



IoRL deliverable D3.2

Building Network Services – Interim report

Editors:	Nawar Jawad – Benjamin Meunier Brunel University London(UBRUNEL)
Deliverable nature:	Report (R)
Dissemination level: (Confidentiality)	Public (PU)
Contractual delivery date:	31 May 2019
Actual delivery date:	12 June 2019
Suggested readers:	network experts and engineers, telecommunication network operators, telecommunication equipment vendors, such as Nokia-Siemens, Samsung and Huawei
Version:	1.0
Total number of pages:	104
Keywords:	IoRL home network, NFV, orchestration, open stack, preliminary implementation, SDN, use case scenarios, VLC

Abstract

This document describes the Internet of Radio Light (IoRL) services together with the architectural design, load balancing process and the envisioned IoRL slicing architecture. Services design utilizes the Software Defined Network (SDN) and Network Function Virtualisation (NFV) technologies to offer smart services. The preliminary testbed on which the IoRL home network is implemented and tested is introduced and service deployments are presented.

Disclaimer

This document contains material, which is the copyright of certain IoRL consortium parties, and may not be reproduced or copied without permission.

All IoRL consortium parties have agreed to full publication of this document.

The commercial use of any information contained in this document may require a license from the proprietor of that information.

Neither the IoRL consortium as a whole, nor a certain part of the IoRL consortium warrant that the information contained in this document is capable of use, nor that use of the information is free from risk, accepting no liability for loss or damage suffered by any person using this information.

The EC flag in this document is owned by the European Commission and the 5G PPP logo is owned by the 5G PPP initiative. The use of the EC flag and the 5G PPP logo reflects that IoRL receives funding from the European Commission, integrated in its 5G PPP initiative. Apart from this, the European Commission and the 5G PPP initiative have no responsibility for the content of this document.

The research leading to these results has received funding from the European Union Horizon 2020 Programme under grant agreement number 761992 — IoRL — H2020-ICT-2016-2018/H2020-ICT-2016-2.

Impressum

Internet of Radio Light

IoRL

WP3 Software Defined Home Network

Task 3.2 End-to-end Traffic Steering through SDN

Editor: Nawar Jawad - – Benjamin Meunier UBRUNEL

Work-package leader: Charilaos Zarakovitis - NCSR

Estimation of PM spent on the Deliverable:

Copyright notice

2017 - 2019 Participants in IORL project

Executive summary

This document describes the services within the IoRL platform, their concept, interactions and integrations. The descriptions of the interfaces between the system layers are also included. The deployment options for IoRL as an indoor small cell solution are presented, highlighting the pros and cons of each deployment option. The document also includes mmWave localization algorithms and techniques. A theoretical network modelling is also provided, specifically including VLC, mmWave and WiFi channels, with the consideration of signal blockage. We present network-layer optimization implementations and performance evaluations with examples. Finally, the document concludes with network slicing implementation mechanisms for the IoRL network.

List of authors

Company	Author	Contribution
Brunel University	Nawar Jawad Mukhald Salih Benjamin Meunier John Cosmas	Building network services/Follow me service VR services
NCSR	Charilaos Zarakovitis Themistocles Anagnostopoulos Anastasios Kourtis	Load balancing SDN-NFV platform implementation The envisioned IoRL slicing architecture
Joda	Mathias Lacaud	Multi-source streaming over RRLH
Warsaw University of Technology	Krzysztof Cabaj, Marcin Gregorczyk, Wojciech Mazurczyk, Piotr Nowakowski, Piotr Żórawski	VNF for security monitoring
Fraunhofer Institute for Integrated Circuits IIS	Rudolf Zetik Ali Eltohamy	MmWave Localisation
Leicester University	Hequen Zhang	L3-L2 interface

Table of contents

IOURL HOME NETWORK	13
INDOOR ENVIRONMENT	13
RRLH ACCESS POINT	13
VIRTUAL INFRASTRUCTURE	13
1 INTRODUCTION	14
1.1 OBJECTIVE OF THIS DOCUMENT	14
1.2 STRUCTURE OF THIS DOCUMENT	14
2 SDN – NFV PLATFORM	15
2.1 <i>SDN – NFV platform concepts</i>	15
2.2 <i>Development of the SDN/NFV Home Environment (intermediate) – Platform, Concepts and Implementation</i>	15
2.3 NETWORK DESIGN ARCHITECTURE	16
2.3.1 <i>Provider Network</i>	17
2.3.2 <i>VLC Network</i>	17
2.3.3 <i>Wi-Fi Network</i>	17
2.3.4 <i>MANO_net and Service_net Networks</i>	18
2.4 MANO LAYER COMPONENTS AND ARCHITECTURE	18
2.4.1 <i>SDN controller</i>	18
2.4.2 <i>NFVO</i>	19
2.5 <i>VNFs</i>	19
3 IOURL SMALL-CELL NETWORK SERVICES BUILDING AND NETWORK DEPLOYMENT	21
3.1 BUILDING NETWORK SERVICES	21
3.1.1 <i>Follow me service</i>	21
3.1.2 <i>Multi-source streaming over RRLH</i>	23
3.1.3 <i>VR Service</i>	32
3.1.4 <i>Security services</i>	34
3.2 L3-L2 INTERFACE	44
3.2.1 <i>Interface protocols</i>	44
3.2.2 <i>Control plane</i>	45
3.2.1 <i>Data plane</i>	47
3.3 IOURL AS SMALL CELL DEPLOYED WITHIN MOBILE NETWORK OPERATOR (MNO)	47
3.3.1 <i>Visible Light Communication-based gNB</i>	47
3.4 MMWAVE LOCALIZATION	50
3.4.1 <i>Algorithms for Location server</i>	50
3.4.2 <i>Tool for the RRLH distribution planning with respect to the localization performance</i>	52
4 THE LOAD BALANCING PROCESS AT THE IOURL SDN/NFV HOME ENVIRONMENT	59
4.1 REVIEW OF RELEVANT LITERATURE	59
4.2 EMPIRICAL LOAD BALANCING SCHEME AND IMPLEMENTATION	61
4.2.1 <i>The role of OpenFlow, Ryu controller, iPerf and Mininet in the empirical design</i>	61
4.2.2 <i>Network design using empirical load balancing logic</i>	63
4.2.3 <i>Implementation using empirical load balancing logic</i>	64
4.3 THEORETICAL LOAD BALANCING MODELLING, OPTIMISATION AND IMPLEMENTATION	67
4.3.1 <i>Network and channel modelling</i>	69
4.3.2 <i>Theoretical load balancing schemes</i>	74
4.3.3 <i>Simulation settings</i>	80
4.3.4 <i>Performance evaluations and implementation examples</i>	84
5 NETWORK SLICING IN IOURL PLATFORM	91
5.1 NETWORK SLICING OVERVIEW	91
5.2 SLICING USE CASE SCENARIO ON THE IOURL PLATFORM	93

5.3 REVIEW OF SLICING IMPLEMENTATION TECHNOLOGIES TO BE IMPLEMENTED INTO THE IO RL PLATFORM 93

 5.3.1 *Using OSM for network slicing* 94

 5.3.2 *Using slice manager component* 95

6 CONCLUSIONS **98**

REFERENCES **99**

List of figures and tables

Figure 1 - Actual picture of the Dell R730xd server at NCSR Data centre.....	15
Figure 2- Snapshot of the OpenStack Dashboard	16
Figure 3 - Illustration of the network architecture.....	16
Figure 4 - Snapshot of the network topology at the OpenStack dashboard.....	17
Figure 5 - Snapshot of the Ryu SDN controller at the topology viewer.....	18
Figure 6 - Snapshot of the OSM graphical user interface	19
Figure 7 - Snapshots of OSM's (right) and OpenStack's (left) dashboard for network service instantiation	20
Figure 8 - MS architecture.....	21
Figure 9 - IoRL testbed	22
Figure 10 - MS-Stream for multi-path streaming through the IoRL network.....	23
Figure 11 - Interconnection of MS-Stream modules in the Home IP Gateway.....	24
Figure 12 - Requirements for the VNF.....	25
Figure 13 - Live Streaming use case in IoRL.....	27
Figure 14 - Decomposition of end-to-end streaming delay.....	28
Figure 15 - End-to-end delay of existing video streaming protocols and MS-Stream.....	29
Figure 16 - Benchmark of the evaluation.....	30
Figure 17 - First scenario: slowly walking user.....	30
Figure 18 - Results of the first evaluation. Single path player on the left, MS-Stream player on the right.....	31
Figure 19 - Second scenario: obstacle between the user and the RRLH network.....	31
Figure 20 - Results of the second evaluation. Single path player on the left, MS-Stream player on the right.....	32
Figure 21- 360 VR Live Streaming System diagram	33
Figure 22 - Multiplayer VR System diagram	33
Figure 23 - VR virtual Tourism System diagram.....	34
Figure 24 - VR VNF System integration	34
Figure 25 - IoRL RAN with multiple remote communication techniques in combination with the IoRL IHIPG with SDN, which hosts security VNF.....	35
Figure 26 - presents the ISF elements deployment in the IHIPG system.	36
Figure 27 - Testbed for the DNS-based and forged MAC addresses approaches.....	36
Figure 28 - Sniffing inactive: first 30 seconds -- silent period (left) and next 30 seconds -- flood period (right)	39
Figure 29 - Sniffing active: first 30 seconds -- silent period (left) and next 30 seconds -- flood period (right).....	39
Figure 30 - Sniffing inactive: first 30 seconds -- silent period (left) and next 30 seconds -- flood period (right)	40
Figure 31 - Sniffing active: first 30 seconds -- silent period (left) and next 30 seconds -- flood period (right).....	40
Figure 32 - Mean-based ROC	41
Figure 33 - Median-based ROC	41
Figure 34 - Standard Deviation-based ROC.....	41
Figure 35 - L2/L3(RRC) and SDN Protocol Stack	45
Figure 36 - Header of IPv4 and IPv6.....	45
Figure 37 - QoS Bits in Differentiated Service of IP Header.....	45
Figure 38 - VLC-gNB small cell.....	48
Figure 39 - AC- VLC-gNBs topology	48
Figure 40 – Dual Connectivity topology	49
Figure 41 - VLC-gNB Distributed Units topology.....	50
Figure 42 - Switched architecture for the TDoA estimation using mmWave	51
Figure 43 - Example of mmWave positioning of an UE by 5 RRLHs delivering 10cm.....	52
Figure 44 - Examples of GDOPs for two different RRLH constellations.....	53
Figure 45 - Updates of particle positions and velocities.....	54
Figure 46 - Optimization scenario with 2 RRLH's	54
Figure 47 - Distribution of 40 particles during the first 3 iterations.....	55
Figure 48 - Evolution of X1-coordinate	55
Figure 49 - Distribution of 40 particles during the last 10 iterations	55
Figure 50 - GDOP for Rx1 at [0,0] and Rx2 at [10,0] Intuitively the best placement.....	56
Figure 51 - GDOP for the optimization result with Rx1 at [0.95,0] and Rx2 at [8.94,0]	56
Figure 52 - GDOP for Rx1 at [0,0] and Rx2 at [10,0] Intuitively the best placement.....	56
Figure 53 - GDOP for the optimization result with Rx1 at [0.95,0] and Rx2 at [8.94,0]	57

Figure 54 - GDOP for Rx1 at [0,0] and Rx2 at [10,0] Intuitively the best placement.....	57
Figure 55 - GDOP for the optimization result with Rx1 at [0.95,0] and Rx2 at [8.94,0].....	57
Figure 56 - Illustration of the of the SDN/NFV load balancing topology.....	63
Figure 57 - Snapshot of the Mininet instantiation	64
Figure 58 - Illustration the flow diagram including Ryu controller and load balancing application processes.....	64
Figure 59 - Snapshot of the load balancing application with options for traffic redirection	65
Figure 60 - Snapshot of the data traffic in the IoRL home network per interface before load balancing.....	66
Figure 61 - Snapshot of the data traffic in the IoRL home network per interface after load balancing	67
Figure 62 - Representation of conventional load balancing topology with intensive L2 processing for access point indexing (left), and proposed load balancing approach with joint network access and access point indexing at L3 (right)	69
Figure 63 - Representation of the hybrid VLC/mmWave/WiFi system in the considered indoor simulation environment.....	81
Figure 64 - Illustration of the VLC AP illuminance (left) and SNR (right) versus the vertical and horizontal distance of a user from the considered VLC AP based on the VLC channel modelling in Section 6.3.1.1 and the simulation settings in Table 6.	83
Figure 65 - Illustration of the mmWave beam power density (left) and SNR (right) versus the vertical and horizontal distance of a user from the considered mmWave AP based on the mmWave channel modelling in Section 6.3.1.2 and the simulation settings in Table 6 and Table 8.....	84
Figure 66 - Illustration of (i) the overall system throughput (left), and (ii) fairness index (right) versus the time intervals considering and for POLB and three version of QPLB: priority-unaware, QoS-unaware and priority and QoS aware.....	87
Figure 67 - Illustration of (i) the individual user throughput (left), and (ii) fairness index per user Class (right) versus the time intervals achieved via priority-and-QoS aware QPLB algorithm.	88
Figure 68 - Illustration of (i) the individual user throughput (left), and (ii) fairness index per user Class (right) versus the time intervals achieved via pure opportunistic POLB algorithm.....	89
Figure 69 - Illustration of the implementation complexity: number of iterations and implementation time versus the number of users joining the system.	89
Figure 70 - Representation of QPLB and POLB scenario examples to clarify the AP allocation rational of each algorithm.	90
Figure 71 - Representation of the lifecycle of network slice instance	91
Figure 72 - Representation of the relating the information models between 3GPP and ETSI NFV.....	92
Figure 73 - Illustration of the network slice management process in an NFV framework	93
Figure 74 - Illustration of the OSM E2E orchestration	94
Figure 75 - Illustration of the envisioned slice management IoRL architecture	95
Figure 76 - Illustration of the service creation sequence diagram for the instantiation and the configuration of a network slice	97

List of Tables:

Table 1 - Mean-based detection results (AUC=0.92)	42
Table 2 - Median-based detection results (AUC=0.98)	43
Table 3 - Standard Deviation-based detection results (AUC=0.75)	43
Table 4 - QoS Mapping Table	46
Table 5 - TOS/TC Field Mapping Table	46
Table 6 - Path loss parameters based on statistical model in [Ref] and [Ref]	72
Table 7 - VLC channel simulation settings	82
Table 8 - parameters used for simulation of mmWave channel	83
Table 9 - The user priorities and minimum QoS requirements used during evaluation	84
Table 10 - Normalised channel gains of the VLC (left) and mmWave (right) APs (in dBm) considering the positioning of 20 users given in Table 8 and the positioning of the RRLHs shown in Figure 60	85
Table 11 - The optimal AP allocation solution of the POLB Algorithm 1 (left) and QPLB Algorithm 2 (right). Users highlighted in red are assigned to the WiFi network.	86
Table 12 - Standardised SST values	92

Abbreviations

3D	Three Dimensional
3GPP	3rd Generation Partnership Project
5QI	5G QoS identifier
IoRL	Internet of Radio Light
VLC	Visible light communication
IHIPGW	Intelligent Home IP Gateway
ACK	Acknowledge
AN	Access Network
AP	Access Point
APA	Access Point Assignment
API	Application Programming Interface
AWGN	Additive White Gaussian Noise
CN	Core Network
CPU	Central Processing Unit
CSMF	Communication Service Management Function
DASH	Dynamic Adaptive Streaming over HTTP
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
DSCP	Differentiated Services Code Point
eMBB	enhanced Mobile Broadband
ETSI	European Telecommunications Standards Institute
FI	Fairness Index
FMS	Follow Me Service
FPR	False Positive Rate
GDOP	Geometric Dilution of Precision
GRE	Generic Routing Encapsulation
HIPG	Home Internet Protocol Gateway
HMD	Head Mounted Display
HTTP	HyperText Transfer Protocol
ICMP	Internet Control Message Protocol
IM	Information Model
IP	Internet Protocol

ISF	Integrated Security Framework
KKT	Karush-Kuhn-Tucker
L1	Layer 1 (Physical layer)
L2	Layer 2 (MAC layer)
L3	Layer 3 (Network layer)
LB	Load Balancer
LCM	Life-Cycle Management
LD	Location Database
LED	Light Emitting Diode
LoS	Line-of-site
LS	Location Server
MAC	Medium Access Control
MANO	Management and Orchestration
MINLP	Mixed-Integer Non-Linear Programming
mmWave	Millimetre Wave
MSS	Multiple Source Streaming
NAT	Network Address Translation
NBI	North Bound Interface
NF	Network Function
NFV	Network Function Virtualisation
NFVI	Network Function Virtualisation Infrastructure
NIC	Network Interface Connection
NLoS	Non-line-of-site
NO	Network Ossification
NS	Network Service
NSD	Network Service Descriptor
NSI	Network Slice Instance
NSMF	Network Slice Management Function
NSSMF	Network Slice Subnet Management Function
NST	Network Slice Template
NST	Network Service Template
OFDMA	Orthogonal Frequency Division Multiple Access
OPNFV	Open Platform for NFV

OS	Operating System
OSM	Open Source MANO
OVS	Open Virtual Switch
PC	Personal Computer
PLC	Power-Line Communication
POLB	Pure Opportunistic Load Balancing
PSD	Power Spectral Density
PSO	Particle Swarm Optimisation
QoS	Quality-of-Service
QoE	Quality of Experience
QFI	QoS Flow Identification
QPLB	QoS- and Priority-aware Load Balancing
RAM	Random Access Memory
RAN	Radio Access Network
REST	Representational State Transfer
RF	Radio Frequency
RRC	Radio Resource Control
RRLH	Remote Radio Light Head
RTT	Round Trip Time
SBA	Service Based Architecture
SBI	South Bound Interface
SCTP	Stream Control Transmission Protocol
SDN	Software Defined Network
SEQ	Sequence
SNR	Signal-to-noise ratio
SST	Standard Slice Type
TC	Traffic Class
TCP	Transmission Control Protocol
TDMA	Time Division Multiple Access
TDoA	Time Difference of Arrival
ToA	Time of Arrival
TOS	Type of Service
TPR	True Positive Rate

UDP	User Datagram Protocol
UE	User Equipment
URLLC	Ultra-Reliable Low Latency Communications
VIM	Virtual Infrastructure Manager
VLC	Visible Light Communication
VM	Virtual Machine
VNF	Virtual Network Function
VNFD	VNF Descriptor
VR	Virtual Reality
WAN	Wireless Access Network
WiFi	Wireless Fidelity
VLC-gNB	IoRL small cell

Definitions

IoRL home network

An IoRL home network is a wireless system setting for indoor environments composed by visible light communication (VLC), wireless fidelity (WiFi) and millimeter wave (mmWave) remote radio-light heads (RRLH) access points sharing same virtual infrastructure.

Indoor environment

An indoor environment is a single building setting in which the VLC, WiFi and mmWave access points are placed.

RRLH access point

A RRLH access point is a transceiver operating either using VLC, WiFi or mmWave radio access technology.

Virtual infrastructure

A virtual infrastructure is a system consisted by virtual network components such as controllers, switches, monitoring tools, etc.

1 Introduction

1.1 Objective of this document

The main objectives of this document are to:

- Identify the SDN-NFV platform that has been used to effectively implement the IoRL home network in practice.
- Explain the design, the rationale and the implementation of the network services that has been customized for the IoRL.
- Define the preliminary solutions for IoRL deployment as part of Mobile Networks, outlining the pros and cons of each deployment scenario.

1.2 Structure of this document

The rest of the document is organised as follows:

- Section 2 explains the SDN-NFV IoRL platform and implementation steps.
- Section 3 explores the network services deployed on top of the developed platform, shedding light on the implementation details.
- Section 4 describes the interface between L2 and L3.
- Section 5 specifies the possible deployment scenarios of the IoRL with Mobile Network Operators. Explaining the details of each scenario, and its compliance with standardization.

2 SDN – NFV Platform

In this section, an overview about the concept and the implementation of the Intelligent Home IP Gateway (IHIPGW) of the IoRL is given, to establish a better understanding of the designed small cell system.

2.1 SDN – NFV platform concepts

The purpose of creating the IHIPGW, is to enable the IoRL small cell to offer intelligent services to its users. Therefore, the system design leverages SDN/NFV technologies to enable flexible, adaptive and reconfigurable services. The platform enables third party service providers to utilize system information to customize the offered services. The next section will present the platform implementation.

2.2 Development of the SDN/NFV Home Environment (intermediate) – Platform, Concepts and Implementation

On the technical part, the Intelligent Home IP Gateway of the IoRL architecture is realized by an SDN/NFV platform, implemented on a physical DELL R730xd server (Figure 1).

The server basic specs are as following:

- CPU: 2x Intel(R) Xeon(R) CPU E5-2620 v3 @ 2.40GHz
- Memory: 192 GB RAM
- Disk 1: 240 GB SSD SATA
- Disk 2: 1 TB SATA
- Network I/F:
 - 2x 10GbE
 - 1x GbE



Figure 1 - Actual picture of the Dell R730xd server at NCSR Data centre

The SDN/NFV platform server is running on Ubuntu Linux 16.04LTS, while OpenStack (Queens version) [1] is used as the cloud operating system and NFVI enabler. OpenStack is currently the prevailing open-source cloud controller with a wide ecosystem of services and plug-ins. It is also the most widely used controller for NFV platforms, also a part of the OPNFV (Open Platform for NFV) suite. The services will be deployed on Virtual Machines (VMs), and as Virtual Network Functions (VNFs). Figure 2 depicts the dashboard of the OpenStack Virtualized Infrastructure Manager (VIM), giving an overview of the cloud infrastructure that implements the Intelligent Home IP Gateway. The OpenStack VIM is responsible for controlling and managing the (Network Function Virtualisation Infrastructure) NFVI compute, storage and

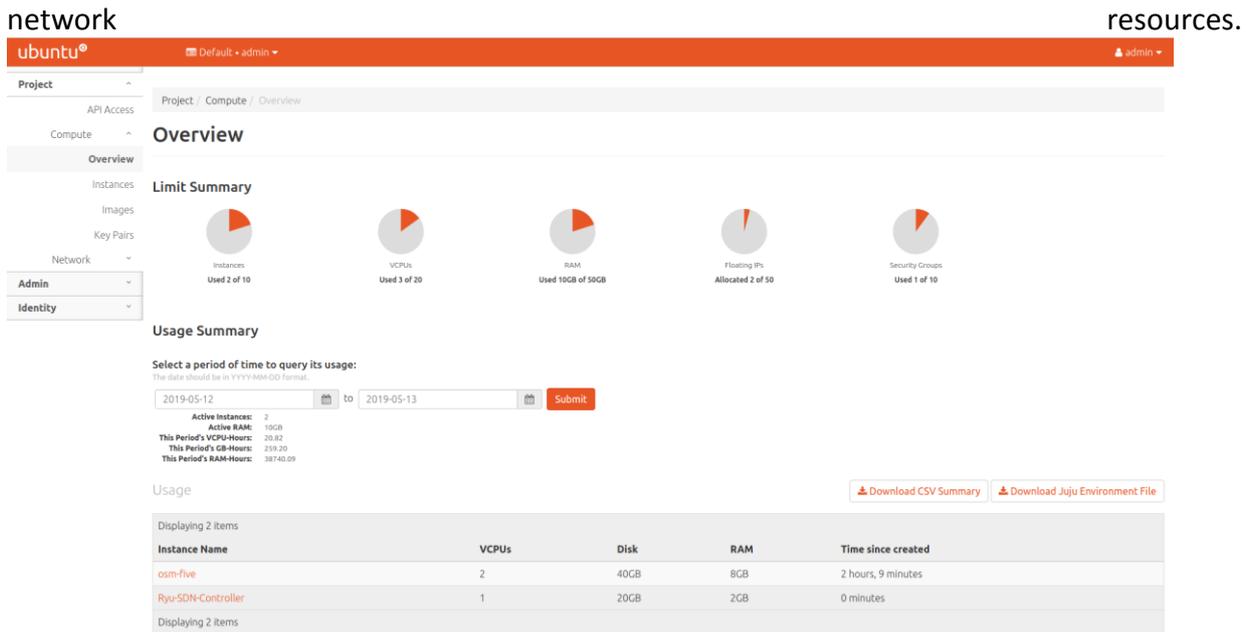


Figure 2- Snapshot of the OpenStack Dashboard

2.3 Network design architecture

The IoRL SDN/NFV platform includes three physical network interfaces, each connected to an external Layer 2 network. In addition to those, two more virtual networks (tenant networks in OpenStack terminology) are created on the NFV Infrastructure, in order to host some Management and Orchestration (MANO) Layer instances, as well as the VNFs that are instantiated on the platform. Figure 3 depicts the detailed Network Architecture, including the IP segments used for each network, while Figure 4 provides an overview of the Network Topology from the OpenStack Dashboard.

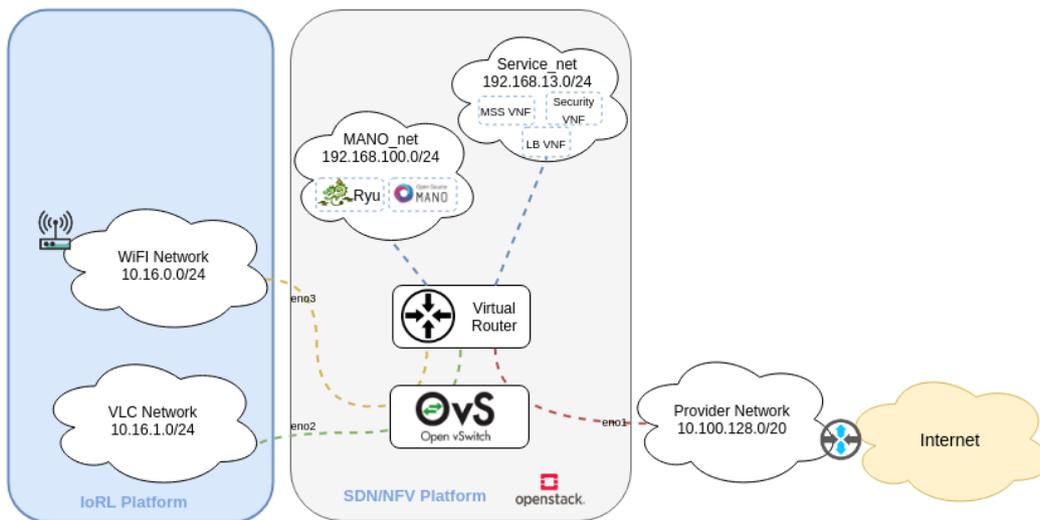


Figure 3 - Illustration of the network architecture

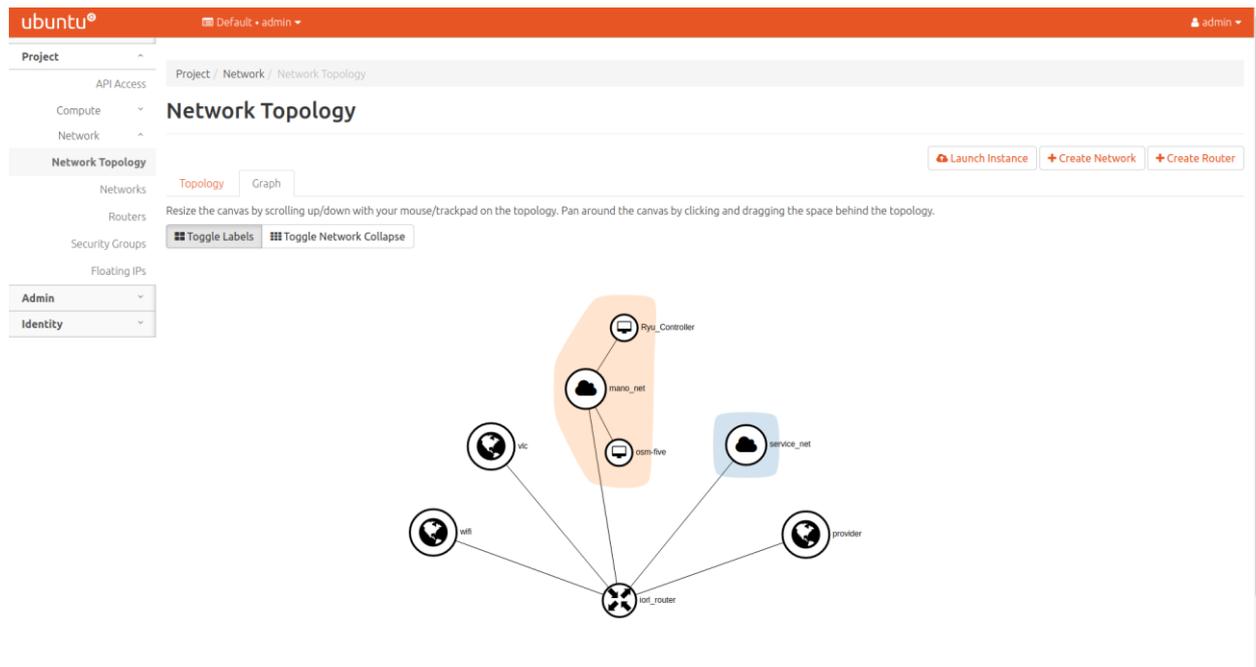


Figure 4 - Snapshot of the network topology at the OpenStack dashboard

The SDN/NFV platform acts as a Layer 3 router for the physical and virtual networks of the IoRL platform. In more details, each network is connected to an OpenStack virtual router [2]. The router's role is to route the traffic between the networks, as well as to act as a gateway for the IoRL platform's internal access networks (VLC network, Wi-Fi Network). An SDN Forwarding Device (SDN switch), on which OpenFlow v1.3 rules can be applied, is also part of the SDN/NFV platform. This SDN Forwarding Device is implemented by an Open Virtual Switch (OVS) on the server, being controlled by a Ryu SDN Controller running as an instance on the NFV Infrastructure.

2.3.1 Provider Network

"Provider" network is a physical Layer 2 network, which is responsible for realising the connection of the IoRL platform to the Internet. The IP segment used for this network is 10.100.128.0/20. The SDN/NFV platform is connected to this network via a 10 Gigabit Ethernet network interface connection (NIC). The interface is mapped to an OVS switch port inside OpenStack, which is then connected to the virtual router.

2.3.2 VLC Network

"VLC" network is another physical Layer 2 network, which implements the VLC/mmWave access network of the IoRL platform. The SDN/NFV server is connected to it via a second 10 Gigabit network interface, which is also mapped to an OVS switch port. The virtual router acts as a gateway to the Internet and NAT point for the "VLC" network. Finally, the IP segment used here is 10.16.1.0/24.

2.3.3 Wi-Fi Network

The "Wi-Fi" network is the last physical Layer 2 network of the IoRL platform. The SDN/NFV server uses a 1 Gigabit network interface to connect to it, while the IP segment for this network is 10.16.0.0/24. The virtual router is a gateway and NAT point for this network as well.

2.3.4 MANO_net and Service_net Networks

“MANO_net” and “Service_net” are two virtual networks inside OpenStack, also connected on the virtual router. MANO_net is the network where all the instances of the MANO Layer are connected. For the IoRL platform purposes, there are two such instances, namely the SDN Controller and the NFV Orchestrator. Service_net is the network which hosts all the VNFs that are deployed on the platform. The segment used for these virtual networks are 192.168.100.0/24 and 192.168.13.0/24 respectively. It is important to note that for getting access to instances attached to these virtual networks from an external network, you have to assign them a public (“floating” in OpenStack terminology) IP via the OpenStack VIM.

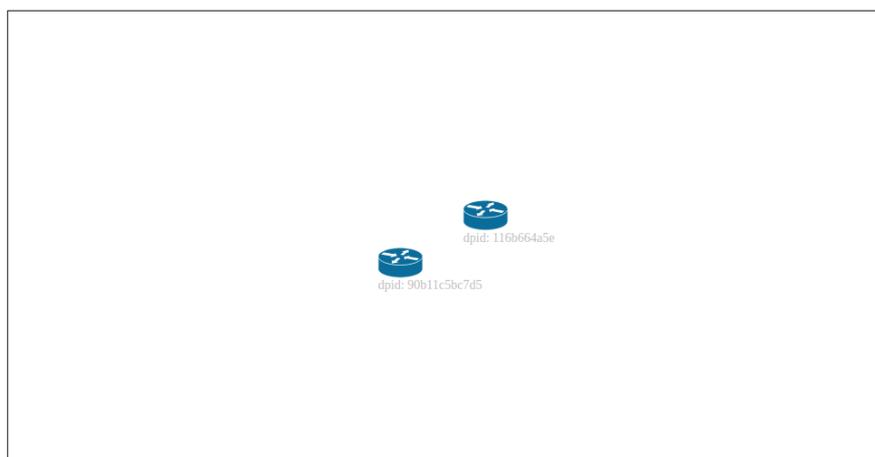
2.4 MANO layer components and architecture

The Management and Orchestration Layer comprises of two instances, the SDN Controller, responsible for controlling the Open Virtual Switch on the NFVI, and the NFV Orchestrator (NFVO), responsible for the instantiation and the Life-Cycle Management (LCM) of the VNFs.

2.4.1 SDN controller

Ryu [3] is the solution that implements the SDN controller on the IoRL SDN/NFV platform. It is deployed on a Virtual Machine on the NFV Infrastructure attached to the MANO_net. Ryu is responsible for controlling the OVS in the server, by installing, updating and deleting SDN rules. It supports OpenFlow version 1.3 for managing network devices and provides software components with well-defined REST APIs that implement the controller’s North Bound Interface (NBI). Various application running as VNFs on the NFV Infrastructure, such as the Load Balancer and the security VNF, can use these REST APIs to create, update or delete SDN rules on the OVS switch. Finally, Ryu offers a graphical topology viewer that can be accessed using any web browser, as depicted in Figure 5.

Ryu Topology Viewer



- { "actions": ["OUTPUT:CONTROLLER"], "idle_timeout": 0, "cookie": 0, "packet_count": 12, "hard_timeout": 0, "byte_count": 3576, "duration_sec": 228, "duration_nsec": 232000000, "priority": 65535, "length": 96, "flags": 0, "table_id": 0, "match": { "dl_type": 35020, "dl_dst": "01:80:c2:00:00:0e" } }
- { "actions": ["POP_VLAN", "OUTPUT:NORMAL"], "idle_timeout": 0, "cookie": 14881932066781862000, "packet_count": 212, "hard_timeout": 0, "byte_count": 20284, "duration_sec": 1442, "duration_nsec": 748000000, "priority": 4, "length": 104, "flags": 0, "table_id": 0, "match": { "dl_vlan": "2", "in_port": 2 } }
- { "actions": [], "idle_timeout": 0, "cookie": 14881932066781862000, "packet_count": 144, "hard_timeout": 0, "byte_count": 12004, "duration_sec": 1444, "duration_nsec": 731000000, "priority": 2, "length": 64, "flags": 0, "table_id": 0, "match": { "in_port": 2 } }
- { "actions": ["OUTPUT:NORMAL"], "idle_timeout": 0, "cookie": 14881932066781862000, "packet_count": 973, "hard_timeout": 0, "byte_count": 77510, "duration_sec": 1444, "duration_nsec": 736000000, "priority": 0, "length": 80, "flags": 0, "table_id": 0, "match": {} }

Figure 5 - Snapshot of the Ryu SDN controller at the topology viewer

2.4.2 NFVO

The NFV Orchestrator component is based on Open Source MANO (OSM), release five [4]. OSM is one of the most popular open-source platforms for NFV orchestration, and, being developed under the ETSI umbrella, is also aligned with the ETSI NFV specifications. OSM controls the life cycle of the VNFs, NSs, and network slices, controls and maintains their configuration, and monitors their in-life health and performance. It offers a North Bound Graphical User Interface providing access to its functionalities as shown in Figure 6.

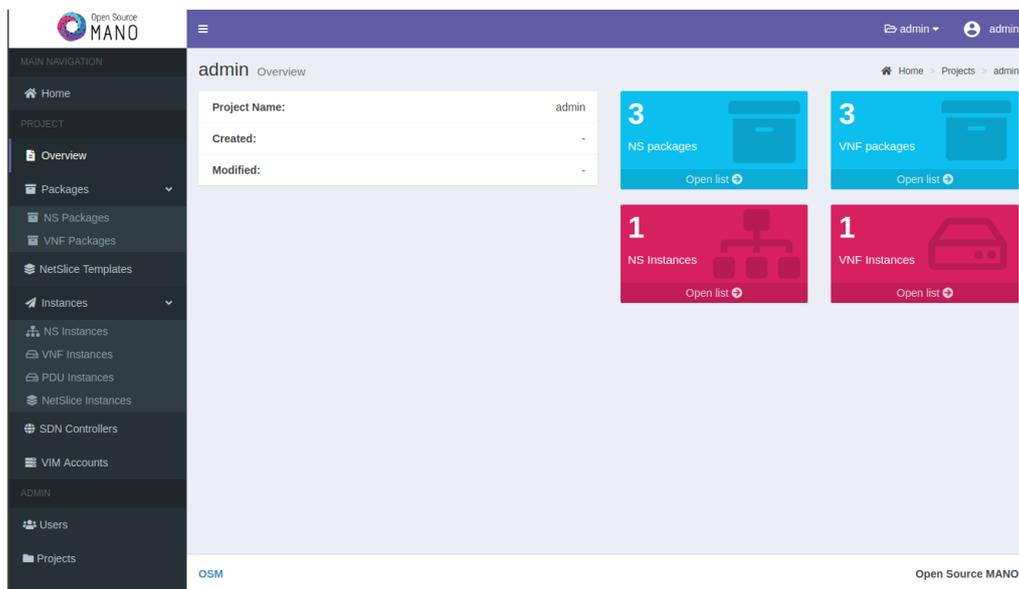


Figure 6 - Snapshot of the OSM graphical user interface

2.5 VNFs

Virtual Network Functions are Virtual Machines, hosted in the SDN/NFV server, that run specific services for the IoRL Platform. They are designed, implemented and distributed in the form of VM image files (e.g. img, raw, qcow2, etc.) by various partners of the project.

