Mobile CDN Enhancements for QoE-Improved Content Delivery in Mobile Operator Networks

**Faqir Zarrar Yousaf, Marco Liebsch, Andreas Maeder, and Stefan Schmid,
NEC Laboratories Europe**

Abstract

With the popularity of YouTube and other video services (e.g., VoD), video content is becoming a dominant traffic type in mobile networks and will soon form the major bulk of the traffic. This poses a serious challenge to mobile network operators when it comes to delivering video content to multiple users in a wireless environment, where the quality of experience of each user must be met as per user expectation. To this end, mobile content distribution networks (mCDNs) are gaining attraction as one possibility to enable controllable and resource-efficient delivery of video content. In this article we motivate the deployment of mCDN serving point (mCSP) nodes in an mCDN infrastructure and describe its benefits for a mobile network operator who wishes to employ an mCDN system in their existing mobile network architecture. In this respect we highlight the key features and requirements of such an mCSP node and the advantages it offers. One particularly beneficial feature of our proposed mCSP node is to ensure fairness during the delivery of progressive video streaming services employed by the likes of YouTube, Daily Motion, and others. In this regard we first highlight the unfairness issue when delivering progressive video streaming services over TCP to multiple users over a wireless network infrastructure and its effect on the user perceived QoE. We then demonstrate, as a proof of concept, the effectiveness of employing application-level scheduling in an mCSP node to ensure fairness among multiple simultaneous progressive video sessions in scenarios where the backhaul link in a mobile network infrastructure may become congested.

With the deployment and advances in broadband wireless access technologies and the proliferation of smart phones and tablets, the popularity of accessing bandwidth-intensive multimedia content on mobile devices is rapidly growing and expected to increase significantly in the coming years. It is forecasted that a large portion of multimedia content will be composed of pre-recorded videos such as movies, short videos (similar to YouTube videos), and TV shows, and mobile video is forecasted to generate over 70 percent of mobile data traffic by 2016 [1].

This poses a serious challenge to existing mobile network infrastructures with constrained resources in the wireless access, backhaul, and core networks. Typically, mobile network operators (MNOs) dealt with such increased traffic forecasts by extending/upgrading the overall network capacity accordingly. However, this becomes more and more difficult to implement due to increased capital/operational expenditure with a low return on investment (ROI).

A major reason for low ROI is the fact that mobile operator networks typically serve as mere bit pipes, and the existing charging models do not scale proportionately to such high-volume traffic generated between mobile users and third-party content providers (CPs) like YouTube, Netflix, and others. In order to increase the revenue, MNOs need to add value to their existing services. In this respect the MNOs are becoming increasingly interested in setting up their own content distribution network (CDN) infrastructures, tightly coupled with their existing transport infrastructure. Such MNO-owned CDNs, referred to as mobile CDNs (mCDNs), will enable them to get a share of the end-to-end content delivery revenue of third-party CPs and to provide premium content delivery services themselves to their customers.

Besides increased revenues, mCDNs will enable MNOs to have better control over the content distribution service for mobile multimedia content with fast, efficient, and fair delivery of both managed content (MC) (i.e., content hosted and served on local caches) as well as over-the-top (OTT) content (i.e., content that is simply cached by the mCDN), especially video traffic.

In this article we motivate the deployment of an mCDN system in 3GPP networks by providing an overview of the functional requirement of an mCDN serving point (mCSP) entity, which can be an integral part of such a system. We focus on applications that use TCP/HTTP to progressively deliver video content to mobile devices, and refer to the elastic characteristics of TCP to adapt to congestion and packet loss situations.

Since TCP congestion control is not aware of the status of a progressive video download on mobile devices, we propose
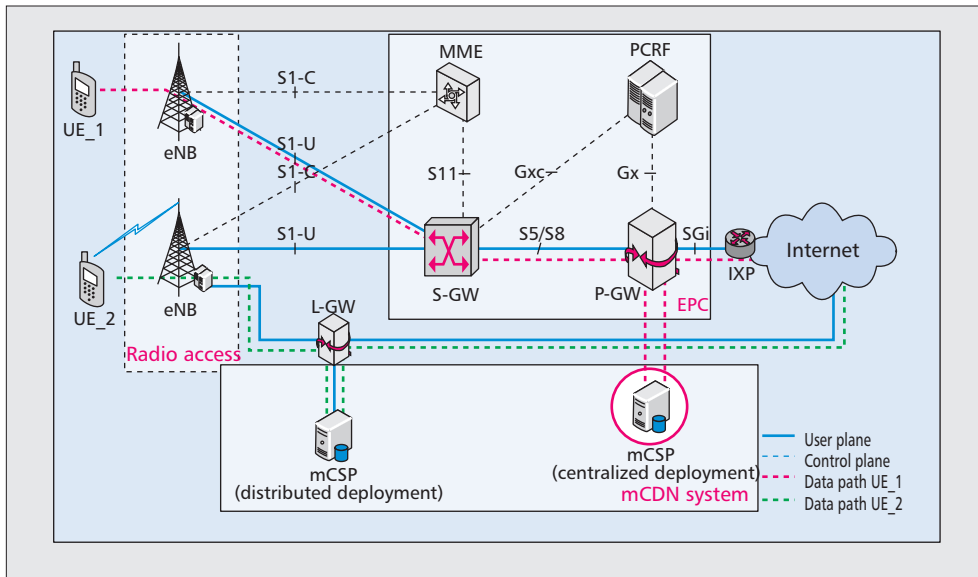
Figure 1. *High-level EPC architecture with mobile CDN serving point.*

to leverage TCP endpoint functionality on the mCSP to enable controlled scheduling of content chunks at the application layer to enable fair delivery of content to mobile devices. The approach to perform per-user chunk scheduling at the application level provides an easy-to-implement approach to throttle or privilege particular chunks for transmission. Such approach is independent of and complements on-the-path traffic engineering performed by routers of the transport network.

