5G Case Study of Internet of Skills: Slicing the Human Senses

Maria A. Lema, Konstantinos Antonakoglou, Fragkiskos Sardis, Nantachai Sornkarn, Massimo Condoluci, Toktam Mahmoodi and Mischa Dohler
Centre for Telecommunications Research, Dept. of Informatics
King's College London,
{maria.lema_rosas, konstantinos.antonakoglou, fragkiskos.sardis, nantachai.sornkarn, massimo.condoluci, toktam.mahmoodi, mischa.dohler}@kcl.ac.uk

***Abstract*—5G is all about integrating new industries in the design of this whole new generation, where the mobile broadband connection is not the one and only use case being focused. This article focuses on a practical implementation of a healthcare oriented Internet of Skills application, where the doctor is able to perform remote diagnosis and palpation with the use of cutting edge haptic technology. We present an examination of the main medical socio-economic drivers, as well as the description of specific technologies used in this practical demonstration. All this, with the main objective of delivering a proof of concept for the design and planning of multi-modal communications in 5G.**

***Index Terms*—medical industry, proof of concept, 5G, Tactile Internet, SDN, haptics**

I. Introduction

In the past decades, the requirements of traditional mobile communications standards were focused in just one use case and application, the broadband access, with the main objective of providing good coverage and high bandwidth for a satisfactory user experience all the time, everywhere. The road to 5G has already changed this paradigm, and several societal drivers are stressing the need for a more reliable and low latency network. For most industries 5G will be a game changer [1] and will enable compelling new offerings. Some industries have already considered a number of potential applications that use tele-presence or remote support. In particular, providing the ability to introduce the sense of touch is gaining heightened interest specifically towards mission critical use cases, which primary objective is to transmit skills and enable real-time interactive systems providing major benefits to society and industry as a whole [2].

The haptic sensing aspect to support good remote operation experiences is being widely investigated in the robotic systems community. In the recent years there has been an increased attention to develop soft robots which are able to capture the sense of touch as well as provide higher flexibility and increased safety in robot-assisted tele-operations. One use case example of stiffness sensors and soft robotics is robot-assisted minimally invasive surgery, where the presence of tactile feedback has proven to improve the clinical outcomes. In particular, the experience is enhanced by being able to measure the pressure applied during gripping and palpation, and helps to find hard abnormalities such as tumours [3].

The health sector has been one of the latest industries to introduce the use of Information and Communication Technologies (ICT). The medical industry is now progressively using technological advances to improve the overall patient experience and reduce health-care costs. General trends of the industry include the decentralization of hospitals, where medical care can be provided at home or on the move with the use of telemedicine, and an overall reduction to the economic cost for health services through the use of technology. The increased access to diagnostic, treatment and preventative care is reflected in a massive cost reduction for the health care system [4], [5].

The Internet of Skills puts together all these trends in the different communities: robotics, industries, society and communications, and enables a new way of communication. Such a real-time interactive system can be interpreted as a closed loop system, where skills need first to be captured in the master domain, then transmitted, successfully reproduced in the slave domain, and finally, the feedback closes the loop. In the master domain skills can be captured with the use of a human system interface (HSI) which is a haptic device (i.e., wearable or enhanced robotic-sensing) that translates the human input into specific instructions of movement and pressure, for example being able to follow the precise movement of the hands. Once the expert's skills are accurately captured, they are transmitted using a low latency and reliable communications system. In the slave domain, the information captured is reproduced using a controlled robot through commands. The control domain is able to accurately sense the remote environment and the controlled robot generates feedback signals which are sent back to the master domain, closing the loop [2]. A truly immersive and interactive Internet of Skills system needs to support both haptic and visual feedback and allow the inclusion of augmented experiences with the use of mixed reality; this is referred to as multi-modal communications system. Fig. 1 shows a functional representation of the high level vision of the Internet of Skills.

