2-15: Success rate for different request arrival rates (median).

An even better performance can be expected with the increased degree of freedom, i.e. with more redundancy, i.e. here with more available paths between two destinations. To verify this, we slightly change the topologies above by providing additional links between aggregation and core. While in the original topologies above there are exactly *n*=2 such links, we increase this number to *n*=4. The corresponding success rate is shown in Figure 2-16. The success rate increases by about 15% because of these additional links. With backoff it almost reaches 100%.

Finally, we also measured the path setup time. The results hold for the large, unchanged topology and high request rate. We observe that the path setup of around 75% of the updates is below 6 ms and that the 95-quantile of the path setup time is below 30 ms.

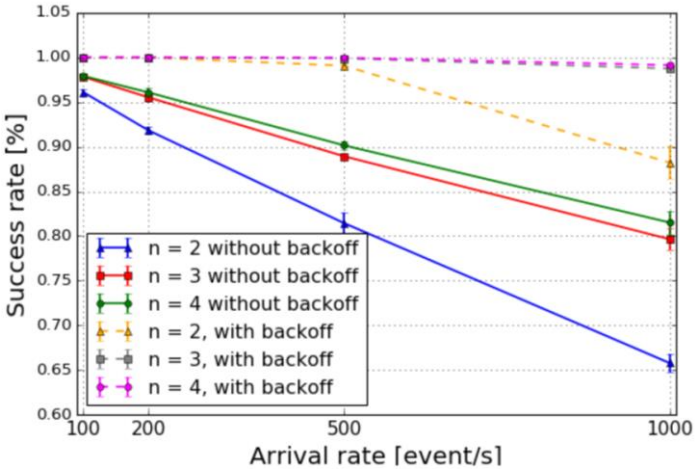


Figure 2-16: Success rate for denser medium topology.

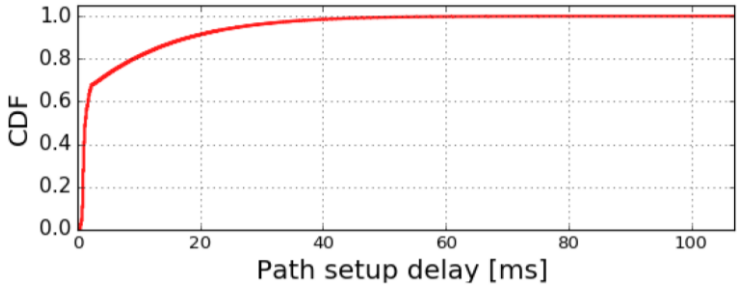


Figure 2-17: CDF of the path setup (k=12, n=2, 1000 requests/sec).

Overall these results are promising. In the future we will consider more fine-grained resources (e.g. switch ports could be locked instead of locking the whole switch, like in all of the above). Besides, we want to explore other schedulers, in addition to or instead of the very pessimistic SS2PL.

2.4 Predictive Backhaul scheduling

In deliverable D3.2, joint backhaul and access radio resource management for Sub-6 GHz was investigated. This contribution complements that by providing joint backhaul and access RRM for dense mmWave RAN deployments.

The rapid-increasing coverage and throughput demand for facilitating enhanced mobile broadband (eMBB) services suggest the evolution of wireless access towards mmWave bands (30-100 GHz), exploiting the benefits of ultra-high bandwidth that can be offered at these bands. However, mmWave signals suffer from increased isotropic free space loss, higher penetration loss, and propagation attenuation due to atmosphere absorption of oxygen molecules and water vapour, resulting in outages and intermittent channel quality. Therefore, higher antenna gain is required at both transceiver sides, where directional transmissions have impact on radio resource usage, multiple access, and interference characteristics, impacting RRM design.

Prior literature in mmWave BH/access RRM primarily focused on the resource optimisation in multi-hop mmWave RANs. In [13], a joint BH and access optimisation mechanism is proposed, which supports multiple simultaneous transmissions to exploit spatial multiplexing gain and allow flexible adaptation of resource usage including transmission duration and power allocation of different links. Several routing and scheduling schemes are proposed in [14] and grouped considering different implementation options, i.e. performing routing and scheduling separately or jointly, in distributed or centralised manner. To this end, the authors in [16] investigated the joint routing and scheduling problem for 5G mmWave BH, formulating it as a capacitated routing problem, where a central depot allocates paths towards particular destinations. This concept was further elaborated in [15] where multiple slices are sharing the same mmWave bands and different Key Performance Indicators (KPIs) necessitate sophisticated routing and scheduling to optimise the perceived service performance.

This study aims to optimise the performance of eMBB services assuming an ultra-high capacity and low latency, while maintaining a stable performance allowing an optimal QoE. It adopts the notion of service prediction and user mobility from [9] but, instead of providing feedback to the application, it uses such information to optimise the network resource management and QoE considering a multi-hop ultra-dense network environment. In addition, it investigates the means to handle control-plane delays due to increased complexity and signalling related to routing and scheduling for high mobility scenarios, and manage efficiently radio fluctuations subject to the access conditions.

The system under study assumes a downlink multi-cell OFDMA cellular network that consists of a dense deployment of mmWave Access Nodes (ANs), which may serve also as relay nodes to neighboring APs under a macro-cell umbrella environment as illustrated in Figure 2-18. In such multi-hop mmWave BH, routing information is collected at a centralised path computation unit (CU), or at the Macro-BS, in order to select paths in a longer time-scale based on traffic load, spectrum-efficiency and other criteria. In addition, due to half duplex constraints and the limitation of the number of beams that can connect small cells and serve users, it is efficient to perform end-to-end scheduling of BH and access in time/frequency and spatial resources (e.g. by beam steering). The entire network consists of a set of macro-cell sites co-located with Mobile Edge Computing (MEC) servers for local processing and storage of user's traffic. A complementary underlining network of m=1, 2, .., M mmWave APs, equipped with directional antennas for the access and the wireless BH is deployed to boost performance in hotspots. Here to mention, that also traffic can be cached to selected APs which include activated gravity points (a.k.a. clusters, based on the similar Least-Action Walk (SLAW) model [33].

We have a set of users *U*, which can be served by different APs for a period of time; hence *u (m,t)* refers to a set of users that is served by AP *m* at time *t*. It is assumed that the mobile users can be either inactive or they have ongoing sessions; however only the connected users with ongoing sessions are assumed within the scope of this study (a user arrives at the system as soon as he starts a session and departs as soon as the session is finalised). The total area is arranged into *i=1,2,..I* gravity points based on the SLAW clustering logic [33]. Each cluster can be served by a single BS, while each BS may include one or more clusters. The spectrum which is allocated to each BS for wireless access and BH is shared in time domain based on the number of active beams towards other nodes (UEs or APs) and the demand of different links.

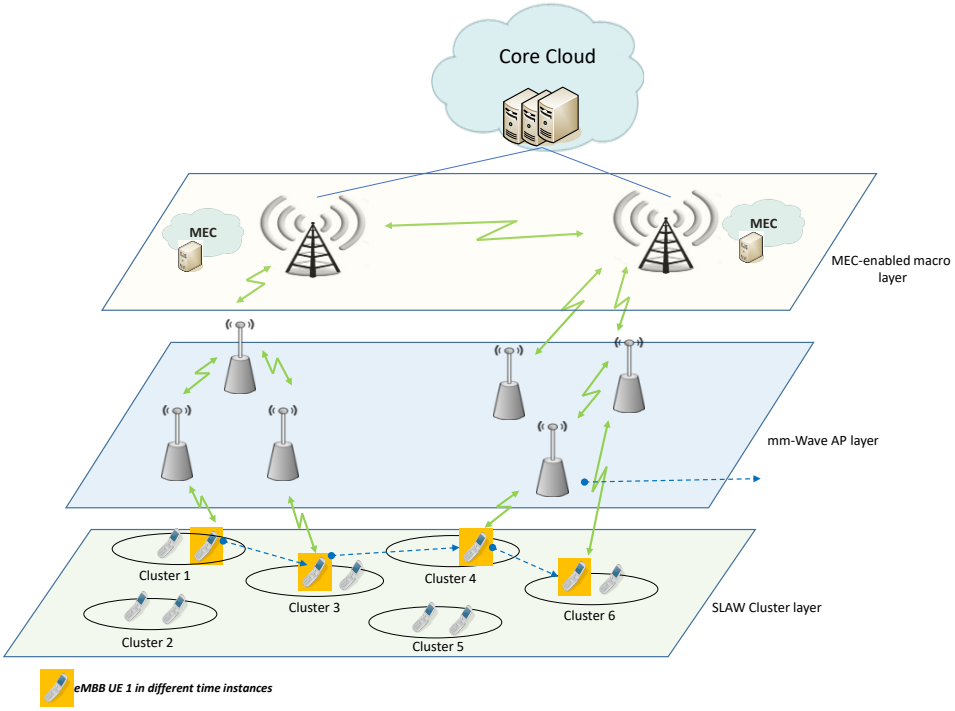


Figure 2-18: A simplified system model.

Let *G (V,E)* be the graph consisting of a set of *V* access and BH nodes and a set of *E* edges. An edge $e \in E$ $\mathrm{e} \in \mathrm{E}$ is a connection between two nodes$v2 \in V$ $\mathrm{v1}, \mathrm{v2} \in \mathrm{V}$. The presence of an edge e indicates that data can be exchanged between *v1* and *v2*. We assume *I* links in the network, where the links are considered un-directional (considering the downstream only). The macro-cell nodes are considered as root nodes denoted as $v0 \in V$ and it is assumed that each AP can have a link towards a single macro-cell site. The time is divided in *f=1, 2,.., F* frames and each frame is further divided $\forall f$ in $t = 1{:}T$ timeslots. It is assumed that we employ space and time-division duplexing between BH and access links, while re-using all spectrum band by each activated link. Therefore, every timeslot of the frame can be used either for BH or access links. Beamforming is employed, i.e. beams with high antenna gain can be formed and pointed to virtually any direction. The limitations of this set-up concentrate on selecting which links to activate in every time-slot, so as to avoid interference taking into account the half duplex constraints and the beam limitations.

To address the aforementioned challenges we will introduce the following contribution in Section 3.3: A probabilistic model for mobility and traffic prediction adopting empirical user mobility and traffic modelling using the Discrete Time Markov Chain (DTMC) to predict the handover sequence and the residence time for each active user with an ongoing session per AP. This model will be used for the contributions which are described in the following sub-sections. In particular:

- The formulation of an end-to-end (e2e) BH and access path optimisation problem, which is divided into two sub-problems: a) multi-hop path selections per time window based on user mobility prediction and traffic demands and b) e2e resource management at the activated links to ensure the desired the KPIs of different services.
- A solution framework that encloses a multi-hop path selection algorithm and RRM mechanism, facilitating also edge caching for eMBB video traffic in order to minimise network load and service latency at the BH.

2.4.1 Joint BH/Access Resource Optimisation Problem Formulation and analysis

The aforementioned prediction model provides information on 1) the period of time where an active user resides within an AP coverage, 2) the sequence of APs that a user will transverse during an active session, 3) the traffic demand of a user per cell and 4) the traffic demand distribution among APs per user. The network resource allocation problem focuses on optimising BH and access resources (in terms of activated links and time division sharing for BH and access) in order to meet eMBB users' KPIs assuming that the network also

supports other types of services. Due to high expected complexity of the conventional joint BH/access resource management problem [9], we decouple it in two sub-problems: a) the user-centric path selection, which can determine the paths required for the duration of the video session based on the channel and load constraints; and b) the scheduling of resources based on the allocated paths in time and power domain. We define the potential traffic demand of a user u over the link *e* as $D_f(e,u)$, to identify the rate requirement that will contribute to the load of that link for each frame, if this link is used to carry the video traffic.A normalised parameter $N_{seg}(m,u)$ represents the demand of a user for the frame based on the number of segments that need to be carried over the link. Each link *e* has an upperbound capacity that corresponds to the maximum rate over that link, defined as $c_e, \forall e \in E$, and a target data rate, which corresponds to the summation of all the user traffic traversing it for each frame, denoted as $\sum_{u\in U} 1_{u(e=\{i,m\},f)} D_f(e,u), \forall t, \forall i \neq m$.

The variable $1_{u(e=\{i,m\},f)}$ is a binary variable which can be derived by the output of the mobility model (in particular $\xi_{m,u}$ is the fraction of time a user resides at an AN, which is normalised to determine the actual frame(s) each user receives by an AP *m*. This variable is 1 if the user *u* resides at cell *m* at frame *f* if and only if *m* the receiving end of edge *e*; otherwise it is 0. Using the notion of link capacity and link load, a new parameter, which captures the cost of the link is:

$$c_{e,f} = \left\lceil \frac{\sum_{u\in U} 1_{u(e=\{i,m\},f)} D_f(e,u)}{c_e} \right\rceil \leq T_{min}, \forall f \in F, \tag{2-1}$$

where $c_{e,f}$, can be interpreted as a cost function of each link, which captures how much delay will be contributed (in terms of time slots t within a frame) to the allocated path (from the macro to the end user) if this link is utilised. This is normalised to a predefined maximum number of timeslots per frame $T_{min} < T$, which is set to ensure that traffic is delivered to all end users from the root nodes within the frame duration.

A. User-centric E2E Path Selection

The objective is to find the paths that the traffic should follow and the links to be activated per frame based on the predictive mobility and traffic so as to maximise the system performance. Here, a path is defined as a tunnel from the Core/Edge Cloud (traffic source or root node) to the end user, comprising multiple hops using mmWave links (as can be seen in Figure 2-18)

$$\min \sum_{f\in F} \sum_{u\in U} \sum_{e\in E} c_{e,f}\, x_e(u,f) \tag{2-2}$$

Subject to:

$$\sum_{f\in F} \sum_{e=\{0,j\}\in E} x_e(u,f) = N(u), \forall u \in U \tag{2-3}$$

$$\sum_{e=\{j,i\}\in E} x_e(u,f) = 1, \forall i \neq 0 \in V, \forall f \in F, \forall u \tag{2-4}$$

$$\sum_{e=\{i,j\}\in E} x_e(u,f) = 1, \forall i \neq 0 \in V, \forall f \in F, \forall u \tag{2-5}$$

$$\sum_{e=\{i,j\}\in E} c_{e,f}\, x_e(u,f) \leq T_{min,u}, \forall u \in U, \forall f \in F \tag{2-6}$$

Here, $x_e(u,f)$ is a binary variable, which is 1 when the link *e* is used by a traffic flow of user *u* at frame *f*, and 0 otherwise. Constraint (2-3) shows that the number of paths assigned for the per user traffic flows for the entire session should be exactly the number of expected handovers *N(u)* (as introduced in section III). In other words, the route changes as soon as the user changes serving AP; and this constraint captures that the per user traffic path originating from a traffic originator needs to be adapted based on the number of predicted handovers; while he has an ongoing session. This allows for user-centric optimisation based on the required changes due to predicted handovers. In (2-4) and (2-8), the number of incoming and outgoing links towards

/from each AP for each users' traffic should be exactly one. Constraint (2-6) is the maximum delay constraint in terms of number of timeslots within a frame, which has to be considered when allocating a path.

B. Multi-service end-to-end BH resource optimisation

Here for the expression of the problem, it is required to derive the link capacities including both BH and access links. This can be generally expressed for each time instance as:

$$C_{i,j}(\beta_{i,j}) = E\left[\sum_{t=1}^{T} a_{i,j,t} log_2\left(1 + \frac{P_{i,j,t} G_{i,j,t} L(d_{i,j}{}^{\rho})}{\bar{I} + \eta}\right)\right], \quad (2\text{-}7)$$

where $a_{i,j,t}$ is a binary variable which is 1 when link *e={i,j}* is active at timeslot *t*; otherwise it is 0. $P_{i,j,t}$ and $G_{i,j,t}$ are the transmit power and channel gain per timeslot from *i* to *j* respectively. $L(d_{i,j}{}^{\rho})$ is the distance dependent path-loss (i.e. d is the distance between *i, j* and *ρ* is the path loss exponent). Furthermore, $\bar{I}$ is the average interference (due to power leakage by other sources) and η is the thermal noise. Also, $\beta_{i,j}$ is the fraction of timeslots from the total number of *T* timeslots per frame where the link should be active. This can be further analysed in order to capture the relation between capacity and the fraction of timeslots as:

$$\begin{aligned} C_{i,j}(\beta_{i,j}) &= E[\sum_{t=1}^{T} a_{i,j,t} log_2\left(1 + \frac{P_{max}\overline{G_{i,j}}\, L(d_{i,j}{}^{\rho})}{\beta_{i,j}(\bar{I} + \eta)}\right)] = \frac{\beta_{i,j}}{T} N log_2\left(1 + \frac{P_{max}\overline{G_{i,j}}\, L(d_{i,j}{}^{\rho})}{\beta_{i,j}(\bar{I} + \eta)}\right) \\ &= \beta_{i,j} log_2\left(1 + \frac{P_{max}\overline{G_{i,j}}\, L(d_{i,j}{}^{\rho})}{\beta_{i,j}(\bar{I} + \eta)}\right), \forall i,j \end{aligned} \quad (2\text{-}8)$$

The objective is to find the optimal fraction of time per link and per user traffic so as to maximize all link capacities based on the pre-defined activated links for each frame, subject to constraints regarding the minimum and maximum rate requirement per link as dictated by the traffic prediction and the forwarding rules. In other words, in previous problem, we find the paths in which the traffic will be sent from root node (Edge/Core Cloud) to end users via a predefined set of hops within a time frame (which consists of fixed number of time-slots). The problem that is tackled here is that as soon as we know the e2e paths and the traffic requirement per session per frame, we need to schedule within the frames when each link should be active (or for which fraction of time the traffic flow should be active) in order to maximise e2e performance. Taking into consideration that this problem applies for each frame independently, this problem can be formulated as:

$$\max_{\xi,\beta} \sum_i \sum_j \sum_{u \in U} \xi_{i,j,u} \cdot x^*{}_{e \to \{i,j\}}(u,f) \cdot C_{i,j}(\beta_{i,j}), \forall i,j \in V, \forall f, \quad (2\text{-}9)$$

subject to:

$$\sum_{j \neq i \in V} \beta_{i,j} = 1, \forall i \in V \quad (2\text{-}10)$$

$$\sum_{u \in U} \xi_{i,j,u} = \beta_{i,j}, \forall i,j \in V \quad (2\text{-}11)$$

$$C_{min,u} \leq \xi_{i,j,u} \cdot x^*{}_{e \to \{i,j\}}(u,f) \cdot C_{i,j}(\beta_{i,j}) \cdot T \leq C_{e \to i,j} \quad (2\text{-}12)$$

where $\xi_{i,j,u}$ is the fraction of time that the traffic flow of user *u* needs to utilize link *i,j* and relates to $\beta_{i,j}$ that represents the utilisation fraction of time per link by constraint (2-11). Constraint (2-10) shows that the sum of β weights from any node towards the others equals one. In addition, (2-12) captures the minimum ($C_{min,u}$ is the minimum required capacity per user) and maximum link capacity requirement.

2.4.2 Solution and Evaluation

The solutions framework comprises two algorithms to cope with the aforementioned problems. One related to the end-to-end path selection problem, wherein each user is mapped into the activated links per frame. The

second concentrating on the allocation of resources in terms of fraction of time slots, which are needed within each frame by each user's traffic flow per link. In particular it is proposed to perform caching of video segments to different APs so as to relax the BH capacity and allow for minimum end-to-end delay.

2.4.2.1 User-centric Path Selection

The objective of this algorithm is to select paths towards UEs (with known traffic demands), originating from particular macro-cells and passing via intermediate APs, so as to maximise the total capacity considering BH conditions and latency, i.e. considering an upper bound number of hops. The algorithm follows a branch-and-bound scheme, where lower bounds are computed by solving a Linear Program (LP) relaxation of the problem. This relaxation is iteratively tightened by adding valid inequalities to the formulation according to the cutting plane approach. The exact method is known as a branch-and-cut algorithm and is thoroughly described in [47] for the case of the Integer Programming (IP) problem. This solution is strongly correlated with the parameterisation of the number of paths, which reflects to the average number of hops which are needed from a traffic originator to the end user. *K-value* was introduced in [16], which can be adjusted to adapt the number of paths. In our case two variants can be defined:

- **BH Rate-Optimal Routing**: Setting of small *K*-value (e.g. K=2), which means that only 2 paths will originate from the macro and need to enclose all the APs with users having ongoing sessions. This is translated to having many hops; however this involves low-distanced LoS links which many offer higher e2e BH capacity. In the other hand this will lead to high delays for the processing, queuing and forwarding of flows. So, an upper time threshold is set within the frame to avoid not meeting the delay bound.
- **BH Delay-optimal Routing**: Setting of large K-value which depends on the deployment (e.g. *K=6* in our topology), and requires the routing of all flows from the traffic originator to be performed via K paths. This may include also longer paths with Non-Line-of-Sight (NLoS) links; hence the e2e BH capacity will be lower. However the on average hops will be lower and the delay will be acceptable.

An exemplary illustration can be seen in Figure 2-19, where CU is the Edge Cloud Server, the red squares are the mmWave ANs and the stars represent the end users.

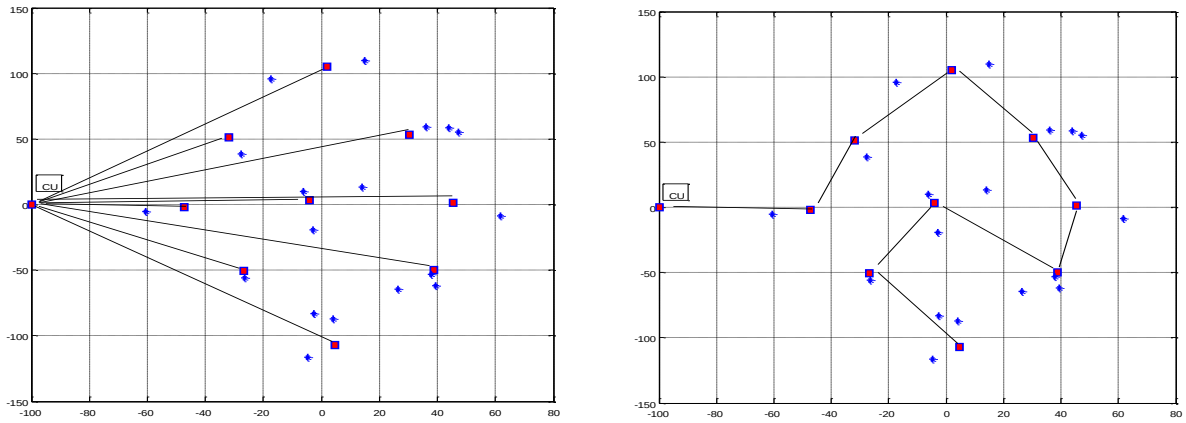


Figure 2-19: Illustration of BH topology for two extreme K-values (k=9 and k=1).