The proposed application-layer scheduler on the mCSP takes advantage of TCP feedback information, such as segment and congestion window size, as well as of data plane context information, such as the video encoding characteristics of the delivered content. The fairness aspect is met by performing scheduling based on locally derived download and playback status information to avoid depletion of the client application's playout buffer.

Related Work

In the literature, a significant number of studies are investigating the effectiveness and design of "general" large-scale CDNs, which are often deployed to increase the reliability and quality of service (QoS) of specific services. For a comprehensive overview of related work to CDNs in general, refer to [2, references therein]. In contrast, the literature on CDNs in mobile networks as a specialized subset is less rich, at least from an architectural point of view. The architecture proposed by the MEDIEVAL project [3] includes CDNs as one measure to increase user quality of experience (QoE) for video services, by integrating the CDN with mobility management and a cross-layer optimization entity, which uses transcoding and scalable video codecs. In this respect, it is different from the approach proposed in this article, which focuses on application-layer scheduling.

An early work on CDNs in mobile networks was published in [4] and proposes a mobile streaming media CDN (MSM-CDN) as a network overlay to the mobile infrastructure for video delivery over a time-varying channel. Mobile CDN as considered in this article does not work as an overlay but benefits from integrating well and interfacing with the mobile network infrastructure.

In [5], it is proposed to complement a General Packet Radio Service (GPRS) network with two caching proxies for streaming video, one placed at the mobility anchor's level in

the topology, another placed in the radio access network. Both caches are connected via a direct IP link to bypass the mobility tunnel for improved content delivery.

Further studies focus on cross-layer, application-aware, or QoE-aware content delivery, such as the video pacing scheme proposed in [6] or the mean opinion score (MOS)-based network resource allocation framework operating on the MAC layer, as proposed in [7]. Another approach described in [8] proposes an adaptive streaming algorithm for 3GPP networks to improve QoS in varying network conditions by means of changing the media rate. Such adaptation depends on the available codec profiles and associated coding bit rates. Scheduling as considered in this article is independent of coding rate adaptation, but leverages the opportunity to control scheduling of content chunks on the transparent cache during progressive download. Nevertheless, the proposed framework for mCDN application-layer scheduling could be modified or extended to include these approaches if deemed necessary by MNOs.

Overview of Mobile CDN Deployment in the Third Generation Partnership Project

An mCDN can be differentiated from traditional third-party CDNs in two aspects:
- The CDN components are administered by the same authority as the mobile operator network; hence, trust between mobile infrastructure components and the CDN is implicit.
- The CDN can be implemented as an integral part of the mobile network infrastructure, which potentially enables locating the content source closer to the end user.

The latter is achieved by decentralizing mCSPs, which serve as points of attachment from an end user's data plane point of view. Such mCSPs can be distributed throughout the core, backhaul, or even radio access networks, whereby the mobile operator must ensure efficient selection and routing of the end user's content request to the most appropriate mCSP. "Most appropriate" means that the selected mCSP must be able to serve the user's request and located close to the user's location to maintain a short content delivery path.

Figure 1 depicts an abstract view of a mobile operator network according to the Evolved Packet System (EPS) architecture [9]. The packet data network gateway (P-GW) represents a topological anchor for mobile devices' IP addresses. Since
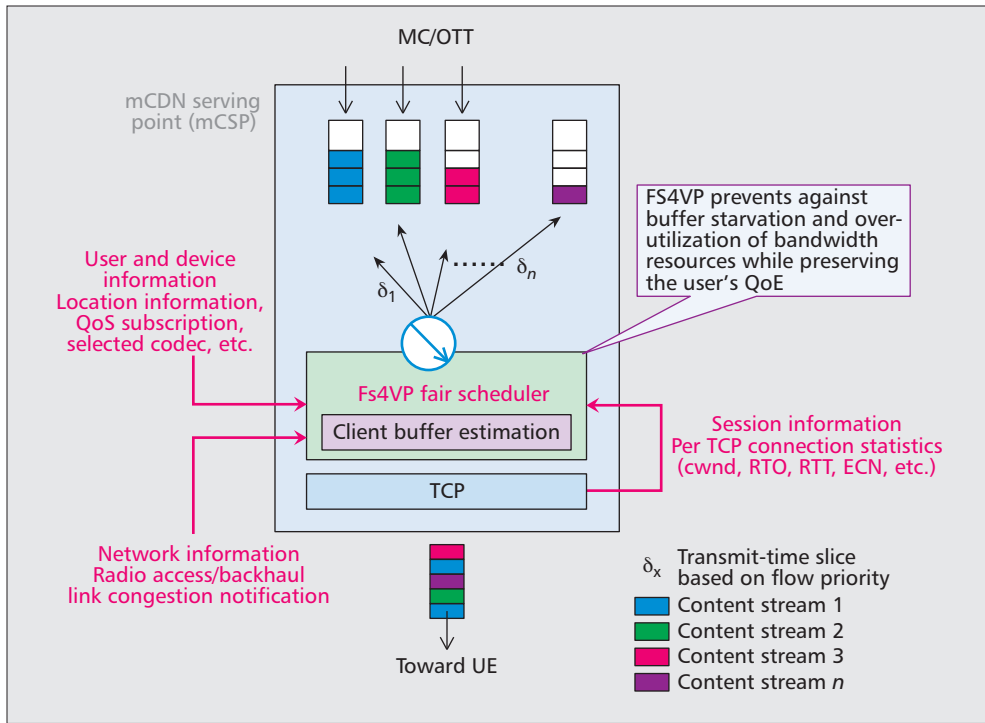
Figure 2. *Conceptual overview of the FS4VP fair scheduler.*

