Air-to-Ground Real-Time Multimedia Delivery: a Multipath Testbed

Manlio Bacco[0000−0001−6733−1873], Pietro Cassarà[0000−0002−3704−4133], and Alberto Gotta[0000−0002−8134−7844]

Institute of Information Science and Technologies (ISTI), National Research Council (CNR), Pisa, Italy; emails: `[name.surname]@isti.cnr.it`

Abstract. In this work, we focus our attention on real-time multimedia flows from Unmanned Aerial Vehicles (UAVs) to the ground, presenting and analysing the data collected in field trials during a real testbed. The objective is assessing whether a video feed of reasonable quality can be provided to the pilot of an UAV to enable Beyond Visual Line of Sight (BVLoS) operations, by exploiting the multiple cellular operators available in the area. Three cellular networks have been jointly used in a multihoming/multipath setup, leveraging the variable coverage offered in both urban and suburban environments. Taking into account both Quality of Service (QoS) and Quality of Experience (QoE) metrics, the target parameters measured in this testbed are: latency, packet error rate, and video quality, which accounts for frames integrity, continuity, and fluidity. Data collected on the field allow to evaluate both QoS and QoE in the presence of a multipath architecture, showing how the latter, in the presence of network diversity, offer the possibility to improve the QoE at the receiver. We also design a framework to characterize the error model and to map it into a QoE model, therefore providing an analytical characterisation of a multipath channel.

Keywords: multipath · video streaming · real-time · QoS · QoE · testbed

1 Introduction

The use of UAVs is increasingly common in a wide spectrum of applications and services [1]. Being connected flying objects that can carry things, connect to networks or provide connectivity, monitor areas, people, and buildings [2], UAVs can prove to be very versatile, fast-moving, and available in a large range of sizes. The use of UAVs as moving Radio Access Network (RAN) [3], especially in the case of 5G networks, is gaining popularity in the literature, and real testbeds have been already carried out demonstrating the feasibility of such an approach. UAVs are also commonly used for monitoring activities [4], among others, through video streaming given that high-resolution cameras, fitting also on small UAVs, are largely available on the market. In this line, use cases of interest, especially considering BVLoS flights, are those related to e.g., inspection of power lines [5] - with flights in the range of tens of kilometers - to spot points in which intervention may be needed; inspection of infrastructure, such as railways [6]; or in smart cities for the purposes of traffic monitoring and management, health services, tourism, and goods delivery [7]. Air-to-ground video feedback is a common solution in the absence of Visual Line of Sight (VLoS) [8].

This work considers scenarios, as those just mentioned, in which real-time video streaming from UAVs to ground stations is needed, i.e., the case of latency-sensitive applications and services. According to 5G classes of services, such scenarios would fall into Ultra-Reliable and Low Latency Communications (URLLC) and enhanced Mobile BroadBand (eMBB) [9]. Further than latency sensitivity, we consider as key requirement the QoE at the Ground Control Station (GCS), aiming at supporting the pilot on the ground in the case of BVLoS flights. In fact, it is most important that the pilot has visual feedback coming from the UAV for safety and security reasons, and such feedback must be both reliable and real-time.

We focus on urban and suburban areas, extending our previous contribution in [10], targeting the provision of real-time *situational awareness* [11], requiring minimal loss rate and low delay.

The results herein presented are based on a real testbed carried out in the city of Pisa, Italy, and its suburban area. We exploit multiple Network Interface Cards (NICs) at the sender - a multihoming setup - to maximise the probability of always having at least one active link delivering the video flow to the GCS. In fact, poorly served areas (as sometimes the case of suburban ones) may cause the links to temporarily drop, as well as heavy traffic in others (as sometimes in cities) may be the cause of an unacceptable large delay. Both situations are to be avoided in a subcritical scenario as the one we consider, thus motivating us to rely on multihoming to increase the probability of having at least one active link, with reduced loss rate and contained delay. The use of different cellular networks for air-to-ground multimedia delivery in our testbed means that redundant alternative paths are in place between sender and receiver, thus increasing the probability of achieving situational awareness. We use three sender-side NICs for data transmission over three different public cellular networks, delivering traffic to a single NIC on the ground, i.e., at the GCS. As highlighted in [12] and actually experienced in our scenario, the various paths are not necessarily disjoint, and this can depend on several reasons. In our testbed, it is likely that some sections of the fixed infrastructure on the ground is shared among different operators in certain areas, meaning that the network QoS of the different links may show non-negligible correlations. Eventually, this may impact on the achievable QoE, as we show in this paper when mapping QoS into QoE.

Based on those premises, this work provides both a real implementation and insights in: (*i*) the opportunistic use of the access networks of multiple cellular Internet Service Providers (ISPs) to deliver a video stream from an UAV towards a GCS in a real testbed; (*ii*) the use of multipath transport to improve the QoE at the GCS; (*iii*) an analytical framework to characterise multipath communications; and, finally, (*iv*) the mapping of QoS statistics into QoE evaluations. The rest of this paper is organized as follows: Section 2 surveys the state of the art, focusing on scenarios similar to the one under consideration herein, and on the use of multipath techniques to deliver multimedia live streams. Section 3 provides details on the system configuration used to carry out the testbed and the related system parameters. Section 4 introduces to the analytical framework proposed in this work, used to characterise the multipath channel and to support the mapping of QoS into QoE, which is presented in Section 5. Finally, Section 6 draws the conclusions and opens to future works.

2 Related Works

In this section, we survey how key requirements of our scenario are approached in the literature, what kind of solutions have been identified so far, and how and why our approach is different from the other ones we consider.

2.1 Use cases and requirements for air-to-ground video feeds

A valuable recollection of use cases leveraging UAVs can be read in [13] - focusing on Internet of Things (IoT) scenarios - covering for instance disaster management, traffic monitoring, crowd surveillance, or environmental monitoring [5] and agricultural applications [14]. In those scenario, the use of cellular connections is typically foreseen, calling for an analysis of the quality of the signal from above. In several studies, such as in [15, 16], the quality of the signal has been evaluated at flying altitudes, showing that 4G connectivity can be effectively used to provide wide-area wireless connectivity to UAVs [15], even if limitations should be taken into account, like the rapid decrease of the received signal as the altitude increases [16]. Thus, the possibility of using UAVs in conjunction with terrestrial networks has been investigated in the literature with positive results. What needs careful evaluation is the actual possibility of achieving the so-called situational awareness in BVLoS conditions via public cellular networks, which calls for careful attention to maximise the probability of *video continuity*, which is one of our key requirements. Further than continuity, the *playout delay*, i.e., the time delay after which a video chunk is played with respect to its generation instant at the source side, must be strictly limited to actually provide real-time visual context. Thus, a minimum value of playout delay adds to the list of considered requirements in this work.

2.2 Multipath protocol solutions and real-time video streaming

Real-time video streaming poses additional requirements to those above. In the case of multipath protocols - on which a valuable survey is available in [17] - one of the first solutions available in the literature and actually implemented is MultiPath TCP (MP-TCP) [18]. Its main advantage is bandwidth aggregation, contrarily to what the Stream Control Transmission Protocol (SCTP) does. SCTP is a multihoming protocol using a single link at a time, whereas the other link are backup options for reliability purposes. In the case of multimedia data delivery, MultiPath RTP (MP-RTP) [19] is the multipath version of RTP, the protocol designed for end-to-end, real-time transfer of streaming media. The available MP-RTP implementations provide additional features, such as the use of Forward Error Correction (FEC) techniques for congestion control [20] so to shift traffic from congested to less congested paths[1]. MultiPath QUIC (MP-QUIC) [22] can be cited as well as emerging multipath solution at the transport layer, offering encrypted, stream-multiplexed, and low-latency data exchanges. An emerging solution is the use of the DASH (Dynamic Adaptive Streaming over HTTP) protocol, which is TCP-based, also in the case of multipath solutions [23].

The core feature of multipath solutions is that disjoint paths between a sender and a receiver can be used as a single logical one to deliver data flows, leveraging network diversity to improve the link availability, increase the available bandwidth, or rely on several low-cost links to mimic the statistics provided by a single high-performance link. Those features can be alternative to each other or even partially achieved together; in our scenario, the aim is in improving the link availability to satisfy the requirement on video continuity. Typically, a key goal is the increase of available bandwidth, which anyway should be pursued by carefully choosing the links to be used to respect any latency constraints in the case of latency-sensitive applications. In fact, heterogeneous networks typically exhibit different network statistics, and multipath solutions may offer worse performance when compared with the plain protocol versions [19].

