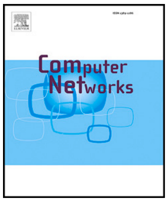
Citation: Amend, M. & Rakocevic, V. (2024). Cost-efficient multipath scheduling of video-on-demand traffic for the 5G ATSSS splitting function. Computer Networks, 242, 110218. doi: 10.1016/j.comnet.2024.110218

Computer Networks 242 (2024) 110218

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet



Cost-efficient multipath scheduling of video-on-demand traffic for the 5G ATSSS splitting function

Markus Amend [a,*], Veselin Rakocevic [b]

[a] *Deutsche Telekom, Deutsche-Telekom-Allee 9, Darmstadt, 64295, Hessen, Germany*
[b] *Department of Engineering, City, University of London, London, EC1V 0HB, United Kingdom*

ARTICLE INFO

Keywords:
Multipath
Scheduler
Video on Demand
Cost optimization
MPTCP
ATSSS
Hybrid access

ABSTRACT

In modern multiservice networks, with terminals equipped with multiple network interfaces, there is a clear trend to move from the dominating single path transport towards multipath. There are obvious benefits of the multipath service delivery – these include better resilience and improved throughput – and the standardization of multipath transport protocols MP-TCP, MP-DCCP, MP-QUIC and their usage in the 3GPP rel. 16 5G ATSSS (Access Traffic Splitting, Steering and Switching) multipath framework pave the way for broad implementation. While the field of traffic distribution algorithms for multipath transport is subject of extensive research, this paper addresses the challenge of cost-based optimization of scheduling in the multipath 3GPP ATSSS context. The paper demonstrates that there is a major conflict for the Video-on-Demand (VoD) traffic between the achievable QoE and the consumed multipath resources when a simple path prioritization algorithm – e.g. the *Cheapest-Path-First (CPF)* – is used to direct traffic. Using real network and testbed trials, this paper shows that for VoD in multipath up to 90% of the expensive path resources are consumed while QoE does not take any advantage from this, primarily because of the natural burstiness of the VoD traffic. The paper then proposes a novel service transparent and lightweight *Cost-Optimized-Multipath (COM)* traffic scheduling algorithm. Using extensive measurement of YouTube video streams and a MP-TCP implementation of the COM scheduler, this work demonstrates that – by finding the right balance between the QoE and the incurred costs – the new scheduler can provide better QoE compared to the single path transport, while eliminating the spurious resource consumption on the expensive path.

1. Introduction

While the Internet connectivity today is mainly provided over a single access network, the last decade has seen deployments which have demonstrated that multi-connectivity – simultaneous usage of more than one access technology – is able to offer better Quality of Experience for a range of applications [1–3]. Multi-connectivity enables a cost-efficient utilization of network resources, while at the same time improving connection resilience and overcoming shortcomings of the single access technology. Aggregated capacities of multiple access paths are made available, providing a platform for the users of multi-connectivity not to be affected by the interrupting handovers between access technologies or limiting throughput capabilities of the single access.

The cost efficiency of multi-connectivity solutions is one of the dominant requirements from the network operators' point of view. As a rule, the transmission costs per bit in 4G or 5G mobile networks are higher than for fixed-network connections, resulting in one cheap and one expensive transmission path when a device can choose between the two types of access. In [4], an economic benefit for customers and operators of mobile networks is identified if traffic can be shifted to Wi-Fi. Similar is confirmed in [5] where different Wi-Fi offloading strategies are discussed. For 5G networks [6] and 5G based Fixed Wireless Access [7] it is shown that the cost per bit goes down due to better spectral efficiency. Due to the higher operating costs of mobile networks and their limited capacity, especially in rural areas, the balance still swings in favor of fixed access in most scenarios. One of the first large commercial deployments of multi-connectivity, known as Hybrid Access (HA) [8] (based on GRE protocol [9], and introduced in Germany in 2014), made use of the cheapest-path-first (CPF) scheduling principle to utilize to the maximum the fixed access pipe for traffic delivery, switching to the more expensive cellular access only when the fixed access pipe became saturated.

In the networks of today, the limitations of the GRE approach in estimating volatile links, e.g., multiple radio links, mean that more mature

---

[*] Corresponding author.
E-mail address: markus.amend@telekom.de (M. Amend).

https://doi.org/10.1016/j.comnet.2024.110218
Received 16 September 2023; Received in revised form 4 January 2024; Accepted 26 January 2024

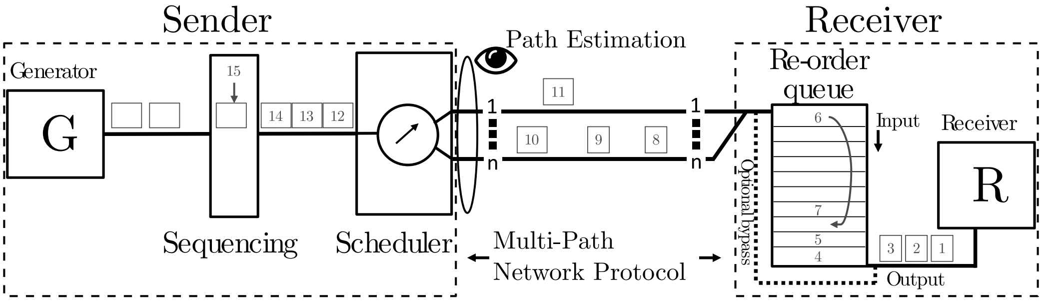


Fig. 1. Components of a multipath system with path estimation and scheduler, multipath transport protocol and sequencing for re-ordering.

and scalable support for multi-connectivity, provided by Multipath TCP (MPTCP [10]), is required. The inherent path measurement of MPTCP offers efficient service to multipath traffic management in 5G networks, leading to standardization developments within the 3GPP Access Traffic Steering, Switching and Splitting (ATSSS), specified first in the 3GPP Rel. 16 [11]. The 5G ATSSS is a multi-connectivity framework for mobile user equipment such as smartphones, but it has also been adopted in the meantime for Hybrid Access [12], substituting GRE. Similar to Hybrid Access, 5G ATSSS multi-connectivity is terminated in the network operators core at a Proxy. The Proxy ensures transparent conversion between the multi-connectivity and traditional single-path transport towards the final destination without requiring service adaptation. MPTCP's obvious limitation to TCP services has recently been complemented by the standardization work on MP-DCCP [13] and MP-QUIC [14] to support the non-TCP services. This is currently being discussed to be integrated into the enhanced ATSSS [15].

The basic components of a multipath system for simultaneous path usage are depicted in Fig. 1. The transmission over at least two paths between a Sender and a Receiver requires a sender-side traffic distribution logic — a Scheduler [16]. To avoid overloading individual path capacities, timely information about the available path capacity – provided by a path estimation entity – must be known to the scheduler. This is typically done based on measurement (e.g. congestion control in MP-TCP/DCCP/QUIC), or in static setups using e.g. the DSL synchronization rate. Typically the scheduler is agnostic to the other components of a multipath system as long as the required input parameters can be provided for the selected scheduling logic.

At the receiver side, an almost mandatory feature is the re-assembly of the transmitted data unless out-of-order delivery can be excluded in the system or the carried service is known to be robust against. Simultaneous transfer over different paths typically leads to scrambling of the original packet order, requiring the use of a re-ordering module, which takes care of that based on the sequencing information. In MPTCP, this follows the TCP inherited principle of strict in-order delivery using re-transmission. For the GRE-based Hybrid Access without re-transmission, re-ordering is time- or buffer-clocked and therefore not strictly delivering packets in order. A multipath network protocol takes care of the transmission, but can also provide sequencing, re-ordering and path estimation, like MPTCP does.

Clearly, the transmission cost in a multipath system is dominated by the decision of the scheduler how often traffic is sent over an expensive path (e.g., cellular) instead of using a cheaper path (e.g., using Wi-Fi or fixed access). For example, the Cheapest-Path-First (CPF) scheduling as used in the Hybrid Access scenarios should optimize the transmission cost as it first saturates the cheap fixed access path before overflowing into the expensive cellular access path.

However, what sounds like a simple solution to keep cost under control, fails in reality in the presence of significant amount of Video-on-Demand traffic. In Fig. 2 a measurement of a commercially deployed GRE based Hybrid Access in Germany, operated by a Tier 1 ISP and using Cheapest-Path-First scheduling, is shown. Two measurements were performed with a typical hybrid access connection with 16 Mbit/s DSL and up to 50 Mbit/s LTE, transmitting a Video-on-Demand (VoD) service (purple line) on the one hand and a linear TV service (magenta line) on the other. Video-on-Demand traffic significantly consumes cellular network resource due to its bursty nature which generates short but significant throughput demands. Contrary to this, a "flat" demand like linear TV, file downloads etc. is typically kept in the fixed access.

Considering the dominance of VoD in today's Internet (estimated by Cisco in 2021 [17] at 80% of all traffic, and confirmed by more recent studies such as [18,19]), the challenge of adopting multi-connectivity standards and solutions to the features of video-on-demand traffic stands out. In this paper we offer a new scheduling solution, designed especially for Video-on-Demand traffic, and demonstrate using extensive tests that this new cost-optimized multipath (COM) scheduling can make a significant difference in comparison to CPF. As an extension of the Cheapest-Path-First principle, COM identifies bursty traffic as generated by VoD if the transmission path provides a higher capacity than the required video playback rate and suppresses aggregation of a higher cost access path as long as QoE is not compromised. In typical Hybrid Access scenarios, this most often eliminates or at least reduces by a quarter the consumption of cellular resources, which is otherwise 20% to 90% for 1080p videos as the comparison between the COM scheduler and the CPF scheduler in Section 5 shows.

The paper firstly analyzes the existing multipath schedulers and their relationship to VoD and cost in Section 2. Section 3 provides the multipath system model and identifies the basic conflict between VoD transmission and cost based multipath scheduling. This is used to formulate the algorithm design goals, present the new algorithm, and discuss its integration in the current 3GPP ATSSS framework. The controlled and real-world testbeds used to evaluate the new algorithm are introduced in Section 4, and Section 5 presents in detail the testbed results, outlining the benefit of the new approach and demonstrate its usability in terms of cost and QoE in the field. Also possible interference with handling of services without VoD transmission characteristic is subject of investigation. Finally the conclusion recommends the usage of the new algorithm as extension to the Cheapest-Path-First principle for network operators of Hybrid Access or ATSSS where transmission cost and QoE are equally important.

2. Related work

The literature research shows a significant body of work in multipath. While most of the research and standardization efforts are focused on the multipath transport layer protocols, significant work exists in the area of multipath scheduler optimization for a wide range of use cases. The development of MP-TCP during the last decade has motivated most of these works, as MP-TCP's congestion control mechanisms offer path characteristic measurements. Hence, multipath transport over heterogeneous and volatile networks can be managed and optimized by feeding the path characteristics into the multipath scheduler logic. At the same time, the video streaming use case spawned multipath scheduling considerations, either in conjunction with MPTCP but also in conjunction with protocols in other layers or directly integrated into the video application.

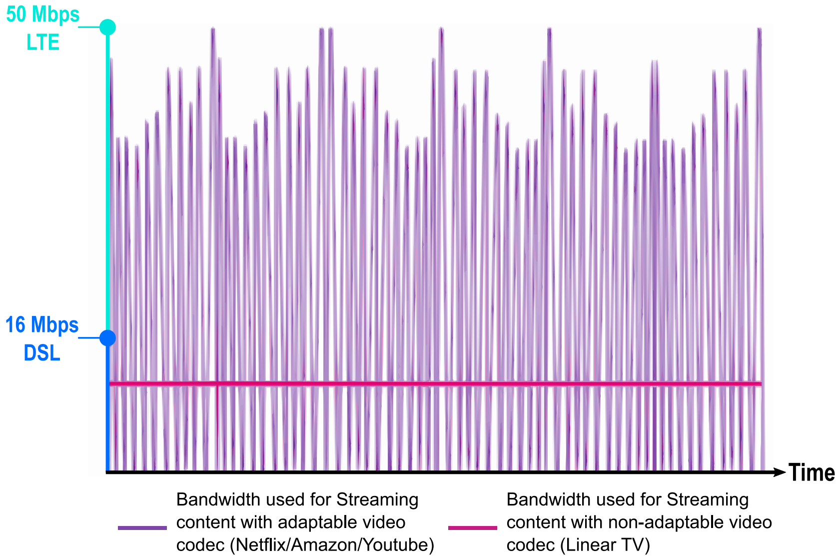


Fig. 2. Cost contradicting Video-on-Demand traffic in Hybrid Access with preferred DSL line over LTE access.

Table 1
Genesis of multipath concepts across the OSI-Layers and their path estimation and re-ordering capabilities.

Concept	Published	OSI layer	Path estimation	Re-ordering considered
LACP [20]	2008	2		✓
PPP-MP [21]	1995	2		✓
LWA [22]	2016	2	✓	✓
LWIP [22]	2016	2	✓	
GRE bonding [9]	2015	3	Sync rate	✓
ILNP [23]	2012	3		
ILA [24]	2015	3		
Shim6 [25]	2005	3		
PMIPv6 [26]	2006	3		
LISP-HA [27]	2015	3	✓	✓
HIP SIMA [28,29]	2007	3	✓	Sender predistorted
DIME [30]	2017	3		
ECCP [31]	2012	3		
Multilink proxy [32]	2009	3	✓	Delay equalization
MP-Bonding [33]	2016	3	✓	✓
MPT [34,35]	2017	3		✓
CMT-SCTP [36,37]	2006	4	✓	✓
ETOM [38]	2012	4	✓	✓
MPTCP [10,39]	2008	4	✓	✓
MP-DCCP [13,40]	2019	4	✓	✓ Part of ref. impl.
MP-QUIC [14,41]	2017	4 (5)	✓	✓ STREAM mode
MPRTP [42,43]	2010	>4	✓	✓
HTTP-RR [44]	2014	>4		✓

This section aims to answer three central questions: First, are there existing multipath concepts that discuss or even solve the problem of unwanted demand on expensive access paths described in the motivation of this work (Section 1)? Second, which existing concept is best suited to develop, implement and evaluate a solution to demonstrate its usefulness for ATSSS and Hybrid Access? Third, in the event that the first question does not provide a solution: What multipath scheduling strategies are known and what are their dependencies?

Developing multipath traffic delivery solutions for modern networks has a long history, dating back almost three decades. Table 1 gives a glimpse into the development of those solutions across the OSI Layers and compares their suitability for heterogeneous environments with volatile path characteristics by means of path estimation and re-ordering capabilities. In order for multipath solutions to persist in environments such as 5G ATSSS, path estimation is necessary for efficient scheduling – as analyzed extensively in this paper – and re-ordering is required to compensate for the different characteristics of the paths.

Most of the layer 2, 3 and >4 concepts in Table 1 have not prevailed. They have only been tested or used in limited scenarios or have failed to provide the basis for heterogeneous multipath environments. The search for a solution in these multipath concepts for the problem presented in the motivation of this work was therefore not successful. Even the special MPRTP for real-time multipath transmission of media content does not offer a solution for cost-efficient scheduling strategies.

Interesting development can be found in the area of layer 4 protocols as also confirmed by the analysis of multipath transport protocols for ATSSS in [45]. Candidates for broader deployments in the future include MPTCP, MP-DCCP and MP-QUIC. These protocols share some of the functionality with reference to Fig. 1: sequencing, path estimation using congestion control and means for re-ordering. What makes them interesting is that all three protocols are specified to be part of 5G ATSSS [11] (MPTCP) or are discussed in this context [15] (MP-DCCP, MP-QUIC). MPTCP is an enhancement of TCP and provides multipath capabilities to TCP services transparently. In contrast to MPTCP, which inherits the strict in-order delivery of TCP, MP-DCCP is a protocol for providing multipath transport for latency sensitive services and/or services with no or less demand on reliable delivery. Especially, when used in an encapsulation framework [46] it enables multipath transport for Layer 2 and 3 traffic. Something similar can be achieved by combining various QUIC functions, such as the multipath function and the DATAGRAM mode, although the re-ordering considerations are not yet at an advanced stage.