Multi-modal traffic in critical communications require flow differentiation and careful traffic management, as well as supporting a diversity in quality of service (QoS) requirements. One of the key aspects of the next generation networks is to guarantee flexibility to support a dynamic QoS handling, which is mainly enabled through network programmabil-

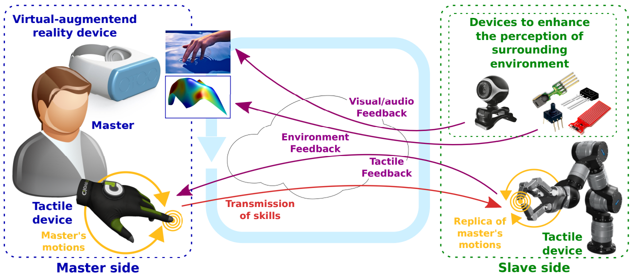


Fig. 1: Internet of Skills high level vision

ity, represented by the Software Defined Network (SDN) paradigm. SDN decouples the control and data planes, and logically centralises the network intelligence and abstracts the network infrastructure, and as a result the SDN control plane is programmable. The main architecture of SDN consists of three parallel layers: infrastructure, control and application layers, and a fourth vertical layer which interfaces with the main layers providing control and management functions: the management layer [6]. The management plane sets, manages and modifies the rules applied to network switches to handle the QoS.

The main objective of this work is to present a proof of concept prototype of a 5G Internet of Skills application where we are able to show how 5G can support tele-presence and the transmission of skills in real-time. This proof of concept has been designed to show two major aspects of 5G:

- Remote sensing with the use of low latency networks: the prototype presented in this article shows the potentials of adding the sense of touch in a tele-operation system, via state-of-the-art sensing techniques and using a tactile Glove as the main controlling engine which is also capable of reproducing the tactile information.
- Network Slicing with the use of SDN: we set-up the transmission of multiple flows, in this case control, vision and touch, through isolated communication channels. SDN allows to dynamically control their independent QoS depending on the network status and demand of each one of these flows.

The rest of this paper is outlined as follows: Section II describes the main motivation behind the remote surgery use case, it gives a brief outline of the current state of tele-operation and how our proof of concept improves upon it, Section III describes in detail all the aspects of this proof of concept, Section IV includes a performance evaluation discussion and finally, Section VI concludes this work and discuss possible future plans.

II. Remote Surgery: Motivation and Prototyping

Tele-surgery or remote surgery, is a natural evolution of tele-mentoring with the use of surgical robots. In tele-mentoring, on-site healthcare professionals are guided by another in a remote location, and the level of mentoring may vary from verbal guidance to remotely controlling the robotic arm. In the case of full remote surgery, the primary surgeon is at a site remote from the patient and all surgical tasks are performed by a robot controlled remotely. To enable this, and carry out the surgical procedure, the computer console (i.e., control console of a robot like the DaVinci) and the remote surgical device are connected by a high-speed ultra-reliable communications network [7]. Remote operation or consultation brings huge advantages to the healthcare system, as it allows the decentralisation of hospitals, and provide enhanced solutions for the remote care.

As of today, medicine relies very much on minimally invasive surgery, and concepts like laparoscopy or products like the DaVinci robot are well accepted in both public and private healthcare systems, and have proven to improve outcomes by reducing hospital stay, recovery time, pain, and post-operative impact. Specifically in the remote care context, the use of robots for surgery or consultation is particularly interesting, since it combines the benefits of minimally invasive interventions with the decentralization of hospitals. The Internet of Skills is a key enabler of remote operation or consultation, where the skills of the healthcare professional are captured, transmitted and reproduced in the remote end.

However, when substituting the doctors hands with a robotic arm, the surgeon loses the sense of touch, which is essential for palpation and locating hard tissue or nodules. Commercial equipment currently used in laparoscopic or robotic surgery lacks the capability of recreating this sense of touch, and the only feedback information the specialist has to rely on are the verbal and visual ones. Current research in the robotics field is focused in the design of high-precision force or stiffness sensors, that can recreate the sense of touch when manipulating a robot [3]. More insight on the haptic perception is given in the following sections.

In an attempt to combine the advantages of both remote medical practice and the enhanced haptic feedback for minimally invasive surgery, in this practical implementation, we have decoupled the overall operating system. On the one hand there is the master domain, where the healthcare professional controls the operating side with the use of a wearable device. On the other hand, there is the slave domain, where a robotic probe performs a palpation task and senses the level of stiffness. In particular, when operating the robotic probe (or robotic finger), the doctor receives stiffness information of the palpation area, which is reproduced in the wearable device. With this, the doctor will identify accurately and in real time localization of the hard nodules within the soft tissue, thanks to the stiffness information. All of this is done through a reliable high-speed communications network.

