©ACM 2014. This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in ACM SIGCOMM Capacity Sharing Workshop (CSWS 2014).

# Dynamic Bandwidth Allocation for Multiple Network Connections: Improving User QoE and Network Usage of YouTube in Mobile Broadband

Florian Wamser, Thomas Zinner, and Phuoc Tran-Gia
University of Würzburg, Germany
{wamser, zinner, trangia}
@informatik.uni-wuerzburg.de

Jing Zhu
Intel Labs
Hillsboro, OR, United States
jing.z.zhu@intel.com

## ABSTRACT

Users expect a high level of application quality without annoying interruptions or delays when using applications in a communication network. This is particularly important in mobile environments where varying channel conditions, user mobility, and interference lead to a variation of the available network resources which ultimately affects application quality. A flexible selection of one or more access technologies, however, allows to overcome resource limitations of one specific access technology. An over-the-top (OTT) virtual access network (VAN) approach allows the aggregation of multiple wireless networks into a single IP pipe to provide users more bandwidth. To minimize energy consumption and radio resource utilization, an application-tailored usage of the access technologies as well as appropriate resource scheduling mechanisms in case of concurrent usage are required. In this paper, we perform a scenario-based investigation of the performance of an OTT VAN architecture. As scenario we choose a user watching a YouTube video clip, while a Wi-Fi and a cellular network are available. We evaluate the user perceived quality, cellular usage, and device power consumption based on testbed measurements.

## Categories and Subject Descriptors

C.2.1 [**COMPUTER-COMMUNICATION NETWORKS**]: Network Architecture and Design—*Network communications; Wireless communication*

## Keywords

Bandwidth Allocation for Multiple Network Connections; Cross-Layer; QoE; Mobile Broadband; YouTube

## 1. INTRODUCTION

The continuous growth of mobile traffic due to the introduction of smartphones, mobile broadband modems, tablets, and mobile applications overwhelms current wireless networks. Easy-to-use apps, and ultra-portable and powerful devices lead to a massive use and a high number of different applications which in turn introduce particularly high demands on the networks.

Today's smartphones are equipped with at least two access technologies, e.g. Wi-Fi and Cellular (3G or 4G LTE). Consequently, it is possible to use all available technologies for Internet access at the same time. A possible implementation of such a system is an OTT virtual access network based on UDP tunneling, which has been widely used in Mobile IP or for VPN (Virtual Private Networks). It allows the flexible usage of available wireless networks in an exclusive or concurrent manner. Hence, different network resources can be aggregated to overcome limitations of a single access technology and to provide a good user-perceived application quality. In order to minimize the energy consumption at the end device and the usage of the Cellular spectrum, the Cellular link should only be used if necessary. Based on the detailed information available for a certain application, different strategies to aggregate the Wi-Fi and Cellular link are possible, and can be compared in terms of benefits as well as drawbacks.

The aim of this paper is to evaluate application-aware algorithms which can aggregate multiple access links to provide additional resources if necessary. For that we focus on a mechanism on network layer, which requires additional hardware, but is also transparent to today's transport protocols. End host-based multipath protocols like Multipath-TCP [1] or SCTP-CMT [2] in contrast require changes at the end hosts, but rely on the available hardware. Multipath protocols further aim at ensuring QoS fairness across all utilized access links. This results in a fair share of network resources. However, with respect to the demand of different devices and applications, it might be necessary to explicitly allocate the network resources unfair to achieve a balanced quality at application layer.

The contribution of this paper is twofold. We define three algorithms in the OTT VAN architecture which take application type or application quality into account. The objective is to address the application demands to achieve an effective network. Then, we evaluate these algorithms on a scenario basis. We investigate a user watching a YouTube video clip, while a Wi-Fi and a Cellular network are available. We evaluate the user perceived quality, Cellular usage, and device power consumption using test-bed measurements.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
*CSWS'14,* August 18, 2014, Chicago, IL, USA.
Copyright 2014 ACM 978-1-4503-2991-0/14/08 ...$15.00.
http://dx.doi.org/10.1145/2630088.2630095.