For the concepts listed in Table 1, typically the scheduling algorithms are implementation specific (not standardized). However, if the goal is to develop and evaluate a new scheduling algorithm an implementation is required, and the available research demonstrates that layer 4 protocols are able to provide the scheduling required path information from their available Congestion Control algorithms.

Congestion Control (CC) as input for scheduling decision has a huge impact on the performance as it controls path usage. For the use of multipath transmission, there are special CC algorithms such as wVegas [47] or other coupled CC algorithms LIA [48], OLIA [49] and BALIA [50]. The purpose of these methods is to ensure fairness to traffic on a single path in a shared bottleneck scenario. Apart from experimental uses, there is no known use beyond that. Moreover, the operator-controlled environments of ATSSS and Hybrid Access do not correspond to the optimization scenario for which such CC algorithms are intended as also stated in [45]. In contrast the CC algorithms Cubic – default in Linux/Mac OS/Windows [51–53] – and BBR – default for Youtube, Google search engine and Alphabet data center communication [54,55] – are used for the majority of Internet traffic and work also in multipath environments. Information provided by the CC – e.g. path availability based on congestion window and latency – can be used by multipath schedulers to identify paths which fits best to an implemented scheduling strategy. Inaccurate information from CC will lead either to underutilization or overconsumption of paths. Both have a direct impact to the multipath aggregation performance. For the purpose of this work the detailed analysis of the CC ↔ scheduler interplay is out of scope, even though concepts demonstrates another optimization potential like in [56] which presents a specialized CC for VoD transmission or in [57] which implements a combined multipath scheduling and CC scheme for maximum bandwidth utilization.

After analyzing existing multipath concepts, which only helped to understand how they work and how they relate to ATSSS and HA, several multipath scheduling strategies are now identified and analyzed in search of a solution to mitigate the observed inefficiency of additional path costs in the transmission of VoD traffic. With focus on the cost optimized multipath video streaming objective of this work, five groups of schedulers were of special interest: *Basic*, *Reduced Head-of-line blocking*, *Video optimized*, *Cost optimized* and *Cost & Video optimized*.

Basic schedulers using round-robin method [58,59] or load balancing [60] fail to respect cost policies which strictly prioritize path against each other.

For schedulers aiming to reduce *head-of-line (HoL) blocking* the foremost goal is to ensure a continuing traffic flow, overcoming any interruptions caused by disjoint path latencies. As [58] points out, opportunistic re-transmission is a general method to improve the responsiveness in a multipath system when it comes to packet loss. Schedulers with different strategies – minimizing the out-of-order delivery, the overall completion time, the shortest delivery or a combination thereof – to reduce HoL blocking are Lowest-RTT-first [16], ECF [61], DAPS [62], Blest [63], OTIAS [64] and STTF [65]. More information about their individual strengths and the path parameters taken into account for optimization can be found in [66]. Peekabo [67] claims to outperform the other schedulers using deterministic and stochastic methods for faster and more accurate decisions in heterogeneous and volatile environments. Another method to resolve HoL blocking, is the usage of different forms of redundant transmission [68]. With the clear goal of maximizing the experience of latency sensitive services, HoL blocking optimized scheduling, however, follows an arbitrary and therefore non-deterministic logic from a path cost point of view.

While some of the above HoL blocking sensitive schedulers have also proven to be efficient for video streaming, there are specific schedulers focusing on *video optimized* scheduling. PO-MPTCP [69] enhances Lowest-RTT-first scheduling by prioritizing video data over non-video one. Other approaches extend this idea to model the multipath system and consider path parameters like throughput, latency, buffering and loss rate. Using this information helps to design scheduling logic which can dispatch video data to ensure arrival pattern necessary for smooth decoding of video at the receiving end. Such schedulers are known to work optimally when the multipath system model is accurate. [70,71] demonstrate this for Scalable Video Coding (SVC), while [72] favors a cross layer approach using MPTCP for gathering path characteristics. Along this line, [73] resembles the idea from [72] and elaborates a coupling of MPTCP information with DASH, the commonly used method for VoD transmission in the Internet. A video coding scheme designed from ground up for path diversity is Multiple Description Coding (MDC) [74]. MDC encodes complementary descriptions from a media stream to built redundancy and therefore resilience to losses when it is requested over diverse paths. [75] states that MDC is suitable for scenarios in which no feedback loop is possible, but under conditions of limited path heterogeneity. This is where [76] provides a solution and combines MDC with a multipath model for optimized scheduling of high-weighted data, similar to the concept outlined in the beginning of this paragraph. Different to the approaches which integrate scheduling with the video application or the video coding, the decoupled solutions tend to optimize multipath scheduling for video-streaming independently. In that sense, [77] focuses on two different scheduling strategies for MPTCP which either prefer the path with the largest congestion window (`CWND`) or the one with the largest estimated throughput. In comparison to the Lowest-RTT-first scheduler, the results vary across the scenarios, with a strong dependency on the congestion control in use. [78] goes a different way and monitors the total and the per-path throughput to detect if a particular path contributes efficiently or if it is beneficial for the total throughput to stop transmission over this particular path. Similar to the *HoL blocking* scheduler, the *video optimized* schedulers does not use any path cost metric.

The task of a *cost optimized* scheduler is then to distribute traffic according to the given path costs. [16] describes the Strict Priority scheduler which consumes the available – non-blocked – paths in the order of cost. This corresponds to the default scheduler in the GRE based HA [9], denoted there as Cheapest-Path-First (CPF). Even if better QoE is possible, the impact of VoD over MPTCP with CPF logic was negatively evaluated in [79]. Due to the bursty nature of the VoD traffic, CPF is unnecessarily triggered to overflow into high cost path generating spurious demand without QoE benefit. This is also not changed by ACPF [80] which extends CPF to optimize aggregation performance when the multipath systems encapsulate congestion controlled end-to-end traffic.

A set of scheduler solutions which aim to combine both the cost efficiency and the optimized video transmission are the *cost & video optimized* ones. Authors of [81] proposed a multipath extension of DASH with a scheduler which favors the low-cost path over the high-cost path as the primary goal. This broadly follows the CPF logic, but is not as strict, as adjustments are made if the deadline for video chunks cannot be met. For implementation, the idea of [72] is resembled, using a cross-layer approach with tight integrated MPTCP and DASH video client. The concept demonstrates high levels of QoE as it profits from the *delay-tolerance* of VoD chunks as it is needed to cope with the delay dispersion imposed by the typical Internet connectivity. Different to DASH, which uses Advanced Video Coding (AVC) for adaptive bitrate streaming, an alternative exists. Scalable Video Coding (SVC) promises to reduce stall events by encoding chunks in ordered layers. Hence, with the reception of the basis layer it is already possible to playback the video and with requesting and receiving higher layers, video quality will be enhanced. SVC is used in [82] to demonstrate an optimal solution for retaining QoE while keeping the best possible defined link preference. In that the multipath scheduling tightly coupled with SVC in [82] is similar to [70,71], but path costs are taken into account. The authors force an application-integrated implementation over a cross-layer approach, even if a combination with MPTCP is considered possible. The disadvantage of this approach is that it requires access to video coding information and complex algorithms to demonstrate the lowest consumption of non-preferred paths, while providing highest video quality levels without any stall events.

Following the analysis of the schedulers presented above, we can see that the best decision capabilities for video optimized scheduling exist when complete and accurate information about video coding, packet delivery deadlines, playback buffer, RTT, available throughput, and other network information is available. This is also confirmed by [83]. While the schedulers which aim to reduce HoL blocking cannot provide this, at least they help to understand how to overcome certain HoL blocking scenarios, e.g., by using opportunistic re-transmission [58]. For the wide-spread video-streaming protocols based on HTTP, DASH and HLS, this is useful, as they can expect a re-transmission based and therefore reliable network transport.

In the context of our work, we can then conclude from the literature research presented in this section that schedulers for video optimized and even cost optimized transmission exist, and can be categorized as:

- *Application integrated* — Multipath scheduling within the video service
- *Hybrid* — Cross-Layer approach with interfaces between video service and scheduler
- *De-coupled* — Video service independent scheduler

The schedulers falling into the first two categories have full access to the parameters mentioned above for video optimized scheduling decision capabilities. On the other hand, due to their service dependency, this does not allow quick changes of the algorithm if required and moreover exclude support of intermediate multipath architectures like HA or 5G ATSSS which claim to be service transparent. Respecting this also excludes *Hybrid* solutions analyzing the video traffic and meta-data to gain insights into the video-transmission characteristics, which is rendered impossible if for example HTTP-based video deployments change to use HTTP/3 over QUIC. Thus, solutions in the context of this work has to be searched in the area of the *De-coupled* approaches.

Our research, however, concludes that no de-coupled – transparent to the video service – solution is available for multipath scheduling of bursty data such as VoD which optimizes both QoE and cost. The relevance of our work is well founded by the existing Hybrid Access deployments and the upcoming 5G ATSSS, which are both limited due to usage of the CPF principle for cost optimization. We will therefore focus in the next section on better understanding of the root cause of the observed QoE and cost mismatch and will develop countermeasures.

3. System model and algorithm description

3.1. System model

We observe a general case of multipath access network as depicted in Fig. 3. The network model consists of two termination points and N distinct paths denoted p_i, where $i = 1, \ldots, N$, with i denoting the individual path in the range of available paths N $\{i, N \in \mathbb{N} | N \geq 2\}$. A system with the minimum number of paths therefore consists of p_1 and p_2. All paths together form a composite multipath connection mc. The data, as a stream of Packet Data Units (PDUs), enter the mc and is split at the first termination point into multiple paths p_i, according to a scheduling logic. At the other termination point, the split traffic from the paths p_i is aggregated again and forwarded. The multipath system is characterized by giving access to the paths p_i simultaneously. This allows fine granular scheduling on the PDU level and is therefore suitable to provide an aggregated experience of the individual path characteristics within the mc.

The typical path transmission characteristics include: the amount of data B, which can be transmitted according to the path throughput, the path latency L including any buffering on the transmission path, the latency variation – Jitter – J and the loss-rate of data units R. While this definition of transmission characteristics applies to both the individual p_i and the composite mc, a resource efficient path selection process for traffic splitting (scheduling) is dependent on an additional characteristic, the path cost C.

In a resource efficient multipath system, C_i assigns therefore a cost value to each p_i with an increasing C indicating a higher cost. The determination of the path cost feature could be for example based on the actual monetary cost or on some latency-related criteria.

We can then characterize the individual paths as a function $\mathfrak{F}()$ of the above parameters, including the cost feature. $\mathfrak{F}()$ is used throughout this section as generic expression to denote all possible functions.

$$p_i = \mathfrak{F}(B_i,\ L_i,\ R_i,\ J_i,\ C_i) \tag{1}$$

whereas for the composite path, the transmission characterization is derived from the set of multiple p_i.

$$mc = \mathfrak{F}(B_{mc},\ L_{mc},\ R_{mc},\ J_{mc}, C_{mc}) \tag{2}$$

As an example, in a two-path system with cost parameters $p_1, C_1 = 0$ and $p_2, C_2 = 1$ a focus would be on cost efficiency, and would result in a scheduling logic that enforces usage of path p_1, with path p_2 (the secondary path) used only when the demand exceeds the first path capabilities. For this it is necessary to obtain path congestion status under operation, in particular for volatile environments.

From an aggregated mc perspective, the individual path characteristics contribute differently. For example, the maximum achievable throughput B_{mc} calculates as

$$B_{mc} = \mathfrak{F}(B_1,\ \ldots\ , B_i) = \sum_i^N B_i \tag{3}$$

and the overall cost C_{mc} is proportional to the **utilized** B_i and B_{mc} denoted as $B_{U\,i}$ and $B_{U\,mc}$

$$C_{mc} = \mathfrak{F}([C_1, B_{U1}],\ [\ldots, \ldots]\ , [C_i, B_{U\,i}]) = \sum_i^N \frac{B_{U\,i}}{B_{U\,mc}} C_i \tag{4}$$

On the other hand, the composite latency L_{mc}, loss-rate R_{mc} and jitter J_{mc} cannot be determined by simple addition, but by a non-linear function, as they depend on multiple factors. Section 3.2 and Section 4 discuss that the parameters leading to non-linear dependencies play a negligible role due to the VoD design with playout buffers which is able to compensate typical access latencies and it will become clear that Eq. (3) is relevant for the QoE of VoD transmission and Eq. (4) to assess its cost. Using this general model, we can then define the optimization objective of any scheduling algorithm. In the case of cost-effective multipath scheduling, the optimization objective can be defined as: *design a scheduling function to distribute PDUs on available paths to minimize the overall cost* C_{mc}. As this is a complex multi-variable optimization problem, in the rest of this section we will present a heuristic solution designed based on measurements and empirical observations, and will then analyze in detail its performance.

3.2. Limitation of CPF and new algorithm description

This section investigates the challenges Cheapest-Path-First (CPF) scheduling faces when VoD traffic is present, develops countermeasures and describes the novel COM scheduler, which is able to alleviate the shortcomings of CPF scheduling.

To understand better how a cost metric can be considered in a multipath system, we performed some measurements to assess the performance of the CPF scheduling logic in the presence of VoD traffic.

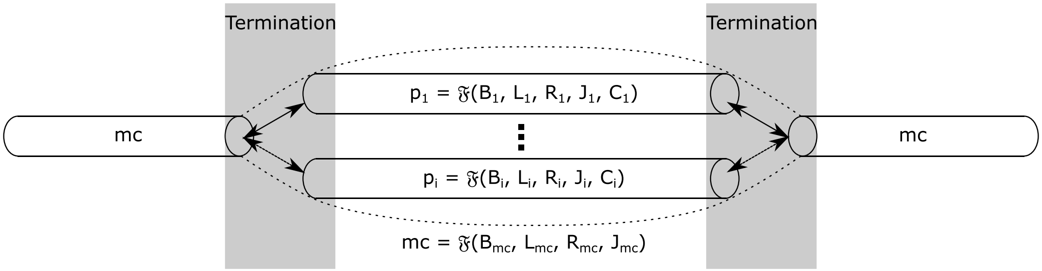


Fig. 3. Network model of a multipath-system.

We modified the MPTCP default scheduler decision logic[1] to follow a simple linear traversal logic shown in Algorithm 2, which identifies the path p_i with the lowest cost C_i available – the non-exhausted send window – for the dispatch of a data segment. This implementation of a CPF scheduler also extends the Linux file system to store a cost indicator along with the network interfaces. From this, the CPF scheduler evaluates the cost of the paths provided by the MPTCP path manager along with the send window (*SWND*) information derived from the congestion control algorithm to verify the least cost path. We use the MPTCP open source prototype with its cross-version stable scheduler implementation in this work because of the protocol's maturity (as described in Section 2) and practical relevance in the 5G ATSSS context. Because of MP-TCP's strict in-order delivery inherited by TCP, focus can be put solely on exploration of scheduling without the impact of re-ordering, as long as buffer dimensioning follows [84] and it can be assumed that the application can sustain some latency which is typically the case if TCP is the selected transport protocol. In the same way, the CPF scheduler could be used for other congestion control based multipath protocols such as MP-DCCP and MP-QUIC. Even for the GRE-based hybrid access solution, CPF could be used if the remaining bandwidth of the fixed access path is monitored using the negotiated fixed access speed information instead of the *SWND* information.

