**Journal of Network and Systems Management manuscript No.**
(will be inserted by the editor)

# SD-RAN Interactive Management using In-band Network Telemetry in IEEE 802.11 Networks

**P. H. Isolani* · J. Haxhibeqiri · I. Moerman · J. Hoebeke · L. Z. Granville · S. Latré · J. M. Marquez-Barja.**

Received: date / Accepted: date

**Abstract** Future digital factories are becoming more and more softwarized. This introduces flexibility, but the industry also demands robustness and Quality of Service (QoS) support. While 5G envisions to enable real-time applications with strict performance requirements, IEEE 802.11 networks continue to be a viable option for indoor scenarios. However, IEEE 802.11 networks currently cannot be programmed fine-grained enough to fully support and ensure QoS. Besides, due to the unknown or coarse-grained monitoring information over the wireless links, detecting performance degradation is still challenging. In this paper, we propose a Software-Defined Radio Access Network (SD-RAN) interactive management approach assisted by In-band Network Telemetry (INT) in IEEE 802.11 networks. We design a system-level architecture that includes the network administrator in the management loop through monitoring, visualization, and configuration activities. To demonstrate the feasibility of our approach, we developed a prototype and evaluated its flexibility in a real-world testbed. We argue that, with the fine-grained network information from INT and the benefits of the Software-Defined Networking (SDN) paradigm, an SDN-tailored system can assist in troubleshooting and assess enhanced QoS delivery.

P. H. Isolani and S. Latré
IDLab, Department of Computer Science, University of Antwerp - imec, Antwerp, Belgium
E-mail: {pedro.isolani steven.latre}@uantwerpen.be)

J. Haxhibeqiri, I. Moerman, and J. Hoebeke
IDLab, Department of Computer Science, University of Ghent - imec, Ghent, Belgium
E-mail: {jetmir.haxhibeqiri, ingrid.moerman, jeroen.hoebeke}@ugent.be)

L. Z. Granville
Computer Networks Group, Federal University of Rio Grande do Sul, Porto Alegre, Brazil
E-mail: granville@inf.ufrgs.br

J. M. Marquez-Barja
IDLab, Department of Electronics - ICT, University of Antwerp - imec, Antwerp, Belgium
E-mail: johann.marquez-barja@uantwerpen.be

**Keywords** IEEE 802.11 networks · In-band Network Telemetry · Quality of Service · SDN · SD-RAN

## 1 Introduction

Despite the 5G ever-increasing demand for mobile services, services and applications are expected to run with better reliability, massive connectivity, and low latency [3]. In the context of future digital factories, IEEE 802.11 networks provide flexibility. However, while 5G advances, such networks must also provide robustness and Quality of Service (QoS) support. Moreover, in this context, applications often end up sharing the wireless infrastructure and Industrial, Scientific, and Medical (ISM) radio bands. Therefore, making it difficult to meet diverging Service Level Agreements (SLAs).

Network slicing, in turn, became the abstraction for providing networking resources on both wired and wireless infrastructures [1]. Besides operating independently from one another, network slices provide precise networking resources and traffic isolation among users and services. Considering that Software-Defined Networking (SDN) has been facilitating network innovation and simplifying network management, SDN-tailored systems are envisioned to ease the creation of network slices. Although SDN simplifies and even solves critical management issues, the instability caused by the presence of wireless interference and physical obstacles still hampers the ample management of wireless networks.

Future IEEE 802.11 networks are calling for flexible and programmable networks that ensure high-reliability in dynamic scenarios. To meet the QoS requirements, existing SDN proposals often make use of network slicing due to its flexibility and traffic isolation capabilities [46]. However, most proposals on network slicing focus on maximizing network usage in terms of throughput. Whilst, essential metrics such as the End-to-End (E2E) latency and jitter, are often left aside. Besides, given the coarse-grained level of the monitoring information, detecting performance degradation in such scenarios is yet challenging. Therefore, to enable flexible runtime management and QoS support, there is a need for fine-grained E2E network monitoring information.

In previous work, we evaluated the impact of airtime-based network slicing on the E2E latency using the Internet Control Message Protocol (ICMP) [44] protocol and we exploited its potential to guarantee latency-related requirements [27]. Thereafter, besides developing and integrating queueing delay measurements of Access Points (APs) into the formulation of a QoS optimization problem [28], we proposed a delay-aware approach for performing runtime Medium Access Control (MAC) management via airtime-based network slicing, traffic shaping, and user association using Multi-Criteria Decision Analysis (MCDA) [31,30]. Although we identified that stringent QoS requirements of 5G Mission-Critical Applications (MCAs) can be potentially met and the ping-pong effect can be avoided, our approach was limited to the coarse-grained level of the monitoring statistics collected at the APs.

To address low-overhead and fine-grained network statistics, we designed, evaluated, and demonstrated an In-band Network Telemetry (INT)-enabled node architecture with a proof of concept implementation [19,29]. On one hand, as INT uses regular data packets to carry telemetry instructions or data, such a mechanism ends up offering low overhead in comparison to others. On the other hand, instead of relying on only overall statistics such as E2E (e.g., throughput, packet loss), INT also enables gathering statistics on a per-hop, per-packet, and per-flow basis. Thus, the E2E flow dynamics, state of the wireless links, and per-hop reliability can be fully assessed.

In this paper, we present our Software-Defined Radio Access Network (SD-RAN) interactive management approach assisted by INT in IEEE 802.11 networks. First, we describe the architecture and the implementation details to realize such a system. Then, we evaluate our approach in a real-world testbed composed of three APs, controlled by an SD-RAN and a backhaul SDN controller, and eight Stations (STAs). In summary, our contributions are twofold:

1. A system-level architecture and a prototype implementation of an SD-RAN interactive management approach assisted by INT in IEEE 802.11 networks; and
2. An experimental evaluation followed by a discussion about the challenges and limitations of our approach.

The remainder of this paper is organized, as follows. In Section 2, we summarize and motivate the need for flexible management and fine-grained monitoring information on wireless networks. In Section 3, we introduce our system architecture. In Section 4, we present the design aspects and implementation details of our developed prototype for IEEE 802.11 networks. In Section 5, we describe our testbed, set-up, and the the results of our experimentation. Finally, in Section 6, we discuss future opportunities and challenges while summarizing the future work.

## 2 Related Work

In the context of SDN, much has been discussed about SDN and network management by itself, mostly from the perspective of where SDN is taken as a management tool [63]. Given the benefit of the SDN centralized view, many approaches have been addressing monitoring, control, and management in a high-level perspective and, thus, enhance the QoS in the IEEE 802.11 Radio Access Network (RAN). On the other hand, statistics polling, active probing, and INT have been the techniques used for network monitoring. In this section, we review the major efforts in those areas.

### 2.1 SDN management and IEEE 802.11 QoS delivery

Despite the growing number of connected devices and data-hungry mobile applications, 5G aims to enable services and applications to run with lower latency, better reliability, massive connectivity, and improved energy efficiency [3]. As Ultra High Definition (UHD) and 3D video content are now a reality and the fact that applications will require Enhanced Mobile Broadband (eMBB), Ultra-Reliable Low Latency Communication (URLLC), and Massive Machine Type Communications (mMTC), industry and standardization entities have been fostering the research towards reliable and improved QoS delivery [2]. In the context of IEEE 802.11 networks, QoS has been pursued via Enhanced Distributed Channel Access (EDCA) parameters adaptation and network slicing. Besides, proposals have often benefited from the centralized view of SDN to present sophisticated solutions for load balancing and mobility support.

After the IEEE 802.11e amendment [24] established the foundations for traffic prioritization, many investigations focused on queuing management as a means to enhance QoS [38, 53, 43, 10]. With the improvements in radio resource utilization provided by the IEEE 802.11n amendment [25], researchers focused on channel optimization and fairness (e.g., modifying or predicting the Aggregated MAC Service Data Unit (A-MSDU) behavior) [50, 35, 5, 39, 54]. To overcome the the well-known IEEE 802.11 Performance Anomaly [20] and enhance the QoS delivery, researchers have been addressing infrastructure sharing via network slicing [46, 48, 22, 4, 21, 11, 27, 28, 47, 30]. Although most work has been only assessed via simulation, there were several practical implementations as well [22, 4, 21, 11].

Coronado *et al.* [11] enabled runtime E2E network slicing while ensuring resource isolation. The authors presented a framework capable of performing programmable, flow-based, and dynamic E2E network slicing over heterogeneous RANs. The proposed solution was implemented and experimented on a real-world testbed where network slicing was analysed in terms of throughput. In addition, the authors make use of the Light Virtual Access Point (LVAP) abstraction, first proposed by Grunenberger and Russeau [17], to handle STA/AP communication from a centralized point of view. The LVAP abstraction allows a physical AP to use different LVAPs for communication with each STA. This, in turn, avoided problems that are caused by legacy handover algorithms such as unnecessary re-associations and connection disruptions.

Several other frameworks [65, 56, 68, 69, 51, 49] used the LVAP abstraction, or similar, to enable seamless handovers. Nonetheless, Ethanol [42] focused in a more complete set of primitives, allowing to configure features such as mobility domain and security. Thus, taking advantage of such frameworks, researchers have been proposing different sorts of user association algorithms and thus enhancing load balancing of flows across APs mobility support, fairness, and QoS [31] [30] [12, 16, 67]. Generally, such user association algorithms have been proposed aiming to minimize the average number of STAs assigned to an AP and maximize the overall throughput of the network and Received Signal Strength Indicator (RSSI) [30].

In contrast, the ORCHESTRA framework [66, 14, 13] enabled seamless handover between different wireless technologies. Throughout a software overlay, the authors propose a virtual MAC layer where network applications interact with a single virtual network interface. In this manner, the proposed virtual

MAC layer can handle packets through multiple Network Interface Card (NIC) seamlessly in addition to perform load balancing of flows between different wireless technologies (e.g., Long-Term Evolution (LTE) and IEEE 802.11). As usual SDN, the ORCHESTRA controller is able to control virtual MAC layer implementation from a centralized point of view.