The remainder of the paper is structured as follows. After the introduction, we summarize related work in the Section 2. The OTT VAN architecture is briefly described in Section 3. Afterwards, the cross-layer resource allocation algorithms are described in Section 4, followed by the description of the experiment scenario in Section 5. In Section 6, the results are presented. Section 7 concludes the paper and presents our most important findings.

## 2. RELATED WORK

There are several approaches available in literature to aggregate multiple access technologies. Most of them focus on the aggregation according to network parameters. However, there are also approaches that take especially into account the Quality of Experience (QoE) of the user in conjunction with the optimization of network-related parameters such as resource utilization. In general, the approaches address at least the following two basic issues: 1) *the technical network architecture*, and 2) *the network/resource selection metrics*.

Mobile traffic offloading uses complementary wireless network technologies to send data that are originally designed for mobile networks [3–6]. With this approach, mobile networks and other wireless networks can be used simultaneously for transmission. In [3], for example, traffic offloading to femtocells and Wi-Fi is proposed. In [4, 5], only Wi-Fi is considered. In [5], a heterogeneous wireless network architecture is introduced to offload mobile data traffic using opportunistic communications in addition to Wi-Fi networks. They also consider mobile to mobile offloading. Further, [6] proposes either single connectivity or full dual mode for Android mobile devices. A key feature is the full dual mode where the device uses both the cellular and the Wi-Fi network while routing network sessions between them. This is particularly similar to our architecture, which is used in this paper. The aggregation metric is, however, different. We take into account energy consumption and especially the application requirements.

Any kind of resource allocation or traffic offloading depends on a particular metric in order to evaluate the best possible allocation or network selection. In [6], a threshold-based metric is used, and network quality of service (QoS) parameters such as bandwidth, delay, jitter, and RSS are the most common ones in literature [6–8]. While it is important to fulfill the network QoS requirements, it is even more important for a network operator to meet the user's requirements. To achieve a good user-perceived quality, some resource management approaches have been proposed to directly consider user preferences or/and higher layer (e.g. application) metrics that can better reflect the end user's QoE than network QoS requirements. In [9], QoE metrics are used to design a handover system for heterogeneous multimedia wireless networks. In [10], a cross-layer scheduling and resource allocation for web applications in an LTE mobile communication network is proposed, which takes into account the QoE and service response time. In [11], an app-radio cross-layer framework for improving QoE of Internet applications is proposed. The results show that the framework can improve the audio MOS of a Skype video call. However, we have seen little efforts from the literature on how to apply cross-layer optimization and use QoE metrics in multi-radio (Wi-Fi, Cellular, etc.) resource management, which is the main interest of this paper.

## 3. DYNAMIC TRAFFIC OFFLOADING ARCHITECTURE

Figure 1 shows the OTT VAN architecture [12] for dynamic traffic offloading and network aggregation. For each network connection, a separate UDP tunnel is established between client and server, for example, via Wi-Fi and cellular radio access network (RAN). As shown in the figure, VAN operates transparently over the top of all RANs. Furthermore, a virtual network interface is needed on both client and a dedicated server in the Internet, which can be supported by today's operating systems, for example loopback interface in Windows and TUN interface in Linux.

As shown in the figure, the client will obtain an IP address for each network connection respectively, which will be used to establish UDP tunnels with the server. After UDP tunnels are established, the client will obtain an additional IP address from the server, and configure its virtual interface accordingly. This virtual IP address will be used by applications running on the client, and therefore the server also serves as the VAN gateway (GW) for the client. As a result, all the traffic to/from the client will go through the server, aka, VAN GW.