all data traffic from and toward a mobile device must traverse the device's P-GW for a certain access point name (APN), these gateways are currently implemented in a centralized manner and located close to the mobile operator's established Internet exchange peering points (IXPs), which provide the interface to the public Internet. The drawback of such centralization is that all data traffic traverses the mobile infrastructure's core network. Serving gateways (S-GWs) and P-GWs enable mobility support by forwarding packets to/from the mobile device's current location, whereas the mobility management entity (MME) represents the mobility management control plane function that establishes mobility states in the gateways according to the mobile device's connectivity state (active or idle) and location. Radio base stations (evolved NodeB, eNB) provide high data-rate wireless access to mobile devices. The policy and charging rules function (PCRF) establishes QoS, taking user subscription data into account.

Standardization efforts are currently also underway to further evolve the packet core architecture to address the core network capacity problem and enable traffic offload at distributed local gateways (L-GWs). Such L-GWs provide similar functionality as a P-GW and allow a mobile device's traffic to be flexibly routed to decentralized mCSPs in the access or backhaul network without the need to traverse a centralized P-GW. Since an L-GW represents a topological anchor for the mobile device's IP address for offloaded traffic, optimally the mCSP, which serves a user's content request, is located close to the mobile device's L-GW. By means of such a setup, content delivery paths and associated transport costs can be kept low.

Progressive download of (video) content is typically achieved via HTTP over TCP. For content delivery from a CDN, the mCSP hereby functions as a split-TCP entity and represents the infrastructure-side endpoint of the user's TCP connection. The mCSP may use another TCP connection to the original CP in the Internet to retrieve requested content that is not available on a local cache, or use either TCP or alternative protocols to fetch requested content from local storage. In order to serve the user's request, the mCSP must

implement the control plane, through which the application (e.g., a web browser) on the user's mobile device requests the content. To support web-based content requests, the mCSP implements an HTTP proxy. Other content delivery protocols, such as Real Time Streaming Protocol (RTSP) or Real Time Multimedia Protocol (RTMP), may also be supported.

With these features and due to its tight coupling with the operator network, the mCSP entity becomes a natural choice for implementing control strategies that will ensure the necessary service quality when progressively delivering video content to multiple users. This implies an implementation of a scheduling function within the mCSP architecture. Compared to existing caching solutions, the mCSPs' scheduling algorithms gain advantage in having access to mobile-subscriber-, mobile-device-, location-, as well as connectivity-specific information through the mobile operator's EPS control plane. This enables the mCSP to tailor the delivery of content chunks per request and coordinate the delivery between multiple requests, aiming at a balance between the overall capacity demands and satisfying all users' expectations on the content delivery service.

Functional Architecture for Scheduling on mCDN Serving Points

The scheduling function of the mCSP entity is the key function that will ensure fairness among multiple mobile users. Fairness in this context means that simultaneous video sessions are progressively delivered to multiple users in such a way that each user receives video content with the expected QoE without allowing any user to hog bandwidth from other users that may thus be disadvantaged. For progressive video download, good QoE can be characterized as when a user is able to view a video without any interruptions or stalling that may occur due to buffer starvation (see, e.g., [10]). Buffer starvation occurs when on average the playback rate is greater than the rate at which video frames are being delivered to the users, and if the prebuffering phase is too short to compensate for the insufficient delivery rate.

A functional architecture of the proposed application-level scheduler for the mCSP entity is depicted in Fig. 2 and is referred to as Fair Scheduler for Video Pacing (FS4VP).

The FS4VP method schedules between multiple TCP flows, relying on a number of input parameters to enable generic pacing strategies. The input parameters are context information the mCSP is able to derive, such as:
- **User and device information:** QoS profile, codec information, device capabilities, and so on that can be acquired from network entities such as the home subscriber server (HSS) or PCRF, or from application layer signaling to the mCSP.
- **Network information:** Radio access technology such as high-speed packet data access (HSPDA) or Long Term Evolution (LTE), radio/backhaul throughput, link state (such as congestion), and so on.
- **TCP session information:** Congestion window (cwnd), receive window (rwnd), round-trip time (RTT), explicit congestion notification (ECN) bits, and so on. Since the mCSP is managing a split TCP, it can derive this information for each TCP session and provide it to the FS4VP scheduler. It should be noted that the ECN information also enables the FS4VP to determine the presence of congestion on the path.