Vassilaras et al. [60] state that future wireless networks have to consider E2E latency requirements for a service chain where feasible slice embedding must also satisfy and guarantee the QoS requirements. Yet, verifying the QoS E2E throughout a wireless network remains challenging. In our previous work [27] [28] [31] [30] we studied the impact of performing airtime-based network slicing and user association at the RAN. However, only a coarse-grained level of monitoring was addressed and, hence, limiting the understanding and the insights about the network behavior. In this paper, we propose an SD-RAN interactive management approach to foster further research on flexible and programmable IEEE 802.11 networks.

### 2.2 Network Monitoring and INT

Network monitoring is one of the fundamental management activities to understand and control network behavior. In SDN, both monitoring and interactive visualizations are crucial to trace functional parameters of novel applications and for the endorsement of network controllers jurisdiction [63] [32]. In this context, accurate and timely monitoring usually impacts the network load while sparse monitoring compromises its accuracy and hence the insights about the data measured. Many studies pointed out possible bottlenecks at the control channel and attempted to alleviate them by distributing the control logic [6]. On the other hand, monitoring techniques such as device statistics polling [57] [55] and active probing [41] also attempted to decrease the communication overhead using an adaptive polling rate [61] or probing rate [45]. However, given the coarse granularity of the monitored information, these techniques are not always able to identify network performance issues [37].

In SDN, network conditions are much more dynamic because of facilitated, frequent software installation and changes. Therefore, INT has gained even more interest in the research community. Several studies have presented the use of INT on different host environments and networks [34, 58, 18, 23, 15, 7]. There, authors have discussed how INT monitoring information assists network troubleshooting and enhance network performance. Bhamare *et al.* [7] present a system that combines INT with active probing to perform monitoring of flexible Virtual Network Function (VNF) service chains and enhance scalability. Their solution consists of the use of active probing, between the SDN controller and network devices, and an INT-enabled network that adds telemetry information into the active probes. To implement the INT-enabled programmable switches, the authors used P4 [8], a high-level language for programming protocol-independent packet processors. Despite the limited overhead, the presence of active probing negatively impacts the control channel.

There are several implementations of an INT monitoring system [58,18,23,33]. Among those, the INT layer has been mainly implemented and tested on wired networks. The authors have proposed an INT layer in Open vSwitch[1] by combining the P4 INT implementation with extended Berkeley Packet Filters (BPFs). On the other hand, others have implemented INT for the SDN controller ONOS[2] using the P4 programming language. To date, apart from the wired networks, INT implementations were only evaluated on Wireless Sensor Networks (WSNs) [33]. The authors presented an INT design for Industrial Wireless Sensor Networks (IWSN) focused on minimizing resource consumption and communication overhead while offering extensive monitoring capabilities. Besides, the authors presented how INT monitoring can be assessed for network management such as to enhance network awareness, enhance network control and isolation, and assist network troubleshooting.

Despite the importance of gathering fine-grained monitoring information at runtime, the time spent processing on analyzing it cannot be neglected. Therefore, some authors [52] [59] focused on the extraction of network events based on the raw monitoring data, gathered with INT. The Prometheus INT exporter [52] enables network information extraction into metrics and pushes its values to a gateway. Later, a database scrapes the latest data retrieved from the gateway periodically. Tu *et al.* [59] presented a high-performance collector for INT. The proposed mechanism is capable of extracting the important network information from the INT raw data. Based on network events, the authors show how monitoring data stored is reduced as well as the Central Processing Unit (CPU) usage due to data processing. Nonetheless, besides controlling the granularity of the monitoring and hence avoid excessive overhead, the network needs to react in response to high-level policies. Therefore, besides that the E2E latency, E2E jitter, and throughput requirements are challenging to be guaranteed in wireless networks, they are often unknown due to the coarse granularity of the monitoring information.

In our previous work [27,31,28,30] we pursued enhanced QoS delivery by performing airtime-based network slicing and user association at the IEEE 802.11 RAN while performing traffic shaping at the end devices. However, we addressed only a coarse-grained level of monitoring and, hence, limited the understanding and insights about the network behavior. We designed and evaluated an INT-enabled node architecture with a proof of concept implementation [19] and demonstrated the use of such fine-grained network information in a real-world testbed [29]. In this work, we present a complete SD-RAN interactive management approach with system-level architecture, implementation details, and comprehensive evaluation. The demonstration of our SDN-based framework in action where fine-grained E2E network statistics can be gathered using INT are used for network management in a real-world testbed is presented in [29].

[1] https://www.openvswitch.org/

[2] https://www.opennetworking.org/onos/

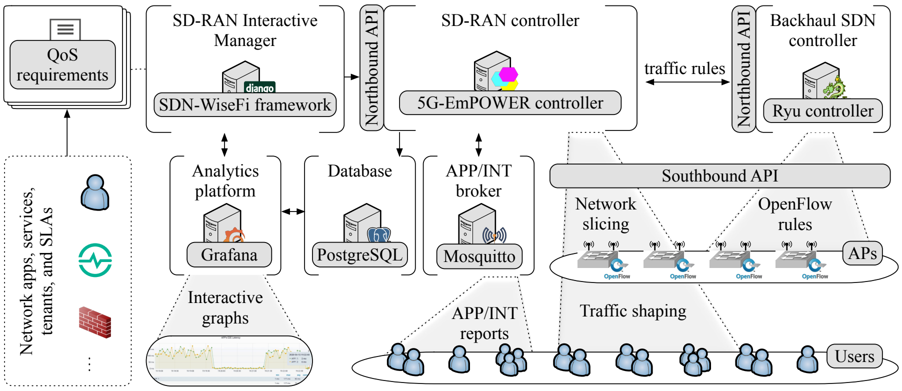



Fig. 1: SDN-enabled layered network architecture.

## 3 System Overview

We consider an SDN-enabled scenario where the network is managed by a centralized controller and therefore has the overall view over the network resources. Typically, the network infrastructure is shared by multiple tenants (i.e., virtual operators or service providers) and has to support specific SLAs as QoS requirements (e.g., the average expected throughput, average/median latency over time, and packet error ratio thresholds). Figure 1 illustrates our SDN-enabled layered network architecture.

### 3.1 Software-defined IEEE 802.11 RAN

In the context of IEEE 802.11 networks, the RAN consists of a set of APs, a.k.a Wireless Termination Points (WTPs), responsible for delivering data from different services to/from several users (STAs). Each AP has resources to be shared and therefore has to be properly managed. To collect measurements at the AP, the IEEE 802.11 interface is set to monitor mode. To control resource utilization and ensure QoS delivery at the RAN, we use network slicing following the Quality of Service Slicing (QoSS) definition of slices [48]. The minimum chunk of wireless resources that can be assigned to a user is defined by an IEEE 802.11 interface at a given AP, identified by the network interface identifier (e.g., MAC address), operating channel (e.g., 1, 6, 11), and the type of channel (e.g., High Throughput (HT) 20MHz, Very High Throughput (VHT) 40MHz).

To perform network reconfigurations, our proposed framework, (a.k.a., *SDN-WiseFi*), provides access to fine-grained monitoring information from INT while facilitating network management. The network control is implemented at the SD-RAN controller which, in turn, communicates with the APs using a persistent Transmission Control Protocol (TCP) connection. Our ap-

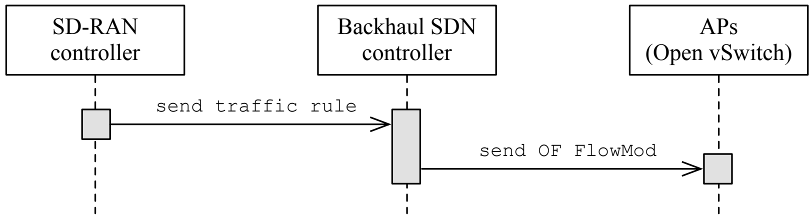



Fig. 2: Sequence diagram of the traffic rule creation process.

proach uses the 5G-EmPOWER platform[3], including the 5G-EmPOWER SD-RAN controller, a backhaul implementation of the SDN controller Ryu, and a programmable agent that runs at each AP.

Two components define the programmable agent that runs at each AP: an Open vSwitch[4] instance that operates under the supervision of the OpenFlow-enabled SDN controller and a Click modular router [36] instance implementing the IEEE 802.11 data-path with a *hypervisor*. In this manner, the SD-RAN controller can request the backhaul controller to tag traffic matching a certain flow through the definition of *traffic rules* (i.e., OpenFlow rules [40]). Figure 2 illustrates the sequence of communication from the SD-RAN controller to the Open vSwitch instance at the APs. First, the SD-RAN controller sends a message to the backhaul SDN controller containing the traffic rule description. The backhaul SDN controller then takes the traffic rule description and installs the defined traffic rules in each of the APs as OpenFlow rules.

This communication between the SD-RAN controller and the Click modular router instance of the APs is given by the *OpenEmpower protocol* that, besides enabling monitoring of statistics at AP, enables to perform reconfigurations such as the reassignment of resources among slices and the definition of transmission policies. Figure 3 depicts a simplified queue structure along with the data traffic flow within a single AP.