Dynamic traffic offloading with Wi-Fi/Cellular aggregation operates at the VAN level between client and server. Here, we introduce the key control parameter *Tunnel Burst Size (TBS)*, defined as the number of consecutive data packets sent over a tunnel. By adapting the TBS of each tunnel, we can balance the traffic load of a TCP flow between multiple access networks and improve the end-to-end throughput. To achieve the maximum throughput, it is clear that the TBS should be set proportional to the available bandwidth of each connection. For example, if the available bandwidth is 2 Mbps and 4 Mbps on Wi-Fi and Cellular respectively, we should set the TBS ratio to 1:2. In other words, for every three packets, we will send one over Wi-Fi and two over Cellular on average.

Please note that packets delivered over multiple networks may experience different end-to-end latency due to congestion or many other factors, and therefore arrive out of order. TCP will suffer from such out-of-order packet delivery. Hence, the VAN receiver must perform re-ordering to ensure packets are delivered to TCP in sequence. Typical mechanisms to ensure a proper ordering of the packets is to implement a re-sequencing buffer, as proposed in [13].

To detect application information, we propose to add an additional functionality - *application monitor* at the server (VAN GW). The application monitor examines IP payload information and detects relevant application information. This extends available DPI methods like [12] by estimating the current state of the application. For YouTube, the current playout buffer and the video quality are estimated at the VAN GW. Our implementation is based on YoMo (YouTube Monitor) [14, 15].

## 4. CROSS-LAYER ALGORITHMS FOR APPLICATION-AWARENESS

We will consider three cross-layer algorithms to take advantage of various application information and QoE metrics for YouTube streaming in the network. They differ in complexity and in the degree of detail of information that they need to know about YouTube.