2.4.2.2 Multi-service end-to-end BH resource optimisation

For each type of traffic (here we assume eMBB traffic and non-real time) the packets are stored in separate queues per destination at the macro-cell. Traffic forwarded via more than one hop should be stored in separate queues in the intermediate nodes. The target is to empty all the queues by the end of each given frame. Here, one constraint is related to half-duplexing, i.e. each node can either transmit or receive per time instance, while the other focuses on the maximum allowable delay to deliver a flow. In case of caching at the selected APs for eMBB services (based on the existence of gravity points), the time is updated to relax such constraints. In particular, video segments of an eMBB flow related to the second or later frame can start to be sent from first frame; however this is with lower priority and depends on the actual load of the links and nodes per frame.

For solving this problem, we showcase a throughput optimal algorithm, which follows the back-pressure concept [48]. Assuming a slotted time, the basic idea of backpressure scheduling is to select a set of non-interfering links for transmission at each slot. Non-interfering links refer to links that do not have the same transmitting and/or receiving end, such that the half duplex constraint is maintained. Here, the objective is to serve a traffic flow of user *u* with the maximum differential backlog. In Figure 2-20 we present the algorithm as a flowchart in more detail.

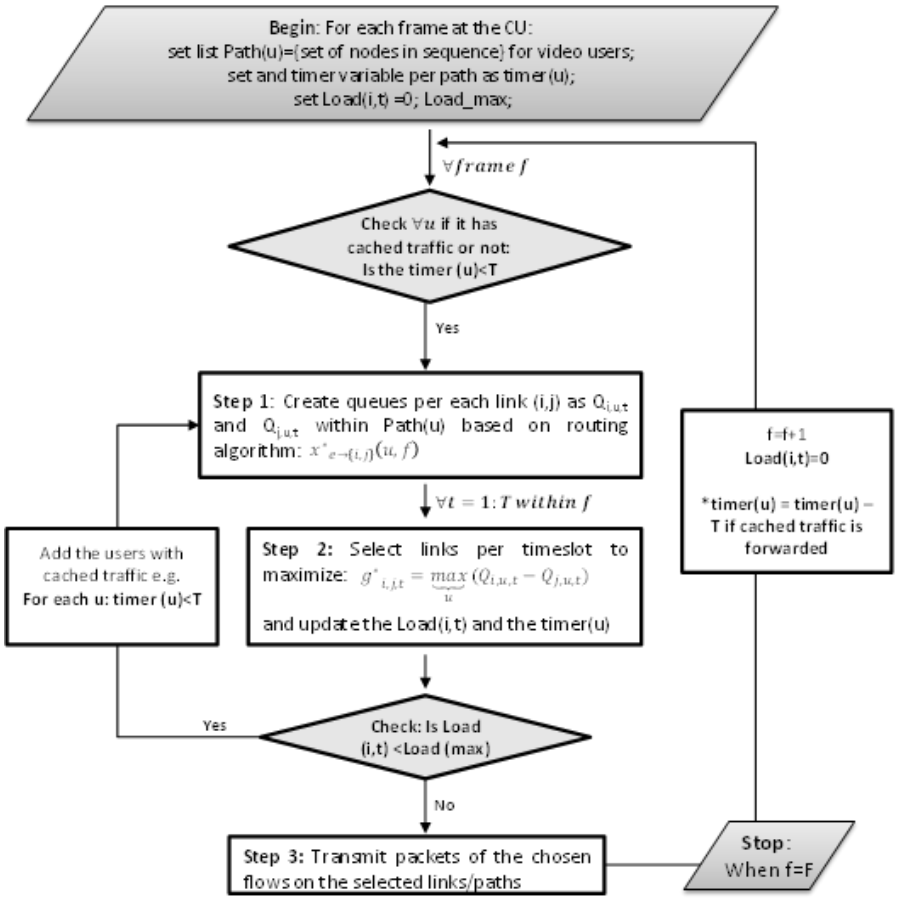


Figure 2-20: Flowchart of the Scheduling Algorithm.

Table 2-1: Simulation Parameters.

APs	9 APs
ISD	20 m
Users	Poisson arrivals per cell (λ=2.5) (medium load per small cell area)
Traffic	50% of users with eMBB video (50 Mb/s per user) 50% non-real time traffic
Carrier	60 GHz
Bandwidth	Case 1: 100 MHz. Case 2: 400 MHz
TTI size / Frame size	0.5 ms / 5 ms
Channel model	For the capacity computation, the 60 GHz path-loss models in LoS and NLoS are used by [49] using LoS probability
Processing Delay	L2 Processing ~ 0.5 ms
Snapshots	5000

To evaluate our proposal, Monte Carlo simulations were performed within a two macro-cell scenario, which covers a random drop of 9 mmWave APs, considering a medium user load (1-4 users) per small-cell area. Table 2-1 provides a summary of the simulation parameters.

Our scheme was compared against the joint routing and scheduling scheme of [16], where two variants where showcased: A) BH Delay-optimal Routing and Scheduling, using K-value equal to 6 (which corresponds to large path sizes) and back-pressure scheduling for the traffic forwarding; hence low number of average hops from the macros to the end user for optimised delay. B) BH Rate-optimal Routing and Scheduling using K=2 and back-pressure scheduling for the traffic forwarding, which corresponds to high number of hops and throughput-oriented scheduling policies.

The metrics used for the evaluations are:

- The *Cumulative Density Function (CDF) of the end-to-end delay* is calculated assuming queuing and processing delays at each node, from the macro-cell to the end user. As shown in Figure 2-21, the e2e delay is much lower in case of *Benchmark B*, while for the proposed approach the e2e delay corresponds to the access delay only (1.5 ms, assuming AP processing and propagation delay).

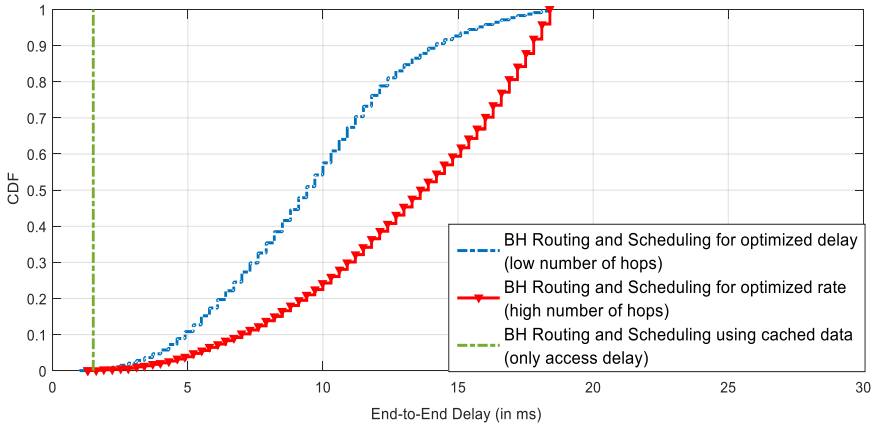


Figure 2-21: CDF of E2E Delay.

- The average mm-Wave AP throughput was captured, assuming two different bandwidth allocations (100 MHz, 400 MHz). As can be seen in Figure 2-22, we observe 30% gain in terms on cell throughput comparing to Benchmark A, while performing similar gain as the Benchmark B for both bandwidth allocation variants.

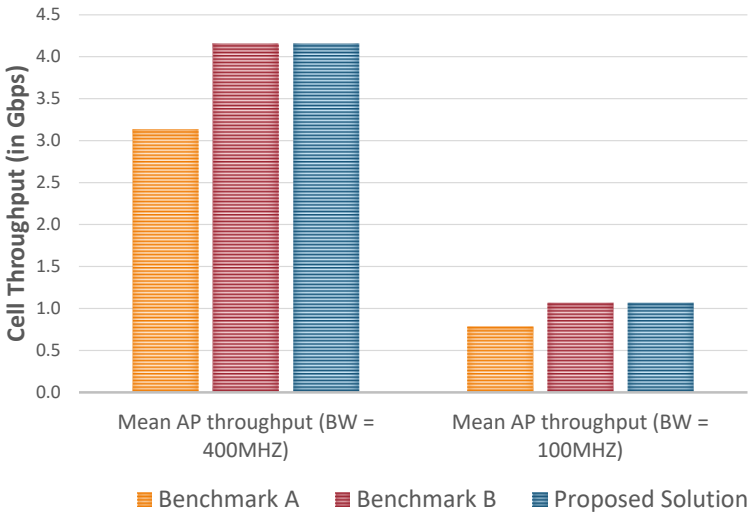


Figure 2-22: Mean Small Cell Throughput for 100 MHz and 400 MHz bandwidth.

2.5 WDM-PON Management Interface

In this section we report the final definition of an operational YANG model and the according implementation of a NETCONF service as proposed in deliverable D3.2 [2], enabling the WDM-PON Optical Line Terminal (OLT) to expose the control and monitoring features listed in Table 2-2 to SDN orchestrators. This feature table is a refined version of the preliminary definitions in deliverable D3.2 [2], which needed adjustments due to the parallel development of optical components. Compared to the previous definitions, two separate commands (marked in blue in Table 2-2) will be required to manage the optical frequencies between the OLT and ONUs.

Table 2-2: SDN features of WDM-PON OLT.

Control items		
Command	**Refers to**	**Action**
Set DS/US frequency	DS/US channel of each OLT network port	Commissioning command Once the DS frequency is set, the paired US frequency will be automatically configured.
Set ONU label	Pilot tone frequency per port	Commissioning command Select a unique PT frequency for the connected remote ONU
Turn off ONU	Specific ONU	Send command on specific wavelength to disable ONU TX
Turn on ONU	Specific ONU	Send command on specific wavelength to enable ONU TX
Monitoring items		
Parameter	**Refers to**	**Details**
RX power	OLT port	Received power on specific wavelength to monitor operation
Optical frequency deviation	OLT port	Grid-deviation of wavelength received from ONU
ONU status	ONU	Status / health (Tx/Rx power, temp.) of ONU, retrieved from the alarm indicator
ONU identifier	ONU	Part/SN number or geo-location of ONU on particular wavelength

The central frequency is defined as in [63]. For each optical channel, different ranges of frequencies are used in the OLT-to-ONU and ONU-to-OLT directions. The optical channel frequencies in the two directions are paired according to Table 2-3.

Table 2-3: Optical frequency mapping table.

OLT-to-ONU (DS) in THz	ONU-to-OLT (US) in THz
194.1	191.5
194.2	191.6
…	…
196	193.4

As for the range of available pilot tone frequencies, the maximum frequency is 52.5 kHz, and the minimum frequency of pilot tone is 47.5 kHz. The minimum frequency spacing of pilot tones should be 50 Hz.

2.5.1 WDM-PON YANG Model

The WDM-PON-specific YANG model is based on the IETF YANG model for network interface management (RFC 7223) [62], and represents every Optical Network Unit (ONU) as an item of the `/ietf-interfaces:interfaces/interface` list. That IETF YANG model is a well-established basic model used by many vendors, and is already well-integrated into many common SDN orchestrators as a core component. Therefore, using such a YANG model for the WDM-PON OLT's management interface reduces the effort of integration with

those orchestrators, and enhances the compatibility with YANG-based management interfaces of other network elements.

However, the basic IETF YANG model only defines common interface parameters and the statistics on the traffic packets, while the WDM-PON system has many additional interface parameters on the PHY layer, as previously defined in deliverable D3.2 [1]. The `adva-wdm-pon` YANG model shown in Table 2-4, therefore augments the `/ietf-interfaces:interfaces/interface` definition ac-cording to those specific monitoring items shown in Table 2-2, and adds the additional control commands as RPC methods.

Table 2-4 below shows the complete YANG model for the OLT of the WDM-PON system, also providing further details about the SDN interface via the included `description` statements.

Table 2-4: YANG model for WDM-PON OLT.

```
module adva-wdm-pon {

  yang-version 1.1;

  namespace
    "http://yang.advaoptical.com/wdm-pon";

  prefix pon;

  import ietf-interfaces {
    prefix if;
  }

  organization
    "ADVA Optical Networking SE";

  contact
    "Stefan Zimmermann <mailto:szimmermann@advaoptical.com>
     Jim Zou <mailto:jzou@advaoptical.com>";

  description
    "This module contains specific network interface definitions for ONUs
     connected to the OLT of an ADVA WDM-PON system.";

  revision "2018-05-14" {
    description
      "Initial revision.";
    reference
      "5G-XHaul Deliverable D3.3";
  }

  typedef port-index {
      type uint8;
      description
        "The zero-based index number of an ONU connection port.";
  }
```

```
typedef channel-frequency {
  type decimal64 {
    fraction-digits 1;
  }
  units "THz";
  description
    "ITU-T optical frequency with 100G spacing.";
}

typedef pilot-tone-frequency {
  type decimal64 {
    fraction-digits 1;
  }
  units "kHz";
  description
    "Frequency for unique ONU labelling.";
}

typedef power {
  type decimal64 {
    fraction-digits 2;
  }
  units "dBm";
  description
    "Optical signal power in dBm.";
}

typedef temperature {
  type decimal64 {
    fraction-digits 1;
  }
  units "degC";
  description
    "Temperature in degree Celsius.";
}

augment "/if:interfaces/if:interface" {
  description
    "WDM-PON OLT extensions for ONU connection interfaces.";

  leaf olt-port {
    type port-index;
    config false;
    description
      "The physical OLT port number.";
```

```
    }

    leaf onu-identifier {
      type string;
      config false;
      description
        "A unique ONU hardware indetifier.";
    }

    leaf downstream-channel {
      type channel-frequency;
      config false;
      description
        "The ITU-T optical signal frequency from OLT to ONU.";
    }

    leaf upstream-channel {
      type channel-frequency;
      config false;
      description
        "The ITU-T optical signal frequency from ONU to OLT.";
    }

    leaf pilot-tone {
      type pilot-tone-frequency;
      config false;
      description
        "The unique ONU labelling frequency.";
    }
  }

  augment "/if:interfaces/if:interface/if:statistics" {
    description
      "WDM-PON OLT extensions for ONU connection statistics.";

    leaf grid-deviation {
      type int8;
      units "GHz";
      description
        "The current deviation from the ITU-T upstream signal frequency.";
    }

    leaf rx-power {
      type power;
      description
        "The current optical signal power received at the OLT.";
```

```
    }

    leaf onu-tx-power {
      type power;
      description
        "The current optical signal power transmitted from the ONU.";
    }

    leaf onu-rx-power {
      type power;
      description
        "The current optical signal power received at the ONU.";
    }

    leaf onu-temperature {
      type temperature;
      description
        "The current ONU temperature.";
    }
  }

  rpc set-downstream-channel {
    description
      "Set ITU-T optical signal frequency from OLT to ONU.";
    input {
      leaf olt-port {
        type port-index;
        description
          "The OLT port number.";
      }
      leaf frequency {
        type channel-frequency;
        description
          "The ITU-T optical signal frequency.";
      }
    }
  }

  rpc set-pilot-tone {
    description
      "Set the unique ONU labelling frequency.";
    input {
      leaf olt-port {
        type port-index;
        description
          "The OLT port number.";
```

```
      }
      leaf frequency {
        type pilot-tone-frequency;
        description
          "The ONU labelling frequency.";
      }
    }
  }

  rpc enable-onu {
    description
      "Enable a connected ONU.";
    input {
      leaf olt-port {
        type port-index;
        description
          "The OLT port number.";
      }
    }
  }

  rpc disable-onu {
    description
      "Enable a connected ONU.";
    input {
      leaf olt-port {
        type port-index;
        description
          "The OLT port number.";
      }
    }
  }
}
```

2.5.2 WDM-PON NETCONF Service implementation

The actual NETCONF/YANG service software is running on a GNU/Linux[3] Operating System (OS) inside the WDM-PON OLT. The service follows a modular approach, comprised of three components connected via Inter Process Communication (IPC):

(1) A YANG configuration data store:
- Holds the current management state of the system according to the `adva-wdm-pon` model shown in Table 2-4.
- Distributes YANG configuration data changes and YANG RPC method request and response data between the other software components.

(2) A NETCONF protocol handler:

[3] https://www.gnu.org

- Communicates with SDN orchestrators via TCP/IP-based Secure Shell[4] (SSH) connections according to RFC6242[5].
- Exposes the YANG configuration data from the data store to the orchestrators.
- Applies configuration change and RPC method requests from the orchestrators to the da-ta store.

(3) The actual WDM-PON-specific management application:

- Controls the hardware components of the WDM-PON system.
- Updates the YANG configuration data store with the state of the hardware components.
- Translates YANG configuration data changes and YANG RPC method requests to actual hardware commands.

Figure 2-23 visualizes the interaction of those components. The modular approach has advantages. First, it offers improved stability compared to a single-process implementation, enabling single components to be restarted independently during the development process and on internal failures while keeping the rest of the system functional. Second, it simplifies the extension of the system, e.g. with components for supporting other YANG-based management protocols like RESTCONF[6].

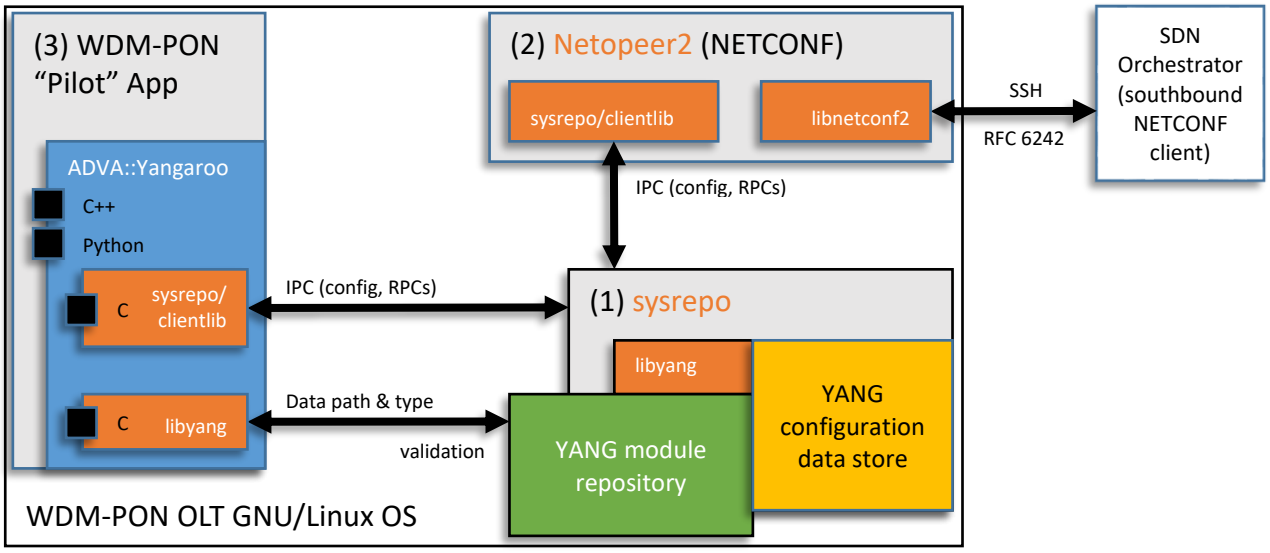


Figure 2-23: ADVA WDM-PON SDN Management software components.

The YANG configuration data store (sysrepo) and the NETCONF protocol handler (Netopeer2), as well as their dependencies libyang and libnetconf2, are third-party OpenSource software projects from CESNET[7] (Czech Educational and Research Network), available from their GitHub accounts https://github.com/CESNET and https://github.com/sysrepo

For efficient programmatic access to the YANG configuration data store, the OpenSource ADVA::Yangaroo software project was created. It offers state-of-the-art C++14+ and Python3+ APIs (Application Programming Interfaces) on top of the conventional C-APIs of the sysrepo client part and libyang, and is scheduled for public release in the end of July 2018. Table 2-5 contains an excerpt of the WDM-PON management application's C++ code, reduced to show some essential Yangaroo features like automatic data type conversions between C++ and YANG, convenient and compact looking YANG tree navigation and data manipulation using C++ operator overloads and property-like tree node object member constructs, and support for lambda functions as RPC callbacks.

[4] https://www.ssh.com/ssh/protocol/

[5] https://tools.ietf.org/html/rfc6242

[6] https://tools.ietf.org/html/rfc8040

[7] https://www.cesnet.cz/?lang=en

Table 2-5: C++ source code excerpt from ADVA WDM-PON OLT Management application.

```
#include <vector>

#include <adva/yangaroo.hpp>

#include "./onu.hpp"

namespace adva { namespace wdm_pon
{
  class Pilot
  {
  private:
    // Internal ONU control
    std::vector<ONU> onus;

  private:
    // Access to ietf-interfaces and adva-wdm-pon modules in YANG
    // configuration data store
    adva::yangaroo::Pouch if_pouch, pon_pouch;

  public:
    Pilot():
      if_pouch("ietf-interfaces"),
      pon_pouch("adva-wdm-pon")
    {
      // Register callback for RPC method enable-onu
      this->pon_pouch.rpcs["enable-onu"] <<
        [this](const auto &input, auto &output)
        {
          const auto index = input["olt-port"].get<std::size_t>();
          this->onus[index].enable();
        };
    }

  public:
    void update()
    {
      // Update /ietf-interfaces:interfaces/interface list with ONUs
      for (auto onu : this->onus)
        {
          // A session allows making several changes in a data store copy
          // before committing them at once to the running data store
          auto session = this->if_pouch.session();

          auto intf = (session["interfaces"] / "interface")
            ["onu" + std::to_string(onu.index())];
```

```
          intf / "type" = "iana-if-type:gpon";
          intf / "description" = "ONU " + std::to_string(onu.index());
          intf / "enabled" = onu.enabled();

          session.commit();
        }
    }

  }; /* class Pilot */

} } /* namespace adva::wdm_pon */
```

Further convenience is currently developed in form of YANG-model-specific C++ API code generation via another ADVA OpenSource project named Alpakka, which will be published along with Yangaroo. It will provide a language-agnostic templating system, which allows YANG-based code generation for virtually any kind of programming language. The C++ code from Table 2-5 will then be further simplified as shown in Table 2-6.