III. Sensory Perception and 5G Networks

Haptic perception relies greatly on kinaesthesia as well as on touch. While kinaesthesia is the reason humans are aware of the weight of objects, pressure, spatial position and movement of their body parts, touch is using various cutaneous receptors under the skin to sense modalities such as texture, pressure, pain and temperature [8].

The future generation of networks, namely 5G networks, will support multi-modal communication with high Quality of Experience (QoE) by enabling the exchange of audio, video, kinaesthetic and tactile data among a multitude of devices in real time [2]. Among these four modalities, vision and kinaesthesia have the highest requirements in terms of bandwidth and low latency respectively. A high-definition 1080p video stream can reach up to more than 80 Mbps when minimally compressed, and a complex movement of a human palm touching an object may require latency down to 1 ms. Aside from the key performance indicators. Another important requirement is the synchronization of all the data streams. Of course, as with current audio-visual communication systems, data compression and perceptual data reduction can be applied to both kinaesthetic and tactile information.

By combining video, audio and touch, we re-create a multi-modal communication, where the doctor is able to control the movements of the robotic finger using a haptic device, and can receive both visual and tactile feedback. Multi-modal communications requires a per flow QoS management, since every traffic type has different requirements: the real-time nature of the use case requires ultra low latency for all flows, but in terms of capacity, video and tactile traffic are very different. Also, the robot control information and the tactile feedback are considered critical traffic, and should be treated with higher priority.

To satisfy the QoS and ensure that critical traffic is always delivered, we use SDN to isolate the different flows on the network and dedicate a slice of the bandwidth to each flow, along with a high priority queue for minimising latency, when needed. In the following, we detail the specific hardware and testbed configurations for the Internet of Skills proof of concept.

A. *Creating the Prototype*

Controlled or Slave Domain: In this proof of concept demonstration, we used a state-of-the-art robotic probe platform developed in [3] as the haptic sensing device. The design of this robotic probe is largely inspired by the human finger and it abstracts its components to represent the way humans sense force. The robotic probe consists of two parts, a controllable stiffness joint, which represents the human metacarpophalangeal joints, and a force sensor at the base, which represents the function of a human tendon.

For the purpose of this demonstration, the stiffness of the joint is kept constant and the force felt at the base is measured using an ATI Nano17 Force/Torque (F/T) transducer (SI-12-0.12, ATI Industrial Automation, USA, resolution of 0.015 Nmm). The F/T transducer connects to the computer via a data acquisition card PCIe-6320 from National Instruments. The sampling rate of the force measurement is taken at 1000 Hz. To allow the robotic probe to move in space it is mounted on a *XY Table*, that guides the movement along the three axis (x,y and z), the XY-linear stage ANT130 (Aerotech Inc., resolution of 1nm), which is connected to the computer via Ethernet. Finally, the software for moving the probe and measuring the force data is programmed in LabView.

The soft tissue sample used in the demo was made of artificial materials replicating the tumour and biological tissue. We use artificial materials because the physical properties of real biological tissue can change over time. In general, the contrast in the stiffness between the malignant tumour and healthy fibroglandular breast tissue is quite prominent. Therefore, we can use the soft silicone and solid plastic material to approximately replicate the healthy tissue and the tumour respectively. The soft phantom was fabricated from the soft clear silicone elastomer RTV27905 (part A and B) from Techsil Company Limited. An ABS plastic bead (referred to as 'hard nodule') of size 15mm diameter was used to represent a tumour inside soft phantom. The soft silicone phantom with an embedded hard nodule was made by inserting the hard nodule in between two layers of the phantom.

(a) Forward (b) Backwards (c) Reset

Fig. 2: Glove gestures for robot commands

Master Domain: The proposed system made use of a tactile device that can act both as a tactile actuator and as a touch sensor. The tactile device is a commercial glove (GloveOne [9]) and can be connected to a PC using a USB cable or via a Bluetooth connection. It consists of two parts: a vibration subsystem which uses piezoelectric vibrators, placed under the fingertips and the palm, mainly used to recreate the sense of touch, and four conductive areas located at the center of the palm, thumb, index and middle finger, that when in contact with each other create closed circuits. These closed circuits represent different gestures as shown in Fig. 2, which we use to communicate the different movement commands to the probe.

For the haptic application, we set up a set of hand gestures using the glove to control the overall system. The first set of gestures is designed to control the movements of the robotic probe through XY Table, each unique gesture makes the robotic probe move backwards or forwards over the soft tissue sample. The gesture information is sent using UDP packets to the LabView software, which moves the robot accordingly and returns readings captured by the robotic probe back to the haptic application, which translates them into vibrations on the glove.