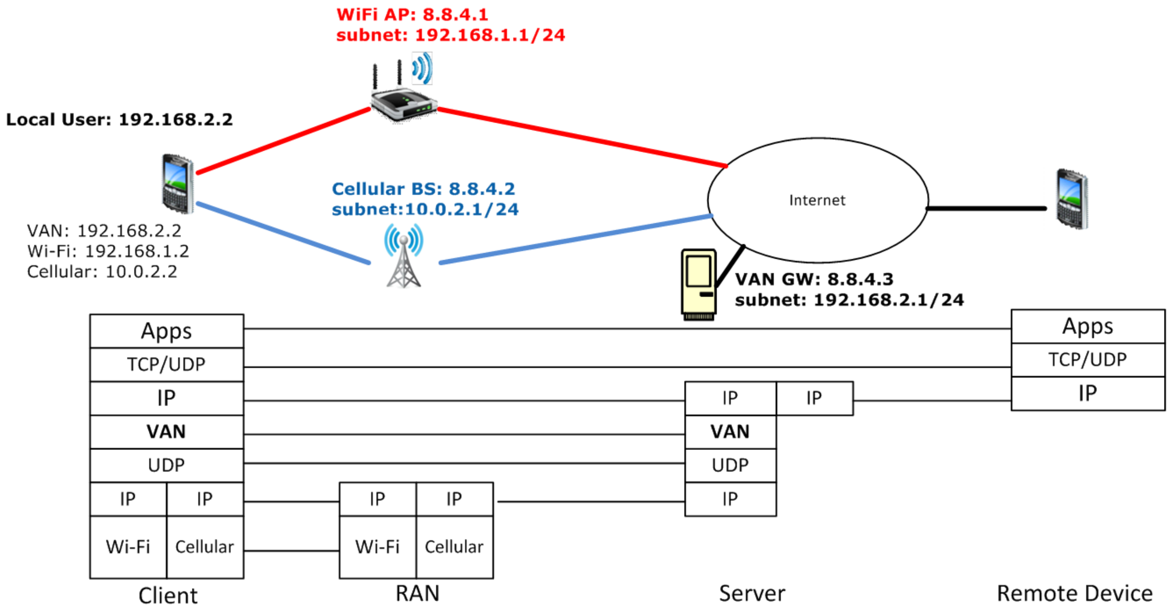



Figure 1: Over-The-Top Virtual Access Network (VAN) Architecture

**Algorithm 1: Aggregation Based on Required Throughput**
The first algorithm determines the bandwidth allocation per network connection based on the uplink request of the YouTube video player. YouTube uses progressive streaming over HTTP. Thus, within the uplink request, the requested video quality is encoded in the HTTP request parameters. The algorithm just uses the detected resolution as input parameter and decides about the required throughput for the user, cf. Table 1. After determining the required throughput, the TBS parameter will be adjusted accordingly. For example, if the available bandwidth is 2 Mbps and 4 Mbps on Wi-Fi and Cellular, with the requested video quality of 1080p and the required throughput of 6 Mbps, we will set the TBS ratio to 1 (Wi-Fi):2 (Cellular) so that both links will be run at their maximum speed. While with the requested video quality of 720p and the required throughput of 3 Mbps, we will set the TBS ratio to 2:1, and in this case the Cellular link will not be fully utilized with the throughput of 1 Mbps.

The advantage of the algorithm is that it needs to detect only the uplink request from YouTube. The downlink video data does not need to be analyzed. Its main disadvantage is that the bitrate of a video clip is commonly not fixed rather changing over time. The values in Table 1 are chosen carefully according to offline measurements so that they are applicable to a great variety of video clips for the selected resolutions.

Table 1: Lookup Table for Required Throughput

| Requested Video Quality | Required Throughput |
|---|---|
| 1080p | 6 Mbps |
| 720p | 3 Mbps |
| 480p | 1 Mbps |
| 360p | 0.5 Mbps |

**Algorithm 2: Dynamic Offloading Based on Buffer Estimation**
The second application-aware algorithm is based on the current buffered playtime for a YouTube video at the client. The idea behind this algorithm is to only add bandwidth of the Cellular link, whenever the client video playback buffer is low and thus, it is absolutely needed. Therefore, the buffer level has to be constantly monitored which is achieved by using YoMo [14, 15]. Unlike in the first algorithm, YoMo does not identify only the requested video quality, but estimates the current buffer level in seconds per client, dependent on the video content. The estimated buffer level is then updated only whenever new video data is downloaded. Thereafter, the required throughput $req\_tpt_{dyn}(x)$ is calculated with the following equation, which uses the current buffer level $x$ in seconds as input. With $tpt_{min}$ as the minimum desired throughput for a video stream and $tpt_{max}$ as the maximum allowed throughput,

$$req\_tpt_{dyn}(x) = \begin{cases} tpt_{max}, & \text{if buffer } x < \alpha \\ tpt_{min}, & \text{if buffer } x > \beta \\ \psi(x-\alpha) + tpt_{max}, & \text{otherwise} \end{cases}$$

where $\psi = -\frac{tpt_{max} - tpt_{min}}{\beta - \alpha}$ and $\beta > \alpha$. $\beta$ is the upper threshold where the video buffer is sufficiently filled and $\alpha$ is the critical threshold where the video runs out of buffered video content. When using this algorithm, the required throughput must not be updated too frequently to avoid signaling congestion, as it takes a few seconds to adapt the offload ratio. Therefore, the value of the required throughput is always rounded to the next higher integer value. The advantage here is that with the consideration of the instantaneous buffer level per user, the quality, and encoding of the current video is taken into account.