Table 2-6: Future C++ source code style in ADVA WDM-PON OLT Management application.

```
#include <vector>

#include <adva/yangaroo.hpp>
#include "./ietf-interfaces.hpp"
#include "./adva-wdm-pon.hpp"

#include "./onu.hpp"

namespace adva { namespace wdm_pon
{
  class Pilot
  {
  private:
    // Internal ONU control
    std::vector<ONU> onus;

  private:
    // Access to ietf-interfaces and adva-wdm-pon modules in YANG
    // configuration data store
    adva::yangaroo::ietf_interfaces if_pouch;
    adva::yangaroo::adva_wdm_pon pon_pouch;

  public:
    Pilot()
    {
      // Register callback for RPC method enable-onu
      this->pon_pouch.rpcs.enable_onu <<
```

```
        [this](const auto &input, auto &output)
        {
          const auto index = input.olt_port();
          this->onus[index].enable();
        };
    }

  public:
    void update()
    {
      // Update /ietf-interfaces:interfaces/interface list with ONUs
      for (auto onu : this->onus)
        {
          // A session allows making several changes in a data store copy
          // before committing them at once to the running data store
          auto session = this->if_pouch.session();

          auto intf = session.interfaces.interface
            ["onu" + std::to_string(onu.index())];

          intf.type = "iana-if-type:gpon";
          intf.description = "ONU " + std::to_string(onu.index());
          intf.enabled = onu.enabled();

          session.commit();
        }
    }

  }; /* class Pilot */

} } /* namespace adva::wdm_pon */
```

3 Spatio-Temporal awareness and ML-based network management

3.1 Introduction

Given the reduced cell size in future very dense 5G networks, understanding traffic demand variations is considered to be essential for Mobile Network Operators (MNOs) to optimise the overall network performance. As an example, in the envisioned 5G networks, RAN functional splits and/or placement are updated following the spatio-temporal traffic dynamics. The 5G-XHaul cognitive control plane has to be designed flexible enough to be integrated with different 5G RANs.

One of the key elements of the 5G-XHaul control plane is its cognitive capabilities, namely the ability to monitor the current RAN state and, based on previous information/statistics, to forecast short-term spatio-temporal traffic demand variations. In this section we will show how the control/management plane can obtain the information, and examples are given.

Spatio-temporal demand models though require the control plane to be able to predict the movements of the mobile users. In 5G-XHaul several alternative designs are used, each will then require different degrees of integration between the RAN and the transport network.

The main contributions of this section are:

- One method using Machine Learning (ML) techniques to reduce network wide energy consumption (Section 3.2).
- Two methods for mobility prediction (IHP and HWDU), based on MRP and combinations of MRP and precise user tracking info (Sections 3.3 and 3.4).
- Evaluation of methods for precise user tracking with a single anchor combining Sub-6 and mmWave (Section 3.5).

3.2 ML-based load estimation and energy efficiency

Requirements for future 5G networks are not exclusively focused on increasing throughput and reducing latency, energy efficiency represents another key challenge which has been studied within the 5G-XHaul framework.

In this section we explore the idea of applying ML techniques to improve management procedures of future 5G networks towards a fully self-organised network (SON) with special focus on energy savings. As suggested in [4], artificial intelligence (AI) could be useful to that aim. AI techniques, such as ML, are able to process large amounts of a variety of network-generated data (e.g. billing data, radio measurements, packet-level statistics, etc.) to produce exploitable knowledge with applications to network planning and optimisation. Through the analysis of network-generated data available from the 5G-XHaul consortium [1], the proposed intelligent system is capable of anticipating the future state of the network, so that appropriate actions can be taken in a timely manner. As a proof of concept, we apply such intelligence to achieve energy savings: we propose a mechanism to switch off/on cells according to their expected traffic conditions. The operation of the proposed system is depicted in Figure 3-1, and it is discussed in the following subsections.

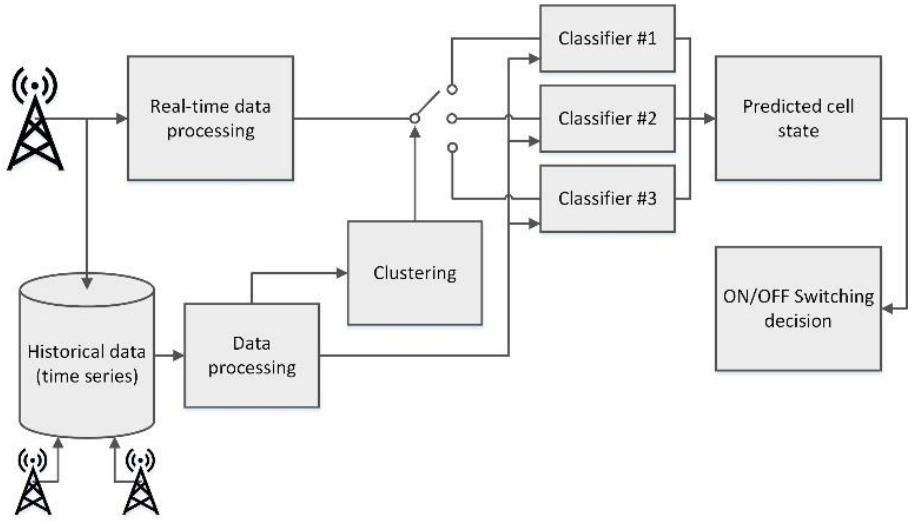


Figure 3-1: Design of a ML-based system for energy savings in mobile networks.

3.2.1 ML techniques: Classification and clustering

Having discarded regression techniques to predict carried throughput or number of users due to the poor accuracy obtained, *classification* puts forward as an interesting alternative. Classification is a supervised learning technique intended to associate an entity, which is defined by the evolution of its features (i.e. relevant metrics), with a given class, selected among a pre-defined set of classes. The classifier uses a training dataset (i.e. a collection of entities and their features along with its proper classification) extracted from historical data to derive a model that will be used to identify classes from new incoming data. In this case, our entities are the BSs present in the RAN and the features considered are those metrics related to cell load, namely, the number of active users, carried traffic and physical radio blocks (PRB) assigned to UEs. For each new sample received from a given BS, the classifier determines its class for the next period. We defined the following three classes:

- **OFF**: BS candidate to be switched off for the next period.
- **ACCEPT**: BS is not a candidate to be switched off and it is expected to carry a medium/low load.
- **FULL**: BS is expected to be highly loaded.

To reduce the complexity of the system, instead of generating a model for each BS in the RAN, which does not scale when MNOs may control tens of thousands of BSs, we use a reduced set of models. In that sense, we use *clustering* in order to identify sets of BSs showing similar behavior so that the same model can be applied to all of them without a significant loss of precision. Clustering is an unsupervised learning technique used to group entities into several clusters considering the similarity of those entities' features. The set of features considered by the clustering block of Figure 3-1 consists of the average number of users, average throughput and average PRB utilisation and their respective variances obtained by processing historical data. Since BSs are likely to be switched off mostly during valley hours, we are interested in using specific models showing high precision for those periods. Hence, those metrics are measured during night hours (0 to 8h).

Therefore, for each of the clusters identified, an independent classification model is generated using for its training process only data from those BSs belonging to that cluster. Accordingly, the state of each BS is evaluated only by the classifier model corresponding to that BS's cluster.

3.2.2 Energy saving mechanism

The last block in Figure 3-1 decides whether a given BS is switched off or kept in normal operation. This block receives periodic updates with information on the predicted state for each BS. Many kinds of algorithms, including other ML techniques, could be applied with the aim of maximising energy savings while, at the same time, minimising the impact over the final users. Note that a reduction in the number of active BSs may entail coverage gaps (i.e. areas not covered by BSs) and, therefore, unserved users, reduced area capacity, possibly affecting QoS, and an increased number of handovers because a cell could be serving UEs when the management system decides to switch it off.

As a proof of concept, we propose a simple approach where the switch on/off decision takes into account the predicted state of a cell and its neighbors: if a cell is classified as *OFF* and a given number of neighboring BSs is in either *FULL* or *ACCEPT* state, then that cell can be switched off. We evaluate two versions of that algorithm:

- **Reckless** algorithm: switches off all BSs classified as *OFF*, regardless of the state of its neighbors. It is an aggressive approach used merely as a benchmark, which will provide larger power savings, but might affect the amount of users having to migrate or even left unserved if no other cell in the area is active.
- **Neighbor-aware** algorithm (**NA-X**): before turning off a candidate cell, the state of neighboring cells in a given area are checked. If the ratio of active cells is higher than a given threshold, X, the system is allowed to turn off that BS with a reduced probability of creating coverage gaps. That assignment can be dynamic to deal with the trade-off decision to balance the amount of user migrations and the power saving goals.

3.2.3 Methodology

With the objective of evaluating different management strategies in cellular networks, we developed a custom-made simulation environment capable of generating realistic scenarios from the traces extracted in a real network. The Java-based application developed for this project generates scenarios of 2,000 x 2,000 m, which are filled with the desired number of multi-sector BSs. BSs are distributed over the scenario in a semi-random way; that is, BSs are first distributed over a regular grid and then deviated randomly from that point and assigned a random coverage radius uniformly distributed between 100 and 600 m. Each cell is also assigned one out of three power profiles depending on its size [6] and assigned to one cluster (following the same proportion found in the real network). Then, each simulated cell is assigned a real cell (of the same cluster) for the rest of the simulation. Once the scenario is set, the simulation loops through the following steps to simulate a whole day:

1. The features of each simulated cell are set based on the assigned real cell's traces; i.e. a given number of UEs is generated under the coverage area of each cell, where the carried traffic and the MCS distribution among those UEs follow that of its assigned real cell.
2. Upon a handover, the amount of PRBs needed by a UE that moves to another cell changes depending on the UE's carried traffic and new MCS distribution. To capture this effect, a penalty parameter ***E*** is computed for each cell according to equation (1), where PRB_t is the total PRB utilisation in the cell (in %), w_i is the weight of each of the N associated UEs, proportional to its MCS [1, 2 or 3] and t_i is the UE's offered traffic.

$$E = \frac{PRB_t}{\sum_i^N w_i \cdot t_i} \quad (1)$$

3. The next cell state (ON, FULL or ACCEPT) is predicted based on the data from the previous samples using the classification model (with a Random Forest approach) corresponding to each cell's cluster.
4. Switch on/off decision on each individual cell is taken according to the selected algorithm (either *Reckless* or *NA-X* version).
5. UEs (and their traffic) generated for a cell that has been switched off are handed over to the closest available cell in ACCEPT state, which is capable of satisfying the required PRBs (multiply UE's throughput by the ***E*** penalty of the new cell). If the UE is not within coverage of an ACCEPT cell, move to the closest FULL cell.
6. Compute relevant statistics: power consumption, number of migrated UEs, number of UEs left unserved, etc.
7. If the end of the day is not reached, load parameters from real traces corresponding to the new time of the day and go to step 1. Else, exit the simulation.

3.2.4 Results

Two strategies for energy saving (i.e. *Reckless* and *NA-X*) are tested in the simulation environment previously described. Both approaches are studied in a possible future 5G deployment, based on the same real

deployment where traces are extracted, but scaled up to represent a future 5G scenario, as envisioned in [8] (HetNet / Outdoor small cell scenario):

Future 5G deployment: multi-tier deployment consisting of a first layer of 100 three-sector macro-cells (radius between 100 m and 300 m) covering the whole area, and a lower layer of small cells (600 BSs with radius between 10 m and 60 m) intended to improve area capacity. Note that macro-cells only consider other macro-cells when applying the NA-X while small cells consider both.

With a density of 3.3 cells (each point of the scenario is served by 3.3 cells on average), the probability of causing coverage gaps is small and, therefore, the total figure of unserved users can be minimised if the algorithms are tuned properly.

In Figure 3-2 we show a comparison of the main metrics obtained with different algorithm configurations, where the macro and small cells layers use different parameters for the NA-X algorithm. The nomenclature *MX_SCY* identifies the threshold parameters ***X*** and ***Y*** for both the macro and small cells, respectively. Note that a NA-0 is equivalent to the reckless approach, while a NA-100 means that no cell is turned off. The figure shows that more restrictive configurations – i.e. higher threshold values – entail less users being migrated and/or left unserved. In the case of *M100_SCY* (i.e. macro-cells never sleep, small cells use a threshold of *Y*), the number of UEs unserved is zero since 100% of the macro layer is always on service, leaving no coverage gaps. However, those configurations perform worse in terms of consumed energy (around 4%, on average, resulting in ~80.0 kWh saved per day). In the other extreme, when both the macro-layer and the small cells run the reckless version of the algorithm (i.e. NA-0 or M00_SC00 in Figure 3-2), energy savings exceed 26% (equivalent to ~500 kWh per day in the simulated scenarios). On the other hand, UEs left without service can exceed 1%. Note that small cells have a low impact on the overall power consumption and that important savings are witnessed only when the macro-layer participates in the energy saving strategy.

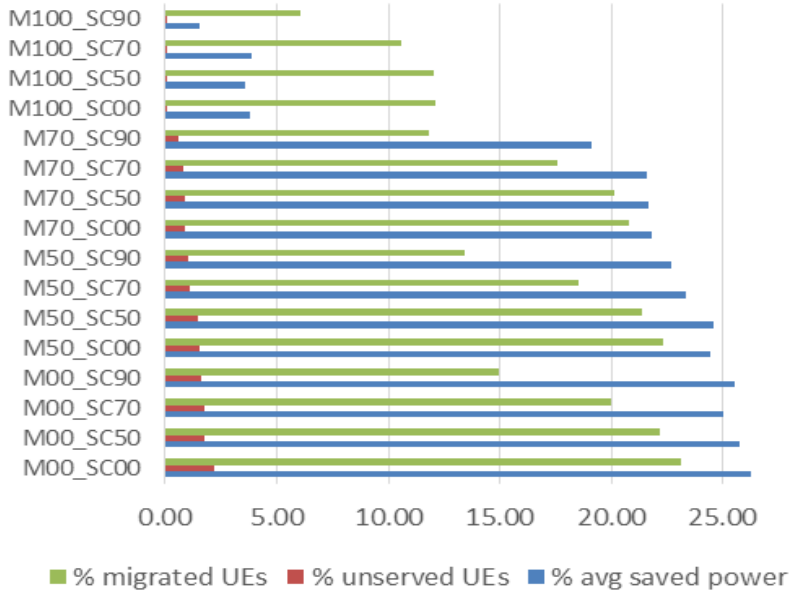


Figure 3-2: Performance figures for different NA-X algorithm configurations in a future, 5G multi-layer deployment.

According to the simulations, energy savings could translate into $1,000-6,000 per year and per km^2 (current prices in Europe). Results also show that switching on/off small cells will not be enough to meet the required reduction in energy consumption. However, switching off at the macro layer may affect the service quality due to coverage gaps, which is an unacceptable outcome for a mobile network operator. Alternatively, macro-cells could switch off specific RATs (e.g. 2G, 3G, etc.) to reduce some energy and still be able to serve users. This approach will require a more sophisticated energy model and is left for a future work.

3.3 Traffic Prediction model for eMBB services

In this study, a mobility and traffic prediction model is introduced, which correlates the probability of users' mobility and the distribution of the associated traffic towards particular ANs. Indeed, we model the behaviour of users in order to provision and optimise the BH and access resources. After studying the traces from COSMOTE in deliverable D3.2, we realised that the cell-level traces could not capture well the user mobility patterns and the per video traffic session analytics.

In this study, for modelling the user mobility, the Self-similar Least-Action Walk (SLAW) model [33] is used, derived by empirical studies of real-life human-walk traces. SLAW exploits the property of "gravity points" or so-called "clusters", i.e., popular areas where users tend to move and accumulate with high probability, providing a realistic outlook of network traffic. The DTMC model is adopted for traffic prediction to derive the stationary probabilities of a mobile user residing at each candidate cluster after a given number of moves, while having an ongoing session. For simplicity we assume that the total area is divided to a grid-type of candidate clusters/gravity points, which are uniformly distributed (in our assumptions a grid of 10x10 candidate clusters was deployed). A user can be either idle or active and can turn to idle after a certain number of cluster changes. With this step we can find the probability of a user following a path, comprising a sequence of cluster "handovers", while having an ongoing session. In the second step, a stationary probability of each state is used (which represents the probability of a user residing in a candidate cluster while having an ongoing session). From this information, in combination with the session duration, we extract the fraction of time that each active user resides in a cluster. This will allow for the efficient resource provisioning, which spans end-to-end (from macro-BS to end user). This can solve the problem of distributing video segments associated with each user to a particular AN provided that a video can be divided into multi segments.

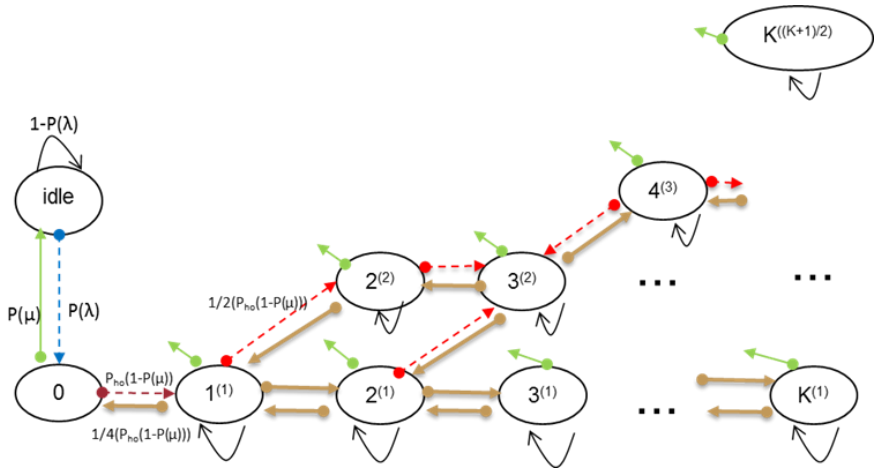


Figure 3-3: State transition diagram.

Figure 3-3 illustrates the State Transition diagram, where we have $S_i^{(j)}$ aggregated states including the state for being idle. *K* is the maximum number of cluster changes. The following probabilities can be defined as: *P(λ)= λt* is the session arrival probability following a Poisson distribution, *P(μ)= μt* is the session departing probability and $P_{ho} = mt$ is the cluster handover probability based on the user mobility rate *m* and potentially the cluster size. When a user is at the idle state, the probability of a session arriving (incoming or outgoing) during a time slot t is *P(λ)*. The current cluster becomes the local point, with a user being able to move to the state $S_0^{(1)}$ with this probability or stay at the idle state with the probability *1 – P(λ)*. When a session is initiated, the probability of a session terminating during a time slot t is *P(μ)*. Based on transition diagram the stationary probabilities can be found by solving the set of equations as discussed in [9].

By solving the Markov chain problem the probability of a user starting a session at a certain cluster and terminating this session at cluster k after j cluster handovers can be calculated. However, this model cannot capture the actual/estimated trace of each user. Each user may start from a distinct point and may have a different direction. The cell handover probability also requires information regarding the clusters involved. It is assumed that both aforementioned points can be derived by the SLAW model in relation with the knowledge of the video duration. In particular, let the handover sequence for user u be: $H_u =< h_1, h_2, \dots, h_m >$ and the number of BSs in this sequence is N_u. Both vectors are assumed to be known using the SLAW model and the probability of having ongoing video sessions while traversing different routes. For example, when a user requests a service

(e.g. DL video session), assuming that the initial position of the user and the duration of the session is known, the route to be followed (how many cluster changes, which can be translated to sequence of HOs) can be derived by the model.

Let π_{idle} and $\pi_{i,j}$ be the stationary probabilities of being at state S_{idle} and $S_i^{(j)}$ respectively as derived in [9]. For a given probability, $p_{m,j} = \sum_{i \in I_m} \pi_{i,j}, \forall j \in K$, and, by taking into account H_u, and user velocity v_u, we can translate the probability of a user moving between clusters to a function corresponding to the fraction of time that each user resides at each cell:

$$\xi_{m,u} = f\left(p_{m,K_u}, H_u, v_u\right) \tag{3-1}$$

Note that K_u is the number of cluster changes from $u(1,1)$ till $u(m, t_u)$ and p_{m,K_u} denotes the probability that user u has an ongoing session at cell m after K_u cluster changes.

Based on this, we can divide the video into equally sized segments. Let the number of segments per user based on the video content be t_u, and T_{max} the maximum number of segments for a video transmission. The number of segments per cell and user can be easily derived by the following equation:

$$N_{seg}(m,u) \triangleq \left\lceil \xi_{m,u} t_u \right\rceil, \tag{3-2}$$

where $N_{seg}(m,u)$ shows how many segments per user u are going to be provisioned or stored at m cell.

3.4 Mobility Prediction models

Maintaining Quality of Service (QoS) is a challenge in mobile communication networks and too is anticipated to be in the next generation of mobile systems (5G). One of the approaches adopted by scientists to meet this challenge is to provide the amount of required resources before arrival of the user to the cell. To this end, knowing the future crossing cell of users appears to be essential. In particular, Mobility Prediction (MP) enables us to predict future crossing cell of users and allocate required resources to the cell in advance thereby reducing the number of failed handovers, alleviating unsuccessful call-attempts in the network and increasing the total throughput of the network.

To predict the future crossing cell of a user, several techniques have been presented in literature. These techniques include History-based prediction [17], Machine Learning algorithms [18][19], trajectory prediction, Kalman filtering [30] and probabilistic models [22][23]. It is to be noted that the users show both random and regular behaviour in their daily life. In particular, the path between home and office can be regarded as an example of regularity in behaviour whereas exploring new areas of the city can be viewed as randomness in users' movements. Therefore it appears technically reasonable to consider both the regularity and randomness in the prediction algorithm in order to enhance its accuracy.

To this end, we use Markov Models [17] to capture the regularity in behaviour of the users and record the instantaneous position of each user within each cell in order to take the randomness into account [24]. An Markov Renewal Process (MRP) is a semi-Markov process wherein the next-state transition probabilities are governed by a Markov process and the sojourn time in any state is dependent on current and next state [1]. Furthermore, we draw on the ideas in [24] to incorporate the impact of randomness into the prediction algorithm. Particularly, we record the user's positions within its current residing cell whereby the transition probability to each neighbouring cell is calculated subsequently. In brief, MRP represents the information about regularity while instantaneous positions contain information about randomness of users' movements.

After collecting information about regularity and randomness of movements, the next step is to combine them in order to reach the right decision about next crossing cell. Among all combining methods, Dempster-Shafer (DS) theory has received huge attraction in recent years. DS theory involves gathering a number of pieces of uncertain information, which are presumed to be independent. Each piece of information is represented by a mass function. Later, all mass functions are combined in order to reach the final decision about future crossing cell [25]. In special cases, as proved in [22], DS theory can be considered equivalent to Bayesian theory of inference.