By taking into account these context parameters (and coupling them with DPI as proposed in [11]), the FS4VP scheduler method can estimate the clients' playout buffer size and hence determine the optimum transmission parameters (e.g., transmission rate, chunk size) for each flow with reference to other ongoing flows in order to prevent buffer starvation. Based on these parameters, the FS4VP will control the content delivery rate (i.e., pace) at which the content data is being sent toward the underlying TCP stack. Hence, this will enable the FS4VP to intelligently and dynamically orchestrate among multiple flows to ensure that enough video data is available in the play buffer in advance of the current playtime to enable smooth playback during adverse network conditions such as congestion in the backhaul/access links and/or fading.

This approach is in contrast to the application flow control technique adopted by YouTube that uses a block sending approach where content is sent at a constant rate in blocks of size 64 kbytes after the initial quick startup phase. This results in burstiness resulting in congestion and accounts for over 40 percent packet loss in residential digital subscriber lines (DSLs) [12].

The proposed framework for the application-level scheduler goes beyond known video pacing schemes, such as proposed in [6], in that the algorithms for scheduling on the mCSP are extendable to take advantage of mobile operator control plane information, such as current network resource status and subscriber information. Knowledge about the transport network topology and linkage of user location information with experienced congestion information allows building topology-congestion maps, which can be taken as a basis for controlled scheduling of content at mCSPs to temporarily prioritize delivery of chunks to a mobile device, whose buffer is about to deplete, above chunks for clients that may have many packets buffered in advance.

Application-Level Scheduling for Progressive Video Delivery Service

By means of an exemplary network configuration, we illustrate the benefit of employing application-level scheduling at the mCSP for ensuring fairness while video content is progressively delivered to multiple users over TCP. We assume that the
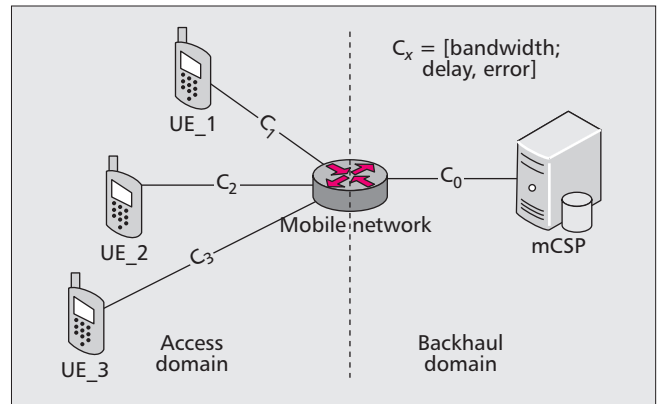

Figure 3. *Simulation topology.*

network provides different bandwidths to different user equipment (UE). The bandwidths to the UE are close to constant, as is the case, for example, in Universal Mobile Telecommunications System (UMTS) networks with dedicated bearers or overprovisioned LTE networks with per-bearer bandwidth limitation. Mobility is not considered. As a proof of concept we assume that the codec profile context information is sufficient for the scheduler at the mCSP node to derive buffer estimates. The reason for choosing a progressive mode of video streaming is that it is employed by YouTube's (and the like) video-on-demand service, which according to [13] is the driving factor of the recent increases in HTTP traffic.

For the purpose of illustration we have developed a simulation model using the OMNeT++'s INET framework [14] with the simulation topology depicted in Fig. 3. In this figure the mCSP is a TCP server, in which we have modeled a TCP server application and a context-aware fair scheduler. The mCSP node is connected to the mobile network via a backhaul link (C_0), which is abstracted by a router device. The clients, on the other hand, are TCP clients inside which we have realized a player model that plays back the received content data from the playout buffer in a manner similar to browser players. In this topology we have three clients, UE_1, UE_2, and UE_3, connected to the router module via links C_1, C_2, and C_3, respectively. The links are characterized by a 3-tuple [*bandwidth*, *delay*, *error*] parameter vector. It may be noted that the depicted topology is a common and representative scenario in mobile networks with limited capacity of backhaul links connecting the eNBs to the mobile network.

In our simulations, we vary the bandwidth of the backhaul link (C_0), and analyze its effect on the data throughput experienced by each UE and the state of the playout buffer. If the playout buffer is empty, video playback is stopped and the buffer is filled until a certain threshold is reached (50 frames in our case). This event is referred to as *stalling*.

We perform two sets of experiments, where in one set content is progressively delivered without any scheduling, and in the second set we schedule the content delivery based on the user codec profile context information. In our experiments, the content is represented by a video stream that is being delivered to the three UE devices upon request using TCP. For our experiments, we used the TCP New Reno variant. The simulation parameters are shown in Table 1.

