Understanding Cross-Layer Effects on Quality of Experience for Video over NGMN

Muhammad Amir Mehmood, Cigdem Sengul, Nadi Sarrar and Anja Feldmann
Technische Universität Berlin, Deutsche Telekom Laboratories, Berlin, Germany
{amir,cigdem,nadi,anja}@net.t-labs.tu-berlin.de

Abstract—**The evolution of wireless network standards, e.g., GSM/GPRS, UMTS, WiFi, WiMAX, and end-user devices has paved the way towards Next Generation Mobile Networks (NGMN), where users are always connected through multiple radio access networks. NGMN technologies target to improve the user experience especially for mobile data and multimedia services, which are in line with user expectations evident from, for instance, the increasingly popular mobile web video streaming. To understand the quality that can be offered to the user, we compare the Quality of Experience (QoE) for web streaming in a prototype NGMN testbed with WiFi and 3G UMTS/HSDPA support. We use *CUBIC* TCP as the transport protocol as it is typically the default TCP variant, e.g., in Android phones. We complement the QoE estimations with network Quality of Service (QoS) parameters such as throughput and delay, and transport layer statistics. The results of our evaluation show that (i) video QoE remains stable in WiFi even with high packet loss, (ii) QoE in HSDPA is sensitive to packet loss even for low loss rates due to high variations in the network QoS, namely, throughput and delay, (iii) the decrease in QoE and QoS in HSDPA is due to its negative interactions with the aggressive congestion control of CUBIC TCP, and (iv) handover from WiFi to HSDPA degrades QoE.**

I. Introduction

Wireless Internet access has grown significantly over the last decade from GRPS and EDGE, to UMTS/HSDPA[1], UMTS/HSPA+, WiFi, WiMax and towards 4G LTE Advanced, promising ever more bandwidth to the users. Due to this multiplicity of choices, the NGMN alliance makes several recommendations including seamless mobility and roaming over different networks, end-to-end Quality of Service (QoS), and real-time and streaming support [1]. These recommendations follow user expectations, as today's users enjoy smart phones, which are equipped with multiple wireless network interfaces (e.g., UMTS or WiFi) and expect to be able to choose the right network according to their own personal preferences, network quality, and cost [2]. Additionally, Internet services, e.g., web video streaming, are growing in popularity among mobile users. One of the popular sites, YouTube Mobile, reports more than 100 million video playbacks per day [3].

Motivated by their popularity, in this paper we focus on Quality of Experience (QoE) in video streaming applications. These applications use pseudo-streaming and the content is delivered using HTTP/TCP [4]. Our goal is to understand the impact of time-varying channel characteristics in heterogeneous wireless networks on web video streaming QoE. The ideal way to assess video QoE is to perform perception tests. These tests help calculate a Mean Opinion Score (MOS), which expresses the mean quality score of a group of users according to ITU-T Rec. P.800 [5]. However, such an evaluation requires time-consuming and expensive subjective tests. Due to these limitations, we follow the objective video quality assessment approach and analyze the expected user experience by both a QoE metric - PSNR (Peak Signal to Noise Ratio) - and QoS metrics such as throughput, and delay under varying network conditions. Furthermore, to expose the cross-layer dynamics, we use TCP statistics which also enables us to correlate QoS and QoE behavior.

Our measurements include two types of access technologies: UMTS/HSDPA and WiFi, which show different link quality behavior over time. The main contribution of our work is an in-depth cross-layer QoS and QoE analysis of web video streaming across these technologies. Our results show that:

- **Better video QoE can be achieved with WiFi.** WiFi communication is limited by interference but shows more stable behavior over time. However, the transmission quality depends significantly on the concurrent demand on the wireless channel.
- **Video QoE in HSDPA is more sensitive to network dynamics.** The sensitivity stems from different reasons including transport layer interactions, scheduling and ARQ (Automatic Repeat Request) mechanisms.
- **The congestion control mechanisms used by TCP have a huge impact on the QoE.** We use *CUBIC* TCP [6], which is the default TCP variant used in todays popular mobile devices, e.g., Android phones. However, CUBIC TCP is designed for high speed networks and employs more aggressive congestion control, which in turn results in less stable communication, especially for HSDPA.
- **Handover from WiFi to HSDPA degrades performance.** Experiments show that WiFi to HSDPA handover degrades video QoE, while HSDPA to WiFi handover immediately improves QoE.