The use of MP-TCP has been tested for video streaming, as in [24, 25], i.e., using elastic protocols typically used in different scenarios, as for instance IoT ones [26]. Typically, real-time multimedia streaming does not occur over TCP because the constraint posed by live feeds makes unnecessary, if not even detrimental, the use of retransmissions in the presence of losses. In the case of elastic protocols, such as TCP, homogeneous paths (i.e., links showing comparable network statistics) represent a condition for satisfactory performance in multimedia streaming. In [12], the authors show how, on the one hand, constant bandwidth on multiple paths results in improved video quality with respect to the case of a single path; on the other hand, how bandwidth fluctuations harm user experiences. It must be noted that MP-TCP suffers from network middleboxes and proxies, thus its use may be limited or broken when traversing them. We excluded MP-TCP as a solution because of unneeded retransmissions, and to avoid potential issues with appliances on the ISPs networks that may impact on the desired QoE level. Real-time and high-quality video streaming is bandwidth-intensive and delay-sensitive [27], and has stringent QoS requirements. In fact, to really achieve real-time video streaming, a one-way delay of maximum 150ms

[1] A valuable survey on the topic of congestion for multipath protocols, including those for video streaming, can be read in [21].

should be taken into account[2], as we do in this work. Other key requirements on QoS are related to jitter and packet loss rate, which must be as contained as possible for high-quality multimedia feeds. The use of FEC-based techniques is a common approach to counteract loss phenomena. Nonetheless, FEC-based multipath protocols in the literature are throughput-oriented and video data is scheduled in a content-agnostic fashion [27], thus not making them the optimal choice in the case of multimedia streaming. For instance, the work in [28] proposes the use of a XOR-based dynamic FEC solution for MP-TCP to lower the probability of retransmissions, thus reducing delays due to lost - and then retransmitted - packets. Such a solution may be of some interest in the case of latency-sensitive applications, but the use of FEC impacts on the TCP congestion window - which defines how many bytes of data are sent per period - because fresh data may be sacrificed in favor of redundancy. Furthermore, a single loss is tolerated per FEC block in [28], a solution that, on the one hand, may not be enough in the case of burst losses; but, on the other hand, provides low computational overhead and a rather simple implementation. Burst losses is a phenomenon that we experienced in our real tests, as shown below. It is also worth highlighting that, the higher the FEC redundancy, the higher the energy consumption [29]. Because of the occurrence of burst losses and of the need of a simple but effective implementation to be run on constrained devices, we do not rely on classical FEC approaches in this work. From the viewpoint of the implementations, those in user-space - rather than at different layers of the stack - are increasingly common because of the flexibility they can provide in different scenarios [28], as we argued in [30] in the case of generic IoT traffic. We will consider user-space solutions in future works for the flexibility they provide.

On a closing note, we briefly acknowledge that novel video coding schemes have been proposed for high-quality video from UAVs to ground users because existing ones do not yet meet the expected QoE according to [31]. In this work, we exploit well-established solutions for video coding, such as the H.264 standard, relying instead on the use of a multipath setup to meet the desired QoE. Multipath solutions provide advantages in the presence of network diversity, as we show in the results. Our scenario is in high mobility conditions in both urban and suburban areas. To strengthen the connection reliability, lightweight FEC solutions are herein preferred to more resource-consuming ones, such as network coding as proposed in similar cases [32].

2.3 The impact of QoS on QoE

Finally, we briefly cover the state of the art on how to map the network statistics (i.e., QoS) into QoE, thus opening to sender-side scheduling strategies that target a predefined QoS that can provide a QoE level above a predefined threshold. QoE takes into consideration the end-user subjectivity, which

[2] ITU-T G.114 recommends a less than 150 millisecond one-way delay as excellent for media quality, although delays between 150 and 400 milliseconds are considered as still acceptable.

depends on QoS and other factors; consequently, subjective and objective quality assessment methods are needed to model the impact of both technical and non-technical factors, as analysed in [33]. Several works [34–36] faced with the definition of mapping QoS onto QoE. In [35, 36], the authors discuss learning approaches for both online and offline mapping. All the proposed mappings build on quality comparisons between the undistorted (source side) video and the potential distorted (destination side) video, namely *reference* and *outcome*, respectively. The quality of the outcome can be rated in terms of Mean Opinion Score (MOS)[3] exploiting the reference. Whether the reference is available or not defines the following types of metrics: Full Reference (FR), No Reference (NR), and Reduced Reference (RR). In the case of FR, both subjective and objective comparisons of the outcome with the reference can be carried out because both are available at destination. Hence, very accurate metrics can be derived.In the case of NR, a quality score must be derived from the outcome only, which of course provides poorer information if compared with the FR case. In a typical scenario, such as the one under analysis in this work, NR-based metrics lack the possibility of discerning between pure quality-related issues from any disturbances due to the network [35, 36]. But the obtained metrics can be estimated through low-complexity algorithms, thus being suitable for online use in resource-constrained and/or real-time settings. Man-in-the-loop's feedback can be collected at the sender in addition to network statistics in order to further tune up NR metrics. RR must be considered as the case in between FR and NR because of e.g., the availability of a QoE model or of data collected in similar scenarios that can be used as reference. In other words, the core difference with respect to NR is the possibility to exploit additional information at the destination to derive a more meaningful QoE metric. We make use of a QoE model in this work, thus positioning our work in the RR case.

3 System Configuration

In this section, we describe the system configuration used in our real testbed, briefly discussing the reference protocol stack in Section 3.1, the hardware setup in Section 3.2, and the software setup in Section 3.4.

3.1 Reference Protocol Stack

Multihoming consists in the capability of a device of leveraging a set of routes provided by two or more ISPs, each one with a distinct IPv4/v6 address for both inbound and outbound traffic. RFC 4116 details the IPv4 practices and goals of a multihoming architecture that are:

– *redundancy*, which can protect a system from some single-point of failure (SPOF). The degree of protection relies on the policies applied to interconnect the system to the providers and how route the information on multiple network interfaces;

[3] Recommendation ITU-T J.247, "Measurement of the quality of service. Objective perceptual multimedia video quality measurement in the presence of a full reference", 08/2008.

– *load sharing*, which account for how outbound traffic is shared across multiple ISPs;
– *policies*, which accounts either for the capability of relaying certain types of traffic to a given set of ISPs according to some budget rules or for path scheduling according to certain QoS metrics;
– *simplicity and scalability* since multihoming solutions may require complex algorithms that must not jeopardize but instead cope with the scalability of a system.

UAVs-based solutions leveraging multihoming capabilities may also benefit of multipath transport protocols, which can enable load sharing or concurrent multipath transfer in multihomed systems. For end-to-end multimedia session, multipath transport can provide several advantages over a single-path transmission, since it can provide higher transmission rate, redundancy between multiple paths, and higher reliability.

However, multipath protocols require that the endpoints must implement and support multipath transport. Establishing a multipath transport scenario based on application-level relay (AR) is one of the multipath routing methods, proposed in [37] as a general framework for multipath transport systems (MPTS-AR). Figure 1 shows the protocol stack, which shifts the multipath management to a shim-layer, which accounts for implementing the relay and redundancy policies other than the multiple sessions establishment. MPTS-AR has several advantages: *(i)* does not require any specific application/transport protocol to work, *(ii)* does not require any modifications to the protocol stack to support multipath capabilities, and *(iii)* opens to the use of different protocol flavors on different paths. In this work, such a framework can be considered as a reference one, which we implemented as later described.

3.2 Hardware Setup

In our setup, three onboard Long-Term Evolution (LTE) routers have been used for multihoming, as shown in Figure 2b, which also zooms the Raspberry Pi (RPi) used to collect and transmit the video feed.

The RPi 3 Model B+ is equipped with a Raspberry Pi Camera Module V2.1 for video streaming, and two USB WiFi dongles 2.4/5GHz, IEEE 802.11b/g/n in addition to the integrated Broadcom BCM43438 wireless interface. Each WiFi NIC is connected to a different LTE home-grade router Huawei E5573 4G-LTE CAT4 with nominal download and upload rates of 150 Mbps and 50 Mbps, respectively, so that three cellular connections may be used simultaneously during the flight. The Huawei E5573 router can connect to clients via cable, Ethernet-over-USB, or wireless. Each router is powered by a LiPo battery or through an USB port. Also the RPi is powered through a powerbank visible below it in Figure 2b, which allows the RPi to run for more than an hour, with a negligible additional weight. In the presented setup, we preferred a wireless connection of the RPi with the Huawei E5573 routers to reduce the impact on the powerbank, and also to avoid any power surge on the USB bus of the RPi. The hardware

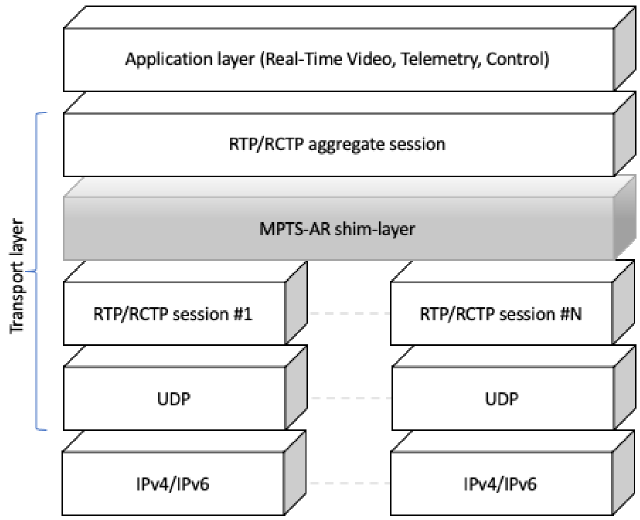


Fig. 1: RTP/UDP protocol stack used for multipath transport systems with application relay.