In our simulations, we assume that UE_1 and UE_2 have on average better channel characteristics than UE_3, which is reflected in higher link bandwidths for UE_1 and UE_2 and lower bandwidth for UE_3. Assuming adaptive codec switching, suitable encoding rates will be assigned to the users. Thus, UE_1 and UE_2 are assigned higher encoding rates than UE_3, which will also determine the playback rate (in bits per second). We assume that the user gets served while

Simulation parameters		
File size	100 Mbytes	
MSS (TCP)	1452 bytes	
Link parameters	C_0	Bandwidth	Variable
		Delay	20 ms
		PER	0.0001
	C_1	Bandwidth	4 Mb/s
		Delay	5 ms
		PER	0.0005
	C_2	Bandwidth	2.5 Mb/s
		Delay	5 ms
		PER	0.0005
	C_3	Bandwidth	1.2 Mb/s
		Delay	5 ms
		PER	0.0005
Advance play buffer size (frames)	Initial	200
	Minimum	50
Playback rate (b/s)	UE_1	2 Mb/s
	UE_2	2 Mb/s
	UE_3	1 Mb/s

Table 1. *Simulation test parameters.*

connected to the same eNB and we don't consider trick-play (pausing, forwarding, rewinding operations) in our evaluation.

Impact of TCP on Progressive Video Delivery

In the first scenario, the mCSP delivers content data without employing any scheduling and dumps the content into the send buffer of the underlying TCP stack. Once in the send buffer, TCP takes over the delivery of the content to the respective users. Figure 4 shows the effect of varying the bandwidth of link C_0 on the data throughput experienced by the three users and the variation in the size of the play buffer. The user QoE can be assessed from the frequency of buffer starvation experienced by each user [4]. Note that buffer starvation leads to playback freezing or motion jerks, hence resulting in poor QoE.

As observed from Fig. 4a, when the bandwidth of the bottleneck link (i.e., C_0) is at 5.8 Mb/s, which is around 25 percent less than the available *aggregate* user link bandwidth of 7.7 Mb/s, the throughput variation experienced by UE_1 and UE_2 is much higher than the relatively low bandwidth requirement of UE_3. In terms of the user QoE, it is seen that all three users experience very low utilization of the play buffer. Also, since the playback rate is higher than the rate at which content is being delivered, all three users experience a

high frequency of buffer starvation events. The frequency at which the buffer starves is depicted in the inset of the graphs showing the play buffer size variation of Fig. 4 for all three UE devices. From this it can be verified that all three users experience extremely poor viewing quality due to frequent playback freezing and motion jerks.

Note that this is the case even though the capacity of the bottleneck link C_0 is larger than the aggregated bandwidth requirement of the video streams, which is 5 Mb/s. This is caused by packet drops at the bottleneck queue, each leading to a reduction of the congestion window at the sender by a factor of 1/2.

Increasing the bandwidth of the bottleneck link to 6.5 Mb/s (Fig. 4b) brings an immediate improvement in the throughput performance for UE_1, which experiences no buffer starvation; hence, smooth playback is experienced by the user. However, the performance of UE_2 and UE_3 does not improve, exhibiting poor QoE. This is attributed to the fact that due to high available access bandwidth for UE_1, the cwnd for UE_1 will be bigger than for the other two; hence, TCP will try to send more traffic to UE_1, thereby hogging the bandwidth away from UE_2 and UE_3 at the C_0. Note that in this case, the effective bandwidth of UE_2 and UE_3 is limited mainly by the access link queues at the eNodeB.

Similar behavior is observed when the bandwidth of C_0 is further increased to 7 Mb/s (Fig. 4c), although still less than the aggregate available bandwidth for the three UE devices. UE_1 is still receiving premium viewing quality with the play buffer filling up quickly, whereas the buffer utilization for both UE_2 and UE_3 remains low, resulting in poor viewing quality.

It is thus observed that TCP may ensure the reliable delivery of data packets; however, it does not ensure fairness among users. This is because of the greedy nature of TCP. where the sending rate is dictated by the cwnd size.

Benefit/Gain of Application Level Scheduling

We make use of the device capability information as a context on the basis of which we pace/schedule the delivery of video data to respective users.

We repeat the same set of experiments, but this time with the FS4VP scheduler controlling and managing the transmission rate between the TCP applications.

As a proof of concept, we use the *codec-profile* selected by the user device during the content request as context information for the scheduler. It may be noted that progressive download service providers like YouTube employ a similar mechanism to adapt the sending rate according to the video bit rate [15].