Algorithm 1: CPF scheduler logic to return the least cost path for each dispatch of a data segment.

```
Initialization:
    $paths[] \leftarrow [[p_1, C_1], ..., [p_i, C_i]]$,    $mincost \leftarrow max$

for $e \in paths$ do
    if $(e.C < mincost)$ AND $(SWND(e.p) > 0)$ then
        $bestpath \leftarrow e.p$
        $mincost \leftarrow e.C$

return $bestpath$
```

A measurement conducted using this implementation confirmed the suspicion expressed already in Section 1 that bursty traffic produced by Internet dominating VoD services presents a major challenge for the CPF scheduling logic. In a setup similar to Hybrid Access, a MPTCP enabled home gateway and a MPTCP termination point, both running MPTCP Proxy, were connected using a commercial 6 Mbps DSL and a commercial cellular LTE connection. This setup is also the one used for demonstrating the purpose of this work and is further described in Section 4. During the transmission of a 1080p VoD stream from an Internet VoD provider, 90% of the video was forwarded over LTE even if priority was on DSL, Fig. 4. Compared to a single path transmission over DSL only, no benefit in terms of QoE was measurable as in both scenarios the video ran smoothly. Hence, the indication provided in Fig. 2 is not misleading. We ask therefore: **How can spurious demand be avoided if apparently no impact on the service delivery can be monitored?**

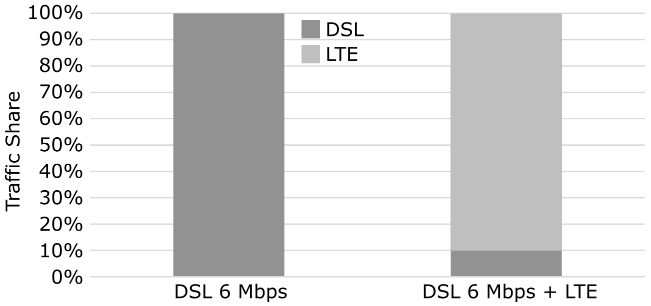


Fig. 4. Traffic share for smooth HTTP VoD streaming with 1920x1080 H.264 over 10 min comparing single path DSL and multipath DSL + LTE with MPTCP and *CPF* scheduler.

These results demonstrate that a re-design of the CPF scheduling logic is required to cope with the bursty nature of the VoD traffic. Ideally, the scheduler should avoid the usage of the high-cost path if the QoE for the user of a VoD application cannot be improved.

To define the design goals of the new algorithm, we should take a look at the nature of traffic bursts in multipath scenarios, depicted in Fig. 5. We can note that:

1. The overflowing part requires a costly transmission using the expensive path, and as such should be avoided.
2. There is a "valley" between the throughput bursts, leaving available capacity on the cheaper path unused.

Additionally, we know from the research in Section 2 that VoD within the scope of its receiver playout buffer capacity is delay tolerant.

With this in mind, we can define design goals for the new algorithm as:

- the algorithm needs to be able to detect traffic bursts causing spurious demand in multi-connectivity scenarios
- the traffic needs to be scheduled as much as possible on the cheaper resources, according to the real application and customer needs.
- the algorithm needs to be generic, simple, and service-agnostic, requiring no service-specific support
- the algorithm must not decrease the user QoE compared to single path transport but should result in a QoE similar to CPF.

If we follow Fig. 5 and use B_{cheap} to denote the capacity of the cheaper resource and B_{demand} to denote the total bandwidth demand, we can derive the following design principle for the new scheduling algorithm:

If Eq. (5) is true, then prevent access to the expensive pipe.

$$\int_0^t B_{cheap} \geq \int_0^t B_{demand} \tag{5}$$

We will call this new scheduler the *Cost Optimized Multipath (COM)*. While Eq. (5) provides many challenges to overcome, including dimensioning the time interval t, determining B_{cheap} and monitoring B_{demand},

---

[1] https://github.com/multipath-tcp/mptcp/blob/mptcp_v0.93/net/mptcp/mptcp_sched.c#L175-L184

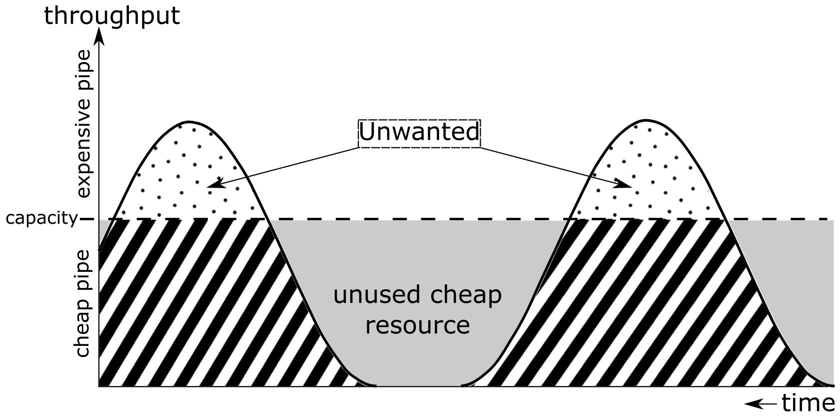


Fig. 5. Unwanted multipath operation when traffic burst overflow into costly paths.

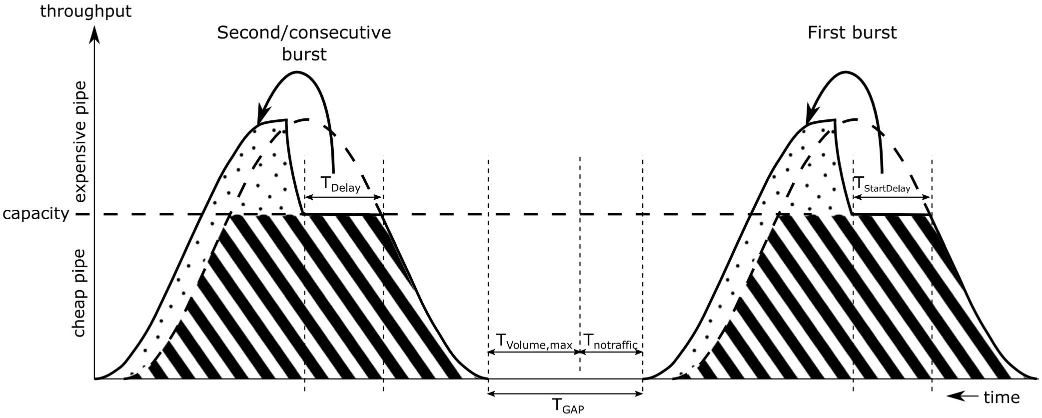


Fig. 6. COM — Practical idea to detect unsaturated link capacity based on the gap time T_{GAP} and T_{Delay} as measure to prevent spurious costly demand. $T_{notraffic}$, $T_{Volume,max}$ allow fine tuning of the gap detection, while $T_{StartDelay}$ optimizes the initial behavior.

an idea presented in Fig. 6 can be used to design a working solution. Instead of monitoring capacities and demands, the T_{GAP} size can be used to identify the saturation information. To do this, the time gap T_{GAP} between consecutive packets can be measured and compared against a threshold value $T_{GAPthresh}$. If the condition given in Eq. (6) is true, access to the expensive path is prevented for a time span of T_{Delay}, with the multipath scheduler sending the packets to the cheaper path only. As a consequence of this, bursts will be stretched over time using a larger share of the cheaper pipe. A clear advantage is that COM does not require any further measurements besides T_{GAP}.

$$T_{GAP} \geq T_{GAPthresh} \tag{6}$$

In Algorithm 2 this principle is included and together with the CPF logic of Algorithm 2 forms the COM scheduler. Whenever the CPF scheduler foresees to send traffic over the low-cost path, the code from Algorithm 2 calculates T_{GAP} and blocks after verification of Eq. (6) the expensive path or releases the path after the time T_{Delay}. During the phase where the expensive path is blocked, the CPF scheduler cannot select this path for dispatching data. This principle, with measuring the burst gap size, applies always to the path where traffic is currently scheduled to. In case a first low-cost path is saturated, COM will give access to a secondary mid-cost path. Providing a third high-cost path will be touched, if a saturation on the mid-cost path is measured, otherwise not. In a further logic not shown, the path blocking is also lifted when the low-cost path is no longer responsive due to a broken link. The variables that require preassigned values for the function of COM are listed in Table 2 and are initialized when the scheduler is engaged first in a TCP session.

As long as the achieved stretch of a burst does not let the VoD client's buffer run out of data, no additional access resources need to be used. The presence of T_{GAP} is an unmistakable sign that the buffer sufficiently re-fills, assuming that paths present in the multipath system have a bandwidth-delay product (BDP) including a safety margin for retransmissions covered by the VoD client playout buffer. Typically, the

Algorithm 2: COM — Generic code logic for gap detection and overflow prevention into expensive path executed for each data to be sent on the low-cost path.

```
Initialization:
    $T_{now} \leftarrow now$,    $T_{GAP} \leftarrow T_{now} - T_{lastdata}$

if $T_{GAP} \geq T_{GAPthresh}$ then
    SET_BLOCK_FLAG($expensive\ path$)
    $T_{block} \leftarrow T_{now}$
else if $T_{Delay} > (T_{now} - T_{block})$ then
    RELEASE_BLOCK_FLAG($expensive\ path$)

$T_{lastdata} \leftarrow T_{now}$
```

Table 2
Configurable parameters to be initialized at the start of COM.

Variable	Initial value	Description
$T_{GAPthresh}$	e.g., 600 ms	Time threshold for detection of burstiness if time distance between consecutive packets – T_{GAP} – is greater or equal
T_{Delay}	e.g., 1000 ms	Time how long the expensive path is blocked in case of burstiness
$T_{lastdata}$	Current time	Helper variable keeping the time of execution
T_{block}	0	Time of last block event

BDP is not an issue in commercial networks, as services like Youtube or others smoothly run over DSL or LTE. Therefore COM is a self-regulated algorithm when it restricts access based on Eq. (6). If an access path provides a BDP that leads to the delivery of unusable – outdated – data, a mechanism is required to remove such paths from the multi-path scheduling. Since this is a general multipath problem that also affects CPF, it is implicitly taken into account when CPF and COM are later compared in the same access environments.

The basic COM algorithm is as simple as maintaining three main variables namely the measured T_{GAP} and the configurable $T_{GAPthresh}$ and T_{Delay}. Fig. 6, however, shows also the optional definition of $T_{notraffic}$, $T_{Volume,max}$ and $T_{StartDelay}$. With $T_{StartDelay}$ the first burst in a connection can be 'flattened' even if no T_{GAP} calculation could be carried out before. This might be useful to further optimize cost, but should be used carefully to prevent unwanted QoE degradation. According to the target traffic characteristic in Fig. 5 the T_{GAP} calculation is in principle a good measure to detect spurious demand of VoD services, but any exchange of smaller amounts of data between bursts, e.g. control information or statistics between client and server, resets the calculation and makes COM assume high demand again. This type of exchange can be easily observed if, for example, a YouTube or Netflix video is paused during playback and a continuous exchange of data with URLs containing "stats" or "log" continues to take place. With the definition of $T_{notraffic}$ and $T_{Volume,max}$, T_{GAP} is split and becomes therefore more fine tuned. $T_{notraffic}$ is the time span in which any traffic will reset the gap calculation and corresponds in the absence of $T_{Volume,max}$ to the former T_{GAP}, otherwise T_{GAP} is represented by the sum of both and becomes less sensitive to smaller data exchange between bursts. This is achieved by allowing a certain amount of data within the time span of $T_{Volume,max}$ specified by the parameter V_{max}. Any data in the period of $T_{Volume,max}$ below the threshold of V_{max} does not lead to a reset of the gap calculation. For this, the algorithm remains able to distinguish between the VoD bursts without being confused by interfering data.

This enhanced logic of COM scheduler is shown in Algorithm 3 and aims to replace the code of Algorithm 2. In addition to the variables defined in Tables 2 and 3 lists the new variables for the finer granular gap detection. Also it defines the $T_{StartDelay}$ from Fig. 6, which executes `set_block_flag(expensive_path)` and `T_block = T_now` once after COM is initialized within a TCP session. For the presented code of the enhanced COM, V_{max} stands for the number of packets, but could also be used to define a volume if the calculation of V_{sum} takes into account the size of the data for scheduling.

Algorithm 3: Enhanced COM — Generic code logic for tolerant gap detection and overflow prevention into expensive path executed for each data to be sent on the low-cost path.

```
Initialization:
    $T_{now} \leftarrow now, \quad T_{GAP} \leftarrow T_{now} - T_{lastdata}$

if $T_{Delay} > (T_{now} - T_{block})$ then
    RELEASE_BLOCK_FLAG($expensive\ path$)

if $T_{GAP} \geq T_{notraffic}$ AND $T_{GAP} < T_{GAPthresh}$ AND $V_{sum} < V_{max}$ then
    $V_{sum} \leftarrow V_{sum} + 1$
    goto end

if $T_{GAP} \geq T_{GAPthresh}$ then
    SET_BLOCK_FLAG($expensive\ path$)
    $T_{block} \leftarrow T_{now}$

$V_{sum} \leftarrow 0$
$T_{lastdata} \leftarrow T_{now}$

end:
```

The impact of the measured T_{GAP} variable along with the configurable optimization parameters $T_{GAPthresh}$ and T_{Delay} defined in Table 2 and their companions for more fine granular optimization control $T_{notraffic}$, V_{max} and $T_{StartDelay}$ defined in Table 3 are discussed as part of the evaluation in Section 5.

3.3. Impact on VoD and other traffic types

The idea of COM is to be applied as a permanent replacement of the CPF without the requirement of service or traffic classification. The simplicity of COM algorithm allows us to make some assumptions on how it behaves under certain traffic scenarios, especially if there is an immediate and comprehensive need for the full aggregated throughput (e.g. file download). In this context, we can observe the following use cases:

- **VoD with $B_{demand} < B_{cheap}$**
  Due to the original bursty nature of the video data, a time gap should be visible between the traffic bursts end-to-end, as long as no intermediate bottleneck disrupts this. If COM can monitor this, it can schedule data to the cheaper path and access to the expensive path is not required.
- **VoD with $B_{demand} > B_{cheap}$**
  Different to the use case above, demanding a higher throughput than the cheaper pipe can provide will constantly fill this path and no gap will be detected. Access is therefore given to the expensive path which is now responsible to drain the overflowing traffic.
- **VoD with adjustable demand**
  It is expected that this will also match the case when a VoD service dynamically adjusts the video resolution according to the available throughput. Such a situation will lead to either the first or to the second use case. At least an upgrade to a higher resolution should not be blocked, since the gap will become shorter or even vanish.
- **File download**
  A constant file download should not be affected at all. This kind of traffic is out of scope of this work since it already works with the *CPF* scheduler, as demonstrated in [79]. In the case the scheduler verifies a constant demand on the cheaper path, justified by the nature of a file download without gaps, access is provided to the expensive path. It also does not matter if the file download demand is below or above the capacity of the cheaper pipe, as the basic *CPF* principle kicks in.
- **Bottleneck before the scheduler**
  In the case the bottleneck is not the cheaper path and the bottleneck appears before the traffic reaches the multipath scheduler, the **file download** use case is applied.

As specified in Section 3.2, the File Download case will apply whenever $\frac{Packetsize}{Bitrate} < T_{GAPthresh}$. In a scenario with a packet size of ∼ 1500 Byte corresponding to a typical specified Maximum Transmission Unit (MTU) of a network link, a new packet is scheduled every 120 ms at a bit rate of 100 kbit/s and 12 ms at a bit rate of 1 Mbit/s without taking jitter into account. If a $T_{GAPthresh}$ is above these values, a file transfer is reliably recognized and if not, the transfer rate is so low that any DSL connection can cope with the rate itself without an additional path.

3.4. COM scheduler within 5G ATSSS

The concept of ATSSS as first specified in 3GPP rel. 16 [11], defining the new feature of a Multi Access PDU (MA-PDU) session to connect a user equipment (UE, e.g. smartphone) to a data network (DN, e.g. Internet). Compared to the traditional PDU session of 3GPP a MA-PDU has two legs — multipath. One leg is over 3GPP, and the other over a non-3GPP access, for example a Wi-Fi or a wireline access. Even with a missing leg, the MA-PDU session stays functional. For the usage of ATSSS, three operating modes are defined by the S's: Steering, Switching and Splitting. In the first two modes, a specific

Table 3
Configurable parameters to be initialized at the start of enhanced COM in addition to Table 2.

Variable	Initial value	Description
$T_{notraffic}$	e.g., 50 ms	Time how long the expensive path is blocked in case of burstiness
V_{max}	e.g., 100 pkts	Max number of packets/volume during $T_{Volume,max} = T_{GAPthresh} - T_{notraffic}$
V_{sum}	0	Volume counter during $T_{Volume,max}$
$T_{StartDelay}$	e.g., T_{Delay}	Block time of the expensive path after COM initialization

access is selected for transmission with Steering as a permanent decision and Switching as a reversible decision for the affected traffic. Conclusively, this means Steering allows initial access selection, while when Switching is configured, traffic can be seamlessly shifted between the legs without interruption. However, both modes rely on single path transport. Contrary to this, the Splitting mode defines the simultaneous usage of the access legs for gaining higher throughput. This requires a per-packet multipath scheduler where ATSSS specifies the following traffic steering modes for splitting:

- smallest delay: Prefers link with lowest RTT.
- load balancing: Link sharing using a specified ratio.
- priority based: Prefers the path with higher priority. Inline with the CPF principle.