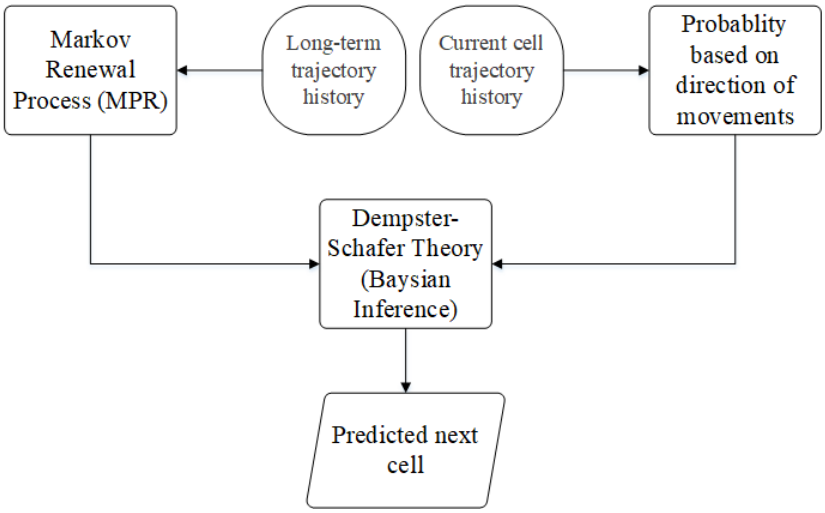


Figure 3-4: Block diagram of the prediction algorithm.

Figure 3-4 shows the scheme of prediction algorithm where probability of each neighbouring cell being next crossing cell is calculated based on two sources of information, namely long-term and short-term data history, and finally is combined through DS theory to predict the next cell.

3.4.1 Dempster-Schafer (DS) Theory

To reach a right and reliable decision about next crossing cell, we are required to utilize all the collected information from two sources of evidence (namely long-term data history and current cell locations). To this end, among all combination theories, DS theory proposes a rule of combination which recently has aroused enormous interest. In following, we briefly give an overview of DS theory and its application to Mobility Prediction [26].

DS theory begins with assuming a Universe of Discourse Θ which is a set of mutually exclusive propositions about a domain. We let 2^{Θ} be the set of all subsets of Θ.

A *mass function* $m: 2^{\theta} \rightarrow [0,1]$ also known as *basic probability assignment* (bpa), is defined with following conditions:

$$m(\emptyset) = 0, \quad \sum_{A_i \subseteq \Theta} m(A_i) = 1. \tag{3-3}$$

It is to be noted that mass function is assigned directly to evidences (subsets of θ) while traditional probability theory assigns numbers to the elements of θ [26]. In our case, where we have evidence only about the individual elements of θ (singletons) *mass function is equivalent to traditional probability theory*.

According to DS theory, suppose m_H and m_L are two mass functions of the same set θ from two distinct and independent sources of evidence, *H* and *L*. The rule of combination which combines bpas is given by [27].

$$m_H \oplus m_L(C) = \frac{\sum_{X \cap Y = C} m_H(X) m_L(Y)}{1 - \sum_{X \cap Y = \emptyset} m_H(X) m_L(Y)}, \forall\ C \neq \emptyset \tag{3-4}$$

Where *X* and *Y* are two subsets of θ i.e. an element of set of subsets, 2^{θ}, and the denominator is a normalisation factor to keep the value of $m_H \oplus m_L(C)$ in [0, 1].

It has been proved in [22] that DS theory is equivalent to Bayesian theory when we assign numbers only to the singletons of set 2^{θ} i.e. the mass functions are Baysian. Consequently the DS rule of combination turns into Baysian rule of inference which is given

$$Pr(C_i|H \wedge L) = \frac{Pr(C_i|H)Pr(C_i|H)}{\sum_{i=1}^{N} Pr(C_i|H)Pr(C_i|H)} \quad (3\text{-}5)$$

Thus the problem of predicting the future crossing cell is equivalent to

$$\arg\max_i \; Pr(C_i|H \wedge L) \quad (3\text{-}6)$$

3.4.2 Markov Model

History-based prediction methods with the aid of Markov models have been addressed in literature [17][18][28]. In this work, we employ MRP from [17] to obtain the probability of each neighbouring cell being chosen as next crossing cell. MRP is a generalisation of renewal process in which the time between renewals are chosen according to a Markov chain [28]. As can be seen in Figure 3-5, each cell is modelled as a state in Markov model and the transition probabilities in Markov model denote the probability that the user transitions to each neighbouring cell. Note that the Markov model for the whole cellular network is too complex to be drawn here. One can easily obtain the entire model by replicating the model in Figure 3-5 for all the cells.

The semi-Markov kernel for a time-homogeneous process is given by [17]

$$Q_{i,j}(t) = Pr\{S_{n+1} = j, T_{n+1} - T_n \leq t | S_n = i\}, \quad (3\text{-}7)$$

where S_n and $S_{(n+1)}$ represent the state of the system after the n-th and (n+1)-th transitions, respectively, with T_n and $T_{(n+1)}$ being the times at which the n-th and (n +1)-th transitions occur, respectively. $Q_{(i,j)}(t)$ denotes the probability that, after transitioning into state i, the process transitions into state j within t units of time. We then rewrite above equation as

$$Q_{i,j}(t) = P_{i,j} G_{i,j}(t), \quad (3\text{-}8)$$

where

$$G_{i,j}(t) = Pr\{T_{n+1} - T_n \leq t | S_{n+1} = j, S_n = i\}. \quad (3\text{-}9)$$

Exponential distribution could typically be assumed to represent $G_{(i,j)}(t)$ [29]. Such distribution is defined with the parameter $\lambda_{(i,j)}$ considered to be the rate of transition from i to j. The probability then is written as

$$Q_{i,j}(t) = P_{i,j}\left[1 - exp(-\lambda_{i,j} t)\right], \quad (3\text{-}10)$$

with $t \in [0, T_{(n+1)} - T_n]$.

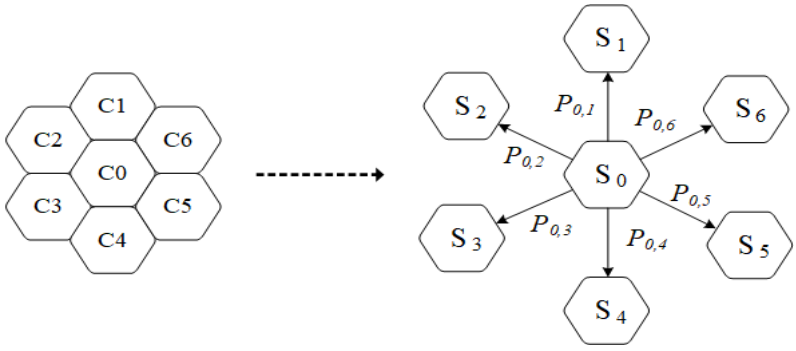


Figure 3-5: Markov Model in a cellular system.

3.4.3 User Positions within the cell: localisation

To take the best possible decision about next crossing cell of users, one cannot only rely on the long-term history of movements. In particular, although users indicate some sort of regularity in their movements e.g. going every day to work, university, etc. there is still a degree of randomness in users' movements due to traffic condition, construction barriers, or exploring new areas. Therefore it appears to be necessary to access other sources of information to be able to incorporate the randomness of movements into prediction algorithm.

A reliable source of information is user's instantaneous location (short-term data history) across the current cell, which can be extracted using the implementations proposed in [2], and in section 3.5 of this document. Specifically, based on the past trajectory of a user in the current cell, we assign probabilities to each neighbouring cell being next crossing cell. We define the current cell location history as vector $L_N = [l_1, l_2, \ldots, l_N]$, whose elements are the locations that user has crossed in the current cell. Clearly, number of the elements N depends on frequency of capturing the user's location. Note that vector L is the past locations of user in currents cell whereas the long-term data history that we utilize to train MRP, is sequences of cells crossed by the user in long period of time e.g. six months or one year. Knowing the vector LN, the following probability is to be calculated $Pr(C_i \mid L_N)$.

Intuitively, we know that each user's movements tend to head for final destination. Therefore monitoring the users' direction of movements within the current cell enables us to perceive their overall direction. As can be seen in Figure 3-6, change of direction can be captured by calculating the variation of user's angle to vertexes of the cell. In particular, with each movement, θ_i will change and as the user moves towards one of the neighbouring cells, the corresponding angle θ_i grows. θ_i can be readily calculated at each point of the cell using cell geometry [30].

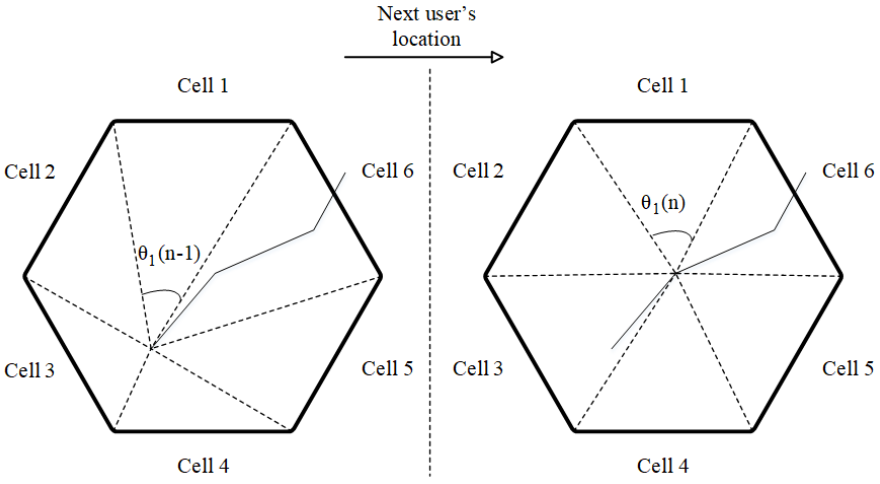


Figure 3-6: User's movement and change of angles to cell's vertexes.

Having known the variation of θ_i, the next step is to assign an instantaneous probability to each neighbouring cell. In particular, with each movement, based on change of angle θ_i we assign a probability to each of the neighbouring cell being chosen as next crossing cell. The assigned probability is as following

$$Pr^{n}_{inst.}(C_i|L_n) = \begin{cases} \frac{\beta(n)}{\sum_{i=1}^{6}[\beta(n)]^{+}} & \beta(n) > 0 \\ 0 & \beta(n) \leq 0 \end{cases} \quad (3\text{-}11)$$

where

$$\beta_i(n) = \theta_i(n) - \theta_i(n-1). \quad (3\text{-}12)$$

To clarify, we assign probabilities according to above equation to the cells towards which the user is moving and assign zero probabilities to the cell from which the user gets away. In other words, we assert that the user approaches a cell (or group of cells) and simultaneously retreats from a number of cells.

We follow the approach introduced in [24] and define a circle wherein the BS records the movements, assigns the instantaneous probabilities, and eventually makes the prediction on the border of the circle (case 1, Figure 3-7). The prediction of next cell in the case 2 where the user enters the cell and leaves it without crossing the aforementioned circle is only based on MRP. In fact, as soon as the user enters a new cell, we assign a temporary prediction based on MRP and as it moves across the cell inside the circle, we record the trajectories and make new prediction at the border of the circle, where the user is likely to leave the circle (and subsequently the cell). The circle radius could be easily determined based on Received Signal Strength (RSS).

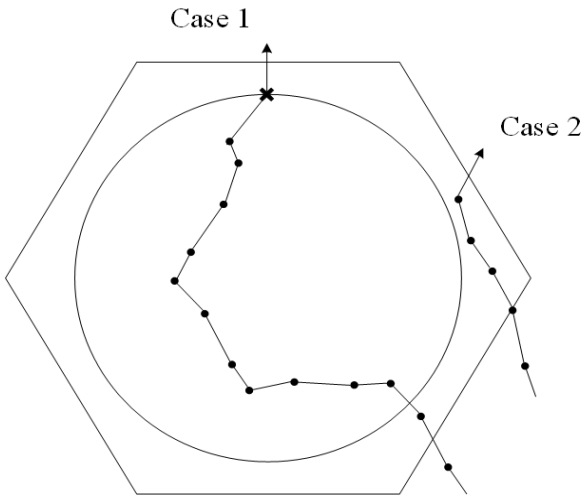


Figure 3-7: Possible movements of a user within a cell.

3.4.4 Exponential Moving Average (EMA)

To determine the probability of each cell being the future crossing cell, we need to consider the movement of the user throughout the current cell. In particular, it is expected that user's overall movement is directed to his intended future cell. Therefore averaging over instantaneous probabilities would be the first idea to take into account such overall behaviour. However, intuitively, user's intended future cell becomes more evident as the user approaches the border of the current cell. In other words, the final direction of movements appears to play a decisive role in next-cell prediction. Given this intuition, instead of using a simple equal weight moving average, we employ Exponential Moving Average (EMA) where the probabilities corresponding to recent movements are given larger weights [31].

3.4.5 Simulation Results and Discussion

To evaluate the performance of algorithm we use the collected data in [32]. The data set includes the trajectory of 4 users, each in different sites. We use only the data from city of New York and the university campus KAIST. Moreover, trajectory of users has been recorded each 30 seconds in XY Cartesian coordinates.

Figure 3-8 indicates the variation of transition probability to neighbouring cells with respect to time. As can be observed, in accordance with CDF of Exponential distribution, $G_{(i,j)}$ (t) converges to 1 as the time grows. In other words, as we expect

$$\lim_{t\to\infty} Q_{i,j}(t) = P_{i,j} \tag{3-13}$$

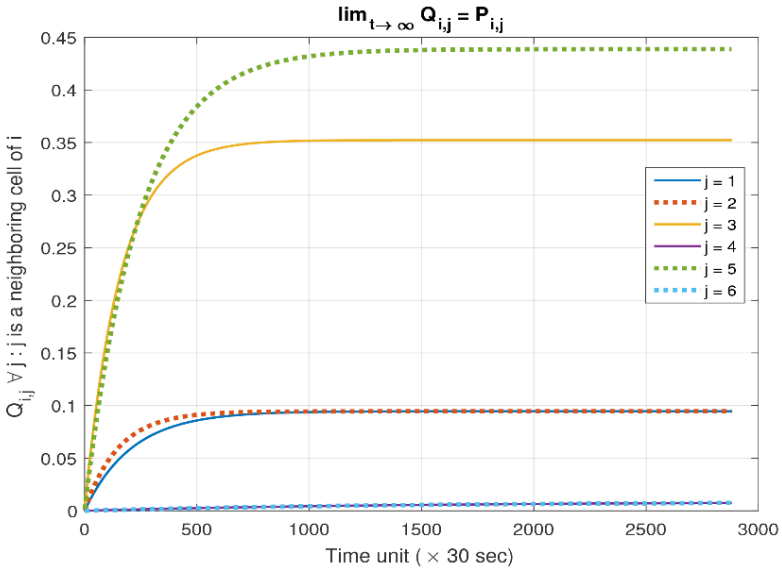


Figure 3-8: Variation of transition probability.

Specifically, in the course of prediction, as the user arrives at the border of virtual circle, $Q_{(i,j)}$ (t) is chosen based on the residing time of the user in current cell up to instance of prediction. Later on it is used by combination rule to make prediction about future crossing cell. In fact, moment of prediction can have significant impact on the outcome of prediction as $Q_{(i,j)}$ (t) varies with respect to time.

Figure 3-9 represents the prediction accuracy versus the ratio of the virtual circle radius to cell radius. The simulation has been performed for two different cities, for each of the two prediction algorithms – one based on the algorithm put forward in this work, and the other based on a Markov Model (MM). As can be seen, the prediction accuracy increases as the virtual circle radius increases. Indeed, as this ratio grows, the number of recorded locations increases and we can incorporate the last movements of the user into the prediction algorithm and, therefore, overcome the random behaviour that might happen in the last moment of user residence in the current cell.

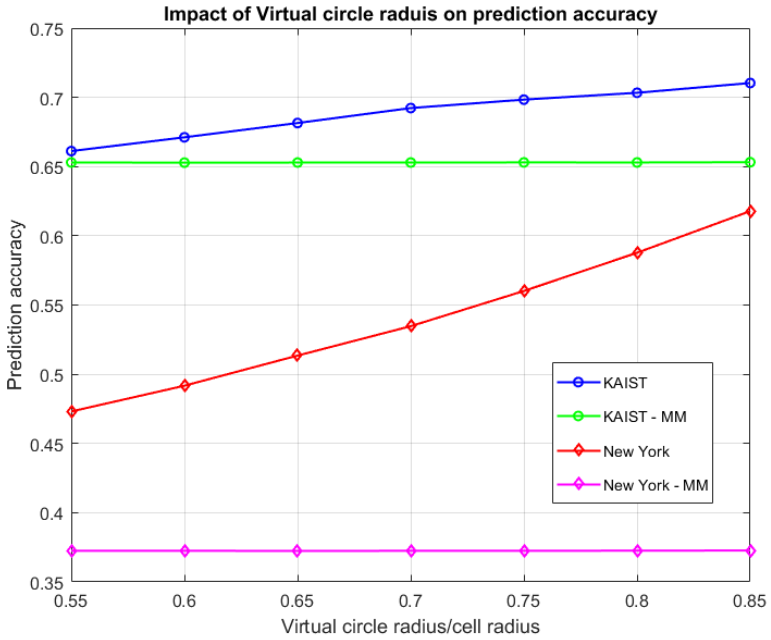


Figure 3-9: Prediction accuracy of the algorithm.

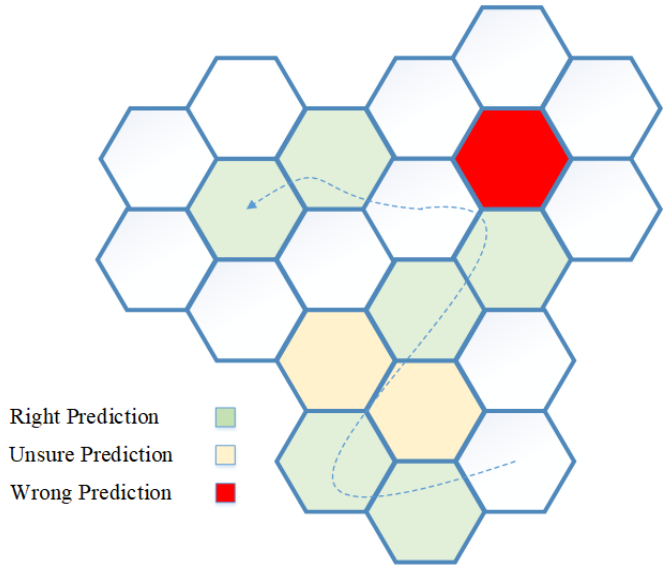


Figure 3-10: Potential scenario where the prediction algorithm can be applied.

To summarise the analysis of our algorithm, Figure 3-10 shows an example of a potential scenario where the prediction algorithm can be applied. The green cells represent the correct predictions, the yellow cells are tentatively predicted cells, and the red cell is a wrong predicted cell. As can be seen, right predictions can be made when there is consistency in the movement of user across the current cell whereas for the yellow cells, namely tentatively predicted cells, it is not clear towards which of the yellow cells the users moves and there is a degree of uncertainty in prediction. The situation grows more difficult where, similar to red cell, the user changes its direction suddenly, leading to a false prediction.

3.5 User localisation – single anchor point

In 5G-XHaul's deliverable D3.2 [1] we provided a deeper view on the capabilities the technologies proposed in the project [34] have to estimate the position of a mobile user. The implementation of trilateration using Sub-6 communications allowed to estimate the position of a user in an indoor scenario using diverse anchor points. This is the case one can face in a city-like scenario (dense urban area), wherever several anchor points are available with the subsequent availability of the signaling required for localisation purposes.

There might be other kind of scenarios in outer areas of the city, or even in the field, where those stations are not installed. Moreover, emergency situations which happen in remote locations, or even indoors – wherever GPS is not available – will benefit from alternative options for localisation of mobile users. In these cases, trilateration is not a feasible option.

For these situations, triangulation – combined angle of arrival (AoA) and time-of-flight (ToF) estimation [34] – represents a good option to provide an accurate location of the mobile user. In 5G networks, the ANs are expected to be equipped with smart antenna solutions, e.g. antenna arrays supporting multiple-input multiple-output (MIMO) techniques or re-configurable antennas [35].

Wireless networks today are not ready to accommodate emergency scenarios, since they have not been optimised in such way. One important technology challenge to support future networks is to cope with the increasing volumes of multimedia information, which might be useful in emergency situations, and need to employ higher-throughput wireless connections. Here is where mmWave-based New Radio (NR) technologies play a role.

The support of features such as localisation and ranging while communicating at Sub-6 and mmWave bands are beneficial not only for future network services (e.g. location-based services), but also for making the most of the mmWave communication itself, in cases like an emergency situation in a non-static scenario. According to [22] [23] [24], multi-connectivity, i.e. a legacy network link and a mmWave link, shall be supported in future wireless networks.

3.5.1 MmWave localisation with Sub-6 assisted AoA

In deliverable D3.2 [1] we reported techniques to localise users as they move in scenarios where several anchor points are available. There are, however, some cases where the UE can only attach to a single anchor node. Assuming the availability of an antenna array at this node, AoA and angle of departure (AoD) can be extracted from the signals exchanged between transmitter and receiver, which can be used for obtaining high accuracy ranging [34].

We address in the following the problem of device localisation in the mmWave band. We propose a solution that leverages the co-existence of Sub-6 GHz and mmWave connectivity at both access and mobile nodes. The proposed solution relies on the AoA estimation using Sub-6 signals at the access node. This information is provided to the mmWave part to facilitate and speed up the subsequent beam training phase, and to favour high-resolution ranging.

3.5.1.1 System model and architecture

We consider a system with two endpoints both having Sub-6 and mmWave radios. At each endpoint the two radio interfaces are synchronised. In our scenario one endpoint is stationary – referred as AN – and the other endpoint is mobile – referred to as User Equipment (UE). The UE has a single Sub-6 antenna and a mmWave antenna array, whilst the AN is equipped with antenna arrays for both frequency bands. Both endpoints have a single RF chain at mmWave (V-Band, 60 GHz), and beamforming is performed in RF domain using a network of phase shifters. A generic system architecture is shown in Figure 3-11. It consists of a mmWave device on top and a Sub-6 device at the bottom.