In this scenario, the chunk size is set to 3 × maximum segment size (MSS) (4536 bytes) with a transmission rate 0.5 percent higher than the playback rate. The transmission rate is kept slightly higher than the playback rate in order to prevent against the lagging of the playout buffer. The playback rate is known to the mCSP based on the codec profile selected during the content request [10]. The transmission rate is heuristically determined in order to prevent against buffer overrun (i.e., to prevent against the playout buffers filling in too quickly).The impact of our FS4VP scheme on the throughput experienced by the UE devices and their playout buffer utilization is shown in Fig. 5. As can be seen, not only is the content delivered to the UE with an almost constant throughput, but the play buffer never gets starved, thereby delivering video content with 100 percent smooth playback, even when the bandwidth of the bottleneck link is as low as 5.8 Mb/s.
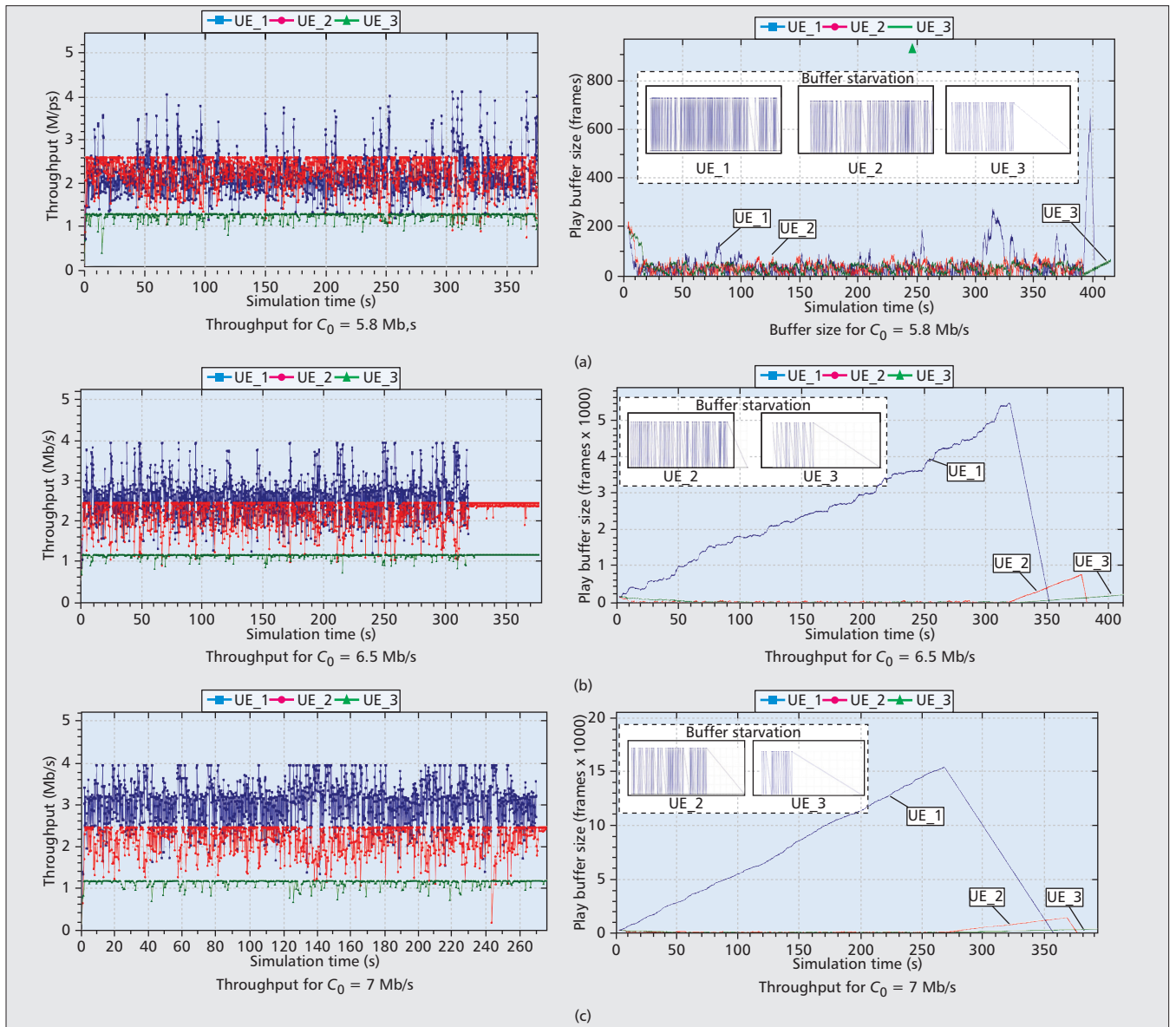
Figure 4. *Throughput and play buffer size variation at clients' nodes for different bandwidth values at the backhaul link.*

These experiments demonstrate the feasibility of controlling the sending rate at the application level and indicate the potential of FS4VP schemes for obtaining the desirable service quality for mobile content delivery, even in bandwidth limited scenarios.

Conclusions

This article addresses the trend to deploy so-called mobile content distribution networks as integral part of a mobile operator network for delivery of content to mobile devices and proposes a framework for Fair Scheduler for Video Pacing (FS4VP) on CDN serving points to improve the content delivery experience over TCP. The functional component of the proposed fair scheduler leverages locally available transport and application layer information, such as TCP session statistics and content encoding rates, to enable fair distribution of shared transport network capacity among mobile users. The fair scheduling algorithm also utilizes information that can be retrieved from the mobile infrastructure components, such as congestion levels or subscriber information. The fairness aspect focuses here on the adaptation of the scheduling characteristics of individual content delivery sessions that traverse a congested network segment. The scheduling characteristics determine the chunk sending rate per delivery session and are periodically updated according to the current information about the congestion as well as the download progress and playout buffer estimation.

