Demonstrating the Benefit of Joint Application and Network Control Within a Wireless Access Network

Florian Wamser, David Hock, Michael Seufert, Thomas Zinner, Phuoc Tran-Gia
University of Würzburg, Institute of Computer Science
{wamser,hock,michael.seufert,zinner,trangia}@informatik.uni-wuerzburg.de

I. Introduction and Problem Description

In today's access networks, the quality experienced by users of Internet applications is not always as good as possible. This is mainly due to the unawareness of the varying requirements of the services in today's network. It cannot assess how the user experiences the service and consequently cannot adapt to the user's demands. As a result, video streaming packets are treated in the same way as file downloads, i.e., real time services and best effort services are treated equally.

To cope with this problem we propose a dynamic cross-layer resource management for access networks with limited network resources based on application information [1]. In our demonstration we show the applicability and the benefit of the proposed architecture with respect to the Quality-of-Experience (QoE) of the involved applications. The Aquarema concept defines four logical units: *(1) application quality monitoring*, *(2) network monitoring*, *(3) resource management* and *(4) a central control entity* for the access network called network advisor, cf. Figure 1. To be able to respond to quality degradation, the QoE of the applications at the client is estimated and observed by application monitors. The data is sent to the central component, the network advisor. It stores all the information received from the application monitors. Additionally, it receives statistics from network monitors such as link utilization or link capacity. In case of a potential QoE degradation, the network advisor uses this information to coordinate the resource management tools that conduct control or resource management actions to adequately react on the current situation. This may include actions on application layer, like changing the resolution of a YouTube clip, or within the network, like adaptive resource allocation.

II. Implementation: The OC^{2}E^{2}AN Framework

The main idea of OC^{2}E^{2}AN (**O**ptimized **C**ontrol **C**enter for **E**xperience **E**nhancements in **A**ccess **N**etworks) is to collect information about the network and the applications in order to perform a suitable resource management. Therefore, the throughput is monitored at each router to the Internet. Further, the applications on the client are monitored by an application monitor to determine the current QoE of the user. The task of the monitor is to keep track of the status of traffic intensive or QoE sensitive applications that are subject to the resource management decisions. The status of an application is a collection of key performance indicators that the customer will directly perceive as quality parameters. These key performance indicators are application specific and describe whether the current performance offered by the network leads to a QoE degradation.

To provide a comprehensive resource management, the current version of OC^{2}E^{2}AN includes support for four different applications.

This work has been funded by the Federal Ministry of Education and Research under grant 01 BK 0800 (G-Lab, http://www.german-lab.de).

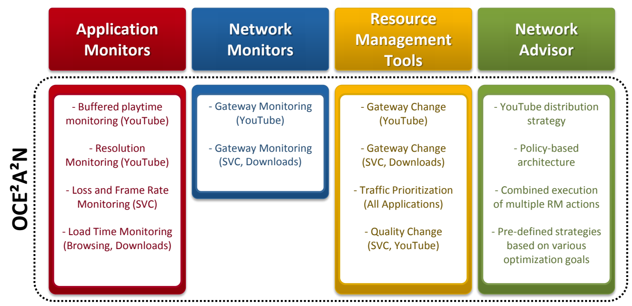

Fig. 1. Overview of OC^{2}E^{2}AN components.

Along with YouTube video streaming, resource management related to web browsing, file downloading and SVC video streaming is included. Table I provides an overview of the application parameters that were used to estimate the QoE of a particular service. According to the QoE model in [2], stalling is by far the main influence factor of YouTube QoE. Therefore, the corresponding application monitor observes the current buffered playtime of a user if he is watching a YouTube video [3]. For web browsing, the loading time of a web page is monitored and for downloads the current throughput is measured, c.f. [4]. The QoE of SVC video streaming is estimated according to the packet loss for certain video layers [5].

	YouTube	**SVC video**	**Browsing**	**Download**
Metrics	(1) stalling, (2) video resolution [2], [3]	(1) packet loss, (2) number of layers [5]	Main object load time [4]	Average throughput [4]

TABLE I
Overview on application parameters used for QoE estimation.

For the different applications, there are several resource management approaches to react either at the application level or at the network level within the access network.

The concept of network level control covers all measures which alter network properties or influence the packet flow in the network. The general goal of this concept is to improve the overall QoE of the users. To achieve this, the network has to react dynamically to changing (network) conditions and requirements of the involved applications. In particular, we implement prioritization and deprioritiziation of single flows in the access network and the distribution of the flows on different available gateways. With the first one, flows of urgent applications that instantaneously require data are preferred over other flows that have a good QoE or can tolerate a lower throughput. With the second action, a load balancing at the gateways according to the user QoE is done to utilize the available network resources.

The second type of resource management which is performed in this demonstration is application layer service control. This includes mechanisms that control applications so that the best QoE for the available network resources is delivered. For the presented application mix, such a control is possible for YouTube and SVC video streaming. For YouTube, an adaptation of the video resolution is performed, which reduces the required bandwidth, whereas the number of transmitted video layers is adjusted for SVC video streaming. All resource management approaches are conducted by the network advisor component based on the monitored application mix and network information.

To give an idea of the QoE increase that can be achieved with OC^2E^2AN, we present results for an exemplary scenario. It consists of four wireless mesh nodes. One serves as access point for the clients, the other three as gateways to the Internet. A total number of six services are subsequently started: 2 YouTube videos, 1 SVC video, 1 web browsing session, and 2 downloads. For these services we measure the user experience as Mean Opinion Score (MOS) with and without our resource management framework.