Onboarding the VM image to the OpenStack's image repository on the SDN/NFV server, is the first step for the deployment of a VNF. The next step is to create the VNF Descriptor (VNFD) for each VNF and upload it to the OSM VNF repository. VNFD is part of the OSM Information Model (IM), based on the YANG model, that is used to describe various parameters of the VNF to be deployed, such as the VM Image, compute, memory and disk resources, network connection points, etc. The last step is to create the Network Service Descriptor (NSD) and upload it to the repository. NSD is also part of the OSM IM and may include one or multiple VNFs, describing the way they are connected over one or more networks.

After NSD and the referenced VNFDs have been onboarded to the OSM Repository, we can instantiate the Network Service using the OSM Web Interface. Figure 7 illustrates the instantiation of a Network Service (NS). On the top-right, we can see the creation of the NS called "iorl_uc1" on the OSM dashboard. This service is described in the "iorl_uc_nsd" NSD and includes three VNFs, Load Balancer (LB), Multiple Source Streaming (MSS) and Transcoder, which we can see on the bottom-right part of the figure. Finally, on the left part of the figure, we see the creation of three new VMs on OpenStack dashboard, one for each VNF.

The figure consists of two side-by-side screenshots of web dashboards. The left screenshot is from the OpenStack dashboard, showing the 'Instances' page. It features a table with columns: Instance Name, Image Name, IP Address, Flavor, Key Pair, Status, Availability Zone, Task, and Power State. Three instances are listed: 'lorl-uct-3-iorl-transcoder-1' (Transcoder, 192.168.13.7, Build, nova, Spawning, No State), 'lorl-uct-2-iorl-MS-5-1' (MSS, 192.168.13.6, Build, nova, Spawning, No State), and 'lorl-uct-1-iorl-LB-1' (LB, 192.168.13.14, Active, nova, None, Running). The right screenshot is from the OSM dashboard, showing 'NS Instances' and 'VNF Instances' pages. The 'NS Instances' page has a table with columns: Name, Identifier, Nsd name, Operational Status, Config Status, and Detailed Status. One instance 'lorl-uct-1' is shown with Operational Status 'OK' and Config Status 'OK'. The 'VNF Instances' page has a table with columns: Identifier, VNFID, Member Index, NS, and Created At. Three VNF instances are listed: 'lorl_trans_vnf1' (3 members), 'lorl_mss_vnf1' (2 members), and 'lorl_uc_vnf1' (1 member).

Figure 7 - Snapshots of OSM's (right) and OpenStack's (left) dashboard for network service instantiation

In the IoRL SDN/NFV platform we use the OSM to instantiate a Network Service (NS) with one or multiple VNFs and perform LCM actions on these VNFs. In addition to that, OSM acts as a VNF repository, where VNFs related to the IoRL platform are stored. Particularly, at current development stage **the SDN/NFV IoRL home environment includes three VNFs**, namely:

1. VNF for multiple source streaming
2. VNF for ffmpeg transcoding
3. VNF for security

For the sake of clarity, **the programming code of these three VNFs is available at** <https://github.com/H2020-5G-IoRLproject/H2020-IoRL-code/>

3 IoRL small-cell network services building and network deployment

3.1 Building Network Services

This section provides a description of the integration and interaction of the services deployed on the IoRL platform.

3.1.1 Follow me service

Follow Me Service (FMS) is a smart multimedia service designed to improve UE’s QoE by enabling the clients to watch the requested videos on the nearest TV screen without sacrificing the client’s mobility. In other words, FMS client can move freely within the home environment from one room to another and FMS service takes care of delivering the video to the nearest TV, by utilizing the available accuracy of the location estimation mechanism as well as the intelligence of the SDN networking.

3.1.1.1 Service interaction

The FMS interacts with multiple network entities to realize service, such as location server, proxy/Cache serve, SDN controller. FMS service architecture depicted in Figure 8.

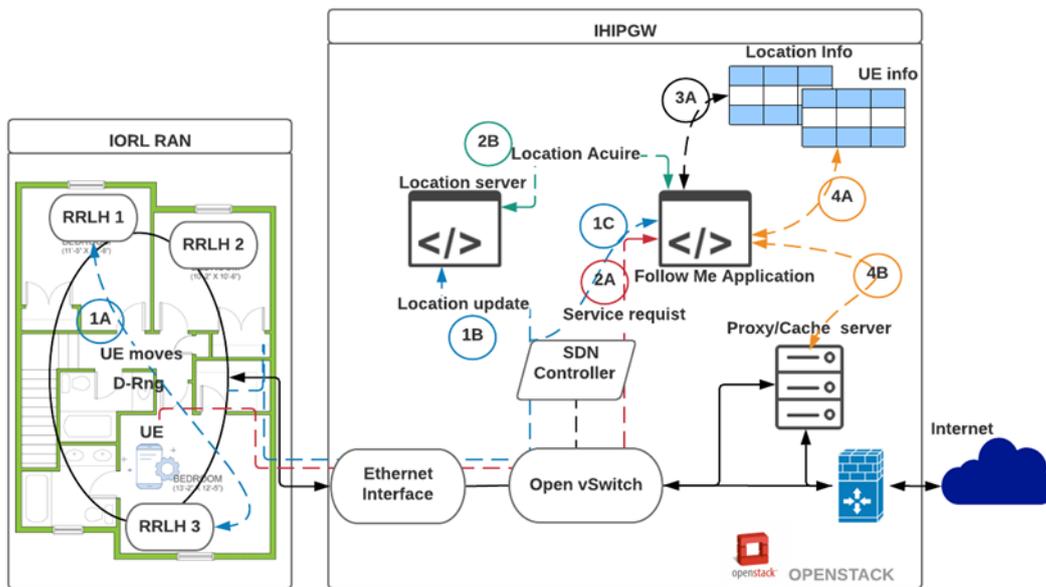


Figure 8 - MS architecture

When a location server sends a location estimation message to the location database, the FMA receives a copy of that message. FMA checks if the UE registered for the FMS by inspecting its local table of the registered UEs, and if a match is found, then it compares the location information, otherwise, the message is ignored (UE is not registered with FMS). If the received message is an initial request message from the UE, then FMA provides the controller with a forwarding policy to configure the OvS to forward the URL request to the proxy server. At the same time, the FMA looks up its local table to check if the UE is a new client or if it is an existing client.

- In case of a new client, FMA sends location acquire message to location database, and stores the UE location information from the returned response message.
- Otherwise, it is an existing client, forwarding to compare location information step.

There are two outcome possibilities when FMA compares the UE current location information against the previously stored location information:

- 1- Client's location information is used by an algorithm to workout TV information (IP and MAC addresses) of the area where the client exists.
- 2- New client has no previous TV information; therefore, the current information is used by the OvS to forward the video to his first location.

If the client's new location is in the same room, then nothing needs to be done because the movement of the UEs does not require traffic redirection. While if the new location is in another room then the follow me procedure is triggered, which includes:

- Storing the current TV information that corresponds to the client's current location.
- Controller modifies the OvS flows to forward the flow to new location. If the UE is not in the FMS enabled area, then the controller sends a pause request message to the proxy server.

3.1.1.2 Service integration

FMS has not completely integrated into the IoRL testbed, since we are in the process of integrating the network entities together. The SDN/NFV server includes the SDN controller as well as the instances, which performs as the database and the location server. The SDN server is connected to the external network (Internet provider), and connected to the other server which deploys 5G L2 processing. The status of the IoRL testbed is depicted in Figure 9.

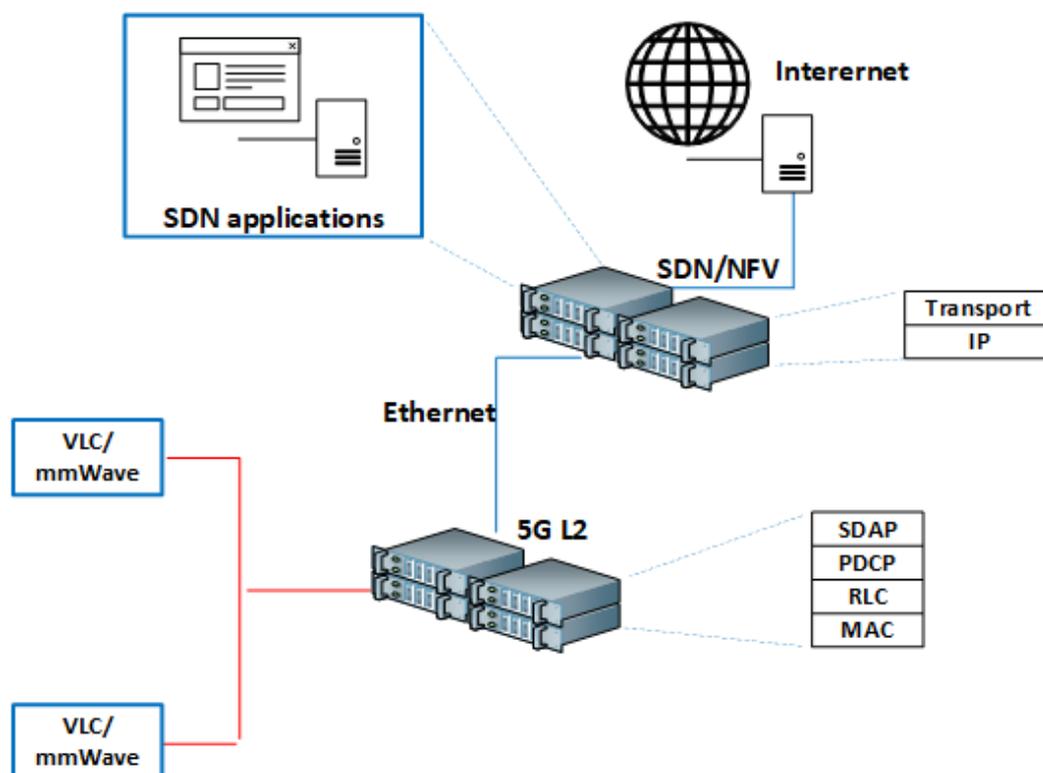


Figure 9 - IoRL testbed

3.1.2 Multi-source streaming over RRLH

In the IoRL Architecture, the video streaming can be managed in two different ways. Firstly, the video can be streamed from the server to the end user using the IoRL system as a black box providing internet access. This first possibility is essential to allow the compatibility with actual streaming services over the internet, and especially video communication tools (UC 1.4, 3.6, 4.2 and 5.4). Secondly, the video can be streamed to the end user using Multiple-Source Streaming (MS-Stream). MS-Stream adds reliability at the application level for the system by streaming sub-flows of video data from different sources or through different paths. Those sub-flows can be read independently, giving a lower video quality, or can be merged, giving a higher video quality.

In the IoRL system, MS-Stream will be used in several use cases to route the video data in high quality through the Remote Radio Light Head network and in low quality through the Wi-Fi access — or any other alternative network — to provide reliability in case of an interruption in the data transmission in the Radio Light network, as shown in Figure 10.

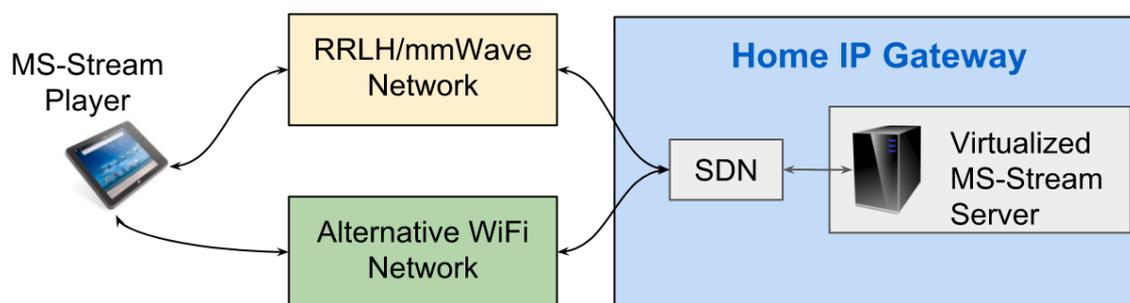


Figure 10 - MS-Stream for multi-path streaming through the IoRL network

During the second period of the project, the MS-Stream solution has been developed both on the server side and on the client side. On the server side, a VNF have been created following the initial description provided in deliverable 3.1. On the client side, a web video player has been developed to be used in computers, smartphone and tablets following the information given in the deliverable 5.1. The video player embeds the MS-Stream algorithms needed to use the VNF effectively to deliver video streams through multi-path. A user interface has been added to control the multiple functionalities of the VNF, as described in deliverable 6.1. The MS-Stream end-to-end solution has been deployed, tested and evaluated in local environments. The MS-Stream VNF also has been validated in the SDN/NFV IoRL platform created by the partners involved in the WP3.

The objectives for the third period of the project regarding MS-Stream will be to deploy and evaluate the solution through the complete IoRL network.

3.1.2.1 MS-Stream VNF

An MS-Stream VNF has been developed during the second period of the IoRL project. This VNF takes the form of a Linux-based qemu virtual machine running several modules in Docker containers. The main modules are the MS-Stream Server and the MS-Stream Transcoder.

3.1.2.1.1 Modules of the VNF

The **MS-Stream Server** is an application that can answer client requests for specific video contents. This module is responsible for the creation of video segments adapted to Multiple-Multiple-Source Streaming. The video segments are created from video data transcoded in numerous qualities. For that purpose, the **MS-Stream Transcoder** comes along with the server. This Transcoder is a module that can transcode input video data into Multiple-Source-ready video data in one or several qualities. Those data then can be pushed to the MS-Stream Server and be available for the MS-Stream Player in the UE, as shown in Figure 11.

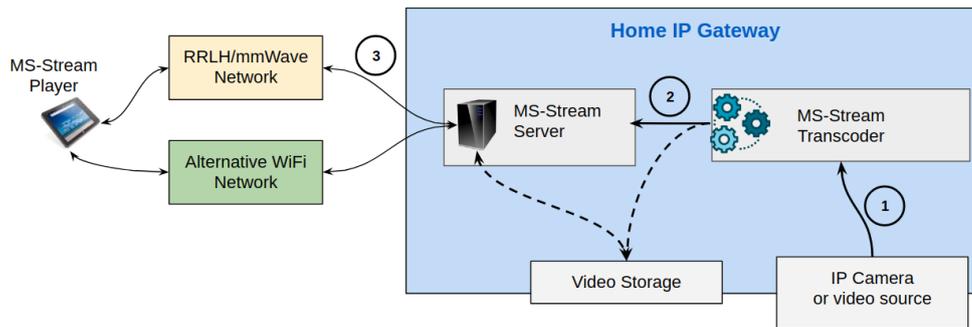


Figure 11 - Interconnection of MS-Stream modules in the Home IP Gateway

Both the server and the transcoder have been written in **NodeJS** [5]. The server is linked with a **MongoDB** [6] database to store the metadata of the available videos. The transcoder uses the **dash muxer** of **ffmpeg** [7] — in addition with the **libx264** [8] open source transcoder — on the video streams to create the Multiple-Source-ready video data.

3.1.2.1.2 Requirements of the VNF

The MS-Stream Server and the MS-Stream Transcoder run on the same virtual machine as a single virtual MS-Stream function. Both functions have some specific needs in terms of hardware resources. The critical needs are in CPU and storage, as shown in Figure 12.

The main resource needed for the transcoding is the CPU. A single 24/7 real time live transcoding in multiple quality for a 4K stream can cost up to 100% of one CPU core. In order to be able to transcode several live streams, the MS-stream VNF needs about 3 or 4 vCPU.

In terms of disk storage, the VNF requires enough storage to save the full data for the videos on demand. For the live streams, the VNF only needs enough space to save a few minutes of data as the server only keeps a window of the last video segments to be delivered to the end users.

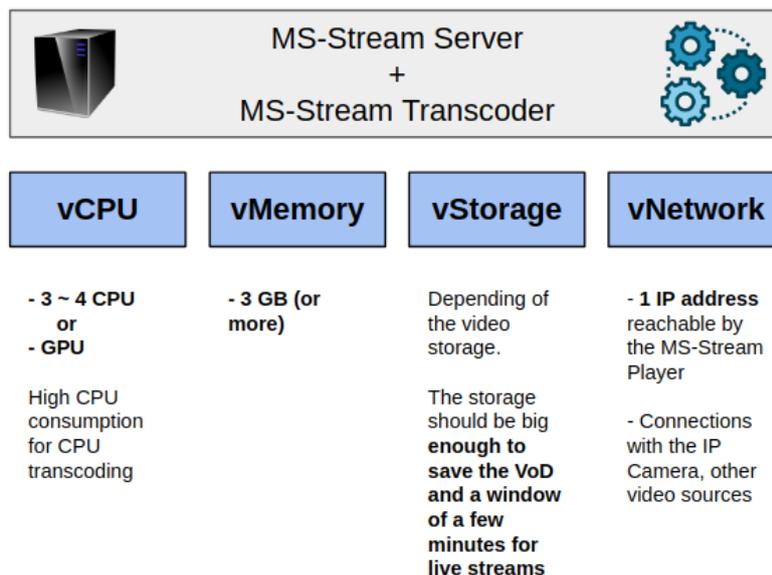


Figure 12 - Requirements for the VNF

3.1.2.1.3 User interface

The video transcoding and video delivery functions are exposed through a REST API and a web interface. The functionalities of the latter are described more precisely in the section 3.3.1 of the deliverable 6.1. Through this web interface, a user is able to start a new transcoding of a video file or a live stream from a camera. Then, the video can be displayed to the end users by a web video player running on the user equipment.

3.1.2.2 Multiple Source Streaming description and algorithms

In addition with the VNF, a MS-Stream video player has been developed for the user equipment side. As MS-Stream is a client centric protocol, the video player is embedding innovative multi-source adaptive algorithms to provide a strong quality of experience for the end users in numerous scenarios where multi-path streaming can be useful.

This subsection is about the description of MS-Stream in the context of IoRL. More detailed information can be found in [9], [10] and [11].

3.1.2.2.1 Multi-Path capability of MS-Stream

MS-Stream is an evolution of HAS solutions (and more specifically, the dynamic adaptive streaming over HTTP–DASH- standard) that simultaneously uses several paths for the download of one video segment. MS-Stream clients request the server through several paths to deliver substreams (referred as descriptions) generated from the existing set of DASH content qualities so as to handle network-path heterogeneity. When retrieved, the requested descriptions are merged in order to reconstruct and display the original requested content quality. In the event of description loss or outdated delivery, content playback continuity is not affected, only image quality is. Additionally, if the considered network paths experience outages or throughput degradation, the MS-Stream client relies on content-adaptation mechanisms to avoid QoE degradation. Thanks to its codec agnosticism and DASH-

compliance, this protocol represents an evolving solution that can be applied to many scenarios.

Prior the streaming session, a manifest file containing information about the available network paths and the video segments is delivered to the client. The MS-Stream content delivery includes the following steps:

- a. The client asks the MS-Stream servers to deliver substreams through network paths.
- b. The MS-Stream server retrieve the segments available in the DASH Storage.
- c. The segments transit on the selected network.
- d. The MS-Stream Aggregator module embedded inside of the player merges the received descriptions so as to reconstruct the original content quality.

Finally, as content is being delivered over N paths, a global and per-path adaptation process is required to deal with path heterogeneity.

Hence, the MS-Stream client is an evolving DASH client, which incorporates a cost-effective segment aggregation module and an adaptation engine capable of content adaptation. From a technical standpoint, MS-Stream ensures DASH-backward compliance: upon the delivery of a regular DASH manifest file, an MS-Stream client can use the single source DASH protocol.

3.1.2.2.2 *MS-Stream adaptation algorithm*

As in the DASH standard, the MS-Stream protocol has adaptation capabilities through a two phase protocol. The first phase consists of **prior-download adaptation decisions** for the upcoming descriptions, composed of two steps: (a) quality selection according to the global bandwidth of the multiple paths and (b) description requests generation to adapt the bitrate of the data to network resources heterogeneity. The second phase consists in performing **in-segment download adaptation** so as to ensure smooth video playback.

The goal of the **prior-download adaptation algorithm** is to create the video segment requests that will be sent to the MS-Stream server.

In a first step, the bandwidth observed on every paths during the download of the last segment is analyzed in order to select the most appropriate video quality. This quality is called the *target quality* and is defined as the quality that should be displayed to the user by the video streaming system. As an example, if the first path provide A Mbps and the second path B Mbps, then the decision is going to be a target quality with a bitrate around $(A + B) * x$ Mbps, where x is a margin for error.

In a second step, the player is willing to select the descriptions to request for every network path. The objective here in the IoRL project is to request a maximum number of segments in the target quality from the RRLH/mmWave network and segments in a lower quality from the alternative network. By doing that, if the connection with the RRLH/mmWave network is optimal, the end user will receive the target quality. On the opposite, if the connection with the RRLH/mmWave network is lost, the player will be able to use low quality segments to display a reliable video streaming experience at the cost of visual quality.

The goal of the **in-segment download adaptation** is to ensure the reliability of the video delivery by being sure there is no pause during the playback. By simultaneously retrieving descriptions from several servers, the probability to receive at least one stream is increased. However, description synchronization is required so as to be resilient to network

heterogeneity and avoid blocking events. A set of three rules has been designed for this synchronization at client side.

(a) For a given content description, if at least one description is retrieved, then other description downloads can be abandoned; this reactive rule ensures the delivery of at least one description before moving on to the next one.

(b) If the buffered content playout reaches a given lower threshold, description downloads can be canceled in order to ensure uninterrupted video experience, hence providing a temporary suboptimal visual quality to the end users; this second reactive rule can only be applied if rule (a) is satisfied.

(c) If the buffered content playout duration exceeds twice the average description duration, then a timeout value is set on HTTP description requests. The timeout value reflects a consumption behavior (aggressive, conservative, etc.) and can be tuned during the streaming session, according to the available buffered content. Once the timeout has elapsed, description requests can be canceled while satisfying rules (a) and (b). This proactive rule enables the use of the buffer to compensate for network characteristic fluctuations on different paths.

3.1.2.3 IoRL use case: MS-Stream for reliable live streaming - QoS/QoE benchmark of the MS-Stream solution

In order to define the interest of MS-Stream in the IoRL project, the player is evaluated for the use case of reliable live streaming. This use case can be found in the UC 1.5, 2.4 and 5.2 of the project. In those use cases, a business actor — or a partner responsible for a demonstration in the project — wants to deliver a live stream to several users connected to the IoRL network on computers, tablets or smartphones, as shown in Figure 13.

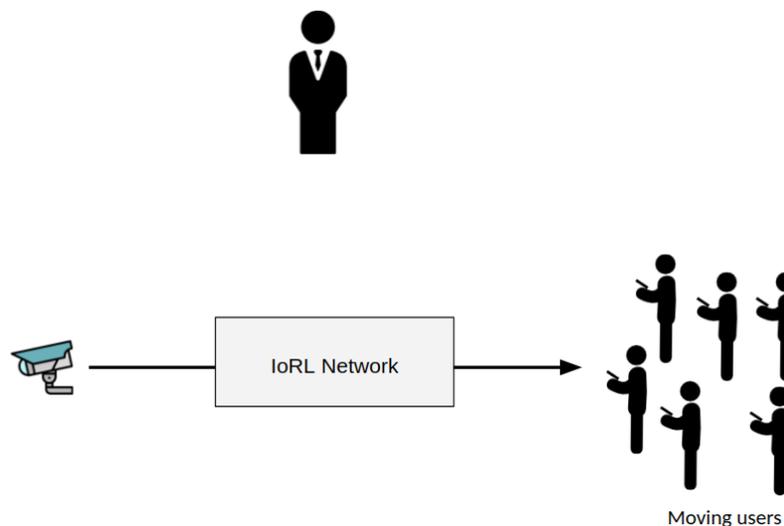


Figure 13 - Live Streaming use case in IoRL

The main goals of video streaming systems is to ensure a great quality of experience for the end users by providing reliability, the highest possible quality and a low delay between the real-time and the video received by the end users.

Reliability can be easily achieved on a stable network connected directly to a fixed TV set or computer. However, it is more complicated when mobile devices and moving end users are the target. MS-Stream is supposed to provide reliability by design by taking advantage of multi-path to deliver adaptive video to the end user without interruptions if one of the available networks is overloaded or not working.

Following the same idea and as described in the previous part, the algorithms embedded in the MS-Stream player tries to provide the best available video quality by sending high quality video segments through the best networks.

Finally, the video delay is another concern with live streaming. In an ideal world, the user would like to have no delay and watch the stream in real time. However, in practice, the delay can come from a number of factors. Only a few video protocols are in real time and it is at the cost of quality and reliability, making them unsustainable for mobile users in the IoRL project.

The following Figure 14 explains where the video delay, called end-to-end video delay, will come from in the project.

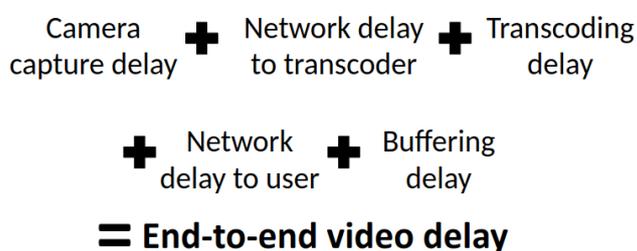


Figure 14 - Decomposition of end-to-end streaming delay

The **end-to-end delay** can be defined as the delay between the time an action happens in reality and the time the same action is watched by an end user. The **camera capture delay** is the delay needed by the camera to capture the picture and send it to the network. Usually, this first delay is reduced to the minimum by the manufacturers. The **network delay to the transcoder** is the network delay needed to send the data to the transcoder. This delay is often similar with the **network delay to the user equipment** and both network delays are supposed to be reduced to the minimum by design in the IoRL RRLH/mmWave network. The transcoding delay is the time needed to transcode the stream and create video segments in several qualities. On top of the computation time, this delay can be significant if long segments are used. If the segment is 3 seconds long, then the transcoder is going to buffer the video data during 3 seconds and to introduce a 3 seconds delay as a consequence. Finally, the **buffering delay** is the delay introduced by the user equipment depending on the video streaming protocol. The goal of the buffering delay is to prevent pauses in the playback by waiting for more data to increase the reliability in case of an unexpected loss of bandwidth.

The average end-to-end delay observed when using the most famous video streaming protocols is presented in Figure 15 below.

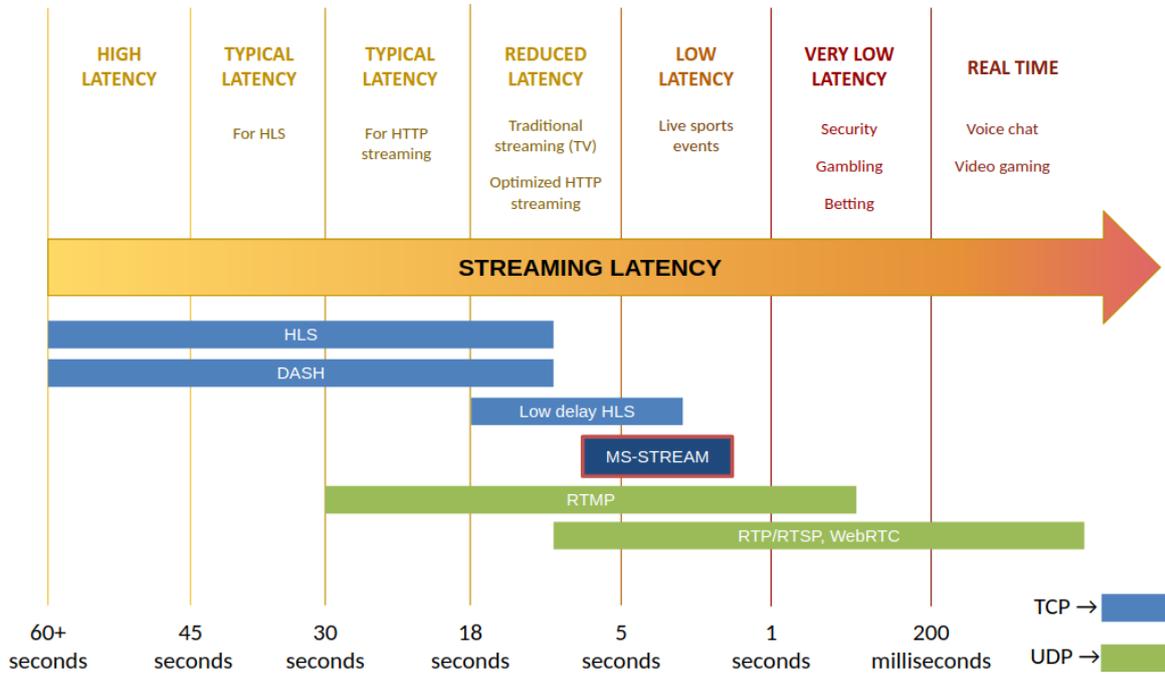


Figure 15 - End-to-end delay of existing video streaming protocols and MS-Stream

The MS-Stream end-to-end delay is usually stable between 1 and 5 seconds. This end-to-end delay is not perfect for very sensitive video streams and can be outperformed in terms of delay by UDP-based protocols in reliable networks. However, this delay is low enough for a majority of live streaming use case and the protocol is better than the most advanced TCP-based protocols.

Even if UDP-based protocols seems to be more efficient for immobile devices, if the network can be considered stable, one main advantage of MS-Stream is the reliability introduced by the multi-path capability for mobile users.

The next parts are about the evaluation of the reliability and the video quality delivered by MS-Stream in order to validate the advantages of the protocol in the context of mobile users connected to indoor wireless networks.

3.1.2.3.1 Performance Benchmark Tests

The goal of the evaluation is to study the impact of MS-Stream in terms of quality of experience for mobile users. In video streaming, the quality of experience (QoE) refers to the subjective perceived quality by the end-users. The QoE includes a lot of criteria. The most important ones are:

- (a) the video stalling due to rebuffering - similar to the reliability
- (b) the average displayed video bitrate - similar to the displayed video quality

In this evaluation, we consider a live streaming from a camera using the MS-Stream VNF. Two video players are connected to one VNF through two network paths. The first path is simulating the IoRL RRLH/mmWave network with a controlled bandwidth (from 0 to 30Mbps). The bandwidth of this network is modified every 200ms according to the different scenarios described in the next sub-section. The second path, on the other hand, is simulating slow alternative network, with a constant 1 Mbps, as shown in top half of Figure 16.

The first player is the MS-Stream player embedding the innovative multi-path algorithms. The second player is single path player slightly improved to periodically test the two networks and potentially select a more appropriate one, as shown in bottom half of Figure 16.

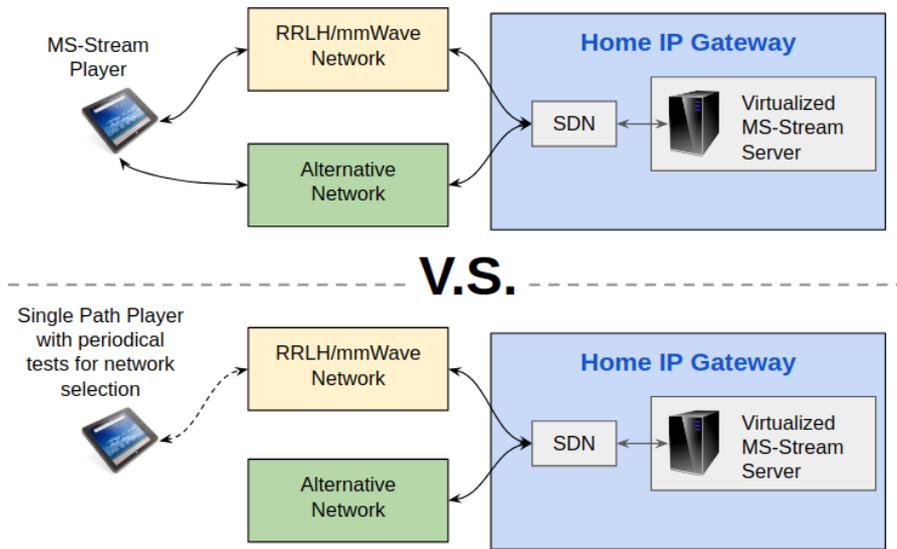


Figure 16 - Benchmark of the evaluation

3.1.2.3.2 *First Scenario: slowly moving user*

On this first scenario, we consider a user moving slowly under the lights, as shown in Figure 17. At the beginning, the user is not under the light and cannot use the high bandwidth of the RRLH/mmWave network. Then, the user is slowly walking under the light before leaving the RRLH/mmWave network again. The RRLH/mmWave network is simulated by a Gaussian with a peak at 30 Mbps when the user is just behind the light.

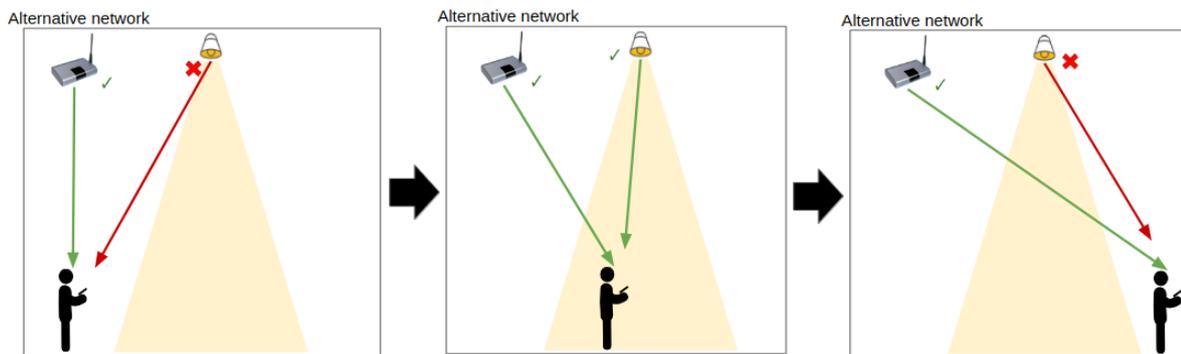


Figure 17 - First scenario: slowly walking user

In the following graphs shown in Figure 18, the cumulative bandwidth of both networks is in orange. The video displayed to the user is shown in blue. The single path player is on the left, and the MS-Stream player on the right

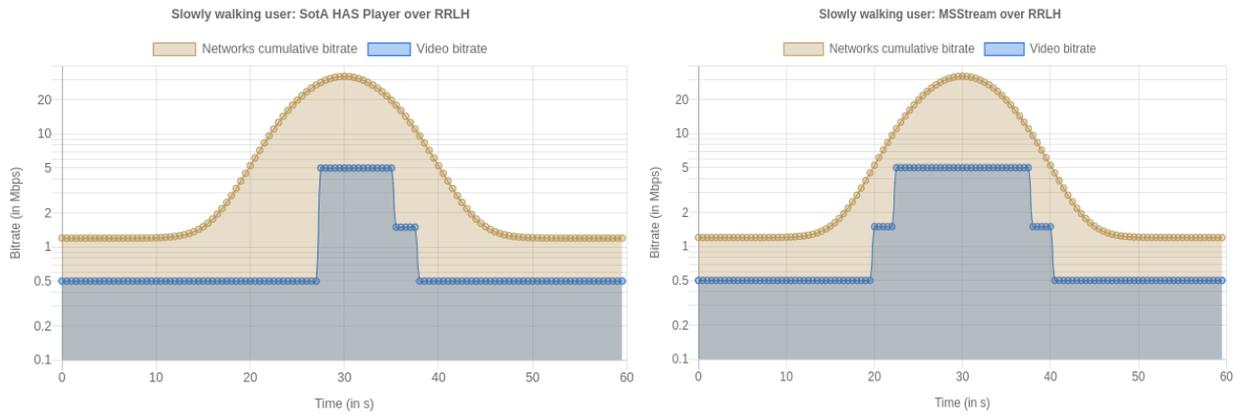


Figure 18 - Results of the first evaluation. Single path player on the left, MS-Stream player on the right

As a result, the MS-Stream player is faster to detect the light because it is connected with both networks. Moreover, the MS-Stream player can be greedier and decrease the quality at the last moment by using its second network as an insurance in case of emergency. As a conclusion, the MS-Stream protocol is able to provide a better video quality in average in this scenario.