(a) (b)

Fig. 2: The testbed platform in use: the UAV on the left, the RPi on top of a large capacity battery on the right, with the mounted Pi Camera Module, and three home-grade LTE routers.

visible in Figure 2b has been used as payload of the UAV into a red housing visible in Figure 2a. The UAV is a custom octo-copter, already used in several activities [2, 5, 14], weighting approximately 5 [Kg], equipped with brushless engines, and able to fly at a maximum speed of 130 [Km/h] for about 15-20 minutes depending on the payload (up to 2.5 [Kg]). According to the definition proposed in [38], such a configuration is similar to the so-called *tight plane-based framework* used in smart cities.

At the receiver side, a laptop acts as GCS for both telemetry and live video stream thanks to the open-source software *QGroundControl*. The receiver has one public IPv4 address. For the sake of completeness, the laptop is based on an Intel Core-i7 processor, 8GB RAM, running Ubuntu Linux. The RPi runs a Raspbian Wheezy distribution. Furthermore, the three LTE routers are equipped with SIM cards of three different Italian providers: Vodafone, Tim, and WindTre.

3.3 Network setup

Figure 3 shows the reference transmitting and receiving setup, as well as the use of three cellular networks. Such a picture shows the three routing units detailed in 3.2, which have a private IP number on different private networks, exemplified with subnets 10.0.1.x, 10.0.2.x, and 10.0.3.x. Each ISP assigns a public IP to the WAN port of the relative router. On the RPi, each NIC is configured with a static IP address taken within the relative subnet of the relative router. An *iptable* configuration was set, by using *Mangle* and *NAT* tables to manipulate and forward the IP packets to the desired subnet according to the source address and toward the same public destination address, i.e., that of the GCS (on the right in Figure 3). Below we report the code to set up the IP forwarding rules for subnet 10.0.1.x:

```
#!/bin/sh
receiver_udp_port=5000
wlan_if_1=wlx74da38c822ff
wlan_addr_1=10.0.1.103
gw_1=10.0.1.1
dst_port_1=5004

sudo iptables -t nat -D OUTPUT -p udp --dport $dst_port_1 -j DNAT
   --to-destination :$receiver_udp_port
sudo iptables -t nat -D POSTROUTING -p udp -o $wlan_if_1 --dport
   $receiver_udp_port -j SNAT --to-source $wlan_addr_1:$dst_port_1

sudo iptables -A OUTPUT -t mangle -p udp
   --dport $dst_port_1 -j MARK --set-mark 4
sudo ip rule add fwmark 4 table SUBNET1
sudo ip route add default via $gw_1 dev $wlan_if_1 table SUBNET1
sudo iptables -t nat -A OUTPUT -p udp --dport $dst_port_1 -j
   DNAT --to-destination :$receiver_udp_port
```

```
sudo iptables -t nat -A POSTROUTING -p udp -o $wlan_if_1
   --dport $receiver_udp_port -j
   SNAT --to-source $wlan_addr_1:$dst_port_1
```

3.4 Real-time video setup

GStreamer is a reference library for video streaming applications: it is an open source multimedia platform, available for the most common operating systems and embedded platforms, like the RPi. The release installed on both desktop and RPi for our experiments is version 1.14.4.

Our development efforts have been concentrated at the sender side, i.e. the platform onboard the UAV. A Gstreamer-based application is composed of a pipeline of software modules, called *plugins*, which implement the needed functional blocks, like encoding and decoding, mux and demux, buffering, scaling, dejittering, and data transport. In more details, the video stream is captured through the camera, scaled to a resolution of 1024x768 pixels at 10fps, and then compressed with an hardware-accelerated H.264 encoder. No adaptive video coding is used at the source, in order to account only for the impact of channel coding and channel erasures, or out-of-sequence packets. The video stream is parsed and encapsulated into Real-time Transport Protocol (RTP) packets, then the application relay (AR) enables the multipath feature: RTP packets are replicated on the respective paths to avoid SPOF. Such an implementation fulfills the goals in RFC 4116, i.e., it provides a lightweight implementation, suitable for constrained devices, achieving very low delay.

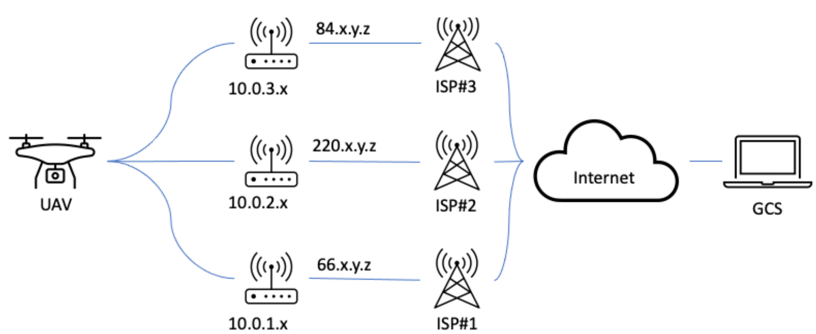


Fig. 3: Transmitting and receiving multipath/multihomed scheme for real-time multimedia flows from an UAV to a fixed GCS.

At the receiver side, a dejitter module - part of the Gstreamer pipeline - has been used to reorder and to remove duplicated packets, the latter likely to occur because of the replicas. The maximum allowed latency of the dejitter buffer is set to $\mathcal{L}$=0.2 [s] in our setup, i.e., it handles out-of-sequence packets delayed up to $\mathcal{L}$ [s]. Such a value is a reasonable tradeoff between piloting requirements and varying network delay conditions when moving at medium-high speed with an UAV. Table 1 summarises the values of the main parameters in use[4].

The target Group of Pictures (GoP) size of the H.264 encoder is set to the default GStreamer value of 90 frames. It translates into a high compression ratio, but introducing large dependency among contiguous frames. The consequence is that, in the case of a partially received video frame because of packet losses, also the subsequent frames pertaining to the GoP are affected, lowering the QoE.

Plugin	**Parameters**
videosrc	video/x-raw, width = 1024 [px], height = 768 [px]
h264enc	target GoP size = 90, target-bitrate = 1 [Mbps]
rtph264pay	packet-size = 1432 [B], payload type 96
dejitter	latency = 200 [ms]

Table 1: System parameters of the GStreamer pipeline.

4 Analytical Framework

This section introduces the analytical error model for a multipath channel (in Section 4.1) and the relative QoS metrics. The impact of the dejitter buffer is discussed in Section 4.2, the main network statistics in Section 4.3, and the error burst length in Section 4.4.

4.1 Multipath error model

We assume the channel model to be governed by a Discrete Time Markov Chain (DTMC) at packet level as in [40], whose process is characterized by the evolution of two states, *Good* (G), and *Bad* (B). We assume that no packet is lost being in G, while all packets are lost being in B. The DTMC model captures the bursty nature of lossy periods with respect to a Bernoulli model, which cannot model the burstiness of wireless and mobile channels. The channel transition of the i-path is described by the following transition matrix:

$$T_i = \begin{pmatrix} P_{GG,i} & P_{GB,i} \\ P_{BG,i} & P_{BB,i} \end{pmatrix},$$

where $P_{X,Y,i}$ is the transition probability from state X to state Y in a period of time equal to the transmission time T_p of a packet. From the DTMC theory, it occurs that:

$$P_{GG,i} = 1 - P_{GB,i}, P_{BB,i} = 1 - P_{BG,i}.$$

[4] The testbed is driving us into the development of a simulator to further investigate the impact of said parameters and to further optimise the sender-side scheduling [39].