The second set of gestures is designed to control the configuration of the system. In particular, two messages are sent: the first one is sent to LabView and resets the robots position, while the second message is directed at a Python script that changes the configuration of the SDN controller. Detailed information on the configuration of the SDN controller is given in the following section.

Visual Feedback: Two camera feeds are also included in the testbed with the main purpose of sending a visual feedback of the robotic probe to the master domain. The main camera feed is full HD at 30fps while the secondary feed is 720p, also at 30fps. The cameras we use are Microsoft LifeCam HD

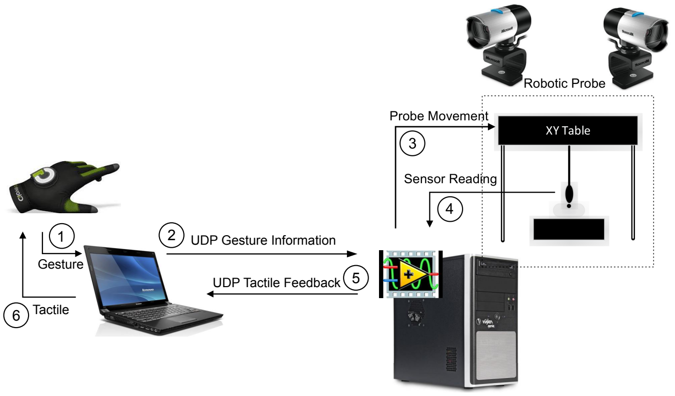


Fig. 3: Prototype demo setup

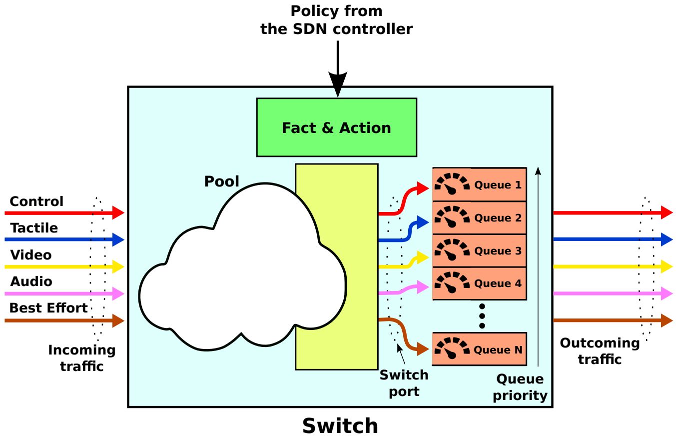


Fig. 4: QoS mechanism using queues and meters in SDN

webcams that feature a built-in hardware encoder and connect to the computer via USB. This reduces processing delays as the computer does not need to encode the video stream. For low latency video streaming we use UltraGrid[1] [10] and it is configured to transmit the raw video as it receives it from the cameras, to avoid increased delay in software compression.

Figure 3 depicts the system and details the message exchange from the two sides of the communication.

B. Connecting with SDN

The main objective of this proof of concept is to implement a full isolation of traffic so that independent QoS metrics can be applied on a per-flow basis. According to the NGMN Alliance (Next Generation Mobile Networks), a network slice instance is a set of network functions and resources to run these functions that may be fully or partially isolated from other network slice instance [11]. Hence, by isolating the traffic and allowing to treat it independently with different forwarding policies, we create a proof of concept of the physical and logical resource reservation aspect of a network slicing instance. At this point, the concept of isolated network functions is left out of scope of this work, and we focus on the resources to isolate a sub-network instance.

To provide a good dynamic QoS management we leverage on the SDN capabilities of network configuration. QoS can be managed through *OpenFlow*, the most consolidate communication interface between the control and infrastructure layers, and it is mainly used to access the infrastructure layer and modify the switch's flow table. The flow tables kept in the switch contain the rules to apply to each flow, which are programmed by the SDN controller and pushed to the infrastructure using OpenFlow. Queues and meters are two OpenFlow features useful for QoS management and traffic isolation. Packets arriving to a switch can be identified as a particular flow and set to a queue which has a predefined configured transmission rate on an output port. Meters allow to set a transmission rate threshold that can trigger other functions once the threshold is exceeded. One packet arriving to the switch may be assigned to multiple meters which trigger different functions once the rate is exceeded. Figure 4 shows the QoS mechanism using meters and queues system in an SDN switch.