Slices are therefore mapped by the hypervisor according to the Service Set Identifier (SSID) and the Differentiated Services Code Point (DSCP). The SSID is the name of an IEEE 802.11 network and the DSCP determines the priority of each IP packet. The hypervisor is in charge of creating, monitoring, and managing network slices according to traffic rules, ensuring performance isolation and efficient resource utilization. Each traffic rule contains multiple aggregation buffers, one for each user in the slice and these aggregation buffers are scheduled using the Round Robin policy. Besides, traffic rules can be combined with transmission policies in which allow the definition of specific parameters for each AP/STA communication. For instance, the Modulation and Coding Schemes (MCSes) index range used by the rate selection algorithm, the Request to Send/Clear to Send (RTS/CTS) threshold, the multicast policy,

[3] https://github.com/5g-empower/5g-empower.github.io
[4] https://www.openvswitch.org/

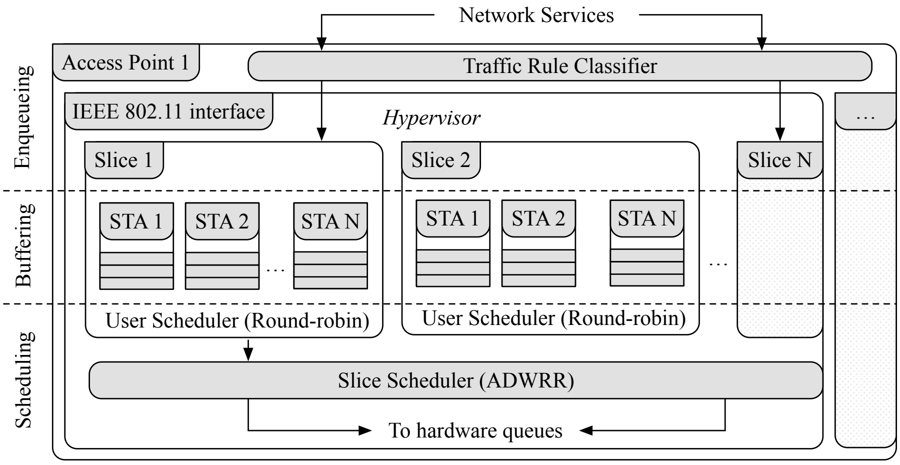



Fig. 3: Simplified slice queue structure and data traffic flow in an AP.

etc. Subsequently, frames are dequeued following the Airtime Deficit Weighted Round Robin (ADWRR) scheduling algorithm.

## 3.2 ADWRR Scheduling Algorithm

The ADWRR is a scheduling algorithm designed to consider the *cost* of transmitting a frame concerning the resources that have to be allocated to it [11]. This cost includes its length and the actual channel conditions when transmitting it towards an STA. Within each of the APs, the ADWRR scheduling algorithm is responsible to assign a fraction of the airtime, a.k.a quantum, to each traffic rule according to its relative priority. Quantum values are allocated to slices' deficit counter at each of their transmission rounds in the scheduler. Transmission rounds follow the Round Robin policy. However, the hypervisor only serves traffic rules whose expected transmission time, estimated by a rate control algorithm (i.e., Minstrel [64]), is smaller than the deficit counter.

Moreover, it is important to consider that the scheduler only iterates upon *active* queues/slices and therefore inactive queues/slices do not cause any performance degradation to the system. With ADWRR, the quantum of an individual slice can be adjusted independently, which allows the airtime of slices to be controlled dynamically. In this manner, higher quantum values can be assigned to a slice supporting services with stricter performance requirements, hence, allocating more radio resources to it. Smaller quantum values can be assigned to slices overusing the wireless NIC. In such cases, frames are discarded at the higher MAC layer when a queue reaches its maximum capacity limit. Thus, leaving radio resources to concurrent slices.

### 3.3 Programmable data plane and INT

To gather the application QoS requirements and the INT measurements over time, we introduced a programmable data plane. First, we extended the existing programmable agent at the APs to enable appending INT statistics into data frames. Then, at the STAs, we designed a programmable agent that, depending on the roles assumed in the flow path (source or destination) initializes or reports INT measurements towards the controller. We use a Message Queuing Telemetry Transport (MQTT) broker[5] where monitoring data is exchanged via the publish–subscribe messaging pattern. Moreover, we use a PostgreSQL database[6] to store such monitoring data. The frequency in which INT measurements flow is configurable and set by the source node in the flow path. On the other hand, to visualize the INT measurements at runtime, we built interactive graphs using Grafana[7] which are further embedded into our management framework.

### 3.4 Interactive Monitoring, Visualization, and Configuration

To enhance network manageability, our focus is to include the network administrator in the management loop through monitoring, visualization, and configuration activities. Therefore, we designed a framework to allow coherent and high-level management, called *SDN-WiseFi*. Besides facilitating network management such as the interaction with network controllers and algorithms, our framework provides interactive visualizations to the fine-grained INT monitoring information through an interactive Graphical User Interface (GUI). The framework provides interactive visualizations such as the topology and the fine-grained statistics of INT through time-series graphs at runtime. Nonetheless, our framework allows the management of SD-RAN controllers in a unified manner, enabling the dynamic orchestration of slices and the definition of transmission policies. In the next section, we present the design aspects and implementation details, followed by the evaluation of our approach in a real-world testbed.

## 4 Design Aspects and Implementation Details

To demonstrate the usefulness of our approach in a real-world setting, we implemented: (*i*) a programmable data plane with INT-enabled features; (*ii*) monitoring and control applications for maintaining statistics and performing the network reconfigurations at the RAN; and (*iii*) *SDN-WiseFi*, an interactive SDN management framework that allows coherent and high-level management of the network.

[5] https://mosquitto.org/

[6] https://www.postgresql.org/

[7] https://grafana.com/

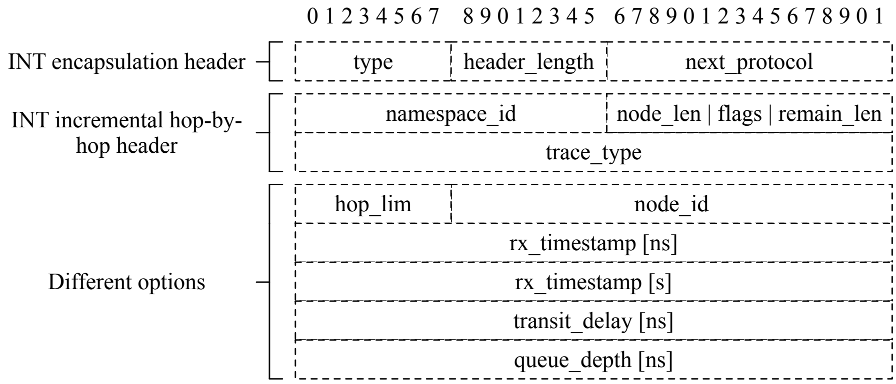



Fig. 4: INT header format.

### 4.1 INT Design Aspects

The INT header format and its encapsulation within the Layer 2 packet are being standardized by the IETF [9] [62]. However, the current INT standardization draft only supports E2E reliability monitoring [9]. This method, for example, cannot determine at which hop the packet was lost. To enable INT monitoring in wireless networks, new hop-by-hop options are defined as well as how they are encapsulated. Figure 4 shows the current composition of the INT headers. The INT encapsulation header determines the *type* of the INT header, the *header length*, and the *next protocol*, composing the INT options fields. The INT header types are exclusionary and stand-in for hop-by-hop, E2E, or proof-of-transit. Each of them is followed by the INT options field. In Figure 4, the INT hop-by-hop header is shown, composed by the following fields:

– *Namespace ID:* field used to identify each node allowed to process INT-enabled data packets;
– *Node length:* the total number of nodes that can add INT measurements before there is no space left in a packet;
– *Flags:* determines if the INT data is overflown, i.e., if there is no space left to add data in the packet, and if the INT report needs to be feed-backed to the network;
– *Remaining length:* determines the space left in the INT data structure before the packet overflows; and
– *Trace type:* determines the type of INT parameters each node adds, including node identifier, source and destination ports, timestamps, etc.

INT measurements are collected in a hop-by-hop or an E2E fashion. The hop-by-hop measurements are related to the links. Whilst, the E2E measurements are related to parameters that are collected only at the source and destination nodes. Such parameters include the counter of INT-enabled mea-

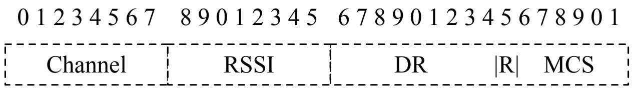



Fig. 5: Option format of the wireless link telemetry info.

surements and the timestamps for transmission and reception at source and destination nodes. Thus, other performance indicators such as E2E latency and jitter can be calculated. The one-way E2E latency is calculated as the difference of the timestamps for the transmission and reception at source and destination nodes, accordingly. On the other hand, the jitter can be calculated as the variation of the E2E latency on a per-packet basis. It is important to notice that nodes have to be time-synchronized to accurately derive such performance indicators.

In addition to the E2E latency and jitter as the options for the E2E INT header, new INT options have to be defined to quantify the performance of wireless links. For the hop-by-hop INT header, we define the options channel, data rate, MCS, processing delay, RSSI, and the re-transmissions flag. The processing delay stands for the time elapsed between the arrival time of INT-enabled packets at the wireless interface and the time when the measurements are added.

The wireless link telemetry option format is shown in Figure 5. The time at which the packet is received is included by setting bit 2 and 3 of trace-type, for setting the second and sub-second part, respectively. The *channel* field specifies the wireless channel at which the link is operating, the *RSSI*, the *DR*, the *R*, and the *MCS* specify the RSSI, data rate, retransmission flag, and the MCS used. We use Layer 2 encapsulation of INT information, with the INT header and option data field residing between Layer 2 and Layer 3 headers. In this manner, APs can add INT data to data packets between wireless nodes connected to the same AP, that is Layer 2 communication. As a downside, packet processing problems occur in case the INT enabled packet traverses a router that is not INT enabled. To this end, we limit the INT usage to private networks, where a single subnet is used and packets do not traverse any router.

Our INT implementation is composed of three modules: *source*, *intermediate*, and *sink*. These modules are installed on each node and, based on the roles they assume in a flow path (e.g., source, intermediate, or destination node), a certain module is triggered. The modules are detailed as follows.