The average packet loss rate can be expressed as:

$$P_{BAD,i} = \frac{P_{GB,i}}{P_{GB,i} + P_{BG,i}}, \tag{1}$$

and the average error burst length ebl_i as:

$$ebl_i = \frac{1}{P_{BG,i}} = \frac{1}{1 - P_{BB,i}}. \tag{2}$$

In the case of two independent paths characterized by G/B channel states, the aggregate behavior of the two paths can be described by means of a four-state DTMC, where (G_1, G_2) is the first state, (G_1, B_2) the second one, (B_1, G_2) the third one, and (B_1, B_2) the last one. In a multipath case with two paths, the transition matrix T_{2-mp} of the four-states DTMC can be expressed, under the hypothesis of independence of the i, j channels, as:

$$T_{2-mp} = \begin{pmatrix} P_{GG,1}P_{GG,2} & P_{GG,1}P_{GB,2} & P_{GB,1}P_{GG,2} & P_{GB,1}P_{GB,2} \\ P_{GG,1}P_{BG,2} & P_{GG,1}P_{BB,2} & P_{GB,1}P_{BG,2} & P_{GB,1}P_{BB,2} \\ P_{BG,1}P_{GG,2} & P_{BG,1}P_{GB,2} & P_{BB,1}P_{GG,2} & P_{BB,1}P_{GB,2} \\ P_{BG,1}P_{BG,2} & P_{BG,1}P_{BB,2} & P_{BB,1}P_{BG,2} & P_{BB,1}P_{BB,2} \end{pmatrix}. \tag{3}$$

Then, the stationary state probability distribution is:

$$\pi = \begin{pmatrix} (1 - P_{BAD,1})(1 - P_{BAD,2}) \\ (1 - P_{BAD,1})P_{BAD,2} \\ P_{BAD,1}(1 - P_{BAD,2}) \\ P_{BAD,1}P_{BAD,2} \end{pmatrix}. \tag{4}$$

Thus, a packet duplicated on both paths is lost if both the channels are in the respective B states, that is:

$$P_{BAD,2-mp} = P_{BAD,1}P_{BAD,2}. \tag{5}$$

Analogously to Eq. (2), the resulting error burst length ebl_{2-mp} of the four-state DTMC can be expressed as:

$$ebl_{2-mp} = \frac{1}{1 - P_{BB,1}P_{BB,2}}. \tag{6}$$

In the case of s paths, the DTMC is composed of 2^s states and the resulting $P_{BAD,s-mp}$, ebl_{s-mp} are given by:

$$P_{BAD,s-mp} = \prod_{i=1}^{s} P_{BAD,i} \tag{7}$$

$$ebl_{s-mp} = \frac{1}{1 - \prod_{i=1}^{s} P_{BB,i}} \tag{8}$$

It is worth nothing that, even relaxing the hypothesis of independence among the s-channels in Eq. (7), the product of $P_{BAD,i}$ is replaced by the joint stationary probability of having all the s channels in B state. Analogously for the ebl_{s-mp}, the product of the $P_{BB,i}$ is replaced by the joint transition probability that all the s channels remain in B state. Under the hypothesis of independence, it is easy to infer that $P_{BAD,s-mp} \xrightarrow{s\to\infty} 0$ and $ebl_{s-mp} \xrightarrow{s\to\infty} 1$. Contrarily, a strong correlation between two paths or more implies that one of the two channels is not providing any significant advantages in terms of network diversity.

4.2 Analysis of the Dejitter Buffer

The dejitter buffer is typically used in the presence of multimedia flows to reduce the impact of jitter, so feeding the decoder in evenly spaced intervals despite the irregularities due to the network. Its contribution is twofold in our case: on the one hand, it allows for reordering out-of-sequence packets and for discarding duplicated packets; on the other hand, the dejitter buffer drops packets sitting in the queue longer than $\mathcal{L}$ [s].

Because of such a behaviour, the dejitter buffer may contribute to the packet loss rate as seen by the H.264 decoder with:

$$P_{drop} = P(\Delta D > \mathcal{L}), \tag{9}$$

where ΔD is the delay between the arrival time of the out-of-order packet X and the instant in which the reordering is successful because the packets before X have all been correctly received. If it takes less than $\mathcal{L}$ to receive the missing packets (successful reordering), then the out-of-order packet is correctly forwarded; otherwise (unsuccessful reordering) it is dropped. Therefore, dropped out-of-sequence packets contribute to P_{drop}.

The resulting loss rate seen by a H.264 decoder is given by Eqs. (7) and (9) as[5]:

$$P_{loss} = P_{BAD,s-mp} + (1 - P_{BAD,s-mp})P_{drop}, \tag{10}$$

considering that only packets correctly received (with rate $1 - P_{BAD,s-mp}$) can be discarded because of $\Delta D > \mathcal{L}$.

4.3 Network statistics in experimental trials

This section describes how the measurement campaign has been conducted and provides the statistical analysis of the collected QoS parameters. The experimental testbed involves an urban and a suburban route, which are shown separately in Figure 4 to highlight the higher density of the Evolved Nodes B (eNBs) in the urban part (see Figure 4b) than in the suburban part (see Figure 4a). Furthermore, the suburban part considers uphill location, thus providing cellular connectivity in Line of Sight (LoS) conditions with several distant eNBs. In our

[5] As in the case of Eq. (7), also Eq. (10) is applicable if the delay process on each of the s paths is assumed as independent from the other ones.

(a) suburban path (Calci, Pisa) (b) urban path (Pisa)

Fig. 4: 3D maps of the real testbed. The path is marked in yellow, the eNBs in blue.

testbed, the main constraint while moving was the availability of cellular connectivity. The traffic at both the transmitter and the receiver has been dumped thanks to the use of Tshark (an open-source network protocol analyzer) to support the following analysis. In the following, we present the one-way delay and the latency accumulated by packets sitting in the dejitter buffer.

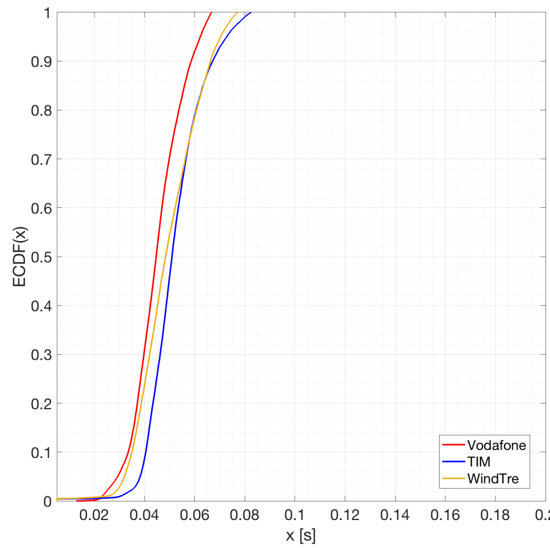
Figure 5 shows the one-way delay of each network.

It is shown for both suburban and urban scenarios in Figures 5a and 5b, respectively. It is worth noting that both Tim and Vodafone operators share a similar average value (≈30-40ms) most of the time in the urban scenario, while WindTre shows a different behaviour. All the operators behaves in a very similar way in the suburban scenario.

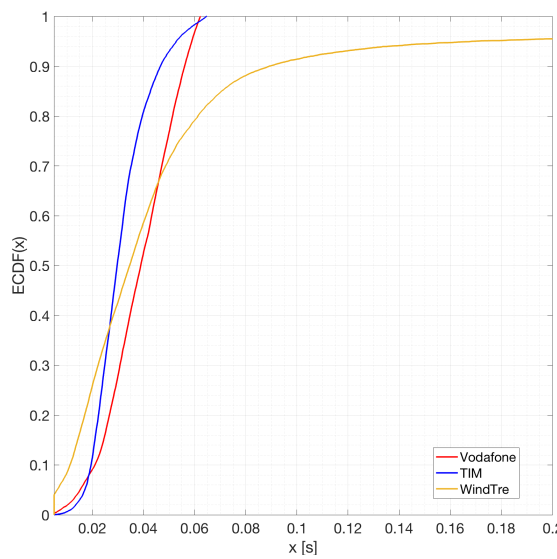
Figure 6 shows the survivor function of the delay accumulated by out-of-sequence packets: the dejitter buffer forwards the first-come copy of each packet (using the Sequence Number (SN)). However, different paths are likely to introduce different latencies, as shown in Figure 5. Taking into account the maximum tolerated latency $\mathcal{L}$, there is a non-null probability that the accumulated latency may exceed such a value, as shown in Figure 6.

Table 2 shows the traffic share among the three operators, i.e., which fraction of the data used by the H.264 decoder comes from Vodafone, Tim, or WindTre, respectively. The testbed in the urban scenario confirms that the packets delivered by WindTre had experienced a lower one-way delay than the other two operators. However, the sharing among the operators remains relatively fair because the distributions of the one-way delays are comparable. Regarding the suburban scenario, the distributions of the one-way delays are almost identical and this is reflected by the almost perfect sharing ratio among the operators.

The cumulative distribution functions (CDFs) of the receiving packet rate for the suburban and urban scenarios are shown in Figures (7a) and (7b), respectively. As discussed throughout this section, the three operators in the suburban scenario exhibit the same performance level, while in the urban one they do not. More precisely, CDFs of the suburban scenario show that the packet rates never go below $15-20$ $[pckts/sec]$. Instead, in the urban scenario, this is verified



(a) suburban scenario



(b) urban scenario

Fig. 5: Empirical Cumulative Distribution Function (ECDF) of the one-way delay.