3.1.2.3.3 *Second scenario: Obstacle*

On this second scenario, we consider an obstacle moving slowly under the lights, as shown in Figure 19. At the beginning, the user is under the light and can use the high bandwidth of the RRLH/mmWave network. Then, an obstacle is suddenly blocking the signal coming from the RRLH/mmWave network.

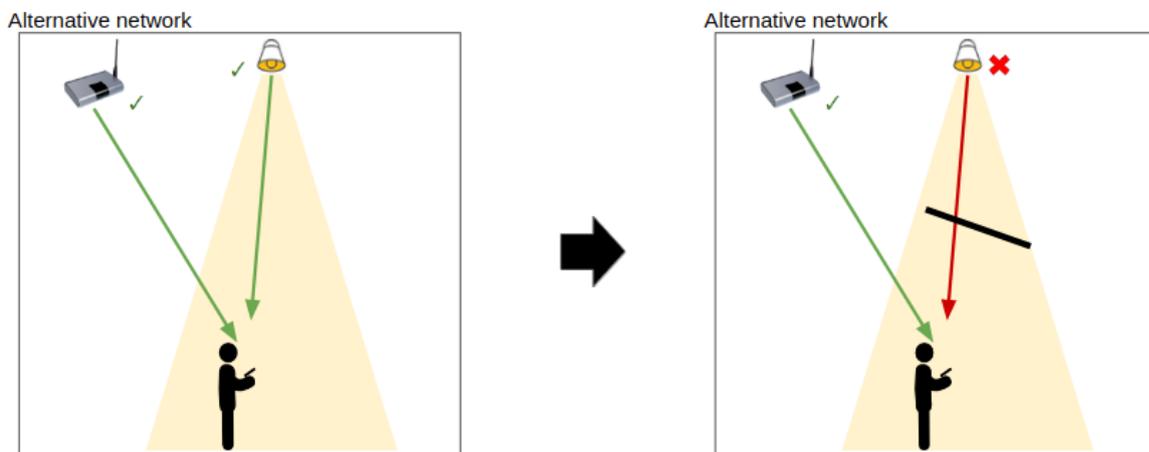


Figure 19 - Second scenario: obstacle between the user and the RRLH network

In the following graphs shown in Figure 20 , the cumulative bandwidth of both networks is in orange. The video displayed to the user is shown in blue. The single path player is on the left, and the MS-Stream player on the right.

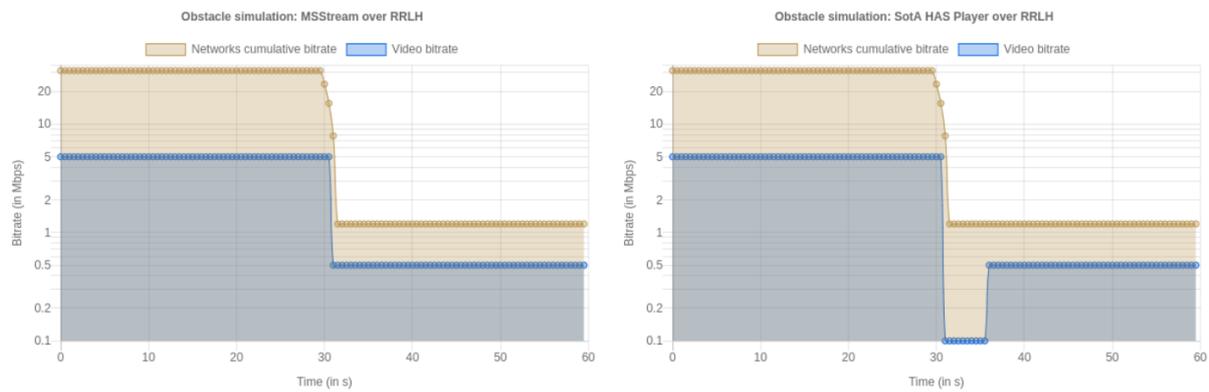


Figure 20 - Results of the second evaluation. *Single path player on the left, MS-Stream player on the right*

In this second scenario, the MS-Stream player is able to select a new quality immediately by using the data received on the second available network. However, the single path player can not adapt to this event and during a few seconds, the video is paused and nothing is delivered to the user. As a conclusion, the MS-Stream protocol is able to provide a better video quality in average in this scenario.

This evaluation has demonstrated in two scenarios the impact of MS-Stream in terms of reliability and video quality for the project. It is worth to note that the single path player is actually more advanced than the state of the art video players because none of them is by default able to switch between two networks in case of an emergency, leaving the user with an interrupted video.

3.1.3 VR Service

Within IoRL there are several VR use case applications under development. Each of which benefits from the suggested high data rates and low latency provided by the IoRL 5G network. Furthermore the IoRL project intends to explore the possibility of utilizing its distributed antenna system to provide 6DOF wireless, tracker-less, multiuser VR experiences to both PC and mobile VR.

3.1.3.1 Service interaction

The services provided range from the following:

- 360 degree live video streaming service
- Multiplayer VR gaming experiences
- 360-degree video streaming for Virtual tourism

360 live streaming

The 360 degree live video streaming service aims to provide users with the ability to live stream media data from a 360 Theta V camera to another user in a VR environment. The intention is to allow the VR user to observe the world through a remote camera.

The current approach the IoRL project has utilised is to use StreamShark, an existing Content Delivery Network (CDN) to distribute the live data over the internet. A VR user is then able to use Firefox Nightly and WebVR to display the video as a 360 VR experience. An overview of this system is demonstrated below in Figure 21.

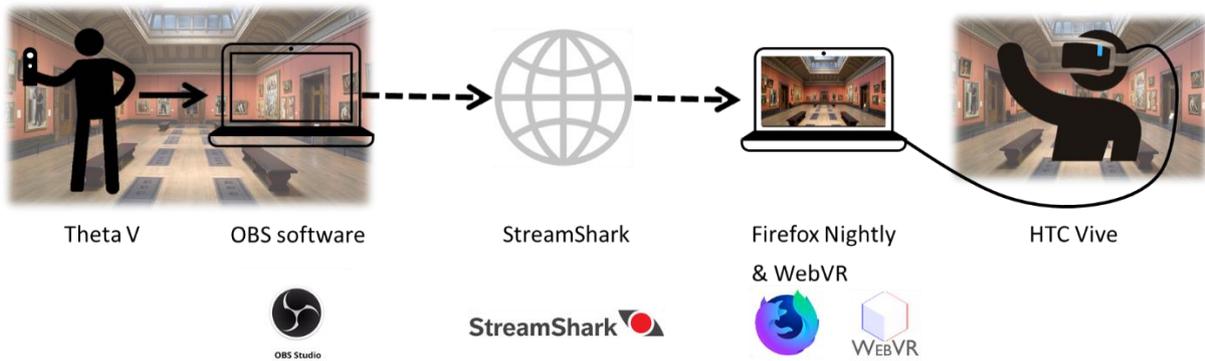


Figure 21- 360 VR Live Streaming System diagram

Multiplayer gaming

Multiplayer VR gaming experiences allow multiple users to interact together in a single environment. This uses high amounts of processing power and bandwidth to provide a shared synchronous experience.

The IoRL project has developed two experiences with the purpose to be made completely multiplayer. The first is the VR bike game, a VR experience in which users can physically ride a bike in a virtual scene. The second is a VR car game, again users can physically control a steering wheel and pedal controller to virtually explore a virtual setting.

The existing method employs a steering controller of some sort to input commands to a computer, Unity 3D software then directly outputs the relevant media data to the VR user(s). This is shown in Figure 22.

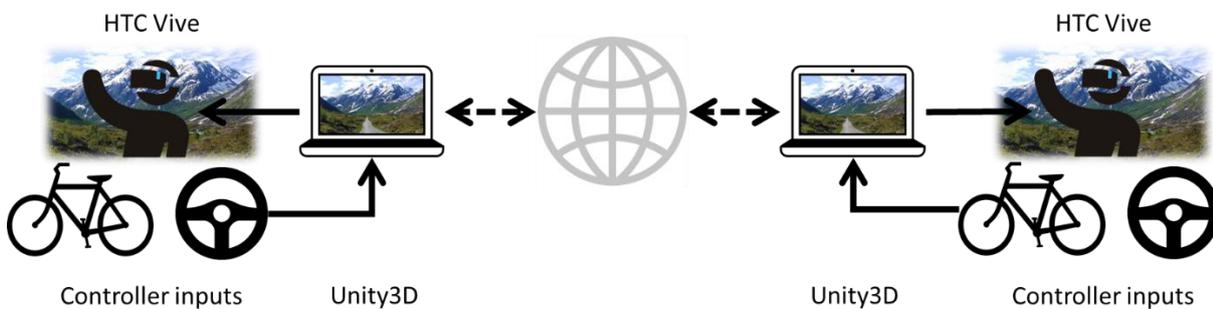


Figure 22 - Multiplayer VR System diagram

Virtual tourism

Virtual tourism involves streaming 360-degree content from either online or within the IoRL network. Both the Dystopian London and Globe Theatre experiences are currently VR games allowing the user to explore freely. The IoRL project means to store this data within a network cache for all users.

The London tour, however, is an online 360 video that will be streamed through the IoRL network to the VR Head Mounted Display (HMD).

Both systems are combined to compose a Virtual Tourism package, this is illustrated in Figure 23

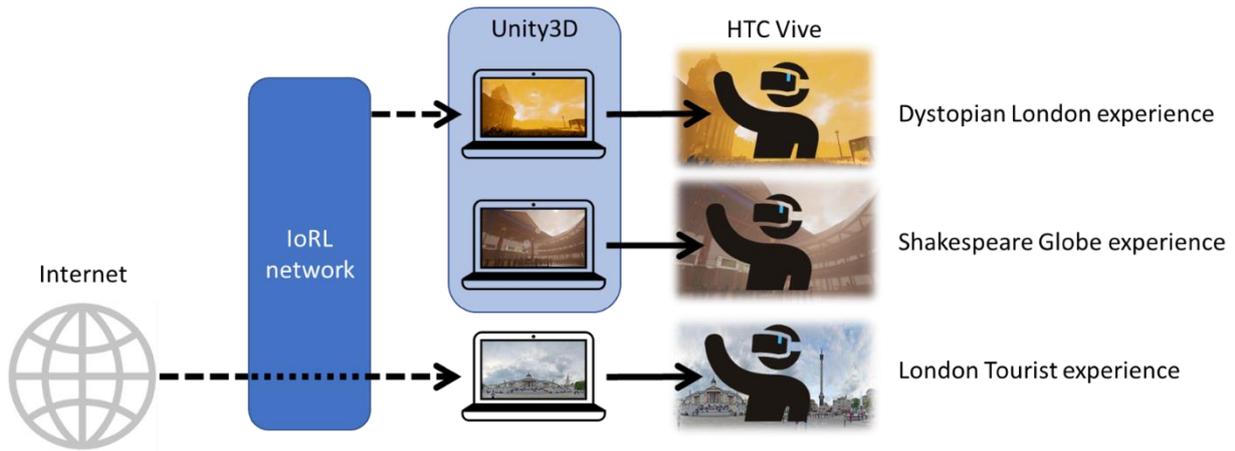


Figure 23 - VR virtual Tourism System diagram

3.1.3.2 Service integration

The VR services provided by the IoRL project will be developed into a single VR Virtual Network Function (VNF) within the IoRL network. This VR VNF will initiate when a VR user requests use of a particular IoRL VR service. Drawing location data from the Location Database (LD) which is calculated from the Location Server (LS). Depending on the VR application, location data will be requested for transmission to the relevant terminals, either to the User Equipment (UE) or the VR VNF applications. This data, along with the VR VNF data will be processed accordingly and ultimately routed through the SDN to the IoRL transport system (RRLH Controllers (RRLHC), Splitters and RRLHs) ending at the users’ VR system. A system overview is illustrated in Figure 24.

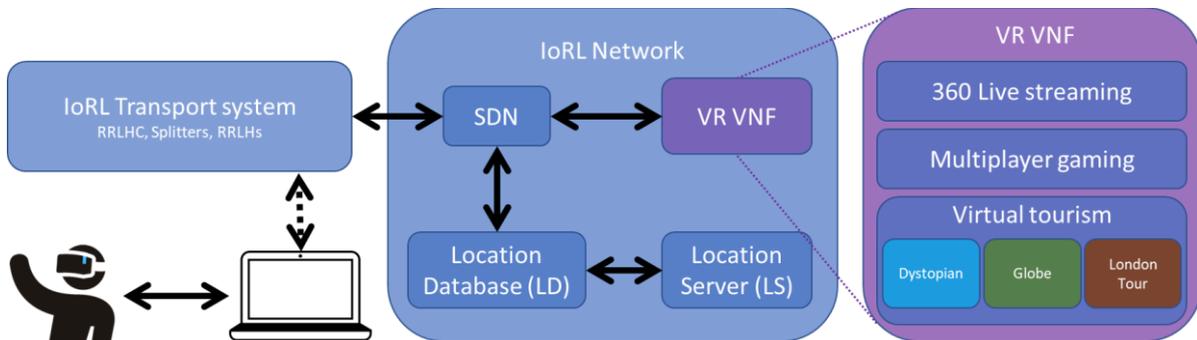


Figure 24 - VR VNF System integration

3.1.4 Security services

In the IoRL Radio Access Network (RAN) multiple remote communication techniques will be utilized, for example, VLC, mmWave and Wi-Fi, as is presented in the Figure 25 Each utilized technique introduced unique threats to the IoRL system. However, when we analyze attacks that span over one technique we observe that in most cases usable traffic must be forwarded by the Intelligent Home IP Gateway (IHIPG). Due to this fact, during initial threat analysis we decided that all IoRL security mechanism will be implemented in this part of the system and called Integrated Security Framework (ISF).

Performed in the first year of the project analysis of vulnerabilities in the various environments, for example, supermarket, train station or home, reveals multiple possible

attack types. During further works in the second year we develop, deploy and experimentally verify multiple ISF modules that could detect and in some cases SDN mitigate attacks.

Currently ISF have modules that could detect and mitigate:

- TCP port scanning detection and prevention,
- DHCP address pool exhaustion detection and prevention,
- Rogue DHCP server detection,
- MAC table overflow prevention,
- Network sniffing detection.

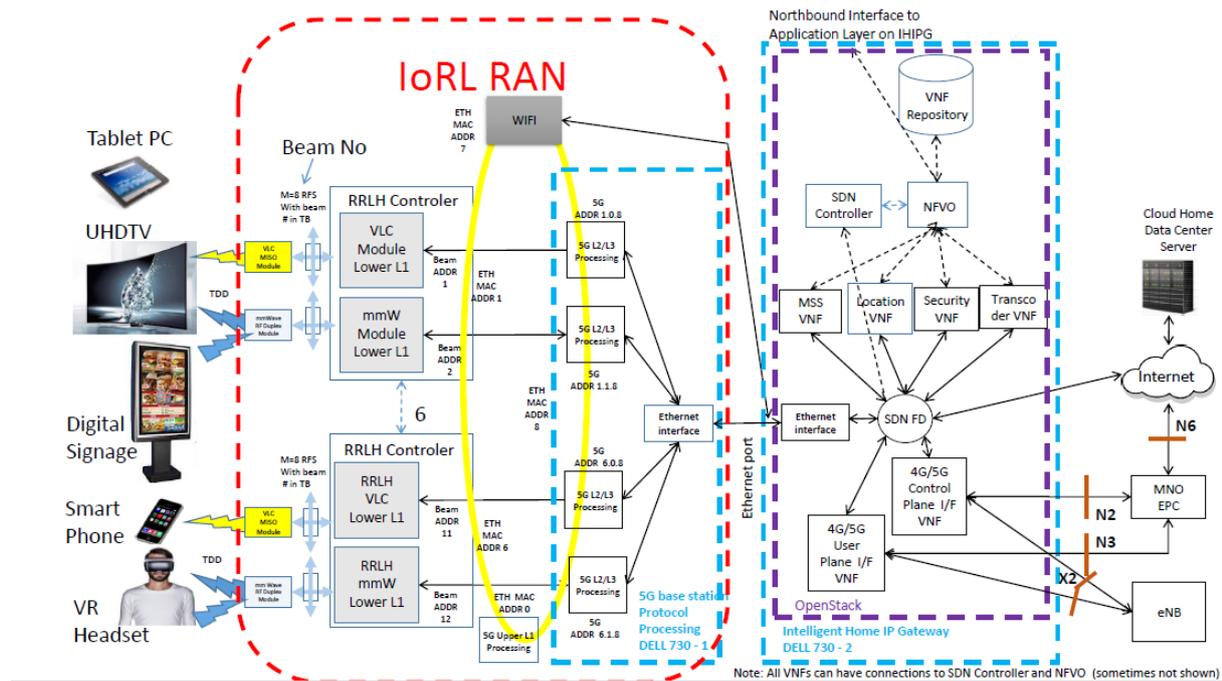


Figure 25 - IoRL RAN with multiple remote communication techniques in combination with the IoRL IHIPG with SDN, which hosts security VNF.

3.1.3.1 Service integration

All proposed security functions will be deployed at the IHIPG as security VNF, and called Integrated Security Framework (ISF). The main aim of the ISF is related to the detection of various attack types directed to or sourced from IoRL RAN, for example, Denial of Service attacks, or hostile scanning activity. The main parts of the ISF are:

- Virtual Machine containing main detection programs,
- SDN security monitoring and management application,
- Web-based security Dashboard.

To realize security VNF virtual machine, as is presented in Figure 26, introduces three interfaces:

- Interface no. 1, to web based interface which is accessed from inside the IoRL system and if needed from outside IoRL system after proper user authorizations,
- Interface no. 2, to the SDN controller, which enables possibility of SDN network reconfiguration by SDN security application,
- Interface no. 3, to configured via SDN mirror port, in which interesting packets for further analysis will be received.

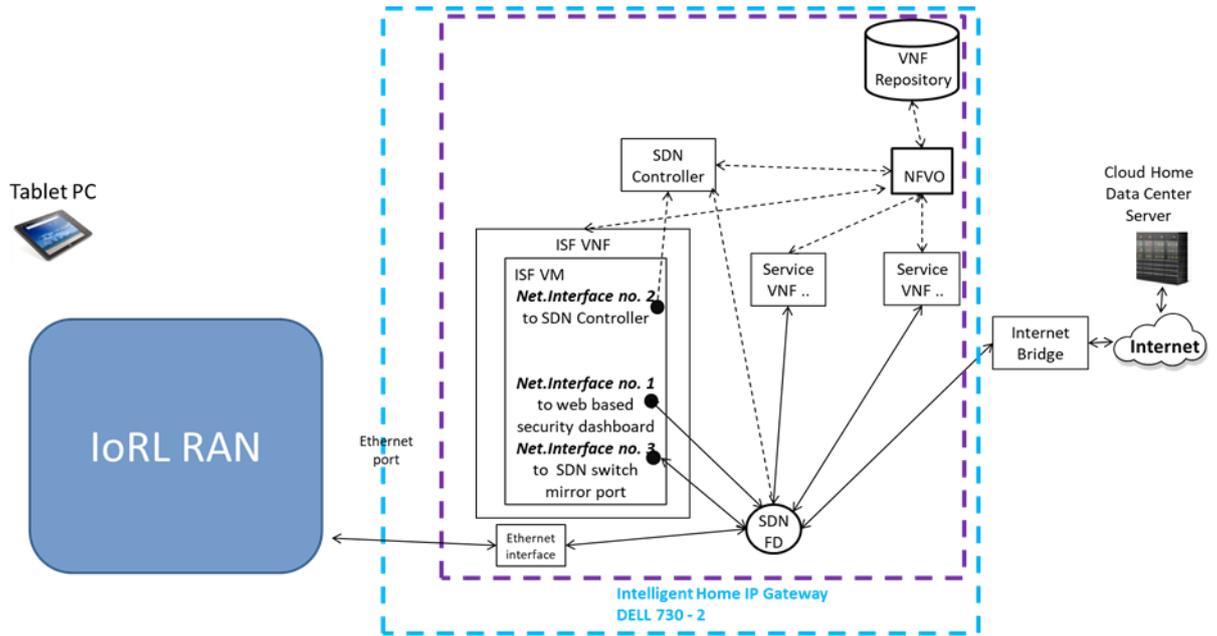


Figure 26 - presents the ISF elements deployment in the IHIPG system.

All details concerning these ISF elements were described in the deliverable 2.4 " Threats Analysis and Integrated Security Framework for the IoRL Use Cases" section 5.

Further in the following sections we will provide information concerning conducted experiments with remote detection of sniffing activity which could lead to eavesdropping attacks.

3.1.3.1.1 Implementation, Experimental Methodology & Results

We have developed and evaluated the proposed defensive mechanisms within the SDN-based environment. The following subsections incorporate details on the utilized test-beds, experimental methodology, and obtained results.

3.1.3.1.2 DNS-based and Forged MAC addresses-based solutions

The two first detection scenarios were realized in our virtualized SDN environment as seen on Figure 27. We have utilized *Open vSwitch* version 2.10.1 and *RYU* controller version 4.30. We have developed a custom *RYU* module that instructs the *Open vSwitch* to forge packets and send them to the sniffing host.

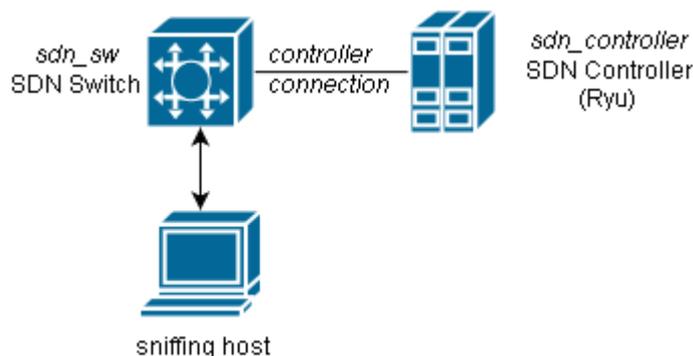


Figure 27 - Testbed for the DNS-based and forged MAC addresses approaches.

During the DNS-based experiment we sent TCP segments to the sniffing host with:

- randomized source and destination MAC addresses,
- randomized source and destination IP addresses,
- randomized source and destination TCP ports,
- randomized SEQ and ACK numbers,
- 2 null-byte payload.

Using this approach we were able to detect reverse DNS requests on the SDN switch coming from the examined machine when the *TcpDump* (with default configuration) was running which matched the randomized IP addresses we provided in the forged packets.

During the MAC-based experiment we sent a correct ICMP echo request to the sniffing machine but with a randomized destination MAC address.

However, it must be noted that we were unable to replicate a response when the machine is sniffing on the following operating systems:

- Fedora 29 Linux (with kernel version 5.0.5),
- Windows XP Professional Service Pack 3,
- Windows 7 Professional Service Pack 1,
- Windows 8.1 Pro,
- Windows 10 Education.

This suggests that this sniffing detection method is no longer effective for the relatively modern OSes.

It is also worth noting that due the nature of the two detection methods presented above, i.e., that each mechanism is either successful in sniffing detection or not and practically no parameter tuning is possible, therefore, we verified only whether they are still feasible but in this paper we do not provide any numerical results.

To summarize the DNS-based approach can be still useful in sniffing detection, however, MAC-based approach is not applicable any more.

3.1.3.1.3 The proposed artificial load-based approach

In the remainder of this subsection we present an experimental test-bed, methodology, and obtained results for the sniffer detection method that relies on inflicting artificial load on the machine under investigation and measuring its RTT times with and without the load.

3.1.3.1.4 Experimental test-bed

For this approach we have temporary resigned from the SDN and controller-generated packets due to a low maximum throughput of the solution (in our tests, around 1,000 packets per second) and instead connected the sniffing host to an another machine (called later the *probing* machine) directly via an Ethernet cable. The *probing* machine was responsible for both creating the packet flood using *macof* and issuing pings. For the purposes of this experiment we have modified the standard version of the *macof* program in such a way that it automatically performs series of 30 seconds of no activity followed by 30 seconds of flooding until the program is terminated. Additionally, a variable packet throughput during the flooding can be configured.

In this scenario we utilized three different computers acting as the sniffing machine:

- Tower PC with AMD Athlon 64 X2 Dual Core 4200+ @ 2.20GHz, 8 GB RAM and Windows 8.1 Enterprise N x64,
- ThinkPad T460p with Intel Core i7-6820HQ @ 2.70GHz, 32 GB RAM and Windows 7 Professional x64 SP1,
- ThinkPad X1 Extreme with Intel Core i7-8850H @ 2.60GHz, 16 GB RAM and Windows 10 Education x64.

Laptops were both put in the “balanced” power management mode for the experiment and tested on battery and AC power separately.

3.1.3.1.5 Experimental methodology

We have determined that in our experimental environment the *probing* machine could handle a maximum flood packet output of around 10,000 packets per second. That is why, we have used this setting for all of our experiments.

For each of three machines that we experimented with (one desktop PC and two laptops) we performed the following experiments:

1. For the whole experiment duration we executed *ping* command with the lowest possible interval (0.01 seconds) targeted at the suspicious host and record the corresponding RTTs,
2. Then for the next 60 seconds we activated the modified version of *macof* tool, however, for the first 30 seconds it sleeps (silent period where no packets are sent) and for the next 30 seconds it generates 10,000 packets per second (flood period). In both cases we note corresponding RTTs,
3. In half of the cases for both of these periods the sniffer was active and for the second half it was not active (sniffing/no sniffing). This allowed us to compare the delays with the artificially inflicted load for the case of normal user machine and for the device with NIC set to the promiscuous mode.

In overall, for all machines we obtained 72 experiment runs (each with silent and flood periods) in 36 cases the sniffer was activated and 36 for which there was no sniffing.

Next, for each silent and flood periods we calculate a mean, median and a standard deviation, separately. Then based on obtained values we determine the values of the three markers using equation:

$$\text{marker}(x) = \left| \frac{x_{\text{flood}}}{x_{\text{silent}}} - 1 \right|, \quad (1)$$

where x is the calculated mean, median or a standard deviation. Then these markers were used for the classification purposes by utilizing a simple threshold comparison with one of the three markers. To perform classification we needed to determine the suitable threshold for this purpose. In order to establish it we performed a 50:50 cross validation, where we used half of the experiments to select the most optimal value and then test it using the second half. Then we repeated the process by switching the halves. The obtained detection results are presented in the next subsection.

3.1.3.1.6 Experimental results

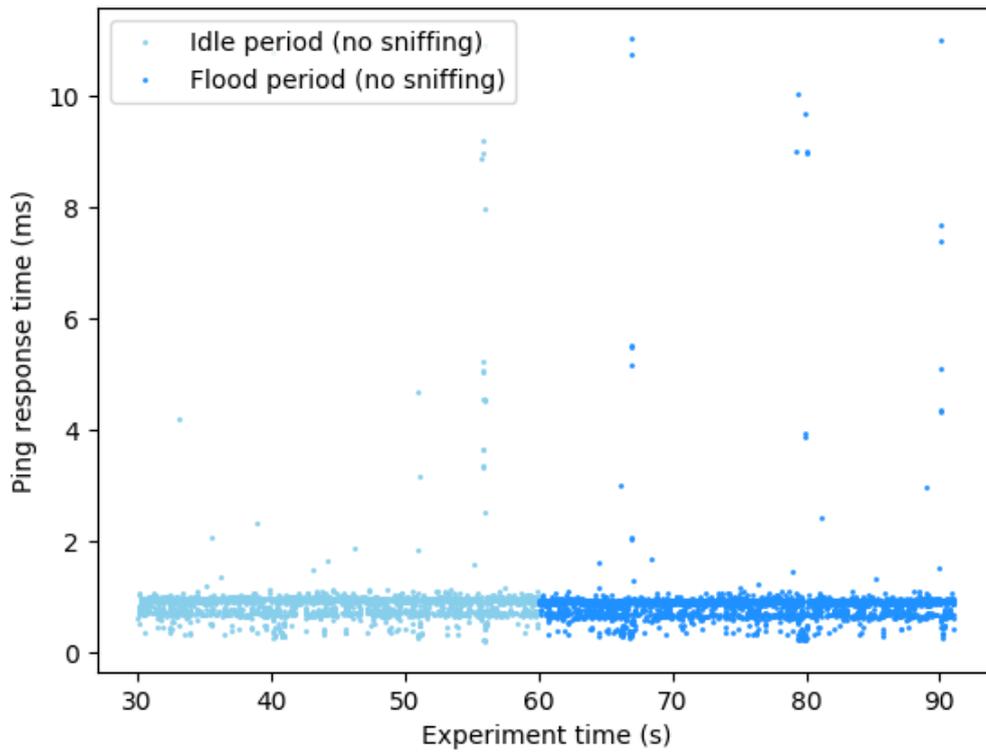


Figure 28 - Sniffing inactive: first 30 seconds -- silent period (left) and next 30 seconds -- flood period (right)

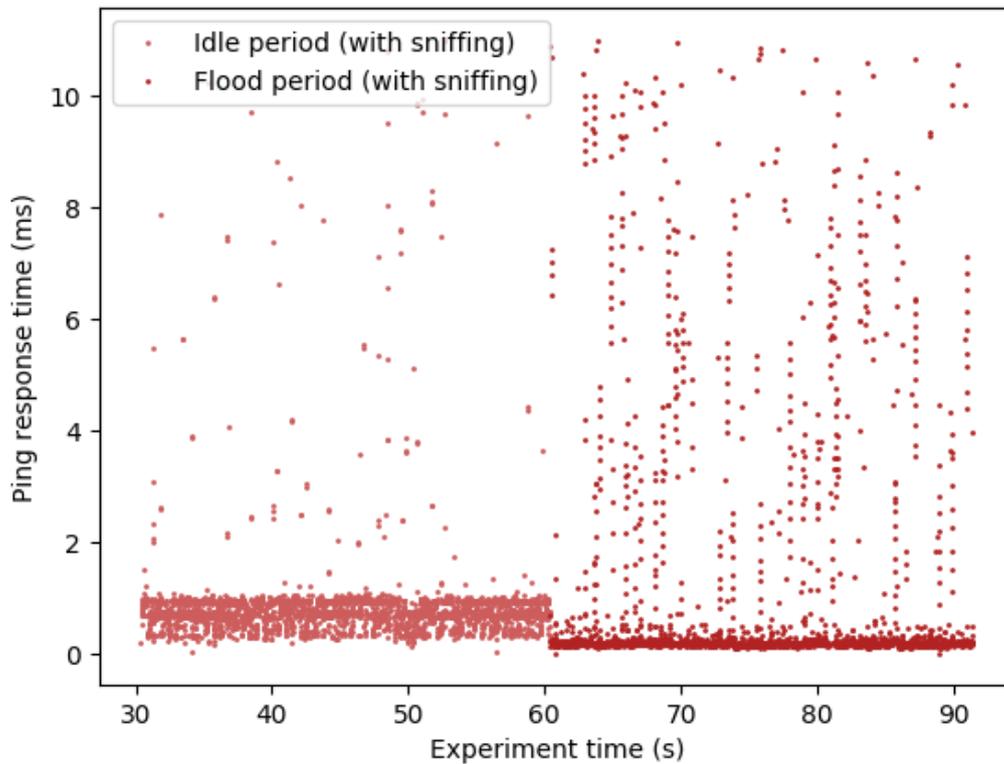


Figure 29 - Sniffing active: first 30 seconds -- silent period (left) and next 30 seconds -- flood period (right)

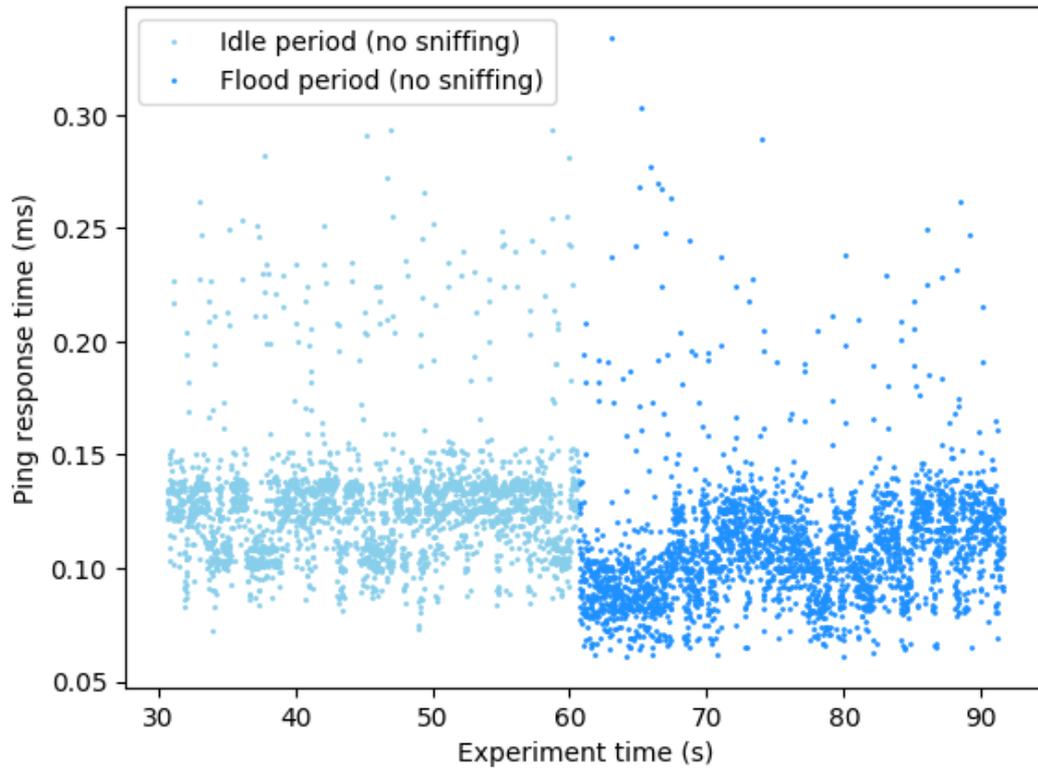


Figure 30 - Sniffing inactive: first 30 seconds -- silent period (left) and next 30 seconds -- flood period (right)

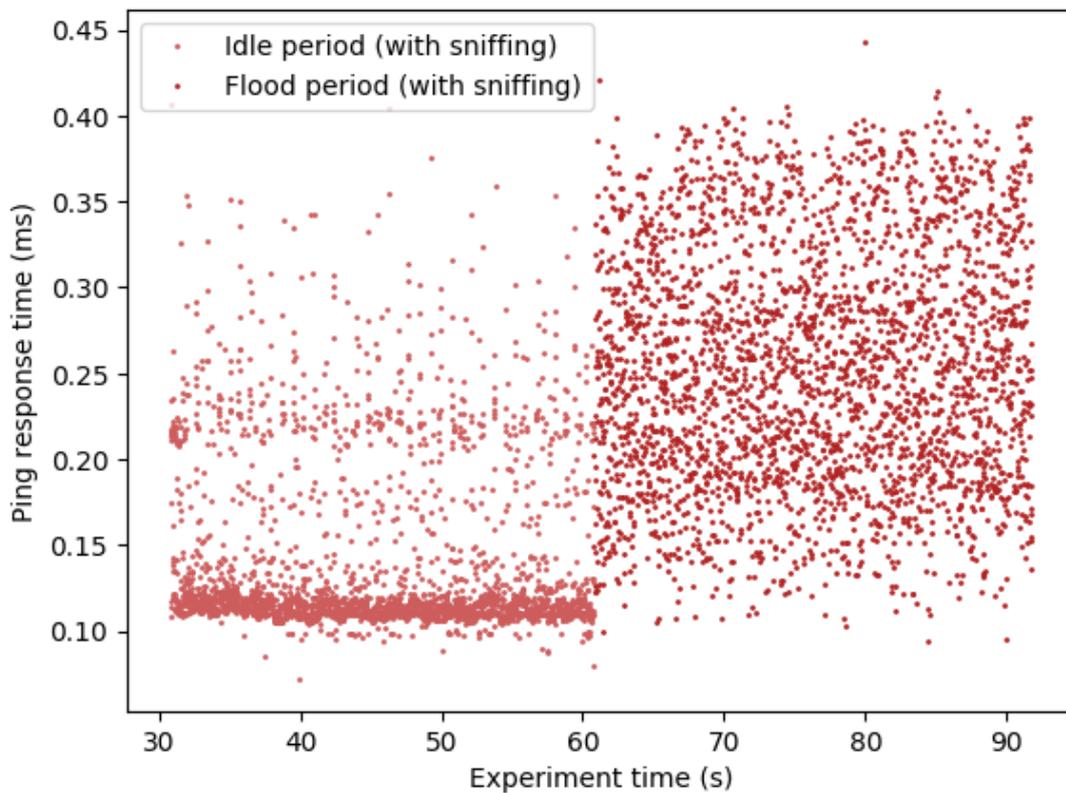


Figure 31 - Sniffing active: first 30 seconds -- silent period (left) and next 30 seconds -- flood period (right)

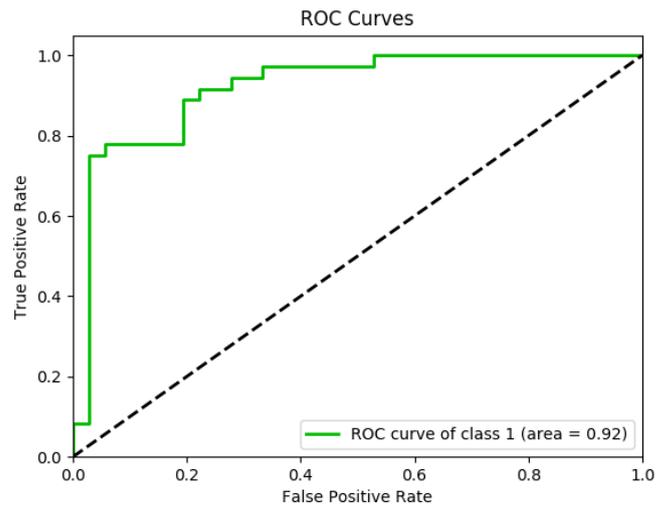


Figure 32 - Mean-based ROC

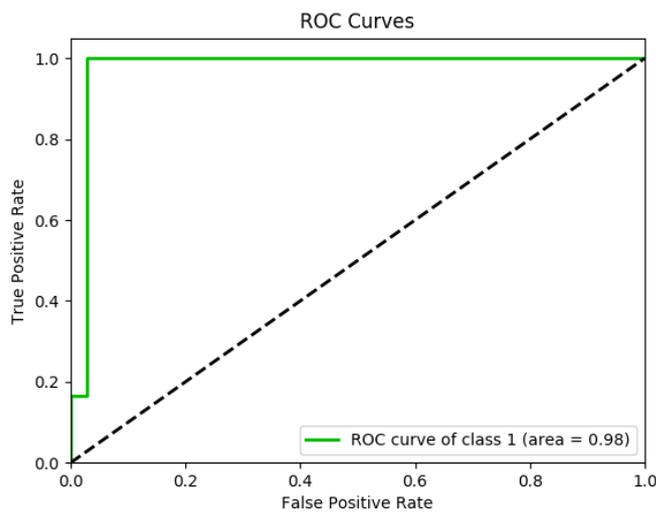


Figure 33 - Median-based ROC

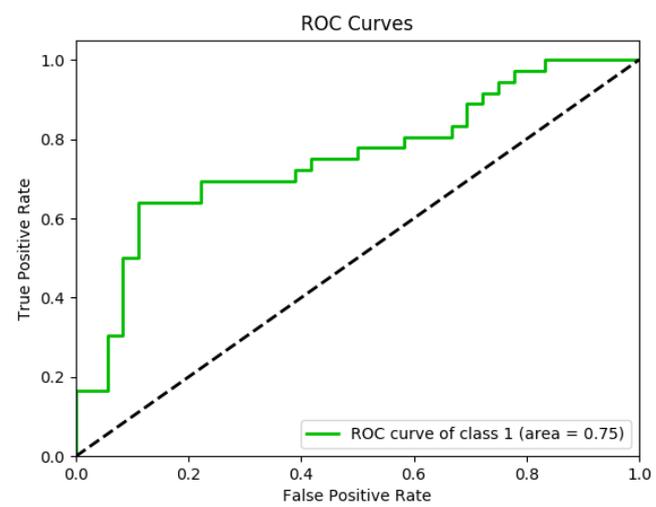


Figure 34 - Standard Deviation-based ROC

Obtained experimental results show that there is indeed no significant difference between ping response times with and without flooding on all machines when the sniffer is not active (see Figure 28 and Figure 30). On the other hand, only the outdated Tower PC behaved accordingly to the expectations when the sniffer is active, i.e., with the increased response times during flooding periods (Figure 31). Surprisingly, both laptops achieved even *shorter* response times under flooding when sniffing was enabled an example for the ThinkPad X1 Extreme is presented in Figure 29. Presumably it is caused by the intelligent overclocking technology implemented in the modern Intel CPUs installed on both laptops.