**Algorithm 3: Burst-wise Offloading Based on Buffer Estimation**
The last algorithm we present is also dependent on the estimated buffered playtime which is obtained by YoMo. The idea behind this algorithm is to burst-wise achieve the maximum throughput by combining both the Wi-Fi link and the Cellular link. Whenever the buffer level falls below a specific threshold, both links will be fully utilized until the buffer level exceeds the threshold $\beta$. Thereafter, the 3G mobile link will be disabled and only the Wi-Fi link is utilized again. Therefore, the required throughput can only be set to either maximum bandwidth or 0.0 Mbps. The corresponding function, which uses the current buffer level as input, can be seen in the following:

$$req\_tpt_{burst}(x, t_n) = \begin{cases} tpt_{max}, & \text{if buffer } x < \alpha \\ 0.0, & \text{if buffer } x > \beta \\ f_{burst}(x, t_{n-1}), & \text{otherwise} \end{cases}$$

with $t_n$ as $n$th calculation point in time and $tpt_{max}$ as the maximum throughput, $req\_tpt_{dyn}(x, 0) = tpt_{max}$. Just like the previous algorithm, this algorithm is supposed to utilize the Cellular link only when it is necessary to prevent stalling. Nevertheless, the Cellular link can be periodically disabled for a duration which is defined as $\beta - \alpha$, the time it takes for the buffer level to drop from the high threshold to the critical threshold assuming zero throughput. With this approach, we can turn off the Cellular link as much as possible to maximize the energy savings.

## 5. MEASUREMENT SCENARIO

We set up a test-bed to evaluate the proposed algorithms. Two access networks, Wi-Fi and Ethernet are connected to the VAN server, and the VAN server also works as the gateway to the public Internet. The Wi-Fi link is limited to 2 Mbps, the Ethernet link for emulating the Cellular link is limited to 4 Mbps in order to take resource limitations such as background traffic, for example, into account. The high threshold $\beta$ is set to 50 s and the critical threshold $\alpha$ is set to 20 s.

We randomly select 10 different YouTube videos with varying bit rates, each with a maximum resolution of 1080p and minimum length of 20 minutes. A complete evaluation run consists of 10 runs per video and resolution, each with a duration of 1000 s. During the runs, the following parameters are measured: the buffered playtime in seconds, the player state, which is either stalling or playing, the current throughput, and the usage ratio of Wi-Fi and cellular communications in relation to the current throughput. The application specific parameters, including buffered playtime and player state, are measured at the client, while the network parameters, such as throughput and usage ratio, are measured at the VAN server.

To quantify the benefit of the algorithms, we consider the following performance metrics:

**Quality of Experience (QoE)**: QoE indicates how satisfied a user is with YouTube video playback. Here, in particular, the interruption of the video playback, i.e. the stalling of the video, is used to measure QoE [16, 17].

**Cellular network resources**: For a mobile user as well as for a network operator, the Cellular resource usage is also important, and measured by the total number of bits (MB) delivered over the Cellular network (emulated by the Ethernet link on the test-bed).

**Energy consumption**: Especially in mobile communications, the energy consumption of the end user device is another important metric.

## 6. PERFORMANCE EVALUATION

This section highlights the results of our evaluations for various YouTube video clips and different video resolutions. First, we consider the impact of the proposed mechanisms on the video QoE in terms of video stalling events. Second, we evaluate the usage of cellular network resources which allows us to draw conclusions of the corresponding costs. Third, we highlight energy consumption as a second metric which is in particular important for mobile users.

### 6.1 Influence of the Algorithms to the Video Buffer Level

Video playback will be disturbed if application demands do not match the available resources, resulting in interruptions of the video playback.

In Table 2, the number of stalling events and the total stalling time is shown as an example for 5 videos in the baseline scenario with aggregation disabled. With an average video rate larger than the maximum Wi-Fi bandwidth (2 Mbps), stalling occurs, see Video #2 to Video #5. Further, the number of stalling events, as well as the stalling duration, increases with the video bit rate.