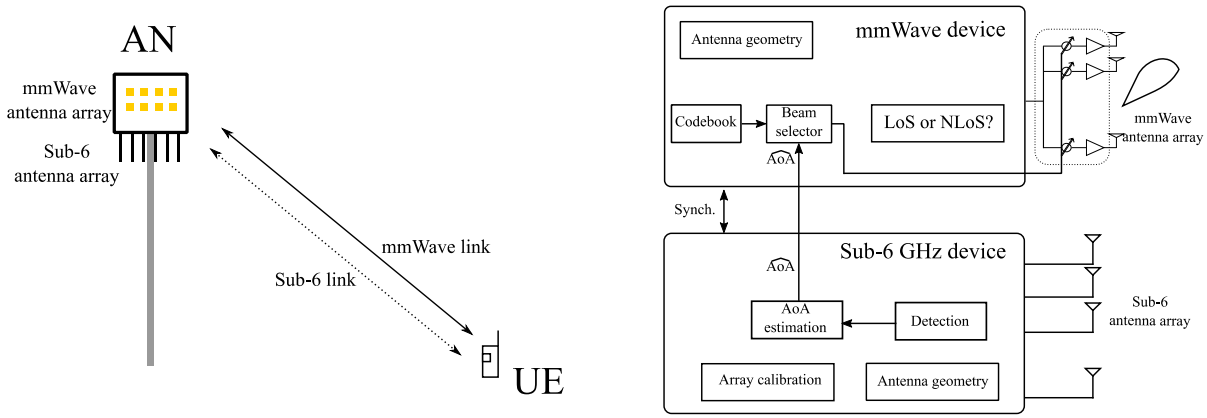


Figure 3-11: (left) System architecture example, (right) AN multi-node architecture.

The mmWave part comprises a uniform linear array (ULA) of Na antennas and a beamforming unit with beam steering capabilities with a certain angular range. Its Sub-6 counterpart has an antenna array of P omni-directional antennas and it is able to detect signal direction of any paired UE. The UE has an mmWave device with an ULA of Ma antennas and a Sub-6 device with a single antenna element. The UE transmits periodically beacon frames in the Sub-6 band to the AN. The AN acquires this signal with the antenna array and processes it to obtain an estimation of the AoA. The AN replies to the UE with a reply frame in the same band signalling the upcoming beam training phase. In addition, this signal can carry additional information, e.g. beam training request from the AN side. The AN uses the estimated AoA using the Sub-6 transmission to steer the beam toward the UE in the mmWave band accordingly. The UE trains its beams and chooses the one with the highest signal-to-noise ratio (SNR). Once the beams are aligned, AN and UE perform distance estimation using Two-way ranging (TWR). Finally, having both AoA and distance estimation between AN and UE, the AN can determine the position of the UE with respect to its position.

Next, we give a details on the signal processing tasks to be performed for aforementioned mmWave localisation. The signal processing engine is depicted in Figure 3-12. It consists of the following tasks: Sub-6 GHz AoA estimation, Sub-6 GHz assisted mmWave beam training, mmWave ranging, line-of-sight (LoS)/non-line-of-sight (NLoS) identification and localisation (i.e. position estimation).

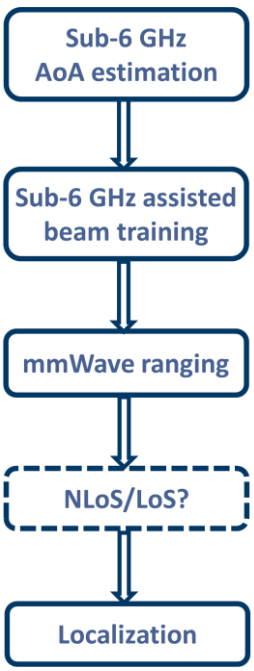


Figure 3-12: Signal processing tasks for mmWave localisation with Sub-6 assisted AoA estimation.

3.5.1.2 Sub-6 GHz AoA estimation

The AN is equipped with an antenna array of P omni-antennas in the Sub-6 band and is able to estimate the angle of arrival (AoA) of the incoming signal from the UE. For this, different estimation algorithms can be used. These are classified into two groups, classical and subspace methods. Classical methods rely on beamforming; i.e. to form a beam in specific direction, scan the angular space and measure the received power. Then, the direction with the highest received power is selected. Subspace methods are super resolution methods exploiting the orthogonality between signal and noise subspace. Most-commonly used methods are MUSIC [36], root-MUSIC [37] and ESPRIT [38]. In this work, we focus is on MUSIC-based methods.

3.5.1.3 Sub-6 GHz assisted mmWave beam training

To combat the path-loss at mmWave frequencies, phased antenna arrays are used to generate directional beams and to provide the necessary beamforming gains. Before any data communication takes place, AN and UE are involved in a closed-loop procedure called beam training. It is an iterative process in which two endpoints perform exhaustive pairing of a number of beams from their codebooks. Codebook beam training procedures are implemented in standards like IEEE 802.11ad and IEEE 802.15.3c.

The use of out-of-band measurements for speeding-up beam training at mmWave has already been tackled in the literature (see [39][40]). In [39] authors have used AoA estimation of a 2.4 GHz to predict LoS mmWave link in an indoor environment. In [40] prediction of the mmWave links in both LoS and NLoS scenarios using AoA estimation of 2.4 GHz signal is performed.

Here, we assume that each endpoint (AN or UE) has a codebook consisting of N (M) beams or sectors, $N \geq N_a$ ($M \geq M_a$), to cover the angular space of interest. The estimated Sub-6 AoA is used to narrow down the angular search space. As a result, the number of AN mmWave beams used for beam training can be significantly reduced. The AN starts the beam training phase using the beam closest to the estimated AoA. In addition, it can request a beam training from the UE in order to refine its beam. This information can be provided within an acknowledge frame sent to the UE after AoA estimation. An example of AN and UE mmWave beams (sectors) and Sub-6 AoA is shown in Figure 3-13.

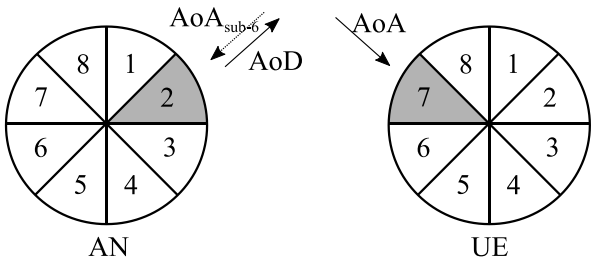


Figure 3-13: Illustration of mmWave beams (sectors) at AN and UE side. Estimated Sub-6 AoA in sector 2, AN angle of departure (AoD) in sector 2 too, and UE AoA in sector 7.

We assume that two nodes are synchronised during Sub-6 communication. Once the Sub-6 AoA is available, the AN triggers the mmWave communication by sending a training frame to the UE towards the direction of the estimated AoA (see Figure 3-13 left). The training frame is sent M times in M successive time slots of duration Ts. At the same time, the UE is listening using different beams in each time slot to find its AoA. Based on the received SNR or other representative metric, the UE finds its best beam (Figure 3-13 right). If there is no optional beam training request from the AN side, the UE sends a reply frame to finalize the beam training session. If the reply frame is received at the AN, the beam training phase is finalised and the ranging phase can start.

However, if the AN has requested beam training (i.e. refinement of its beam), after a reply frame is received, the beam refinement phase at the AN side starts. The AN trains its beam in successive $S \cdot Ts$ time slots ($S < N$), while the UE sends to the AN the training frame using the previously trained beam. After $S \cdot Ts$ time slots, the beam training phase ends. In the best case, when there is no beam refinement request from the AN, the beam training procedure lasts for $M+1$ time slots, $(M+1) \cdot Ts$. With the AN beam refinement included, it lasts for $M+S+1$ time slots, $(M+S+1) \cdot Ts$. Usually S is chosen to be two, which corresponds to beams right and left from the beam used in the first phase.

The beam training procedure is shown in Figure 3-14. Assuming $N = M = 16$, in the best case scenario Sub-6 GHz assisted beam training would last for 17 time slots compared to exhaustive beam search which needs $N \cdot M+1 = 257$ time slots. This way beam training latency can be reduced up to 93.4 %.

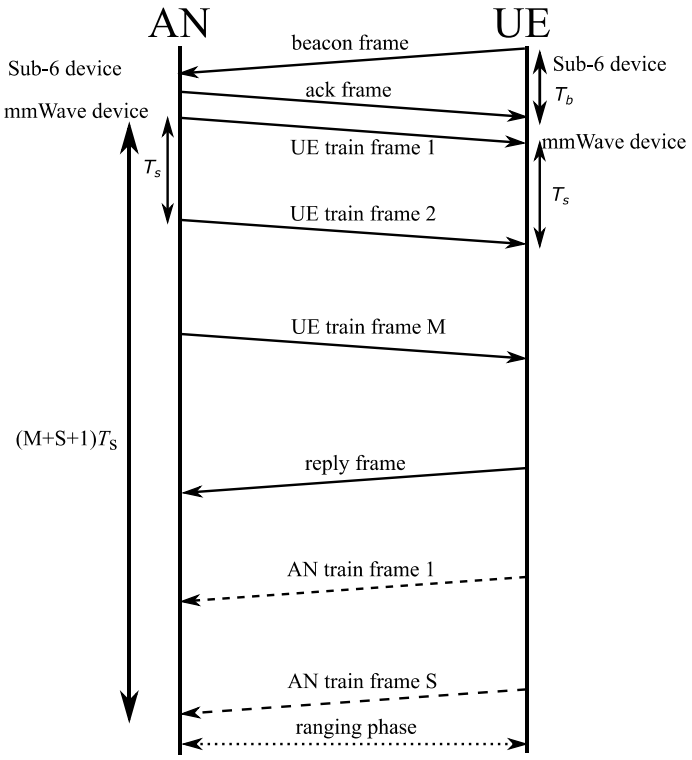


Figure 3-14: Beam training time diagram (Ts > 2(Tprop + Tproc)), Tprop – propagation time, Tproc – frame processing time.

3.5.1.4 mmWave ranging

In D4.13 [34] different ranging methods were introduced. Here, we favour time-of-flight (ToF) ranging methods, namely the ToA method. Both nodes must be synchronised before the ToA method is performed. Node N1 transmits a waveform which is received by node N2. Node N2 knows the exact time of transmission of the waveform and estimates the ToA of the received waveform. Having both time of transmission and the ToA, the ToF can be easily estimated. This method requires precise synchronisation, which cannot be easily achieved.

LoS/NLoS detection

ToF-based ranging methods require LoS propagation scenario to achieve adequate ranging accuracy. In a NLoS environment there is no direct path between a transmitter and a receiver. Hence, radio signal travels longer, i.e. over NLoS path, so the traveling time will be positively biased. This would lead to false distance estimate and, consequently, to localisation error. Therefore, it is important to identify if the signal is propagating over LoS or NLoS path. There are different methods to perform LoS/NLoS identification (e.g. [41]). Mostly these rely on extracting different features from channel impulse response (CIR) estimates (e.g. mean, variance, skewness, kurtosis, Rician K-factor, etc.) and then applying some advanced signal processing techniques (e.g. machine learning). In this work, this part is not evaluated, but it is identified as important one and it will be pursued in future work.

Device position estimation

The final step is the UE position estimation. Having the AoA and distance estimates, the AN can estimate the position of the UE as shown in Figure 3-15. Assuming AN is located at the origin O(0,0), the position of UE is obtained as:

$$x_{UE} = R\cos(\Omega),$$
$$y_{UE} = R\sin(\Omega),$$

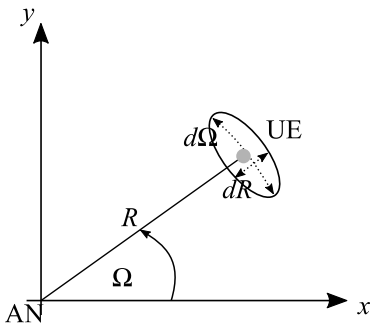


Figure 3-15: UE position estimation (analytically and graphically).

where R represents distance estimate and Ω is the AoA estimate.

3.5.1.5 Scenario description, system setup and measurements

For Sub-6 AoA estimation experiments, software defined radio (SDR) devices USRPs X300/X310 from Ettus are used. The SDR UE is equipped with UBX160 daughterboard and has single dipole antenna, whilst the SDR acting as the AN has TwinRXs daughterboards connected to 4-dipole antenna array.

We use omnidirectional dual band antennas (2.45 and 5.8 GHz) having 2.5 and 4.6 dBi gain, respectively. For mmWave tests, 60 GHz Analogue Front-Ends (AFEs) with beamforming capability are used [42]. To generate the baseband signal at the AN side, an arbitrary waveform generator (AWG) is used. At the UE side, the baseband signal from the AFE is sampled using a digital sampling oscilloscope. The acquired samples are stored in memory and processed offline.

The measurements were performed in an anechoic chamber and office environment. In D4.13 we have performed a number of ranging measurements at 60 GHz in an anechoic chamber. Here we show them once again for the sake of completeness. The estimated distance as a function of the true distance is shown in Figure 3-16. It can be noticed that the estimated distance accurately matches the true distance. The root mean squared error (RMSE) is below 1 cm. The CDF of the distance estimates around the mean value reveals that the achievable precision falls below 2 cm for each of the distance measures.

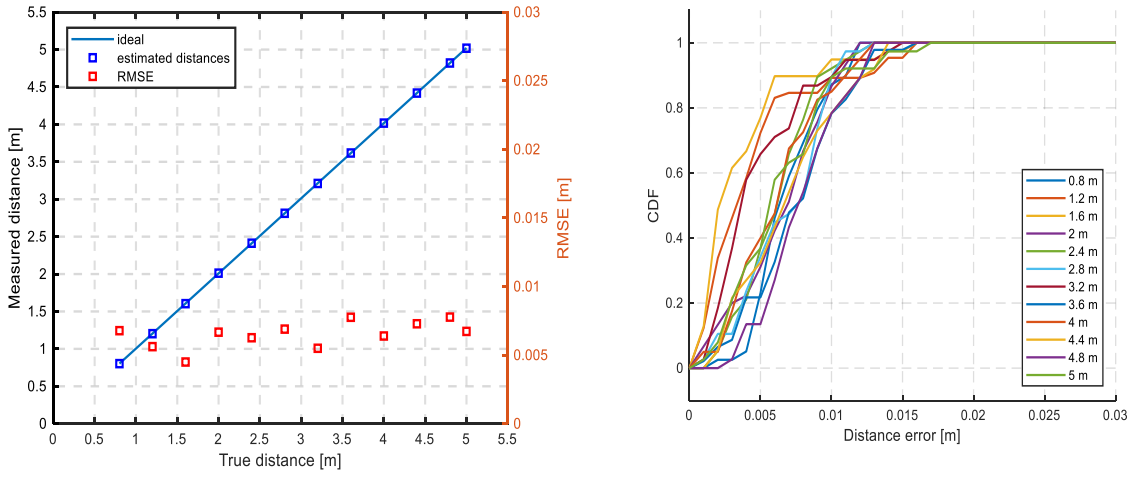


Figure 3-16: Estimated distance vs true distance with RMSE (left); CDF of the distance estimates (right).

Furthermore, we performed AoA measurements in the anechoic chamber as shown in Figure 3-17. The distance between the transmitter (UE) and the receiver (AN) is five meters. The system is calibrated according to [43]. The pilot tone is sent at the centre frequency of 2.45 GHz with a sampling rate of 1 MSps. The test is performed for different angles. At the AN side, the signal is recorded and processed with MUSIC and root-MUSIC methods. Results are plotted in Figure 3-18, whereas the estimated angles are given in Table 3-1.

Values that are obtained based on 1000 measurements show that the estimated AoAs fall within 2 degrees from the true AoA and that the standard deviation is below 0.2 degrees.

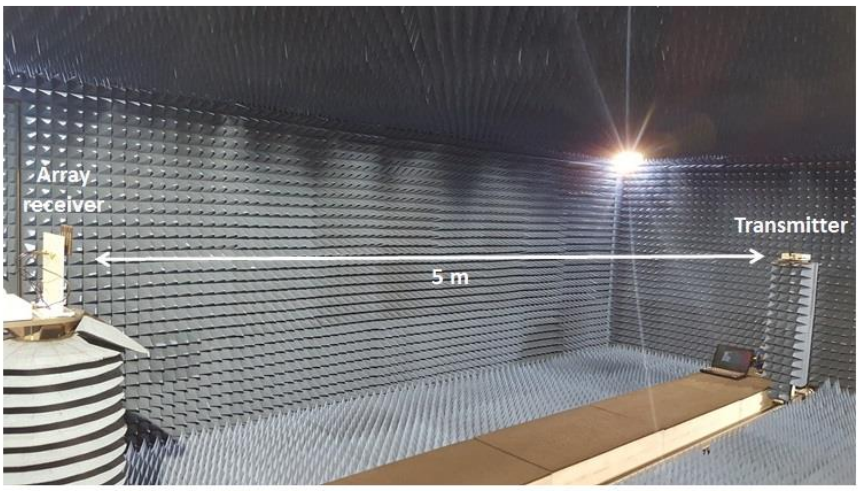


Figure 3-17: AoA measurement in the antenna chamber

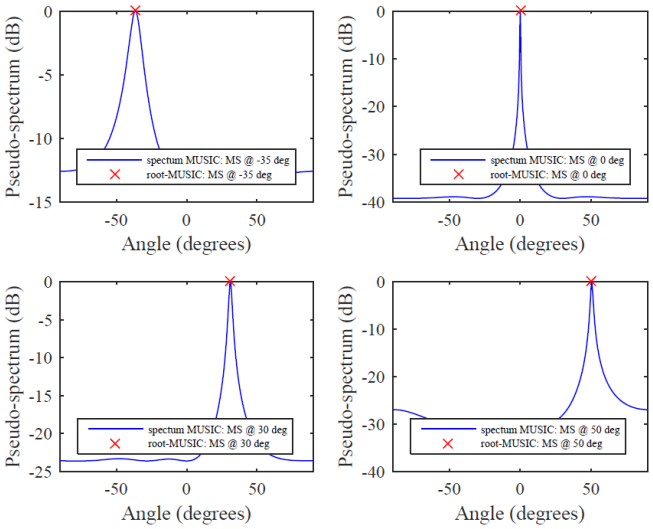


Figure 3-18: AoA estimation of the pilot tone impinging the antenna array at the angles -35, 0, 30 and 50 degrees.

Table 3-1: True and estimated AoA.

AoA true (°)	-35	0	30	50
AoA est. (°)	-36.4	0.02	31.1	50.6

In addition to the measurements in the antenna chamber, an additional set of measurements were performed in an office environment shown in Figure 3-19. The AN is placed on a wooden cabinet at the height of 175 cm, while the UE was positioned at five different locations at the height of 100 cm such that Line-of-Sight (LoS) path with the AN exists. The office floor plan with the measurement locations is illustrated in the same figure. In this case the measurements were performed at the frequency of 5.8 GHz in order to avoid interference from the WiFi APs used for Internet access operating in the 2.4 GHz frequency band.

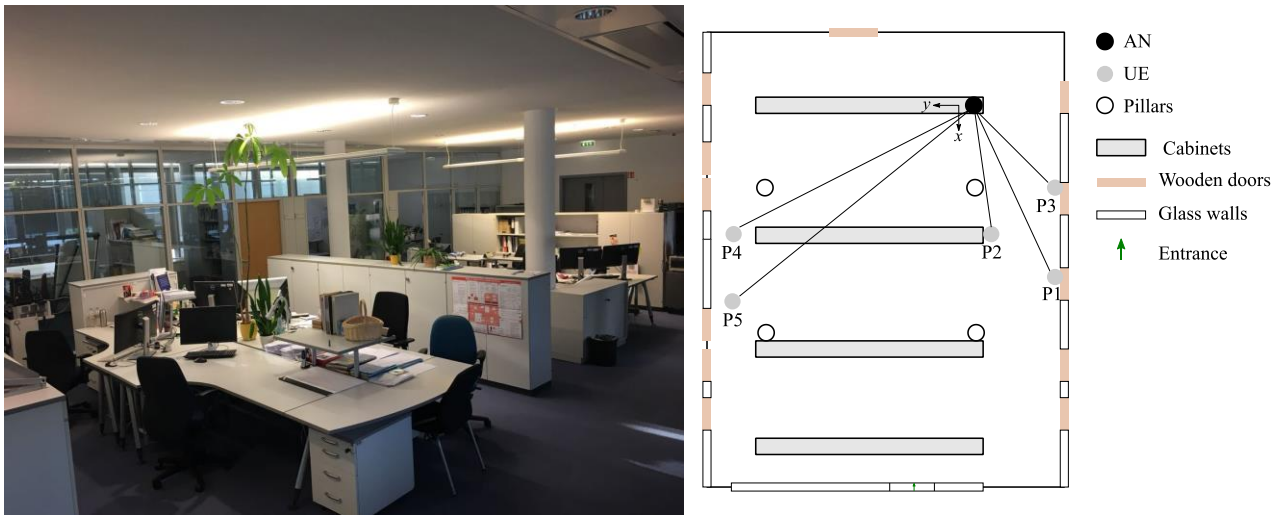


Figure 3-19: Office environment where the measurements were taken and the floor plan.

All measurements were performed in a static scenario. For each location (P1-P5), AoA and ranging distance were measured. The mean estimated AoA and the mean range estimate, R, are reported in Table 3-2. Ranging results with RMSE are shown also in Figure 3-20.

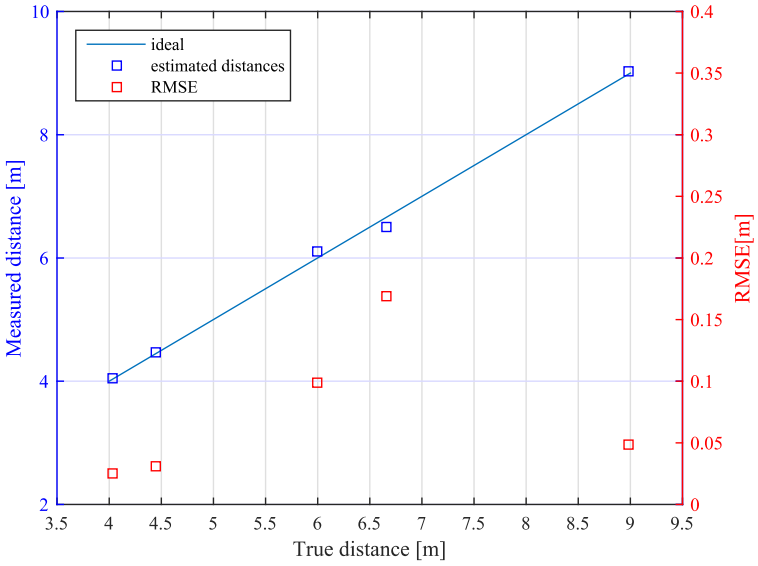


Figure 3-20: Estimated distance vs true distance and root mean squared error.

Table 3-2: AoA estimation and ranging accuracy.