In our setup, we use an OpenFlow switch along with the OpenDaylight(ODL) SDN controller. ODL is a production quality controller supported by the Linux foundation and has a modular design that allows us to load only the set of functions we wish to use. For this proof of concept, we create a minimal setup that allows us to push flows to the switch and also enables a representational state transfer (REST) interface for pushing configurations onto the controller.

Our setup consists of a PC running LabView for controlling the robot over the network, along with Ultragrid for streaming video from the two cameras. We also run a Linux virtual machine (VM) for ODL which directly binds to one of the network interfaces of the PC for Openflow communication with the switch. The second network interface of the PC is used for the traffic coming from the camera, LabView and glove flows. Additionally, a virtual host only network connects the Linux VM to the host operating system (OS) for passing configuration messages to the SDN controller. Finally, we use a laptop for running the haptic software and receiving the camera feeds. The physical setup along with the different connections is illustrated in Figure 5.

We distinguish the traffic between different applications based on the source and destination IP addresses as well as the TCP/UDP ports. Since our implementation uses traffic shaping on each slice, this set up can be considered as a use case of an edge switch where inbound traffic is placed on a slice and a set of configurations is applied in order to guarantee a minimum rate, and also implement a cap so that it wont interfere with other traffic. The minimum rate is configured using strict priority queueing where the highest priority queue is served first until it is empty, and subsequently the same process is repeated for all the other queues. OpenFlow meters are configured in each slice with a Drop action, once a predefined threshold is exceeded, limiting the amount of bandwidth that configured slice can use. Since queues control the egress rate while meters control the ingress rate, this setup allows us to control both aspects of the traffic per physical port. The slicing configuration is presented in Table I, along with

[1] *UltraGird* is a software implementation of high-quality low-latency video and audio transmissions using commodity PC and Mac hardware

TABLE I: Slice setup on physical ports, queues and meters

Nodes	Flow Direction	Port/Queue	Meter/Action
PC to Robot	LabView to XY	P1/Q7	512Kbps/Drop
Robot to PC	XY to LabView	P2/Q7	512Kbps/Drop
PC to Laptop	CamA to UltraGrid	P3/Q7	80Mbps/Drop
PC to Laptop	CamB to UltraGrid	P3/Q7	60Mbps/Drop
Laptop to PC	Glove to LabView	P2/Q1	–
Laptop to PC	Glove to ODL	P2/Q2	–
PC to Laptop	LabView to Glove	P3/Q1	–

TABLE II: Queue configuration per egress port

Port/Queue	Min Rate	Max Rate	Burst
P1/Q7	1Mbps	1.5Mbps	188Kbps
P2/Q7	1Mbps	1.5Mbps	188Kbps
P3/Q7	150Mbps	150Mbps	28Mbps
P3/Q7	150Mbps	150Mbps	28Mbps
P2/Q1	512Kbps	512Kbps	–
P2/Q2	512Kbps	512Kbps	–
P3/Q1	512Kbps	512Kbps	–

the physical switch ports that each node is connected to. In order to configure the minimum and maximum rates for each slice, we performed measurements for each application using Windows performance monitor and a legacy gigabit Ethernet (GbE) switch. Based on the collected performance results, we set up the queues rates as shown in Table II.

The Python script runs on the host OS and communicates with the ODL via the REST interface over the virtual network between the host and the VM. There are two configurations stored on the Python script, which are quite similar apart from the secondary cameras flow, which restricted to a lower maximum rate on the Openflow meter, which results in disrupting its feed. The change in configuration is triggered by a gesture of the glove in the master domain, and upon repeating the gesture on the glove, the original configuration is restored by the Python script and the camera feed is restored. This change in the SDN controller configuration reprograms the switches rules for a certain flow, allowing to show how the traffic is completely isolated, and while one flow might be completely disrupted the QoS of the other data feeds remains unchanged.