Table 2: Number of Stalling Occurrences and Total Stalling Time for 5 Different Videos

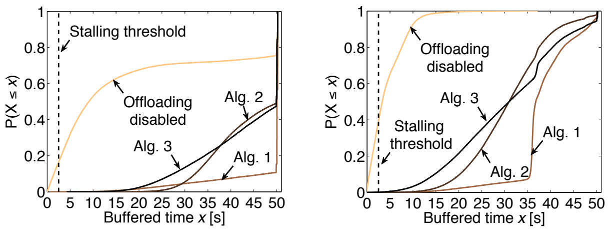
| | video rate | number of stallings | total stalling time |
|---|---|---|---|
| #1 | 1722 kbps | 0 | 0 s |
| #2 | 2154 kbps | 15.5 | 106 s |
| #3 | 2825 kbps | 26.6 | 202.9 s |
| #4 | 4101 kbps | 32.5 | 402.5 s |
| #5 | 4502 kbps | 33.6 | 413.6 s |

Figure 2 depicts the buffered playtime in seconds as CDF for 720p and 1080p resolution. The dashed vertical line for a buffer time of 2.5 s represents the stalling threshold. If the buffer falls below this value, the video stalls. The maximum buffer at the client is 50 s, due to the application layer flow control of YouTube.

Figure 2a shows the 720p scenario and Figure 2b the 1080p scenario. It can be seen that for both resolutions stalling occurs if aggregation is disabled. For 1080p, however, the probability for stalling is approximately doubled compared to the 720p case. Further, it can be seen that all algorithms manage to prevent stalling by utilizing the Cellular link.

### 6.2 Effects on Cellular Resources

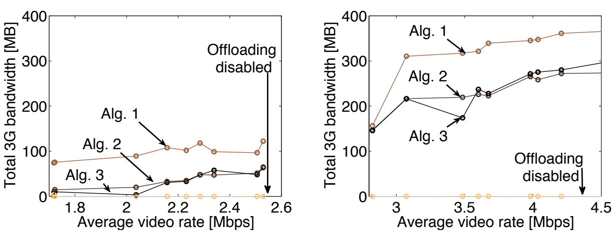
Today, Cellular usage is more and more charged on per bit basis instead of flat rate. Therefore it might be beneficial for the end user to minimize the Cellular data usage as much as possible without scarifying the user's expected QoE. Therefore, we investigate the total Cellular usage (in MB) during a playtime of 1000 s. The results for the proposed algorithms are illustrated in Figure 3. The x-axis depicts the average video bit-rate, the y-axis the Cellular usage. Figure 3a




(a) 720p (b) 1080p

Figure 2: Cumulative Distribution Function of the Buffer Level

highlights the results for 720p, Figure 3b shows the results for 1080p. Cellular is disabled without aggregation. Furthermore, regardless of which aggregation algorithm is used, the Cellular usage increases along with the average bit-rate. Moreover, the two buffer-based algorithms Dynamic (Alg. 2) and Burst-wise (Alg. 3) use the least amount of Cellular resources, and the difference between the two is negligible. In comparison, the static non-buffer based algorithm (Alg. 1) consumes far more Cellular resources.



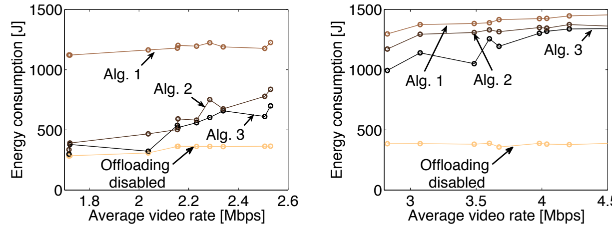

(a) 720p (b) 1080p

Figure 3: Comparison of the Total 3G Throughput

## 6.3 Energy Consumption of the User Equipment

Modern mobile devices, such as smartphones and tablets are limited by their battery life. Hence, we analyze the energy consumption of the various aggregation algorithms. To map Cellular usages to energy consumption, we rely on the power measurements proposed in [18]. We model the energy consumption with $\alpha_d = 122.12$ and $\beta = 817.88$ for Cellular (3G), and $\alpha_d = 137.01$ and $\beta = 132.86$ for Wi-Fi, respectively.