Location	AoA (°)		*R* (m)		Abs. error	
	True	Est.	True	Est.	AoA (°)	*R* (cm)
P1	-26	-29.7	6.66	6.5	3.7	16
P2	-10	-15	4.45	4.47	5	2
P3	-38	-43.2	4.03	4.04	5.2	1
P4	44	44.3	6.00	6.1	0.3	10
P5	30	25	8.98	9.02	5	4

The estimated distance is very close to the true distance. Minimal RMSE is 2.5 cm and maximal is 17 cm. For some measuring locations, especially those close to wall or glass wall, there is a bit larger discrepancy between estimated and true distance. This can be explained by the fact that signal was received over the reflected path, as high side lobes in the antenna pattern of the used AFEs were observed.

Regarding the AoA, the maximal absolute error is around 5 degrees. The reason for the higher errors compared to the antenna chamber might be due to the harsh propagation environment (office environment). In addition, more precise calibration of the antenna array across angles and versus temperature is needed (i.e. periodic calibration is needed).

3.5.2 UE localisation using combined angle of arrival and ranging at Sub-6

In deliverable D3.2, a localisation approach using time of arrival was reported [2]. Using this approach, UEs receiving signals from 4 to 5 ANs can perform two/three dimensional localisation. This is a good approach in dense user scenarios, where many ANs are available and visible to the UEs. The ANs must be synchronised, which is the major challenge in this approach. The approach shown in D3.2 does not require that the UEs transmit any frames in order to obtain their position. This is a huge advantage in dense user scenarios since the usage of the wireless medium for localisation is minimal. Nevertheless, achieving visibility of such number of ANs is not always possible. Moreover, having a large number of ANs can be expensive in most of the cases. One such case is for example a rural area where the ANs are dispersed and only a few are visible to the UEs. In this case it is hard to perform precise localisation of the UEs. Therefore, in scenarios with limited number of ANs it is always preferable to use the combined AoA/ranging approach for localisation.

We have tested the AoA/ranging approach in the Sub-6 GHz band in order to estimate the achievable performances. This approach is explained in Subsection 3.5 and shown in Figure 3-15. The estimation of AoA in the Sub-6 GHz band, implemented and tested here, is explained in Subsection 3.5.1.2. In the approach implemented and tested, a two way ranging (TWR) is used. The details of this approach are given in D4.13 in subsection 4.2.2 [34]. We perform here a TWR with scheduled windows.

This approach was tested in an office scenario. One AN and one UE were used. The AN has an ULA of four omni antennas with a gain of 2.5 dBi each, used for AoA estimation and a separate patch antenna with a gain of 7 dBi for performing TWR. The UE has two patch antennas with 7 dBi gain each. One is used for transmitting a waveform used for AoA estimation by the AN, and the other antenna is used for TWR. The main reason for splitting these functions of AoA and ranging is the capability of the used equipment. In this approach we use a software defined radios which can receive on four antennas, but cannot transmit. Therefore, we use a separate radio for TWR, which can transmit and receive on a same antenna.

For the AoA, the carrier frequency is 5.8 GHz and the bandwidth is less than 20 kHz. A single sine wave is transmitted from the transmitter and is used for AoA estimation. For the TWR a carrier of 5.75 GHz is used. The 3 dB bandwidth is 25 MHz and the sampling rate per I and Q signals is 50 MSps. A squared root rised-cosine (SRRC) filter is used as a pulse shaping and matched filter. For performing TWR, two successive receive windows are scheduled, one for each node, as explained in deliverable D4.13 in subsection 4.2.2. During these receive windows the nodes transmit a maximum length sequence, 1023 chips long and BPSK modulated. This is basically the ranging sequence, needed for time of arrival estimation. It has strong cross-correlation peak, which is used for time of arrival estimation.

The combined AoA/TWR is tested in an office environment. One AN is placed at a height of 1.8 metres and the antennas from the UE are placed on the same height on a moving cart. Few different positions are chosen and the angle and the distance are estimated. The floorplan depicting the positions of the AN and the different positions of the UE are shown in Figure 3-21. Positions at which the UE was placed during the measurements are marked from 1 to 6. On each of these positions, the angle and the distance were estimated and position was estimated. A few hundred estimates were made for each of these positions in order to obtain the mean value of the measurements, which corresponds to the accuracy. The dispersion of the measurements around the mean value corresponds to the precision.

In Figure 3-22 the true and the estimated positions are shown. The green circle is the AN, which is placed at coordinates (0, 0). The blue circles are the true positions of the UE. They are marked with numbers from 1 to 6. The red dots are position estimates. As can be noticed, for positions which are closer to the AN, the error is smaller, whereas for the positions that are further away the error is larger. The reason for this is that for the positions that are further away the received power is lower and for the positions that are near to the AN, the received power is larger and therefore the precision is higher.

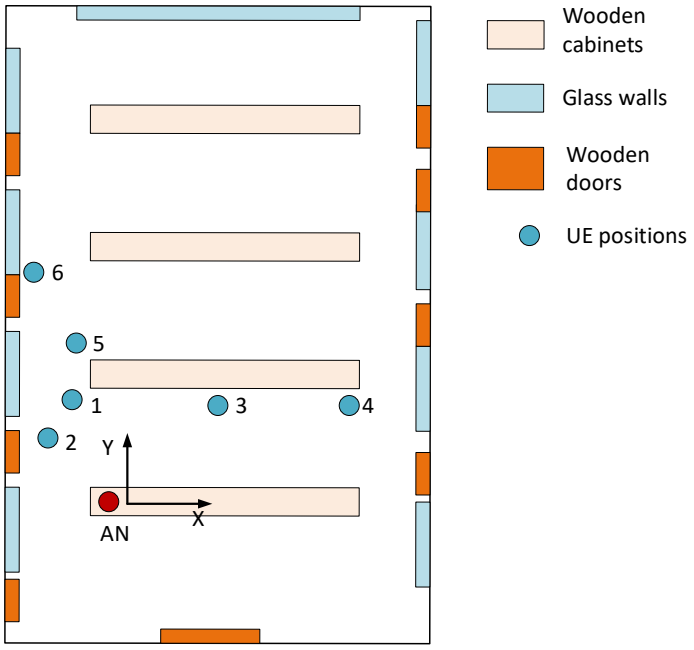


Figure 3-21: Floorplan and positions of AN and UE.

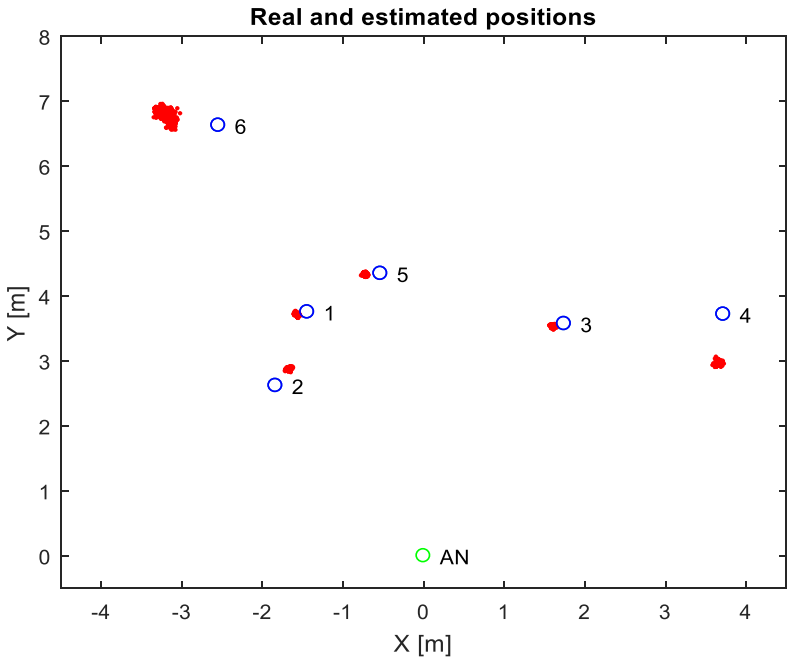


Figure 3-22: True and estimated position.

Estimated angles RMSE is also larger for positions 4 and 6, again due to the limited received power.

The high estimated distance and angle errors would lead to high position estimation errors for positions 4 and 6. This can be also noticed from the results reported in Table 3-3. Nevertheless, the positioning error remains in values less than 1 metre.

Table 3-3: Position of the UE and the RMSE in distance, angle and position estimation.

UE pos.	True position (x; y) [m]	RMS distance error [m]	RMS angle error [º]	Estimated position error [m]
1	(-1.4410; 3.7539)	0.0277	1.9769	0.1418
2	(-1.8354; 2.6213)	0.1315	4.8214	0.3045
3	(1.7434; 3.5745)	0.0932	1.5867	0.1433
4	(3.7194; 3.7194)	0.5505	5.9499	0.7549
5	(-0.5338; 4.3474)	0.0225	2.5396	0.1958
6	(-2.5444; 6.6284)	0.4172	4.1658	0.6752

4 5G-XHaul SLA Definition

The service management processes are responsible to guarantee the fulfilment of the accepted Service Level Agreements (SLAs). An SLA defines the different parameters (i.e. Key Performance Indicators, KPIs) that a specific service must comply. These SLAs are subject to a contract with a tenant of the network. Hence, the non-compliance of an SLA supposes a penalty to the service provider normally in monetary terms. The consumer specifies the Service Level Objectives (SLOs) and for its part, the provider informs about their service characteristics and quality guarantees, it is known as Service Level Agreement Template (SLAT).

It is important within the SLA management environment that both provider and management domains can implement their policies and operational objectives for their own resources (i.e. taking control over the network access, transported information and the managed nodes). Therefore, coordination between both domains is necessary for the treatment of monitoring alarms.

In the framework of 5G-XHaul, it is necessary a mechanism to allow the negotiation of SLAs between the stakeholders involved in the services, namely the 5G-XHaul infrastructure provider, owning the 5G-XHaul transport network, and the tenant, which could be a Mobile Network Operator (MNO), interested in leasing BH and/or FH connectivity services, and potentially also cloud services to host virtual BBUs. The interested reader is referred to deliverable D2.1 [64] for a description of the 5G-XHaul user roles. So, an option for the requirement statement may be that the definition of the SLAs is performed through a portal with an intuitive Graphical User Interface (GUI). It must be a tool with the needed application forms, distinguishing each final service, where customers can select the value for the different parameters to be accepted for the service (e.g., number of CPUs required, the bandwidth of the transport network or the maximum latency allowed in the transmission). The value assigned to these parameters constitutes the SLOs that will be part of the final SLA. The implementation of the aforementioned platform is beyond scope of the project and will be an objective of the future work (conceptual idea showed in Figure 4-1).

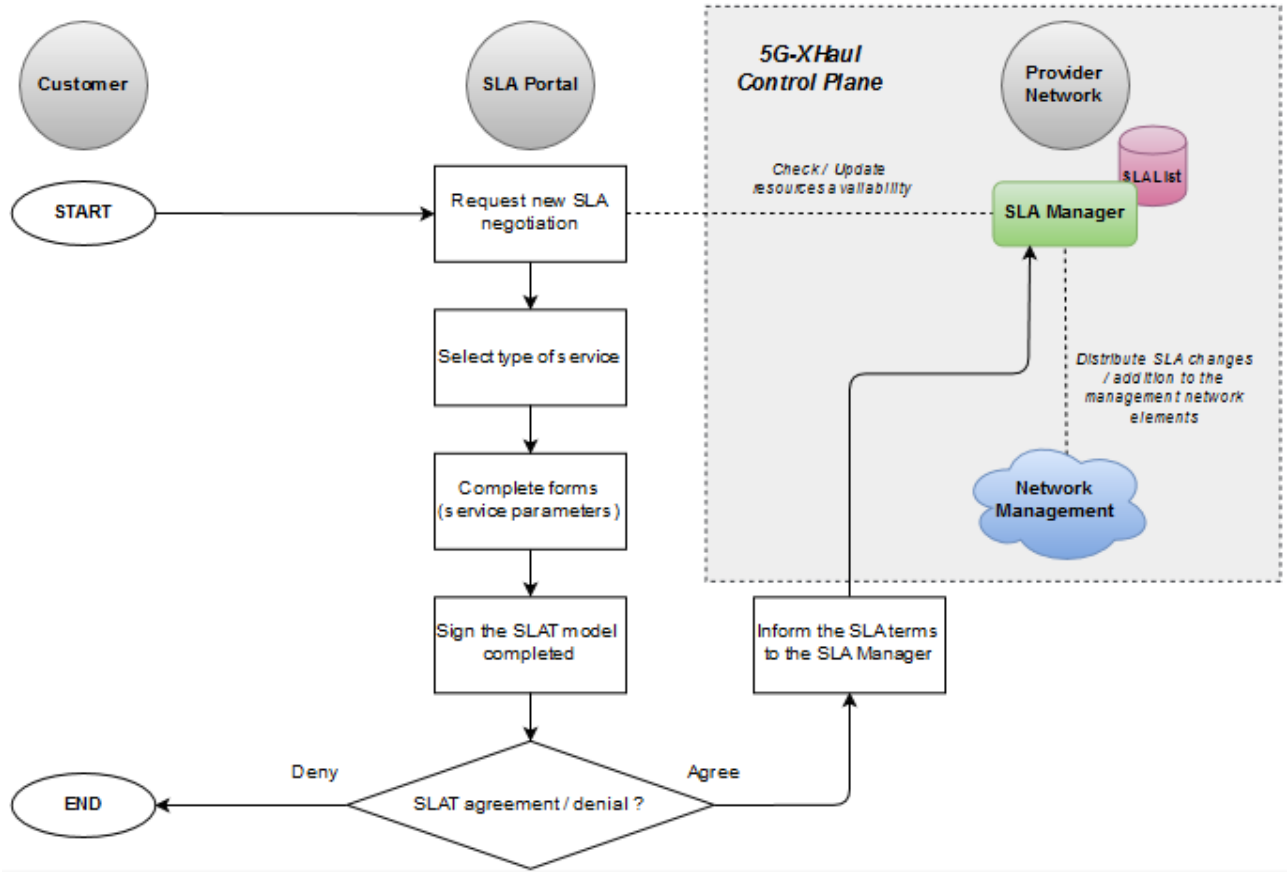


Figure 4-1: New 5G-XHaul SLA negotiation.

The network orchestrator integrating this tool must include functionalities to update the available resources in a proactive way: knowing the current available resources in the network elements through an alarm/notification system; or in a reactive way: using a request/response methodology once a new SLA petition arrives to the platform. In the former case, the tenant may see the available resources and may select among them. In the latter case, the system must check the petition, send an internal request and inform to the customer if the provisioning request could be accepted or not. An example connectivity service that could be exposed by a 5G-XHaul network is the end to end connection request described in Section 2.1, which would be invoked by the network orchestrator.

The SLA portal must be accessible for the tenants to issue requests within the allowed features. Once the service request is accepted (or denied) the customer may check the cost of these services and the impact (i.e. penalties for the provider) of the violation of this SLA. This final agreement results in the SLAT, where the characteristics of the offered service, as well as the possible penalties to be applied in case of non-compliance, are indicated. Table 4-1 includes the description of the fields required in the SLAT for the definition of an SLA.

Table 4-1: 5G-XHaul SLAT model.

SLAT Field	Description	Value
Provider_Name	Name of the service provider / operator	(e.g. SerPro)
Provider_Class	Type of service provider	(e.g. ISP)
Provider_Description	Extended definition of the service provider including personal in contact, legal information, etc.	--
Consumer_Name	Name of the service consumer / costumer	(e.g. SerCon)
Consumer_Class	Type of service consumer	(e.g. Golden Class)
Consumer_Description	Extended definition of the service consumer including personal in contact, legal information, etc.	--
Service_Class	Level of the service (to different classes of services, QoS)	(e.g. Level 1)
Service_Cost	Unit of cost related with the Service_Class	(Quantity in corresponding unit)
Service_Description	Extended definition of the rendered service	(e.g. This service ensures the …)
Service_Type	Functionality of the service required	(e.g. Connectivity service)
Service_Operations	Description of the technical operation involved in the service accomplishment	(e.g. Monitoring transport node / event management / re-routing path)
Service_Objectives_Description	The complete list of the SLO	(e.g. delay, bandwidth, security)

Service_Objectives_Accepted	The maximum/minimum level of guarantee the SLA's compliance	(e.g. 100 µs)
Service_Objectives_Alarm	The maximum/minim level to alert the system of possible SLA's non-compliance	(e.g. 85 µs)
Service_Recovery	The description of the typical actions to recovery after a service alarm	(e.g. reallocation of resources, rerouting the network path, etc.)
SLA_Penalties	Description of the possible penalties suffered by the provider in case of SLA's non-compliance.	(e.g. 5% discount of the service cost)

This SLAT information must be translated to a pseudo-language (metadata: parameter/value) to be communicated to the control and management elements. This metadata must be interpreted by the orchestrator to discriminate the KPIs for the different services. This description of the service's capabilities and objectives must be completed through some kind of well-known standard API for web services (e.g. REST / SOAP) depending on the need.

According to the 5G-XHaul control plane (CP) architecture, properly described in deliverable D3.2 [1], it is necessary to determine the entities in charge of the orchestration and the correctly accomplishment of the SLAs. Figure 4-2 presents again the schematic view of the elements taking part in the network control functions with the pro-posed elements for the treatment and monitoring of the SLAs.

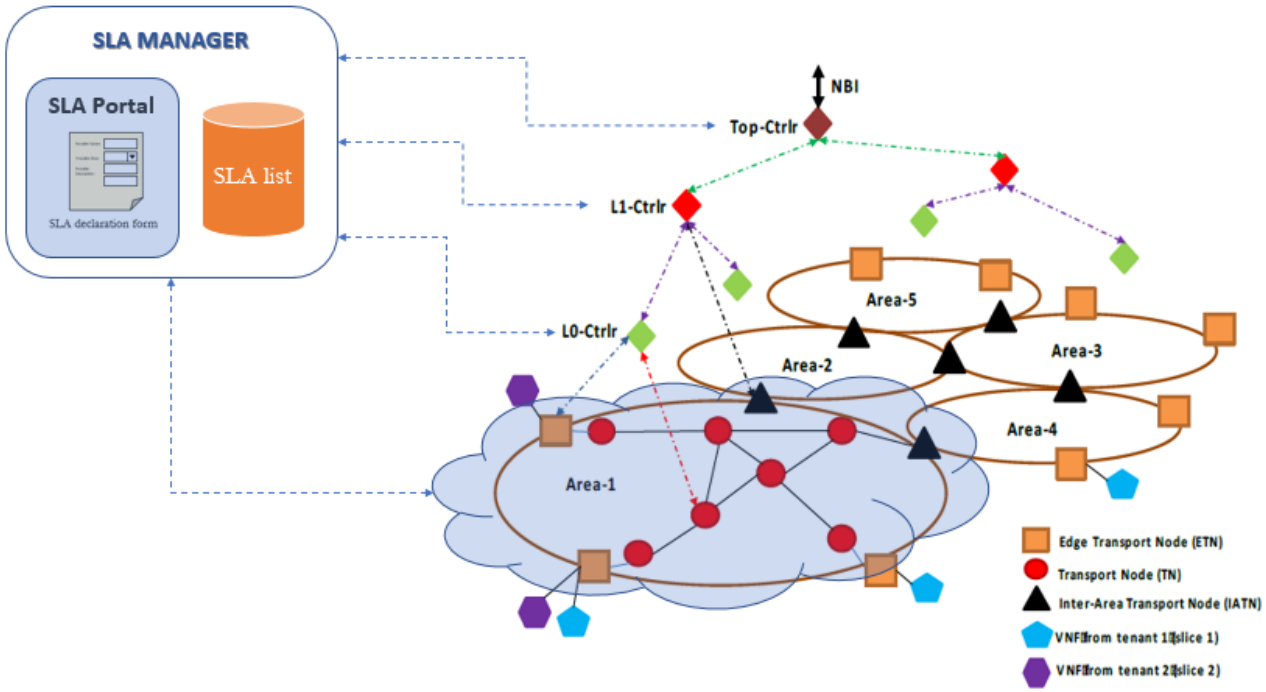


Figure 4-2: 5G-XHaul Control Plane Architecture with the proposed SLA orchestration.

As Figure 4-2 shows, there are four different controllers, whereby define the scope of each one of them is vital to achieve a correct performance. The CP incorporates the followings controllers:

- Local Agents (LA) embedded in each transport node (i.e. TN, ETN and IATN).

- Area Controllers (AC) that rules the control plane functionalities of a single local area. Identified as Level-0 controllers (L0-Ctrlr).
- Level-1 controllers (L1-Ctrlr) which are in charge of maintaining connectivity between several ACs.
- Top controllers (Top-Ctrlr) are responsible for provisioning per-tenant slices and orchestrating connectivity across different 5G-XHaul areas.

Focusing in the Local Agents, the main functions to carry out are related to the treatment of network events (e.g. alerts notifying a network node down or the congestion of a link in the transport network), the dynamic allocation of resources based in the different QoS or network slice, mapping and re-routing incoming traffic by VNF to the tunnels provided by the AC. The LA informs to the AC through an alarm mechanism in case there is no more available capacity for the configured incoming flows. The various SDN south-bound interfaces developed in 5G-XHaul for the mmWave and Sub6 wireless technologies, WDM-PON, and TSON, can be used for this purpose. The interested reader can find a detailed description of these interfaces in deliverables D3.1 [1] and D3.2 [2].

The Area Controllers, via local agents, are informed about the network topology (including link capacity), the estimated traffic demand and the policy directives accorded with the network operator to set up tunnels and to allocate the available resources in their local area. The AC can adapt to the network dynamics and update the transport tunnels in order to satisfy the tenant requirements. It also informs the traffic demand predicted to the rest of controllers involved in the orchestration.

To implement the orchestration of multiple network domains, 5G-XHaul uses the Control Orchestration Protocol (COP) [7] as described in detailed in section 2.1.

On the one hand, the Level-1 network controller coordinates the Level-0 controllers in their area. On the other hand, Top controllers take the application plane requests and forward these to the L1-cntrlr (translated to high-level commands).

Regarding the monitoring tasks in the control plane, it is necessary to incorporate a series of elements of management with a main orchestrator, e.g. NFV MANO [65], to check conformance of the parameters agreed in the SLAs. It is thus necessary to add a coordinator or manager of SLAs. This SLA coordinator must communicate with the main orchestrator, as well as with the various controllers mentioned above.

The local agents may do some of this monitoring functions and inform the upper layers about events related to network performance. An alternative option is the utilisation of active probes installed in each node in the network, i.e. TN, ETN and IATN, to measure the parameters (KPIs) under analysis and anticipate possible issues that could be caused in the network.