#### *4.1.1 INT source module*

This module runs at the end devices and is responsible to initialize the hop-by-hop and E2E INT headers. In the case of a hop-by-hop INT header, this module is responsible for setting the bits of the *trace type* field for each information required to be collected along the data path. On the other hand, for the E2E INT header, this module initializes the INT options for encapsulating the source timestamp and the counter of INT-enabled data packets. Besides, this module is designed to interact with the application layer via a publish-subscribe messaging pattern. In this manner, the application layer can determine which parameters and frequency to be monitored based on the application QoS requirements. To speed up the interaction between the INT and application layer, the internal publish-subscribe messaging Application Programming Interface (API) is implemented using the ZeroMQ [8] with a brokerless model. Via the same interface, the application layer can determine the frequency of the INT as well as the flows to be monitored.

#### *4.1.2 INT intermediate module*

This module runs at the intermediate nodes (e.g., APs, switches, and routers) and is responsible to parse the INT source header and add more INT statistics to its packet. Based on the set bits of the *trace type* field of the hop-by-hop header, this module is responsible for adding the necessary INT parameters. As hop-by-hop parameters are related to the physical channel parameters (e.g., RSSI and channel), the module reads such parameters from the physical header. For this, we set the interface on monitor mode so the necessary information is read from the radio tap header. We use the monitor mode because the physical layer data is carried to higher layers on a packet-basis rather than polling the physical layer for parameters once every packet is received. As a consequence, this module is limited to parameters that the driver carries in the radio tap header. Based on the driver used, some parameters are interchanged. For example, some drivers add only the MCS index while others add the actual data rate. Once filled with more INT data, the INT-enabled packet is forwarded to the next hop.

#### *4.1.3 INT sink module*

This module runs at the end devices and it is responsible to parse, calculate, and send the INT measurements to the SDN controllers. First, this module collects all INT data along the flow path and calculates the E2E metrics, i.e., the E2E latency and jitter. Then, a JavaScript Object Notation (JSON) data structure is created, comprising the INT monitoring data. Last, this JSON data structure is sent to the MQTT broker via a publish-subscribe messaging system. The *sink* module is implemented using the MQTT Mosquitto library[9].

### 4.2 Control and Monitoring Applications

The applications we developed are implemented as Python modules, loaded either when the SD-RAN controller is started or using a Representational

[8] https://zeromq.org/

[9] https://mosquitto.org/man/libmosquitto-3.html

State Transfer (REST) API. The monitoring applications obtain statistics in a twofold manner. First, application QoS requirements and INT measurements are collected via the publish–subscribe messaging pattern. Second, by performing network polling at the APs using the 5G-EmPOWER primitives [49]. The two types of *monitoring* applications are detailed as follows:

- *Monitoring of application QoS requirements and INT*: The application requirements and the fine-grained monitoring information from INT are gathered via the MQTT broker. Thus, these monitoring applications must subscribe to the *INT* and applications' QoS requirements topics. The frequency that INT measurements arrive at the SD-RAN controller is given by the frequency that INT-enabled packets are reported.
- *Statistics polling at APs*: To perform network polling at the APs, monitoring applications establish communication with the APs through the *southbound interface*, implemented by the *OpenEmpower* protocol using a persistent TCP connection. With configurable polling intervals, monitoring applications gather statistics such as the throughput of slices on the APs, transmitted/received throughput of STAs, RSSIs, and channel utilization.

Nonetheless, we maintain a set of calculated metrics based on a configurable series of measurements. To avoid the masking effect in the presence of outliers on the latency-related metrics, the median of the measurement window is calculated, i.e., the Simple Moving Median (SMM). Otherwise, we calculate the Simple Moving Average (SMA). To enable high-level management of network resources, we implemented two control applications at the SD-RAN controller. The parameters used to configure each application is exposed and can be configured via a REST API. The details we present as follows:

- *Transmission policy control*: this application receives a set of parameters from a REST API to be applied as the transmission policy for a given AP/STA communication. The set of parameters include the set of MCSes that can be used by the rate selection algorithm, the RTS/CTS threshold, and the multicast policy. Transmission policies are specified based on the Layer 2 destination address. Therefore, a specific transmission policy per AP/STA communication can be created.
- *Airtime-based slicing*: this application adapts the airtime assigned to slices of an AP, i.e., quantum. In a loop fashion, this application reduces or increases the quantum configuration of slices according to their applications requirements. In this manner, the airtime is released from slices competing for resources that need to be prioritized or vice-versa. The parameters of this application include the increase and decrease rates that slices are adapted, the maximum and minimum quantum values to be assigned, the application and slice identifiers, and the configuration loop interval.

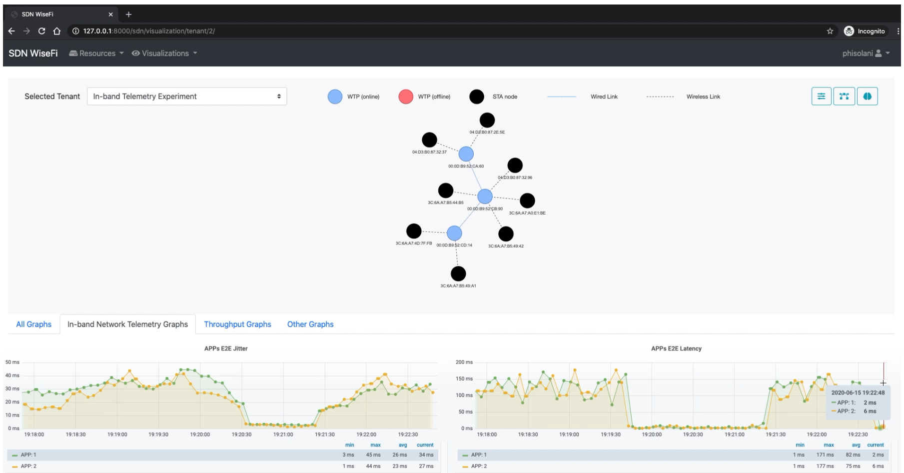



Fig. 6: Interactive GUI of the developed prototype.

### 4.3 SDN-WiseFi framework

We designed our prototype following the Model-View-Template (MVT) design pattern and we chose Python 3.7 with Django 3.0.7[10] as a development framework. By using a REST API as a management interface, our framework allows interaction with the *northbound interface* of different network controllers and therefore sets their expected behavior. Along with the configuration options of monitoring and control applications, our framework provides interactive visualizations regarding the topology and monitoring statistics. The interactive visualizations are done using Grafana and further embedded into our GUI. The monitoring statistics show the hop-by-hop and E2E statistics gathered with INT and the ones gathered through network polling at the AP. The implementation is available on Github [11] and Figure 6 presents the GUI of our developed prototype.

The visualization of the physical topology allows to identify the status and to which APs STAs are currently associated. We used the term WTP to refer to APs. At the top-middle of the physical topology are the symbols and colors used to indicate the status of WTPs (online/offline), STAs, and types of links (wired/wireless). Next to each node is shown their MAC addresses. At the top-right corner of the GUI, the buttons to access the configurations of the network. Below the physical topology visualization, the interactive charts show INT measurements at runtime as well as the statistics gathered from the 5G-EmPOWER primitives. In the example from Figure 6, the charts display the results from the E2E latency and jitter of two different applications. Although, by selecting an application or node, this information can be filtered.

[10] https://www.djangoproject.com/
[11] https://github.com/phisolani/SDNWiseFi

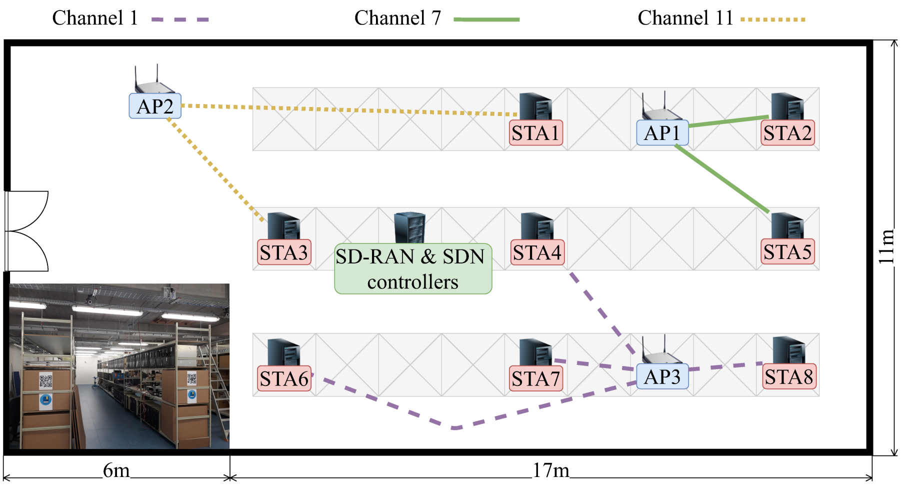



Fig. 7: Testbed experimental setup.

## 5 Evaluation

In this section, we evaluate our approach through experimentation conducted in a real-world testbed. The data presented in this study are openly available in FigShare [26]. We build our SDN-enabled infrastructure in the Industrial IoT lab[12] located in Ghent, Belgium. This area is a representative warehouse environment of 253 $m^2$. At the entrance of the warehouse, there is an open space of 6 m length for drone tests, navigation, and Virtual Reality (VR) experiments followed by an area with racks of 17 m length. The ceiling above is 2.5 m in height and provides railings with power and connectivity to integrate hardware and sensors. Three APs are installed and eight Intel NUC nodes, equipped with IEEE 802.11n cards, are used as STAs. A single computer, connected to the wired segment, is used to run all network controllers and systems. We have set the APs to operate on non-overlapping channels where AP 1, 2, and 3, are set to channels 7, 11, and 1, respectively. Figure 7 shows how nodes are displayed in the testbed.