The rest of the paper is organized as follows. In Section II, we describe our NGMN testbed. We discuss our experimental methodology in Section III. In Section IV, we present the performance results. Section V presents related work and we summarize and discuss future work in Section VI.

[1]We use Universal Mobile Telecommunications System – UMTS and High Speed Downlink Packet Access – HSDPA interchangeably in the rest of the paper.

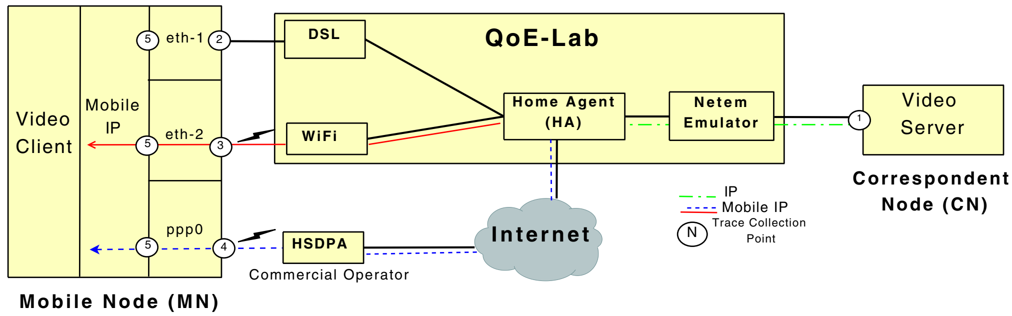


Fig. 1. Mobisense experimental setup for the video services in the NGMN.

II. NGMN Testbed

To understand the factors that affect QoS and QoE of web video streaming in NGMNs, the first task is to create a heterogeneous networking environment, which supports a mixture of different wired and wireless access technologies. Additionally, it is necessary to be able to trigger different networking conditions in a controlled manner to explore a wide variety of NGMN scenarios. To this end, we extend our NGMN VoIP testbed – Mobisense [7] – which already had access to WiFi, UMTS/HSDPA and DSL. We extend its capabilities by adding support for video services and linking it to our other large scale testbed – *QoE-Lab* [8], which enabled using control mechanisms such as mobility management with Mobile IPv4, network emulation and network monitoring.

The main components of our integrated testbed is shown in Figure 1, which consists of the Mobile Node (MN), the Correspondent Node (CN) and the Mobile IPv4 Home Agent (HA). The MN acts as the video client and the CN as the video server. All communication between the CN and the MN is managed by the HA. A *netem* network emulator exists between the CN and the HA to emulate different packet loss rates.

The CN and MN are laptops running Ubuntu Linux 2.6.28.16. The HA was configured on a *Cisco 7204 VXR* router with *IOS 12.1*. The MN has a WiFi, a HSDPA, and a 1 Gbps Ethernet interface. An additional virtual interface is created by the Mobile IP protocol. The CN is connected to the HA through 1 Gbps Ethernet, passing through the *netem*. The HA is dual-homed, i.e., connected to both the QoE-Lab and the Internet. We connect the HA to the Internet since HSDPA access is available through commercial operators only. The WiFi access point was configured as a standard IEEE 802.11g router with the transmission rate of 54 Mbps. The HSDPA connection as provided by a large European service provider operates at 7.2 Mbps for downlink.

For the mobility support, we rely on a "make-before-break" policy using *lmip*, a closed-source implementation of a Mobile IPv4 client. We use *lmip* at the MN, which allows the MN to perform network handovers during on-going sessions. Note that, while there is IPv4/TCP communication between the CN and the HA, the communication between HA and MN goes through a Mobile IPv4 UDP tunnel (port 434) irrespective of the selected access network. This constitutes a typical hybrid transport layer communication scenario while roaming.