These management elements must send alarms to the main orchestrator when a specific KPI is close to violate the SLA agreed level. In this case (e.g. the latency agreed in the SLA is 100 µs and the current latency measured in the transport nodes are 90 µs), the orchestrator is informed and it triggers a new path provisioning for the compromised transport service. In the worst case (i.e. an SLA breach), the management elements inform the orchestrator and, in addition to a path re-provisioning as in the previous case, the orchestrator forwards a message informing about the SLA violation to the tenant with the implied penalties.

The KPIs declared in the SLAs must be known by each of the network management and control modules, which, taking into account the scope of the different controllers, interact between them using the previously mentioned control orchestration protocol. COP has two different control protocol interfaces (CPI), one for the abstract resources (Application-CPI / A-CPI) and one for the request/consumption of the resources (Da-ta-CPI / D-CPI). The interested reader is referred to Section 2.1.

Next, we review in sections 4.1.1 and 4.1.2 typical SLA definitions related to connectivity and cloud services, and propose in section 4.1.3 a template for the definition of 5G-XHaul SLAs combining both connectivity and cloud services.

4.1.1 Connectivity Services SLAs

Referring to transport connectivity services today, network operators need to attend to the petitions from different kinds of stakeholders including corporate users, Mobile Virtual Network Operators (MVNO) and/or wholesale customers.

In the agreements with an MVNO, one of the most significant parameters is the bandwidth to be reserved and to dedicate to the tenant. Main issues with SLAs are those related to the availability of the connectivity service and the throughput of the infrastructure (i.e. bandwidth, latency, number of packet loss, etc.). For 5G future verticals, it is expected that latency plays also an important role, especially for uRLLC and some eMBB time sensitive services. The interested reader is referred to deliverable D2.1 [64] for analysis of the 5G vertical requirements on a 5G-XHaul operator.

Regarding wholesale customers interests, focusing for example on the financial entities, latency becomes a key parameter due the time restrictions. Integrity and reliability of the information exchanged through the transport network are also critical.

Table 4-2 presents the typical connectivity KPIs declared in the SLA definition, which must be monitored by the control plane.

Table 4-2: General Connectivity Services SLA Parameters' Definition.

General SLA Parameters	Service	Definition	Indicative values
Bandwidth	All Services	The maximum data rate available for the service	Depends on service (Mbps)
Latency	All Services	Maximum time allowed on a one-way direction communication.	Depends on service (µs)
Jitter	Connectivity Services (residential and dedicated)	Level of timing deviation between the data sent / received among two endpoints.	Depends on service (%)
Packet Loss	All Services	Max. number of packet loss between the nodes in the IP network	Depends on service (%)
Availability	All Services	Time that the service is not available per (specific) time window. Unavailability due to Planned Maintenance or linked to service offering (e.g. in case offer is valid for 24x5) is not taken into consideration.	Depends on service (min/month)
Reliability	All Services	Frequency of non-availability of the service per month	Depends on service (min/month)
Planned Maintenance	All Services	Time (over a specific time window) that the service is not available due to planned maintenance activities.	Depends on service (min/month)
Monthly Availability Percentage (%)	All Services	(Time that the service is Available per month – Time that the service is Unavailable per month)/Time that the service is Available per month	Depends on service (min/month) (%)

The connectivity services in the network operator's agreements define typically at least three differentiated classes: silver class (SC), golden class (GC) and multimedia class (MC). Table 4-3 contains the parameterisation of these previous points to the different connectivity services analysed. These services classes will be reviewed in section 3.3.3 to fulfil the 5G-XHaul requirements.

Table 4-3: SLA Parameters' Definition for different Connectivity Services.

Connectivity Services	Typical SLA Parameters	Comments/ Sub-parameters	Indicative SLA guarantees
MVNO/5G verticals Infra-structure agree-ments	**Type of service**	Determine the kind of service offered (see section 4.1.3.1)	Fronthaul or Backhaul
	Functional split	Determine the applicable functional split	Type A / B / C (see [66])
	Bandwidth	Guaranteed Data Transfer speed in the network	Depends on agree-ment (Mb/s – Gb/s)
	Latency	Guaranteed maximum time delay in the transmission	Depends on agree-ment (µS)
	Availability	Guaranteed minimum Unavailability time less than 60 min per month	99,90%
	Reliability	Number of service outages	1 time/month
Financial Services	**Bandwidth**	Guaranteed Data Transfer speed in the network	Depends on agree-ment (Mb/s – Gb/s)
	Latency	Guaranteed maximum time delay in the transmission	Depends on agreement (µs)
	Availability	Guaranteed minimum Unavailability time less than 60 min per month	99,90%
	Reliability	Number of service outages	1 time/month
B2B Convergent Infrastructure based services	**Packet Loss**	Guaranteed maximum number of packet loss in the network	SC ≤ 0.9 % GC ≤ 0.8 % MC ≤ 0.7 %
	Latency	Guaranteed maximum time delay in the transmission	SC ≤ 45 ms GC ≤ 35 ms MC ≤ 25 ms
	Jitter	Guaranteed maximum time deviation e2e	MC ≤ 2 ms
	Availability	Guaranteed minimum Unavailability time less than 60 min per month	99,90%
	Reliability		1 time/month
B2B Dedicated Infrastructure based services	**Packet Loss**	Guaranteed maximum number of packet loss in the network	SC ≤ 0.7 % GC ≤ 0.6 % MC ≤ 0.5 %
	Latency	Guaranteed maximum time delay in the transmission	SC ≤ 35 ms GC ≤ 25 ms MC ≤ 15 ms
	Jitter	Guaranteed maximum time deviation e2e	MC ≤ 2 ms
	Availability	Guaranteed minimum Unavailability time less than 60 min per month	99,95%
	Reliability	Number of service outages	1 time/month

4.1.2 Cloud Services SLAs

In the current landscape of cloud services, cloud providers offer a wide range of services based on their customers' needs, thus ranging from:

- Cloud Infrastructure Services (IaaS - Bare Metal Services) offerings: that is; provisioning of plain hardware to customers.
- Cloud Services offerings: that is; provisioning of virtual, cloud (compute/storage) resources to customers.
- Platform as a Service offerings: that is; provisioning of IT services (hardware infrastructure and platform software) usually required for a company's operational purposes.

At the same time, besides the well-established public cloud providers (such as Amazon WS, Microsoft Azure, etc.), telecom operators are moving to the direction of deploying and extending their DCs towards offering cloud services to their customers. The SLAs underpinning these offerings depend on the type of service and market positioning of the telecom operator/cloud provider, the customer needs and contracts, the monitoring capabilities/features of the cloud infrastructure, and various other factors. For private clouds and corporate users, the cloud services SLAs metrics that are usually supported are presented in Table 4-4. The performance evaluation based on each metric is defined per case, i.e. by each cloud provider, as there is no standard cloud performance evaluation framework.

An indicative way in which the aforementioned metrics are translated for different cloud services is presented Table 4-5.

Table 4-4: General Cloud SLA Parameters' Definition.

General SLA Parameters	Service	Definition	Indicative values
Performance		Calculated on a per service basis.	Depends on parameter
Availability	All Services	Time that the service is not available per (specific) time window. Unavailability due to Planned Maintenance or linked to service offering (e.g. in case offer is valid for 24x5) is not taken into consideration.	60 min/month
Reliability	All Services	Frequency of non-availability of the service per month	3 times/month
Planned Maintenance	All Services	Time (over a specific time window) that the service is not available due to planned maintenance activities.	60 min/month
Monthly Availability Percentage (%)	All Services	(Time that the service is Available per month – Time that the service is Unavailable per month)/Time that the service is Available per month	95%

Table 4-5: SLA Parameters' Definition for different Cloud Services.

Cloud Services	Typical SLA Parameters	Comments / Sub-parameters	Indicative SLA guarantees
Storage Space Provisioning Service	**Availability**	Guaranteed minimum Unavailability time less than 60 min per month	99,90%
	Reliability		1 time/month
	Performance	Guaranteed Data Transfer speed between the Storage Space and the VMs	between: 35Mbit - 2Gbit
		Data Transfer link between the Storage Space and the VMs	between: 1Gbit - 10Gbit
VMs Provisioning Service	**Availability**	Guaranteed minimum Unavailability time less than 60min per month	99,90%
	Reliability		1 time/month
	Performance	CPU	4 VCPU's
		Memory – RAM	12 GB
		Storage Space – Hard Disk Space	70 GB
		Additional features offered but not necessarily in SLAs	
		Traffic	2 TB / server
		Input Output Operations per second (IOPS)	800
Databases Provisioning Service (meaning Storage space + DB platform e.g. SQL server)	**Availability**	Guaranteed minimum Unavailability time less than 60 min per month	99,90%
	Reliability		1 time/month
	Performance	Guaranteed Data Transfer speed between the Database VMs and the Storage Space	between: 35 Mb/s – 2 Gb/s
		Data Transfer link between the Database VMs and the Storage Space	between: 1 Gb/s – 10 Gb/s

It shall be noted that, depending on the cloud offerings and the customer profile, monitoring services (e.g. if Memory, CPU, Hard Disk Usage, etc.) can be offered as Managed Services.

In the context of 5G-XHaul, DCs would be accessible by multiple tenants for hosting of various network services in the form of cloud services provisioning. Irrespectively of the service to be hosted, the aforementioned SLAs/SLAs metrics can form a basis for 5G-XHaul services SLAs. In this respect, monitoring of the aforementioned parameters and delivery of values to other network levels/systems (e.g. service level orchestrators/ monitoring systems) through the northbound API shall be possible.

4.1.3 5G-XHaul Enabled SLAs

Exploiting the 5G-XHaul capabilities and leveraging on existing connectivity and cloud services SLAs provided by network operators, new SLAs/SLA templates for 5G-XHaul need to be generated. In these new SLAs, the following capabilities should be reflected:

- The versatility of provided services; being one or a combination of the following:
  - Backhaul network connectivity services, of various types.
  - Fronthaul network connectivity services, of various types (various Functional Splits).
  - Access network connectivity services.
  - Auxiliary services such as synchronisation.
  - Cloud services, e.g. for Cloud-RAN deployments (hosting of vBBUs), for hosting core network elements.
- The dynamic, spatio-temporal provisioning of resources to tenants/users.
- The support of dynamically (time) changing QoS for each service.
- The capability of scaling resources based on specific triggering events (e.g. performance measurements, monitored events, external requests).
- The evaluation of services provisioning on the basis of a number of KPIs such as:
  - Availability.
  - Reliability.
  - Performance: in terms of guaranteed and acceptable values of a wide range of monitored parameters such as:
    - Bit Rates.
    - Latency between two end points.
    - Jitter.
    - Packet Loss.
    - Cloud Resources, such as storage space, vCPUs, RAM, etc.

To this end we can identify the following three major 5G-XHaul SLAs:

- TYPE A SLA: Transport Network Services
- TYPE B SLA: Transport Network and Cloud Services
- TYPE C SLA: End-to-End Telecommunication Services and Cloud Services

Next, we describe and illustrate with example each of the previous SLA categories.

4.1.3.1 TYPE A SLA: Transport Network Services

The users of this type of services could be (Mobile) Network Operators (MNOs) or other tenants aiming at providing telecommunication services to their own customers. Many market players can be considered as tenants, including the commonly identified verticals (in the 5G ecosystem), such as: major infrastructure owners (stadium/airport/shopping malls, etc.), transportation industry, etc. These MNOs/tenants are considered to own an access and core network deployment while lacking a transport network infrastructure. For instance:

- A MNO/tenant owning a Small Cell infrastructure and a remote core network may ask for backhaul services, of specific data rates, latency, jitter, packet loss, etc. depending on the services to be offered.
- A MNO/tenant owning a RRH infrastructure to be connected to a pool of remote BBUs (virtual or physical), and to a core network, may ask for fronthaul services for the RRH-BBUs links and for backhaul/backbone services from that point to the core network end.

The various parameters and the values of the SLAs will correspond to the specificities of the MNOs'/tenants' infrastructure and services per case. The information to be included in a SLA template of this type is presented in Table 4-6, and a SLA example is provided in Table 4-7:

Table 4-6: TYPE A SLA: Transport Network Services SLA Parameters.

Service	SLA Parameters	Indicative Values
Backhaul Service OR Fronthaul Service	**Time Period**	Start date – End date e.g. 1/1/2020 - 30/4/2020
	Periodicity	Continuous; Every second weekend; 8:00am – 17:00pm; Etc.
	Location of the Access Network Nodes (ANNs)	At the NN Stadium area and surroundings; At a specific area/municipality; At a specific building block(s)/campus; At a specific hotspot area (shopping mall, park, etc.); Along a main/national road; Along the train route/lines; Nationwide; Etc.
	Service Details & QoS	**Functional Split**: Type A, B, C (see [66]) **Data Rate**: 1 Gb/s; 2 Gb/s every second weekend and 500 Mb/s the rest of the time; Etc. **Latency**: e.g. 2 ms between the two ends of the transport link **Jitter**: e.g. < 20 ms **Packet Loss**: e.g. 10^{-6}
	Availability	5 min/month
	Reliability	< 1 time/month
	Monthly Availability (%)	99.9% (of requested time)
	Scaling Rules	These may vary depending on the service and agreed run-time policies, e.g. scale up/down depending on utilisation, upon reaching 90% of resources' utilisation

Table 4-7: Indicative Type A SLA.

Indicative Type A SLA				
Time Period	**Periodicity**	**Location**	**Service Details & QoS**	**Other**
1/1/2020 – 30/4/2020	Every 2nd weekend	At the NN Stadium area & surroundings	**BH, FS-B** **Data Rate:** 2 Gb/s **Latency:** 2 ms e2e of transport link	**Availability:** 5 min/month **Reliability:** < 1 time/month
1/1/2020 – 1/1/2021	8:00 am-17:00 pm	Building blocks/ campus; Hotspot area (shopping mall, park, etc.)	**BH, FS-B** **Data Rate:** 1 Gb/s **Latency:** 2 ms e2e of transport link	

1/1/2020 – 1/1/2021	Continuous	At a specific area/municipality; Nationwide; Along a main/national road	**FH, FS-A** **Data Rate:** 500 Mb/s **Latency:** 200 μs ([66]) e2e of transport link	**Monthly Availability:** 99.9%
1/1/2020 – 1/1/2021	5:00 am-24:00 pm	Along the train route/lines	**FH, CPRI** **Data Rate:** 500 Mb/s **Latency:** 200 μs ([57]) e2e of transport link	

4.1.3.2 TYPE B SLA: Transport Network and Cloud Services

The customers of this type of services would be MNOs or other tenants -as in the previous case- owning an access network deployment while lacking a transport network and cloud infrastructure, and aiming at providing telecommunication services to their own customers. For instance:

- A MNO/tenant owning a Small Cell infrastructure may ask for backhaul services, of specific data rates, latency, jitter, packet etc., depending on the services to be offered, and for cloud services to host the virtual instances of its core network and/or services/applications.
- A MNO/tenant owning a RRH infrastructure to be connected to a pool of remote vBBUs, and to a core network, may ask for fronthaul services for the RRH-BBUs links and for backhaul/backbone services towards the core network end, as well as for cloud services to host the vBBUs and/or the virtual instances of their core network elements and possibly services/applications. In case of an application comprising more than one distributed elements, the latter case corresponds to defining service function chaining with SLAs.

The various parameters and the values of the SLAs will correspond to the specificities of the MNOs'/tenants' infrastructure and services per case. The information to be included in a SLA template of this type is presented in Table 3-8, and a SLA example in Table 3-9:

Table 4-8: TYPE B SLA: Transport Network and Cloud Services SLA Parameters.

Service	SLA Parameters	Indicative Values
Backhaul /Fronthaul Service	As in Type A SLA for Transport Network Services	
Cloud Service For hosting virtual instances of Telecom Network Components (e.g. vBBUs, virtual instances of Core Network Elements)	**Time Period**	Start date – End date
	Periodicity	Continuous; Every 2nd weekend; etc.
	Location	**Corresponds to Location of Access Network Nodes (ANNs) where the end-users are expected to reside** At the NN Stadium area and surroundings; At a specific area/municipality; At a specific hotspot area (shopping mall, park, etc.) Nationwide; Etc.
	Cloud Service Resources	**CPU**: **e.g. 4 VCPU's** **Memory - RAM**: e.g. 12 GB **Storage Space - HD Space**: e.g. 70 GB
	Availability	5 min/month
	Reliability	<1 time/month
	Monthly Availability	99.9% (of requested time)

	Scaling Rules	Scale up, upon reaching 90% of resources utilisation
Cloud Service for hosting Applications	**Time Period**	Start date – End date
	Periodicity	Continuous; Every 2nd weekend; etc.
	Location	**Corresponds to the end-users' location** At the NN Stadium area and surroundings; At a specific area/municipality; At a specific hotspot area (shopping mall, park, etc.); Nationwide; Etc.
	Cloud Service Resources -or in case of Service Function Chaining: Cloud Service Components resources	**Service/ Service Component (1..n)** **CPU (1…n)**: e.g. 4 VCPU's **Memory – RAM (1…n)**: e.g. 12 GB **Storage Space – HD Space**: e.g. 70 GB **Links/Link Performance with other components**: Link 1-2: Latency=200 ms, BW: 10 Mb/s Link 1-k: Latency=200 ms, BW: 5 Mb/s …
	Availability	30 min/month
	Reliability	<1 time/month
	Monthly Availability	95% (of requested time)
	Scaling Rules	Scale up, upon reaching 90% of resources utilisation

Table 4-9: TYPE B SLA: Transport Network and Cloud Services Indicative SLA.

Indicative Type B SLA						
Time Period	**Service**	**Periodicity**	**Location**	**Resources (QoS & Scaling Rules)**	**Associated with**	**Other**
1/1/20 - 30/4/20	**BH, FS-B**	Every weekend	**Link Location Info** **End point A:** Location of ANNs: At the NN Stadium area and surroundings; **End point B:** Location of Cloud Instances of Core network elements	**Data Rate:** 2 Gb/s To be scaled with 200 Mb/s increment based on utilisation. **Lat.:** 2 ms, e2e of transport link	Cloud 2	**Availability:** 5 min/month **Reliability:** < 1time/month **Monthly Availability:** 99.9%
1/1/20 - 1/1/21	**FH1, FS-A**	Every day 8:00 am-17:00 pm	**Link Location Info** **End point A:** Location of RRHs: specific building blocks/campus; hotspot (mall, park, etc.) **End point B:** Location of Cloud Instances of vBBUs	**Data Rate:** 1Gb/s To be scaled with 500 Mb/s increment based on utilisation. **Lat.:** 2 ms, e2e of transport link	Cloud1 Cloud2	
1/1/20 - 1/1/21	**FH2, CPRI**	5:00 am-24:00 pm	**Link Location Info** **End point A:** Location of RRHs; Along the train route/lines; **End point B:** Location of Cloud Instances of vBBUs	**Data Rate:** 500Mbps To be scaled with a 200 Mb/s increment based on utilisation. **Lat.:** 20 ms, e2e of transport link	Cloud1 Cloud2	
1/1/20 - 1/1/21	**Cloud1 (for vBBUs)**	Cont.	**Location of the "End Users":** Location of ANNs	**CPU:** 4 VCPUs To be scaled with a 1 VCPU increment based on utilisation. **RAM:** 12 GB **Storage/ HD Space:** 70 GB	FH1	**Availability:** 5 min/month **Reliability:** < 1 time/month **Monthly Availability %:** 99.9%
1/1/20 - 1/1/21	**Cloud2 (for Core Network Instances)**	Cont.	**Location of the "End Users"**: Location of the Access Network Nodes	**CPU:** 4 VCPUs **RAM:** 12 GB **Storage/ HD Space:** 70GB	BH, FH1, FH2	
1/1/20 - 1/1/21	**Cloud3 (for Apps)**	Every 2nd weekend	**Location of the "End Users":** Nationwide, at the NN stadium etc.	**CPU:** 4 VCPUs **RAM:** 12GB **Storage/ HD Space:** 70GB	-	**Availability:** 30 min/month **Reliability:** < 1 time/month **Monthly Availability %:** 95%

4.1.3.3 TYPE C SLA: End-to-End Telecommunication Services and Cloud Services

The customers of this type of services would be MVNOs or other tenants (as in the previous cases) as well as Industry 4.0 operators, emergency services providers, security/surveillance services providers, multimedia content services providers/broadcasters actually, leasing end-to-end telecommunication services and possibly cloud services for hosting their applications. The SLAs of this type will be very similar to existing network services and cloud services SLAs. However, exploiting the 5G-XHaul capabilities, these SLAs can be dynamic in time and in terms of resources provided and QoS levels. The various parameters and the values of the SLAs will correspond to the specificities of the services to be supported per case.

4.1.3.4 Auxiliary Services in SLAs

Besides connectivity and cloud services, synchronisation over the 5G-XHaul transport network should also be provided as a service – thus under the terms and conditions and with the quality defined in a SLA – in specific use cases. The details of the 5G-XHaul synchronisation framework have been presented in the 5G-XHaul document "*Synchronisation Framework over Heterogeneous Fronthaul and Backhaul*" [69].

In addition to mobile communication services, verticals such as the financial sector, the electric power sector with regards to Smart grid operations, the media and broadcasting sector, and Industry 4.0, timing synchronisation for their applications is of paramount importance. Delivering timing synchronisation as a service over the 5G-XHaul transport network could be considered as an additional service; thus can be either included in the aforementioned SLAs, or can be defined in a separate SLA. The parameters to be defined in such a SLA would be the following:

- The Synchronisation Framework/Technology:
  - Time/Frequency/Phase Synchronisation.
  - Synchronisation protocol.
- The Time Error.
- Security aspects when delivering synchronisation for the application layer.
- Availability.
- Reliability.

5 CP impact on 5G-PPP KPIs

The 5G-PPP contract [67] includes the following KPIs:

- K1: 1000x area capacity.
- K2: Average service creation from 90 hours to 90 minutes.
- K3: Maturation of NetOS, NFV and more SDN capable devices.
- K4: 10 % lower energy consumption with respect to 2010.
- K5: Improvements in automated network control, reduced signalling and OPEX by at least 20% compared to today.
- K6: Integrated Net/Comp/Storage under unified programmable infrastructure.
- K7: Very dense deployments to connect > 7 trillion wireless devices serving over 7 billion people.
- K8: E2E latency reduced 5x (< 1 ms).
- K9: Integration of satellite to improve range/reach of services.
- K10: Improved FH/BH. Architectural enhancements for multi-operator, multi-RAN.
- K11: Increase spectral availability. Efficient HW for mmWave communications.
- K12: Improved optical performance in access, aggregation, core and DC.