The aim of our work is to design a new *priority-based* scheduler which is optimized for VoD and inline with the 3GPP ATSSS specification.

Following the multipath 3GPP rel. 16 specification, the transport layer protocol covering **all** three S's is the MPTCP. With the MPTCP implementation of COM as part of our work, ATSSS splitting as defined in 3GPP rel. 16 is fully supported.

In Fig. 7 the underlying 5G system (5GS) architecture is shown with the 3GPP numbered reference points which denote the connection between the different network entities. For the establishment and configuration of the MA-PDU session between the User Equipment (UE) and the 5G network the control plane entities Access and Mobility Management Function (AMF) and Session Management Function (SMF) are necessary including communication across the reference points N1, N2 and N11. A successful establishment of a MA-PDU session leads also to the establishment of a user plane connection between UE and User Plane Function (UPF) over the N3 reference point and connects the UE finally with the Data Network (DN, e.g. the Internet) through N6. For the configuration of the UPF resources, N4 provides a connection to the control plane of the 5G network and is for example used to configure the ATSSS settings which affect the data transmission from DN to UE. While this is the typical communication flow when 3GPP access is used, the same reference points can be used over non-3GPP access to establish a MA-PDU leg. This is achieved by the usage of the Non-3GPP InterWorking Function (N3IWF) which connects the UE over IPsec tunnel (NWu) over the reference points Y1 and Y2 through the 5G network. The latter two, for example, represent the connection of the UE to a Wi-Fi access point and from there to the N3IWF. Finally, this architecture enables the UPF to work with a single access-independent N6 IP address, making the multipath transport transparent for the DN. While Fig. 7 shows the non-roaming (HPLMN) scenario and usage of an untrusted non-3GPP access, the two legs principle stays the same if roaming scenarios and/or trusted non-3GPP access is considered.

Between UPF and UE the multipath system of ATSSS for splitting corresponds to the one specified in Fig. 3 implementing the components of a multipath system shown in Fig. 1.

In a quite comparable way, Hybrid Access using ATSSS is specified in 3GPP rel. 16 Wireless Wireline Convergence (WWC) [85]. Compared to the UE scenario (Fig. 7) the UE is replaced by a 5G residential gateway (5G-RG) as shown in Fig. 8. Instead of the untrusted non 3-GPP access with reference points Y1 and Y2 and N3IWF, Y4 specifies a wireline access such as DSL or FTTH with the Access Gateway Function (AGF) as entry point to the 5G core. From a multipath system perspective no change is provided though, with the exception that a second MPTCP Proxy on the 5G-RG is defined for transparent multipath transport to 5G-RG connected devices. Another WWC exclusive feature is the support of 4G and 5G for 3GPP access connectivity.

This brings us in the next step to the consideration of the ATSSS rel. 16 steering functions which needs to support access aggregation and prioritization. In Fig. 9 both steering functions are depicted with MPTCP highlighted in red as the only one able to fulfill the requirement of splitting. The other steering function, ATSSS-LL, is not designed for this case because there is no means to detect an exhausted prioritized path to switch traffic to a non-prioritized path.

Following these findings, it can be noted, that both, the MPTCP implementation of the *CPF* logic (Algorithm 2) and the implementation of the COM logic in Algorithm 3 can be used without modification for the purpose of *priority-based* steering in ATSSS.

4. Methodology and testbed

In order to evaluate the performance of the COM scheduler, firstly a methodology needs to be developed to explore COM's impact and secondly, a suitable testing environment is required.

There is a clear expectation associated with the use of multipath transport that it will compensate for the weaknesses of singlepath transport and thus deliver better, or at least not worse, QoE. However, the preliminary results presented in Fig. 4 lead to the assumption that CPF has under certain scenarios no benefits for the service and just produces spurious cost when the high cost path is used. A solution space for COM develops thereof in the space presented in Fig. 10, which shows the area of tension in a multipath system as defined in Section 3 between the additional cost required and the achievable QoE gain. The dimension of the system is clearly defined by the *Non-Aggregation* (NA) operation point in the origin $O(0|0)$ and the *Unlimited-Aggregation* (UA) operation point $P(max|max)$. By this definition for a given transmission scenario, *NA* represents the single low-cost path performance, while *UA* represents the multipath system where any path can be used as desired as for example CPF scheduler implements in the Hybrid Access scenario. If one take the experiment in Fig. 4 as an example, this results in a solution space between $O(0|0)$ – DSL only test – and $P(\rightarrow max|0)$ – multipath with CPF test – since the QoE has not changed, but the cost.

Therefore, it follows that the performance of COM must prove itself against the Hybrid Access scenario with CPF (*UA*) and the corresponding single low-cost path transport scenario (*NA*), e.g. DSL only, which seems in some cases to already provide the best QoE as shown in Fig. 4.

In the light of Fig. 10 the verification of COM's operation path is the main testing objective with the following general principle: *Moving the today's Hybrid Access UA operation point within the solution space and minimizing the access to the costly resource while keeping the QoE at a sufficient level.*

Traversing Fig. 10 solution space requires a testbed equipped with the MPTCP scheduler implementation of COM in combination with the use of an VoD service as the main optimization goal of this work, but also suitable for testing non-VoD traffic to ensure safe interaction with this traffic.

As a consequence, the testbed in Fig. 11 deploys a typical Hybrid Access scenario, mapping the multipath scheduling relevant entities 5G-RG and UPF from the ATSSS based WWC architecture in Fig. 8. A Hybrid Access Router provides residential connectivity and communicates over DSL and LTE – both permitted access types in WWC – with

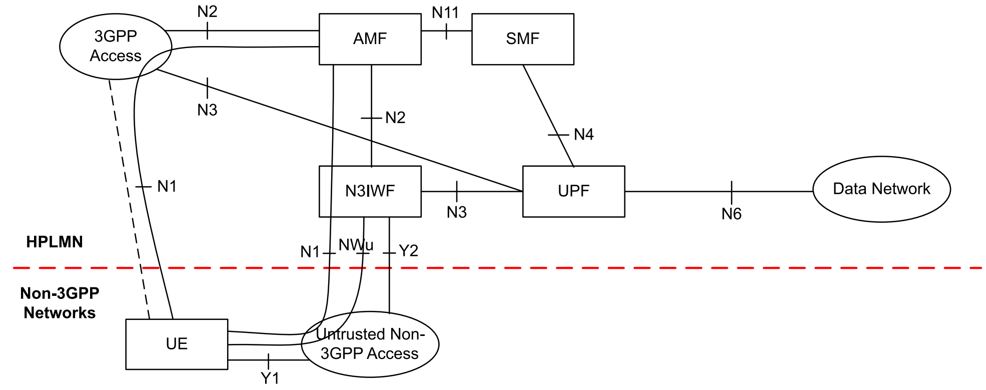


Fig. 7. 5G system with ATSSS architecture in the non-roaming scenario and untrusted non-3GPP access [11].

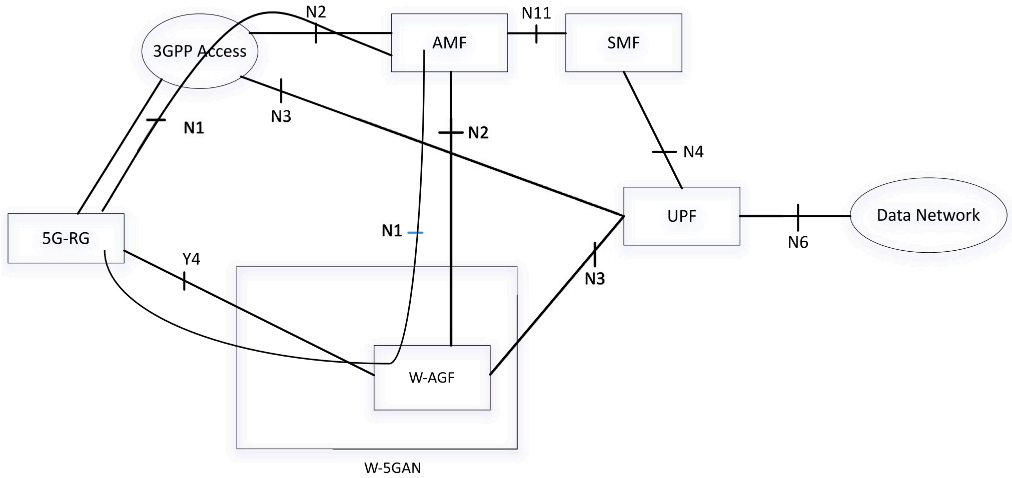


Fig. 8. Wireless Wireline Convergence architecture for Hybrid Access using 3GPP ATSSS in the non-roaming scenario [11].

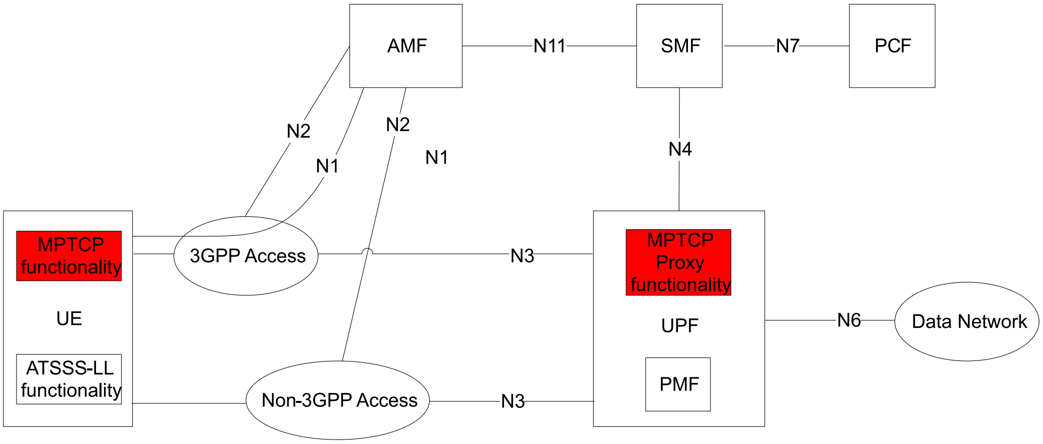


Fig. 9. 5G system with ATSSS architecture in the non-roaming scenario and untrusted non-3GPP access [11].

a Proxy entity to terminate the multipath transport and provides the gateway for Internet access. The decision to use LTE instead of 5G is due to the lack of 5G connectivity at the test site. However, this has no impact on the evaluation, as specification of LTE and 5G characteristics for Mobile Broadband (MBB) connectivity only show differences in the peak data rate. With the implementation of MPTCP and a TCP Proxy on Router and Proxy, TCP communication between services originating in the Internet and clients behind the router, are transparently enabled for multipath transport. Both, Router and Proxy implements the COM and CPF scheduler as per Algorithm 3 and Algorithm 2 and can be configured to apply one of both or transport over the cheap singlepath. The Proxy scheduler takes care of the downlink traffic from Internet to client and the Router scheduler for the uplink traffic from client to Internet.

For the purpose of evaluation, the cost impact is derived by comparing the overall cost C_{mc} (Eq. (4)) between the individual test scenarios: Singlepath, CPF and COM. The QoE can be assessed in a similar way. With VoD as service under test the QoE measurement can follow the ITU P.1203 model for gaining a MOS value. A MOS value expresses the QoE for video and audio content in a typical range between 1 (bad)

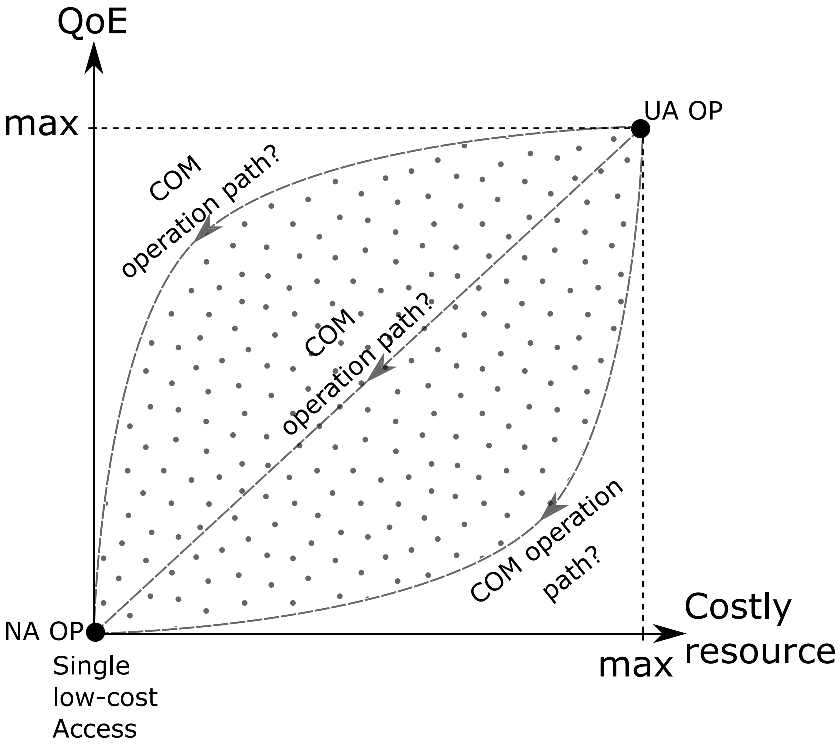


Fig. 10. COM solution space considering QoE and costly resource consumption.

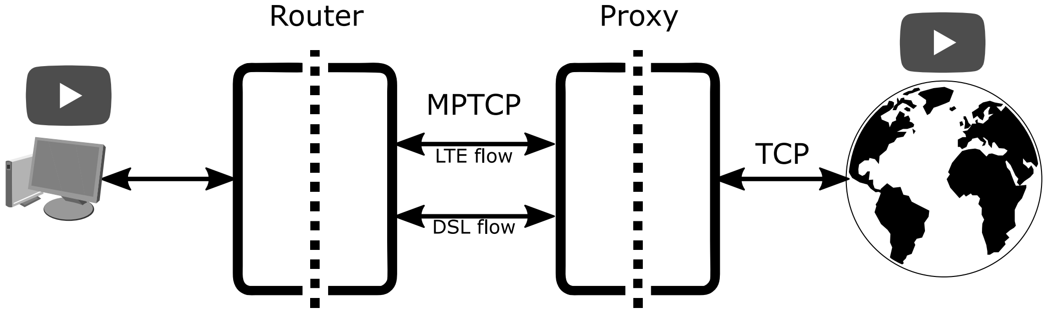


Fig. 11. MPTCP testbed for verification of COM impact over online services.

and 5 (excellent). However, this model is quite complex to implement as also outlined in [86–90].

However, a thorough analysis of the ITU MOS model [91] outlines that as long as the video frame rate can be considered stable – true for common VoD – and audio is out of scope (due to lower bitrate demand assumed to follow video QoE) three factors impact the QoE: the time $t_{initial}$ needed after requesting a video to start the playback, the time t_{stall} which covers the interrupts afterwards and the delivered video resolution r. In Section 3.2 it is discussed that the buffering concept of VoD provides delay tolerance and is typically designed to compensate for the occurring latency, jitter and loss of a 4G/5G or fixed access. If this applies to the individual accesses, it also applies to the combined accesses and makes the QoE dependent on the achievable throughput defined in Eq. (3).