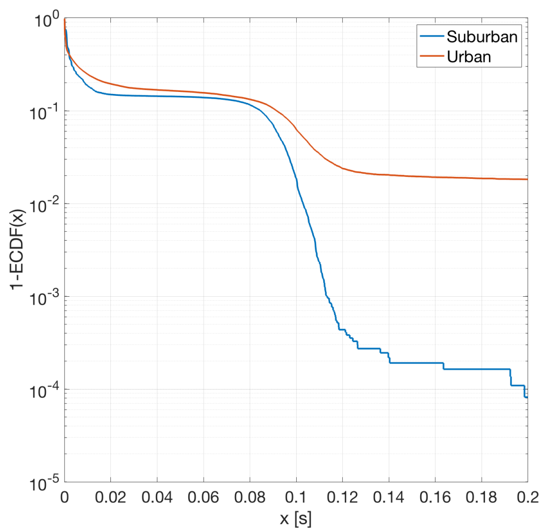
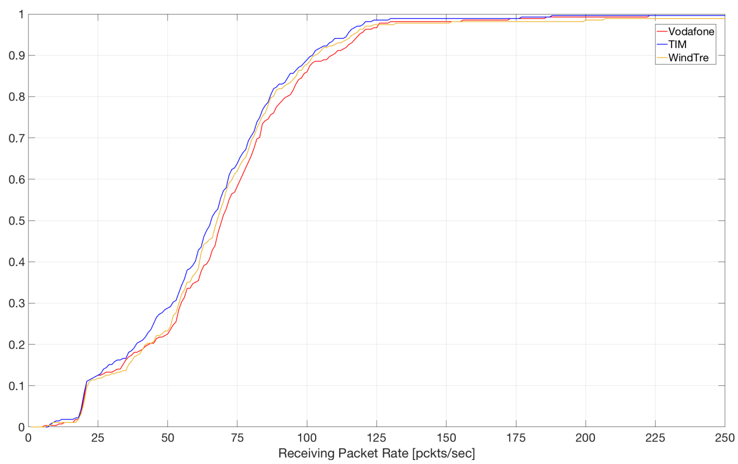


Fig. 6: Empirical survivor function of the latency accumulated by out-of-sequence packets in the dejitter buffer.

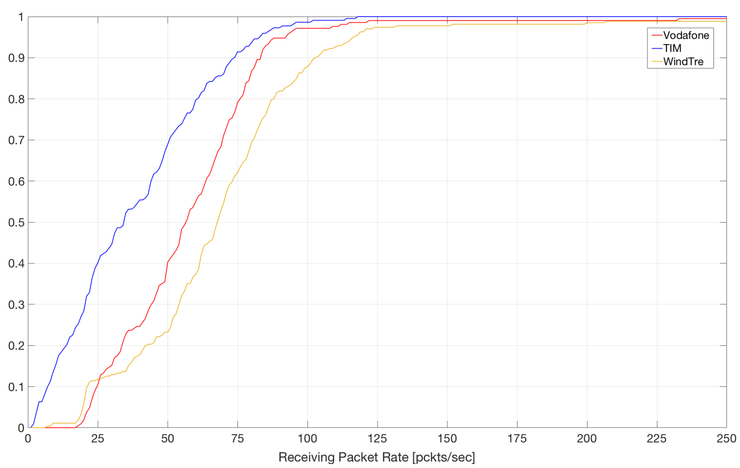
scenario	**operator**	**traffic sharing**
U	Tim	0.3039
U	Vodafone	0.3309
U	WindTre	0.3651
SU	Tim	0.3203
SU	Vodafone	0.3398
SU	WindTre	0.3398

Table 2: Fraction of received traffic via each cellular network (after the dejitter buffer).

only in the case of Vodafone and WindTre, but not of TIM. Finally, the average packet rate of the suburban scenario is similar for the three operators and falls within the range $60 - 70$ $[pckts/sec]$, while in the urban scenario, WindTre and Vodafone have an average rate in the range $60 - 65$ $[pckts/sec]$. Instead, TIM has a lower value, i.e., in the order of $40 - 45$ $[pckts/sec]$.



(a) suburban scenario



(b) urban scenario

Fig. 7: Cumulative Density Function (CDF) of the received bandwidth in packets (RTP/UDP) per second for both suburban and urban scenarios.

The error loss processes experienced on each path in both the scenarios are in Table 3: they report the underlying DTMC processes as estimated from the dataset. In addition, Table 3 reports the estimated DTMC process as seen after the dejitter buffer, i.e. after multiplexing, reordering, and filtering of the three flows.

scenario	**operator**	T_i	$\mathbf{\Pi_i}$
U	Vodafone	$\begin{pmatrix}0.9965 & 0.0035\\0.0294 & 0.9706\end{pmatrix}$	$\begin{pmatrix}0.8944\\0.1056\end{pmatrix}$
U	WindTre	$\begin{pmatrix}0.9997 & 0.0003\\0.0185 & 0.9815\end{pmatrix}$	$\begin{pmatrix}0.9879\\0.0121\end{pmatrix}$
U	Tim	$\begin{pmatrix}0.9783 & 0.0217\\0.0999 & 0.9001\end{pmatrix}$	$\begin{pmatrix}0.8216\\0.1784\end{pmatrix}$
U	Aggregated	$\begin{pmatrix}0.9818 & 0.0182\\0.9816 & 0.0184\end{pmatrix}$	$\begin{pmatrix}0.9818\\0.0182\end{pmatrix}$
SU	Vodafone	$\begin{pmatrix}0.9997 & 0.0003\\0.5455 & 0.4545\end{pmatrix}$	$\begin{pmatrix}0.9994\\0.0006\end{pmatrix}$
SU	WindTre	$\begin{pmatrix}0.9997 & 0.0003\\0.5455 & 0.4545\end{pmatrix}$	$\begin{pmatrix}0.9994\\0.0006\end{pmatrix}$
SU	Tim	$\begin{pmatrix}0.9881 & 0.0119\\0.1932 & 0.8068\end{pmatrix}$	$\begin{pmatrix}0.9421\\0.0579\end{pmatrix}$
SU	Aggregated	$\begin{pmatrix}0.99988 & 0.00012\\0.99974 & 0.00026\end{pmatrix}$	$\begin{pmatrix}0.99992\\0.00008\end{pmatrix}$

Table 3: Transition matrices and stationary probabilities, after the dejitter buffer, of the empirical DTMC model of the channels of the three operators in the urban (U) and suburban (SU) scenarios, as well as the resulting *aggregated* channel.

4.4 Error burst length

The error burst length ebl can be calculated using Eq. (8), which provides the values of 7 [pkts] and of 1.2 [pkts] before the dejitter buffer, and 1.018 [pkts] and 1.000 [pkts] after the dejitter buffer for the urban and suburban case, respectively. The differences in the error burst lengths capture the intrinsic decorrelating nature of a multipath system that not only works in difference of space, but also of time. In fact, different delays are experienced by the replicas in different paths. This effect significantly contributes in reducing the error correlation on consecutive RTP packets within the dejitter buffer. According to Eq. 2 in [41], the resulting one-step-correlation ρ between two consecutive packets in a DTMC is given by:

$$\rho = P_{BB} + P_{GG} - 1, \tag{11}$$

Therefore, the resulting ρ_i for both urban (U) and suburban (SU) cases are respectively calculated as: $\rho_U = 0.832$ and $\rho_{SU} = 0.152$. By solving the equation system composed of Eqs. (7), (8), and (11), the error burst length ebl_i can be expressed in terms of $P_{BAD,i}$ and ρ_i as:

$$ebl_i = \frac{1}{(1-\rho_i)(1+P_{BAD,i})}, \tag{12}$$

which fully characterizes the error process paired with Eq. (10) [42]. In fact, limiting ρ, i.e., reducing the burstiness of error sequences on packets, translates into the need for lower redundancy to protect the information [41].

5 Mapping Quality of Service into Quality of Experience

In the use case under consideration, quantifying the feeling of a remote operator about the received video feedback, in terms of QoE, becomes crucial towards the mapping of the latter with respect to QoS. A reference metric used to measure the feeling of a user about a video is based on MOS evaluations, performed using the statistical inference on the opinion scores, usually within a five-point interval, such as {*bad, poor, fair, good, excellent*}. QoE can be affected by video artifacts, missing frames, poor fluidity, and so on, which mainly depend from QoS parameters such as packet loss, delay, jitter, and maximum tolerable latency.