In Figure 4, the average energy consumption in Joule per 1000 s playtime is plotted against the average bit-rate of the video. Again, we detail the results for 720p in Figure 4a and 1080p in Figure 4b. In both cases, the overall energy consumption is lowest if only one interface, Wi-Fi, is used. This, however, results in a degraded QoE due to stalling events. Using both access networks simultaneously, however, increases the energy consumption. Alg. 1 (algorithm based on fixed required throughput) results in a continuous usage of both networks, which leads to the highest energy consumption. The two dynamic algorithms taking application quality information into account consume much less energy. Here, the total energy consumption will be reduced if the second network is used in a burst way, instead of continuously, for example, adjusting the download ratio based on the current buffer level in Alg. 2. Further, it can be seen that the energy savings are higher for the 720p scenario, because with lower throughput requirement, the Cellular link can be switched off more often and stay off for longer.




(a) 720p (b) 1080p

Figure 4: Comparison of Energy Consumption

Table 3 shows the average energy consumption results per 1000 s playtime based on the same set of data. The dynamic buffer-based algorithm (Alg. 2) consumes at least 46.8 % less energy for videos at a resolution of 720p and about 2.1 % for videos at 1080p compared to the non-buffer based algorithm (Alg. 1). The burst-wise algorithm (Alg. 3) performs even better and consumes 53,3 % less energy for videos at a resolution of 720p and 10.9 % less for 1080p. The high power efficiency with a buffer based algorithm is achieved by using the Cellular link only when it is needed from the perspective of the user's perceived QoE.

Table 3: Average Energy Consumption for a Playtime of 1000 s

| | Wi-Fi only | Alg. 1 (Uplink Request) | Alg. 2 (Dynamic) | Alg. 3 (Burst-wise) |
|---|---|---|---|---|
| 720p | 341.5 J | 1179.8 J | 590.7 J | 518.7 J |
| 1080p | 381.5 J (stalling) | 1404.1 J | 1299.0 J | 1182.9 J |


## Acknowledgements

The authors thank Steffen Höfner for his patience in the measurements and for conducting the experiments. Furthermore, thanks to the anonymous reviewers for their valuable comments.


# 7. CONCLUSION

In this paper, we presented an evaluation of a dynamic offloading architecture based on an OTT virtual access network architecture for bandwidth allocation with multiple access networks. We analyzed three different dynamic offloading algorithms for YouTube which differ in complexity and their impact on user and network. These mechanisms flexibly allocate network resources according to application information of YouTube which are detected within the network by a monitoring tool.

Our results show that any of the proposed offloading algorithm allow to enhance the QoE for end users if one of the networks can not provide enough resources. This is, however, due to the concurrent usage of cellular network resources. Therefore, we evaluated the usage of cellular network resources which allows us to draw conclusions of the corresponding costs. Finally, we investigated the impact of the offloading strategies on energy consumption which is in particular important for mobile users. The buffer based algorithms save up to 78 % of 3G traffic and consume up to 53 % less energy compared to the application-unaware algorithms.

As future work, we will investigate the scalability of our approach and evaluate it in a large field trial with many users and different varying channel conditions. We aim at providing a holistic resource allocation with respect to applications' needs for popular applications to allow for an efficient mobile network.

# 8. REFERENCES


[1] D. Wischik, M. Handley, and M. B. Braun, "The resource pooling principle," *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 5, pp. 47–52, 2008.

[2] J. R. Iyengar, P. D. Amer, and R. Stewart, "Concurrent multipath transfer using SCTP multihoming over independent end-to-end paths," *IEEE/ACM Transactions on Networking*, vol. 14, no. 5, pp. 951–964, 2006.

[3] A. Aijaz, H. Aghvami, and M. Amani, "A survey on mobile data offloading: Technical and business perspectives," *IEEE Wireless Communications*, vol. 20, no. 2, pp. 104–112, 2013.

[4] K. Lee, J. Lee, Y. Yi, I. Rhee, and S. Chong, "Mobile data offloading: how much can WiFi deliver?" in *International Conference Co-NEXT*, 2010, p. 26.