We collect experiment traces by using *tcpdump* on all the physical and virtual interfaces of CN and MN. To monitor the dynamics of TCP, we use *tcp-hook* [9], a Linux kernel module based on the In-kernel Protocol Sniffer (IPS). Mainly, it provides a hook between TCP and the network layer. We deploy *tcp-hook* on the CN to observe TCP congestion control dynamics such as TCP congestion window size and estimated round trip times at the video server. We repeat the experiments multiple times with different packet loss rates.

III. Video Quality Evaluation in NGMNs

In this section we present our experiment methodology and discuss the metrics used for evaluating QoS and estimating QoE for web streaming.

A. Experiment Methodology

To create a web streaming environment, we install the TCP-based Tribler video streaming system [10] in the QoE-Lab testbed. A 10 minute movie-sequence, encoded in H.264 at 24fps (VGA-resolution), is streamed from the CN to the MN. On the video server (CN), we use the CUBIC TCP congestion control algorithm, which is the default in Linux distributions as well as Android phones. The key feature of CUBIC is that the congestion window growth relies on the real time between two congestion events [6]. It uses a cubic function for its window adjustment algorithm, hence its name. In addition, we use the default Linux settings, which include TCP window auto-tuning to dynamically adjust TCP's sender and receiver windows to better use the link capacity.

To characterize the performance, we vary the following:

Access technology: We associate the MN to WiFi and HSDPA networks in separate experiment instances. Note that WiFi is a simple technology, which uses CSMA/CA to grant access to the shared wireless medium. In contrast, HSDPA uses more complex mechanisms such as *fast link adaptation* to continuously adjust the modulation and coding scheme, *packet scheduling* that exploits the link quality information, and *Hybrid Automatic Repeat Request (HARQ)* and *soft combining* to let terminals recover from errors by explicitly requesting retransmissions and exploiting parts of erroneous frames [11]. These differences are expected to have an impact on transmissions and hence, web video streaming QoE.

Packet loss rate: As packet loss tends to be the dominant reason that affects multimedia quality, we inject additional

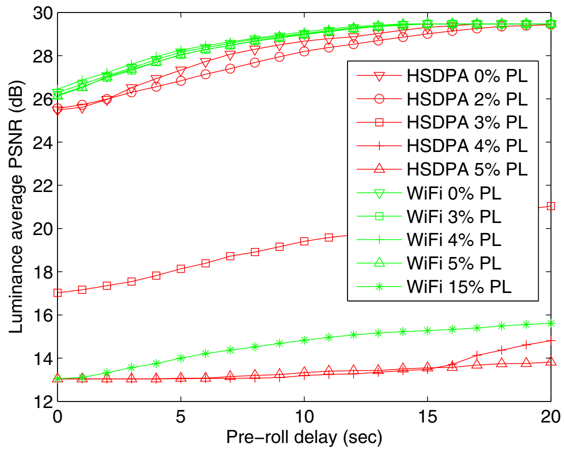


Fig. 2. PSNR (dB) vs. pre-roll delay (sec) at MN in WiFi and HSDPA

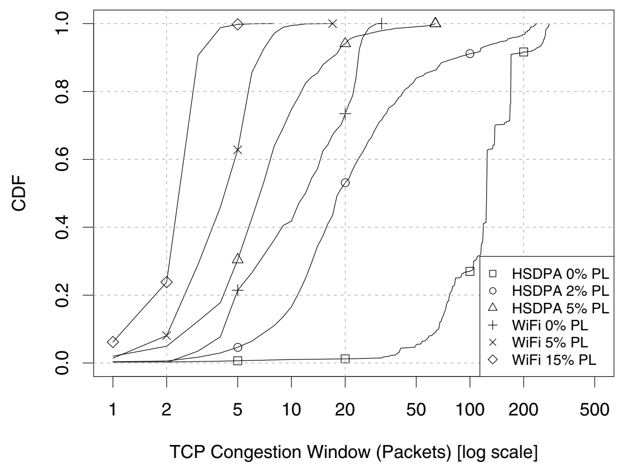


Fig. 3. TCP congestion window at CN in WiFi and HSDPA.