Fig. 2 compares the overall QoE measured as MOS by means of a cumulative distribution function for four different cases. In the

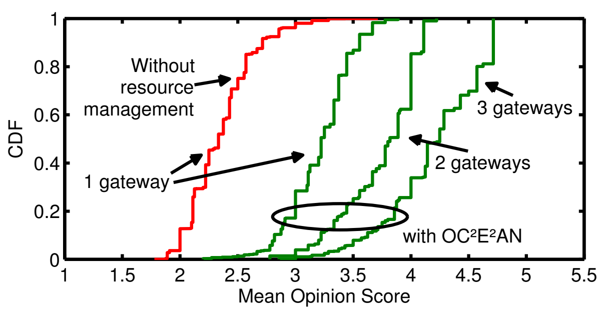

Fig. 2. Comparison of QoE with and without OC^2E^2AN

first case, only one gateway is used and OC^2E^2AN is deactivated. This corresponds to the normal behavior without any resource management. The average MOS of 2.35 is insufficient since the network is overloaded, indicating that most services are not running on an acceptable quality level. When using OC^2E^2AN (second case) the average MOS can be increased to a mean value of 3.21 without activating any additional gateways. With 2 or 3 active gateways (third and fourth case), the average MOS can be further increased to 3.74 and 4.20 respectively.

III. Demonstration Scenario and Presentation

To demonstrate the functionality of OC^2E^2AN as well as to foster an active discussion with the audience, we set up a demo scenario as shown in Fig. 3. Three laptops are connected to an access network equipped with the OC^2E^2AN framework, serving as end users which run different applications. The access network contains different gateways towards the Internet. As applications, video streaming of a two minute SVC video, an arbitrary YouTube video in standard definition resolution, up to six web downloads, and web browsing are used.

To illustrate the functionality of OC^2E^2AN as well as to allow for an easy interaction with the framework, a comprehensive graphical user interface has been developed. It consists of a statistics part and a network structure part and is displayed on the screens of the two monitors in Fig. 3. The statistics GUI, see left monitor, provides detailed information on all currently running applications. Grouped by application, each flow and its properties are displayed and the current QoE of each flow is indicated by color. The bottom part of this GUI, contains graphs showing the average QoE and network load in the network, as well as control elements to interact with OC^2E^2AN. The network structure GUI, see right monitor, visualizes the current network situation and active resource management mechanisms. It shows all the nodes in the access network and is able to highlight a selected flow from the user to the gateway. Moreover, it contains the current prioritization status and a log of all applied resource management actions so that the user can supervise the control actions and the current situation.

The network advisor is written in the Java programming language as well as all the application monitors except the one for SVC video streaming. This one is implemented in C++ since it is integrated within the video player being programmed in C++ as well. The network monitor is a C++ tool, too. It is designed for performance since it monitors the live situation in the network. The application monitor YoMo and the accompanying Mozilla Firefox extension are publicly available at [3].

OC^2E^2AN monitors the network and the QoE of the users as specified in Table I. It performs resource management actions such as dynamic traffic prioritization for certain critical flows. If necessary, it additionally changes the services' quality depending on the current network situation.

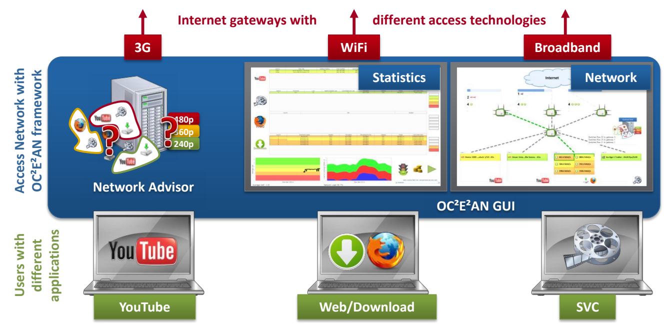

Fig. 3. Setup and components of OC^2E^2AN

The applications on the user laptops can either be started manually or automated according to a fixed chronological sequence. Similarly, the available resource management mechanisms can either be activated and deactivated manually or OC^2E^2AN can be setup to use different pre-defined strategies each composed of a set of collaborating resource management mechanisms. Continuously, the QoE of the users as well as the load in the network are displayed in the GUI, so that the effects of the different resource management actions can be visualized. That way, the audience of the demo can observe the influence of certain resource management mechanisms on the situation in the network as well as the improvements reached by using well-adjusted strategies.

References

[1] B. Staehle, M. Hirth, R. Pries, F. Wamser, and D. Staehle. Aquarema in Action: Improving the YouTube QoE in Wireless Mesh Networks. In *BCFIC'11*, Riga, Latvia, February 2011.

[2] T. Hoßfeld, R. Schatz, M. Seufert, M. Hirth, T. Zinner, and P. Tran-Gia. Quantification of YouTube QoE via Crowdsourcing. IEEE International Workshop on Multimedia Quality of Experience - Modeling, Evaluation, and Directions (MQoE 2011), Dana Point, CA, USA, December 2011.

[3] YouTube Application Monitor (YoMo), publicly available at http://www.german-lab.de/go/yomo.

[4] S. Egger, T. Hoßfeld, R. Schatz, and M. Fiedler, Waiting Times in Quality of Experience for Web Based Services. QoMEX 2012, Yarra Valley, Australia, July, 2012.

[5] T. Hoßfeld, M. Fiedler, and T. Zinner. The QoE Provisioning-Delivery-Hysteresis and Its Importance for Service Provisioning in the Future Internet. In *Proceedings of the 7th Conference on Next Generation Internet Networks (NGI)*, Kaiserslautern, Germany, June 2011.