### 5.1 Methodology & Workload

Throughout this evaluation, we generate four flows and analyze the results over five minutes. To represent both control and communication protocols, which are latency-sensitive, we generated ICMP and User Datagram Protocol (UDP) synthetic traffic using ping[13] and Iperf3[14]. To avoid INT data overflown, the

[12] https://www.ugent.be/ea/idlab/en/research/research-infrastructure/industrial-iot-lab.htm
[13] https://man7.org/linux/man-pages/man8/ping.8.html
[14] https://github.com/esnet/iperf

Table 1: Workload of applications during the experimentation.

| APP | SRC | DST | Flow | Data rate | Latency | Jitter |
|---|---|---|---|---|---|---|
| 1 | STA 6 | STA 7 | UDP | 1 Mbps | ≤20 ms | ≤10 ms |
| 2 | STA 4 | STA 8 | UDP | 4 Mbps | N/A | N/A |
| 3 | STA 5 | STA 2 | ICMP | 0.5 kbps | N/A | N/A |
| 4 | STA 1 | STA 3 | ICMP | 0.5 kbps | N/A | N/A |

buffer length for UDP packets is set to 500 bytes whereas ICMP is 64 bytes. The workflow parameters of each application and node pairs are given in Table 1.

To provide traffic isolation, we instantiate a dedicated slice per application. The quantum values of slices are initialized with 12 000 us each and the queue size of each slice is 500. Flows are generated between different pairs of STAs, therefore, include both uplink (UL) and downlink (DL) transmissions. UL and DL flows are characterized as the traffic generated from STAs to APs and from APs to STAs, accordingly. We created the two following experimental setups:

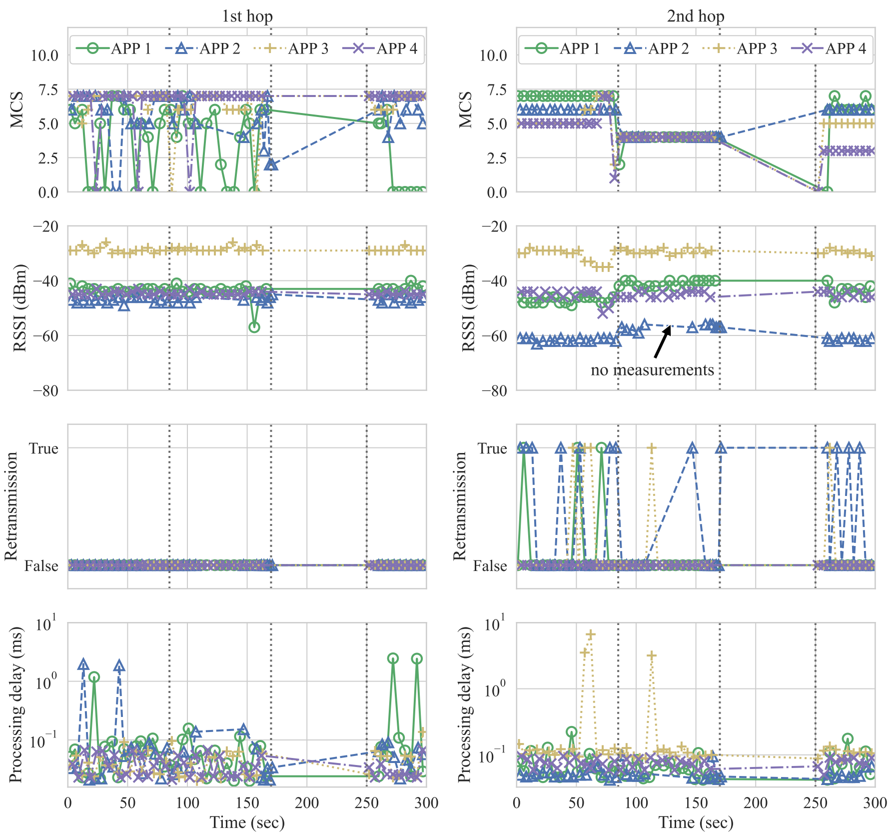
- Experiment 1: we investigate the impact of changing the MCSes range, used by APs' Minstrel rate control algorithm, on the reliability of flows.
- Experiment 2: as APs are usually the bottlenecks for the traffic in IEEE 802.11 networks, we analyze whether the QoS delivery can be enhanced by performing airtime-based network slicing at the RAN.

In both experiments, the SD-RAN controller polls for slice statistics every second while the INT-enabled packets are transmitted every five seconds. At the SD-RAN controller, the SMA and the SMM are calculated regarding the last ten measurements. To calculate the E2E latency of applications requires clock synchronization among our nodes and, to do so, we used the Network Time Protocol (NTP).

### 5.2 Experiment 1: MCS selection and reliability

In this experiment, we perform changes to the MCSes enabled for AP/STA communications, as being the network administrator. By lowering the MCS rates, reliability might be enhanced in scenarios with low Signal to Noise Ratio (SNR) conditions. For example, in scenarios such as in future digital factories, production lines are usually confined to specific locations under private ownership and therefore the level of interference in such private spaces can be controlled. However, because frames would require more airtime to be transmitted, this increases the probability of collisions and, hence, retransmissions. Although, we aim at showing how our approach assists the network administrator to understand the network behavior and address troubleshooting. The workload and QoS requirements are presented in Table 1 and the reconfigurations are presented in Table 2.

Figures 8 and 9 show the hop-by-hop and E2E network statistics collected from our INT implementation along the experiment time span. The markers




(a) 1st hop.

(b) 2nd hop.

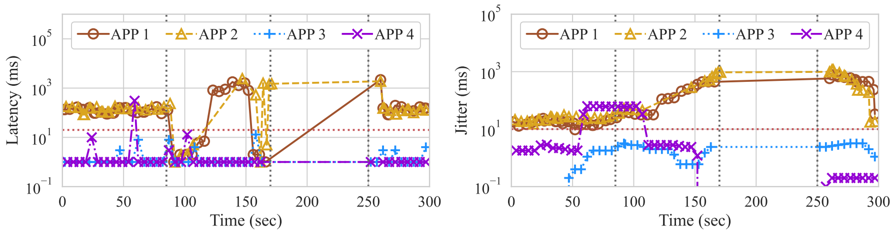
Fig. 8: Experiment 1 hop-by-hop INT measurements.

Table 2: Network reconfigurations performed in experiment 1.

| Event | Time (sec) | Description | Values |
|---|---|---|---|
| 1 | 85 | Low MCSes | 0 to 4 |
| 2 | 170 | High MCSes | 12 to 15 |
| 3 | 250 | Default MCSes | 0 to 15 |

show the presence of INT measurements related to the four applications we ran. The gray vertical dotted lines show the network reconfigurations performed by the network administrator and the red horizontal dotted lines show the QoS requirements.

Figures 8a and 8b show the statistics of the first and second hops. The first hop is related to STA/AP transmissions and the second is conversely, i.e., UL and DL transmissions. The statistics present information regarding




(a) E2E latency.

(b) E2E jitter.

Fig. 9: Experiment 1 E2E latency and jitter of applications.

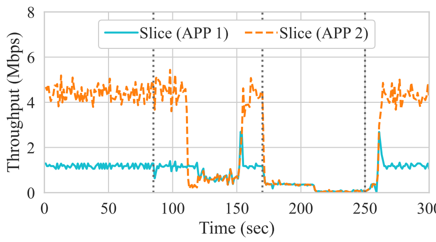



Fig. 10: Experiment 1 slice throughput of applications 1 and 2.

MCS, RSSI, the retransmission flag, and processing delay per node and application. During this experiment, although the RSSIs are stable and favorable, the MCSes vary among applications. Also, apart from a few outliers, the INT processing delay is often very low (below half millisecond) and the retransmission flag shows no retransmissions for first-hop transmissions. Second-hop statistics show, besides retransmissions, weaker RSSIs regarding application 2 (i.e., RSSIs perceived from AP 3 at STA 4).

In an endeavor to reduce the occurrence of retransmissions at the second hop, the administrator lowers the MCSes (event 1). As a result, the MCSes used are of the highest value allowed, i.e., MCS 4, rather than varying between 5 and 7. This has shown a first positive impact in terms of retransmissions where no retransmissions were reported during a period of ≈30 sec. However, application 2 has also not reported INT measurements for, approximately, the same period (highlighted in Figure 8b).

To understand the absence and delays of such measurements, we analyze the E2E statistics of applications and the throughput of slices for applications 1 and 2, which demand the most throughput among them. Figures 9a and 9b show that the E2E latency and jitter whereas Figure 10 presents the throughput of slices for applications 1 and 2. At first, the E2E latency is positively affected by lowering the MCSes. However, after a few seconds (≈30 sec), values dramatically increase while the throughput of slices decreases. Slices handling applications 1 and 2 have their throughput reduced by ≈14 % and ≈43 %,

Table 3: Network reconfigurations performed in experiment 2.

| Event | Time (sec) | Description | Values |
|---|---|---|---|
| 1 | 132 | APP 2 quantum decrease | 12 000 us to 2 us |
| 2 | 177 | APP 2 low quantum | 2 us |
| 3 | 271 | Default quantum | 12 000 us |

respectively. Whilst the throughput of slices recover, the E2E latency of applications 1 and 2 reach maximums of ≈1.8 sec and ≈2.4 sec where, previously, values were no higher than 200 ms. Besides, Figure 9b presents a higher jitter after lowering the MCSes.

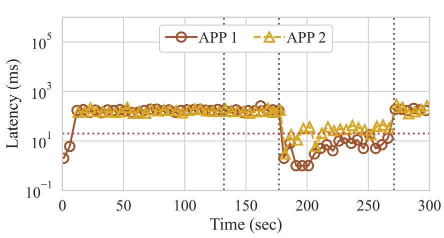
As a second reconfiguration (event 2), the administrator increases the allowed MCSes. However, the selected MCSes uses two spatial streams, and therefore require two discrete antennas at both transmitter and receiver. In our setup, nodes use only one spatial stream and therefore the MCSes can only range between 0 to 7 when STAs operate in the 2.4 GHz band. Because of such misconfiguration, INT measurements are not received at the controller while the throughput has dropped almost entirely. Realizing that, the administrator performs the third reconfiguration (event 3) where all MCSes are allowed to be selected by the Minstrel rate control algorithm. Thus, reestablishing connectivity for all nodes.