As already mentioned in previous subsection we have used three markers that rely on mean, standard deviation, and median to verify which of these metrics would be most suitable for the sniffing detection purposes. The obtained results for each metric are presented in Table 1, Table 2 & Table 3, and the corresponding ROC curves are illustrated on Figure 32 - Figure 34

From these results it is visible that the median-based approach yields the best detection performance with True Positive Rate (TPR) equal to 1, False Positive Rate (FPR) of 0.028, and AUC=0.98. Although for the mean-based technique AUC is quite high and it equals 0.92 the FPR and TPR values are not satisfying. For example, we are able to achieve quite high TPR=0.889 but at the same time FPR are too excessive, i.e, 0.194. Alternatively, for another threshold the TPR is lower 0.778 and so is the FPR value which is 0.056. Finally, the standard deviation-based detection produced the worst results with AUC of only 0.75, TPR around 0.7 and FPR of 0.222.

To summarize, for the proposed measurement-based sniffing detection method median-based approach should be utilized as it offers the best detection performance with the acceptable false positives level.

Table 1 - Mean-based detection results (AUC=0.92)

False Positive Rate (FPR)	True Positive Rate (TPR)	Threshold value
0.000	0.083	0.765
0.028	0.750	0.285
0.056	0.778	0.258
0.194	0.889	0.175
0.222	0.917	0.166
0.278	0.944	0.151
0.333	0.972	0.133

0.528	1.000	0.105
-------	-------	-------

Table 2 - Median-based detection results (AUC=0.98)

False Positive Rate (FPR)	True Positive Rate (TPR)	Threshold value
0.000	0.167	0.760
0.028	0.167	0.756
0.028	1.000	0.323

Table 3 - Standard Deviation-based detection results (AUC=0.75)

False Positive Rate (FPR)	True Positive Rate (TPR)	Threshold value
0.000	0.167	1.230
0.056	0.306	1.080
0.083	0.500	0.917
0.111	0.639	0.669
0.222	0.694	0.375
0.389	0.722	0.300
0.417	0.750	0.275
0.500	0.778	0.199
0.583	0.806	0.157

0.667	0.833	0.127
0.694	0.889	0.106
0.722	0.917	0.097
0.750	0.944	0.087
0.778	0.972	0.058
0.833	1.000	0.040

3.1.3.1.7 Conclusions

We have studied existing approaches to the sniffing detection and discovered that many of them are outdated and thus no longer effective. Based on these findings we proposed a novel solution that is able to discover machines that have NICs set to the promiscuous mode. Our approach is based on the *macof* and ping tools and it relies on inflicting artificial load on the investigated machine and measuring its RTT times with and without the load. The initial experimental prove that such an approach is effective.

This new solution together with others previously proposed that are still applicable to today's networks have been incorporated within the Integrated Security Framework which is being developed for the IoRL system.

3.2 L3-L2 interface

The Generic Routing Encapsulation (GRE) tunnel is used to deliver user data between L2/L3(RRC) and SDN directly. The IP addresses of all UEs are fixed and predefined in L3(RRC) and SDN. SDN sets the QoS Flow Identification (QFI) into the header of IP (user) packet and send IP packet to L2/L3(RRC) via GRE tunnel. When L2/L3(RRC) receives IP packet from SDN, it will obtain the QFI from DSCP(IPv4) or Traffic Class (IPv6) and UE IP address from the destination IP address in the header. According the QFI and a predefined QoS mapping table defined in L2/L3(RRC), L2/L3(RRC) should allocate the corresponding slots resources and select mmWave or VLC mode for L1 transmission.

3.2.1 Interface protocols

The user IP packet is carried over GRE tunnel, the protocol stack is shown in Figure 35.



Figure 35 - L2/L3(RRC) and SDN Protocol Stack

3.2.2 Control plane

3.2.2.1 The QFI Field Definition in IP (user) Header

The QFI can be exchanged between the L2/L3(RRC) and SDN as a part of the packet network layer header, namely Type of Service (TOS) for IPv4 or Traffic Class for IPv6. As can be seen in Figure 36 TOS give us 8 bits to be used for QFI, two bits currently unused, which leave us with 6 bits to specify the QFI.

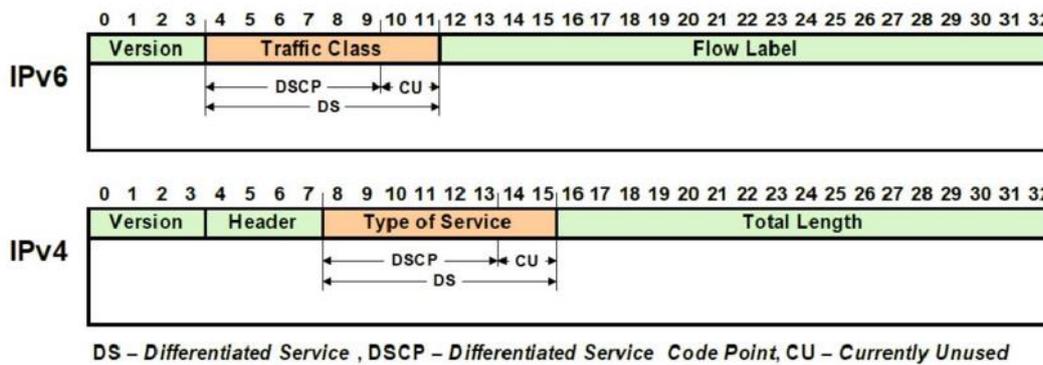


Figure 36 - Header of IPv4 and IPv6

In Figure 37, the first 2 bits LSB are ignored, then using the other 6 bits to specify QFI:

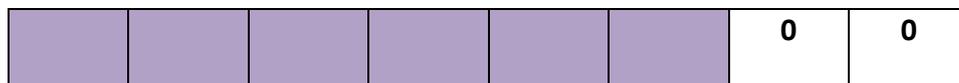


Figure 37 - QoS Bits in Differentiated Service of IP Header

3.2.2.2 QoS Requirement Mapping Table

The standardization 5QI to QoS characteristic mapping from TS 23.501 table 5.4.7-1 [69] is selected for the QoS requirement. The table is shown in Table 4.

Table 5.7.4-1: Standardized 5QI to QoS characteristics mapping

5QI Value & QFI	Resource Type	Priority Level	Packet Delay Budget	Packet Error Rate	Example Services
1	GBR	20	100 ms	10^{-2}	Conversational Voice
2		40	150 ms	10^{-3}	Conversational Video (Live Streaming)
3		30	50 ms	10^{-3}	Real Time Gaming, V2X messages
4		50	300 ms	10^{-6}	Non-Conversational Video (Buffered Streaming)
65		7	75 ms	10^{-2}	Mission Critical user plane Push To Talk voice (e.g., MCPTT)
66		20	100 ms	10^{-2}	Non-Mission-Critical user plane Push To Talk voice
75		25	50 ms	10^{-2}	V2X messages
5		Non-GBR	10	100 ms	10^{-6}
6	60		300 ms	10^{-6}	Video (Buffered Streaming) TCP-based (e.g., www, e-mail, chat, ftp, p2p file sharing, progressive video, etc.)
7	70		100 ms	10^{-3}	Voice, Video (Live Streaming) Interactive Gaming
8	80		300 ms	10^{-6}	Video (Buffered Streaming) TCP-based (e.g., www, e-mail, chat, ftp, p2p file
9	90				sharing, progressive video, etc.)
69	5		60 ms	10^{-6}	Mission Critical delay sensitive signalling (e.g., MC-PTT signalling)
70	55		200 ms	10^{-6}	Mission Critical Data (e.g. example services are the same as QCI 6/8/9)
79	65		50 ms	10^{-2}	V2X messages

Table 4 - QoS Mapping Table

Because we only have 6 bits for QFI, to express all QFI in this table, we need a conversion table for mapping TOS/TC field bits to real QFI. The conversion table is shown in Table 5

Table 5 - TOS/TC Field Mapping Table

Bits of TOS/TC field	QFI
00000100(0x04)	1
00001000(0x08)	2
00001100(0x0C)	3
00010000(0x10)	4
00010100(0x14)	5
00011000(0x18)	6

00011100(0x1C)	7
00100000(0x20)	8
00100100(0x24)	9
00101000(0x28)	65
00101100(0x2C)	66
00110000(0x30)	69
00110100(0x34)	70
00111000(0x38)	75
00111100(0x3C)	79

3.2.1 Data plane

The user data, which is carried in data field of the user IP packet, will be sent together with its control data to L2/L3(RRC).

3.3 IoRL as small cell deployed within Mobile Network Operator (MNO)

This section presents IoRL as can be seen by MNOs, it will include short introduction for the system and the present the available options that comply with 3GPP architecture designs.

3.3.1 Visible Light Communication-based gNB

Visible Light Communication-based gNB (VLC-gNB) is a 5G small cell solution for indoor environments, as shown in Figure 38, it consists of two main subsystems linked together, the radio access network subsystem and the networking and services subsystem.

The radio access network subsystem consists of mmWave and VLC modules which are utilizing 60 GHz unlicensed or 40 GHz licensed bands and visible light communication to release the radio resources for the indoor environments. These technologies enabled the VLC-gNB to provide Gbps data rate and sub-meter location accuracy indoors [12].

The networking and services subsystem consists of the Intelligent Home IP Gateway (IHIPGW). It offers intelligent management, flexible deployment, and add-on services for the VLC-gNB. The intelligence and flexibility are offered by use of SDN and Virtualised Networking Functions (VNF) technologies, which enable the system to deploy UE's location server with sub-meter accuracy, which in-turn supports the deployment of add-on services such as smart TV services [12] location based data access services [13].

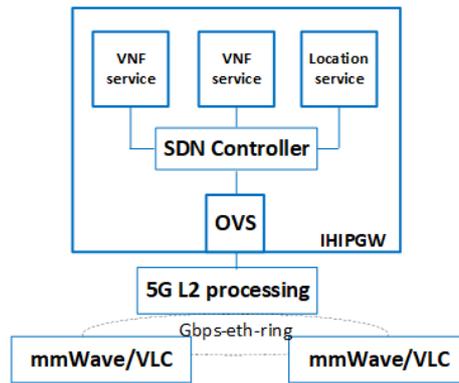


Figure 38 - VLC-gNB small cell

The VLC-gNB provides an intelligent solution for different indoor environments such as, home, museum, supermarket and tunnel stations...etc. [14]. It provides better QoS for UEs and offers local internet breakout, to reduce backhaul traffic, latency and improve user experience [13]. The next step for the VLC-gNB is to be deployed as a part of MNO RAN. However, the integration of the VLC-gNB with RAN should be considered carefully in order to provide a solution that does not downgrade the benefits gained during operation in the standalone environments. There are multiple possible deployments of the VLC-gNB indoor small cells as shown below.

3.3.1.1 Conventional topology All – Connected (AC) deployment:

Each VLC-gNB small cell visible and connected back to the core network.

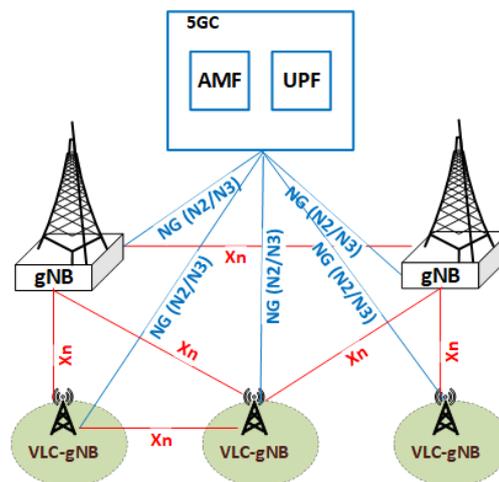


Figure 39 - AC- VLC-gNBs topology

In AC-VLC-gNB deployment, each VLC-gNB small cell is visible to the core network as shown in Figure 39, the UE traffic is traversed back to the core, without the involvement of the outdoor gNB. The VLC-gNB small cells use NG interface (N2/N3) to connect with 5G core while using Xn interface to connect to all other VLC-gNBs and gNBs. Adopting AC-VLC-gNB deployment makes the cost and the handover signalling relatively high, while enables higher flexibility and lower latency in comparison to the other possible deployments.

3.3.1.2 Dual Connectivity (DC) deployment:

DC supports Stand Alone (SA) and Non-Satnd Alone (NSA) deployments, the latter deployment is considered to enable gradual transition to 5G network by enabling indoor gNB small cell to work with LTE outdoor eNB.

3.3.1.2.1 gNB and VLC-gNB DC:

UE is connected to outdoor gNB acting as a Master Node (MN) and one VLC-gNB small cell acting as a Secondary Node (SN), as shown in Figure 40. The MN is connected to the 5G core via NG interface and to the SNs via Xn interface.

3.3.1.2.2 eNB and VLC-gNB DC:

UE is connected to outdoor eNB acting as a Master Node (MN) and one VLC-gNB small cell acting as a Secondary Node (SN). The MN is connected to the Evolved Packet Core (EPC) via S1 interface and to the SN gNB via the X2 interface. The SN gNB might also be connected to the EPC via the S1-U interface and other SN gNBs via the X2-U interface.

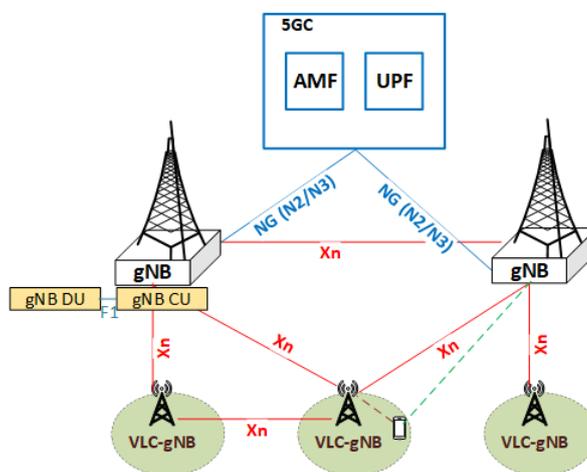


Figure 40 – Dual Connectivity topology

Adopting DC deployment makes the cost relatively high, while enabling more flexibility, lower latency and handover signalling in comparison to the other possible deployments.

3.3.1.3 VLC-gNB as Distributed Unit (DU) deployment

In DU-VLC-gNB deployment, each VLC-gNB has only Radio Link Control RLC layer, MAC layer and Physical layer at each DU, while the Centralized Unit (CU) for a group of the VLC-gNB DUs are kept as a VNF at the gNB, named Virtual Gateway (V-GW). As shown in Figure 41, V-GW connects to VLC-gNB DUs using F1 interface. gNB uses NG interface to connect to 5GC and Xn interface to connect to the other gNBs. V-GW is implemented as a Virtualised Network Function (VNF) and resides within gNB to optimize the signalling and the operation of the VLC-gNB DUs by providing one point of interaction with gNB to all connected VLC-gNB DUs. Also it enables the VLC-gNBs to provide intelligent services since it utilizes Network Function Virtualisation (NFV) technology to offer virtualised network entities such as V-proxy/cache servers.

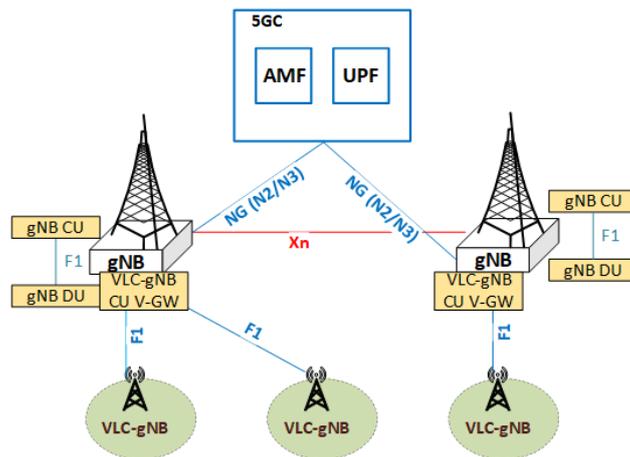


Figure 41 - VLC-gNB Distributed Units topology

Adopting DU-VLC-gNB deployment makes the cost, flexibility and the handover signalling relatively low, while making the latency relatively higher.

3.4 MmWave Localization

The Location Server design covers a short system overview and a list of functions implemented for TDoA based location estimation together with supporting functions required to test the location estimation functions.

The description of the planning tool covers summary about the optimization concept based on the particle swarm optimization, a simplified example illustrating the optimization steps and a more realistic example showing the strength of the planning tool..

3.4.1 Algorithms for Location server

Location server is a VNF implemented at intelligent HIPG to estimate location coordinates of all connected UEs in the IoRL network. RRLHC and UEs measure location relevant parameters that are used by location server in location estimation process.

In the case of mmWave localization, the measured parameters are TDoAs that are estimated by RRLH controller based on ToA measurements performed by at least 4 RRLHs. The 4 RRLHs are the minimum number of sensors to make 3D localization available. RRLHs will be synchronized in a switched architecture which was proposed in D5.1 and which is illustrated in Figure 42.

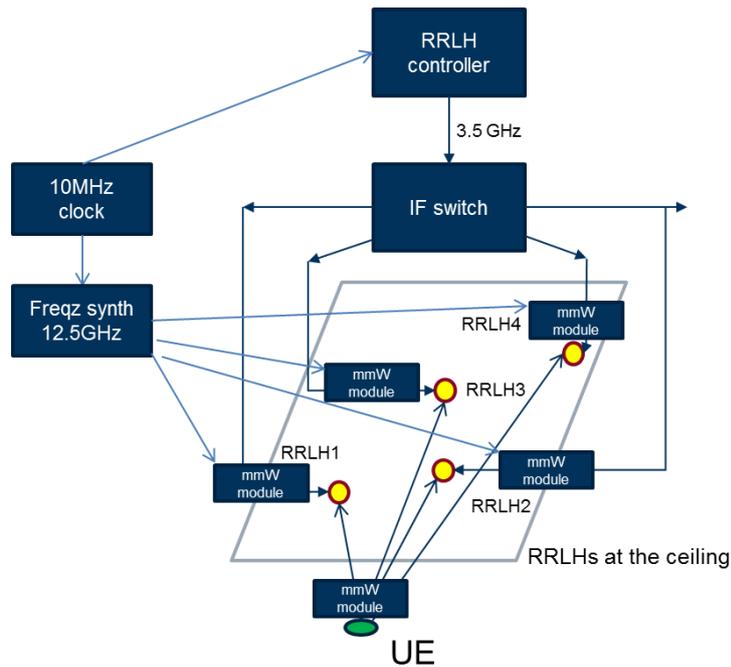
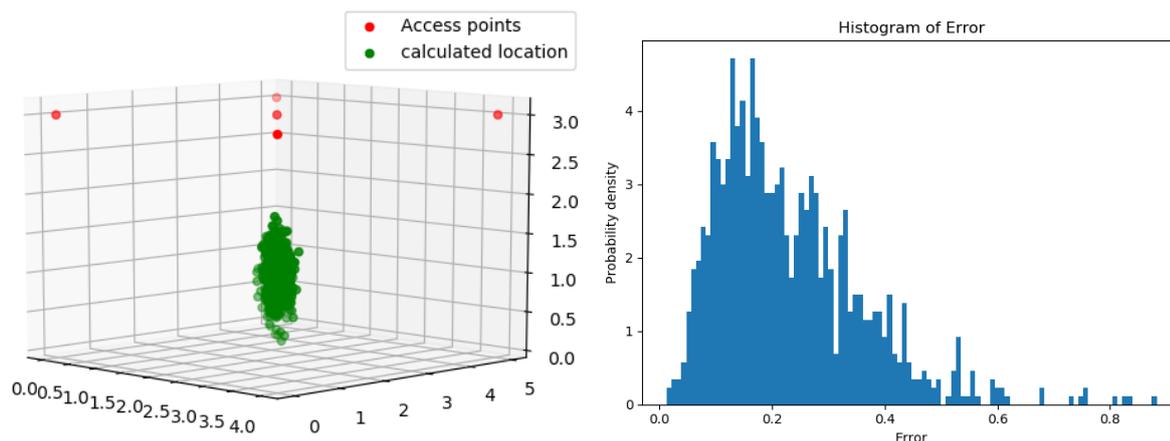


Figure 42 - Switched architecture for the TDoA estimation using mmWave

The following functions were implemented in a Python library by FhG:

- `localizationlb.TDOA_Is(RRLH_x,y,z,TDOA,Niter)`
 - a function for positioning using LS algorithm
- `localizationlb.TDOA_ts(RRLH_x,y,z,TDOA,Init,Niter,flag)`
 - a function for positioning using Taylor series based algorithm
 - it is an iterative algorithm
- `localizationlb.TDOA(RRLH_x, y, z, UE)` computes TDOAs
 - a function for simulations, it computes TDOAs for a given scenario and can be used for evaluation of positioning algorithms

An example of such a simulation is given in Figure 43. One UE was estimated in a room (4m x 5m x 3m) by 5 RRLHs that were distributed at the ceiling (red dots). The position estimates Figure 43(a) were simulated for TDOA estimates that were corrupted by AWGN with 10cm standard deviation. This resulted in the positioning error with histogram that is depicted in Figure 43(b).



a) Position Estimates corrupted by AWGN

b) Position Error Histogram

Figure 43 - Example of mmWave positioning of an UE by 5 RRLHs delivering 10cm

3.4.2 Tool for the RRLH distribution planning with respect to the localization performance

A tool for planning the optimum geometrical distribution of RRLHs was elaborated by FhG. The planning tool was implemented in Python in order to avoid licensing cost connected with Matlab and the usage of special optimization Matlab packages.

The aim of this planning tool is to propose an (sub-)optimum constellation of RRLHs with respect to its coverage and the localization precision. Thus, the result of the planning is a set of two dimensional coordinates of all RRLHs that are used to localize UEs within a predetermined area. The RRLHs are assumed to be situated within one plane at the ceiling. This plane can also have irregular shape defined by a polygon. In order to evaluate performance of a certain constellation it is necessary to determine the precision of such RRLH constellation.

One possible way to determine the localization precision of a certain RRLH constellation is to use Monte Carlo simulation. Realistic TDOAs are simulated by ideal TDOAs that are corrupted by AWGN of a predefined standard deviation. The noisy TDOAs result in a set of noisy location estimates of an UE situated at a specific position. These location estimates create a scatter plot with a shape similar to an ellipsoid as shown in Figure 43. Thus the result of such Monte Carlo simulation can be represented by three values of standard deviations related to the axes of the 3D ellipsoid. These standard deviations $(\lambda_1, \lambda_2, \lambda_3)$ determine the precision of the localization using certain RRLH constellation for a specific UE position and TDOA noise level σ_{TDOA} . In literature, these three values are usually combined into one which is referred to as the geometric dilution of precision (GDOP)

$$GDOP = \frac{\sqrt{\lambda_1^2 + \lambda_2^2 + \lambda_3^2}}{\sigma_{TDOA}} \quad (2)$$

Since the localization precision also depends on the UE position, it is necessary to determine GDOP for different UE positions. Thus, GDOP is basically 3D function of UE coordinates. GDOP

creates a heat map showing how good the localization is at different positions of UE within the targeted area. Examples of a two-dimensional heat maps, with fixed height of the UE, are shown in Figure 44 for two different RRLH constellations. The targeted area was a room with dimensions 3m x 3m x 3m. And only a 2D GDOP was computed for a height of the target at 1m. It is evident how the localization precision strongly depends on the antenna constellation and the UE position as well. The standard deviation of TDOA estimates was assumed to be 50cm in this example was.

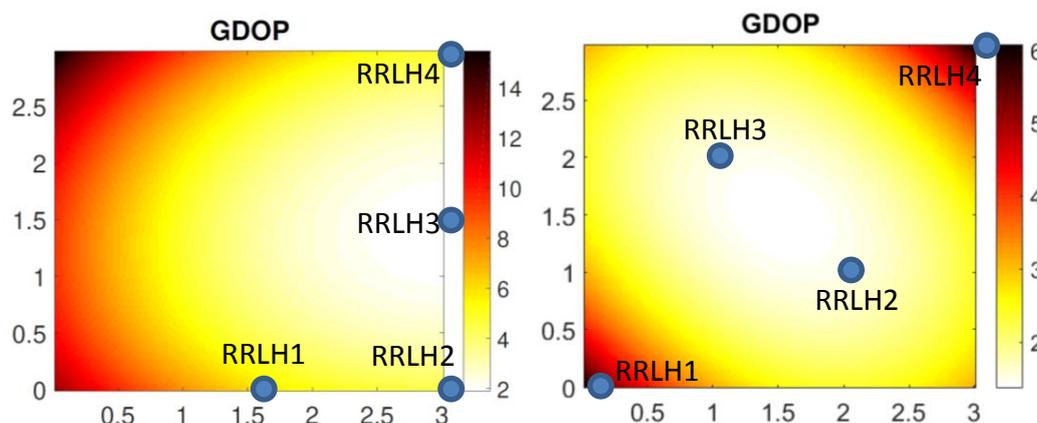


Figure 44 - Examples of GDOPs for two different RRLH constellations

In order to find out a (sub-)optimum RRLH constellation it is necessary to solve an optimization problem in which a relatively large number of unknowns (twice the number of RRLHs) depends only on small number of criterions (e.g. one GDOP value for a certain RRLH constellation). A usual way to solve such a problem is to optimize a cost function. Since the cost function is highly nonlinear the traditional gradient based optimization approaches will most probably not be suitable candidates. They will tend to converge to one of numerous local minima. Optimization procedure that found inspiration in the biology such as of Particle Swarm Optimization (PSO) will most probably be more appropriate candidate. PSO is a method that optimizes a problem by iteratively trying to improve a candidate solution by moving particles of a population around in the search-space according to simple mathematical equations using the notion of the particle position and its velocity. PSO was originally intended for simulating movement of organisms such as birds in a flock or bees in a swarm [15].

In the PSO algorithm, a swarm contains a set of particles \mathbf{p}_k^i ($i = 1$ to number of particles) that evolve with time k . The swarm moves with time (iterations) towards the minimum of a cost function. In our case, the cost f_k^i of the i^{th} particle at time k is given by single GDOP value that depends on the RRLH coordinates. In order to compute a single GDOP value describing the localization and eventually the coverage performance of certain RRLH constellation it is necessary to estimate 3D GDOP values and to compress it into one value by e.g. taking a mean value. However, this is time consuming procedure if GDOP is determined by means of Monte Carlo simulations. Fortunately, GDOP can be estimated analytically [16] which drastically reduces the computational complexity. This allows GDOP to be used as the cost function to optimize the geometrical distribution of RRLHs.

Particles, i.e. vectors of different RRLH coordinates, evolve with iterations according to

$$\mathbf{p}_{k+1}^i = \mathbf{p}_k^i + \mathbf{v}_{k+1}^i \Delta t, \quad (3)$$

where Δt is the time increment between two iterations and \mathbf{v}_{k+1}^i is the i^{th} particle velocity at time $k+1$ which is computed according to the following update formula

$$\mathbf{v}_{k+1}^i = c_1 \mathbf{v}_k^i + c_2 \mathcal{U}(0,1) \frac{(\mathbf{p}^i - \mathbf{p}_k^i)}{\Delta t} + c_3 \mathcal{U}(0,1) \frac{(\mathbf{p} - \mathbf{p}_k^i)}{\Delta t}, \quad (4)$$

where \mathbf{p}^i is the best position of the i^{th} particle according to its cost values f_k^i over all previous iterations, \mathbf{p} is the best global particle of the whole swarm, $\mathcal{U}(0,1)$ is a random variable uniformly distributed between 0 and 1. The coefficient c_1 is the inertia factor which controls the motion influence from the previous iteration. The coefficient c_2 is the self-confidence factor which controls the motion according to particle memories (the best of each particle in its evolution). The coefficient c_3 is the swarm-confidence factor which controls the motion according to the best particle in the swarm. Explanation of the particle position and velocity update formula is shown in Figure 45. The new velocity \mathbf{v}_{k+1}^i contains a part of the previous velocity and random parts of the motion towards the best i^{th} particle and motion towards the best particle within the swarm.

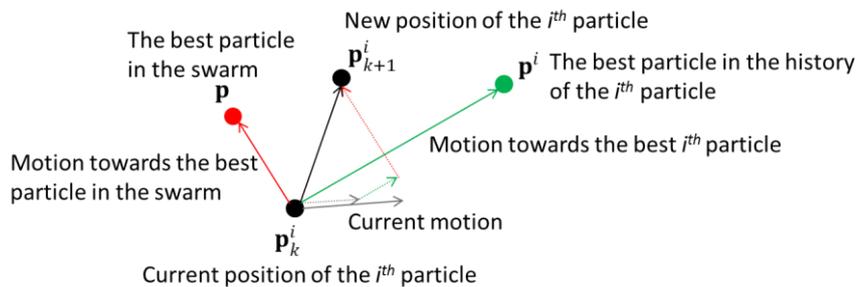


Figure 45 - Updates of particle positions and velocities

The following simulation visualizes the basic principles of PSO based optimization of the RRLH constellation. The optimization scenario is shown in Figure 46. Two RRLHs are situated at a ceiling of a 10m x 5m x 2m room. The RRLHs can be distributed only along one line in order to allow simple visualization of the optimization process with only 2-dimensional search space. The swarm contained 300 particles each of them was represented by a vector with X-coordinates of both RRLHs. The swarm evolved over 200 iterations looking for a global optimum according to the value of a GDOP based cost function. The evolution of 40 particles during the first 3 iterations is depicted in Figure 47. It is obvious that the particles are randomly scattered within the 2-dimensional search space.

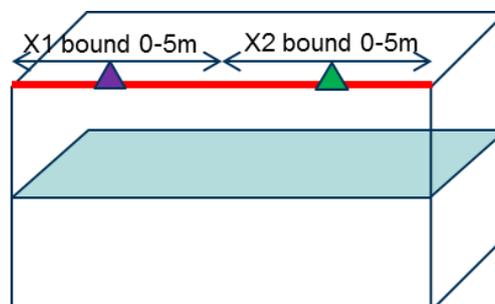


Figure 46 - Optimization scenario with 2 RRLH's

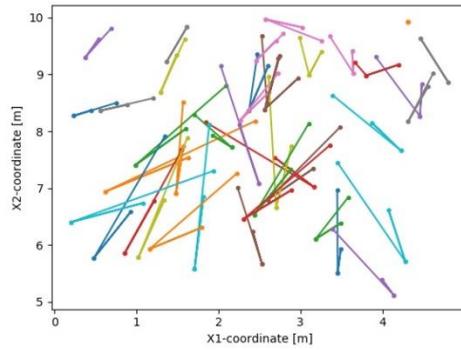


Figure 47 - Distribution of 40 particles during the first 3 iterations

Figure 48 illustrates the evolution of X-coordinates of the 40 particles. At the beginning the X-coordinates reach values randomly between their predefined bounds – 0m and 5m. After certain number of iterations, their values converge to the global optimum. Figure 49 shows the last 10 iterations of the 40 particles. All particles are scattered in a vicinity of the global optimum that was estimated to be $[x_1=0.94m, x_2=8.94m]$. This defines the optimum RRLH constellation with RRLH1 at $[0.94m, 0m]$ and RRLH2 at $[8.94m, 0m]$.

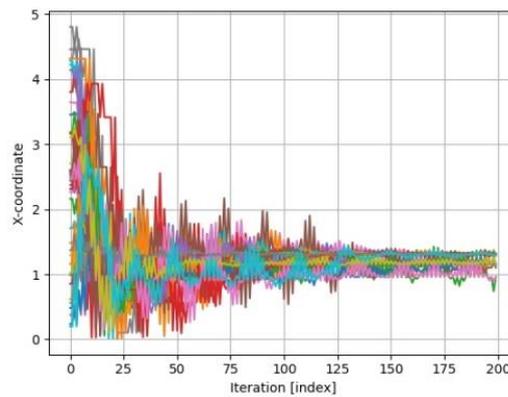


Figure 48 - Evolution of X1-coordinate

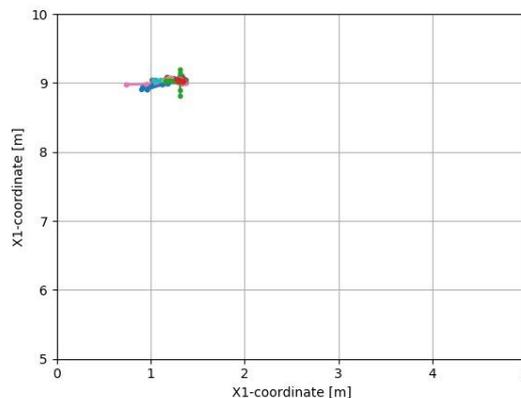


Figure 49 - Distribution of 40 particles during the last 10 iterations

Figure 50 and Figure 51 illustrate the benefit of the planning tool. Figure 50 shows the GDOP distribution of an intuitively selected RRLH constellation within which the RRLHs are situated in a way to spread the largest aperture of the two element “antenna array”. Figure 51 depicts the result of the optimizer. The area in between the RRLHs that featured the worse

positioning precision in the case of the intuitive approach (about 3.7m) reaches now about 60cm better localization precision.