Although in Section 2 the interplay between CC and multipath scheduler is excluded from the scope of this work, it requires the selection of a CC for the operation of MPTCP. The approach is to verify the co-existence of multipath scheduler with the most common CCs used for VoD transmission, Cubic and BBR. In Section 2 it is outlined that Cubic's ACK-clocked mechanism and the model-based BBR are used by the great majority of VoD applications today. It is expected that this approach will produce a trend in the results for the two most important CC algorithms that also applies to other CC algorithms. The only impact that may occur with other CC algorithms is that the maximum data rate that can be achieved over a transmission path is affected. This will of course affect the absolute performance of the CPF, COM and single cheap path transmission test scenarios, but in relative terms it will follow the above trend and, most importantly, it will not change the scheduling logic. This statement also includes the CCAs optimized for multipath transmission which mainly see their advantage on fairness to prevent that the combined throughput of a multipath connection does not take up more than its fair share over potentially shared bottleneck links. These are also not relevant because the target scenarios ATSSS and HA typically have no shared bottleneck or if so, for example in the case of Wi-Fi over N3IWF (Section 3.4) or the cellular link itself, tend to have per-user queues on lower layers. Finally, this approach is also seen in line with the requirement of this work, that the research regarding optimized multipath scheduler has no dependency to a particular multipath network protocol and therefore can be implemented in MPTCP – uses CC – but can also be implemented in a different protocol like GRE based HA — uses no CC.

The testbed enables tests with both static and adaptive resolution videos from Youtube as the leading VoD service. Pre-tests with other VoD applications from *Vimeo* or *hls.js* confirmed that those services all rely on the same transmission technologies DASH or HLS and interact with COM in the same way. Links with different DSL throughputs from a commercial DSL offering allow to vary the low-cost path characteristic, while LTE is provided by a public commercial network. The selection of commercial access paths follows the idea of coming close to a real implementation of Hybrid Access or ATSSS. For this reason, not all access characteristics are known or can be measured continuously. However, with regard to the comparison of CPF and COM, the absolute performance values of the accesses are less relevant, as the focus is on the relative performance gain. With the hardware selected for the Router and the Internet located Proxy Server it is ensured that no throughput bottleneck is created by low Central Processing Unit (CPU) or Memory (RAM) performance. Both entities are equipped with Linux Kernel 4.9, MPTCP Linux reference code 0.93[2] and tcp-intercept.[3] Further settings configure a path manager enabling only one MPTCP subflow per link or access, TCP buffer settings according to [84] considering the expected sum bandwidth and the typical Linux provided Cubic and BBR congestion control mechanisms. Unless specified otherwise, `tcp_rmem=tcp_wmem=4096 1048576 10048576` defines a maximum TCP send and receiver buffer of 10MB. The respective traffic scheduler and configuration under test is always deployed on each side of the MPTCP termination points. If multipath is activated, the DSL low-cost path is preferred over the LTE path, unless singlepath DSL usage makes a prioritization useless. Whenever COM is used for measurement, $T_{StartDelay}$ is set to the value of T_{Delay}.

At the client side a scripted environment uses Chromium browser to request automatically a full HD (1080p) Youtube[4] video over TCP. To avoid effects from arbitrary advertisements during start or run of the video a Chromium plugin *uBlock Plus Adblocker* is used. While this helps to make test iterations reproducible, it also uses the Youtube embedded player to load only the video and no surrounding information which are usually displayed on the Youtube website. Along with the embedded player the Youtube Iframe API[5] provides access to the information required to determine the QoE parameters identified above. Finally, the Chromium plugin *Youtube Auto HD + FPS* allows to specify for a certain video resolution and gets rid of Youtube's by default enabled Adaptive Bitrate Streaming (ABR) which lowers the resolution if throughput is not sufficient.

5. Results and analysis

When COM was initially motivated in [79], $t_{GAPthresh}$ for detecting burstiness and T_{Delay} for preventing temporarily access to high cost path formed the parameter set. In [79] it was demonstrated in a local testbed that the high cost path consumption significantly reduces with COM.

In this paper the focus is on the enhanced COM logic with new $t_{notraffic}$ and V_{max} parameter and COM's interplay with the Internet VoD traffic, and this section includes a range of testing results. Firstly, Section 5.1 demonstrates results of extensive tests with controlled VoD traffic over the Internet, used to determine the initial COM parameter set. This reduces the number of measurement variables to T_{Delay} and DSL throughput. This is then further used for testing with Youtube traffic in Section 5.2 to measure the cost and QoE when VoD with static video resolution is scheduled. In a further step in Section 5.3 this measurement is enhanced towards VoD with dynamic video resolution, which needs a more comprehensive QoE consideration. Finally, Section 5.4 summarizes and analyzes the measurement results.

All tests or better data points presented throughout this section were conducted once within the testbed using the DSL and LTE access as described in Section 4. Collecting only one test sample is not a limitation, as the evidence of validity is drawn from the trend of results. To avoid the test system being biased by cached values from previous tests, the system was reset by a reboot after each run of a sample.

To better understand the result charts, it is helpful to know that lower values in the line charts and more blue colored bars in the bar charts indicate better performance compared to other results in the same chart. All charts are organized so that they are plotted over T_{Delay} with the leftmost value $T_{Delay} = 0$ corresponding to the CPF principle and the rightmost value $T_{Delay} = inf$ corresponding to single usage of low-cost path.

5.1. Determination of COM initial parameters

For the purpose of testing COM across different Internet use cases, it is necessary to derive an initial parameter set. This means basically to evaluate if $t_{GAPthresh} = 600$ ms as selected in [79] continues to be appropriate and moreover to determine reasonable values for $t_{notraffic}$ and V_{max}. This is achieved by setting the parameters to static values one after the other and the parameter to be determined is variable. This is first implemented for $t_{GAPthresh}$ in Section 5.1.1 and then for $t_{notraffic}$ and V_{max} in Section 5.1.2. The effect of these two assessments on the application of T_{Delay}, the actual differentiator and optimization parameter for CPF, is examined and evaluated in terms of costs and QoE.

5.1.1. Determination of $t_{GAPthresh}$

In Figs. 12–14 the impact of $t_{GAPthresh}$ across different DSL throughput of 1 Mbps, 2 Mbps and 6 Mbps is investigated within a range of T_{Delay} from 0 s–10 s.

It must be noted that $T_{Delay} = 0$ s corresponds to the CPF scheduling principle – non effective COM – while the larger T_{Delay} becomes, the traffic is finally scheduled on DSL only. This is in particular because an initial delay $T_{StartDelay} = T_{Delay}$ after connection establishment is applied without needing a $t_{GAPthresh}$ calculation. Based on experiments and observations, a static $t_{notraffic} = 50$ ms and $V_{max} = 100$ pkts was configured to keep the focus on the change of LTE consumption and number of freezes at $t_{GAPthresh} = 200, 400, 800$ ms. In the case T_{Delay} increases, this reduces the achievable throughput B_{mc} in the multipath system. The higher T_{Delay} is configured, the more B_{mc} is reduced and causes an increasing number of freezes due to insufficient transmission capacity to fill the clients' playout buffer. Even if the number of freezes does not give all details of the QoE measurement, e.g. the length of freezes, it is sufficiently accurate to get an indication. All three results streaming a 1080p video[6] show LTE consumption which is decreasing with the increasing T_{Delay}, while freezes occur less and later are appearing later in time, with this point increasing with the increasing DSL throughput. This is an expected result from [79], however the focus is still not on the absolute gain, but rather on the relative trend. This development shows that the different values of $t_{GAPthresh}$ have minimal impact on the LTE share, as lines are close together — at least for 200 ms and 400 ms. Especially at higher T_{Delay} a $t_{GAPthresh}$ of 800 ms has smaller disadvantages. The opposite can be found when analyzing the number of freezes, as a result of greater use of the LTE capacity. Overall, a $t_{GAPthresh}$ between 200 ms and 400 ms does not result in performance differences, while between 400 ms and 800 ms a trade-off between the video freezes and the LTE capacity consumption exists. **As a reasonable trade-off further measurements are continued with** $t_{GAPthresh} = 600$ ms as it is assumed that this provides a compromise between both dimensions. It should be noted that this is a snapshot that leads to this result based on the static assumptions of $t_{notraffic}$ and V_{max} that were determined empirically. It may be that there are constellations that achieve even better overall results in combination. However, the primary goal here is not to find the best parameter set,

---

[2] https://github.com/multipath-tcp/mptcp/tree/mptcp_v0.93

[3] https://github.com/VRT-onderzoek-en-innovatie/tcp-intercept

[4] https://www.youtube.com/watch?v=aqz-KE-bpKQ

[5] https://developers.google.com/youtube/iframe_api_reference

[6] Big Buck Bunny 1080p video consumed with hls.js demo server.

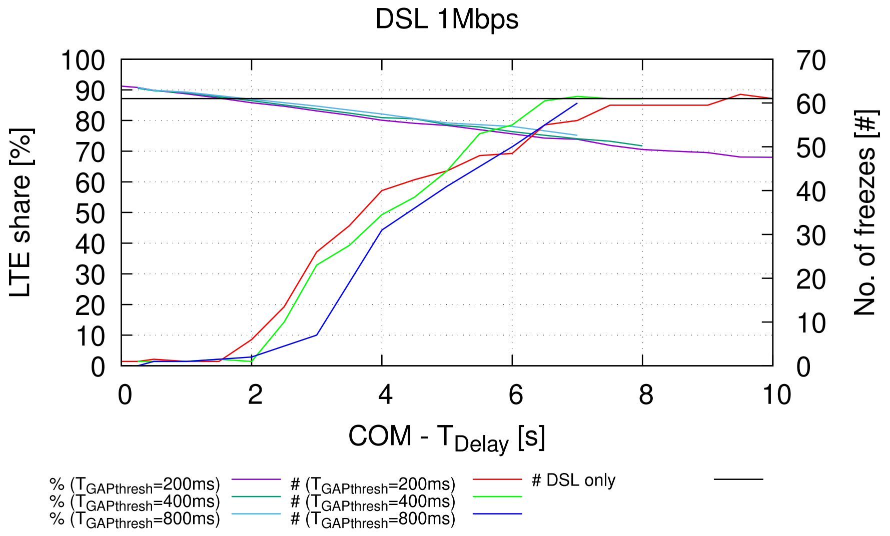


Fig. 12. Variable $t_{GAPthresh}$ at 1 Mbps DSL rate, $t_{notraffic} = 50$ ms and $V_{max} = 100$ pkts.

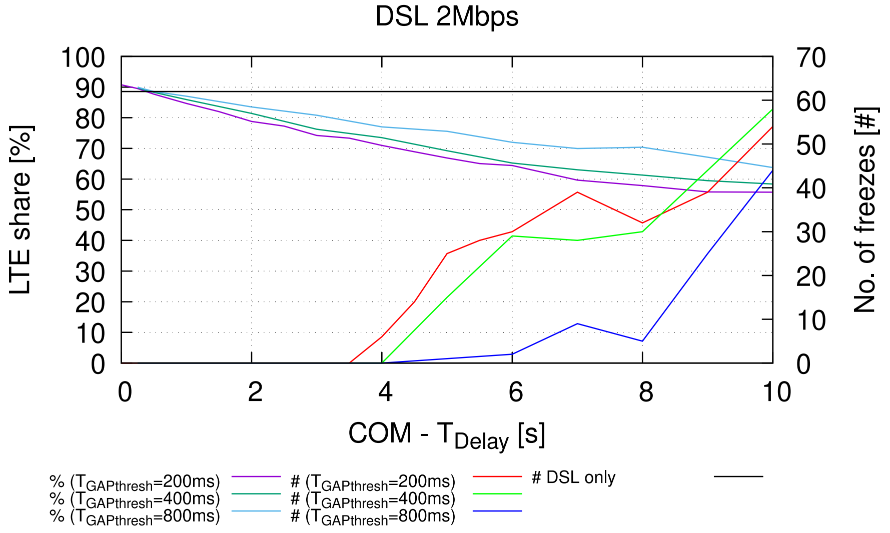


Fig. 13. Variable $t_{GAPthresh}$ at 2 Mbps DSL rate, $t_{notraffic} = 50$ ms and $V_{max} = 100$ pkts.

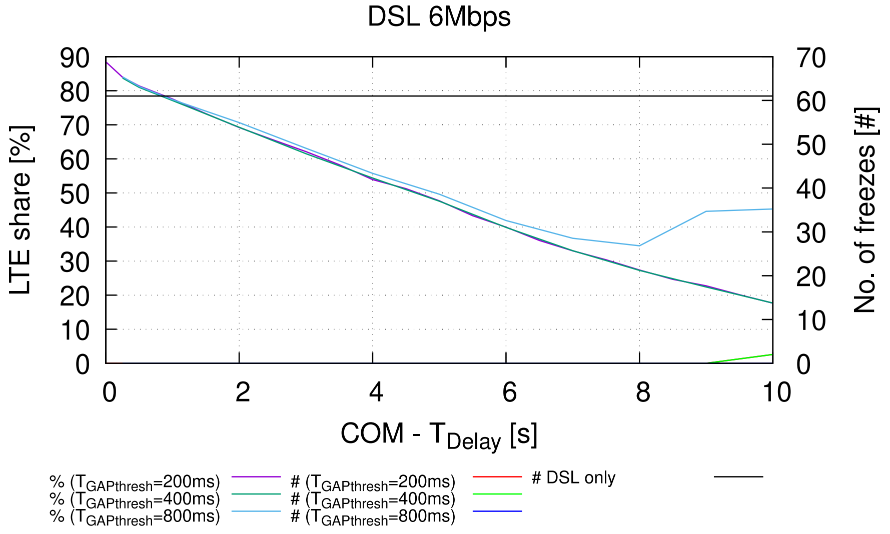
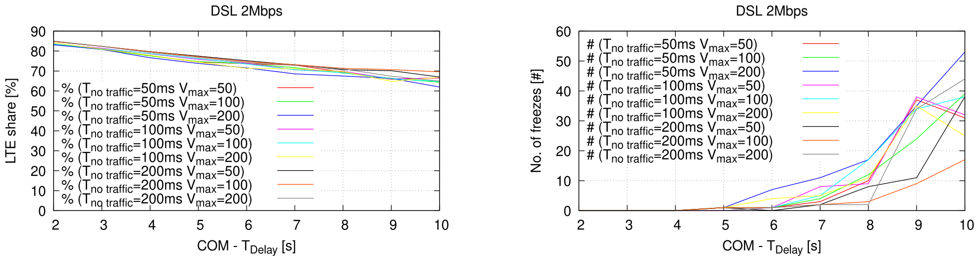


Fig. 14. Variable $t_{GAPthresh}$ at 6 Mbps DSL rate, $t_{notraffic} = 50$ ms and $V_{max} = 100$ pkts.



(a) LTE share (b) Video freezes

Fig. 15. Variable $t_{notraffic}$ and V_{max} at 2 Mbps DSL rate and $t_{GAPthresh} = 600$ ms.

but to find a good starting point for testing the performance of COM. Tests at higher DSL throughput, e.g., 16 Mbps, as well as running the measurement series with 720p video streaming confirmed the analysis above showing the same trend of results in respect to $t_{GAPthresh}$.

5.1.2. Determination of $t_{notraffic}$ and V_{max}

For the evaluation of $t_{GAPthresh}$, static values of $t_{notraffic} = 50$ ms and $V_{max} = 100$ pkts were used as this showed good results during some earlier experiments. As part of the next step to determine an initial COM parameter set, $t_{GAPthresh} = 600$ ms is used as static value and $t_{notraffic}$ and V_{max} are varied for the 2 Mbps DSL access. The latter results from the fact that the 2 Mbps evaluation in Fig. 13 shows the largest variance of LTE usage and freezes compared to the other presented DSL throughputs. As the upfront experiments with $t_{notraffic} < 50$ ms did not provide any positive results, the exploration range is set to $t_{notraffic} = \{50, 100, 200\}$ ms using 50 ms as starting point and $V_{max} = \{50, 100, 200\}$ pkts with 100 pkts as medium value. Again the impact on the LTE share and the number of freezes is monitored in Figs. 15(a) and 15(b). **Also here, a higher LTE share keeps the freezes low and vice versa. It is therefore decided to continue with the results that give a line in the center of the outer lines which is the set of $t_{notraffic} = 50$ ms and $V_{max} = 100$ pkts.** It should be noted that this is a snapshot that leads to this result based on the static assumptions of $t_{GAPthresh}$ that were determined in previous Section 5.1.1. Similar as already outlined in Section 5.1.1, it may be that there are constellations that achieve even better overall results in combination. However, the primary goal here is not to find the best parameter set, but to find a good starting point for testing the performance of COM.