0-15% random packet loss in the video streams using *netem*.
Existence of network handovers: To test vertical network handovers and the effect of "make-before-break" policy on QoE, we force the MN to associate from WiFi to HSDPA and vice versa during an ongoing video transmission.
Video pre-roll delay: The pre-roll delay is the buffering time at the client's video decoder before playing back the video. Higher values delay the video start time but improve user perception. The pre-roll delay is varied from 0 to 20 s.

B. Video QoS and QoE

To capture the user expectations in mobile multimedia delivery, it is necessary to understand both QoS and QoE guarantees of NGMN [12]. To estimate the QoE of the video stream, we use the most commonly used metric in objective quality evaluations to estimate the user perception: PSNR (Peak Signal to Noise Ratio). We measure the PSNR and calculate it for different pre-roll delays between the original and the received video clip at the video client on the MN [13].

The video decoder uses copy-previous error concealment to compensate lost parts of the stream. If the video stream has N frames, the Mean Squared Error (MSE) for frame F_n ($0 \leq n < N$), given that F'_n is displayed instead of F_n by the decoder, is

$$M_n = \frac{1}{X \cdot Y} \sum_{x=1}^{X} \sum_{y=1}^{Y} [F_n(x,y) - F'_n(x,y)]^2, \qquad (1)$$

where the frame size is $X \times Y$ pixels. Denoting M as the average M_n for N video frames, the average quality in terms of PSNR, in dB, is then computed as:

$$PSNR = 10 \cdot log_{10} \frac{(255)^2}{M}, \qquad (2)$$

where 255 is the maximum luminance value of a pixel for 8-bit pictures.

To complement the QoE metric and to understand the corresponding network level QoS, we also compute throughput, delay, and TCP statistics such as RTO (Retransmission Time-Out), RTT (Round Trip Time) to ACK TCP segments, number of lost segments, duplicate acknowledgements, and fast retransmissions. Analyzing information from different layers provides insight into the QoS for different access technologies and NGMN conditions. Next, we present our performance evaluation in terms of these QoS and QoE metrics.

IV. RESULTS

In this section we present our evaluation of quality of web video streaming in NGMN. We first discuss the QoS and QoE performance in WiFi and HSDPA networks separately. We also take a deeper look at TCP performance in WiFi and HSDPA, which illustrate the importance of understanding cross-layer trade-offs when evaluating video QoE in NGMNs. Finally, we conclude with a discussion of impact of network handovers.

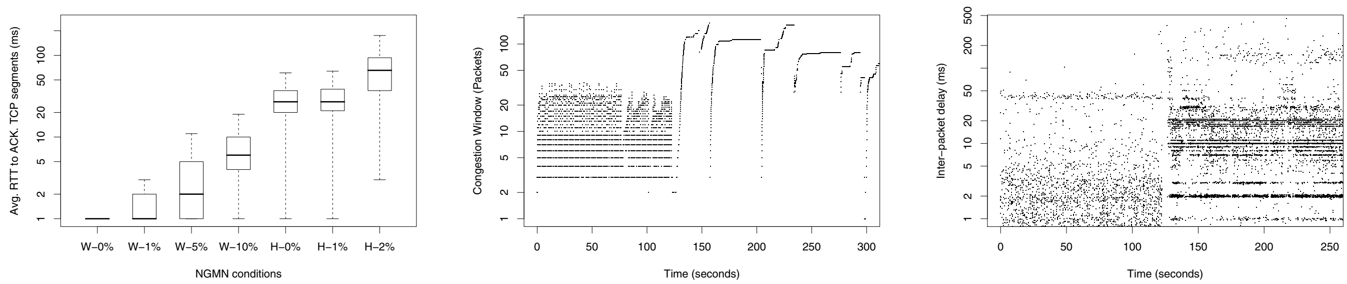
A. Web Video Streaming Performance with WiFi and HSDPA

We first evaluate video QoE in terms of PSNR in WiFi and HSDPA networks with different packet loss rates. We also vary the pre-roll delay to understand its effect on improving QoE. We observe that the performance is significantly different for the *same* packet loss rate depending on the access technology (WiFi or HSDPA). Fig. 2 shows that WiFi generally achieves higher PSNR compared to HSDPA. For instance, with 5% packet loss, the video quality remains high for WiFi (above 26 dB). However, with HSDPA and the same loss rate, the PSNR values become unacceptable. Such low quality is experienced for WiFi only with loss rates higher than 15%. Also, for a fixed PSNR value, WiFi requires shorter pre-roll delay (e.g., at 28 dB, pre-roll delay of $\approx 5s$ for WiFi and $\approx 10s$ for HSDPA) and therefore provides a better QoE.