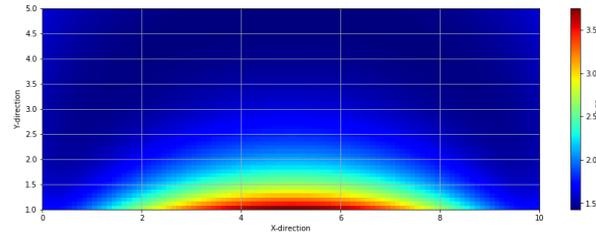


Figure 50 - GDOP for Rx1 at [0,0] and Rx2 at [10,0] Intuitively the best placement

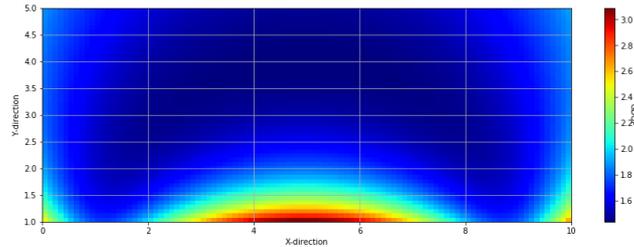


Figure 51 - GDOP for the optimization result with Rx1 at [0.95,0] and Rx2 at [8.94,0]

The example discussed so far was only used to illustrate the basic functionality of the RRLH distribution planning tool. More realistic example of the optimization tool performance is given in Figure 52 and Figure 53. Here, the room where UEs shall be localized has a more complex polygonal shape. The task of the optimization tool was to find out the optimum RRLH geometry with respect only on the localization precision and without any respect on the coverage. Figure 52 shows the optimum RRLH geometry for 5 RRLHs and Figure 53 illustrates the achieved precision for this geometry. The white areas show also the places in which there is no coverage of the localization system. At these positions the UE cannot be localized in 3D since some of LOS are blocked by the walls of the room.

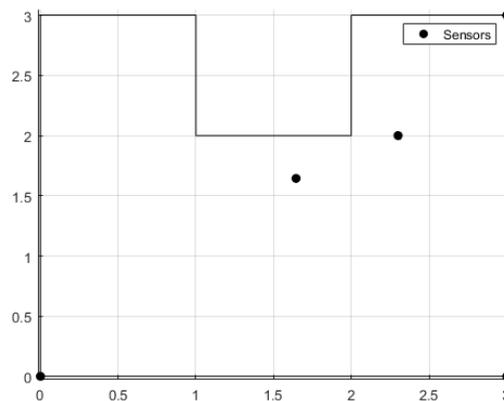


Figure 52 - GDOP for Rx1 at [0,0] and Rx2 at [10,0] Intuitively the best placement

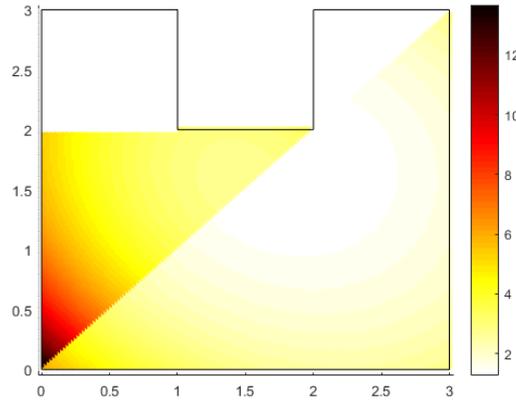


Figure 53 - GDOP for the optimization result with Rx1 at [0.95,0] and Rx2 at [8.94,0]

Figure 54 and Figure 55 illustrate results of the optimization tool when the coverage aspect was also taken into account and influenced the cost function of the PSO optimizer. It is obvious that there is a trade-off between the localization precision and the system coverage. While improving the coverage, the percentage of the covered area, the localization precision is slightly decreased (compare Figure 54 and Figure 55). This trade-off can be controlled by a weighing parameter within the cost function that is balancing the two criteria – the coverage and the precision.

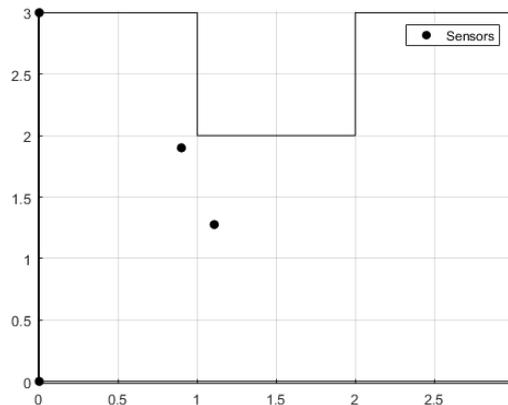


Figure 54 - GDOP for Rx1 at [0,0] and Rx2 at [10,0] Intuitively the best placement

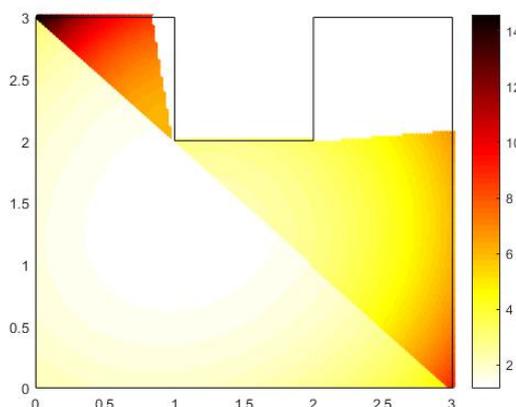


Figure 55 - GDOP for the optimization result with Rx1 at [0.95,0] and Rx2 at [8.94,0]

The strategy of the described RRLH geometry planning tool can be also used for looking for optimum RLLH constellations with respect to other criteria than the localization precision. Optimizations with respect to capacity and the coverage of the communication system is one of its possible extensions

4 The Load Balancing Process at the IoRL SDN/NFV Home Environment

To meet the future demand for mobile data traffic and network reliability, the IoRL system exploits the visible light communication (VLC) and millimetre wave (mmWave) spectrum. However, typical VLC and mmWave access points (APs) cover relatively small

areas, of approximately 2-3 m diameter [17]. In contrast, with the conventional radio-frequency (RF) systems, like wireless fidelity (WiFi) routers that operate on the traditional sub-6 GHz band, can transmit up to 40 meters indoors, and can reach the shadow area where light and beams do not cover. For example, combining the high-speed transmission of VLC and the ubiquitous coverage of RF, the hybrid network in [18] shows that joint VLC and WiFi access can significantly improve the performance of indoor wireless communications over each network working alone.

The access point assignment (APA) issue however becomes a prominent challenge in hybrid networks. Unlike homogeneous networks, hybrid networks provide radically different overlaying coverage. Also, different types of APs differ in coverage range and system capacity, which makes APA more challenging. For example, when compared to VLC and/or mmWave, WiFi has a much larger coverage range but smaller system capacity, and therefore, using the signal strength strategy method, which always assigns a user to the AP offering the highest signal-to-noise ratio (SNR), would cause users to be more attracted to WiFi often leading to (i) traffic overload in WiFi and (ii) idleness in VLC/mmWave.

For this reason, load balancing is necessary in hybrid networks, like in the IoRL network.

Next sub-Section performs a literature review to examine recent advancements in load balancing solutions.

4.1 Review of relevant literature

Most relevant literature seeks for solutions to supplement the standalone VLC and mmWave networks with RF networks. This is because, compared to VLC and mmWave networks, RF networks have not only ubiquitous presence (high coverage area) and proper operation in non-line of site (LoS) environments but also the devices connected to RF networks do not suffer from VLC/mmWave interference and vice-versa [19]. Therefore, adding one or more RF APs to VLC and/or mmWave networks mitigates the LoS blockages, handover overhead, inter-cell interference, etc. However, the problem of finding a compromise between the high

coverage area RF networks and the high VLC/mmWave capacity networks remains. That is,

how to distribute the users among the APs (either RF, VLC or mmWave) to improve the overall system's performance with an acceptable fairness of the system.

The key idea is to associate the users who suffer from interference, handover overhead, blockages, etc. to the RF APs and keep the other users connected to the VLC and mmWave networks. Several techniques have been proposed to balance the load and tackle such APA issues by an efficient user distribution among either VLC/RF or mmWave/RF APs. Load balancing consists of two missions: the APs' assignment in APA and resource allocation, whether this resource is a time slot in TDMA or a subcarrier in OFDMA schemes.

Specifically, the APA in [20] distributes users between one RF AP and one VLC AP, where some users associated to the VLC AP alleviate the load of the RF AP, and the infeasible VLC connections are transferred to the RF AP. In [21] authors investigate the advantages of combining multiple RF and VLC APs and propose the dynamic distribution of users on both the VLC and RF networks, based on the users' channel condition, i.e., SNR, so as, users can migrate to APs with higher data rates. The APA in [22] is implemented assuming that the resources are allocated fairly among users to conclude that the hybrid network improves the performance significantly, compared to either VLC or RF standalone networks. Authors in [23] associate the users to a VLC network, and then, re-allocate the users receiving a lower data rate than a predefined threshold to RF APs. In [24] a centralised and distributed optimisation problem is formulated for user association to the APs (whether this AP is VLC, mmWave or RF AP) with allocating the resources jointly among users. In [25] a centralised problem for maximising proportional fairness is formulated as a mixed-integer non-linear programming (MINLP), with particularly high complexity. The same study proposes the distributed version of the algorithm which has lower, yet impractical, complexity. The load balancing scheme in [26] considers that static users are connected to VLC APs, while moving to an RF AP. Using same scenario, [27] formulated two proportional fairness problems of joint APA and resource allocation, and separate APA and resource allocation, to show that the former approach achieves better quality-of-service (QoS) for the users, but with a significant higher complexity, up to 1000 times greater, than the later. The study in [28], instead of assigning users to a specific AP, the problem is formulated by considering network hierarchal assignment by means of first assigning the network to each user, and then select the appropriate AP, in the assigned network, for each user. However, the formulated problem is for static systems, which motivated [29] and [30] to provide insights for dynamic systems including the non-LoS and LoS channel blockages. The assumption is that, users with high occurrence rate of channel blockages are assigned to the RF network, whereas users with low rate of blockages remain in the VLC network. Like in [31], the formulated load balancing problem is also MINLP, which makes the resulted algorithm highly complex.

Therefore, besides the system design, **how to formulate the problem and resolve it in low complexity is a non-trivial question**. For example, the solution methods used by the aforementioned studies for balancing the load in hybrid networks are:

- optimisation based algorithms,
- evolutionary game theory, and
- fuzzy-logic based algorithms.

These solution methods are considered the most common approaches and have been studied and compared in [32], which concluded that fuzzy logic-based algorithms outperform the other approaches when APA is to apply in dynamic systems, while the optimisation-based algorithms are best for the static systems by means of complexity and optimality level. In this research direction, [33], optimised a system consisting of a cascaded power-line communication (PLC)/VLC link, along with as RF link, where the total transmitted power under QoS constraints was minimised. The formulated optimisation problem was shown to be a convex problem that could be solved efficiently via Lagrangian method. In [34], authors formulate a power and carrier allocation problem for energy efficiency optimisation in software-defined VLC/RF network, which is the only study to consider load balancing through SDN/NFV consideration, to the best of our knowledge. The optimisation problem considers

the backhaul constraints, QoS requirements, and the inter-cell interference constraints. With the help of the SDN controller, the resource allocation strategy

can be requested as an application from the application layer, then through the SDN controller, the requested strategy can be implemented in the APs in the physical

layer. Because the objective function is the nonconvex function, the Dinckelbach (dual) approach was also used to convert the problem into a serial of convex optimisation problems.

Simi

lar to [35] our scope is to exploit the advantages of SDN/NFV technology to resolve the load balancing solution at the application layer and implement it at the physical layer by proposing a much smoother problem formulation and more practicable solution. For instance, as we show later in our theoretical modelling in Section 4.3, our design method converts the problem into convex meaning that the solution can be approached via pure Lagrangian (convex) analysis, which is shown to facilitate sufficiently lower algorithmical complexity with higher optimality level than Dinckelback-type (dual) approaches [36].

4.2 Empirical load balancing scheme and implementation

This Section presents the design and implementation of a load balancing approach based on empirical logic.

The main conclusion from our previous literature review is that load balancing consists a significant technology, which helps to save power and improve resource utilisation in network [37]. This is mainly because current network infrastructure relies on a vertical architecture with a variety of software and hardware creating, thereby, a system with limited flexibility, which issues undesirable effects recently coined as *Network Ossification* (NO).

Our hypothesis is that we can combat NO using SDN/NFV that facilitates breaking the vertical integration between the network control plane and its data plane, which bypasses the inflexibility of mainstream load balancing solutions that use dedicated load balancers for forwarding the user (or client) requests to different servers requiring dedicated hardware support that is expensive, lacks flexibility and is easy to become a single point failure [38]. More specifically, our SDN controller will be able to decide how to manage the packet or which task to perform such that to choose packets that can be dropped or added to a new flow entry for forwarding similar packets in next time instances, in a fast, inexpensive and convenient way.

Our aim is to distribute the different traffic flows carried by the IoRL home network through the different parallel VLC, mmWave and WiFi paths between source and destination.

Our focus will draw on using the OpenFlow protocol for establishing the communication between a Ryu controller and the network elements, where we will rely on empirical logic to implement an SDN/NFV load balancing process based on throughput threshold criteria and build and measured using Mininet simulator and iperf, respectively.

4.2.1 The role of OpenFlow, Ryu controller, iPerf and Mininet in the empirical design

We use OpenFlow because OpenFlow's central-control model is shown to avoid the need for constructing global policies from switch-by-switch and user-by-user configurations, and it can

also support near-optimal traffic management [39] For instance, the behaviour of a forwarding device can be summarised in two steps [40]:

Step 1: When it receives a packet that does not match a certain entry in its routing table, it contacts the controller that defines how the packet should be forwarded or discard the packet

Step 2: When the received packet matches a rule in its routing table, the corresponding action in the forwarding table is performed.

Moreover, as discussed in previous deliverable D3.1, there are five topmost open source controllers in terms of their usage, namely, POX, Ryu, Trema, FloodLight, and

OpenDaylight. For our empirical modelling **we choose Ryu controller because** of its

efficient performance for the development and prototyping of network control software. Also, Ryu has a Pythonic OpenFlow interface, which we are familiar with and used is similar developments to explore prototype distribution, SDN debugging, network virtualisation, controller design, and programming models. Ryu supports OpenFlow and can run even in low-performance computers (e.g. laptops) because it can be bundled with just-in-time PyPy compiler and CPython interpreter, which offer easy deployment.

Furthermore, **we use Mininet to** create a realistic virtual network, running real kernel, switch and application code, on a single machine (either virtual machine, cloud or native), in seconds, with a single command. Mininet is generally a convenient way to develop, share, and experiment with OpenFlow. Besides, at current development state of the IoRL system simulation in Mininet looks the only way we can test and evaluate the proposed load balancing solutions, while their actual implementation will take place at the IoRL SDN/NFV home environment (i.e. in VNF form) when the L2 system will be available.

To measure the maximum achievable bandwidth in our system **we also use iPerf**, which is a reliable tool for IP networks to support tuning of various parameters related to timing, buffers and protocols (TCP, UDP, SCTP with IPv4 and IPv6).

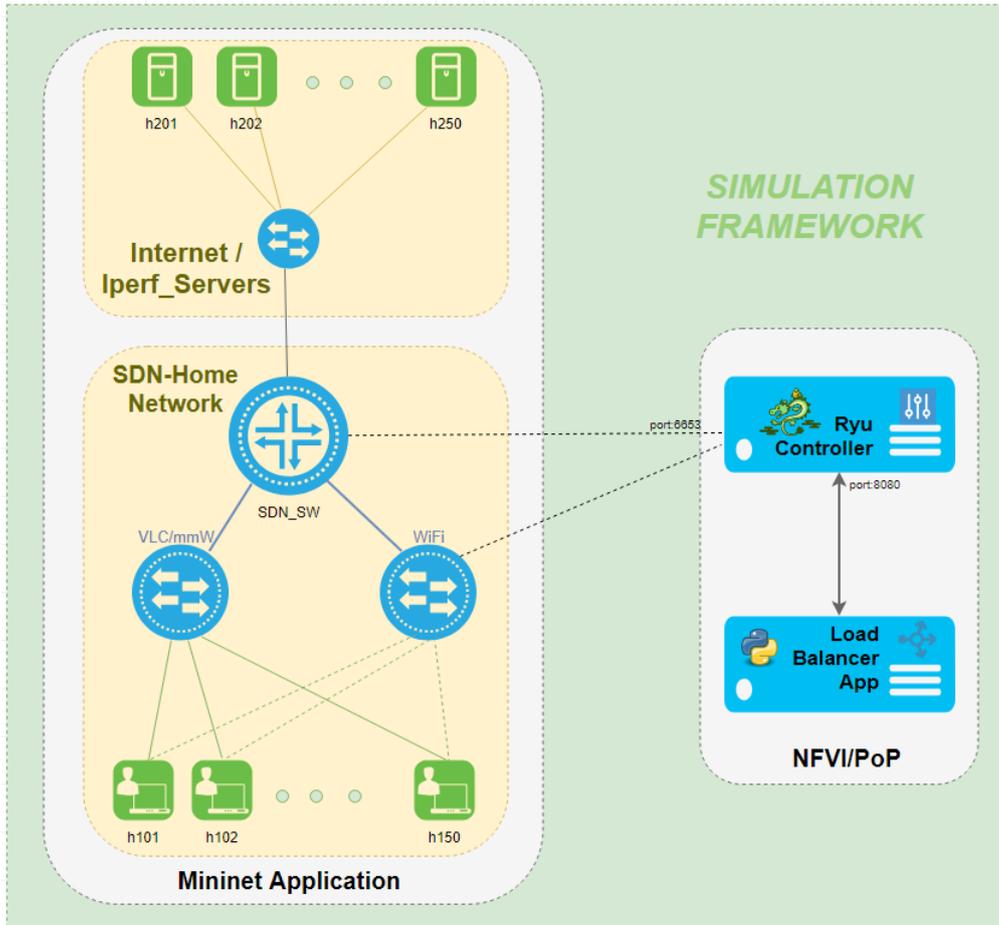


Figure 56 - Illustration of the of the SDN/NFV load balancing topology

4.2.2 Network design using empirical load balancing logic

Figure 56 illustrates the network topology, which we have taken for load balancing evaluation, and Figure 57 provides a snapshot of the topology instantiation in Mininet. At the right-hand side of the figure, we see the Ryu controller to implement the load balancing probes, while at the left-hand side Mininet is used for creating the virtual network topology including the WiFi and VLC/mmWave AP deployment. Our topology includes four SDN switches and fifty pairs of virtual iPerf monitored clients, where each client is set to request random traffic load. Under such scenario the Ryu controller is to install the flows at the SDN switches and communicate remotely with the load balancing application using the Restful API. In that way, the load balancing application will be able to perform tracking of connected users to WiFi and VLC/mmWave APs and install the new flows for each client based on a throughput threshold, namely *thr1*.

```

mininet@mininet-vm: ~
mininet@mininet-vm:~$ sudo mn --top single,6 --mac --arp --controller=remote
*** Creating network
*** Adding controller
Unable to contact the remote controller at 127.0.0.1:6633
*** Adding hosts:
h1 h2 h3 h4 h5 h6
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1) (h3, s1) (h4, s1) (h5, s1) (h6, s1)
*** Configuring hosts
h1 h2 h3 h4 h5 h6
*** Starting controller
*** Starting 1 switches
s1
*** Starting CLI:
mininet>

```

Figure 57 - Snapshot of the Mininet instantiation

When the total traffic requested by all users is above $thr1$ then the load balancing application will allocate random users to the WiFi network and keep the rest users to the high-speed VLC/mmWave access. In the same example, when the total traffic requested by all users is below $thr1$, then some users will randomly be allocated from the WiFi to the VLC/mmWave network. The flow diagram of the load balancing process is illustrated in Figure 58.

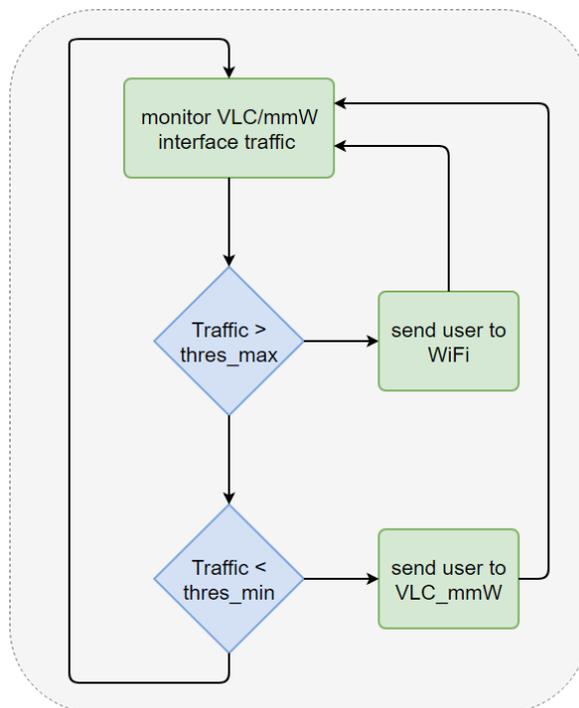


Figure 58 - Illustration the flow diagram including Ryu controller and load balancing application processes

4.2.3 Implementation using empirical load balancing logic

We implemented the load balancing process as VNF and instantiated in the IoRL home SDN/NFV environment following the process described in previous Section 4.2.2. For the sake of clarity, **the programming code of the load balancing VNF is available at**

<https://github.com/H2020-5G-IoRLproject/H2020-IoRL-code/> and we illustrate in Figure 59 the interface of the load balancing VNF with options for traffic redirection among the available WiFi and VLC/mmWave links.

```
themis@mnlab:~/load_balancing_bu/ryu$ cd ../app/
themis@mnlab:~/load_balancing_bu/app$ python lb.py

Start Load Balancing Application : (1)
Stop Load Balancing Application  : (2)
Print Hosts                       : (3)
Print Throughput                  : (4)
Exit                             : (0)
Please make a choice: █
```

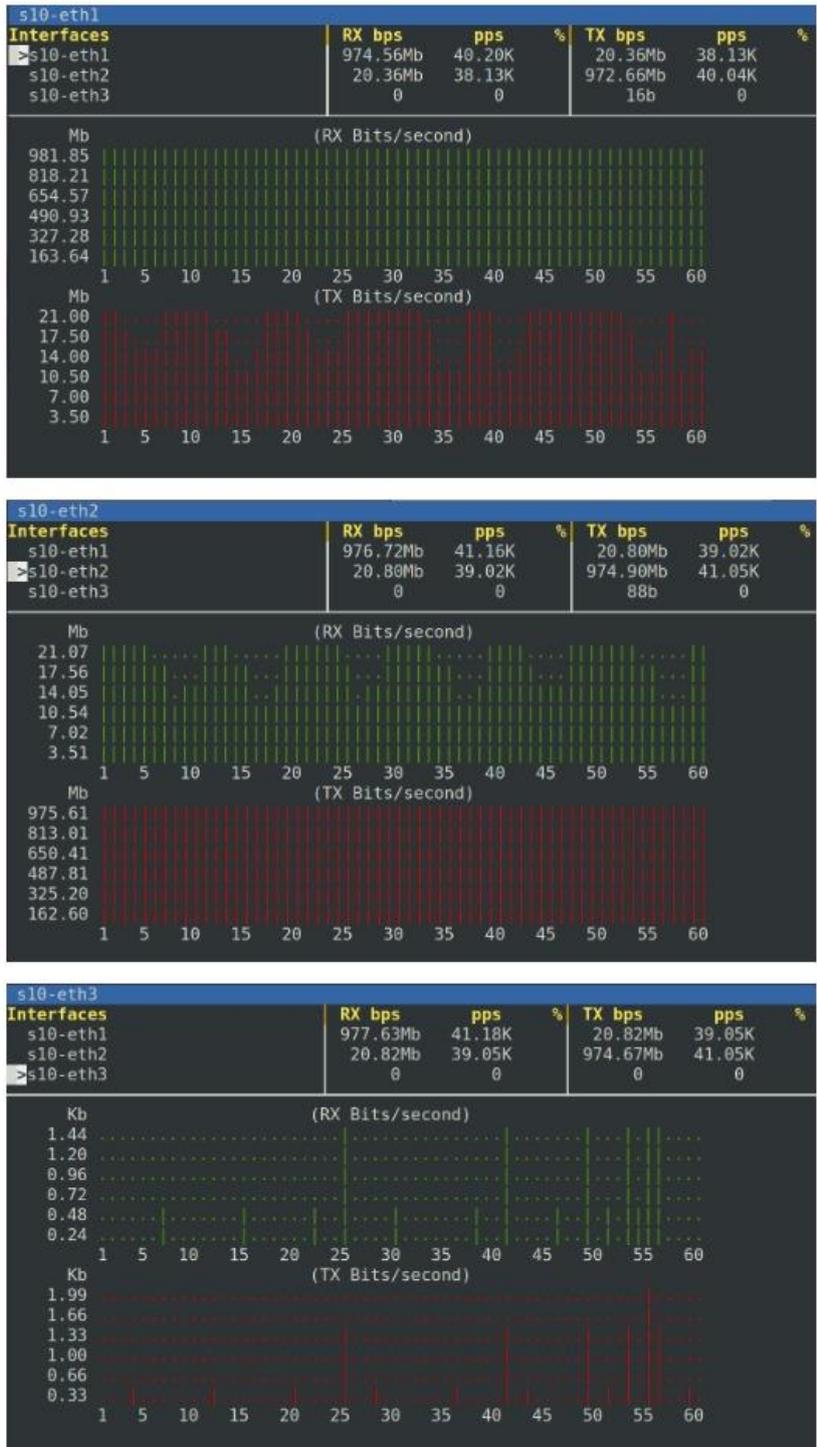
Figure 59 - Snapshot of the load balancing application with options for traffic redirection

Furthermore, Figure 60 and Figure 61 illustrate the data traffic in the IoRL home network before and after the instantiation of the load balancing VNF, respectively. For this experiment, we considered (i) 50 users in total with same priority each one requesting a random amount of throughput, (ii) 1 Gbps maximum WiFi link capacity, (iii) 1 Gbps maximum VLC/mmWave link capacity, and (iv) 10 Gbps maximum link capacity between the SDN/NFV server and the Internet (referred as external interface). From Figure 60 we see that before load balancing, all traffic is assigned to the VLC/mmWave link, while upon enabling the load balancing in Figure 61 traffic is redirected from the VLC/mmWave to WiFi link, e.g., traffic decreases at VLC/mmWave interface. This is because our application switches the network access to random users whenever the total network throughput exceeds the VLC/mmWave link capacity, which has been predefined by setting the threshold $thr1$ to $thr1 = 1$ Gbps .

Remark that ***we name such load balancing logic as “empirical”*** because although it has been proved practicable by numerous related applications, it is yet suboptimal for the following reasons:

- it relies on intuition without considering specific allocation policies and performance optimisation;
- it is user service and user priority agnostic since it does not consider assigning minimum QoS requirements and weights to individual users;
- it improves the total throughput performance in a rather asymptotic than systematic manner, e.g., without using specified strategy.

In an effort to improve our system and address the issues of the empirical design, the next sub-Section proposes two load balancing schemes based on analytical channel and policy modelling and optimisation of specific performances.

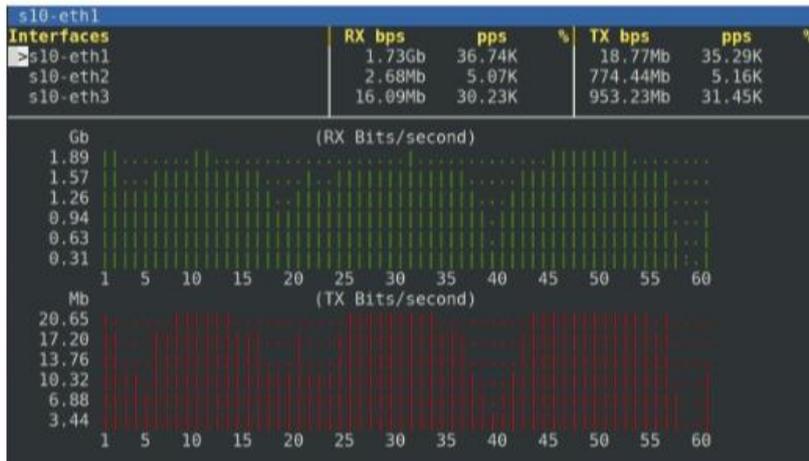


External Interface

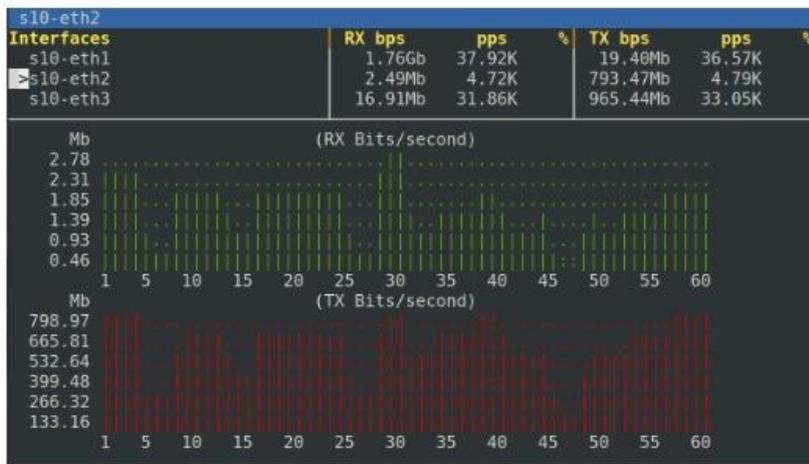
VLC/mmWave Interface

WiFi Interface

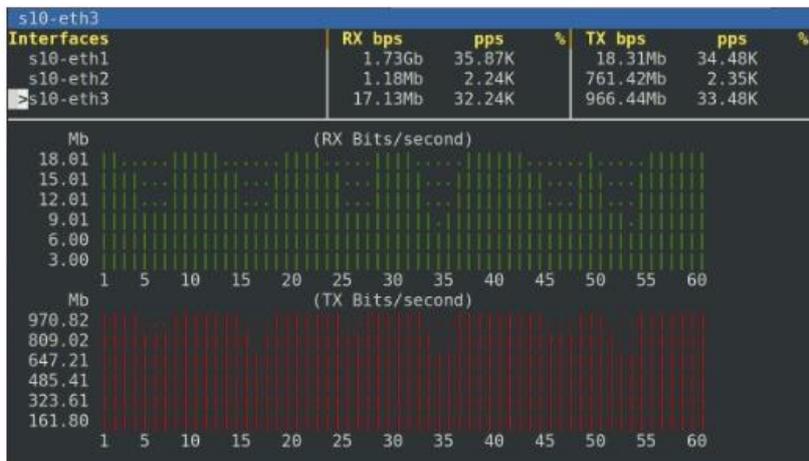
Figure 60 - Snapshot of the data traffic in the IoRL home network per interface before load balancing



External Interface



VLC/mmWave Interface



WiFi Interface

Figure 61 - Snapshot of the data traffic in the IoRL home network per interface after load balancing

4.3 Theoretical load balancing modelling, optimisation and implementation

This sub-Section is to propose and specify the benefits of analytical load balancing processes by considering that the channel state information and user coordinates can be sent from the MAC layer (L2) to Network layer (L3) in matrix form at fixed time intervals.

The purpose of such effort is the identification of the practical differences between our empirical load balancing logic, presented in previous Section 4, and explicit algorithm designs, built on axiomatic principles of system optimisation that are likely to better specify more compound load balancing criteria justified via decision-making problems and solution analysis.

The importance of our study for the IoRL home network can be perceived upon recalling from our previous deliverable D3.1 that the VLC and mmWave technologies are particularly prone to shadowing and blockage phenomena due to the many physical obstacles situated inside rooms/buildings, which unavoidably occur in practice and may force steering most traffic load from high-speed VLC/mmWave to low-speed WiFi access for long time periods. We further contribute by investigating the load balancing problem considering the QoS requests and priorities of each user individually in hybrid networks with three-tier VLC/mmWave/WiFi access, while relevant modellings consider two-tier network access, i.e., either mmWave/WiFi or VLC/WiFi. Besides, to the best of our knowledge, most of these studies mainly consider outdoor scenarios, while the application of load balancing process in indoor environments has yet been carefully studied, as intended in this deliverable.

The main research contributions of this work are highlighted below.

- We derive explicit solutions on the optimal AP indexing for two proposed algorithms via analysis. Such efforts are limited in relevant projects and literature studies, where programming-based (empirical) logic is mostly adopted, e.g., [41] [42] [43] [44] [45].
- We resolve the optimal indexing for specific APs rather than the indexing of the type of network access, as considered conventionally [46] [47] (see the below Figure 62 for example). This may also pave the way for the load balancing result to be calculated by taking advantage of the powerful computing capacity available in L3, which saves resources valuable for other tasks of L2, where computing capacity is fundamentally less.
- We account three types of network accesses (i.e. VLC, mmWave, WiFi), whereas most related designs perform load balancing between two types of network access (e.g. mmWave/WiFi or VLC/WiFi).
- We provide the detailed implementation pseudo code for each algorithm with extended examples and simulation results, where we demonstrate the superiority of our approaches in terms of individual user throughput performance due to fairness achieved via the additional minimum QoS and user priority criteria.

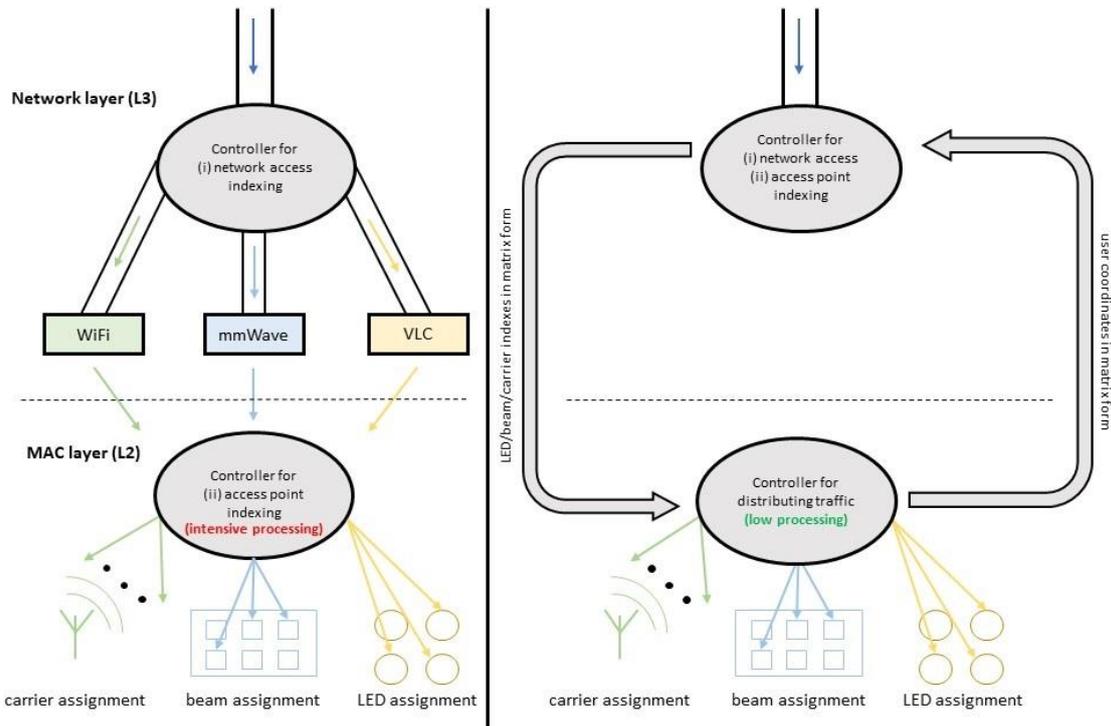


Figure 62 - Representation of conventional load balancing topology with intensive L2 processing for access point indexing (left), and proposed load balancing approach with joint network access and access point indexing at L3 (right).

Finally, we remark that for the sake of fair comparisons between the empirical and analytical schemes, as well as to arrange a clear picture of their main differences, we will consider relatively simple modellings of channel interference, user positioning, traffic distribution, etc., while more inclusive models are subject to future study, in case the evaluation results encourage such investigation.

4.3.1 Network and channel modelling

Let us consider a hybrid VLC, mmWave and WiFi home access network, where multiple users are distributed into a square-shaped room. A single WiFi access point (AP) is placed on the ground at the centre of the room, and four Remote Radio Light Heads (RRLHs) are fitted on the ceiling that situate the LED photodiodes and the mmWave modules. Each light-beam and mmWave-beam acts as an individual high-speed AP covering a confined small area, while the WiFi AP provides lower-speed data rate but covers the entire room area. To avoid modelling channel interference, we assume that the VLC APs operate at different frequencies, and that the mmWave APs have fixed directional beams [48]. All APs are connected to the IoRL Home IP Gateway (HIPG), which drives the data to the central SDN/NFV controller that performs the traffic steering among the three available RATs by evaluating the signal-to-noise-rate (SNR) with respect to user positioning in the room. For instance, the relative distance of user j with coordinates of (x_j, y_j) for AP i with coordinates of (x_i, y_i) can be calculated by the formula:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{5}$$

Next sub-Sections focus on modelling each VLC, mmWave and WiFi channel to derive the SNR function including effects due to channel blockage and shadowing.