Overall, the conducted tests have shown that cost and QoE are mainly impacted at lower throughputs of the low cost path, when the identified parameters are varied. The multiple degree of freedoms of the COM scheduler are reduced with the identified parameter set for $t_{GAPthresh}$, $t_{notraffic}$ and V_{max}, to solely T_{Delay}. With that, all future evaluations can focus on investigating the impact of T_{Delay} on cost and QoE at different throughput values of the low-cost path.

5.2. Youtube measurement with static video resolution

After determining the initial COM parameter set, the actual verification of COM focuses on a detailed analysis of the COM-T_{Delay} parameter compared to the CPF principle ($T_{Delay} = 0$) and single usage of the low-cost path ($T_{Delay} = T_{StartDelay} = inf$). With the LTE share – consumption of the high-cost path – and QoE parameters for video resolution r, initial load time $t_{initial}$ and buffering time during playback t_{stall}, the individual gains can be calculated. To cover realistic scenarios, the low-cost path throughput (DSL) is varied in the range 1 Mbps–100 Mbps. This range corresponds to real DSL deployments and results will show that still higher throughputs will not deliver meaningful insights into the effect of COM. In principle, the LTE access is not a limiting factor as up to 300 Mbps were available during measurements. The service under test was selected to be Youtube as a major VoD provider. Since VoD allows video to be sent with static or adaptive video resolutions, both scenarios are under investigation consuming a 1080p video[7] for 300 s. The measurement time $t_{measurement} = t_{playback} + t_{initial} + t_{stall} = 300$ s is long enough to allow a reasonable playback time to show significant results and avoid major outliers. The measurement time includes the request for the video, but not the setup of the MPTCP session in advance. A selection of 1080p resolution is considered fair as this corresponds to typical screen resolutions and available encoding at the VoD providers. Nevertheless, one can conclude from the following results for 1080p, which impact the video resolution settings have. Due to the fact that COM is implemented on the MPTCP layer, Congestion Control using Cubic and BBR applies as recommended in Section 2.

In the first set of results Figs. 16–18, the static 1080p Youtube video is evaluated, continuing with a T_{Delay} range {0, 0.25, 0.5, 0.75, 1, 1.5, 2, 3, 4, 5, 7, 10, inf}. With pure DSL use ($T_{Delay} = inf$), the LTE consumption is not present, and the highest possible t_{stall} and $t_{initial}$ have an impact on QoE. On the opposite, the CPF principle ($T_{Delay} = 0$) with access to the largest throughput has the drawback of the largest LTE usage while QoE results demonstrate the best experience. In the range 10–100 Mbps, it can be stated that single DSL provides the same QoE as CPF, basically meaning that in the latter case any LTE consumption is spurious demand and therefore unnecessary cost creation. This means that, with $T_{Delay} = \{2, \ldots, 4\}$ s, the usage of LTE can be safely reduced to almost zero without impacting QoE. This is a significant gain over the CPF-like LTE usage of 20%–70%. For lower throughputs an increasing T_{Delay} means lower QoE compared to CPF. On the other hand, this is considered acceptable, as it is still much better than DSL only. Looking into the same range of $T_{Delay} = \{2, \ldots, 4\}$ s, only $t_{initial}$ causes some longer loading time of the video, while LTE consumption goes down to in maximum one eight (6 Mbps), a half (3 Mbps) or three quarter (1 Mbps). These observations are mainly for Cubic but BBR results show the same trend, even if the LTE consumption has a minimal flattened slope, which leads to a QoE only impacted at higher T_{Delay}.

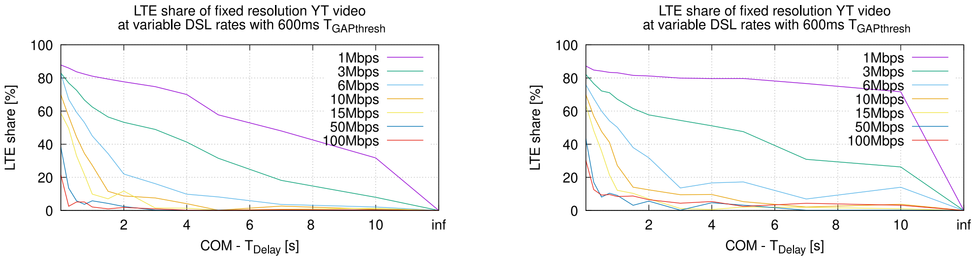
5.3. Youtube measurement with dynamic video resolution

In the case of video streams which adapt the resolution according to the available throughput, it is particularly exciting to evaluate the interaction with COM. The setup remained unchanged with regards to the measurements of the static resolution video case, even the same video sequence has been used. The only difference was the activation of the adaptive video resolution. In this more complex scenario, the representation of QoE results moves away from line graphs to bar diagrams as this allows us to capture the different shares of the video resolutions (144p, 240p, 360p, 480p, 720p, and 1080p) and the loading times $t_{initial}/t_{stall}$ during playback. A bar represents $t_{measurement}$ with the different shares of $t_{initial}$, t_{share}, and $t_{playback}(r)$ with video resolution $r = \{144p, 240p, 360p, 480p, 720p, 1080p\}$.

The LTE consumption shown in Fig. 19 is mostly similar to the static video resolution results in Fig. 16. Without T_{Delay} configured –

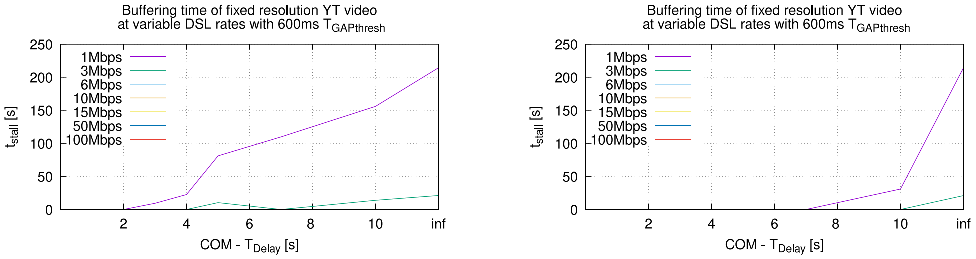
---

[7] https://www.youtube.com/watch?v=aqz-KE-bpKQ



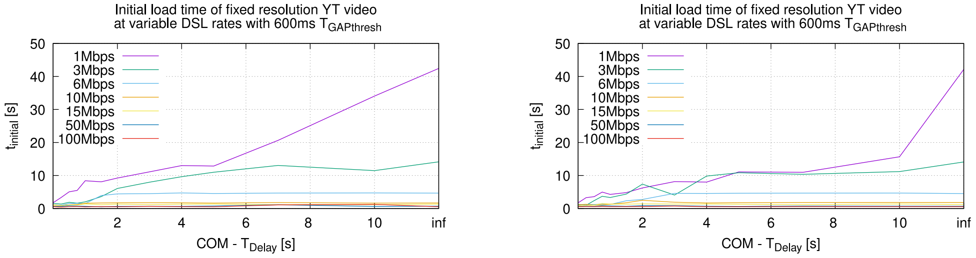
(a) Cubic (b) BBR

Fig. 16. LTE share of static 1080p YT video at variable DSL rate, $t_{GAPthresh} = 600$ ms, $t_{notraffic} = 50$ ms and $V_{max} = 100$ pkts.



(a) Cubic (b) BBR

Fig. 17. Buffering time of static 1080p YT video at variable DSL rate, $t_{GAPthresh} = 600$ ms, $t_{notraffic} = 50$ ms and $V_{max} = 100$ pkts.



(a) Cubic (b) BBR

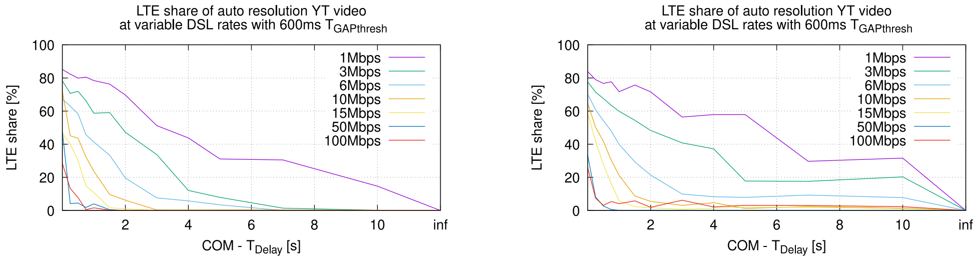
Fig. 18. Initial load time of static 1080p YT video at variable DSL rate, $t_{GAPthresh} = 600$ ms, $t_{notraffic} = 50$ ms and $V_{max} = 100$ pkts.

CPF behavior – the LTE share is identical, which is confirmed by the streamed video resolution shown for Figs. 20 (1 Mbps), 21 (3 Mbps), and 22 (6 Mbps). As CPF does not limit the available throughput, 1080p video streaming applies, as confirmed by the almost fully blue bars at $T_{Delay} = 0$. For all further T_{Delay} which show larger blue bars – 1080p streaming – the composition of the lines follow the results of the static video resolution playback. This is especially true for the DSL throughput ≥ 6 Mbps, which is not shown here because all bars are blue at any T_{Delay}. Differences are mainly visible for the 1 Mbps and 3 Mbps results. In both cases, video resolution falls below 1080p which leads to a greater drop in the LTE share compared to static video resolution. But even if the video resolution adapts to the lower values at certain T_{Delay}, 3 Mbps results at least provide resolution of 720p for $T_{Delay} > 3$ s. In the case of 1 Mbps this has higher dependency on Congestion Control, but a drop to 720p happens for Cubic for $3\text{ s} < T_{Delay} \leq 4$ s followed by drops to 480p, while BBR is even more vulnerable to lower T_{Delay} showing shares of 480p earlier. Most likely in this limited range COM interferes with BBR's mechanism to find the optimal operation point. Another remarkable development compared to the static video resolution case is that t_{stall} and $t_{initial}$ are quite low. The adaptive resolution mechanism reduces stalling and initial loads to a minimum — if necessary, this is compensated by a lower resolution. COM does not seem to change this in any way.

To emphasize the understanding of the results from a cost perspective: *If DSL only in the very right bar is shown blue, CPF in the very left bar cannot make it better.*

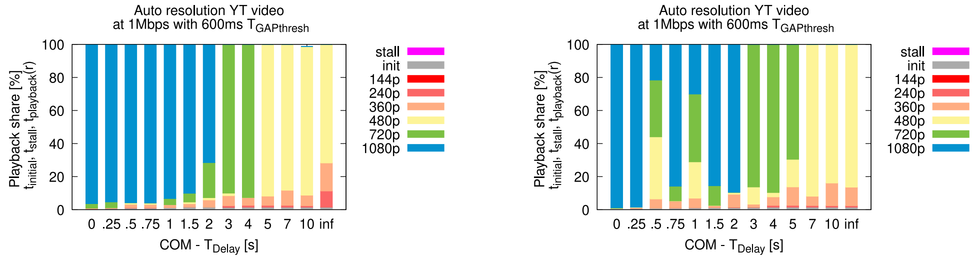
This basically means when the best QoE can already be provided over the single low-cost path – the DSL path –, CPF has no advantage. Considering, however, the high CPF-induced LTE share across the DSL throughput range of 1–100 Mbps, significant costs are raised without any benefit. COM scheduler addresses this by lowering or removing the LTE share when Hybrid Access is used to provide connectivity, while not adding any QoE advantage over the single DSL path use.

The main conclusion from the QoE perspective can be defined as: *If DSL-only is not blue, COM provides always better QoE up to the CPF level.* In the tested range of COM parameters, COM (and eventually CPF) always provides better experience compared to single-path DSL. Depending on the direction from which one wants to optimize, either the CPF QoE, or the single-path DSL QoE can be defined as the benchmark. If selecting the latter, a T_{Delay} of 4 s seems to be efficient, as it provides across all DSL throughputs at least 720p video resolution (except some 480p at BBR 1 Mbps), and when DSL provides more than 3 Mbps, the resolution goes up to 1080p. In contrast, the cost is cut by a half for 1 Mbps DSL when Cubic CC is used and to about one third when BBR is used. For DSL at 3 Mbit/s, the costs for Cubic are close to zero, while the same trend is observed for BBR, albeit with a slight shift from 6 Mbit/s



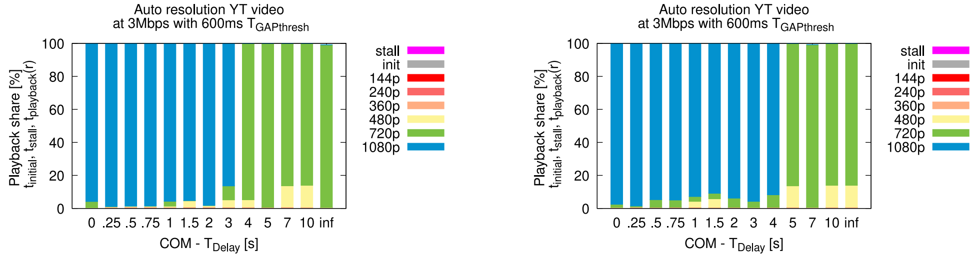
(a) Cubic (b) BBR

Fig. 19. LTE share of adaptive resolution YT video at variable DSL rate, $t_{GAPthresh} = 600$ ms, $t_{notraffic} = 50$ ms and $V_{max} = 100$ pkts.



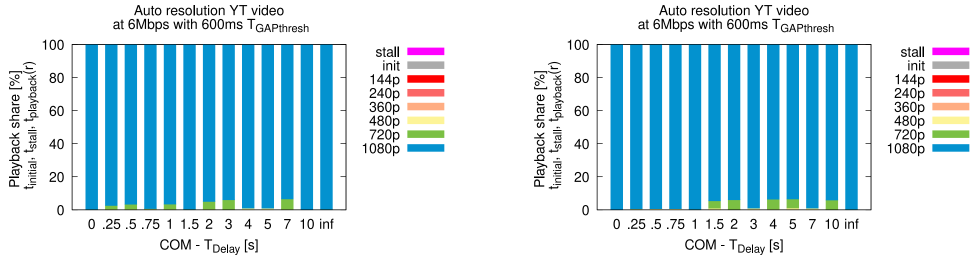
(a) Cubic (b) BBR

Fig. 20. QoE parameters of adaptive resolution YT video at 1 Mbps DSL rate, $t_{GAPthresh} = 600$ ms, $t_{notraffic} = 50$ ms and $V_{max} = 100$ pkts.



(a) Cubic (b) BBR

Fig. 21. QoE parameters of adaptive resolution YT video at 3 Mbps DSL rate, $t_{GAPthresh} = 600$ ms, $t_{notraffic} = 50$ ms and $V_{max} = 100$ pkts.



(a) Cubic (b) BBR

Fig. 22. QoE parameters of adaptive resolution YT video at 6 Mbps DSL rate, $t_{GAPthresh} = 600$ ms, $t_{notraffic} = 50$ ms and $V_{max} = 100$ pkts.

onwards. Also a T_{Delay} of 4 s is appropriate if contrary, the CPF QoE with almost always 1080p video resolution is the benchmark, at least for the DSL throughput greater than 3 Mbps. For the other throughput values, smaller T_{Delay} values are required. For example, $T_{Delay} = 2$ s for throughput 3–6 Mbps with a cost cut by almost a half at 3 Mbps and for the lower throughput towards 1 Mbps $T_{Delay} < 1$ s, with minimal cost reduction.

5.4. Analysis

After discussing the results of Youtube VoD streaming with static and adaptive video resolution it is time for an overall analysis starting with a general statement: So far, the COM results with Youtube VoD streaming **always** show a QoE benefit compared to the single path transmission or the same QoE in case DSL alone already delivers a maximum QoE, while the costly path consumption is **always** reduced compared to CPF, **in most cases significantly**. Furthermore, COM can **always** provide the same QoE level as CPF when the low-cost path throughput goes beyond 6 Mbps, while minimizing and most often eliminating CPF's always present spurious demand on the high-cost path (in the range up to 100 Mbps DSL). At the lower boundary, below 6 Mbps on the low-cost path, a fine granular tuning of the COM algorithm is required.

In the first evaluation step, an initial parameter set for COM was determined. This parameter set for COM consists of $t_{GAPthresh} = 600$ ms, $t_{notraffic} = 50$ ms and $V_{max} = 100$ pkts, and we can observe that under this conditions T_{Delay} is the parameter with the greatest impact to steer the operating point between QoE and cost. Especially when the low-cost path provides lower throughput, the intersection of cost and QoE is subject to prioritizing one of both. Towards the higher low-cost path throughput, QoE is consistent and cost is only impacted by COM as outlined above.