Next, we highlight how the innovations presented in this deliverable impact some of these KPIs, especially those related to control plane issues.

- *K2: Average service creation from 90 hours to 90 minutes*
  - The 5G-XHaul e2e control plane introduced in section 2.1 directly addresses this KPI by allowing to automate the provisioning of connectivity services across heterogeneous technology domains. We have demonstrated in section 2.1 how we achieve the provisioning of e2e paths across a wireless and a wired domain in less than 500 ms. We posit that, given the observed performance, the 5G-XHaul control plane should be able to keep connectivity provisioning times for more complex networks in the order of several seconds, which largely fulfils the target KPI. Notice that today connectivity provisioning for different domains requires manual effort and coordination between different operations teams, which results in larger service creation times.
  - The 5G-XHaul SLA definition introduced in section 4 also addresses this KPI, since it reduces the negotiation overhead required to set up bilateral SLAs, which is an important component to the overall service creation time. In addition, the mechanisms put in place by the 5G-XHaul control plane allow to monitor for SLA compliance.
- *K3: Maturation of NetOS, NFV and more SDN capable devices*
  - Several efforts carried out in 5G-XHaul have contributed directly to the adoption of SDN capabilities in various types of wireless and optical transport devices. In this deliverable we have demonstrated in section 2.5 a new SDN agent for a WDM-PON which, to the best of our knowledge, represents the state of art for this system.
- *K4: 10 % energy consumption with respect to 2010*
  - In section 3.2 we have demonstrated a ML-based network management technique, which attempts to power down base stations in the periods of low load. We have seen potential network wide energy savings above 20%, with minimal impact on user experience. Our proposed network management technique can therefore contribute other energy saving strategies, e.g. at the device or component level, in order to achieve the target 5G-PPP KPI.
- *K5: Improvements in automated network control, reduced signalling and OPEX by at least 20% compared to today*
  - We posit that the user mobility prediction models introduced in section 3.3 and 3.4, along with the user positioning mechanisms introduced in section 3.5, are an important enabler for network automation. We have provided a concrete example in section 2.4 for the case of automated Radio Resource Management for dense Small Cell networks, where we have achieved cell throughput gains above 30% by means of predicting how users will move.

- Reduced signalling is an important feature in SDN architectures, where logically centralised control planes need to communicate with the network elements in the field. In this regard, in section 2.2 we have presented an optimisation method to decide on the placement of a cluster of SDN controllers while trading-off delay and signalling.

- *K6: Integrated Net/Comp/Storage under unified programmable infrastructure*
  - The e2e 5G-XHaul control plane presented in section 2.1 represents a step forward towards a unified programmable compute and network infrastructure. In particular, in a practical system the 5G-XHaul ETN is a function that would be integrated inside the hypervisor of a virtualised compute infrastructure, whereby compute nodes in remote locations would be connected through the distributed transport network featuring the 5G-XHaul control plane. In fact an integration between the 5G-XHaul control plane and OpenStack, a common cloud management system, has been demonstrated in deliverable D5.3 [68].

Another contribution towards a unified programmable network and compute infrastructure is presented in section 2.3, where we describe a mechanism to provide transactional update properties to SDN devices. This feature is essential in large distributed control planes with various controllers. This allows the avoidance of conflicts and being able to roll back to a consistent network state if a complex operation, such as an e2e path setup, fails at some of its intermediate steps.

6 Summary and conclusions

This deliverable completes the work in the 5G-XHaul control plane, putting forward a detailed specification and demonstration of the end-to-end 5G-XHaul control plane, which provides connectivity across heterogeneous domains – wireless (Sub-6 and mmWave) and optical. The principles of the control plane and the main architectural building blocks were those captured in deliverables D3.1 [1] and D3.2 [2], respectively.

This deliverable provides as well a set of mechanisms belonging to the higher levels of the 5G-XHaul control plane hierarchy. These allow the negotiation of SLAs between the stakeholders involved in the services, namely the 5G-XHaul infrastructure provider, owning the 5G-XHaul transport network, and the tenant, which could be a MNO. The deliverable includes as well the demonstration of a multi-domain connectivity service between two control plane areas located at NITOS and i2CAT, hence fully validating the 5G-XHaul hierarchical control plane. Additionally, this deliverable proposes some additional mechanisms related to the lower levels that complement those reported in deliverable D3.2 [2], such as the WDM-PON southbound interface and the design of a new protocol to provide transactional network updates to traditional SDN southbound protocols.

Regarding the interaction of the 5G-XHaul transport solution with the mobile network, this deliverable provides the necessary tools required to exploit the information extracted from the RAN. These tools range from the gathering of that information by means of techniques presented in WP4, to the prediction of the users' location and load in the RAN in order to optimise the overall network performance, reduce network wide energy consumption, etc. A concrete example for the case of automated RRM for dense Small Cell networks has been presented, where we have achieved cell throughput gains above 30% by means of predicting how users will move.

To conclude, the work in this deliverable, supported by that presented in previous WP3 deliverables, has successfully achieved a complete definition of the control plane principles and mechanisms, followed by the validation and evaluation of the end-to-end 5G-XHaul control plane. Moreover, the assessment of network management algorithms and services which optimise the performance of the 5G-XHaul network both in optical and wireless domains has been reported.

7 References

[1] 5G-XHaul deliverable D3.1: “Analysis of state of the art on scalable control plane design and techniques for user mobility awareness. Definition of 5G-XHaul control plane requirements”, July 2016.

[2] 5G-XHaul deliverable D3.2: “D3.2 Design and evaluation of scalable control plane, and of mobility aware capabilities and spatio-temporal demand prediction models”, 5G-XHaul Project, June 2017, http://www.5g-xhaul-project.eu/download/5G-XHaul_D3_2.pdf.

[3] Strauss Project, Deliverable 3.2 “Preliminary report on the building blocks for the network virtualization, OpenFlow control and SDN orchestrator”, June 2015.

[4] Learning and Exploiting Time Domain Traffic Patterns in Cellular Radio Access Networks,” in Ma-chine Learning and Data Mining in Pattern Recognition. Lecture Notes in Computer Science, vol 9729. Springer, June 2016.

[5] 5G-XHaul, Deliverable D2.4 – Network Topology Definition, December 2016.

[6] K. Hiltunen, “Utilizing eNodeB Sleep Mode to Improve the Energy-Efficiency of Dense LTE Networks”, in Proc. of 24th IEEE Personal Indoor and Mobile Radio Communications (PIMRC), 2013.

[7] “Global transport SDN prototype demonstration,” OIF / ONF White Paper, 2014.

[8] M. Maternia, J.F. Montserrat, et. al. “Performance evaluation framework,” D2.1 of METIS-II project, January 2016.

[9] E. Liotou et al, "QoE-SDN APP: A Rate-guided QoE-aware SDN-APP for HTTP Adaptive Video Streaming," in IEEE Journal on Selected Areas in Communications, vol. PP, no. 99, pp. 1-1.

[10] Network Implementation Testbed using Open Source platforms (NITOS). https://nitlab.inf.uth.gr/NITlab/nitos.

[11] The Internet Topology Zoo. http://www.topology-zoo.org/dataset.html.

[12] Y. Li et. al, “Radio Resource Management Considerations for 5G Millimeter Wave Backhaul and Access Networks”, Communications Magazine, IEEE, June 2017.

[13] Y. Li et al, "A Joint Scheduling and Resource Allocation Scheme for Millimeter Wave Heterogeneous Networks," 2017 IEEE Wireless Communications and Networking Conference (WCNC), San Francisco, CA, 2017, pp. 1-6.

[14] H. Chiu, K. Yeung, K.-S. Lui, “J-CAR: an efficient joint channel assignment and routing protocol for IEEE 802.11-based multi-channel multi-interface mobile ad hoc networks”, IEEE Transactions on Wireless Com-munications, 8 (4) (2009), pp. 1706–1715

[15] E. Pateromichelakis, K. Samdanis, Q. Wei and P. Spapis, "Slice-Tailored Joint Path Selection & Scheduling in mm-Wave Small Cell Dense Networks," GLOBECOM 2017 - 2017 IEEE Global Communications Conference, Singapore, 2017, pp. 1-6.

[16] E. Pateromichelakis, M. Shariat, A. Ul Quddus and R. Tafazolli, "Joint routing and scheduling in dense small cell networks using 60 GHz backhaul," *2015 IEEE International Conference on Communication Workshop (ICCW)*, London, 2015, pp. 2732-2737.

[17] H. Abu-Ghazaleh and A. S. Alfa, “Application of mobility prediction in wireless networks using markov renewal theory,” IEEE Transactions on Vehicular Technology, vol. 59, no. 2, pp. 788–802, 2010.

[18] Hongbo SI, Yue WANG, Jian YUAN and Xiuming SHAN, "Mobility Prediction in Cellular Network Using Hidden Markov Model," 2010.

[19] Theodoros Anagnostopoulos, Christos Anagnostopoulos, Stathes Hadjiefthymiades, "Mobility Prediction based on Machine Learning," 2011.

[20] L. Ghouti, "Mobility Prediction Using Fully-Complex Extreme Learning Machines," 2014.

[21] Nancy Samaan, and Ahmed Karmouch, "A Mobility and Prediction Architecture and Based on Contextual and Knowledge and Spatial Conceptual and Maps," 2005.

[22] Blasch, Erik and Dezert, Jean and Pannetier, "Overview of Dempster-Shafer and belief function tracking methods," 2013.

[23] I. F. Akyildiz and W. Wang, “The predictive user mobility profile framework for wireless multimedia networks,” IEEE/ACM Transactions On Networking, vol. 12, no. 6, pp. 1021–1035, 2004.

[24] N. P. Kuruvatti, A. Klein, J. Schneider, and H. D. Schotten, “Exploiting diurnal user mobility for predicting cell transitions,” pp. 293–297, 2013.

[25] N. Wilson, “Algorithms for dempster-shafer theory,” in Handbook of defeasible reasoning and uncertainty management systems, pp. 421–475, Springer, 2000.

[26] N. Samaan and A. Karmouch, “A mobility and prediction architecture and based on contextual and knowledge and spatial conceptual and maps,” IEEE TRANSACTIONS ON MOBILE COMPUTING, VOL. 4, NO. 6, NOVEMBER/DECEMBER, 2005.

[27] J. N. Heendeni, K. Premaratne, M. N. Murthi, J. Uscinski, and M. Scheutz, “A generalization of bayesian inference in the dempster-shafer belief theoretic framework,” pp. 798–804, 2016.

[28] O. C. Ibe, Markov Processes for Stochastic Modeling., vol. 2nd edition. Elsevier, 2013.

[29] P. Buchholz, J. Kriege, and I. Felko, Phase-Type Distributions, pp. 5–28. Cham: Springer International Publishing, 2014.

[30] Tong Liu, Paramvir Bahl, and Imrich Chlamtac, "Mobility Modeling and Location Tracking and Trajectory Prediction and in Wireless and ATM Networks," 1998.

[31] D. Haynes, S. Corns, and G. K. Venayagamoorthy, “An exponential moving average algorithm,” pp. 1–8, 2012.

[32] Injong Rhee, Minsu Shin, Seongik Hong, Kyunghan Lee, Seongjoon Kim, Song Chong, "CRAWDAD dataset ncsu/mobilitymodels (v. 2009 07 23), downloaded from https://crawdad.org/ncsu/mobilitymodels/20090723, https://doi.org/10.15783/C7X302," Jul 2009.

[33] K. Lee, S. Hong, S. J. Kim, I. Rhee, and S. Chong, “SLAW: Self-Similar Least-Action Human Walk,” IEEE/ACM Trans. Netw., vol. 20, no. 2, pp. 515-529, Apr. 2012.

[34] 5G-XHaul deliverable D4.13: “Synchronization and Localization for Cooperative Communications”, April 2017.

[35] 5G-PPP “5G Vision”, Feb. 2015, https://5g-ppp.eu/wp-content/uploads/2015/02/5G-Vision-Brochurev1.pdf.

[36] R. O. Schmidt, ”Multiple Emitter Location and Signal Parameter Estimation”, IEEE Transactions on Antennas and Propagation, Vol. AP-34, No. 3, pp. 276-280, 1986.

[37] A. J. Barabell, ”Improving the resolution performance of eigenstructure based direction-finding algorithms”, in Proc. ICASSP, Boston, MA, pp. 336339, April 1983.

[38] R. Roy, A. Paulraj and T. Kailath, ”Estimation of Signal Parameters via Rotational Invariance Techniques - ESPRIT”, Military Communications Conference - Communications-Computers: Teamed for the 90s (MILCOM), Monterey, pp.41.6.141.6.5, 1986.

[39] T. Nitsche, A. B. Flores, E. W. Knightly and J. Widmer, ”Steering with eyes closed: Mm-Wave beam steering without in-band measurement,” 2015 IEEE Conference on Computer Communications (INFOCOM), Kowloon, pp. 24162424, 2015.

[40] A. Patra, L. Simi and M. Petrova, ”Design and experimental evaluation of a 2.4 GHz-AoA-enhanced beam-steering algorithm for IEEE 802.11ad mm-wave WLANs,” 2017 IEEE 18th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM), Macau, 2017, pp. 1-10.

[41] M. Ramadan, V. Sark, J. Gutierrez and E. Grass, "NLOS Identification for Indoor Localization using Random Forest Algorithm," WSA 2018; 22nd International ITG Workshop on Smart Antennas, Bochum, Germany, 2018, pp. 1-5.

[42] 5G-XHaul deliverable D4.10, “Final report on mm-Wave circuits and systems for high rate point to multipoint links”, March 2018.

[43] ”Phase synchronization capability of TwinRX boards and DoA estimation”, Whitepaper, Available at: www://kb.ettus.com

[44] mmMagic Project, Deliverable D4.2, ”Final radio interface concepts and evaluations for mm-wave mobile communications”, submitted on 30.06.2017.

[45] mmMagic Project, Deliverable D3.1, “Initial concepts on 5G architecture and integration”, delivered on 31.03.2016.

[46] M. Ehrig et al., ”Reliable wireless communication and positioning enabling mobile control and safety applications in industrial environments,” 2017 IEEE International Conference on Industrial Technology (ICIT), Toronto, ON, 2017, pp. 1301-1306.

[47] D. Naddef and G. Rinaldi, “Branch-and-cut algorithms for the capacitated VRP”, In book: *The Vehicle Routing Problem*, Society of Industrial and Applied Mathematics, 2001, pp.29-51.

[48] L. Tassiulas and A. Ephremides, “Stability Properties of Constrained Queuing Systems and Scheduling

Policies for Maximum Throughput in Multihop Radio Networks", *IEEE Tran. on Automatic Control*, 37(12):1936–1948, Dec. 1992.

[49] A. Maltsev, E. Perahia, R. Maslennikov, A. Lomayev, A. Khoryaev, and A. Sevastyanov, "Path Loss Model Development for TGad Channel Models," IEEE 802.11-09/0553r1, May 2009.

[50] T. Taleb, et al., "On Multi-access Edge Computing: A Survey of the emerging 5G Network Edge Cloud Architecture and Orchestration", IEEE Communications Surveys and Tutorials, Vol.19, No.3, pp.1657-1681, 3rd Quarter 2017.

[51] Cisco Visual Networking Index: Forecast and Methodology, 2016-2021, Sept. 2017.

[52] NGMN 5G White paper, 2015.

[53] M. Seufert, S. Egger, M. Slanina, T. Zinner, T. Hoßfeld, and P. Tran-Gia, "A survey on quality of experience of HTTP adaptive streaming," IEEE Commun. Surv. Tutorials, vol. 17, no. 1, pp. 469–492, Jan. 2015.

[54] M.Curic, Z. Despotovic, A. Hecker, G. Carle, "Transactional Network Updates in SDN", accepted at IEEE EuCNC 2018.

[55] ACIDrepo, https://github.com/OVSacid/ACIDRepo, 2018.

[56] L. Schiff, S. Schmid, and M. Canini, "Ground control to Major Faults: Towards a fault tolerant and adaptive SDN control network," in Dependable Systems and Networks Workshop, 2016 46th Annual IEEE/IFIP International Conference on, pp. 90–96, IEEE, 2016.

[57] L. Schiff, S. Schmid, and P. Kuznetsov, "In-band synchronization for distributed sdn control planes," SIGCOMM Comput. Commun. Rev., vol. 46, no. 1, pp. 37–43, Jan. 2016.

[58] M. Canini, P. Kuznetsov, D. Levin, and S. Schmid, "Software transactional networking: Concurrent and consistent policy composition," in HotSDN '13. New York, NY, USA: ACM, 2013, pp. 1–6.

[59] T. Mizrahi and Y. Moses, "Software defined networks: It's about time," in IEEE INFOCOM 2016, April 2016, pp. 1–9.

[60] R. Nadiv and T. Naveh, "Wireless backhaul topologies: Analyzing backhaul topology strategies," White Paper, Ceragon, August 2010.

[61] M. Howard, "Using carrier Ethernet to backhaul LTE," White Paper, Infonetics Research, February 2011.

[62] RFC7223: https://tools.ietf.org/html/rfc7223

[63] ITU-T Std. G.698.1. "Draft new Recommendation ITU-T G.698.4 (ex G.metro)".

[64] 5G-XHaul Project, Deliverable D2.1 "Requirements Specification and KPIs Document", submitted on March 1st, 2016.

[65] ETSI GS NFV 003 V1.2.1.

[66] 5G-XHaul Deliverable D2.1, "Requirements Specification and KPIs Document", November 2015.

[67] 5G PPP contractual agreement. Available online: https://5g-ppp.eu/contract/

[68] 5G-XHaul Deliverable D5.3, "Demonstration and Evaluation of the 5G-XHaul Integrated Prototype", August 2018.

[69] 5G-XHaul Internal Document, "Synchronisation Framework over Heterogeneous Fronthaul and Backhaul", July 2018.

8 Acronyms

Acronym	Description
3GPP	Third Generation Partnership Project
5G	Fifth Generation Networks
5G-PPP	5G Infrastructure Public Private Partnership
ABNO	Application-Based Network Operations
AC	Area Controller
ACID	atomicity, consistency, isolation and durability
ACP	Atomic Commit Protocol
ADC	Analogue-to-Digital Converter
AI	Artificial Intelligence
AoA	Angle of Arrival
API	Application Program Interface
ARQ	Automatic Repeat Request
AVC	Attribute Value Change
BER	Bit Error Rate
BB	Baseband
BBU	Baseband Unit
BH	Backhaul
BS	Base Station
CAPEX	Capital Expenditures
CBR	Constant Bit Rate
CDF	Cumulative Distribution Function
CIR	Committed Information Rate
CN	Core Network
COP	Control Orchestration Protocol
CP	Cyclic Prefix
CPI	Controller Plane Interfaces
CPRI	Common Public Radio Interface
C-RAN	Cloud Radio Access Network (aka Cloud-RAN)
DAC	Digital-to-Analogue Converter
DBMS	Distributed Data Base Management Systems
DC	Data Centre
DFR	Dynamic Flow Rules
DL	Downlink
DRB	Data Radio Bearer

DS	Dempster-Shafer
DTMC	Discrete Time Markov Chain
e2e	end-to-end
eMBB	enhanced Mobile Broadband
eNB	Evolved Node B
EoGRE	Ethernet over GRE tunnel
EPC	Evolved Packet Core
ETN	Edge Transport Node
ETNC	ETN Controller subcomponent
ETT	Expected Transmission Time
FH	Fronthaul
FIFO	First In, First Out
FIB	Forwarding Information Base
FLRR	Fast Local Link Reroute
GUI	Graphical User Interface
IATN	Inter-Area Transport Node
JO	Joint Optimisation
KPI	Key Performance Indicator
L2SID	Layer 2 Segment ID
LA	Local Agent
LoS	Line-of-Sight
LP	Linear Program
LTE	Long Term Evolution
MAC	Medium Access Control
MANO	Management and Network Orchestration
MCs	Macro Cells
MD-SAL	Model Driven Service Abstraction Layer
MEC	Mobile Edge Computing
MIMO	Multiple-Input Multiple-Output
MME	Mobility Management Entity
mmWave	Millimetre Wave
MNO	Mobile Network Operator
MPLS	Multiprotocol Label Switching
MSE	Mean Squared Error
NBI	North-Bound Interface
NFV	Network Function Virtualisation
NGMN	Next Generation Mobile Networks

NLoS	Non-Line-of-Sight
NMS	Network Management System
NPU	Network Processor Unit
NS	Network Service
ODL	OpenDayLight
OLT	Optical Line Terminal
ONF	Open Networking Foundation
ONU	Optical Network Unit
OS	Operating System
OTN	Optical Transport Network
P2P	Point-to-Point
PBB	Provider Backbone Bridge
PCM	Path Computation Manager
PIR	Peak Information Rate
PM	Performance Monitoring
PON	Passive Optical Network
PRB	Physical Radio Block
QoE	Quality of Experience
QoS	Quality of Service
QoT	Quality of Transmission
RAN	Radio Access Network
RAT	Radio Access Technology
RCA	Resource Control Agent
RCP	Resilient Control Plane
RF	Radio Frequency
RM	Resource Manager
SDN	Software Defined Networking
SDR	Software Defined Radio
SINR	Signal to Interference plus Noise Ratio
SLA	Service Level Agreement
SLO	Service Level Objectives
SLAE	Sub-Wavelength Lambda Allocation Engine
SLNR	Signal to Leakage plus Noise Ratio
SHPH	Shortest Path Heuristic Algorithm
SLAW	Self-similar Least-Action Walk
SNR	Signal-to-Noise Ratio
SON	Self-Organized Network

SRB	Signaling Radio Bearer
SRRC	Squared Root Rised Cosine
SS2PL	Strong Strict Two-Phase Locking
STP	Spanning Tree Protocol
TAF	Transport Adaptation Function
TC	Transport Class
TCAM	Ternary Content-Addressable Memory
TDD	Time Division Duplex
TE	Traffic Engineering
TED	Traffic Engineering Database
TEID	Tunnel End Point ID
TM	Topology Manager
TNC	TN Controller subcomponent
ToA	Time of Arrival
ToF	Time of Flight
TSON	Time-Shared Optical Network
TSO	Two-Stage Optimization
TSPEC	Traffic Specification
TWR	Two-way ranging
UE	User Equipment
UL	Uplink
URLLC	Ultra Reliable Low Latency Communications
vDP	Virtual datapath
VIM	Virtualized Infrastructure Manager
VLAN	Virtual Local Area Network
VM	Virtual Machine
VN	Virtual Network
VNF	Virtual Network Function
VNO	Virtual Network Operator
VNP	Virtual Network Provider
WDM	Wavelength Division Multiplexing
WP	Work Package