Table I illustrates the same behavior from the network QoS point of view. While WiFi maintains a high throughput even with 10% loss rate, the throughput of HSDPA degrades immediately with 2% packet loss. Furthermore, for 0% loss rate, HSDPA exhibits less stability compared to WiFi with significantly higher standard deviation. This is also captured by the PSNR results, which point at unacceptable web video

TABLE I
AVERAGE THROUGHPUT AND TCP STATISTICS FOR WIFI AND HSDPA FOR VARYING PACKET LOSS.

Network	Loss (%)	Average Throughput and Standard deviation (kbps)	Lost Segments	Duplicate ACKs.	Fast Retransmits	Avg. # of RTOs per 1 s	% of intervals with RTO
WiFi	0	654 ± 230	17	131	11	1	0.917
	2	632 ± 243	663	1745	314	2.10	25
	5	661 ± 228	1752	4726	969	2.13	55.12
	10	680 ± 307	2793	8668	1932	2.01	65.64
HSDPA	0	698 ± 415	234	310	3	3.66	9.67
	1	655 ± 296	393	4628	78	2.73	17
	2	330 ± 155	518	3757	113	3.23	30.92



(a) Average RTT to ACK TCP segment variations in 1s bin, W:WiFi, H:HSDPA

(b) Congestion window vs. time, WiFi to HSDPA handover at 125 s, Loss=0%

(c) Inter-packet delay vs. time, WiFi to HSDPA handover at 125 s, Loss=0%

Fig. 4. Cross-layer interactions for WiFi and HSDPA with different packet loss rates and handover scenarios in NGMN (y-axis is log scale).

streaming qualities with HSDPA. We identify that the poor performance of HSDPA is due to the time varying channel availability due to shared downlink and higher RTTs that undermine TCP performance. We next explain these cross-layer effects in more detail.

B. A Closer Look at TCP and Lower Layer Interactions

Here, we discuss TCP statistics that clarify the performance differences between WiFi and HSDPA. In CUBIC TCP, when RTTs are short, the window growth rate could be lower as compared to traditional TCP due to the fixed window growth rate [6]. We observe this effect in WiFi, which experiences shorter RTTs and hence maintains smaller congestion windows compared to HSDPA. Fig. 4(a) shows the variations in average RTT to ACK TCP segments in 1 s bin. For WiFi RTTs are much shorter than those for HSDPA across all packet loss rates. Furthermore, the HSDPA RTT values show a higher variance at all packet loss rates hinting at the use of more complex link layer mechanisms (e.g., HARQ). Based on these RTT values, we calculate the bandwidth-delay product (BDP) for 0% packet loss, and get ≈ 7 KB for WiFi and ≈ 45 KB for HSDPA. This difference in BDP is also evident from the congestion window evolution (see Fig. 3). While WiFi congestion window sizes lie between $5-20$, HSDPA shoots its congestion window up to ≈ 250. However, as shown earlier, this does not necessarily lead to high throughput due to aggressive congestion window evolution.

Table I, where total number of losses, duplicate ACKs and fast retransmits are also reported, sheds more light on the difference in congestion window evolution in WiFi and HSDPA. In WiFi packet losses are higher. However, the majority of the losses are recovered during fast retransmit even for high loss rates (e.g., 10%). Still, these packet losses result in reducing the window size by a factor of $\beta = 0.2$ [6] and hence limit congestion window growth. In contrast, in HSDPA, the number of lost segments are lower (e.g., for 2% packet loss, the number of lost segments is 518 for HSDPA, and 663 for WiFi). However, the number of fast retransmissions is also significantly lower (e.g., 113 compared to 314, in HSDPA and WiFi). We infer that since the number of duplicate ACKs is high but the number of fast restransmits is low, the number of 3 consecutive duplicate ACKs is low in HSDPA. Hence, the high number of duplicate ACKs for HSDPA is mainly due to packet re-ordering and not packet loss. Essentially, as a result of the re-transmission activity of HARQ processes in HSDPA, subframes may arrive out-of-sequence. This is evident from the TCP traces, however, not possible to verify directly due to the use of commercial UMTS service.