The potential of application-level scheduling has been demonstrated by means of a discrete event simulation. The results show that although TCP ensures the reliable delivery of data packets and strives to provide a basic level of fairness through its built-in congestion avoidance mechanisms, it fails to provide the desirable level of QoE to mobile users even though sufficient capacity is available. Taking into consideration application-level information such as content-specific information (e.g., media codes and associated playback rates) and application status information (e.g., delivery progress and playout buffer size estimations) in addition to network information (e.g., congestion levels) enables QoE to be improved as long as the network is not underprovisioned. And even in the latter case, the scheduler will be able to deliver satisfactory QoE to high-priority services and customers, according to MNO policies.
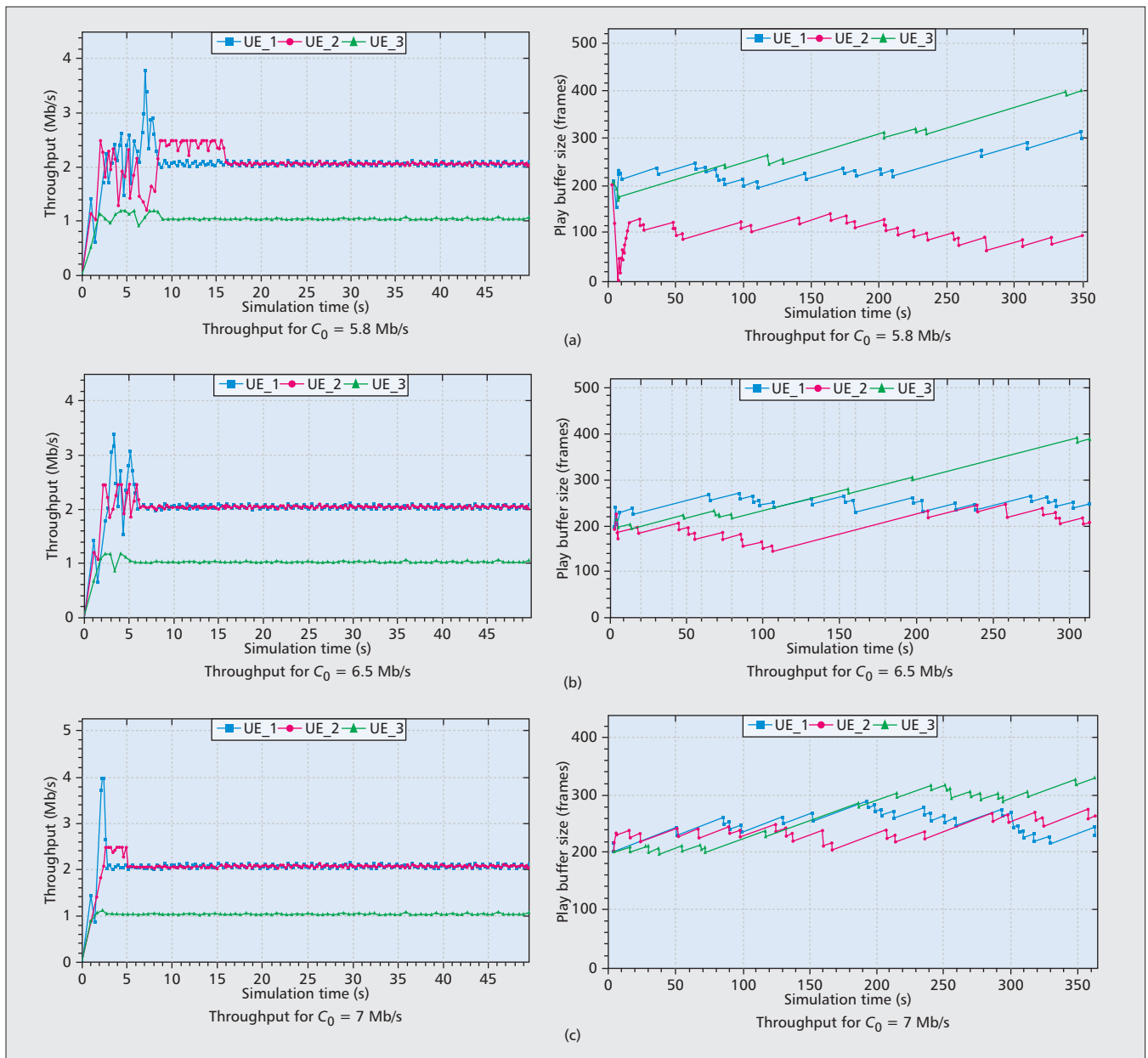
Figure 5. *Effect of the bandwidth of bottleneck link on the throughput and buffer size at the clients' nodes — with scheduling.*

Challenges for research and implementation remain in the concrete design of algorithms that gain advantage from control plane information. These include the generation of congestion-topology graphs as well as the identification and treatment of content delivery flows that traverse a congested region, as well as how to derive QoE for progressive video streaming from limited input such as proposed in [12], or development of appropriate fairness metrics for QoE.