A T_{Delay} of 4 s seems, however, to provide over all tests a good trade-off between the costly traffic share reduction and the perceived QoE. For non-volatile multipath systems such as Hybrid Access, this value might be easily adapted to the optimal T_{Delay} for a certain DSL (low-cost path) throughput according to the presented results. Using a $t_{GAPthresh}$ of 600 ms seems to be a good choice (no results generated for other values so far). Using different CCs changes the landscape in terms of maximum efficiency, but the general statement from above is not invalidated. The differences in results can be explained by the individual calculation of the send window on the cheaper path and to what extent the send window can satisfy the burst demand. The investigated representative range of the DSL throughput from 1 to 100 Mbps shows in any case a cost optimization by COM. Even at high DSL throughput such as 50 Mbps or 100 Mbps, where one might think that VoD burst can be completely covered by DSL, COM eliminates significant spurious demand in the range of 30%–40%.

Similar tests were executed to verify COMs efficiency with video resolutions different to 1080p. It was noted that lower video resolution typically means lower throughput demand and higher video-resolution means higher throughput demand. For the use of COM, this can be reduced to a simple formula: Maximum efficiency comes with highest video resolution.

6. Conclusion

This paper addresses a real-world problem in multipath transmission networks when multipath scheduling is determined solely by the access costs (such as in today's Hybrid Access), but bursty traffic undermines this by leading to unexpected transmission costs. In particular, this applies to the VoD services, where spurious demand is measured on the expensive access without resulting in better QoE. A detailed analysis of this problem identified the peaks of the traffic bursts to cause this unnecessary overflow. This led to the definition of design goals for a robust cost-based scheduling algorithm which can address this problem. This paper presents a new algorithm called Cost Optimized Multipath (COM) which will simultaneously reduce the cost of multipath use for network operators and also retain the QoE levels required by the end-users. This is achieved by a simple change that takes into account the time gap between packets in addition to the access costs. COM is designed to be a service agnostic approach to optimize transmission cost when pure cost based multipath scheduler fails. It conforms to the specification expectation for the splitting function of the 5G ATSSS multipath framework.

Using extensive testing of Youtube service, with different TCP congestion controls, and varying DSL throughputs to cover typical Hybrid Access offerings, COM proved its performance enhancing potential in all test scenarios. With one stable COM parameter set in most test scenarios, the best QoE with significantly lower (or zero) spurious demand was achieved. Surprisingly, when the low-cost access is provided with high 100 Mbps throughput, COM is able to eliminate a 30% expensive access consumption compared to CPF. Only in scenarios where the throughput of the DSL access (low-cost path) is below 6 Mbps or 3 Mbps, a more fine-grained adjustment of the parameter set is recommended if a QoE level has to be maintained to be at the level achieved with unregulated multipath scheduling (when expensive LTE access is extensively used).

The importance of this work lies in particular in the fact that the majority of today's traffic is VoD based and tends to be bursty whenever the video playback rate is lower than the transmission capacity which becomes more likely with ATSSS. Another alternative is to avoid bursty traffic patterns. Some examples how to achieve this in a multipath context are given in Section 2 but they all fail to integrate with the specific requirement of ATSSS today which is service transparent. Another solution is to overcome the current VoD transmission principle that combines transport layer algorithms with ABR by a rate controlled ABR, as described in [92], which smoothes the VoD traffic pattern. As long as this is not widespread, there is no alternative to COM. Otherwise COM does not counteract this.

The results of this work based on tests with the largest VoD provider in the Internet suggest under specified circumstances a risk-free use of COM as a substitute for CPF even for other VoD providers beside Youtube. In the unlikely event that other VoD providers cut the videos into much smaller or larger pieces, as in the YouTube scenario tested, the optimization effect of COM will be lower if the gap size is below the threshold or the peak rates are lower or the blocking of the expensive path does not happen long enough. Whereby the confidence can actually only be increased by expanding the testbeds presented here to include more usage scenarios and completing intensive measurement campaigns, or by testing COM in scaling commercial deployments.

In the next steps the authors will consider an implementation in a commercial deployment to perform further measurements and gather evidence also regarding the implementation on other OSI layers. A new research direction which is of interest to the authors is the combination of COM with intelligent prediction of traffic throughput.

CRediT authorship contribution statement

Markus Amend: Conceptualization, Data curation, Formal analysis, Methodology, Project administration, Resources, Software, Validation, Visualization, Writing – original draft. **Veselin Rakocevic:** Conceptualization, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

[1] N. Keukeleire, B. Hesmans, O. Bonaventure, Increasing broadband reach with hybrid access networks, IEEE Commun. Stand. Mag. 4 (1) (2020) 43–49, http://dx.doi.org/10.1109/MCOMSTD.001.1900036.

[2] Q. De Coninck, M. Baerts, B. Hesmans, O. Bonaventure, Observing real smartphone applications over multipath TCP, IEEE Commun. Mag. 54 (3) (2016) 88–93, http://dx.doi.org/10.1109/MCOM.2016.7432153.

[3] M. Condoluci, S.H. Johnson, V. Ayadurai, M.A. Lema, M.A. Cuevas, M. Dohler, T. Mahmoodi, Fixed-mobile convergence in the 5G era: From hybrid access to converged core, IEEE Netw. 33 (2) (2019) 138–145, http://dx.doi.org/10.1109/MNET.2018.1700462.

[4] J. Lee, Y. Yi, S. Chong, Y. Jin, Economics of WiFi offloading: Trading delay for cellular capacity, IEEE Trans. Wireless Commun. 13 (3) (2014) 1540–1554, http://dx.doi.org/10.1109/TWC.2014.010214.130949.

[5] Y. He, M. Chen, B. Ge, M. Guizani, On WiFi offloading in heterogeneous networks: Various incentives and trade-off strategies, IEEE Commun. Surv. Tutor. 18 (4) (2016) 2345–2385, http://dx.doi.org/10.1109/COMST.2016.2558191.

[6] L. Kim, 5G economics – The numbers, 2017, https://techneconomyblog.com/2017/07/.

[7] L. Kim, Fixed wireless access in a modern 5G setting – What does it bring that we don't already have? 2023, https://techneconomyblog.com/2023/01/.

[8] N. Leymann, Hybrid Access Deployment @ DT, Tech. Rep., Internet Engineering Task Force, 2016, URL https://datatracker.ietf.org/meeting/97/materials/slides-97-banana-hybrid-access-deployment-at-deutsche-telekom-01.pdf.

[9] N. Leymann, C. Heidemann, M. Zhang, B. Sarikaya, M. Cullen, Huawei's GRE tunnel bonding protocol, 2017, http://dx.doi.org/10.17487/RFC8157, RFC 8157, URL https://www.rfc-editor.org/info/rfc8157.

[10] A. Ford, C. Raiciu, M.J. Handley, O. Bonaventure, C. Paasch, TCP extensions for multipath operation with multiple addresses, 2020, http://dx.doi.org/10.17487/RFC8684, RFC 8684, URL https://www.rfc-editor.org/info/rfc8684.

[11] 3GPP, System architecture for the 5G system (5GS), Technical Specification (TS) 23.501, 3rd Generation Partnership Project (3GPP), 2021, version 16.11.0.

[12] BBF, 5G Wireless Wireline Convergence Architecture, Technical Report (TR) 470, Broadband Forum, 2020, issue 1.

[13] M. Amend, A. Brunstrom, A. Kassler, V. Rakocevic, S. Johnson, DCCP extensions for multipath operation with multiple addresses, in: Internet-Draft draft-ietf-tsvwg-multipath-dccp-13, Internet Engineering Task Force, 2024 in preparation, URL https://datatracker.ietf.org/doc/draft-ietf-tsvwg-multipath-dccp/13/.

[14] Y. Liu, Y. Ma, Q.D. Coninck, O. Bonaventure, C. Huitema, M. Kühlewind, Multipath extension for QUIC, in: Internet-Draft Draft-ietf-quic-multipath-06, Internet Engineering Task Force, 2023 in preparation, URL https://datatracker.ietf.org/doc/draft-ietf-quic-multipath/06/.

[15] 3GPP, Study on Access Traffic Steering, Switching and Splitting Support in the 5G System Architecture; Phase 3, Technical Report (TR) 23.700-53, 3rd Generation Partnership Project (3GPP), 2022, version 0.2.0.

[16] O. Bonaventure, M. Piraux, Q.D. Coninck, M. Baerts, C. Paasch, M. Amend, Multipath schedulers, in: Internet-Draft Draft-bonaventure-iccrg-schedulers-02, Internet Engineering Task Force, 2021 in preparation, URL https://datatracker.ietf.org/doc/draft-bonaventure-iccrg-schedulers/02/.

[17] Cisco, Cisco Visual Networking Index: Forecast and Methodology, 2016–2021, Tech. Rep., Cisco, 2017.

[18] Sandvine, The Mobile Internet Phenomena Report, 2021, Tech. Rep., Sandvine, 2021.

[19] Ericsson, Ericsson Mobility Report 2022, Tech. Rep., Ericsson, 2022.

[20] IEEE Standard for Local and Metropolitan Area Networks–Link Aggregation, IEEE Std 802.1AX-2008, 2008, http://dx.doi.org/10.1109/IEEESTD.2008.4668665.

[21] D. Carr, T. Coradetti, B. Lloyd, G. McGregor, K.L. Sklower, The PPP multilink protocol (MP), 1996, http://dx.doi.org/10.17487/RFC1990, RFC 1990, URL https://www.rfc-editor.org/info/rfc1990.

[22] 3GPP, Evolved universal terrestrial radio access (E-UTRA) and evolved universal terrestrial radio access network (E-UTRAN); Overall description; Stage 2, in: Technical Specification (TS) 36.300, 3rd Generation Partnership Project (3GPP, 2016, version 13.2.0.

[23] R. Atkinson, S. Bhatti, Identifier-locator network protocol (ILNP) architectural description, 2012, http://dx.doi.org/10.17487/RFC6740, RFC 6740, URL https://www.rfc-editor.org/info/rfc6740.

[24] T. Herbert, P. Lapukhov, Identifier-locator addressing for IPv6, in: Internet-Draft Draft-herbert-intarea-ila-01, Internet Engineering Task Force, 2018 in preparation, URL https://datatracker.ietf.org/doc/draft-herbert-intarea-ila/01/.

[25] E. Nordmark, M. Bagnulo, Shim6: Level 3 multihoming shim protocol for IPv6, 2009, http://dx.doi.org/10.17487/RFC5533, RFC 5533, URL https://www.rfc-editor.org/info/rfc5533.

[26] K. Chowdhury, K. Leung, B. Patil, V. Devarapalli, S. Gundavelli, Proxy mobile IPv6, 2008, http://dx.doi.org/10.17487/RFC5213, RFC 5213, URL https://www.rfc-editor.org/info/rfc5213.

[27] M. Menth, A. Stockmayer, M. Schmidt, LISP hybrid access, in: Internet-Draft Draft-menth-lisp-ha-00, Internet Engineering Task Force, 2015 in preparation, URL https://datatracker.ietf.org/doc/draft-menth-lisp-ha/00/.

[28] S. Pierrel, P. Jokela, J. Melén, K. Slavov, A policy system for simultaneous multiaccess with host identity protocol, in: Proc. ACNM, 2007, pp. 71–77.

[29] A. Gurtov, T. Polishchuk, Secure multipath transport for legacy internet applications, in: 2009 Sixth International Conference on Broadband Communications, Networks, and Systems, 2009, pp. 1–8, http://dx.doi.org/10.4108/ICST.BROADNETS2009.7186.

[30] S. Sevilla, J.J. Garcia-Luna-Aceves, A deployable identifier-locator split architecture, in: 2017 IFIP Networking Conference (IFIP Networking) and Workshops, 2017, pp. 1–9, http://dx.doi.org/10.23919/IFIPNetworking.2017.8264833.

[31] M. Arye, E. Nordström, R. Kiefer, J. Rexford, M.J. Freedman, A formally-verified migration protocol for mobile, multi-homed hosts, in: 2012 20th IEEE International Conference on Network Protocols, ICNP, 2012, pp. 1–12, http://dx.doi.org/10.1109/ICNP.2012.6459961.

[32] K. Evensen, D. Kaspar, P. Engelstad, A.F. Hansen, C. Griwodz, P. Halvorsen, A network-layer proxy for bandwidth aggregation and reduction of IP packet reordering, in: 2009 IEEE 34th Conference on Local Computer Networks, 2009, pp. 585–592, http://dx.doi.org/10.1109/LCN.2009.5355198.

[33] M. Bednarek, G.B. Kobas, M. Kühlewind, B. Trammell, Multipath bonding at layer 3, in: Proceedings of the 2016 Applied Networking Research Workshop, ANRW '16, ACM, New York, NY, USA, 2016, pp. 7–12, http://dx.doi.org/10.1145/2959424.2959439, URL http://doi.acm.org/10.1145/2959424.2959439.

[34] B. Almsi, G. Lencse, S. Szilgyi, Investigating the multipath extension of the GRE in UDP technology, Comput. Commun. 103 (C) (2017) 29–38, http://dx.doi.org/10.1016/j.comcom.2017.02.002.

[35] G. Lencse, S. Szilagyi, F. Fejes, M. Georgescu, MPT network layer multipath library, in: Internet-Draft Draft-lencse-tsvwg-mpt-10, Internet Engineering Task Force, 2022 in preparation, URL https://datatracker.ietf.org/doc/draft-lencse-tsvwg-mpt/10/.

[36] J. Iyengar, P. Amer, R. Stewart, Concurrent multipath transfer using SCTP multihoming over independent end-to-end paths, IEEE/ACM Trans. Netw. 14 (5) (2006) 951–964, http://dx.doi.org/10.1109/TNET.2006.882843.

[37] P.P.D. Amer, M. Becke, T. Dreibholz, N. Ekiz, J. Iyengar, P. Natarajan, R.R. Stewart, M. Tüxen, Load sharing for the stream control transmission protocol (SCTP), in: Internet-Draft Draft-tuexen-tsvwg-sctp-multipath-26, Internet Engineering Task Force, 2024 in preparation, URL https://datatracker.ietf.org/doc/draft-tuexen-tsvwg-sctp-multipath/26/.

[38] K.c. Lan, C.Y. Li, Improving TCP performance over an on-board multi-homed network, in: 2012 IEEE Wireless Communications and Networking Conference, WCNC, 2012, pp. 2961–2966, http://dx.doi.org/10.1109/WCNC.2012.6214311.

[39] D. Wischik, M. Handley, M.B. Braun, The resource pooling principle, SIGCOMM Comput. Commun. Rev. 38 (5) (2008) 47–52, http://dx.doi.org/10.1145/1452335.1452342.

[40] M. Amend, E. Bogenfeld, M. Cvjetkovic, V. Rakocevic, M. Pieska, A. Kassler, A. Brunstrom, A framework for multiaccess support for unreliable internet traffic using multipath DCCP, in: 2019 IEEE 44th Conference on Local Computer Networks, LCN, 2019, pp. 316–323, http://dx.doi.org/10.1109/LCN44214.2019.8990746.

[41] Q. De Coninck, O. Bonaventure, Multipath QUIC: Design and evaluation, in: Proceedings of the 13th International Conference on Emerging Networking EXperiments and Technologies, CoNEXT '17, Association for Computing Machinery, New York, NY, USA, 2017, pp. 160–166, http://dx.doi.org/10.1145/3143361.3143370.

[42] V. Singh, S. Ahsan, J. Ott, MPRTP: Multipath considerations for real-time media, in: Proceedings of the 4th ACM Multimedia Systems Conference, MMSys '13, Association for Computing Machinery, New York, NY, USA, 2013, pp. 190–201, http://dx.doi.org/10.1145/2483977.2484002.

[43] V. Singh, T. Karkkainen, J. Ott, S. Ahsan, L. Eggert, Multipath RTP (MPRTP), in: Internet-Draft Draft-ietf-avtcore-mprtp-03, Internet Engineering Task Force, 2016 in preparation, URL https://datatracker.ietf.org/doc/draft-ietf-avtcore-mprtp/03/.