IV. Experimental Setup and Performance Discussion

For testing purposes, we setup three different physical hosts with one of them acting as a traffic sink (receiver) and two of them acting as traffic generators (transmitters). We configure *iperf3* to generate background traffic at 400 Mbps from one of the hosts while the second host generates traffic at 80 Mbps, which represents one of the camera feeds. The reason for analysing the camera traffic is that it requires a high bandwidth as well as the lowest possible latency to synchronise with the tactile traffic. In particular, to allow for a correct functioning of the system and a correct overall user experience both visual

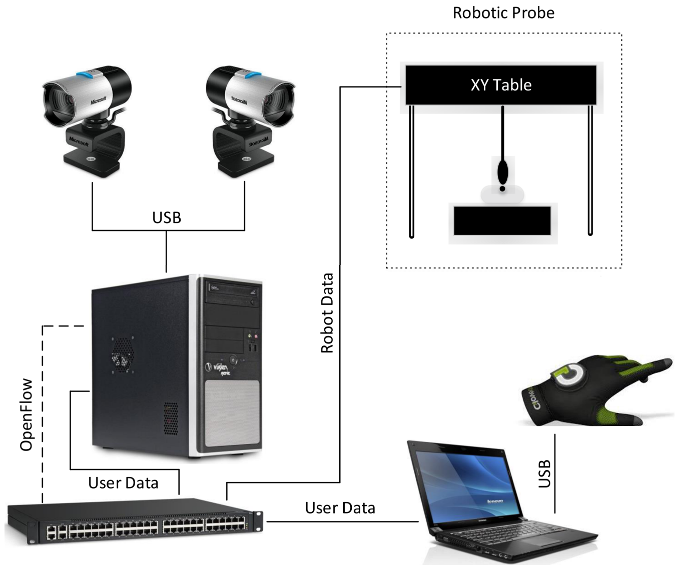


Fig. 5: Prototype demo setup

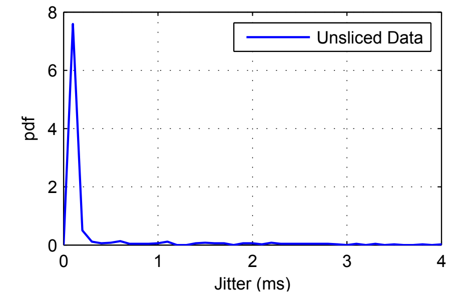
TABLE III: Testbed configuration

Parameter	Value
BE Traffic	400 Mbps
Video Traffic	80 Mbps
Traffic type	UDP
Switch capacity	1 Gbps

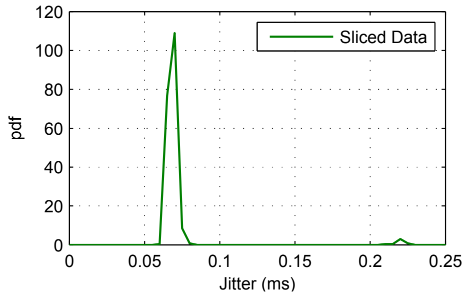
and tactile feedback should have similar end to end latencies. To verify the functionality of traffic isolation through network slicing and the proposed QoS management implementation, key performance indicators of latency, jitter and packet loss are measured for both un-sliced and sliced network set ups. Furthermore, we consider one switch in our testing scenario, however the results obtained can be easily extrapolated to a network with a higher number of switches.

Our experiment consists of 6 sets of measurements for each set-up lasting 90 seconds each and reporting link statistics of every second. This generates a dataset of 540 measurements for each set-up. Both traffic generators transmit UDP packets towards the sink host. Since the camera traffic is considered to be critical for the purpose of this use case, the traffic flow is placed on the higher priority queue in the sliced network scenario. The rest of the traffic generated is treated as best effort (BE). Table III summarises the test-bed configuration parameters for the performance evaluation.

Our observations from benchmarking show that slicing can provide a benefit even in scenarios where there is a single switch on the network. The one-way latency measured in the camera traffic was very much consistent: 8.5 ms for the unsliced scenario and 0.7 ms for the sliced scenario. This difference in latency is given because of the nature of the traffic treatment, in the unsliced configuration all traffic goes to the same pool and it is pushed out of the switch without any differentiation, having the same forwarding rules despite being BE or critical. When slicing is configured, the critical



(a) pdf of jitter in unsliced configuration



(b) pdf of jitter in sliced configuration

Fig. 6: Jitter comparison

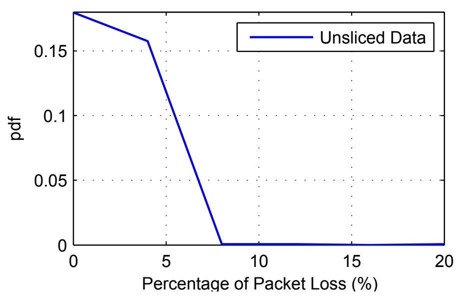
traffic is always mapped to the highest priority queue, which is served always first, as pictured in Figure 4.