5.1 Analytical model

In [43, 44], the authors discuss the results of a metric, suitable for mobile networks, which maps QoS into QoE as follows:

$$QoE = k_1 - \frac{k_2}{1+(\frac{k_3}{Q})^{\eta}}, \tag{13}$$

with k_1, k_2 defining the maximum and the minimum value of QoE, with Q and $\{k_3, \eta\}$ instead depending on both network QoS and used video encoder. The model in Eq. (13) is also adopted in [45] to analyse the mapping between QoS and QoE in a video streaming scenario, where QoS is assumed to be a function of the loss rate; we adhere to the same assumption in what follows. The parameter Q in Eq. (13), which is the Non-Decodable Frame Rate, is defined as the complementary of that in [46], which is the ratio of the number of non-decodable frames to the total number of frames sent by a video source. The works in [45,46] analyse scenarios in which a sequence of interdependent MPEG-based encoded frames are transmitted (as in our case), assuming that the propagation of the spatial error due to packet loss impacts on the frames that are dependent on a given previous frame. The MPEG streams are sequences of GoPs, which in turn are sequences of I, P, and B frame types. The loss of even a single packet may cause a video frame to be undecodable, according to [45, 46]. In turns, it means that I frames in a GoP are successfully decoded only if all packets are correctly received. A P frame is decodable only if the preceding I or P frames are successfully decoded and the packets delivering the P frame are successfully received. Finally, the B frames in a GoP are decodable only if the preceding and succeeding I or P frames are both successfully decoded and all the B packets are successfully received. Hence, in the case of MPEG-based video, the expected Q value can be analytically evaluated as a function of the loss rate:

$$Q = 1 - \frac{N_{dec}}{N_I + N_P + N_B} \tag{14}$$
$$N_{dec} = N_{dec_I} + N_{dec_P} + N_{dec_B}$$

where the summation at the denominator is the total number of the I, P and B frames that compose the video flow, and the numerator is the the number of the successfully decoded frames. The number of decodable I, P and B frames can be evaluated as:

$$N_{dec_I} = (1 - P_{loss})^{\overline{I}} N_{GoP} \tag{15}$$

$$N_{dec_P} = (1 - P_{loss})^{\overline{I}} N_{GoP} \sum_{i=1}^{n_P} (1 - P_{loss})^{i \cdot \overline{P}} \tag{16}$$

$$N_{dec_B} = [(1 - P_{loss})^{\overline{I} + n_P \overline{P}} + \sum_{i=1}^{n_P} (1 - P_{loss})^{i \cdot \overline{P}} (1 - P_{loss})^{\overline{B}}]$$
$$(M - 1)(1 - P_{loss})^{\overline{I} + \overline{B}} N_{GoP} \tag{17}$$

where N_{GoP} is the number of GoPs in the video flow; $\overline{I}$, $\overline{P}$ and $\overline{B}$ are the average number of packets composing frames I, P and B in a GoP, respectively; n_P and n_B are the average numbers of P and B frames in a GoP, respectively; and $M-1$ is the average number of B frames between I-P or P-P frames. Table 4 shows the results coming from our testbed, highlighting that only I and P frames were in use.

	Frame Type		
	I	**P**	**B**
Avg. frames per GoP	1	55	0
Avg. RTP packets per frame	10	6	0

Table 4: Average real GoP size (our testbed).

5.2 Testbed results

In this section, we provide the evaluation of the perceived video quality at the GCS, according to Peak Signal-to-Noise Ratio (PSNR), Structural SIMilarity (SSIM), and MOS evaluations [47]. The values can be read in Table 5. The PSNR and SSIM metrics have been calculated for each operator and juxtaposed to the aggregated multipath flow to highlight the advantages provided by the use of a multipath setup.

As a premise, it is worth noting that the proposed multipath approach achieves at least a performance level equal to that of the most performing single path. This effect can be seen in Table 3, which, as explained above, shows the results obtained for the channel models of the considered network operators. In detail, in the urban scenario, the stationary probability of being in the BAD state for the multipath case (*Aggregated*) is comparable with that of the best performing operator (WindTre). However, looking at the transition matrix, the transition probability from BAD to GOOD is significantly increased; it can be explained by a contained improvement of the video quality since the probability of no transmission errors is increased. In the suburban case, the stationary probability of being in the BAD state in the multipath case is one order lower (0.00008) than that of a single path (WindTre is 0.0006), as well as the transition probability from BAD to GOOD has increased. A multipath setup contributes in decreasing the probability of the channel of being in the BAD state thanks to the path diversity: in fact, the probability that each path is simultaneously in a BAD state is much lower, as shown in Eq. (7). In other words, it is rather likely that at least one path can properly support the video delivery. The performance improvement due to a multipath setup is also reflected in the video quality statistics, as shown in Table (5). Such results show that multipath in the suburban scenario brings a reduced improvement over the best performing operator WindTre, which already provides excellent performance in terms of MOS, SSIM, and PSNR. Differently, a significant improvement is visible in the urban scenario, in which the multipath provides better video quality than each single link. An evident quality improvement, comparing the urban case with the suburban one, can be also seen in Figures (7a) and (7b), in which path diversity in the suburban case is evidently less pronounced than in the urban one.

operator	**bad/poor**	**fair**	**good/excellent**	**avg PSNR**	**avg SSIM**	**MOS**
(U) Vodafone	7.1%	2.4%	90.5%	46.85	0.86	good
(U) WindTre	6.9%	0.7%	92.4%	45.49	0.88	good
(U) Tim	41.8%	0.7%	57.5%	33.23	0.59	poor
(U) Aggregated	7.0%	0.3%	92.7%	47.6	0.93	excellent
(SU) Vodafone	7.0%	2.3%	90.7%	46.93	0.90	good
(SU) WindTre	11.9%	0.1%	88.0%	47.41	0.92	excellent
(SU) Tim	18.0%	1.0%	81.0%	43.14	0.82	good
(SU) Aggregated	6.3%	0.1%	93.6%	47.46	0.92	excellent

Table 5: Statistics on the per-frame video quality based on PSNR [dB] (5th column) and SSIM (6th column) metrics in both urban (U) and suburban (SU) scenarios. The 2nd, 3rd, and 4th columns report the shares of video frames per opinion score. A subjective evaluation is shown in the last column according to the UAV pilot.

Furthermore, we map QoS into QoE according to Eq. (13). In this respect, we rely on PSNR only (thus neglecting SSIM) in order to be coherent with the model in Eqs. (15), (16), and (17). Table 6 reports the mapping among MOS, PSNR, and P_{loss} calculated as follows: given the mapping between the MOS evaluations and the PSNR values in Table 5, we derive the PSNR ranges for given MOS value as in Table 6. Then, the video feeds have been fragmented to

MOS	bad	poor	fair	good	excellent
PSNR (dB)	< 20	≥ 20 < 30	≥ 30 < 40	≥ 40 < 50	≥ 50
$\mathbf{P_{loss}}$	≥ 0.25	≥ 0.2 < 0.25	≥ 0.1 < 0.2	≥ 0.05 < 0.1	< 0.05

Table 6: Mapping MOS evaluation into PSNR values with respect to P_{loss}.

obtain short video sections each exhibiting only PSNR values falling into one of the five MOS intervals. In this way, P_{loss} can be calculated and mapped with the average PSNR value per video fragment; the results of such a procedure can be read in Table 6. This mapping has been used to determine the two thresholds

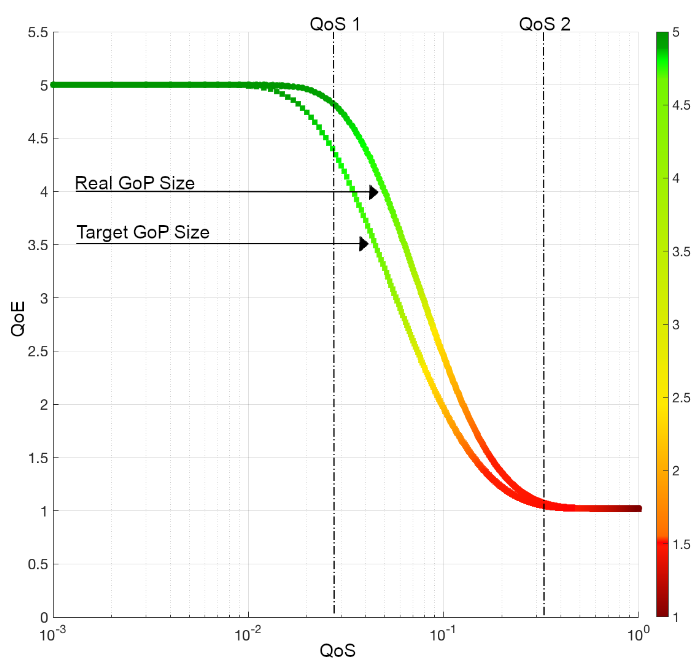


Fig. 8: Mapping QoS into QoE as a function of the P_{loss} parameter: target GoP size versus real (measured) GoP size.

QoS_1 and QoS_2 shown in Figure 8 and presented in what follows. Referring to Eq. (13), the parameters to be estimated are: k_1, set to the maximum QoE value $k_1 = 5$; k_2, set to the minimum QoE value $k_2 = 1$; η, which determines the slope of the curve in Fig. 8; and k_3, which is related to both the difference $k_2 - k_1$ and the slope of the curve. Hence, determining the thresholds $[QoS_1, QoS_2]$ limits the range of values within which the parameter η can be chosen; the same goes for the parameter k_3, which also depends on $k_2 - k_1$ as said before. The derived values are: $k_2 = 990$, $k_3 = 1.1818$, and $\eta = 33$. Thus, Figure 8 shows the relative mapping of the QoS into QoE, with the QoS degrading when moving from QoS_1 to QoS_2. The region of very high QoE (excellent), up to QoS_1, represents the case of a slight degradation of QoS with negligible effects on QoE. When the QoS degradation falls within [QoS_1, QoS_2] (i.e., the second region), QoE start decreasing (ranging from good to poor). Finally, once passed the threshold QoS_2 (third region), the QoE should be considered as bad, that is, unacceptable QoE causing users to give up on the service. According to the model presented in Section 5.1, QoS is presented in Fig. 8 as a function of P_{loss}, according to curve related to the target GoP size (90 frames as per Table 1).