References

[1] Cisco Visual Networking Index, Global Mobile Data Traffic Forecast Update, 2011–2016.
[2] M. Pathan and R. Buyya, "A Taxonomy and Survey of Content Delivery Networks," tech. rep. GRIDS-TR-2007-4, University of Melbourne, Australia, Feb. 2007.
[3] N. Amram *et al.*, "QoE-Based Transport Optimization for Video Delivery over Next Generation Cellular Networks," *6th Wksp. Multimedia Applications over Wireless Networks*, 2011.
[4] S. Wee *et al.*, "Research and Design of a Mobile Streaming Media Content Delivery Network," *IEEE Int'l. Conf. Multimedia & Expo*, July 2003.
[5] B. Yang, J. Liao, and X. Zhu, "Two-Level Proxy: the Media Streaming Cache Architecture for GPRS Mobile Network," *LNCS*, vol. 3961, Springer, 2006.
[6] M. Ghobadi, Y. Cheng, and A. Chain, "Trickle: Rate Limiting YouTube Video Streaming," *Proc. USENIX ATC '12*, Boston, MA, June 2012.
[7] S. Khan *et. al.*, "MOS-Based Multiuser Multiapplication Cross-Layer Optimization for Mobile Multimedia Communication," *Advances in Multimedia*, Hindawi, vol. 2007, 2007.
[8] Y. Falik, A. Averbuch, and U. Yechiali, "Transmission Algorithm for Video Streaming over Cellular Networks," *J. Wireless Networks*, Kluwer, vol. 16, no. 5, July 2010.
[9] 3GPP, "General Packet Radio Service (GPRS) Enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) Access," TS 23.401 V11.2.0, June 2012.
[10] T. Hossfeld *et al.*, "Quantification of YouTube QoE via Crowdsourcing," *Proc. IEEE MQoE 2011*, Dana Point, CA, Dec. 2011.
[11] R. Schatz, T. Hossfeld, and P. Casas, "Passive YouTube QoE Monitor-

ing for ISPs," *6th Int'l. Conf. Innovative Mobile and Internet Services in Ubiquitous Computing*, 2012.

[12] S. Alcock and R. Nelson, "Application Flow Control in YouTube Video Streams," *SIGCOMM Comp. Commun. Rev.*, vol. 41, no. 2, Apr. 2011, pp 24-30.

[13] G. Maier *et al.*, "On Dominant Characteristics of Residential Broadband Internet traffic," *Proc. 9th ACM SIGCOMM Internet Measurement Conf.*, 2009, pp. 90-102.

[14] INET Framework for OMNeT++, http://inet.omnetpp.org/

[15] P. Ameigeiras *et al.*, "Analysis and Modeling of YouTube Traffic," *Trans. Emerging Tel. Tech.*, vol. 23, 2012, pp. 360-77.

Biographies

FAQIR ZARRAR YOUSAF (zarrar.yousaf@neclab.eu) received his B.Sc. (1996) and M.Sc. (1999) in electrical engineering from N-W.F.P. University of Engineering and Technology, Peshawar, Pakistan. He received an M.Sc. in telecommunication and computers from George Washington University in 2001. He completed his Ph.D. from Dortmund University of Technology (TU Dortmund), Germany, in April 2010, where the topic of his research was seamless handover in mobile-IPv6-based next generation networks. His research work has been published in several peer-reviewed international journals, conferences, book chapters, and Internet-Drafts for IETF. He has served as a TPC member and reviewer for various conferences and journals. Currently he is working as a research scientist at NEC Laboratories Europe in Heidelberg, Germany.

MARCO LIEBSCH (marco.liebsch@neclab.eu) is currently working as a senior researcher at NEC Laboratories Europe in the area of mobility management, mobile content distribution, and mobile cloud networking. He has worked in different EU research projects and is contributing to standards in the IETF and 3GPP. For his thesis on paging and power saving in IP-based mobile communication networks, he received a Ph.D. from the University of Karlsruhe, Germany, in 2007. He has a long record of IETF contributions as well as RFC, journal, and conference publications.

ANDREAS MAEDER (andreas.maeder@neclab.eu) received his Ph.D. from the University of Wuerzburg, Germany, in 2008. He is currently affiliated with NEC Laboratories Europe as a senior researcher, working on next generation mobile wireless networks. The focus of his current research is application-driven optimization and QoE-based user plane congestion management in mobile networks. From 2008 to 2011 he participated in the standardization of IEEE 802.16m/WiMAX 2.0. Currently, he attends 3GPP and contributes to the EPS service architecture. He was and is involved in several research projects funded under the FP7 framework of the European Commission. He has served as TPC member for several IEEE and other international conferences and workshops. He was TPC Chair of the IEEE Broadband Wireless Access Workshop (BWA) 2012. He is an author of 20+ patent applications, as well as 35+ conference papers, articles, and book chapters.

STEFAN SCHMID (stefan.schmid@neclab.eu) received his M.Sc. in computer science (Dipl.-Inf.) from the University of Ulm, Germany, in 1999. In November 2002 he graduated from Lancaster University with a Ph.D. in computer science. After his post-doctorial research, he joined NEC Laboratories Europe in Heidelberg, where he was actively involved in several EU FP6 and FP7 research projects in the areas of 4G and future Internet research. In 2007, he was appointed group manager of the Next Generation Networking research team. Since 2009 he leads the Mobile and Wireless Networks research group at NEC Laboratories. He also directs NEC Europe's standardization activities in 3GPP and the Small Cell Forum, and is an active member of the System Architecture Working Group in 3GPP. He has published more than 60 scientific papers in international workshops, conferences, and journals.