### 5.3 Experiment 2: network slicing and QoS

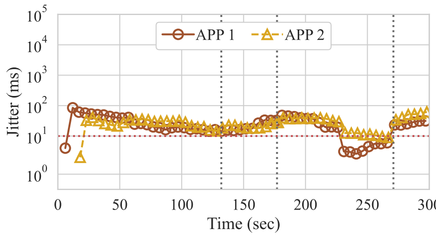
In this experiment, we analyze the impact of airtime-based network slicing on the E2E metrics of applications 1 and 2. In addition to higher demands, applications 1 and 2 compete for resources in the same AP. Figure 11 shows the E2E network statistics collected from our INT implementation along the experiment time span. The markers show the presence of INT measurements related to the four applications we ran. The gray vertical dotted lines show the network reconfigurations performed by the network administrator and the red horizontal dotted lines show the QoS requirements. The workload and QoS requirements are presented in Table 1 and the reconfigurations are in Table 3.

Figures 11a and 11b show the E2E latency and jitter for applications 1 and 2. In this experiment, we started application 2 a few seconds later (≈10 sec) which allowed us to verify that the QoS requirements of application 1 can be satisfied with the current placement and transmission capabilities of nodes. However, when applications 1 and 2 are active and therefore must share resources on the same AP (i.e., AP 3), the QoS requirements of application 1 are not satisfied. In this case, the E2E latency and jitter are above 20 ms and 10 ms with averages of 159 ms and 166 ms, respectively.

In an attempt to delay frames from application 2 and, hence, accelerate the dequeueing of frames from application 1, the network administrator performs a first reconfiguration (event 1 in Table 3). The network administrator activates

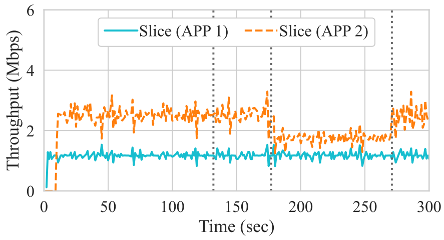



(a) E2E latency.

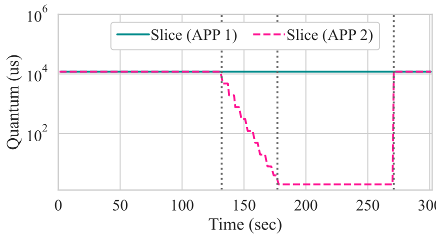



(b) E2E jitter.

Fig. 11: Experiment 2 E2E latency and jitter of applications.




(a) Throughput of slices.




(b) Quantum of slices.

Fig. 12: Experiment 2 slice throughput and quantum configurations for applications 1 and 2.

a control application at the SD-RAN controller that, at every five seconds, reduces the quantum of application 2 by 60 % of its current value in case the QoS requirements of application 1 are not satisfied. This, until the quantum for such slice, reaches a minimum value threshold of 2 us. Figure 12 presents the throughput of slices and their quantum configurations.

In Figure 12b, the quantum configuration for the slice handling application 2 has reached the minimum value after event 2 in second 177. As a consequence, frames are being held longer in the queue and eventually dropped if the capacity limit is reached. As the buffer length for UDP packets is set to 500 bytes and the queue size of each slice is 500, the maximum throughput that can be held within the queue is equivalent to 2 Mbps. Figure 12a presents when the throughput of application 2 drops due to the low quantum configuration. As a consequence, the E2E latency of both application drop and application 1 meets its QoS requirements. At around second 240, the E2E jitter is also met. Finally, to demonstrate again the impact of the quantum configurations on the E2E metrics, the default configuration is set once again (event 3). As a result, the throughput of application 2 is reestablished while the E2E latency and jitter of both applications increase.

## 6 Conclusion

In this paper, we have addressed an SD-RAN interactive management approach assisted by INT in IEEE 802.11 networks. With the centralized view of the network, our approach enables the network administrator to visualize fine-grained network metrics and interact with the network via airtime-based network slicing and the definition of transmission policies. IEEE 802.11 networks are still a viable option for indoor networks while latency-related have been often neglected. Therefore, we believe that SDN-tailored systems need to support fine-grained monitoring as well as enhanced QoS delivery. Differently from most work in the literature, our approach monitors the wireless links using INT and enables enhanced network awareness throughout interactive visualizations.

Inspired by future digital factories scenarios, we presented the potential of having an SDN centralized management system assisted by the fine-grained network information from INT. Throughout our experimentation, we demonstrated how our approach can assist network management at runtime. Apart from the longstanding need for enhanced network programmability, we can highlight two major challenges to be addressed. First, the low-level of the tunable parameters the lack of validation for reconfigurations make it difficult to identify misconfigurations. Although INT provides fine-grained network information, due to unknown network conditions, obstacles, and possible hardware failures, is yet a difficult task to correlate network configurations with the performance metrics. Second, as wireless conditions change quickly, management systems should react fast to performance degradation, e.g., in order of milliseconds or less. Therefore, although the known complexity of the wireless environment, there is a need for intelligent and, most likely, decentralized management solutions.

As future work, we plan to address the design of intelligent, distributed, and collaborative network management solutions. As a first step, we plan the design of a more deterministic scheduling algorithm for precise QoS service differentiation on the AP. As consequence, we plan to adapt our management approach to consider decentralized control actions targeting local and global end-goals.

**Acknowledgements** This research received partial funding from The European Union's Horizon 2020 Research and innovation program, under grant agreement No. 826284 (ProTego) and from the FLEXNET project: "Flexible IoT Networks for Value Creators" (Celtic 2016/3), in the Eureka Celtic-Next Cluster. This research has also received partial funding by the the Flemish FWO SBO S003921N VERI-END.com (Verifiable and elastic end-to-end communication infrastructures for private professional environments) project, the Flemish Government under the "Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen" program, and from the FWO-Flanders, under grant agreement #G055619N.

## Conflict of interest

The authors declare that they have no conflict of interest.

## References


1. 3GPP: Study on management and orchestration of network slicing for next generation network. Tech. Rep. TR 28.801 V15.0.0T, 3GPP (2017)
2. 3GPP: Service requirements for the 5G system. Technical Specification (TS) 22.261, 3rd Generation Partnership Project (3GPP) (2020). URL `https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3107`. Version 18.0.0
3. 5GPPP Architecture Working Group: View on 5G Architecture (2020). URL `http://doi.org/10.5281/zenodo.3265031`
4. Aleixendri, J.J., Betzler, A., Camps-Mur, D.: A practical approach to slicing Wi-Fi RANs in future 5G networks. In: 2019 IEEE Wireless Communications and Networking Conference (WCNC), pp. 1–6 (2019). DOI 10.1109/WCNC.2019.8885777
5. Azhari, S.V., Gurbuz, O., Ercetin, O.: Qos based aggregation in high speed ieee802.11 wireless networks. In: 2016 Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net), pp. 1–7 (2016). DOI 10.1109/MedHocNet.2016.7528418
6. Bannour, F., Souihi, S., Mellouk, A.: Distributed SDN Control: Survey, Taxonomy, and Challenges. IEEE Communications Surveys and Tutorials **20**(1), 333–354 (2018)
7. Bhamare, D., Kassler, A., Vestin, J., Khoshkholghi, M.A., Taheri, J.: IntOpt: In-Band Network Telemetry Optimization for NFV Service Chain Monitoring. In: ICC 2019-2019 IEEE International Conference on Communications (ICC), pp. 1–7. IEEE (2019)
8. Bosshart, P., Daly, D., Gibb, G., Izzard, M., McKeown, N., Rexford, J., Schlesinger, C., Talayco, D., Vahdat, A., Varghese, G., Walker, D.: P4: Programming protocol-independent packet processors. SIGCOMM Comput. Commun. Rev. **44**(3), 87–95 (2014). DOI 10.1145/2656877.2656890. URL `https://doi.org/10.1145/2656877.2656890`
9. Brockners, F., Bhandari, S., Pignataro, C., Gredler, H., Leddy, J., Youell, S., Mozes, D., Mizrahi, T., Lapukhov, P., Chang, R., Bernier, D., Lemon, J.: Data fields for in-situ oam, draft-ietf-ippm-iom-data-05 (2019)
10. Charfi, E., Gueguen, C., Chaari, L., Cousin, B., Kamoun, L.: Dynamic frame aggregation scheduler for multimedia applications in ieee 802.11n networks. Transactions on Emerging Telecommunications Technologies **28**(2), e2942 (2017)
11. Coronado, E., Riggio, R., Villa1ón, J., Garrido, A.: Lasagna: Programming Abstractions for End-to-End Slicing in Software-Defined WLANs. In: 2018 IEEE 19th International Symposium on" A World of Wireless, Mobile and Multimedia Networks"(WoWMoM), pp. 14–15. IEEE (2018)
12. Coronado, E., Riggio, R., Villalón, J., Garrido, A.: Wi-balance: Channel-aware user association in software-defined Wi-Fi networks. In: NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium, pp. 1–9 (2018)
13. De Schepper, T., Bosch, P., Struye, J., Donato, C., Famaey, J., Latré, S.: ORCHESTRA: Supercharging wireless backhaul networks through multi-technology management. Journal of Network and Systems Management (2020). DOI 10.1007/s10922-020-09528-x
14. De Schepper, T., Bosch, P., Zeljkovié, E., Mahfoudhi, F., Haxhibeqiri, J., Hoebeke, J., Famaey, J., Latré, S.: ORCHESTRA: Enabling Inter-Technology Network Management in Heterogeneous Wireless Networks. IEEE Transactions on Network and Service Management **15**(4), 1733–1746 (2018). DOI 10.1109/TNSM.2018.2866774
15. Geng, J., Yan, J., Ren, Y., Zhang, Y.: Design and Implementation of Network Monitoring and Scheduling Architecture Based on P4. In: Proceedings of the 2nd International Conference on Computer Science and Application Engineering, p. 182. ACM (2018)
16. Gomez, B., Coronado, E., Villalón, J., Riggio, R., Garrido, A.: User Association in Software-Defined Wi-Fi Networks for Enhanced Resource Allocation. In: Proc. of IEEE WCNC. Seoul, South Korea (2020)
17. Grunenberger, Y., Rousseau, F.: Virtual Access Points for Transparent Mobility in Wireless LANs. In: 2010 IEEE Wireless Communication and Networking Conference, pp. 1–6 (2010). DOI 10.1109/WCNC.2010.5506709
18. Gulenko, A., Wallschläger, M., Kao, O.: A Practical Implementation of In-Band Network Telemetry in Open vSwitch. In: 2018 IEEE 7th International Conference on Cloud Networking (CloudNet), pp. 1–4. IEEE (2018)