Fig. 8 also shows the curve calculated by using the real GoP size (56 frames as per Table 4). The target size defines a lower bound for such a mapping because of its higher value, which would translate into higher dependency among consecutive frames, thus generating higher spatial error rates per given P_{loss}. Once such a mapping is available, a further improvement can be achieved through learning-based sender-side policies to be implemented: when no data are available, the target GoP size can be used as a lower bound for QoE. Then, in the presence of a feedback loop - common in video streaming scenarios - both the loss rate and the GoP size can be estimated, in turn allowing for a finer tuning of the model parameters. Such a mechanism can lead to a more efficient use of the network links (for instance, using a subset of links) to optimise the use of resources while contemporary satisfying the video quality objective.

5.3 Open Research Challenges

In this section, we briefly go through several open research challenges deriving from this work and described in the literature. The first one, briefly mentioned before, is related to video coding and to the possibility to leverage cross-layer designs to improve the achievable performance level. Coding-aware and coding-intrusive techniques [17] can be used in this regard. Coding-aware means that the encoding parameters are sent down to the network layer, which chooses the network path (i.e., a set of available links, in the case of multipath) that is expected to target the desired QoE at the receiver. Contrarily, coding-intrusive means that the application-layer encoding procedure is fed with networks statistics to adapt the former to the latter. Coding-intrusive techniques are more used than coding-aware ones, because single-link scenarios are commonly taken into account as reference, but the use of multipath techniques opens to the greater potential of

coding-aware techniques for two main reasons: *(i)* any decisions taken at the application layer will likely consider the link as a single (logical) one, thus forcing an algorithm to rely on average values of the network statistics; instead, a decision taken at the network layer can leverage the full knowledge of each link statistics, thus opening to more targeted strategies. Furthermore, *(ii)* flexible strategies at the network layer have the potential to provide better results in terms of QoE - for the same reason as before - also when energy constraints are taken into account, i.e., by respecting energy constraints while choosing the links. As said before, coding-intrusive techniques require cross-layer approaches [48] and more complexity hidden in the network layer. Achieving results similar to those provided by coding-aware techniques can prove challenging, but can provide a larger flexibility. Conceptually, it translates into multimedia-centric networking as alternative to the more common network-centric one.
A further challenge that requires careful attention is the packet reception order and the presence of a large number of duplicates - likely to occur if redundancy is used with multipath/multihoming, as in our scenario - because it influences how both duplicates and out-of-order packets are treated at the receiver [49]. For instance, in the case of the Gstreamer dejitter buffer, a certain number of out-of-order packets can mislead the Gstreamer pipeline into forcing a so-called *resync* (video jumping back in time) because of the erroneous (duplicated) SNs at the receiver with respect to the expected ones. Such a phenomenon hampers video continuity, and calls for optimised strategies at the receiver buffer, which must guarantee both video continuity and low playout delay, two requirements contrasting each other. In fact, video continuity would benefit of a larger buffer accumulating packets and absorbing any interference due to the network, while low playout delay dictates a very short buffer to reduce buffering as much as possible. Because of this, novel strategies are needed to better handle the requirements of critical real-time video streaming.

6 Conclusions

In this work, we have presented the results of a real testbed to deliver real-time air-to-ground multimedia feeds. The use cases of interest are those involving the use of UAVs in BVLoS conditions. We have presented an analytical framework to model the error in a multihoming/multipath setup, leveraging multiple physical channels, which in our scenario are represented by three cellular connections to opportunistically use the networks of different ISPs. Network statistics have been collected in the testbed to characterise QoS parameters of interest for a real-time multimedia system, such as loss rate and error burst length, which are used in our framework to demonstrate how the QoE can be improved at the GCS thanks to multipath features. Network diversity plays an important role in this scenario, and the more the diversity can be exploited, the more the QoE can be improved; otherwise, the system provides a performance level equivalent to that offered by the use of a single network link. The largest network diversity found in our testbed is in the urban part of it. The framework we propose herein to characterise a multipath channel is complemented by the analytical mapping of QoS into QoE evaluations, and the measurements collected in the real testbed have been used to show how a multihoming/multipath system, as the one herein proposed, can be used to target a given QoE at the GCS. Given our objective to improve the QoE at the receiver, we have shown how the MOS reported by the UAV pilot is greater (or equal, at a minimum) to the MOS the same pilot reports in the absence of multipath in both urban and suburban scenarios in the tests we carried out. The improved QoE reported by the UAV pilot is confirmed by computing PSNR and SSIM measurements on the received video feed. In future works, we will consider the use of reinforcement learning to automatically adapt the scheduling strategy to the network conditions.

References

1. X. Cao, P. Yang, M. Alzenad, X. Xi, D. Wu, and H. Yanikomeroglu, "Airborne Communication Networks: a Survey," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 9, pp. 1907–1926, 2018.
2. M. Bacco, P. Barsocchi, P. Cassará, D. Germanese, A. Gotta, G. R. Leone, D. Moroni, M. A. Pascali, and M. Tampucci, "Monitoring Ancient Buildings: Real Deployment of an IoT System Enhanced by UAVs and Virtual Reality," *IEEE Access*, vol. 8, pp. 50 131–50 148, 2020.
3. N. Nomikos, E. T. Michailidis, P. Trakadas, D. Vouyioukas, H. Karl, J. Martrat, T. Zahariadis, K. Papadopoulos, and S. Voliotis, "A UAV-based Moving 5G RAN for Massive Connectivity of Mobile Users and IoT Devices," *Vehicular Communications*, vol. 25, p. 100250, 2020.
4. B. Alzahrani, O. S. Oubbati, A. Barnawi, M. Atiquzzaman, and D. Alghazzawi, "UAV assistance paradigm: State-of-the-art in applications and challenges," *Journal of Network and Computer Applications*, vol. 166, p. 102706, 2020.
5. M. Bacco, S. Chessa, M. Di Benedetto, D. Fabbri, M. Girolami, A. Gotta, D. Moroni, M. A. Pascali, and V. Pellegrini, "UAVs and UAV Swarms for Civilian Applications: Communications and Image Processing in the SCIADRO Project," in *International Conference on Wireless and Satellite Systems*. Springer, 2017, pp. 115–124.
6. S. Bertrand, N. Raballand, F. Viguier, and F. Muller, "Ground Risk Assessment for Long-Range Inspection Missions of Railways by UAVs," in *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2017, pp. 1343–1351.
7. N. Mohamed, J. Al-Jaroodi, I. Jawhar, A. Idries, and F. Mohammed, "Unmanned Aerial Vehicles Applications in Future Smart Cities," *Technological Forecasting and Social Change*, vol. 153, p. 119293, 2020.
8. R. Muzaffar, E. Yanmaz, C. Raffelsberger, C. Bettstetter, and A. Cavallaro, "Live Multicast Video Streaming from Drones: an Experimental Study," *Autonomous Robots*, vol. 44, no. 1, pp. 75–91, 2020.
9. S. Si-Mohammed, M. Bouaziz, H. Hellaoui, O. Bekkouche, A. Ksentini, T. Taleb, L. Tomaszewski, T. Lutz, G. Srinivasan, T. Jarvet *et al.*, "Supporting Unmanned Aerial Vehicle Services in 5G Networks: New High-Level Architecture Integrating 5G With U-Space," *IEEE Vehicular Technology Magazine*, 2020.
10. M. Bacco, P. Cassara, A. Gotta, and V. Pellegrini, "Real-Time Multipath Multimedia Traffic in Cellular Networks for Command and Control Applications," in

2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall). IEEE, 2019, pp. 1–5.