[44] J. Kim, Y.-C. Chen, R. Khalili, D. Towsley, A. Feldmann, Multi-source multipath HTTP (MHTTP): A proposal, SIGMETRICS Perform. Eval. Rev. 42 (1) (2014) 583–584, http://dx.doi.org/10.1145/2637364.2592029.

[45] H. Wu, S. Ferlin, G. Caso, O. Alay, A. Brunstrom, A survey on multipath transport protocols towards 5G access traffic steering, switching and splitting, IEEE Access 9 (2021) 164417–164439, http://dx.doi.org/10.1109/ACCESS.2021.3134261.

[46] M. Amend, E. Bogenfeld, A. Brunstrom, A. Kassler, V. Rakocevic, A multipath framework for UDP traffic over heterogeneous access networks, in: Internet-Draft Draft-amend-tsvwg-multipath-framework-mpdccp-01, Internet Engineering Task Force, 2019 in preparation, URL https://datatracker.ietf.org/doc/draft-amend-tsvwg-multipath-framework-mpdccp/01/.

[47] M. Xu, Y. Cao, E. Dong, Delay-based congestion control for MPTCP, in: Internet-Draft Draft-xu-mptcp-congestion-control-05, Internet Engineering Task Force, 2017 in preparation, URL https://datatracker.ietf.org/doc/draft-xu-mptcp-congestion-control/05/.

[48] C. Raiciu, M.J. Handley, D. Wischik, Coupled congestion control for multipath transport protocols, 2011, http://dx.doi.org/10.17487/RFC6356, RFC 6356, URL https://www.rfc-editor.org/info/rfc6356.

[49] R. Khalili, N. Gast, M. Popovic, J.-Y.L. Boudec, Opportunistic linked-increases congestion control algorithm for MPTCP, in: Internet-Draft Draft-khalili-mptcp-congestion-control-05, Internet Engineering Task Force, 2014 in preparation, URL https://datatracker.ietf.org/doc/draft-khalili-mptcp-congestion-control/05/.

[50] A. Walid, Q. Peng, J. Hwang, S.H. Low, Balanced linked adaptation congestion control algorithm for MPTCP, in: Internet-Draft Draft-walid-mptcp-congestion-control-04, Internet Engineering Task Force, 2016 in preparation, URL https://datatracker.ietf.org/doc/draft-walid-mptcp-congestion-control/04/.

[51] S. Hemminger, [TCP]: make cubic the default, URL https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git/commit/.

[52] Apple, xnu-2782.1.97. URL https://github.com/apple-oss-distributions/xnu/blob/main/bsd/netinet/tcp_cc.h.

[53] P. Balasubramanian, Updates on windows TCP, 2017, URL https://datatracker.ietf.org/meeting/100/materials/slides-100-tcpm-updates-on-windows-tcp.

[54] N. Cardwell, Y. Cheng, C.S. Gunn, S.H. Yeganeh, V. Jacobson, BBR: Congestion-based congestion control, Commun. ACM 60 (2) (2017) 58–66, http://dx.doi.org/10.1145/3009824.

[55] N. Cardwell, Y. Cheng, C.S. Gunn, S.H. Yeganeh, S. Ian, I. Jana, V. Victor, J. Priyaranjan, S. Yousuk, V. Jacobson, BBR congestion control work at google IETF 101 update, 2018, URL https://datatracker.ietf.org/meeting/101/materials/slides-101-iccrg-an-update-on-bbr-work-at-google-00.

[56] M. Yanev, S. McQuistin, C. Perkins, Does TCP new congestion window validation improve HTTP adaptive streaming performance? in: Proceedings of the 32nd Workshop on Network and Operating Systems Support for Digital Audio and Video, NOSSDAV '22, Association for Computing Machinery, New York, NY, USA, 2022, pp. 29–35, http://dx.doi.org/10.1145/3534088.3534347.

[57] S.R. Pokhrel, A. Walid, Learning to harness bandwidth with multipath congestion control and scheduling, IEEE Trans. Mob. Comput. 22 (2) (2023) 996–1009, http://dx.doi.org/10.1109/TMC.2021.3085598.

[58] C. Paasch, S. Ferlin, O. Alay, O. Bonaventure, Experimental evaluation of multipath TCP schedulers, in: Proceedings of the 2014 ACM SIGCOMM Workshop on Capacity Sharing Workshop, CSWS '14, Association for Computing Machinery, New York, NY, USA, 2014, pp. 27–32, http://dx.doi.org/10.1145/2630088.2631977.

[59] T. De Schepper, J. Struye, E. Zeljković, S. Latré, J. Famaey, Software-defined multipath-TCP for smart mobile devices, in: 2017 13th International Conference on Network and Service Management, CNSM, 2017, pp. 1–6, http://dx.doi.org/10.23919/CNSM.2017.8256043.

[60] K.W. Choi, Y.S. Cho, Aneta, J.W. Lee, S.M. Cho, J. Choi, Optimal load balancing scheduler for MPTCP-based bandwidth aggregation in heterogeneous wireless environments, Comput. Commun. 112 (2017) 116–130, http://dx.doi.org/10.1016/j.comcom.2017.08.018, URL https://www.sciencedirect.com/science/article/pii/S0140366417302426.

[61] Y.-s. Lim, E.M. Nahum, D. Towsley, R.J. Gibbens, ECF: An MPTCP path scheduler to manage heterogeneous paths, in: Proceedings of the 13th International Conference on Emerging Networking EXperiments and Technologies, CoNEXT '17, Association for Computing Machinery, New York, NY, USA, 2017, pp. 147–159, http://dx.doi.org/10.1145/3143361.3143376.

[62] N. Kuhn, E. Lochin, A. Mifdaoui, G. Sarwar, O. Mehani, R. Boreli, DAPS: Intelligent delay-aware packet scheduling for multipath transport, in: 2014 IEEE International Conference on Communications, ICC, 2014, pp. 1222–1227, http://dx.doi.org/10.1109/ICC.2014.6883488.

[63] S. Ferlin, Ö. Alay, O. Mehani, R. Boreli, BLEST: Blocking estimation-based MPTCP scheduler for heterogeneous networks, in: 2016 IFIP Networking Conference (IFIP Networking) and Workshops, 2016, pp. 431–439, http://dx.doi.org/10.1109/IFIPNetworking.2016.7497206.

[64] F. Yang, Q. Wang, P.D. Amer, Out-of-order transmission for in-order arrival scheduling for multipath TCP, in: 2014 28th International Conference on Advanced Information Networking and Applications Workshops, 2014, pp. 749–752, http://dx.doi.org/10.1109/WAINA.2014.122.

[65] P. Hurtig, K.-J. Grinnemo, A. Brunstrom, S. Ferlin, Ö. Alay, N. Kuhn, Low-latency scheduling in MPTCP, IEEE/ACM Trans. Netw. 27 (1) (2019) 302–315, http://dx.doi.org/10.1109/TNET.2018.2884791.

[66] B.Y.L. Kimura, D.C.S.F. Lima, A.A.F. Loureiro, Packet scheduling in multipath TCP: Fundamentals, lessons, and opportunities, IEEE Syst. J. 15 (1) (2021) 1445–1457, http://dx.doi.org/10.1109/JSYST.2020.2965471.

[67] H. Wu, O. Alay, A. Brunstrom, S. Ferlin, G. Caso, Peekaboo: Learning-based multipath scheduling for dynamic heterogeneous environments, IEEE J. Sel. Areas Commun. 38 (10) (2020) 2295–2310, http://dx.doi.org/10.1109/JSAC.2020.3000365.

[68] A. Frömmgen, Programming Models and Extensive Evaluation Support for MPTCP Scheduling, Adaptation Decisions, and DASH Video Streaming (Ph.D. thesis), Technische Universität, Darmstadt, 2018, URL http://tuprints.ulb.tu-darmstadt.de/7709/.

[69] W. Lu, D. Yu, M. Huang, B. Guo, PO-MPTCP: Priorities-oriented data scheduler for multimedia multipathing services, Int. J. Digit. Multimed. Broadcast. 2018 (2018) 1413026, http://dx.doi.org/10.1155/2018/1413026.

[70] D. Jurca, P. Frossard, Video packet selection and scheduling for multipath streaming, IEEE Trans. Multimed. 9 (3) (2007) 629–641, http://dx.doi.org/10.1109/TMM.2006.888017.

[71] D. Jurca, P. Frossard, Distortion optimized multipath video streaming, in: Proceedings of the International Packet Video Workshop, 2004, URL http://infoscience.epfl.ch/record/87129.

[72] X. Corbillon, R. Aparicio-Pardo, N. Kuhn, G. Texier, G. Simon, Cross-layer scheduler for video streaming over MPTCP, in: Proceedings of the 7th International Conference on Multimedia Systems, MMSys '16, Association for Computing Machinery, New York, NY, USA, 2016, pp. 1–12, http://dx.doi.org/10.1145/2910017.2910594.

[73] Y.P.G. Chowrikoppalu, Multipath Adaptive Video Streaming over Multipath TCP (Master's thesis), University of Saarland, 2013, URL https://www.nt.uni-saarland.de/wp-content/uploads/2019/05/MasterThesisYash.pdf.

[74] J. Apostolopoulos, T. Wong, W. tian Tan, S. Wee, On multiple description streaming with content delivery networks, in: Proceedings.Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies, 2002, pp. 1736–1745, http://dx.doi.org/10.1109/INFCOM.2002.1019427.

[75] S. Mao, S. Lin, S. Panwar, Y. Wang, E. Celebi, Video transport over ad hoc networks: multistream coding with multipath transport, IEEE J. Sel. Areas Commun. 21 (10) (2003) 1721–1737, http://dx.doi.org/10.1109/JSAC.2003.815965.

[76] G. Xie, M.N.S. Swamy, M.O. Ahmad, Optimal packet scheduling for multi-description multi-path video streaming over wireless networks, in: 2007 IEEE International Conference on Communications, 2007, pp. 1618–1623, http://dx.doi.org/10.1109/ICC.2007.271.

[77] R. Matsufuji, D. Cavendish, K. Kumazoe, D. Nobayashi, T. Ikenaga, Multipath TCP path schedulers for streaming video, in: 2017 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, PACRIM, 2017, pp. 1–6, http://dx.doi.org/10.1109/PACRIM.2017.8121920.

[78] S. Maheshwari, P. Lundrigan, S.K. Kasera, Scheduling virtual wifi interfaces for high bandwidth video upstreaming using multipath TCP, in: Proceedings of the 20th International Conference on Distributed Computing and Networking, ICDCN '19, Association for Computing Machinery, New York, NY, USA, 2019, pp. 1–10, http://dx.doi.org/10.1145/3288599.3288620.

[79] M. Amend, V. Rakocevic, J. Habermann, Cost optimized multipath scheduling in 5G for video-on-demand traffic, in: 2021 IEEE Wireless Communications and Networking Conference, WCNC, 2021, pp. 1–6, http://dx.doi.org/10.1109/WCNC49053.2021.9417415.

[80] M. Pieska, A. Rabitsch, A. Brunstrom, A. Kassler, M. Amend, Adaptive cheapest path first scheduling in a transport-layer multi-path tunnel context, in: Proceedings of the Applied Networking Research Workshop, ANRW '21, Association for Computing Machinery, New York, NY, USA, 2021, pp. 39–45, http://dx.doi.org/10.1145/3472305.3472316.

[81] B. Han, F. Qian, L. Ji, V. Gopalakrishnan, MP-DASH: Adaptive video streaming over preference-aware multipath, in: Proceedings of the 12th International on Conference on Emerging Networking EXperiments and Technologies, CoNEXT '16, Association for Computing Machinery, New York, NY, USA, 2016, pp. 129–143, http://dx.doi.org/10.1145/2999572.2999606.

[82] A. Elgabli, K. Liu, V. Aggarwal, Optimized preference-aware multi-path video streaming with scalable video coding, IEEE Trans. Mob. Comput. 19 (1) (2020) 159–172, http://dx.doi.org/10.1109/TMC.2018.2889039.

[83] S. Afzal, V. Testoni, C.E. Rothenberg, P. Kolan, I. Bouazizi, A holistic survey of wireless multipath video streaming, 2019, http://dx.doi.org/10.48550/ARXIV.1906.06184, URL https://arxiv.org/abs/1906.06184.

[84] F. Zhou, T. Dreibholz, X. Zhou, F. Fu, Y. Tan, Q. Gan, The performance impact of buffer sizes for multi-path TCP in internet setups, in: 2017 IEEE 31st International Conference on Advanced Information Networking and Applications, AINA, 2017, pp. 9–16, http://dx.doi.org/10.1109/AINA.2017.26.

[85] 3GPP, Wireless and wireline convergence access support for the 5G system (5GS), in: Technical Specification (TS) 23.316, 3rd Generation Partnership Project (3GPP), 2022, version 16.8.0.

[86] K. Yamagishi, QoE-estimation models for video streaming services, in: 2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA ASC, 2017, pp. 357–363, http://dx.doi.org/10.1109/APSIPA.2017.8282058.

[87] R.R. Ramachandra Rao, S. Göring, P. Vogel, N. Pachatz, J.J.V. Villarreal, W. Robitza, P. List, B. Feiten, A. Raake, Adaptive video streaming with current codecs and formats: Extensions to parametric video quality model ITU-T P.1203, Electron. Imaging 2019 (10) (2019) 314-1–314-7, http://dx.doi.org/10.2352/ISSN.2470-1173.2019.10.IQSP-314, URL https://www.ingentaconnect.com/content/ist/ei/2019/00002019/00000010/art00015.

[88] S. Göring, A. Raake, B. Feiten, A framework for QoE analysis of encrypted video streams, in: 2017 Ninth International Conference on Quality of Multimedia Experience, QoMEX, 2017, pp. 1–3, http://dx.doi.org/10.1109/QoMEX.2017.7965640.

[89] W. Robitza, S. Göring, A. Raake, D. Lindegren, G. Heikkilä, J. Gustafsson, P. List, B. Feiten, U. Wüstenhagen, M.-N. Garcia, K. Yamagishi, S. Broom, HTTP adaptive streaming QoE estimation with ITU-T Rec. P. 1203: Open databases and software, in: Proceedings of the 9th ACM Multimedia Systems Conference, MMSys '18, Association for Computing Machinery, New York, NY, USA, 2018, pp. 466–471, http://dx.doi.org/10.1145/3204949.3208124.

[90] G. Miranda, D.F. Macedo, J.M. Marquez-Barja, Estimating video on demand QoE from network QoS through ICMP probes, IEEE Trans. Netw. Serv. Manag. (2021) 1, http://dx.doi.org/10.1109/TNSM.2021.3129610.

[91] ITU-T, Parametric bitstream-based quality assessment of progressive download and adaptive audiovisual streaming services over reliable transport, in: Recommendation P.1203, International Telecommunication Union, Geneva, 2017.

[92] B. Spang, S. Kunamalla, R. Teixeira, T.-Y. Huang, G. Armitage, R. Johari, N. McKeown, Sammy: Smoothing video traffic to be a friendly internet neighbor, in: Proceedings of the ACM SIGCOMM 2023 Conference, ACM SIGCOMM '23, Association for Computing Machinery, New York, NY, USA, 2023, pp. 754–768, http://dx.doi.org/10.1145/3603269.3604839.

Markus Amend has a Diploma Degree in Informations and Telecommunications engineering from Technische Hochschule Mittelhessen. Since 2013, he is a researcher and strategist at Deutsche Telekom and heads the Core & Network Services unit of Group Technology since 2022. He is active in standardization, open source development, prototyping and especially algorithms/protocols for multi-connectivity solutions. In the area of multipath network protocols he is experienced with 3GPP ATSSS, MPTCP and is the main driver of MP-DCCP.

Dr Veselin Rakocevic is Associate Professor of Networks and Communication Systems at City, University of London, where he has been working since 2002. He has been active in research in wireless networking for over 20 years, publishing widely in journals and conferences. He has been involved in a number of national and European research projects and worked with industrial partners including Deutsche Telekom, BT, Fujitsu, and others. He holds a PhD degree from the University of London and a Dipl-Ing degree in Electrical and Electronic Engineering from the University of Belgrade, Serbia.