19. Haxhibeqiri, J., Isolani, P.H., Marquez-Barja, J.M., Moerman, I., Hoebeke, J.: In-band network monitoring technique to support sdn-based wireless networks. IEEE Transactions on Network and Service Management **18**(1), 627–641 (2021). DOI 10.1109/TNSM.2020.3044415
20. Heusse, M., Rousseau, F., Berger-Sabbatel, G., Duda, A.: Performance anomaly of 802.11b. In: IEEE INFOCOM 2003. Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE Cat. No.03CH37428), vol. 2, pp. 836–843 vol.2 (2003). DOI 10.1109/INFCOM.2003.1208921
21. Høiland-Jørgensen, T., Hurtig, P., Brunstrom, A.: PoliFi: Airtime Policy Enforcement for WiFi. In: 2019 IEEE Wireless Communications and Networking Conference (WCNC), p. 1–6. IEEE Press (2019). DOI 10.1109/WCNC.2019.8885440. URL `https://doi.org/10.1109/WCNC.2019.8885440`
22. Høiland-Jørgensen, T., Kazior, M., Täht, D., Hurtig, P., Brunstrom, A.: Ending the Anomaly: Achieving Low Latency and Airtime Fairness in WiFi. In: 2017 USENIX Annual Technical Conference (USENIX ATC 17), pp. 139–151. USENIX Association, Santa Clara, CA (2017). URL `https://www.usenix.org/conference/atc17/technical-sessions/presentation/hoilan-jorgesen`
23. Hyun, J., Van Tu, N., Hong, J.W.K.: Towards knowledge-defined networking using in-band network telemetry. In: NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium, pp. 1–7. IEEE (2018)
24. IEEE: IEEE Standard for Information technology–Local and metropolitan area networks–Specific requirements–Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications - Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements. IEEE Std 802.11e-2005 (Amendment to IEEE Std 802.11, 1999 Edition (Reaff 2003) pp. 1–212 (2005). DOI 10.1109/IEEESTD.2005.97890
25. IEEE: IEEE Standard for Information technology– Local and metropolitan area networks– Specific requirements– Part 11: Wireless LAN Medium Access Control (MAC)and Physical Layer (PHY) Specifications Amendment 5: Enhancements for Higher Throughput. IEEE Std 802.11n-2009 (Amendment to IEEE Std 802.11-2007 as amended by IEEE Std 802.11k-2008, IEEE Std 802.11r-2008, IEEE Std 802.11y-2008, and IEEE Std 802.11w-2009) pp. 1–565 (2009). DOI 10.1109/IEEESTD.2009.5307322
26. Isolani, P.H.: Sd-ran interactive management using int in ieee 802.11 networks (2021). URL `https://figshare.com/articles/dataset/SD-RAN_Interactive_Management_using_INT_in_IEEE_802_11_Networks/14838480/1`
27. Isolani, P.H., Cardona, N., Donato, C., Marquez-Barja, J., Granville, L.Z., Latré, S.: SDN-based Slice Orchestration and MAC Management for QoS delivery in IEEE 802.11 Networks. In: 2019 Sixth International Conference on Software Defined Systems (SDS), pp. 260–265 (2019). DOI 10.1109/SDS.2019.8768642
28. Isolani, P.H., Cardona, N., Donato, C., Pérez, G.A., Marquez-Barja, J.M., Granville, L.Z., Latré, S.: Airtime-Based Resource Allocation Modeling for Network Slicing in IEEE 802.11 RANs. IEEE Communications Letters **24**(5), 1077–1080 (2020). DOI 10.1109/LCOMM.2020.2977906
29. Isolani, P.H., Haxhibeqiri, J., Moerman, I., Hoebeke, J., Marquez-Barja, J.M., Granville, L.Z., Latré, S.: An SDN-based Framework for Slice Orchestration using In-Band Network Telemetry in IEEE 802.11. In: 2020 6th IEEE Conference on Network Softwarization (NetSoft), pp. 344–346 (2020). DOI 10.1109/NetSoft48620.2020.9165358
30. Isolani, P.H., Kulenkamp, D.J., Marquez-Barja, J.M., Granville, L.Z., Latré, S., Syrotiuk, V.R.: Support for 5G Mission-Critical Applications in Software-Defined IEEE 802.11 Networks. Sensors **21**(3) (2021). DOI 10.3390/s21030693. URL `https://www.mdpi.com/1424-8220/21/3/693`
31. Isolani, P.H., Kulenkamp, D.J., Marquez-Barja, J.M., Latré, S., Syrotiuk, V.R.: Delay-aware Slicing and MAC Management using MCDA in IEEE 802.11 SD-RANs (2021). (to appear)
32. Isolani, P.H., Wickboldt, J.A., Both, C.B., Rochol, J., Granville, L.Z.: Interactive monitoring, visualization, and configuration of OpenFlow-based SDN. In: 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), pp. 207–215 (2015). DOI 10.1109/INM.2015.7140294

33. Karaagac, A., De Poorter, E., Hoebeke, J.: In-Band Network Telemetry in Industrial Wireless Sensor Networks. IEEE Transactions on Network and Service Management (2019)
34. Kim, C., Sivaraman, A., Katta, N., Bas, A., Dixit, A., Wobker, L.J.: In-band network telemetry via programmable dataplanes. In: ACM SIGCOMM (2015)
35. Kim, D., An, S.: Throughput enhancement by dynamic frame aggregation in multi-rate wlans. In: 2012 19th IEEE Symposium on Communications and Vehicular Technology in the Benelux (SCVT), pp. 1–5 (2012). DOI 10.1109/SCVT.2012.6399389
36. Kohler, E., Morris, R., Chen, B., Jannotti, J., Kaashoek, M.F.: The Click Modular Router. ACM Transactions on Computer Systems (TOCS) **18**(3), 263–297 (2000). DOI 10.1145/354871.354874. URL `http://doi.acm.org/10.1145/354871.354874`
37. Lapukhov, P., Chang, R.: Data-plane probe for in-band telemetry collection. Internet-Draft draft-lapukhov-dataplane-probe-01, Internet Engineering Task Force (2016)
38. Luo, H., Shyu, M.: An Optimized Scheduling Scheme to Provide Quality of Service in 802.11e Wireless LAN. In: 2009 11th IEEE International Symposium on Multimedia, pp. 651–656 (2009). DOI 10.1109/ISM.2009.109
39. Maqhat, B., Dani Baba, M., Rahman, R.A., Saif, A.: Performance analysis of fair scheduler for a-msdu aggregation in ieee802.11n wireless networks. In: 2014 2nd International Conference on Electrical, Electronics and System Engineering (ICEESE), pp. 60–65 (2014). DOI 10.1109/ICEESE.2014.7154608
40. McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S., Turner, J.: OpenFlow: Enabling Innovation in Campus Networks. SIGCOMM Computer Communincation **v.38**(2), 69–74 (2008)
41. Metter, C., Burger, V., Hu, Z., Pei, K., Wamser, F.: Towards an Active Probing Extension for the ONOS SDN Controller. In: 2018 28th International Telecommunication Networks and Applications Conference (ITNAC), pp. 1–8. IEEE (2018)
42. Moura, H., Alves, A.R., Borges, J.R., Macedo, D.F., Vieira, M.A.: Ethanol: A software-defined wireless networking architecture for ieee 802.11 networks. Computer Communications **149**, 176 – 188 (2020). DOI https://doi.org/10.1016/j.comcom.2019.10.010. URL `http://www.sciencedirect.com/science/article/pii/S0140366419305997`
43. Pang, W.L., Chieng, D., Ahmad, N.N.: Adaptive priority sliding admission control and scheduling scheme for dcf and edca wlans. Wireless Personal Communications **70**(1), 295–321 (2013). DOI 10.1007/s11277-012-0695-2. URL `https://doi.org/10.1007/s11277-012-0695-2`
44. Postel, J.: Internet control message protocol. STD 5, RFC Editor (1981). URL `http://www.rfc-editor.org/rfc/rfc792.txt`
45. Quan, L., Heidemann, J., Pradkin, Y.: Trinocular: Understanding internet reliability through adaptive probing. In: ACM SIGCOMM Computer Communication Review, vol. 43, pp. 255–266. ACM (2013)
46. Richart, M., Baliosian, J., Serrat, J., Gorricho, J.: Resource Slicing in Virtual Wireless Networks: A Survey. IEEE Transactions on Network and Service Management **13**(3), 462–476 (2016). DOI 10.1109/TNSM.2016.2597295
47. Richart, M., Baliosian, J., Serrat, J., Gorricho, J.L., Agüero, R.: Slicing With Guaranteed Quality of Service in WiFi Networks. IEEE Transactions on Network and Service Management **17**(3), 1822–1837 (2020). DOI 10.1109/TNSM.2020.3005594
48. Richart, M., Baliosian, J., Serrati, J., Gorricho, J., Agüero, R., Agoulmine, N.: Resource allocation for network slicing in WiFi access points. In: 2017 13th International Conference on Network and Service Management (CNSM), pp. 1–4 (2017). DOI 10.23919/CNSM.2017.8256046
49. Riggio, R., Marina, M.K., Schulz-Zander, J., Kuklinski, S., Rasheed, T.: Programming Abstractions for Software-Defined Wireless Networks. IEEE Transactions on Network and Service Management **12**(2), 146–162 (2015). DOI 10.1109/TNSM.2015.2417772
50. Sarret, M.G., Ashta, J.S., Mogensen, P., Catania, D., Cattoni, A.F.: A multi-qos aggregation mechanism for improved fairness in wlan. In: 2013 IEEE 78th Vehicular Technology Conference (VTC Fall), pp. 1–5 (2013). DOI 10.1109/VTCFall.2013.6692041
51. Schulz-Zander, J., Mayer, C., Ciobotaru, B., Schmid, S., Feldmann, A.: OpenSDWN: Programmatic Control over Home and Enterprise WiFi. In: ACM Sigcomm Symposium on SDN Research (SOSR) (2015). URL `http://eprints.cs.univie.ac.at/5561/`