4.1.1.1 VLC channel modelling with blockage considerations

A typical VLC channel with blockage is comprised of line-of-sight (LoS) and non-line-of-sight (NLoS) light-beam paths, which reach and do not reach the end-user, respectively. For instance, given the AP index $i=1,\dots,I$ and user index $j=1,\dots,J$, the LoS path can be described by the straight line between i and j , and the corresponding Euclidean distance is denoted by d_{ij} . Also, by denoting with (i) α_{ij} and β_{ij} the angles of irradiance and incidence, respectively, (ii) $\Phi_{1/2}$ the radiation angle where the light intensity is half of the intensity of the main-beam direction, (iii) A^{PD} the physical area of LEDs (photodiodes), (iv) g^f the gain of the optical filter, (v) B^{max} the semi-angle of the field of view (FoV) of each LED, and (vi) n the refractive index, we can represent the channel gain of LoS path with

$$G_{ij}^{\text{LoS}} = \frac{\left(1 - \frac{\ln(2)}{\ln(\cos(\Phi_{1/2}))}\right) \cdot A^{\text{PD}}}{2 \cdot \pi \cdot d_{ij}^2} \cdot \cos\left(\frac{\ln(2)}{\ln(\cos(\Phi_{1/2}))}\right) (\alpha_{ij}) \quad (6)$$

$$\times g^f \cdot g^c \cdot \cos(\beta_{ij}), \forall i \in I, j \in J$$

and the optical concentrator gain with

$$g^c = \begin{cases} \frac{n^2}{\sin^2(B^{\text{max}})}, & 0 \leq \beta_{ij} \leq B^{\text{max}} \\ 0, & \beta_{ij} > B^{\text{max}} \end{cases} \quad (7)$$

Furthermore, the modelling of NLoS can include complex policies with n -order reflections of the light-beams. For the purpose of simplicity, in this deliverable D3.2 we consider first-order reflections, which consist of two segments: (a) from AP i to a point x on the wall, and (b) from the point x to home user j . We denote with (i) d_{ix} and d_{xj} the Euclidean distances of these two segments, respectively, (ii) φ_{ix} and θ_{ix} the angles of radiance and incidence with respect to the first segment, (iii) φ_{xj} and θ_{xj} the angles for the second segment, and (iv) A^w and L a random surface of the wall with its respective wall reflectivity. The expression of NLoS channel gain can be then given by

$$G_{ij}^{\text{NLoS}} = \int_{A^w} \frac{\left(1 - \frac{\ln(2)}{\ln\left(\cos\left(\frac{\Phi_1}{2}\right)\right)}\right) \cdot A^{\text{PD}}}{2 \cdot (\pi \cdot d_{ix} \cdot d_{xj})} \cdot L \cdot \cos\left(\frac{\ln(2)}{\ln\left(\cos\left(\frac{\Phi_1}{2}\right)\right)}\right) (\varphi_{ix}) \quad (8)$$

$$\times g^f \cdot g^c \cdot \cos(\varphi_{xj}) \cdot \cos(\theta_{ix}) \cdot \cos(\theta_{xj}) dA^w, \forall i \in I, j \in J,$$

and, by recalling (6), (7) & (8), the total gain of each VLC channel link (i, j) is given by

$$G_{ij}^{\text{VLC}} = G_{ij}^{\text{LoS}} + G_{ij}^{\text{NLoS}}, \forall i \in I, j \in J \quad (9)$$

Moreover, we consider a blockage process, which occurs due to signal reflection to walls, furniture, home user bodies, etc., and impacts the total gain for each (i, j) VLC link in (9). Remark that we will study the issue of VLC and mmWave channel blockage, assuming that users are immobile during the period of interest, and that user's position is randomly and uniformly distributed in the room, where each user can be connected to one AP only, e.g., either VLC, mmWave or WiFi. With respect to VLC channel blockage, there are three key elements for each (i, j) VLC link pair: occurrence rate λ_{ij} , occupation rate η_{ij} and blockage degree ξ_{ij} . Occurrence rate describes the number of channel blockages happening per unit time, occupation rate is the proportion of time where a user experiences channel blockage and blockage degree is a binary variable to indicate at what extent the channel blockage affects the VLC channel quality, i.e., $\xi_{ij} = 1$ stands for a complete blockage, while $\xi_{ij} = 0$ means no blockage. Without loss of generality, we consider $\xi_{ij} = 1$ for all users when channel blockage occurs. At the VLC receiver, the gathered photons are converted into an electric current denoted by [49].

$$I_{ij} = \frac{(1 - \xi_{ij}) \cdot R \cdot P \cdot G_{ij}^{\text{VLC}}}{\zeta}, \quad \forall i \in I, j \in J \quad (10)$$

with R the detector responsivity, P the transmitted light(optical) power, and ζ the optical to electric power conversion coefficient. In light of (10) we recall that VLC modulates the light intensity of LEDs, so as, to convey information bits [50]. Photodiodes, semiconductor devices that convert light into current, can be then employed at the receiver to detect signals using the electric current expression in (10), which yields that the SNR of a VLC user can be written as

$$SNR_{ij}^{\text{VLC}} = \frac{I_{ij}}{N^{\text{VLC}} \cdot BW^{\text{VLC}}}, \quad \forall i \in I, j \in J \quad (11)$$

with N^{VLC} and BW^{VLC} the power spectral density (PSD) of noise at the LED, and the system bandwidth of the VLC AP, respectively.

4.1.1.2 mmWave channel modelling with shadowing considerations

it is well-known that the mmWave signal is susceptible to high path loss, fading, noise, and interference. All these phenomena cause serious degradation in SNR at the receiver leading to poor overall system performance. Without loss of generality, mmWave communication channel has been represented by the double directional channel model, where the channel gain between user j and the mWave AP $n \in N$ is given by [51].

$$H_{nj}^{\text{mmWave}} = \sum_{q=1}^Q \sum_{c=1}^C \left[H_{qc} \cdot e^{\gamma_{qc}} \cdot (\tau - \tau_{qc}) \cdot (\varphi^R - \varphi_{qc}^A) \right. \\ \left. \times (\varphi^T - \varphi_{qc}^D) \cdot (\theta^R - \theta_{qc}^A) \cdot (\theta^T - \theta_{qc}^D) \right], \quad \forall n \in N, j \in J, \quad (12)$$

where $q = 1, \dots, Q$ is the index of the number of multipath components, $c = 1, \dots, C$ the index of the number of rays in the room (i.e. each IoRL RRLH has 4 mmWave rays in total). Each ray is

represented by the path gain H_{qc} , the phase γ_{qc} , time of arrival τ_{qc} , and the azimuth angle-of-arrival ϕ_{qc}^A , azimuth angle-of-departure ϕ_{qc}^D , zenith angle-of-arrival θ_{qc}^A , zenith angle-of-departure θ_{qc}^D , with the respective $\varphi^R, \varphi^T, \theta^R, \theta^T$ the intensities at the mmWave transceiver. Hence, the task of our channel modelling is thus to find all these parameters for a mmWave communication channel. Interestingly, [52] and [53] propose comprehensive channel models to fit the LoS and NLoS transmission links from 30GHz to 73GHz frequency bands. Remark that these bands are selected because they have been mostly deployed in mmWave cellular network and are likely to be adopted by the IoRL network too. These two studies specify that small-scale fading has a less impact on mmWave signal propagation, and hence the large-scale fading effect is considered in measurements. Also, they specify that path loss effects in mmWave channel can be approached by a statistical model based on realistic measurements, which is summarised as

$$PL_{ij}^{\text{mmWave}} = a + b \cdot 10 \cdot \log_{10}(d_{ij}) + \xi, \quad \xi \in (0, \sigma_s^2), \quad (13)$$

with PL_{ij}^{mmWave} the path loss in dB, a and b are the least square fits of floating intercept and slope over the measured distances up to 200 meters, d_{ij} the distance between user j and the mmWave AP i , σ_s^2 is the variance of the lognormal shadowing ξ and all parameters are specified in Table 6

Table 6 - Path loss parameters based on statistical model in [Ref] and [Ref]

Path loss parameters based on statistical model in [54] and [55]				
Pathloss parameter	a	b	σ_s^2	ξ
30GHz NLoS	72.0	2.92	8.7	1.0
LoS	61.4	2.0	5.8	5.0
73GHz NLoS	86.6	2.45	8.0	1.0
LoS	69.8	2.0	5.8	8.0

We use the path loss model in (13) to determine the received signal power at each user j , and then we will use the signal power to estimate the SNR of the j -th user in the ν -th mmWave AP, which is described by

$$SNR_{nj}^{\text{mmWave}} = 10 \cdot \log_{10} |h_{nj}|^2 + P_n + G_j - PL_{nj}^{\text{mmWave}} - \left(10 \cdot \log_{10} BW^{\text{mmWave}} + N^{\text{mmWave}} + N_0^{\text{mmWave}} \right), \quad (14)$$

where p_i is the i -th mmWave AP total transmit power, N_0^{mmWave} and N^{mmWave} the noise power density and the noise figure at the AP, respectively, $G_j = 20 \cdot \log_{10} \left(\frac{\pi \cdot l}{\lambda} \right)$ is calculated using the transceiver antenna length of l and propagation environment density λ . Furthermore, assuming that each link experiences independent Nakagami distributed fading, the small-scale fading component $|h_{nj}|^2$ can be modelled as normalised Gamma random variable $|h_{nj}|^2 \sim \mathfrak{Z}(\sigma^\nu, 1/\sigma^\nu)$, with σ^ν to be very large integer for LoS link, and $\sigma^\nu = 1$ when the link is NLoS. For the sake of clarity, we refer Section 4.3.3 for the exact values of the parameters in (13) and (14) used during our simulations.

4.1.1.3 WiFi channel modelling

The WiFi channel has been extensively modelled by the relevant literature, like in [56], [57], [58], [59], [60], [61]. The purpose of the sub-Section is to pick most inclusive among these extensive modellings and simplify its channel description towards a smooth SNR function that facilitates the theoretical evaluation of traffic steering processes. For instance, according to [62], the path loss model of indoor RF propagation under central carrier frequency f^c is subject to the free space loss $F^{\text{FS}}(d) = 20 \cdot \log_{10}(d) + 20 \cdot \log_{10}(f^c)$ up to a breakpoint distance d^{BP} between the WiFi AP and the user, and it is described by

$$F(d) = \begin{cases} F^{\text{FS}}(d) + X^\sigma, & d \leq d^{\text{BP}} \\ F^{\text{FS}}(d) + 35 \cdot \log_{10} \left(\frac{d}{d^{\text{BP}}} \right) + X^\sigma, & d > d^{\text{BP}} \end{cases} \quad (15)$$

where X^σ is a Gaussian random variable with zero mean and deviation σ . Considering a Ricean Λ -factor for channel coefficient distribution (i.e. $\Lambda = 1$ before the breakpoint at d^{BP} and $\Lambda = 0$ after the breakpoint), where X^σ has unit variance, i.e., X^1 , the multipath propagation of the WiFi channel is given by [[63], eq. (3.13)]

$$H_j^{\text{WiFi}} = \sqrt{\frac{\Lambda}{\Lambda+1}} \cdot e^{j\omega} + \sqrt{\frac{\Lambda}{\Lambda+1}} \cdot X^1, \quad \forall j \in J \quad (16)$$

with ω the angle of arrival/departure of the LoS component. With (15) and (16), the channel gain of the WiFi channel for each user j at distance d_j from the AP can be defined as

$$G_j^{\text{WiFi}} = H_j^{\text{WiFi}} \cdot 10^{-\frac{F(d_j)}{10}}, \quad \forall j \in J \quad (17)$$

with the corresponding SNR to be calculated by

$$SNR_j^{\text{WiFi}} = \frac{G_j^{\text{WiFi}} \cdot P^{\text{WiFi}}}{BW^{\text{WiFi}} \cdot N^{\text{WiFi}}} \quad (18)$$

with P^{WiFi} the total transmit power, BW^{WiFi} the total bandwidth and N^{WiFi} the transmit power spectral density at the WiFi receiver.

4.3.2 Theoretical load balancing schemes

Having defined the LoS and NLoS formulas of SNR for VLC, mmWave and WiFi channel in (11), (14) and (18), respectively, we propose two load balancing algorithm designs for AP assignment. The first algorithm follows the logic of our empirical approach, where switching between APs occurs accounting certain thresholds in terms of throughput. In that case the AP assignment process is purely opportunistic because the decision is not drawn on specific user criteria or characteristics, such as minimum user QoS requirements, user priorities, etc. That means those users with best positioning (or highest SNR) may occupy most (if not all) APs, while the users in bad positioning may be allocated with sufficiently less APs or even none. Hence, we propose a second algorithm that aims to maximise the overall throughput subject to minimum QoS and priority criteria for each user, towards fairer and more practicable AP allocation.

4.3.2.1 Pure opportunistic load balancing scheme

The first proposed algorithm, namely Pure Opportunistic Load Balancing (POLB), follows a two-stage logic. At first stage, it fully utilises the highest possible spatial spectral efficiency of the system by prioritising users to be allocated to VLC APs. At second stage, it uses the data rate thresholds $thr1 > thr2$ to allocate users achieving lower data rates than (i) $thr1$ to mmWave APs, and (ii) $thr2$ to WiFi APs. More precisely, POLB initially applies a criterion of maximal effective throughput, that is, for user j , the VLC AP achieving the highest communication link data rate is expressed as

$$S_j^1 = \arg \max_{i \in D^{\text{VLC}}} SNR_{ij}^{\text{VLC}} \quad (19)$$

with $D^{\text{VLC}} = \{i | i \in [1, I], i \in \square\}$ the set of the VLC APs. We remark that in the IoRL system each AP can service more than one user, however, to make our design tractable for demonstration we assume that each AP can be allocated to one user only, while users can link with several and different type of APs, which, based on Shannon's capacity formula, yields that the data rate for each user j at AP $i \in D^{\text{VLC}}$ is given by $R_j^{\text{VLC}} = (BW^{\text{VLC}} \cdot \log_2(1 + SNR_{ij}^{\text{VLC}}))$. Similarly, the mmWave AP achieving the highest communication link is given as

$$S_j^2 = \arg \max_{n \in D^{\text{mmWave}}} SNR_{nj}^{\text{mmWave}} \quad (20)$$

with $D^{\text{mmWave}} = \{n | n \in [1, N], n \in \square\}$ the set of the mmWave APs, and $R_j^{\text{mmWave}} = (BW^{\text{mmWave}} \cdot \log_2(1 + SNR_{nj}^{\text{mmWave}}))$ the data rate of user j at AP $n \in D^{\text{mmWave}}$. In the re-allocation part, users satisfying the condition $R_j^{\text{VLC}} < thr1$ are linked with mmWave APs, while the users with $R_j^{\text{mmWave}} < thr2$ are allocated to WiFi. However, in the WiFi system, the data rates of users also need to be improved to satisfy the average data rate requirement. For this reason, POLB allocates the optimal WiFi carriers to the users according the SNR_j^{WiFi} in the current state. Therefore, similar to the VLC and mmWave system, POLB employs the criterion of maximal throughput for channel allocation in the WiFi system too. That is, the optimal WiFi channel for user j is the channel with highest SNR among others, which is expressed as

$$S_j^3 = \arg \max_{w \in D^{\text{mmWave}}} \text{SNR}_{wj}^{\text{WiFi}} \quad (21)$$

with $D^{\text{WiFi}} = \{w | w \in [1, W], w \in \square\}$ the set of the WiFi channels and $R_j^{\text{WiFi}} = BW^{\text{WiFi}} \cdot \log_2(1 + \text{SNR}_{wj}^{\text{WiFi}})$. In view of (19), (20) and (21), the optimal AP allocation for each user j can be determined by seeking for each user j for the highest SNR_j^* among the three access technologies, which is given by the search

$$\text{For each } j \text{ within } 1: J$$

$$S_j = \begin{cases} S_j^1, R_j^{\text{VLC}} \geq \text{thr1} \\ S_j^2, \text{thr1} \leq R_j^{\text{mmWave}} \leq \text{thr2} \\ S_j^3, R_j^{\text{WiFi}} < \text{thr2} \end{cases} \quad (22)$$

In that case, the instantaneous user rate can be defined as $R_j = R_j^{\text{VLC}} + R_j^{\text{mmWave}} + R_j^{\text{WiFi}}$, with $R^{\text{op.Total}} = \sum_{j=1}^J (S_j \cdot R_j)$ the maximum achievable throughput over all users and APs. The search in (22) is implemented using the below Algorithm 1, which is of low linear complexity in the number of participating users and the number of APs, i.e., in the order of $O((I+J+W) \times J)$, with $O(\cdot)$ the big-O notation.

Algorithm 1: POLB implementation pseudo code (Purely Opportunistic Load Balancing)	
1:	Initialisation: acquire the AP positioning $\{\chi_i, \psi_i\}, \{\chi_n, \psi_n\}, \forall i \in I, n \in N$
2:	acquire user coordinates $\{\chi_j, \psi_j\}$ from matrices $\mathbf{V}_{I \times J}$ and $\mathbf{M}_{N \times J}$ of L2
3:	acquire LoS/NLoS parameters $\alpha_{ij}, \beta_{ij}, \varphi_{ix}, \theta_{ix}, \varphi_{xj}, \theta_{xj}, \xi_{ij}$ from L2
4:	for all user $j=1$ to J do
5:	set $H_{nj}^{\text{mmWave}} = \alpha, H_j^{\text{WiFi}} = \beta$ homogeneous $\forall j$, with α and β positive small numbers
6:	compute d_{ix}, d_{xj}, d_{ij} using (1)
7:	compute $\text{SNR}_{ij}^{\text{VLC}}, \text{SNR}_{ij}^{\text{mmWave}}, \text{SNR}_j^{\text{WiFi}}$ using (7), (10) and (14), respectively
8:	compute $R_j(\text{SNR}_{ij}^{\text{VLC}}, \text{SNR}_{ij}^{\text{mmWave}}, \text{SNR}_j^{\text{WiFi}})$ using the results of Step 7
9:	compute S_j^1, S_j^2 and S_j^3 using (15), (16) and (17), respectively
10:	if $R_j \geq \text{thr1}$

11:	allocate user j to VLC AP $S_j^* = S_j^1$ according to (18)
12:	else if $thr1 \leq R_j \leq thr2$
13:	allocate user j to mmWave AP $S_j^* = S_j^2$ according to (18)
14:	else
15:	allocate user j to WiFi AP $S_j^* = S_j^3$
16:	end for

4.3.2.2 QoS- and priority-aware load balancing scheme

The second proposed algorithm, namely QoS- and Priority-aware Load Balancing (QPLB), is to alter the opportunistic AP distribution of Algorithm 1, with a fairer AP distribution logic for maximising the overall throughput subject to priority weights and minimum QoS throughput requirement for each user. More precisely, QPLB defines the utility \hat{R}_j to describe the aggregated throughput of each user over all the allocated APs (i.e. considering three types of network access) as

$$\begin{aligned} \hat{R}_j &= (R_{ij}^{\text{VLC}} + R_{nj}^{\text{mmWave}} + R_{wj}^{\text{WiFi}}) \\ &= \left[\sum_{i=1}^I s_{ij} \cdot BW^{\text{VLC}} \cdot \log_2(1 + SNR_{ij}^{\text{VLC}}) + \sum_{n=1}^N s_{nj} \cdot BW^{\text{mmWave}} \cdot \log_2(1 + SNR_{nj}^{\text{mmWave}}) \right. \\ &\quad \left. + s_{wj} \cdot \sum_{w=1}^W BW^{\text{WiFi}} \cdot \log_2(1 + SNR_{wj}^{\text{WiFi}}) \right], \forall j, \end{aligned} \quad (23)$$

where $s_{ij} \in \{0,1\}$, $s_{nj} \in \{0,1\}$, $s_{wj} \in \{0,1\}$ are the elements of the two-dimensional matrices $\mathbf{S}_{I \times J} = [s_{ij}]$, $\mathbf{S}_{N \times J} = [s_{nj}]$, $\mathbf{S}_{W \times J} = [s_{wj}]$ of the AP indexing over VLC, mmWave and WiFi access for each user j , respectively. Also, QPLB considers assigning weights to distinguish users into three priority classes, namely high priority Class 1, medium priority Class 2 and low priority Class 3, which are specified by the set $\{q_j^{\text{Class}(\zeta)}\}$, with $\zeta = 1, 2, 3$ the index of user Class and $q_j^{\text{Class}(1)} > q_j^{\text{Class}(2)} > q_j^{\text{Class}(3)}$. Based on (23) and the weight definition, we formulate the system throughput maximisation problem constrained on user's classes and their minimum QoS requirements described below.

$$\begin{aligned} \max_{s_{ij} \in \{0,1\}, s_{nj} \in \{0,1\}, s_{wj} \in \{0,1\}} f &= \sum_{j=1}^J (q_j \cdot \hat{R}_j) \\ \text{subject to: (C1)} \quad \sum_{i=1}^I s_{ij} &\leq 1, \forall j, \quad \text{(C3)} \quad \sum_{w=1}^W s_{wj} \leq 1, \forall j, \quad \text{(C5)} \quad q_j \cdot \hat{R}_j \geq \text{QoS}_j^{\min}, \forall j, \\ \text{(C2)} \quad \sum_{n=1}^N s_{nj} &\leq 1, \forall j, \quad \text{(C4)} \quad \sum_{j=1}^J q_j = 1, \end{aligned} \quad (24)$$

The optimisation objective f in (24) represents the sum weighted throughout of all system users, which is to be maximised over the indexes s_{ij} , s_{nj} , s_{wj} . Also, constraints (C1)-(C3) are to relax the binary (discrete) indexes s_{ij} , s_{nj} , s_{wj} into continuous variables $s_{ij} \in (0,1]$, $s_{nj} \in (0,1]$, $s_{wj} \in (0,1]$ respectively by describing them as the proportion of time that an AP is allocated to user j (known as time-sharing relaxation). Finally, (C4) balances the distribution of priority classes among users ensuring that the summation of the assigned weights should equal to one, while constraint (C5) ensures that the assigned throughput to each user should be at least equal to its minimum QoS defined as $QoS_j^{\min} \leq thr2$. Given that AP indexes have been relaxed into continuous variables, the optimisation objective function f and the constraints (C1)-(C5) are either logarithmic concave or affine differentiable functions at points s_{ij} , s_{nj} , s_{wj} meaning that the optimisation problem (24) is concave with respect to s_{ij} , s_{nj} , s_{wj} and it converges to global optimal points $\{s_{ij}^*\}$, $\{s_{nj}^*\}$, $\{s_{wj}^*\}$ within the space determined by the intersection between regions $\langle I \times J \rangle$, $\langle N \times J \rangle$, $\langle W \times J \rangle$ of $\mathbf{S}_{I \times J} = [s_{ij}]$, $\mathbf{S}_{N \times J} = [s_{nj}]$, $\mathbf{S}_{W \times J} = [s_{wj}]$ for all users, respectively. Therefore, we can use the Lagrangian method to resolve problem (24), which we do following three steps. At first step, we define the Langrangian function f_L of problem (24) as

$$\begin{aligned}
f_L &= \sum_{j=1}^J q_j \cdot \hat{R}_j - \sum_{j=1}^J \eta_j^{(1)} \left(\sum_{i=1}^I (s_{ij}) - 1 \right) - \sum_{j=1}^J \eta_j^{(2)} \left(\sum_{n=1}^N (s_{nj}) - 1 \right) - \sum_{j=1}^J \eta_j^{(3)} \left(\sum_{w=1}^W (s_{wj}) - 1 \right) \\
&\quad + \beta \cdot \left(\sum_{j=1}^J q_j - 1 \right) + \sum_{j=1}^J \left((\gamma_j \cdot q_j \cdot \hat{R}_j) - QoS_j^{\min} \right) \\
&= \sum_{j=1}^J q_j \cdot \left[\sum_{i=1}^I s_{ij} \cdot BW^{\text{VLC}} \cdot \log_2(1 + SNR_{ij}^{\text{VLC}}) + \sum_{n=1}^N s_{nj} \cdot BW^{\text{mmWave}} \cdot \log_2(1 + SNR_{nj}^{\text{mmWave}}) \right] \\
&\quad + s_{wj} \cdot \sum_{w=1}^W BW^{\text{WiFi}} \cdot \log_2(1 + SNR_{wj}^{\text{WiFi}}) - \sum_{j=1}^J \eta_j^{(1)} \left(\sum_{i=1}^I (s_{ij}) - 1 \right) - \sum_{j=1}^J \eta_j^{(2)} \left(\sum_{n=1}^N (s_{nj}) - 1 \right) - \sum_{j=1}^J \eta_j^{(3)} \left(\sum_{w=1}^W (s_{wj}) - 1 \right) \\
&\quad + \beta \cdot \left(\sum_{j=1}^J q_j - 1 \right) + \sum_{j=1}^J \left(\gamma_j \cdot q_j \cdot \left[\sum_{i=1}^I s_{ij} \cdot BW^{\text{VLC}} \cdot \log_2(1 + SNR_{ij}^{\text{VLC}}) + \sum_{n=1}^N s_{nj} \cdot BW^{\text{mmWave}} \cdot \log_2(1 + SNR_{nj}^{\text{mmWave}}) \right] \right. \\
&\quad \left. + s_{wj} \cdot \sum_{w=1}^W BW^{\text{WiFi}} \cdot \log_2(1 + SNR_{wj}^{\text{WiFi}}) \right] - QoS_j^{\min} \Big), \tag{25}
\end{aligned}$$

where $\eta_j^{(1)}$, $\eta_j^{(2)}$, $\eta_j^{(3)}$, β and γ_j are the Langrangian multipliers related to constraints (C1)-(C5), respectively. At second step, we recall the primal feasibility and complementary

slackness Karush-Kuhn-Tucker (KKT) conditions that yield $\frac{\partial f_L}{\partial s_{ij}} \Big|_{s_{ij}, s_{nj}, s_{wj}, \eta_j^{(1)}, \eta_j^{(2)}, \eta_j^{(3)}, \beta, \gamma_j = s_{ij}^*, s_{nj}^*, s_{wj}^*, \eta_j^{(1)*}, \eta_j^{(2)*}, \eta_j^{(3)*}, \beta^*, \gamma_j^*} = 0$, $\frac{\partial f_L}{\partial s_{nj}} \Big|_{s_{ij}, s_{nj}, s_{wj}, \eta_j^{(1)}, \eta_j^{(2)}, \eta_j^{(3)}, \beta, \gamma_j = s_{ij}^*, s_{nj}^*, s_{wj}^*, \eta_j^{(1)*}, \eta_j^{(2)*}, \eta_j^{(3)*}, \beta^*, \gamma_j^*} = 0$, which with some

manipulations result to

$$\eta_j^{(1)*} = (\gamma_j^* + 1) \cdot q_j \cdot BW^{\text{VLC}} \cdot \log_2(1 + SNR_{ij}^{\text{VLC}}) \tag{26}$$

$$\eta_j^{(2)*} = (\gamma_j^* + 1) \cdot q_j \cdot BW^{\text{mmWave}} \cdot \log_2(1 + SNR_{nj}^{\text{mmWave}}) \quad (27)$$

$$\eta_j^{(3)*} = (\gamma_j^* + 1) \cdot q_j \cdot BW^{\text{WiFi}} \cdot \log_2(1 + SNR_{wj}^{\text{WiFi}}) \quad (28)$$

From equations (26)-(28) we can derive the optimal Langrangian multipliers $\eta_j^{(1)*}$, $\eta_j^{(2)*}$, $\eta_j^{(3)*}$ related to the time-sharing indexing constraints (C1)-(C3). Remark that each multiplier is a function of the weight q_j and the optimal Langrangian multiplier γ_j^* of each user. Although $\{q_j\}$ are known input for each user, $\{\gamma_j^*\}$ are yet unknown variables and depend on the minimum QoS requirement QoS_j^{\min} of each user. Intuitively, if we find $\{\gamma_j^*\}$ then $\{\eta_j^{(c)*}\}$ can be adjusted according to both user weights and minimum QoS meaning that it can be later used as comparison metric to derive the optimal indexes \hat{s}_{ij}^* , \hat{s}_{nj}^* , \hat{s}_{wj}^* of each user at each network access, i.e.,

$$\begin{aligned} \{\hat{s}_{ij}^*, \hat{s}_{nj}^*, \hat{s}_{wj}^*\} &= \arg \max \{\eta_j^{(1)*}, \eta_j^{(2)*}, \eta_j^{(3)*}\}, \\ \{\eta_j^{(1)*}, \eta_j^{(2)*}, \eta_j^{(3)*}\} &= \begin{cases} \{\eta_j^{(1)*}, \eta_j^{(2)*}, \eta_j^{(3)*}\}, & \text{if } \{\hat{s}_{ij} = \hat{s}_{ij}^*, \hat{s}_{nj} = \hat{s}_{nj}^*, \hat{s}_{wj} = \hat{s}_{wj}^*\} \\ 0, & \text{otherwise} \end{cases} \end{aligned} \quad (29)$$

At third step, we find $\{\gamma_j^*\}$ by resolving the below system of equations.

$$\begin{aligned} f_j(\hat{s}_{ij}, \hat{s}_{nj}, \hat{s}_{wj}) &= \gamma_j \cdot q_j \cdot \left[\sum_{i=1}^I \hat{s}_{ij} \cdot BW^{\text{VLC}} \cdot \log_2(1 + SNR_{ij}^{\text{VLC}}) + \sum_{n=1}^N \hat{s}_{nj} \cdot BW^{\text{mmWave}} \cdot \log_2(1 + SNR_{nj}^{\text{mmWave}}) \right. \\ &\quad \left. + \hat{s}_{wj} \cdot \sum_{w=1}^W BW^{\text{WiFi}} \cdot \log_2(1 + SNR_{wj}^{\text{WiFi}}) \right] - QoS_j^{\min} = 0, \quad \forall j \end{aligned} \quad (30)$$

where $f_j(\hat{s}_{ij}, \hat{s}_{nj}, \hat{s}_{wj}) < 0$ indicates that the QoS constraint (C5) in (24) is violated. Also, to resolve (30) QPLB assumes that indexes \hat{s}_{ij} , \hat{s}_{nj} , \hat{s}_{wj} are uniformly distributed and equal for all APs, i.e., $\hat{s}_{ij} = \hat{s}_{nj} = \hat{s}_{wj} \in (0, 1]$. The physical meaning of such assumption is to assign the same amount of information to be transmitted from each AP to a specific user, which helps to compare the service capacity of each AP with respect to current user position (or respective SNR), and adjust the respective γ_j according to the required QoS_j^{\min} . Besides, at this stage, we seek for a reasonable way to compare the service each AP can afford for the same user, otherwise, if $\hat{s}_{ij} \neq \hat{s}_{nj}$ then the adjustments of $\{\gamma_j\}$ will be impracticable, e.g., an AP i with higher SNR than AP n could be randomly assigned with lower index $\hat{s}_{ij} < \hat{s}_{nj}$. Nevertheless, QPLB resolves the system in (30) using iterative bisection process described in the below Algorithm 2, which can be summarised by the search

$$\begin{aligned}
& \text{For each } i \text{ within } 1: I \\
& \quad n \text{ within } 1: N \\
& \quad w \text{ within } 1: W \\
& \quad \hat{s}_{ij}^* = \arg \max \eta_j^{(1)*}, \hat{s}_{nj}^* = \arg \max \eta_j^{(2)*}, \hat{s}_{wj}^* = \arg \max \eta_j^{(3)*},
\end{aligned} \tag{31}$$

and has complexity in the order of $O(I \times N \times W + (J \times \ln(\mu)))$, with $\mu > 0$ the optimality threshold set at each iteration of the bisection process.

Algorithm 2: QPLB implementation pseudo code (QoS- and Priority-aware Load Balancing)	
1:	Initialisation: <follow similar process as in Steps 1-8 of Algorithm 1 >
2:	for all user $j=1$ to J do
3:	set homogeneous $\hat{s}_{ij} = \hat{s}_{nj} = \hat{s}_{wj} \in (0,1], \forall i, n, w$
4:	set a feasible region of γ denoted as $[\underline{\gamma}_{inwj}^0, \overline{\gamma}_{inwj}^0]$ such that $ \begin{cases} f_j(\underline{\gamma}_j^0) < 0 \\ f_j(\overline{\gamma}_j^0) > 0 \end{cases} \text{ for all } j \in [1, J], $ with $\nu=0, \dots, N$ the iteration index
5:	update $\gamma_j^\nu = \frac{\gamma_j^\nu + \overline{\gamma}_j^\nu}{2}$, $ \underline{\gamma}_j^{\nu+1} = \begin{cases} \gamma_j^\nu & \text{if } f_j(\hat{s}_{ij}, \hat{s}_{nj}, \hat{s}_{wj}, \gamma) > 0 \\ \overline{\gamma}_j^\nu & \text{if } f_j(\hat{s}_{ij}, \hat{s}_{nj}, \hat{s}_{wj}, \gamma) < 0 \end{cases}, $ $ \overline{\gamma}_j^{\nu+1} = \begin{cases} \gamma_j^\nu & \text{if } f_j(\hat{s}_{ij}, \hat{s}_{nj}, \hat{s}_{wj}, \gamma) > 0 \\ \underline{\gamma}_j^\nu & \text{if } f_j(\hat{s}_{ij}, \hat{s}_{nj}, \hat{s}_{wj}, \gamma) < 0 \end{cases}, $ until $f_j(\hat{s}_{ij}, \hat{s}_{nj}, \hat{s}_{wj}, \gamma) < \frac{\mu}{J}$ with μ a very small number (optimality level)
6:	repeat bisection algorithm in Step 5 until we find a γ such that $ \sum_{j=1}^J f_j(\delta_{inw}, \gamma) ^2 < \mu $ (stopping criterion)
7:	end for
8:	for $i=1$ to I (VLC APs)
9:	compute $\eta_j^{(1)*}$ using (22)

10:	$\hat{s}_{ij}^* = \arg \max \eta_j^{(1)*}, \eta_j^{(1)*} = \begin{cases} \eta_j^{(1)*}, & \text{if } \hat{s}_{ij} = \hat{s}_{ij}^* \\ 0, & \text{otherwise} \end{cases}$ compute
11:	end for
12:	for $n=1$ to N (mmWave APs)
13:	compute $\eta_j^{(2)*}$ using (23)
14:	$\hat{s}_{nj}^* = \arg \max \eta_j^{(2)*}, \eta_j^{(2)*} = \begin{cases} \eta_j^{(2)*}, & \text{if } \hat{s}_{nj} = \hat{s}_{nj}^* \\ 0, & \text{otherwise} \end{cases}$ compute
15:	end for
16:	for $w=1$ to W (WiFi APs)
17:	compute $\eta_j^{(3)*}$ using (24)
18:	$\text{compute } \hat{s}_{wj}^* = \arg \max \eta_j^{(3)*}, \eta_j^{(3)*} = \begin{cases} \eta_j^{(3)*}, & \text{if } \hat{s}_{wj} = \hat{s}_{wj}^* \\ 0, & \text{otherwise} \end{cases}$
19:	end for
20:	for all user $j=1$ to J do (compute throughput of each user)
21:	compute $\hat{R}_j(\hat{s}_{ij}^*, \hat{s}_{nj}^*, \hat{s}_{wj}^*)$
22:	end for

In summary, we designed and resolved two load balancing schemes to define the allocation of specific APs to the home users: the first scheme is referred as POLB and follows the pure opportunistic logic similar to our empirical approach presented Section 4, while the second scheme is referred as QPLB and establishes AP allocations with user QoS- and priority-awareness. In next sub-Section we use simulations to evaluate the performance of the proposed algorithms and present simplified examples to discuss their practical significance.

4.3.3 Simulation settings

We consider a $15 \times 15 \times 3$ metres open office space, where one WiFi AP is placed on the ground at the centre of the office, and eight Remote Radio Light Heads (RRLHs) are fitted on the ceiling as shown in Figure 63. Each RRLH includes one LED and one mmWave antenna. Each LED and mmWave antenna acts as a VLC AP and mmWave AP, respectively, covering a confined area, while the WiFi AP provides coverage for the entire room. Recall that in our study we consider that the VLC APs operate at different frequencies and the mmWave beams are directed at fixed points, so as, they do not interfere with each other.

For the user positioning and to study the issue of VLC/mmWave channel blockage, we assume the users are uniformly distributed within the office and that they are immobile at each interval, which lasts 1.5 sec, i.e., during the load balancing decision processing. Between time intervals, i.e., during load balancing processing time, users have random speeds that are uniformly distributed between 0 and 2 metres per second. During this time, the direction of the movement of each user changes randomly at every interval where the load balancing takes place and it is uniformly distributed between 0 and 2π .