The same rationale applies to the jitter. Figure 6 compares the probability density function (pdf) of the measured jitter before and after the slicing. From the results it is appreciated that both average and standard deviation of the experimental jitter is reduced. Notably, the average jitter is reduced in an 88%, and the standard deviation is reduced in a 98%.

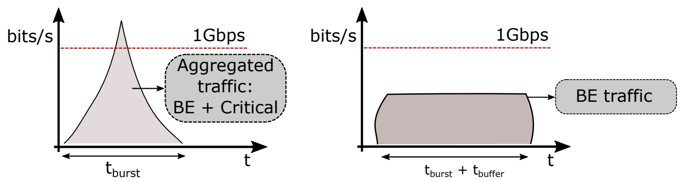
When comparing packet loss there is a similar effect when going from an unsliced to sliced configuration; Figure 7(a) shows the pdf of the packet loss in the unsliced configuration. In particular, when traffic is not differentiated and no QoS management is performed, the bursts of traffic that ingress the switch can be aggregated at some point in time resulting in traffic bursts higher than the switch capacity, as sketched in Figure 7(b). Since there is no traffic matching, the switch will discard packets to satisfy its maximum allowed data rate. When slicing traffic, the camera flow is always placed in the highest priority queue, and the BE traffic is forced to wait until higher priority queues are emptied to be served. As a result, there is no packet loss in the camera flow, but the BE traffic suffers an increase in the queueing latency, (t_{buffer}).

V. Conclusion and Future Work

In this work we have investigated and built a proof of concept of the physical and logical resource reservation aspect of a network slicing instance. We have used SDN features of queueing and meters, that together allow to perform traffic differentiation and isolation, as well as accurate QoS management. This allows to satisfy the different requirements for each different flow type.



(a) pdf of packet loss in unsliced configuration



(b) Traffic burst and packet loss

Fig. 7: Packet loss in experimental setup

We have built this in the context of multi-modal critical communications, including vision and touch. In the context of real-time tele-operating systems good quality of experience is mandatory, and the SDN features explored have shown to enable Internet of Skills systems.

Acknowledgement

This work is partly sponsored by Ericsson 5G Tactile Internet industry grant to King's College London and by the 5GPP VirtuWind Project (Virtual and programmable industrial network prototype deployed in operational wind park).

References

[1] Mobile and wireless communications Enablers for the Twenty-twenty Information Society (METIS), "Updated scenarios, requirements and KPIs for 5G mobile and wireless system with recommendations for future investigations - Deliverable 1.5," May 2015.

[2] A. Aijaz, M. Dohler, A. H. Aghvami, V. Friderikos, and M. Frodigh, "Realizing the tactile internet: Haptic communications over next generation 5g cellular networks," *CoRR*, vol. abs/1510.02826, 2015. [Online]. Available: http://arxiv.org/abs/1510.02826

[3] N. Sornkarn and T. Nanayakkara, "Can a soft robotic probe use stiffness control like a human finger to improve efficacy of haptic perception?" *IEEE Transactions on Haptics*, vol. PP, no. 99, pp. 1–1, 2016.

[4] "Digital Health in the UK An industry study for the Office of Life Sciences," White Paper, Office for Life Sciences, September 2015.

[5] "Touching lives through mobile health Assessment of the global market opportunity," White Paper, GSMA, February 2012.

[6] D. Kreutz, F. M. V. Ramos, P. E. Verssimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14–76, Jan 2015.

[7] B. Challacombe, L. Kavoussi, A. Patriciu, D. Stoianovici, and P. Dasgupta, "Technology insight: telementoring and telesurgery in urology," *Nature Clinical Practice Urology*, vol. 3, no. 11, pp. 611–617, 2006.

[8] A. El Saddik, M. Orozco, M. Eid, and J. Cha, *Haptics technologies: bringing touch to multimedia*. Springer Science & Business Media, 2011.

[9] NeuroDigital Technologies, "Gloveone," http://gloveonevr.com/.

[10] Ultragrid, "Wiki," https://www.sitola.cz/igrid/index.php/UltraGrid.

[11] N. Alliance, "Description of Network Slicing Concept," NGMN Alliance, Tech. Rep., 01 2016.