52. Serkantul: Prometheus int exporter (2016). URL https://github.com/serkantul/prometheus_int_exporter
53. Serrano, P., Banchs, A., Patras, P., Azcorra, A.: Optimal configuration of 802.11e edca for real-time and data traffic. IEEE Transactions on Vehicular Technology **59**(5), 2511–2528 (2010). DOI 10.1109/TVT.2010.2043274
54. Seytnazarov, S., Kim, Y.: Qos-aware adaptive a-mpdu aggregation scheduler for voice traffic in aggregation-enabled high throughput wlans. IEEE Transactions on Mobile Computing **16**(10), 2862–2875 (2017). DOI 10.1109/TMC.2017.2672994
55. Shirali-Shahreza, S., Ganjali, Y.: Traffic statistics collection with FleXam. In: ACM SIGCOMM Computer Communication Review, vol. 44, pp. 117–118. ACM (2014)
56. Suresh, L., Schulz-Zander, J., Merz, R., Feldmann, A., Vazao, T.: Towards Programmable Enterprise WLANS with Odin. In: Proceedings of the First Workshop on Hot Topics in Software Defined Networks, HotSDN '12, pp. 115–120. ACM, New York, NY, USA (2012). DOI 10.1145/2342441.2342465. URL http://doi.acm.org/10.1145/2342441.2342465
57. Van Adrichem, N.L., Doerr, C., Kuipers, F.A.: Opennetmon: Network monitoring in openflow software-defined networks. In: 2014 IEEE Network Operations and Management Symposium (NOMS), pp. 1–8. IEEE (2014)
58. Van Tu, N., Hyun, J., Hong, J.W.K.: Towards ONOS-based SDN monitoring using in-band network telemetry. In: 2017 19th Asia-Pacific Network Operations and Management Symposium (APNOMS), pp. 76–81. IEEE (2017)
59. Van Tu, N., Hyun, J., Kim, G.Y., Yoo, J.H., Hong, J.W.K.: INTCollector: A High-performance Collector for In-band Network Telemetry. In: 2018 14th International Conference on Network and Service Management (CNSM), pp. 10–18. IEEE (2018)
60. Vassilaras, S., Gkatzikis, L., Liakopoulos, N., Stiakogiannakis, I.N., Qi, M., Shi, L., Liu, L., Debbah, M., Paschos, G.S.: The Algorithmic Aspects of Network Slicing. IEEE Communications Magazine **55**(8), 112–119 (2017). DOI 10.1109/MCOM.2017.1600939
61. Wang, B., Su, J.: FlexMonitor: A Flexible Monitoring Framework in SDN. Symmetry **10**(12), 713 (2018)
62. Weis, B., Brockners, F., Hill, C., Bhandari, S., Govindan, V., Pignataro, C., Gredler, H., Leddy, J., Youell, S., Mizrahi, T., Kfir, A., Gafni, B., Lapukhov, P., Spiegel, M.: EtherType Protocol Identification of In-situ OAM Data, draft-weis-ippm-ioam-eth-01 (2019)
63. Wickboldt, J.A., Jesus, W.P.D., Isolani, P.H., Both, C.B., Rochol, J., Granville, L.Z.: Software-defined networking: management requirements and challenges. IEEE Communications Magazine **53**(1), 278–285 (2015). DOI 10.1109/MCOM.2015.7010546
64. Xia, D., Hart, J., Fu, Q.: Evaluation of the Minstrel rate adaptation algorithm in IEEE 802.11g WLANs. In: 2013 IEEE International Conference on Communications (ICC), pp. 2223–2228 (2013). DOI 10.1109/ICC.2013.6654858
65. Yap, K.K., Sherwood, R., Kobayashi, M., Huang, T.Y., Chan, M., Handigol, N., McKeown, N., Parulkar, G.: Blueprint for Introducing Innovation into Wireless Mobile Networks. In: Proceedings of the Second ACM SIGCOMM Workshop on Virtualized Infrastructure Systems and Architectures, VISA '10, p. 25–32. Association for Computing Machinery, New York, NY, USA (2010). DOI 10.1145/1851399.1851404. URL https://doi.org/10.1145/1851399.1851404
66. Zeljković, E., De Schepper, T., Bosch, P., Vermeulen, I., Haxhibeqiri, J., Hoebeke, J., Famaey, J., Latré, S.: ORCHESTRA: Virtualized and programmable orchestration of heterogeneous WLANs. In: 2017 13th International Conference on Network and Service Management (CNSM), pp. 1–9 (2017). DOI 10.23919/CNSM.2017.8255999
67. Zeljković, E., Slamnik-Kriještorac, N., Latré, S., Marquez-Barja, J.M.: ABRAHAM: Machine Learning Backed Proactive Handover Algorithm Using SDN. IEEE Transactions on Network and Service Management **16**(4), 1522–1536 (2019)
68. Zhao, D., Zhu, M., Xu, M.: SDWLAN: A flexible architecture of enterprise WLAN for client-unaware fast AP handoff. In: Fifth International Conference on Computing, Communications and Networking Technologies (ICCCNT), pp. 1–6 (2014). DOI 10.1109/ICCCNT.2014.6963100
69. Zhao, D., Zhu, M., Xu, M.: Supporting "One Big AP" illusion in enterprise WLAN: An SDN-based solution. In: 2014 Sixth International Conference on Wireless Communications and Signal Processing (WCSP), pp. 1–6 (2014). DOI 10.1109/WCSP.2014.6992121

**Pedro Heleno Isolani** received the B.Sc. degree in information systems from the State University of Santa Catarina, Brazil, in 2012, and the M.Sc. degree from the Federal University of Rio Grande do Sul, Brazil, in 2015, focused on the management of software-defined networking. He recently received the Ph.D. degree in computer science from the the University of Antwerp - imec, Belgium. His current research interests include network functions virtualization, software-defined networking, and programmability of wireless medium access control protocols.

**Jetmir Haxhibeqiri** received the master's degree in engineering (information technology and computer engineering) from RWTH Aachen University, Germany, in 2013, the Ph.D. degree in engineering computer science from Ghent University in 2019, with his research on flexible and scalable wireless communication solutions for industrial warehouses and logistics applications. He is currently a Postdoctoral Researcher with the Internet Technology and Data Science Lab, Ghent University and imec. His current research interests include wireless communications technologies (IEEE 802.11, IEEE 802.15.4e, and LoRa) and their application, IoT, wireless time-sensitive networking, in-band network monitoring, and wireless network management.

**Ingrid Moerman** received a degree in electrical engineering in 1987, and a Ph.D. degree from Ghent University in 1992, where she became a part-time Professor in 2000. She is a Staff Member with IDLab, Core Research Group, imec with research activities embedded with Ghent University and the University of Antwerp. Her main research interests include cooperative and intelligent radio networks, real-time software-defined radio, time-sensitive networks, dynamic spectrum sharing, coexistence across heterogeneous wireless networks, vehicular networks, open-source prototyping platforms, software tools for programmable networks, next-generation wireless networks (5G/6G), and experimentally supported the research.

**Jeroen Hoebeke** received the master's degree in engineering computer science from Ghent University in 2002, the Ph.D. degree in engineering computer science in 2007, with his research on adaptive ad hoc routing and Virtual Private Ad Hoc Networks. He is currently an Associate Professor with the Internet Technology and Data Science Lab of Ghent University and imec. He is conducting and coordinating research on wireless (IoT) connectivity, embedded communication stacks, deterministic wireless communication, and wireless network management. He has authored or co-authored more than 150 publications in international journals or conference proceedings.

**Lisandro Zambenedetti Granville** is a Professor at the Federal University of Rio Grande do Sul. His interests include network management, software-defined networking, and network functions virtualization. He served as the TPC Co-Chair of IFIP/IEEE DSOM 2007 and IFIP/IEEE NOMS 2010, the General Co-Chair of IFIP/IEEE CNSM 2014, the TPC Co-Chair of IEEE Net-

Soft 2018, and the TPC Vice-Chair do IEEE ICC 2018. He is the President of the Brazilian Computer Society.

**Steven Latré** received a Master of Science degree in computer science and a Ph.D. degree in computer science engineering from Ghent University, Belgium. He is an Associate Professor at the University of Antwerp and the Director of the Research Centre imec, Belgium. He is leading the IDLab Antwerp Research Group (85+ members), which is performing applied and fundamental research in the area of communication networks and distributed intelligence. His personal research interests are in the domain of machine learning and its application to wireless network optimization.

**Johann M. Marquez-Barja** is currently a Professor at the University of Antwerp and a Professor with IMEC, Belgium. His main research interests include 5G advanced architectures, edge computing, flexible and programmable future end-to-end networks, and IoT communications and applications. His research interests also include vehicular communications, mobility, and smart cities deployments. He is also interested and active in education development, being actively involved in different research actions to enhance engineering education, in particular remote experimentation in e-learning systems. He is a member of ACM, and a Senior Member of the IEEE Communications Society, IEEE Vehicular Technology Society, and IEEE Education Society, where he participates in the board of the Standards Committee.