11. D. Cavaliere, S. Senatore, and V. Loia, "Proactive UAVs for cognitive Contextual Awareness," *IEEE Systems Journal*, vol. 13, no. 3, pp. 3568–3579, 2018.
12. A. A. Hodroj, M. Ibrahim, and Y. Hadjadj-Aoul, "A Survey on Video Streaming in Multipath and Multihomed Overlay Networks," *IEEE Access*, 2021.
13. M. Marchese, A. Moheddine, and F. Patrone, "IoT and UAV Integration in 5G Hybrid Terrestrial-Satellite Networks," *Sensors*, vol. 19, no. 17, pp. 1–19, 2019.
14. M. Bacco, A. Berton, A. Gotta, and L. Caviglione, "IEEE 802.15.4 Air-Ground UAV Communications in Smart Farming Scenarios," *IEEE Communications Letters*, pp. 1–4, July 2018.
15. X. Lin, V. Yajnanarayana, S. D. Muruganathan, S. Gao, H. Asplund, H.-L. Maattanen, M. Bergstrom, S. Euler, and Y.-P. E. Wang, "The Sky Is Not the Limit: LTE for Unmanned Aerial Vehicles," *IEEE Communications Magazine*, vol. 56, no. 4, pp. 204–210, 2018.
16. B. Van der Bergh, A. Chiumento, and S. Pollin, "LTE in the Sky: Trading Off Propagation Benefits with Interference Costs for Aerial Nodes," *IEEE Communications Magazine*, vol. 54, no. 5, pp. 44–50, 2016.
17. S. Afzal, V. Testoni, C. E. Rothenberg, P. Kolan, and I. Bouazizi, "A Holistic Survey of Wireless Multipath Video Streaming," *arXiv preprint arXiv:1906.06184*, 2019.
18. S. Barré, C. Paasch, and O. Bonaventure, "Multipath TCP: from Theory to Practice," in *International Conference on Research in Networking*. Springer, 2011, pp. 444–457.
19. V. Singh, S. Ahsan, and J. Ott, "MPRTP: Multipath Considerations for Real-Time Media," in *Proceedings of the 4th ACM Multimedia Systems Conference*. ACM, 2013, pp. 190–201.
20. B. Kreith, V. Singh, and J. Ott, "FRACTaL: FEC-based Rate Control for RTP," in *Proceedings of the 25th ACM international conference on Multimedia*, 2017, pp. 1363–1371.
21. S. Zhang, W. Lei, W. Zhang, Y. Guan, and H. Li, "Congestion Control and Packet Scheduling for Multipath Real Time Video Streaming," *IEEE Access*, vol. 7, pp. 59 758–59 770, 2019.
22. T. Viernickel, A. Froemmgen, A. Rizk, B. Koldehofe, and R. Steinmetz, "Multipath QUIC: a Deployable Multipath Transport Protocol," in *2018 IEEE International Conference on Communications (ICC)*. IEEE, 2018, pp. 1–7.
23. W. Zhang, W. Lei, and S. Zhang, "A Multipath Transport Scheme for Real-Time Multimedia Services based on Software-Defined Networking and Segment Routing," *IEEE Access*, vol. 8, pp. 93 962–93 977, 2020.
24. B. Wang, W. Wei, Z. Guo, and D. Towsley, "Multipath Live Streaming via TCP: Scheme, Performance and Benefits," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 5, no. 3, p. 25, 2009.
25. Y. Xing, K. Xue, Y. Zhang, J. Han, J. Li, J. Liu, and R. Li, "A Low-Latency MPTCP Scheduler for Live Video Streaming in Mobile Networks," *IEEE Transactions on Wireless Communications*, 2021.
26. M. Bacco, T. De Cola, G. Giambene, and A. Gotta, "Advances on Elastic Traffic via M2M Satellite User Terminals," in *2015 International Symposium on Wireless Communication Systems (ISWCS)*. IEEE, 2015, pp. 226–230.
27. J. Wu, C. Yuen, B. Cheng, Y. Yang, M. Wang, and J. Chen, "Bandwidth-Efficient Multipath Transport Protocol for Quality-Guaranteed Real-Time Video over Het-

erogeneous Wireless Networks," *IEEE Transactions on Communications*, vol. 64, no. 6, pp. 2477–2493, 2016.
28. S. Ferlin, S. Kucera, H. Claussen, and Ö. Alay, "MPTCP meets FEC: Supporting Latency-Sensitive Applications over Heterogeneous Networks," *IEEE/ACM Transactions on Networking*, vol. 26, no. 5, pp. 2005–2018, 2018.
29. J. Wu, B. Cheng, M. Wang, and J. Chen, "Energy-Aware Concurrent Multipath Transfer for Real-Time Video Streaming over Heterogeneous Wireless Networks," *IEEE Transactions on circuits and systems for video technology*, vol. 28, no. 8, pp. 2007–2023, 2017.
30. M. Bacco, P. Cassarà, M. Colucci, and A. Gotta, "Modeling Reliable M2M/IoT Traffic over Random Access Satellite Links in Non-Saturated Conditions," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 5, pp. 1042–1051, 2018.
31. X. Xiao, W. Wang, T. Chen, Y. Cao, T. Jiang, and Q. Zhang, "Sensor-Augmented Neural Adaptive Bitrate Video Streaming on UAVs," *IEEE Transactions on Multimedia*, vol. 22, no. 6, pp. 1567–1576, 2019.
32. G. Giambene, D. K. Luong, V. A. Le, T. de Cola, and M. Muhammad, "Transport Layer Performance Combining Multipath and Network Coding in Mobile Satellite Networks," *International Journal of Satellite Communications and Networking*, vol. 35, no. 6, pp. 583–603, 2017.
33. I. W. Damaj, D. K. Serhal, L. A. Hamandi, R. N. Zantout, and H. T. Mouftah, "Connected and Autonomous Electric Vehicles: Quality of Experience Survey and Taxonomy," *Vehicular Communications*, vol. 28, p. 100312, 2021.
34. U. Engelke and H.-j. Zepernik, "Perceptual-Based Quality Metrics for Image and Video Services: A Survey," in *Next Generation Internet Networks Conference*. IEEE, May 2007, pp. 190–197.
35. M. Alreshoodi, E. Danish, J. Woods, A. Fernando, and C. De Alwis, "Prediction of Perceptual Quality for Mobile Video Using Fuzzy Inference Systems," *Transactions on Consumer Electronics*, vol. 61, pp. 546–554, November 2015.
36. E. A. A. Riker, M. Mu, and S. Zeadally, "Real-time QoE Prediction for Multimedia Applications in Wireless Mesh Networks," in *IEEE Int. Conference CCNC'12*, January 2012, pp. 592–596.
37. W. Zhang, W. Lei, S. Liu, and G. Li, "A General Framework of Multipath Transport System based on Application-Level Relay," *Computer Communications*, vol. 51, pp. 70–80, 2014.
38. H. Kim, L. Mokdad, and J. Ben-Othman, "Designing UAV Surveillance Frameworks for Smart City and Extensive Ocean with Differential Perspectives," *IEEE Communications Magazine*, vol. 56, no. 4, pp. 98–104, 2018.
39. M. Bacco, P. Cassarà, A. Gotta, and M. Puddu, "A Simulation Framework for QoE-Aware Real-Time Video Streaming in Multipath Scenarios," in *International Conference on Ad-Hoc Networks and Wireless*. Springer, 2020, pp. 114–121.
40. H. Liu and M. El Zarki, "Performance of H.263 Video Transmission over Wireless Channels using Hybrid ARQ," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 9, pp. 1775–1786, 1997.
41. N. Celandroni and A. Gotta, "Performance Analysis of Systematic Upper Layer FEC Codes and Interleaving in Land Mobile Satellite Channels," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 4, pp. 1887–1894, 2011.
42. Y. J. Liang, J. G. Apostolopoulos, and B. Girod, "Analysis of Packet Loss for Compressed Video: Effect of Burst Losses and Correlation between Error Frames," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 7, pp. 861–874, 2008.

43. J. Poncella, G. Gomez, A. Hierrezuelo, F. J. Lopez-Martinez, and M. Aamir, "Quality Assessment in 3G/4G Wireless Networks," *Business Media, Wireless Personal Communications*, vol. 76, pp. 363–377, November 2014.
44. V. F. Monteiro, D. A. Sousa, T. F. Maciel, F. R. M. Lima, E. B. Rodrigues, and F. R. P. Cavalcanti, "Radio Resource Allocation Framework for Quality of Experience Optimization in Wireless Networks," *Network*, vol. 29, pp. 33–39, November/December 2015.
45. M. Sidibe, H. Koumaras, I. Kofler, A. Mehaoua, A. Kourtis, and C. Timmerer, "A Novel Monitoring Architecture for Media Services Adaptation Based on Network QoS to Perceived QoS Mapping," *Signal, Image and Video Processing*, vol. 2, pp. 307–320, October 2008.
46. C. L. H. Koumaras, A. Kourtis and C. Shieh, "A Theoretical Framework for End-to-End Video Quality Prediction of MPEG-based Sequences," in *IEE Int. Conf. on Networking and Services*, July 2007, pp. 1–5.
47. Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image Quality Assessment: from Error Visibility to Structural Similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.
48. A. Aliyu, A. H. Abdullah, O. Kaiwartya, F. Ullah, U. M. Joda, and A. N. Hassan, "Multi-Path Video Streaming in Vehicular Communication: Approaches and challenges," in *2017 6th ICT International Student Project Conference (ICT-ISPC)*. IEEE, 2017, pp. 1–4.
49. M. Radi, B. Dezfouli, K. A. Bakar, and M. Lee, "Multipath Routing in Wireless Sensor Networks: Survey and Research Challenges," *sensors*, vol. 12, no. 1, pp. 650–685, 2012.