Furthermore, we analyze the average number of retransmission timeouts (RTO) and how often RTOs occurs. Here, the total experiment time is divided into 1 s intervals, and the average number of retransmission timeouts is calculated for the intervals when at least one timeout occurs. TCP over HSDPA experiences more timeouts at a given interval and also, the percentage of intervals with timeout events increase as the loss rates increase. On the other hand, especially for the 0% loss case, WiFi experiences significantly less timeouts. Due to these combined effects, WiFi is able to maintain better video QoE.

C. The Effect of Vertical Network Handovers

Finally,we show an example case where we force a handover from WiFi to HSDPA at 125 s. As we discussed in Section II, we use a “make-before-break” policy. Figs. 4(b) and 4(c), depict the congestion window evolution and inter-packet delay with time, respectively. The results indicate that “make-before-break” policy interrupts web video streaming minimally and the user perception is not expected to be affected. Furthermore, in the graphs, the trends associated with each radio access technology is clearly visible indicating that the user perception is mainly determined by the access technology. Our results show that users should take advantage of WiFi networks to get better QoE for web video streaming applications, which are sensitive to high variations in throughput and retransmission timeouts. For HSDPA, CUBIC TCP is not a good choice and degrades performance.

V. Related work

The performance comparison of 3G and WiFi networks has gained recent interest in the research community. For instance, in [14], [15], the potential of opportunistic use of WiFi networks to reduce the load in 3G networks is investigated. Both works show that, in mobile scenarios, WiFi experiences frequent disconnections but it can provide higher throughput. On the other hand, 3G has stable coverage but offers lower throughput. Additionally, several works evaluated QoS/QoE for multimedia applications in either 3G or WiFi networks. In [16], HSDPA user experience was studied for HTTP/TCP and VoIP applications in a live network. It was shown that maximum capacity of the link is reached with relatively larger payloads and TCP performance is dependant on both uplink and downlink performance. In [17], the impact of MAC layer local retransmission mechanisms in 3G networks are listed as increased delay and rate variability for TCP. Similar to our results, performance degradation due to the aggressive congestion window evolution of CUBIC TCP was noted for WiMAX networks in [18].

Improving multimedia quality in NGMNs is also an active area of research. In [19], a cross-layer optimizer was proposed, which uses information about MAC layer conditions to perform video streaming rate adaptations (i.e., codec switching). Codec switching is also supported in [20], [21]. IEEE 802.21 [22] work group focuses on primarily network (vertical) handovers and enables co-operative handover decision making between clients and networks. Several works [23], [24], [25] study performance impact of vertical handovers for multimedia traffic in heterogeneous networks, typically via simulations. In this paper, we complement these related works by using both QoS and QoE metrics as well as TCP statistics to understand the web video streaming performance for different access technologies in real life conditions. These results provide a deeper understanding for the cross-layer affects between MAC and transport layers, and the conditions that bring gains from network switching.

VI. Summary and Outlook

Adoption of mobile web streaming will be dependent on the quality that the users receive as they roam between different networks. In this paper, we present a QoE study of web video streaming in NGMN scenarios. Our results can effectively be used for policy and decision making strategies such as video codec change-over or bit-rate adaptation to improve the video quality. Essentially, we show that with two different access technologies, WiFi and HSDPA, the QoS and the QoE performance is strictly tied to the interactions between the underlying MAC layer and the transport layer mechanisms. The default kernel settings (i.e., TCP Cubic variant with auto-tuning) are not recommended for HSDPA networks and call for cross-layer adaptation mechanisms, where TCP congestion control as well as the local ARQ mechanisms such as HARQ at the link layer are selectively used depending on the web video streaming QoS and QoE. To this end, for future work, we plan to evaluate different TCP congestion control mechanisms for web video streaming in NGMNs.