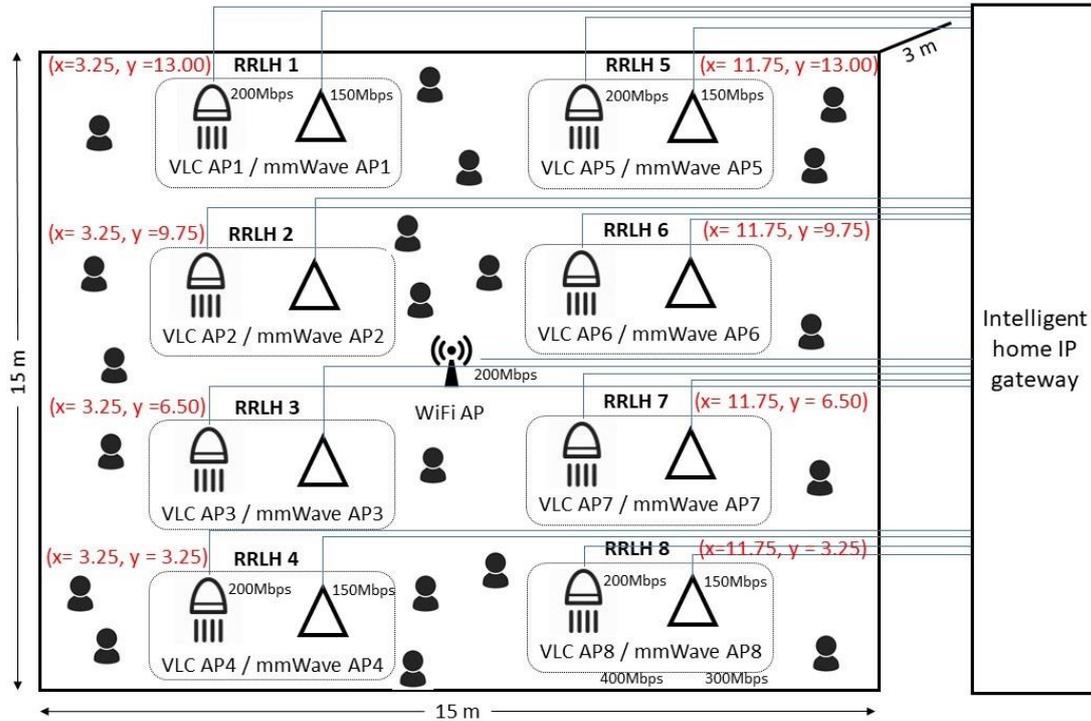


Figure 63 - Representation of the hybrid VLC/mmWave/WiFi system in the considered indoor simulation environment

The RF channel has been considered as in [64] and [65], and by using the Matlab’s reference software library for WLAN channels available in [66] we modelled channel propagation and shadowing effects, which facilitates calculating (15)-(16) and obtaining the channel gain G_j^{WiFi} of each user using (17). Having defined G_j^{WiFi} and by setting the WiFi noise power density $N^{WiFi} = -174$ dBm/Hz, WiFi carrier bandwidth $BW^{WiFi} / N = 20$ MHz, WiFi transmitted power $P^{WiFi} / I = 20$ dBm, we have calculated the SNR_j^{WiFi} in (14) as a function dependent on current user positioning, e.g., specified by variables $\{d_{wj}\}, \forall w, j$.

For the VLC channel, we assume that the source of emission and the reflected points on wall have a Lambertian¹ radiation pattern of [1, 6, 7], and we set the parameters specified in Table 6 to calculate equations (6)-(10) and derive the SNR_{ij}^{VLC} in (11) as function of user positioning with respect to the VLC APs, e.g., specified by variables $\{d_{ix}\}$ and $\{d_{xj}\}$, $\forall i, j$. To survey the illuminance distribution and the corresponding SNR of VLC APs, we assume four VLC APs situated on the ceiling of the office, where each VLC AP includes two LEDs located at the position like in Figure 63. Based on our simulation settings and the VLC channel modelling in Section 4.1.1.1, the illuminance and the SNR performances with respect to vertical and horizontal user coordinates are illustrated in Figure 64.

Table 7 - VLC channel simulation settings

VLC channel simulation settings			
Parameter/Symbol	Value	Parameter/Symbol	<u>Value</u>
Half intensity radiation angle ($\Phi_{1/2}$)	30 deg.	Physical area of LEDs (A^{PD})	<u>1 cm²</u>
Optical filter gains (g^f, g^c)	<u>1.0, 0.7</u>	Wall reflectivity (L)	<u>0.8</u>
Angle of irradiance (α_{ij})	<u>40 deg.</u>	Angles of radiance for first and second segment ($\varphi_{ix}, \varphi_{xj}$)	<u>35 deg.</u> <u>50 deg.</u>
Maximum angle of incidence (B^{\max})	90 deg.	Angles of incidence for first and second segment (θ_{ix}, θ_{xj})	<u>15 deg.</u> <u>22 deg.</u>
Number of LEDs (L)	<u>1</u>	Optical to electric power conversion coefficient ζ	<u>0.9</u>
Detector responsivity (R)	<u>0.53 A/W</u>	Transmitted optical power (P)	<u>35 dBm</u>
Bandwidth per LED (BW^{VLC} / L)	<u>20 MHz</u>	PSD of noise (N^{VLC})	10–21 A ² /Hz

¹ Recall that the Lambertian emission means that the light intensity emitted from the source has a cosine dependence on the angle of emission with respect to the surface normal.

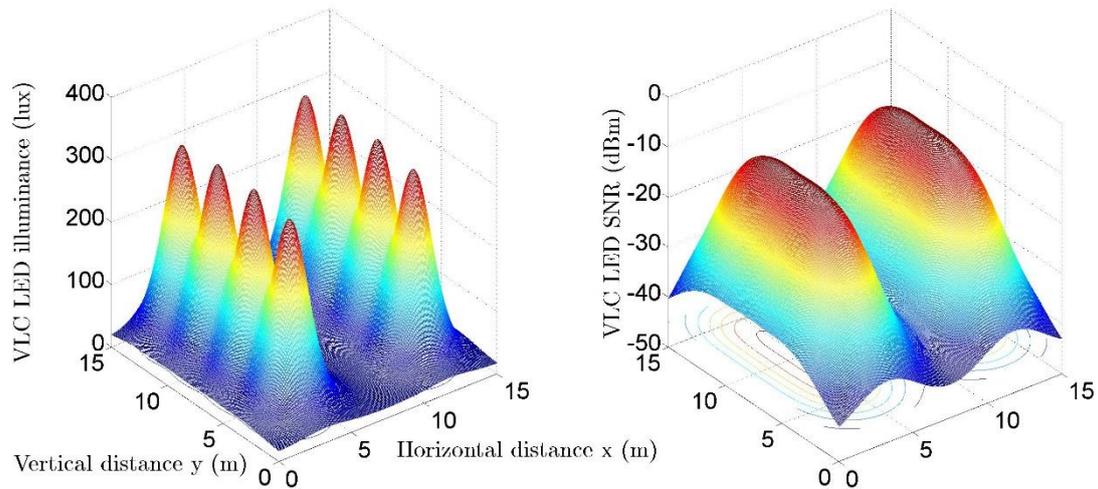


Figure 64 - Illustration of the VLC AP illuminance (left) and SNR (right) versus the vertical and horizontal distance of a user from the considered VLC AP based on the VLC channel modelling in Section 6.3.1.1 and the simulation settings in Table 6.

For the mmWave channel, we followed the Matlab’s reference software library for spatial and millimetre-wave channels available at [67], which helps to define the parameters related to equations (13) and (14), which are specified in Table 6 of Section 4.1.1.2 and the below. To survey the power density and the corresponding SNR of mmWave APs, we assume four mmWave APs situated on the ceiling of the office, where each mmWave AP has four beams transceivers located at the position like in Figure 63. To survey the power density and the corresponding SNR of mmWave APs, we assume four mmWave APs situated on the ceiling of the office, where each mmWave AP has four beams transceivers located at the position like in Figure 63. Based on our simulation settings and the mmWave channel modelling in Section 4.1.1.2, the illuminance and the SNR performances with respect to vertical and horizontal user coordinates are illustrated in Figure 65.

Table 8 - parameters used for simulation of mmWave channel

parameters used for simulation of mmWave channel	
Parameter/Symbol	Value
Bandwidth per beam (BW^{mmWave} / N)	20 MHz
AP transmit power (p_i)	40 dBm
Noise power density (N_0^{mmWave})	-174 dBm/Hz
Propagation environment density (λ)	0.01
Transceiver antenna length (l)	0.005 m to 0.1 m

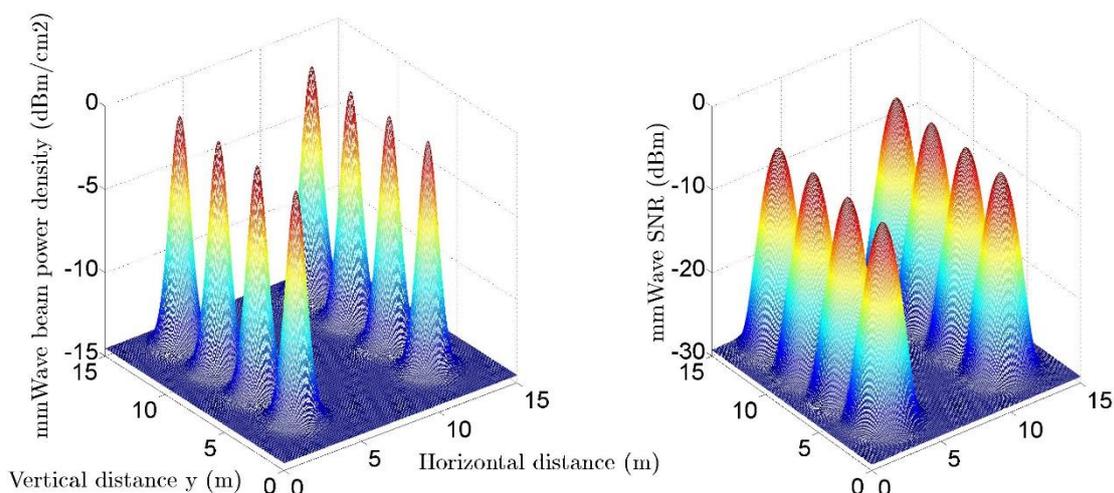


Figure 65 - Illustration of the mmWave beam power density (left) and SNR (right) versus the vertical and horizontal distance of a user from the considered mmWave AP based on the mmWave channel modelling in Section 6.3.1.2 and the simulation settings in Table 6 and Table 8

4.3.4 Performance evaluations and implementation examples

Let us consider that our hybrid network has to serve 20 users within the office, each one having different priorities and minimum QoS requirements, specified in Table 9.

Table 9 - The user priorities and minimum QoS requirements used during evaluation

The user priorities and minimum QoS requirements used during evaluation			
User index (j)	User priority (q_j)	User minimum QoS requirement in Mbit (QoS_j^{\min})	User coordinates at first time interval in meters (x, y)
1	0.02 (Class 3)	50	1.7882 11.5797
2	0.02 (Class 3)	50	14.0974 13.9928
3	0.02 (Class 3)	50	9.6833 14.5911
4	0.02 (Class 3)	50	7.1919 2.8804
5	0.02 (Class 3)	50	9.5898 2.0831
6	0.02 (Class 3)	50	8.1707 10.4440
7	0.02 (Class 3)	50	9.7097 1.4073
8	0.055 (Class 2)	100	8.1583 7.8811

9	0.055 (Class 2)	100	10.8157	7.9552
10	0.055 (Class 2)	100	7.8374	12.9171
11	0.055 (Class 2)	100	14.9056	7.2728
12	0.055 (Class 2)	100	3.2801	5.9018
13	0.055 (Class 2)	100	1.5870	10.0715
14	0.055 (Class 2)	100	1.6455	11.1189
15	0.079 (Class 1)	150	0.9539	7.8008
16	0.079 (Class 1)	150	6.0687	5.2157
17	0.079 (Class 1)	150	6.7256	2.2500
18	0.079 (Class 1)	150	11.4526	3.9322
19	0.079 (Class 1)	150	5.4872	8.7914
20	0.079 (Class 1)	150	9.4184	0.6668

Given the users' positioning at first time instance, the normalised channel gains (in dBm) with respect to the VLC and mmWave AP (i.e. the positions of RRLHs considered in Figure 63) are calculated as shown in the below Table 10, while the normalised channel gains of the WiFi AP vary between -11 and -13 dBm and are omitted due to space limit, e.g., generate a 4096 x 20 matrix as we have considered each carrier as a WiFi AP.

Table 10 - Normalised channel gains of the VLC (left) and mmWave (right) APs (in dBm) considering the positioning of 20 users given in Table 9 and the positioning of the RRLHs shown in Figure 63

user	VLC_LED									mmWave beam								
	1	-27.7686	-19.2391	-23.0518	-8.6946	-12.5056	-27.3264	-30.6821	-29.5015	-25.4980	-26.1621	-26.8717	-8.3721	-6.5554	-24.4159	-31.1904	-30.1303	
2	-23.0695	-37.8411	-34.3244	-25.7840	-23.1976	-5.5642	-20.3010	-12.5079	-8.4179	-23.0255	-20.7977	-18.2984	-27.4373	-9.8153	-12.9510	-6.4336		
3	-16.4236	-13.9400	-37.2277	-14.8955	-39.6215	-36.3584	-18.2944	-28.7967	-31.3569	-28.1671	-18.1624	-23.0756	-25.5456	-22.6891	-17.2745	-23.6193		
4	-28.1622	-23.6607	-22.4380	-10.4328	-22.1470	-23.0381	-39.2880	-24.5052	-27.4579	-3.5658	-21.5152	-30.6930	-10.1249	-5.9549	-5.6274	-2.6185		
5	-9.1148	-38.4164	-14.7164	-5.0231	-29.3206	-24.0583	-18.9648	-34.7738	-10.5194	-8.5904	-23.7989	-15.3363	-9.8343	-15.7050	-29.4730	-22.4444		
6	-10.1736	-14.3712	-27.8382	-14.9258	-19.3998	-26.2255	-20.4065	-24.5498	-15.9861	-15.7893	-4.8347	-4.5619	-11.6307	-7.7271	-6.0643	-19.9589		
7	-5.6918	-13.7530	-37.1055	-24.2784	-6.2172	-31.6153	-10.4883	-36.5621	-11.7696	-30.7560	-8.3272	-13.7202	-26.8869	-16.3720	-11.9635	-15.4209		
8	-38.7502	-37.4685	-22.8444	-35.4816	-15.2491	-34.9371	-6.4112	-19.6831	-20.9062	-25.7014	-7.4309	-20.8835	-26.6655	-19.5513	-28.9244	-25.8874		
9	-38.9631	-7.4004	-39.0967	-6.4789	-33.0505	-18.4004	-15.9922	-17.5345	-8.9090	-15.5562	-10.8291	-25.8854	-19.1205	-7.2009	-16.9895	-11.5355		
10	-9.3351	-15.4790	-11.9048	-36.6653	-17.1276	-7.9627	-36.3075	-22.1146	-19.3965	-12.0028	-6.3974	-22.7357	-19.1549	-26.6698	-20.4586	-4.1064		
11	-21.3594	-25.7054	-4.8915	-9.5841	-7.9161	-30.6856	-13.6458	-13.9587	-20.0947	-3.7729	-29.7888	-12.3592	-10.5805	-23.6871	-19.4940	-5.0784		
12	-27.9843	-12.1155	-5.4074	-34.1805	-22.8887	-16.6196	-15.8738	-37.5209	-19.9964	-24.2272	-5.9454	-30.4045	-22.9740	-29.7757	-22.9476	-6.0802		
13	-21.1565	-27.2559	-18.8746	-33.0164	-17.8392	-34.3912	-19.3104	-5.3365	-15.4528	-17.2038	-19.4328	-17.6057	-25.8877	-16.7792	-6.8800	-25.6667		
14	-7.4088	-32.9430	-19.7264	-37.1258	-30.8818	-18.0103	-29.7804	-13.8903	-8.0627	-7.9978	-21.1145	-30.6144	-15.2477	-21.6465	-17.8365	-9.7720		
15	-24.5341	-22.5595	-15.9016	-9.8056	-23.3659	-33.9875	-10.0273	-20.0926	-17.4144	-14.8158	-21.5381	-4.2079	-15.3865	-28.7037	-2.9622	-9.1361		
16	-9.9727	-7.7784	-29.2845	-22.6656	-33.1582	-11.1782	-35.4963	-29.5886	-14.2319	-7.0607	-27.9387	-8.2110	-15.9699	-18.1558	-26.8143	-9.3094		
17	-14.3518	-32.8584	-8.2155	-19.1735	-33.5919	-9.5332	-7.9105	-19.0815	-15.2414	-24.5508	-16.6786	-25.2508	-10.3712	-10.4662	-12.1996	-15.1447		
18	-31.3808	-5.1681	-19.0731	-11.0750	-11.6315	-35.7959	-21.1608	-11.3644	-15.7963	-13.0505	-26.5057	-29.4256	-22.2613	-31.2787	-27.4013	-27.7506		
19	-36.4432	-26.0294	-15.3313	-25.1314	-9.3293	-6.5428	-6.0636	-34.9688	-15.5265	-30.2545	-17.8677	-25.4765	-29.1099	-3.0928	-9.3821	-22.9460		
20	-30.4955	-8.9362	-15.7274	-26.2176	-33.7349	-29.0352	-31.3620	-25.4470	-18.5342	-9.5152	-22.8305	-10.8660	-29.2558	-11.7873	-19.4447	-24.0123		

Table 11 - The optimal AP allocation solution of the POLB Algorithm 1 (left) and QPLB Algorithm 2 (right). Users highlighted in red are assigned to the WiFi network.

POLB - Algorithm 1			QPLB - Algorithm 2		
	VLC LED	mmWave beam		VLC LED	mmWave beam
1	00001000	00000000	1	00000000	00001000
2	00000100	00000000	2	00000000	00000000
3	00000000	00000000	3	00000000	00000000
4	00000000	00000001	4	00000000	00000100
5	00010000	00000000	5	00000000	00000000
6	00000000	00010000	6	00000000	00010000
7	10000000	00000000	7	00000000	00000000
8	00000010	00000000	8	00000010	00000000
9	00000000	00000100	9	00000000	00000001
10	00000000	00000000	10	00000100	00000000
11	00100000	00000000	11	00000000	01000000
12	00000000	00000000	12	00000000	00100000
13	00000001	00000000	13	00000000	00000010
14	00000000	10000000	14	00000000	10000000
15	00000000	00000010	15	00010000	00000000
16	00000000	01000000	16	10000000	00000000
17	00000000	00001000	17	00100000	00000000
18	01000000	00000000	18	00000001	00000000
19	00000000	00000100	19	00001000	00000000
20	00000000	00000000	20	01000000	00000000

Similarly, we calculate the normalised channel gains at second time instance, third time instance, and so on, which facilitates deriving the optimal AP allocation solution of the POLB Algorithm 1 and QPLB Algorithm 2 shown in Table 11. As observed from Table 11, POLB allocates the AP with the maximum channel gain corresponded to each user, while the AP allocations in QPLB follow a fairer rational according to users' Classes. For instance, in POLB the Class 1 user 20 (i.e. with high-priority and $QoS_{20}^{\min} = 150$ Mbps QoS) and the Class 2 users 12 and 10 (i.e. with medium priority and $QoS_{20}^{\min} = 100$ Mbps) are assigned to the WiFi network meaning that their requirements are not fulfilled. In contrast, all low-priority Class 3 users (except user 3) have VLC or mmWave access, which however is suboptimal as these users occupy high-speed APs that could be used for serving the highly-demanded users 11, 12 and 20. On the other hand, QPLB assigns the VLC APs to the Class 1 and Class 2 users, the mmWave APs to the Class 2 users and the residual mmWave APs to the Class 3 users. The QPLB allocation is fairer since only the Class 3 users are assigned to the WiFi, which is enough to serve their requirements. However, due to the fact that Class 3 users may have higher channel gains than Class 1 and Class 2 users, QPLB is likely to slightly decrease the overall system throughput compared to POLB, which we examine in next sub-Section.

4.3.4.1 Overall throughput and fairness performance

The difference between the allocation rational in Table 11 is because POLB aims to opportunistically maximise the overall system throughput without considering the individual user priorities and minimum QoS requirements, while these user characteristics constraint

QPLB to decide the AP allocation in a fairer manner. For instance, let us observe at the left-hand side of Figure 66 the overall system throughput versus the time intervals issued by each algorithm. When all users have same priorities and no minimum QoS requirements and by setting $thr1=0.75$ Gbit and $thr2=1.5$ Gbit, POLB results to higher throughput performance than QPLB of approximately 0.3 Gbit. For instance, upon considering either the user priorities or the minimum user QoS specified in Table 9, the overall throughput of QPLB is 2.7 Gbit in average, while upon considering both the individual user priorities and minimum QoS, the overall throughput of QPLB decreases to 2.5 Gbit in average. This is because at each time interval some high priority Class 1 users and/or some users with high QoS_j^{min} are placed in dead/shadowing areas, where either the VLC or the mmWave or even both APs have particularly low SNR (i.e. LED illuminance and/or beam power density are low). However, although its decreased throughput performance, the priority and QoS consideration in QPLB enables it to decide the AP allocation at each time interval in a fairer manner compared to POLB. This can be seen at the right-hand side of Figure 66, where we plot the Jain's fairness index (FI) at each time interval defined as $FI = \left(\sum_{j=1}^J R_j \right)^2 / J \cdot \sum_{j=1}^J R_j^2$, which is considered as

convenient metric to rate the fairness over a set of given values [68]. From the graph we see that all three versions of QPLB result to FI higher than 0.9, which, in physical terms, indicates the perception of QPLB for accounting the individual user priorities and QoS demands. In contrast, the FI in POLB varies between 0.85 and 0.7, which indicates that fairness in POLB occurs rather asymptotically than systematically as in QPLB.

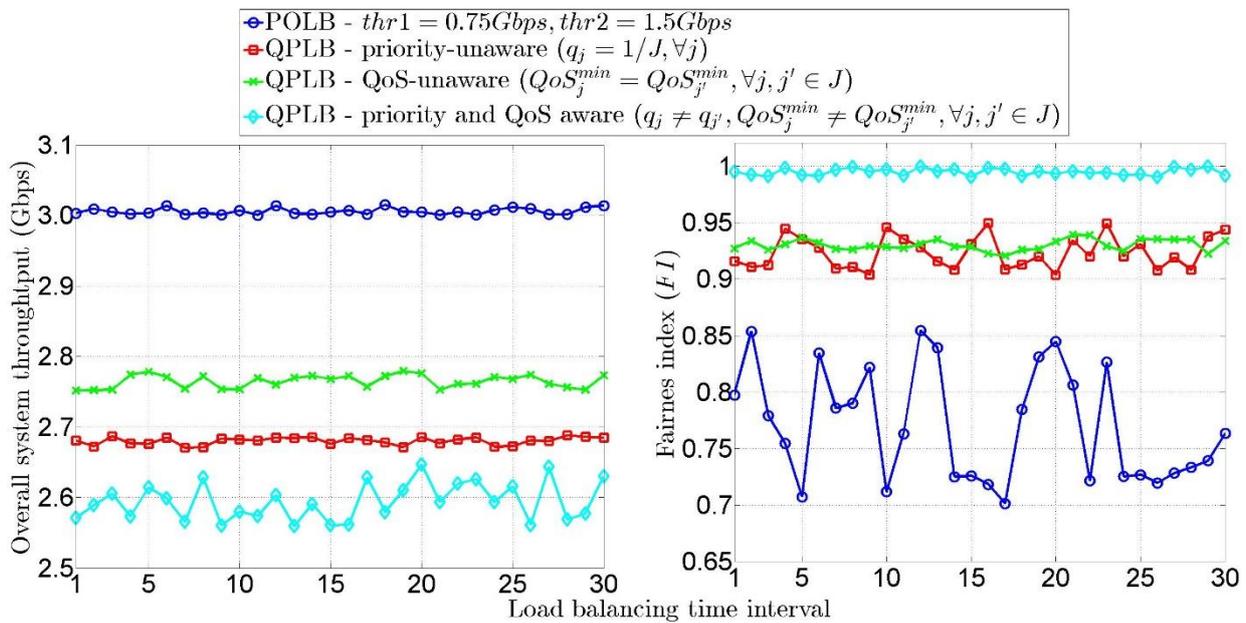


Figure 66 - Illustration of (i) the overall system throughput (left), and (ii) fairness index (right) versus the time intervals considering and for POLB and three version of QPLB: priority-unaware, QoS-unaware and priority and QoS aware.

4.3.4.2 Individual user throughput and implementation complexity

To gain a deeper insight for the individual user throughput performance, the left-hand side of Figure 67 plots the individual throughput assigned to three different Class users versus the number of users that join the system. Intuitively, the more users join the system, the more resources (APs) need to be combined to support the user demands. Recall that the Class 1

user has higher QoS requirements than the Class 2 and Class 3 users according to the settings in Table 9. From the figure we see that when resources are in surplus (e.g. $4 \leq J \leq 14$) all users are assigned with almost identical throughput, while when resources become limited (due to the many users that joined the system, e.g., $14 \leq J \leq 20$) users are assigned with at least their minimum QoS requirement. Furthermore, in the right-hand side of Figure 67 we see that the corresponding fairness index per user Class is almost similar among users of different Classes, with 0.011 deviation at maximum. For example, the FI for Class 1 users varies between 1.01 to 0.99, the FI of Class 2 users between 0.995 to 0.985 and the FI of Class 3 users between 1.001 to 0.98.

In contrast, we see at the left-hand side of Figure 68 that the individual user throughput performance achieved by POLB is less fair than QPLB as there are instances where Class 3 users have higher throughput than Class 1 and Class 2. Besides, for $14 \leq J \leq 20$ the Class 1 and Class 2 users are assigned with less than their minimum QoS requirements due to the QoS and user priority unawareness of POLB.

In consequence, QPLB facilitates a more convenient way for allocating APs of different networks to system users according to users' minimum throughput demands, priorities and positioning compared to POLB. Recall that the rationale of POLB is considered as reference logic for allocating APs adopted by most relevant literature, whereas QPLB-type algorithms have yet been examined although their potentials to improve the system reliability.

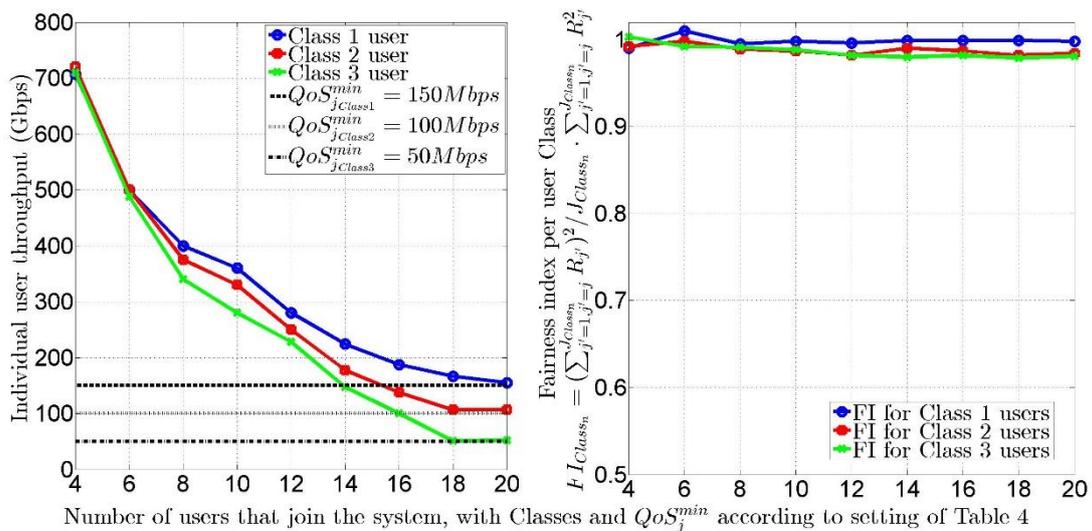


Figure 67 - Illustration of (i) the individual user throughput (left), and (ii) fairness index per user Class (right) versus the time intervals achieved via priority-and-QoS aware QPLB algorithm.

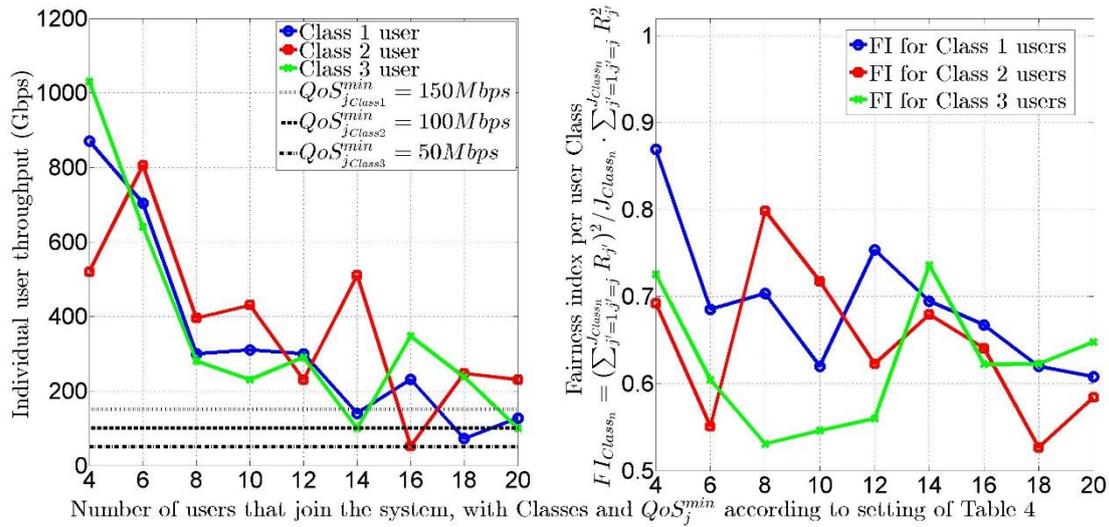


Figure 68 - Illustration of (i) the individual user throughput (left), and (ii) fairness index per user Class (right) versus the time intervals achieved via pure opportunistic POLB algorithm.

The price, however, of the “fairer” AP allocation of QPLB is an increase at its implementation complexity as observed in Figure 69. At the left-hand side of the figure we see that QPLB requires about 10 times more iterations than POLB to derive its optimal solution in (31). At the right-hand side of the figure we see the corresponding time needed for each algorithm to derive its optimal solution considering that each iteration lasts 50ms in time. That is, POLB requires less than 1 sec to converge, while QPLB requires 6 to 9 sec. In view of such results, our viewpoint is that although QPLB’s implementation time is up to 9 times higher than POLB, (i) it is not prohibited to be applied in practice for AP allocation, especially considering the vast computing capacity available at L3, which is likely to implement QPLB in much lower time, and (ii) it is more practicable than POLB due to its QoS and user priority awareness.

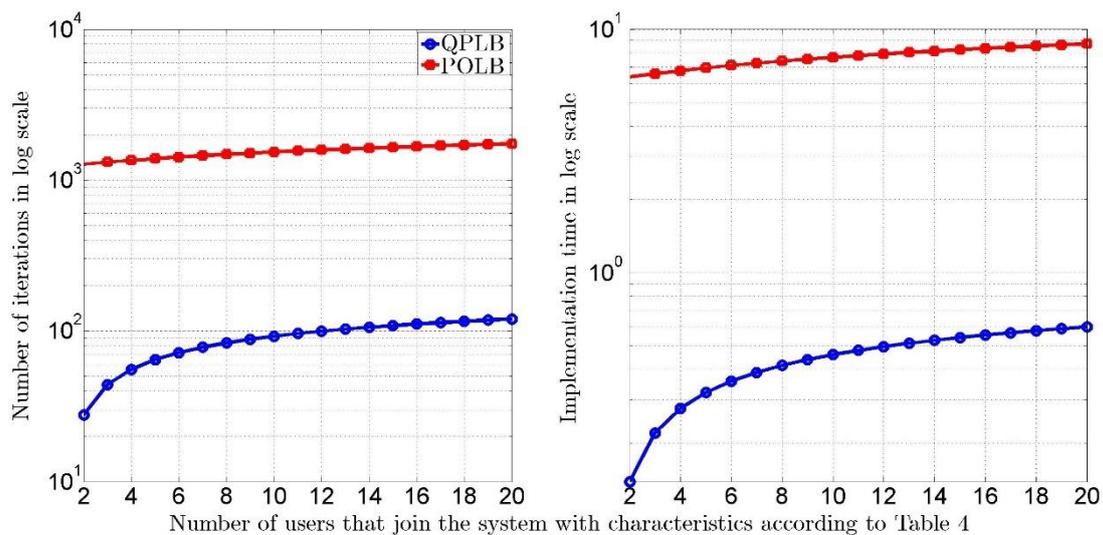


Figure 69 - Illustration of the implementation complexity: number of iterations and implementation time versus the number of users joining the system.

4.3.4.3 Implementation example

In continue, we provide an example to better clarify the AP allocation rational along with the throughput and fairness performance of QPLB. We assume that one VLC AP, two mmWave APs and a WiFi AP are available at the office, where three users of different classes change their positions between the time intervals n and $n+1$ as shown in Figure 70 from left to right. Also, we consider that the WiFi AP can be assigned to one user only and not shared between users.

During QPLB allocation, at time interval n , the Class 1 user is likely to be assigned with the mmWave AP1 and the VLC AP, although the VLC AP is in NLoS with respect to this user and in LoS with respect to Class 2 user. Similarly, the Class 2 user is likely to be serviced by mmWave AP2, although the mmWave AP2 is in NLoS with the Class 2 user and in LoS with the Class 3 user. At time interval $n+1$, i.e., when users change positions, the Class 1 user is assigned with the NLoS VLC AP3 and LoS mmWave AP2, the Class 2 user with the LoS mmWave AP1 and the Class 3 user with the WiFi.

In contrast, during POLB allocation, users are likely to be assigned with the APs that are in LoS with respect to their positioning. These are the APs with the highest SNR level for each respective user positioning (i.e. pure opportunistic), which although results to higher throughput performance (as discussed previously in Figure 66) it is user QoS and priority unaware, thus, less fair and practicable than QPLB.

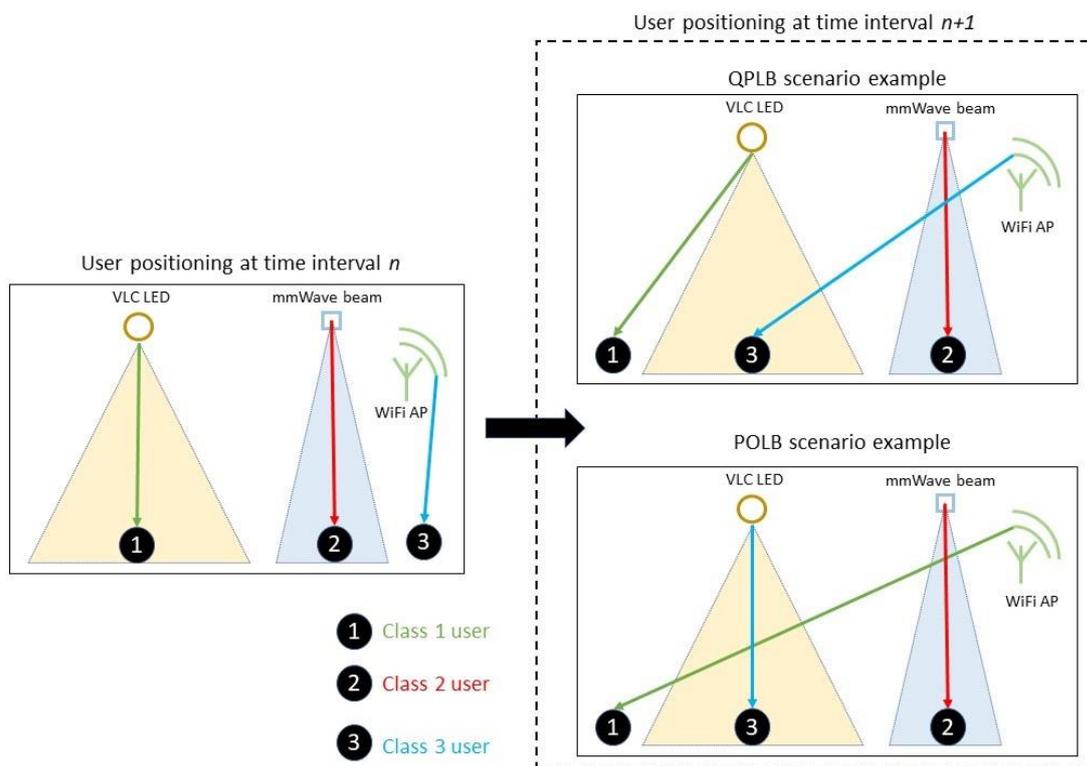


Figure 70 - Representation of QPLB and POLB scenario examples to clarify the AP allocation rational of each algorithm.