[5] B. Han, P. Hui, and A. Srinivasan, "Mobile data offloading in metropolitan area networks," *ACM SIGMOBILE Computing and Communications Review*, vol. 14, no. 4, pp. 28–30, 2011.

[6] A. Bar, D. Mimran, L. Chekina, Y. Elovici, and B. Shapira, "Nesto - network selection and traffic offloading system for android mobile devices," in *Wireless Communications and Mobile Computing Conference (IWCMC)*, 2013, pp. 337–342.

[7] E. Stevens-Navarro, Y. Lin, and V. Wong, "An MDP-based vertical handoff decision algorithm for heterogeneous wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 2, pp. 1243–1254, 2008.

[8] X. Yan, Y. Ahmet Şekercioğlu, and S. Narayanan, "A survey of vertical handover decision algorithms in fourth generation heterogeneous wireless networks," *Computer Networks*, vol. 54, no. 11, pp. 1848–1863, 2010.

[9] C. Quadros, E. Cerqueira, A. Neto, A. Pescapé, A. Riker, R. Immich, and M. Curado, "A quality of experience handover system for heterogeneous multimedia wireless networks," in *Computing, Networking and Communications (ICNC)*. IEEE, 2013, pp. 1064–1068.

[10] P. Ameigeiras, J. J. Ramos-Munoz, J. Navarro-Ortiz, P. Mogensen, and J. M. Lopez-Soler, "QoE oriented cross-layer design of a resource allocation algorithm in beyond 3G systems," *Computer Communications*, vol. 33, no. 5, pp. 571 – 582, 2010.

[11] J. Zhu, R. Vannithamby, C. Rodbro, M. Chen, and S. Vang Andersen, "Improving QoE for Skype video call in Mobile Broadband Network," in *IEEE Global Communications Conference (GLOBECOM)*, 2012, pp. 1938–1943.

[12] IPOQUE, "Case Study - Intel Labs Using DPI from IPOQUE," http://www.ipoque.com/sites/default/files/mediafiles/documents/CS_Intel_1308.pdf, 2013.

[13] T. Zinner, K. Tutschku, A. Nakao, and P. Tran-Gia, "Using Concurrent Multipath Transmission for Transport Virtualization: Analyzing Path Selection," in *Proceedings of the 22nd International Teletraffic Congress (ITC)*, Amsterdam, Netherlands, Sep. 2010.

[14] B. Staehle, M. Hirth, R. Pries, F. Wamser, and D. Staehle, "YoMo: a YouTube application comfort monitoring tool," in *New Dimensions in the Assessment and Support of Quality of Experience for Multimedia Applications*, Tampere, Finland, Jun. 2010.

[15] F. Wamser, D. Hock, M. Seufert, B. Staehle, R. Pries, and P. Tran-Gia, "Using buffered playtime for QoE-oriented resource management of YouTube video streaming," *Transactions on Emerging Telecommunications Technologies*, vol. 24, Apr. 2013.

[16] T. Hoßfeld, R. Schatz, M. Seufert, M. Hirth, T. Zinner, and P. Tran-Gia, "Quantification of YouTube QoE via crowdsourcing," in *IEEE International Workshop on Multimedia Quality of Experience (MQoE 2011)*, Dana Point, CA, USA, Dec. 2011.

[17] F. Dobrian, V. Sekar, A. Awan, I. Stoica, D. Joseph, A. Ganjam, J. Zhan, and H. Zhang, "Understanding the impact of video quality on user engagement," *ACM SIGCOMM Computer Communication Review*, pp. 362–373, 2011.

[18] J. Huang, F. Qian, A. Gerber, Z. M. Mao, S. Sen, and O. Spatscheck, "A close examination of performance and power characteristics of 4G LTE networks," in *ACM International conference on mobile systems, applications, and services*, 2012, pp. 225–238.