VII. Acknowledgments

We thank Pablo Vidales for insightful discussions and Alexander Manecke for his help in setting up the testbed.

References

[1] *Next Generation Mobile Networks Beyond HSPA & EVDO*, NGMN Alliance, Dec. 2006, white paper, http://www.ngmn.org/.

[2] OpenEPC, "Always Best Connected - Access Network Discovery and Selection," http://www.openepc.net/en/openepc/demo_scenarios/.

[3] "Youtube mobile gets a kick start," http://youtube-global.blogspot.com/2010/07/youtube-mobile-gets-kick-start.html.

[4] M. Saxena, U. Sharan, and S. Fahmy, "Analyzing Video Services in Web 2.0: A Global Perspective," in *NOSSDAV*, May 2008.

[5] ITU-T Rec. P.800, *Methods of Subjective Determination of Transmission Quality*, ITU, 1996.

[6] S. Ha, I. Rhee, and L. Xu, "CUBIC: A new TCP-friendly high-speed TCP variant," *ACM SIGOPS*, 2008.

[7] P. Vidales, N. Kirschnick, F. Steuer, B. Lewcio, M. Wältermann, and S. Möller, "Mobisense Testbed: Merging User Perception and Network Performance," in *Tridentcom*, 2008.

[8] A. Mehmood, A. Wundsam, S. Uhlig, D. Levin, N. Sarrar, and A. Feldmann, "QoE-Lab: Towards evaluating Quality of Experience for Future Internet Conditions," in *Tridentcom*, 2011.

[9] J. Yoo, T. Huhn, and J. Kim, "Active capture of wireless traces: overcome the lack in protocol analysis," in *Proceedings of WiNTECH*, 2008.

[10] "Tribler," http://www.tribler.org/trac.

[11] E. Dahlman, S. Parkvall, J. Skold, and P. Beming, *3G Evolution: HSPA and LTE for Mobile Broadband*. Academic Press, 2008.

[12] F. Agboma and A. Liotta, "Addressing user expectations in mobile content delivery," *Mobile Information Systems*, vol. 3, no. 3,4, pp. 153–164, Dec. 2007.

[13] S. Agarwal, J. P. Singh, A. Mavlankar, P. Baccichet, and B. Girod, "Performance of p2p live video streaming systems on a controlled test-bed," in *Tridentcom*, 2008.

[14] P. Deshpande, X. Hu, and S. R. Das, "Performance comparison of 3G and metro-scale WiFi for vehicular network access," in *IMC*, Nov. 2010.

[15] R. Gass and C. Diot, "An experimental performance comparison of 3G and Wi-Fi," in *PAM*, May 2010.

[16] M. Jurvansuu, J. Prokkola, M. Hanski, and P. Perälä, "HSDPA performance in live networks," in *ICC*, 2007.

[17] M. C. Chan and R. Ramjee, "TCP/IP performance over 3G wireless links with rate and delay variation," in *MobiCom*, Sep. 2002.

[18] E. Halepovic, Q. Wu, C. Williamson, and M. Ghaderi, "TCP over WiMAX: A measurement study," in *MASCOTS*, Sept. 2008.

[19] A. Abdallah, D. E. Meddour, T. Ahmed, and T. Rasheed, "Cross-layer design for optimized video streaming over heterogeneous networks," in *IWCMC*, June-July 2010.

[20] R. Pantos and W. May, "Http live streaming," June 2010, http://tools.ietf.org/html/draft-pantos-http-live-streaming-04.

[21] "Quavlive adaptive http streaming," http://www.quavlive.com/.

[22] "IEEE 802.21," http://www.ieee802.org/21/index.html.

[23] A. Calvagna and G. D. Modica, "A user-centric analysis of vertical handovers," in *WMASH*, 2004.

[24] A. Duran, E. del Pliego, and J. Alonso, "Effects of handover on voice quality in wireless convergent networks," in *IEEE RWS*, Jan 2007.

[25] B. Grönvall and I. Marsh, "Performance evaluation of voice handovers in real 802.11 networks," in *WiNMee*, Apr. 2006.