5 Network Slicing in IoRL Platform

5.1 Network slicing overview

Network slicing is a 5G cutting edge technology, that enables the creation of multiple virtual networks on top of a physical architecture, allowing operators to provide portions of their networks that fit with the requirements by different vertical industries, such as automotive, energy, food and agriculture, city management, government, healthcare, manufacturing, public transportation, and many more. This form of virtual network architecture combines the principles behind two closely related network virtualization technologies, Software Defined Networking (SDN) and Network Function Virtualization (NFV), allowing for better network flexibility and moving modern networks towards software-based automation.

This section describes the current state of the art on 3GPP specifications related to network slicing. Network slicing support is part of the new SBA specified by 3GPP in TS 23.501 [69]. The definition of network slicing provided by the 3GPP proposes that “the network slice is a complete logical network (providing Telecommunication Services and Network Capabilities) including Access Network (AN) and Core Network (CN).”

The management and orchestration of network slices, specified in 3GPP TR 28.801 [70], also defines the concept of Network Slice Instance (NSI). NSI is a managed entity in the operator’s network that includes all functionalities and resources necessary to support a certain set of communication services, with lifecycle independent of the lifecycle of the service instances, as the service instances are not necessarily active through the whole duration of the NSI. An NSI is composed by NFs and the connectivity between them is described by the Network Slice Template (NST).

Network Slice lifecycle can be described in four phases, as depicted in Figure 71:

- i. Preparation Phase: Includes the creation, validation and the on-boarding of the NST, as well as the preparation of the network environment.
- ii. Instantiation, Configuration and Activation phase: Includes the resource provisioning process and the instantiation, configuration and activation of the network functions and components that are part of network slice.
- iii. Run-time Phase: Includes traffic handling, reporting and activities related to modification, such as upgrade, reconfiguration, scaling, etc.
- iv. Decommissioning phase: Includes the deactivation of the NSI and the release of network resources that were used as part of that.

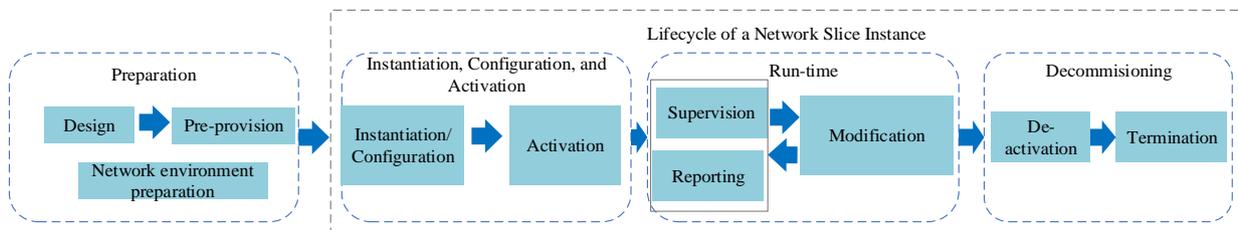


Figure 71 - Representation of the lifecycle of network slice instance

The list of Standard Slice Types (SST) specified by 3GPP in TS 23.501 is shown in Table 12.

Table 12 - Standardised SST values

Standardised SST values		
Slice/Service type	SST value	Characteristics
eMBB	1	Slice suitable for the handling of 5G enhanced Mobile Broadband.
URLLC	2	Slice suitable for the handling of ultra- reliable low latency communications.
MIoT	3	Slice suitable for the handling of massive IoT.

Recently, ETSI NFV provided a report [71], analyzing how the defined network slicing use cases could be related to the current NFV concepts and supported by the ETSI NFV architectural framework and by NFV-MANO. As a result, they concluded to the mapping between the 3GPP and ETSI NFV Information Models depicted in Figure 72.

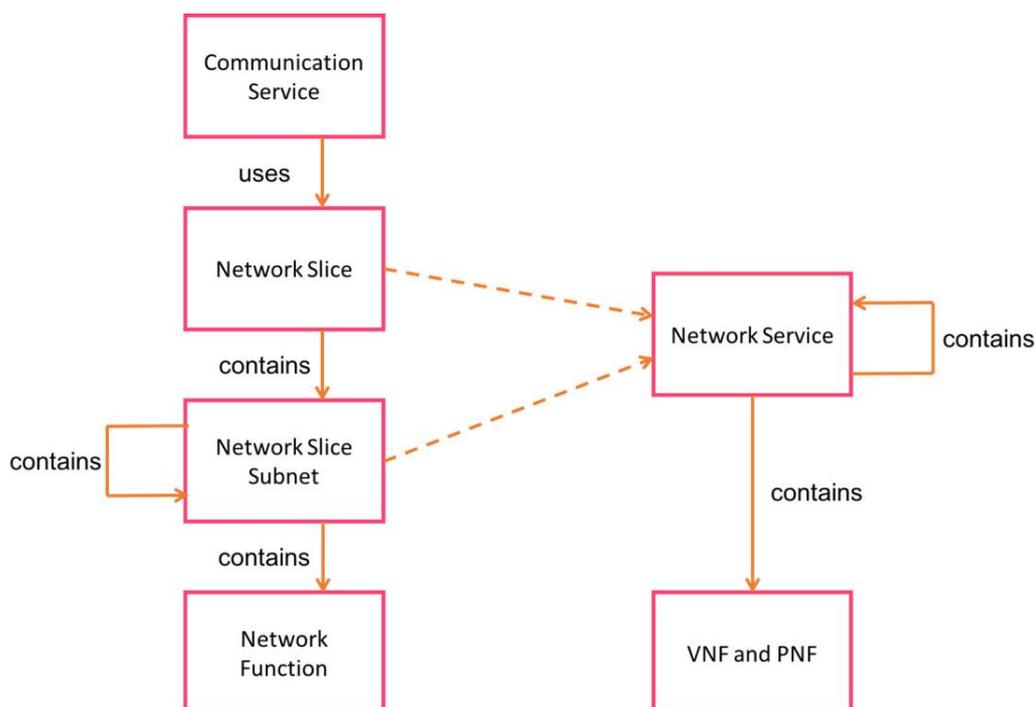


Figure 72 - Representation of the relating the information models between 3GPP and ETSI NFV

In the same report, Figure 73 defines the 3GPP Slice related network functions, namely Communication Service Management Function (CSMF), Network Slice Management Function (NSMF), and Network Slice Subnet Management Function (NSSMF), in an NFV framework, as well as the reference point that can be used for the interaction between these functions and NFV-MANO.

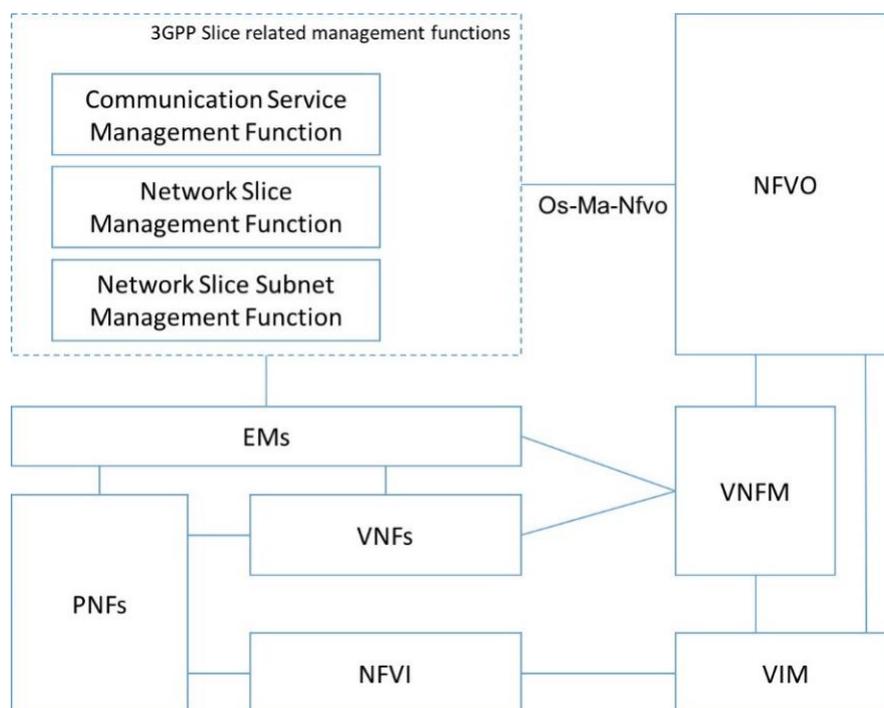


Figure 73 - Illustration of the network slice management process in an NFV framework

5.2 Slicing use case scenario on the IoRL platform

A network slice can be described as the sum of various sub-slices of different network domains, such as the WAN, the Core Cloud and the Edge Cloud. Taking into consideration the IoRL use cases and architecture, as they are described in the previous IoRL deliverables D2.1 [72] and D2.2 [73], we can make the assumption that the IoRL platform, being so close to (or even part of) the user premises, stands in the edge domain of a wider network. Therefore, the SDN/NFV can be used as the edge cloud domain for network operators, in order for them to deploy network services.

The SDN/NFV platform, using OpenStack as VIM, can support multiple tenants with complete isolation of compute resources, networks, namespaces, etc. Moreover, the SDN switch supports the creation of different OpenFlow rules for each tenant/slice, each having different QoS parameters and traffic steering policies. Finally, regarding the IoRL RAN, using the mapping parameters between the SDN/NFV platform and the 5G base station protocol processing server, as described in section 3.2:L3-L2 interface, we can assign different QoS parameters to each slice, based on the requirements.

Taking advantage of these features, IoRL can offer an edge domain with shared resources that the network operators can use to deploy their services.

5.3 Review of slicing implementation technologies to be implemented into the IoRL platform

The first step towards network slicing is separating the control plane from the data plane. The data plane consists of various customizable elements that can be chained together programmatically to provide connectivity, with each element running on top of a shared physical infrastructure. The slice view will be provided and controlled from a central software component, part of the control plane, the Slice Manager. The Slice Manager controls all the

devices comprising the network for which it is responsible, while it provides an interface for creating, modifying, monitoring and deleting network slices.

The network slice is defined using the Network Service Template (NST), a descriptor where the network slice characteristics are described, such as the list of NFV components (Network Services) that need to be instantiated, WAN configuration, QoS, monitoring level, Life-Cycle stages, etc. The NST must be written in a human/machine readable language, such as TOSCA or YAML. Based on the on-boarded NST, Slice Manager has to make the mapping between the available data plane resources and the described slice requirements.

Another important topic of network slicing is the network slice Life-Cycle Management (LCM), as it provides the ability to the network slice to be agile, facilitating the creation of slices. During lifetime network slices can be scaled out or scaled in, in case more or less resources and performance is required, respectively. The slice scaling policies are either defined in the NST that is on-boarded on the Slice Manager or monitoring data can be used by the Slice Manager to take autonomous decisions.

In the IoRL platform, there can be two possible solutions regarding the implementations of the Slice Manager. (i) Use the feature of network slicing offered by the OSM that is already installed on the system, as described previously in Section 2.4.2, and 2.5 (ii) create an autonomous component that will be responsible for creating, modifying and deleting network slices. Next, we will go through the technical details of each solution.

5.3.1 Using OSM for network slicing

In the latest release FIVE, OSM supports the End-to-End orchestration of a network slice, as depicted in Figure 74.

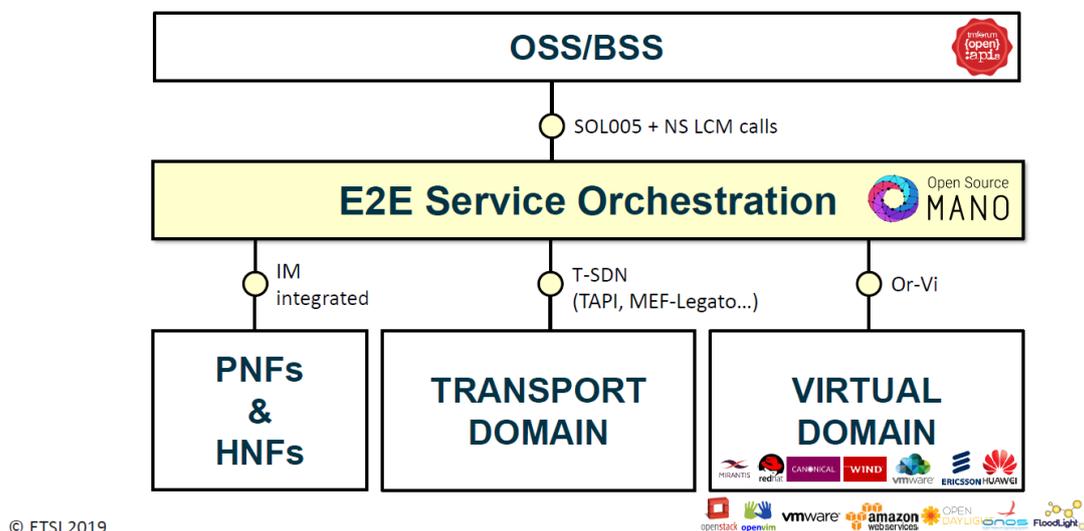


Figure 74 - Illustration of the OSM E2E orchestration

In a Proof-of-Concept demonstration during the 5th OSM Hackfest [72][73], OSM was used (i) for the creation of slices with different characteristics across different domains in the mobile network including radio, core and transport and (ii) as a central management unit of the newly created slices. In more details, the demonstration shows first how a slice for enhanced Mobile Broadband (eMBB, that might apply for instance for future 360° video delivery) can be created on demand and, in turn, how a second slice for Ultra-reliable Low-latency

Communications (URLLC, applicable to cases like real-time remote machine control) is created following the same procedure.

5.3.2 Using slice manager component

Adopting similar solutions with the ones that were used in other H2020 project, such as 5GTANGO, MATILDA and 5GENESIS, a new component, the Slice Manager, will be created and hosted as a Management and Orchestration layer component for the IoRL platform. The Slice Manager architecture, internal components as well as interfaces with other components, as envisioned for the IoRL platform, are depicted in Figure 75.

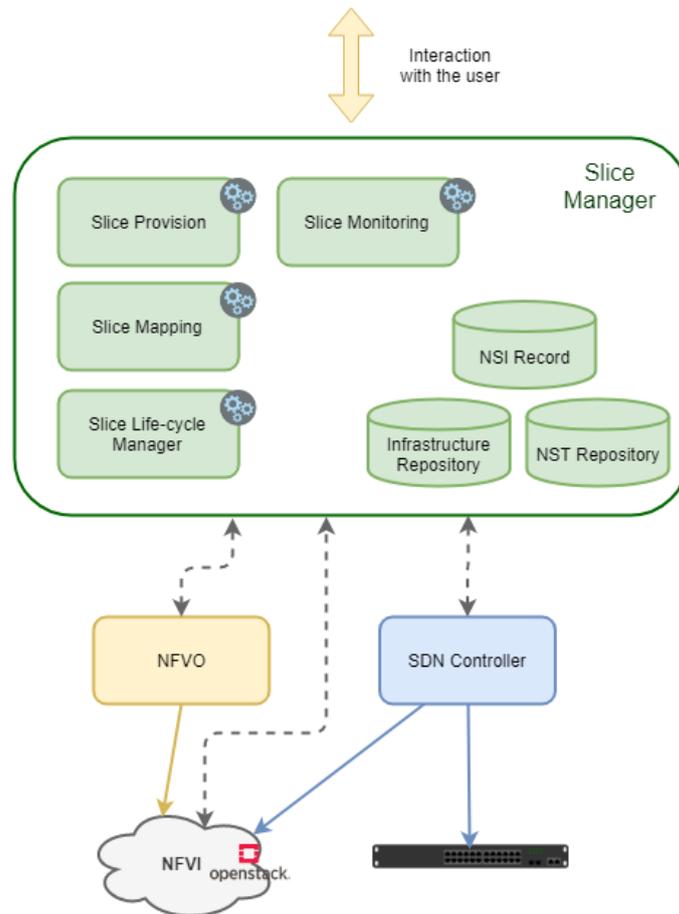


Figure 75 - Illustration of the envisioned slice management IoRL architecture

Through the NBI, the Slice Manager interacts with the user, e.g. the network operator. It receives the NST for creating network slices and provides the API for managing and monitoring them. Through the South Bound Interface (SBI), it communicates with the other components of the platform, namely the VIM, the NFVO and the SDN controller, in order to manage the functions in the network.

Slice Manager will operate in three phases for the creation of a new slice: (i) Resource Provisioning, (ii) Network Service Instantiation and (iii) Slice Activation. An example of the workflow for the instantiation and the configuration of a network slice is the following, the sequence diagram of which is depicted in Figure 76:

-
- The network operator requests the creation of a new slice using Slice Manager's NBI, selecting a particular slice profile, in order to deploy the required Communication Service (i.e. comprising of a number of NS plus, as described in sub-Section 5.2). The Network Slice Template is created parsed by the Slice Manager.
 - Slice Manager returns a slice id to the operator, for further management and monitoring purposes.
 - Resource Provisioning:
 - Slice Manager communicates with the VIM and the SDN Controller, in order to provision resources (sub-network slices).
 - VIM creates a new tenant for the newly created slice.
 - Slice Manager communicates with the NFVO in order to register the new tenant.
 - SDN controller creates virtual links or/and flows on SDN switches with specific resource-QoS requirements, as declared in the NST, in order to activate appropriate traffic steering for a particular slice.
 - Network Service Instantiation:
 - Slice Manager communicates with the NFVO in order to make the deployment and instantiation of the Network Services included in the Communication Service.
 - Slice Activation:
 - Slice Manager communicates with the SDN Controller in order to activate the SDN flows created for this slice in the Resource Provisioning phase.

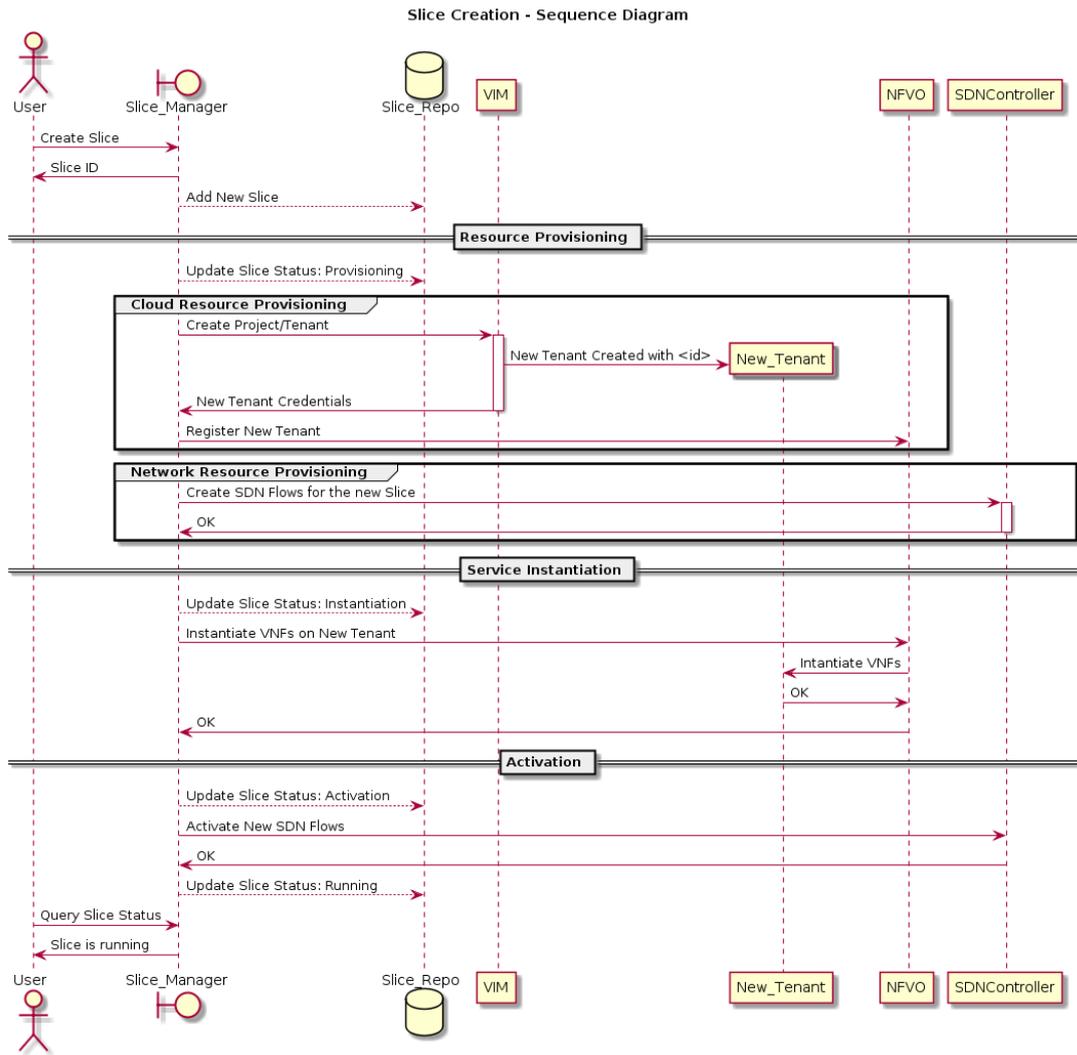


Figure 76 - Illustration of the service creation sequence diagram for the instantiation and the configuration of a network slice

6 Conclusions

The concept and the integration of IoRL services with the consideration of inter-layer interfaces were presented. Introducing the system's localization techniques based on MmWave modules. We also performed VLC and MmWave channel modelling with signal blockages considerations, providing network-layer optimization, along with network slicing implementation mechanisms. Meanwhile, we provided a general overview for the overall system as a 5G small-cell, highlighting the deployment options within a MNOs, with pros and cons of each deployment option in compliance with 3GPP network architecture.

References

- [1] "OpenStack Queens, [Online]: <https://docs.openstack.org/queens/>," [Online].
- [2] "OpenStack virtual router specifications, [Online]: <https://docs.openstack.org/python-openstackclient/queens/cli/command-objects/router.html>," [Online].
- [3] "Ryu controller, [Online]: <https://osrg.github.io/ryu/>," [Online].
- [4] "Open Source MANO, Rel.5, [Online]: https://osm.etsi.org/wikipub/index.php/OSM_Release_FIVE," [Online].
- [5] "Foundation," Node.js, [Online]. Available: <https://nodejs.org/en/>," [Online].
- [6] "mongoDB," mongoDB, [Online]. Available: <https://www.mongodb.com/>," [Online].
- [7] "FFmpeg," FFmpeg, [Online]. Available: <https://ffmpeg.org/>," [Online].
- [8] "x264," VideoLan organisation, [Online]. Available: <https://www.videolan.org/developers/x264.html> ..
- [9] "[9] J. M. e. a. j.Bruneau-Queyreix, "PMS: A Novel Scale-Adaptive and Quality-Adaptive Hybrid P2P/Multi-Source Solution for Live Streaming," ACM Transactions on Multimedia Computing, Communications and Applications, vol. 14, p. Atricle 35, 2018."
- [10] "J. B. -. Q. e. a. S.Silva, "MUSLIN: Achieving High, Fairly Shared QoE Through Multi-Source Live Streaming," ACM Multimedia Systems (MMSys) -Packet Video Workshop, demonstration track, 2018."
- [11] "M. L. e. a. J.Bruneau - Queyreix, "QoE Enhancement Through Cost-Effective Adaptation Decision Process for Multiple-Server Streaming over HTTP," IEEE International Conference on Multimedia and Expo (ICME), 2017."
- [12] 3GPP, "5G Specifications, http://www.3gpp.org/ftp/Specs/archive/23_series/23.501/," 3rd Generation Partnership Project (3GPP), 2017.
- [13] "N. Jawad et al., "Smart Television Services Using NFV/SDN Network Management," in IEEE Transactions on Broadcasting. doi: 10.1109/TBC.2019.2898159".
- [14] "Kareem Ali, Akram Alkhatar, Nawar Jawad, John Cosmas, "978-1-5386-4729-5/18/\$31.00 ©2018 IEEE IoRL Indoor Location Based Data Access, Indoor Location Monitoring & Guiding and Interaction Applications" IEEE International Symposium on Broadband Multimedia S".
- [15] "J. Cosmas et al., "5G Internet of radio light services for Musée de la Carte à Jouer,"

- 2018 Global LIFI Congress (GLC), Paris, 2018, pp. 1-6. doi: 10.23919/GLC.2018.8319095".
- [16] D. W. e. al, "Particle swarm optimization algorithm: an overview.," in *Soft Computing* 22, 2018, pp. 387 - 408..
- [17] T. Wegner, "Precision of UWB localization in realistic scenarios," 2018..
- [18] H. Haas, L. Yin, Y. Wang, and C. Chen, "What is LiFi?" *J. Lightw. Technol.*, vol. 34, no. 6, pp. 1533–1544, Mar 2016.
- [19] D. Basnayaka and H. Haas, "Hybrid RF and VLC systems: Improving user data rate performance of VLC systems," in *IEEE 81st Vehicular Technology Conference (VTC Spring)*, Glasgow, May 2015, pp. 1–5.
- [20] M. Ayyash, H. Elgala, A. Khreishah, V. Jungnickel, T. Little, S. Shao, M. Rahaim, D. Schulz, J. Hilt, and R. Freund, "Coexistence of WiFi and LiFi toward 5G: concepts, opportunities, and challenges," *IEEE Communications Magazine*, vol. 54, no. 2, pp. 64–71.
- [21] I. Stefan and H. Haas, "Hybrid visible light and radio frequency communication systems," in *Vehicular Technology Conference (VTC Fall)*, 2014 IEEE 80th. IEEE, 2014, pp. 1–5.
- [22] D. A. Basnayaka and H. Haas, "Hybrid RF and VLC systems: Improving user data rate performance of VLC systems," in *Proc. IEEE Veh. Technol. Conf. (VTC Spring)*, Glasgow, pp. 1–5, UK, May 2015.
- [23] X. Li, R. Zhang, and L. Hanzo, "Cooperative load balancing in hybrid visible light communications and WiFi," *IEEE Trans. Commun.*, vol. 63, no. 4, pp. 1319–1329, 2015.
- [24] Y. Wang, D. A. Basnayaka, and H. Haas, "Dynamic load balancing for hybrid Li-Fi and RF indoor networks," in *Proc. IEEE Int. Conf. Commun. Workshops (ICCW)*, London, UK, 2015, pp. 1422–1427.
- [25] X. Li, R. Zhang, and L. Hanzo, "Cooperative load balancing in hybrid visible light communications and WiFi," *IEEE Trans. Commun.*, vol. 63, no. 4, pp. 1319–1329, 2015.
- [26] F. Kelly, "Charging and rate control for elastic traffic," *Transactions on Emerging Telecommunications Technologies*, vol. 8, no. 1, pp. 33–37, 1997.
- [27] Y. Wang and H. Haas, "Dynamic load balancing with handover in hybrid Li-Fi and Wi-Fi networks," *IEEE/OSA J. Lightw. Technol.*, vol. 33, no. 22, pp. 4671–4682, Nov. 2015.
- [28] Y. Wang, D. A. Basnayaka, X. Wu, and H. Haas, "Optimization of load balancing in hybrid LiFi/RF networks," *IEEE Trans. Commun.*, vol. 65, no. 4, pp. 1708–1720, 2017.
- [29] X. Wu, M. Safari, and H. Haas, "Joint optimisation of load balancing and handover for hybrid LiFi and WiFi networks," in *Proc. IEEE Wireless Commun. and Netw. Conf.*

- (WCNC), pp. 1–5, San Francisco, USA, 2017.
- [30] X. Wu and H. Haas, “Access point assignment in hybrid LiFi and WiFi networks in consideration of LiFi channel blockage,” in *Signal Processing Advances in Wireless Communications (SPAWC), 2017 IEEE 18th International Workshop on*. IEEE, 2017.
- [31] X. Wu, D. Basnayaka, M. Safari, and H. Haas, “Two-stage access point selection for hybrid VLC and RF networks,” in *Personal, Indoor, and Mobile Radio Communications (PIMRC), 2016 IEEE 27th Annual International Symposium on*. IEEE, 2016, pp. 1–6.
- [32] F. Kelly, “Charging and rate control for elastic traffic,” *Transactions on Emerging Telecommunications Technologies*, vol. 8, no. 1, pp. 33–37, 1997.
- [33] Y. Wang and H. Haas, “A comparison of load balancing techniques for hybrid LiFi/RF networks,” in *Proceedings of the 4th ACM Workshop on Visible Light Communication Systems*, pp. 43–47, 2017.
- [34] M. Kashef, M. Abdallah, and N. Al-Dhahir, “Transmit power optimization for a hybrid PLC/VLC/RF communication system,” *IEEE Transactions on Green Communications and Networking*, 2017.
- [35] H. Zhang, N. Liu, K. Long, J. Cheng, V. C. M. Leung, and L. Hanzo, “Energy efficient subchannel and power allocation for the software defined heterogeneous VLC and RF networks,” *IEEE Journal on Selected Areas in Communications*, vol. PP, no. 99, pp. 1–1, 2.
- [36] H. Zhang, N. Liu, K. Long, J. Cheng, V. C. M. Leung, and L. Hanzo, “Energy efficient subchannel and power allocation for the software defined heterogeneous VLC and RF networks,” *IEEE Journal on Selected Areas in Communications*, vol. PP, no. 99, pp. 1–1, 2.
- [37] S. Boyd, Stephen Boyd and L. Vandenberghe, “Convex Optimization”, Department of Electrical Engineering.
- [38] Y. Zhou, L. Ruan, L. Xiao, R. Liu, “A Method for Load Balancing based on Software Defined Network”, School of Computer Science and Engineering Beihang University, Beijing, China, 2014.
- [39] M. B. Navarro, “Dynamic Load Balancing in Software Defined Networks”, Aalborg University, 2014.
- [40] J. R. Jiang, W. Yahya and M. T. Ananta, “Load Balancing and Multicasting using the Extended Dijkstra’s Algorithm in Software Defined Networking. National Central University Jhongli City, Taiwan 2011.
- [41] Z. Shang, W. Chen, Q. Ma, B. Wu, “Design and implementation of server cluster dynamic load balancing based on OpenFlow”, Lanzhou University Communication Network

Center 2013.

- [42] X. Li, R. Zhang, and L. Hanzo, "Cooperative load balancing in hybrid visible light communications and WiFi," *IEEE Trans. Commun.*, vol. 63, no. 4, pp. 1319–1329, 2015.
- [43] Y. Wang, D. A. Basnayaka, and H. Haas, "Dynamic load balancing for hybrid Li-Fi and RF indoor networks," in *Proc. IEEE Int. Conf. Commun. Workshops (ICCW)*, London, UK, 2015, pp. 1422–1427.
- [44] X. Li, R. Zhang, and L. Hanzo, "Cooperative load balancing in hybrid visible light communications and WiFi," *IEEE Trans. Commun.*, vol. 63, no. 4, pp. 1319–1329, 2015.
- [45] F. Kelly, "Charging and rate control for elastic traffic," *Transactions on Emerging Telecommunications Technologies*, vol. 8, no. 1, pp. 33–37, 1997.
- [46] Y. Wang and H. Haas, "Dynamic load balancing with handover in hybrid Li-Fi and Wi-Fi networks," *IEEE/OSA J. Lightw. Technol.*, vol. 33, no. 22, pp. 4671–4682, Nov. 2015.
- [47] Y. Wang, D. A. Basnayaka, X. Wu, and H. Haas, "Optimization of load balancing in hybrid LiFi/RF networks," *IEEE Trans. Commun.*, vol. 65, no. 4, pp. 1708–1720, 2017.
- [48] X. Wu, M. Safari, and H. Haas, "Joint optimisation of load balancing and handover for hybrid LiFi and WiFi networks," in *Proc. IEEE Wireless Commun. and Netw. Conf. (WCNC)*, pp. 1–5, San Francisco, USA, 2017.
- [49] T. Bai, V. Desai, and R. W. Heath Jr., "Millimetre wave cellular channel models for system evaluation," in *Proceedings of the 2014 International Conference on Computing, Networking and Communications, ICNC 2014*, pp. 178–182, USA, Feb 2014.
- [50] T. Bai, R. Vaze, and R. W. Heath, "Analysis of Blockage Effects on Urban Cellular Networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 9, pp. 5070–5083, 2014..
- [51] H. Burchardt, N. Serafimovski, D. Tsonev, S. Videv, and H. Haas, "VLC: Beyond point-to-point communication," *IEEE Commun. Mag.*, vol. 52, no. 7, pp. 98–105, Jul 2014.
- [52] M. Steinbauer, A. F. Molisch, and E. Bonek, "The double-directional radio channel," *IEEE Antennas and Propagation Magazine*, vol. 43, no. 4, pp. 51–63, 2001.
- [53] T. Bai, V. Desai, and R. W. Heath Jr., "Millimeter wave cellular channel models for system evaluation," in *Proceedings of the 2014 International Conference on Computing, Networking and Communications, ICNC 2014*, pp. 178–182, USA, February 2014.
- [54] M. R. Akdeniz, Y. Liu, M. K. Samimi et al., "Millimeter wave channel modeling and cellular capacity evaluation," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1164–1179, 2014.

-
- [55] T. Bai, V. Desai, and R. W. Heath Jr., "Millimeter wave cellular channel models for system evaluation," in Proceedings of the 2014 International Conference on Computing, Networking and Communications, ICNC 2014, pp. 178–182, USA, February 2014.
- [56] M. R. Akdeniz, Y. Liu, M. K. Samimi et al., "Millimeter wave channel modeling and cellular capacity evaluation," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1164–1179, 2014.
- [57] X. Wu, M. Safari, and H. Haas, "Access point selection for hybrid Li-Fi and Wi-Fi networks," *IEEE Transactions on Communications*, vol. 65, no. 12, pp. 5375–5385, 2017.
- [58] J. G. Mohta and S Das, "Channel modelling and performance analysis of WiFi", *International Journal of Engineering Research and General Science*, vol. 3, no. 3, Jun 2015.
- [59] Y. Wang, X. Wu and H. Haas, "Load Balancing Game with Shadowing Effect for Indoor Hybrid LiFi/RF Networks," in *IEEE Transactions on Wireless Communications*, vol. 16, no. 4, pp. 2366-2378, Apr 2017.
- [60] E. Perahia, C. Cordeiro, P. Minyoung and L. L. Yang, "IEEE 802.11ad: Defining the Next Generation Multi-Gbps Wi-Fi," *IEEE Consumer Commun. and Networking Conference (CCNC)*, pp. 1-5, Jan 2010.
- [61] D. Saliba, R. Imad and S. Houcke, "Wifi channel selection based on load criteria," *International Symposium on Wireless Personal Multimedia Communications (WPMC)*, pp. 332-336, Bali, 2017.
- [62] H. A. Omar, K. Abboud, N. Cheng, K. R. Malekshan, A. T. Gamage and W. Zhuang, "A Survey on High Efficiency Wireless Local Area Networks: Next Generation WiFi," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 4, pp. 2315-2344, Fourthquarter 2016.
- [63] V. Erceg, L. Schumacher, P. Kyritsi, and et al., *TGn Channel Models. IEEE 802.1103/940r4*, 2004.
- [64] E. Perahia and R. Stacey, *Next Generation Wireless LAN: 802.11n and 802.11ac*. Cambridge University Press, 2013.
- [65] X. Wu, M. Safari, and H. Haas, "Access point selection for hybrid Li-Fi and Wi-Fi networks," *IEEE Transactions on Communications*, vol. 65, no. 12, pp. 5375–5385, 2017.
- [66] D. Saliba, R. Imad and S. Houcke, "Wifi channel selection based on load criteria," *International Symposium on Wireless Personal Multimedia Communications (WPMC)*, pp. 332-336, Bali, 2017.
- [67] Matlab Simulink WLAN channel modelling, [Online]: <https://www.mathworks.com/help/wlan/gs/wlan-channel-models.html>.

-
- [68] Matlab Simulink WLAN channel modelling, [Online]: <https://www.mathworks.com/help/comm/ref/stdchan.html>.
- [69] Jain's fairness index [Online]: https://en.wikipedia.org/wiki/Fairness_measure.
- [70] 3GPP, "Study on management and orchestration of network slicing for next generation network," 2018. [Online]. Available: <https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3091..>
- [71] E. G. NFV-EVE, " "Network Functions Virtualisation (NFV) Release 3; Evolution and Ecosystem; Report on Network Slicing Support with ETSI NFV Architecture Framework," Dec 2017.
- [72] IoRL, "Deliverable D2.1 - Definition and Description of the IoRL Use Cases and Derivation of User Requirements," Dec 2017.
- [73] IoRL, "Deliverable D2.2 - System Functional Requirements and Architecture," Feb 2018..
- [74] OSM Hackfest [Online]: https://osm.etsi.org/wikipub/index.php/5th_OSM_Hackfest.