Towards Sustained Multi Media Experience In The Future Mobile Internet

Kurt Tutschku and Albert Rafetseder
Chair of Future Communication
University of Vienna
Universitätsstrassse 10/T11
A-1090 Vienna, Austria
[kurt.tutschku,albert.rafetseder]@univie.ac.at

Jochen Eisl
Nokia Siemens Networks
GmbH & Co. KG
St.-Martin-Str. 53
81669 München, Germany
jochen.eisl@nsn.com

Werner Wiedermann
Telekom Austria AG
Lassallestrasse 9
A-1020 Wien, Austria
werner.wiedermann@telekomaustria.at

***Abstract*—High quality and sustained multi media experience is expected to be a major feature of the Future Mobile Internet (FMI). Currently deployed novel mobile network technologies such as LTE (Long Term Evolution) make use of the Internet and its best-effort TCP/IP protocol stack to enable applications. The orientation towards TCP/IP is logical due to the success of the Internet. However, TCP/IP still provides almost no support for Quality of Service (QoS). Hence, it seams to be a contradiction that high quality services can be reached by TCP/IP.**

In this contribution, we discuss recent advances for sustained high quality multi media experience in FMI. Therefore, we first cast some new lights on the achievement in video streaming. We review the LTE and EPS (Evolved Packet System) architecture of the anticipated FMI, and discuss particularly the concept of bearers, which is also used an abstraction for QoS support. Subsequently, we introduce new network concepts which might contribute to improved QoS and performance management in the FMI such as flow-based QoS enforcement in routers, Network Virtualization, and Network Federation. Based on this analysis, we develop an initial service for supporting improved flow management for video streaming service in the FMI. Finally, we outline how flow management can improve the quality of videos, but outline also what happens if inappropriate flow switching mechanisms are applied.

I. Introduction

Currently deployed, novel mobile network technologies, such as LTE (Long Term Evolution) make use of the Internet and its best-effort TCP/IP protocol. The use of TCP/IP is logical due to the success of the Internet, but is also remarkable since the TCP/IP concept provides two abstraction levels, which might be contradicting to current concepts of services in public land mobile networks (PLMN). The first abstraction level is the IP packet, also known as IP-over-x. It permits packet transport without knowing the details of the transmission. The second abstraction is obtained by the socket concept, c.f. [1]. Sockets form a very simple interface to data transfer services of best-effort type only The traditional strength of PLMNs, however, is the provisioning of high quality-services.

LTE is now expected to elevate the end-user experience for IP-based application to levels equal to wire-line networks. The implementation of this aim, though, leads to new problems. According to [2], "the dilemma for carriers (...) is that LTE's all-IP architecture will create a more open environment for Over the Top (OTT) applications, including VoIP services, which threaten to further commoditize the network". A possible converged Future Mobile Internet (FMI) architecture that encompasses different technical domains (wireline and wireless, decentralized and operator-centric) should address the specifics of the different notions of services. The expected polymorphic architecture for the Future Internet (FI) [3] might facilitate different notions of services in the FMI. However, the currently discussed high-level architectures for FI, e.g. as investigated in the GENI (Global Environment for Network Innovation) project [4] or the MANA architecture [5], do not detail mechanisms for service differentiation and sustained service provisioning that can be applied in concrete network deployments, such as LTE.

In this contribution, we outline first in Sec. II the concept for sustained multi media experience, such as continuous video transmission in FMI systems. Therefore, we first cast some lights on the requirements for continuous video streaming. After that, we review the LTE and EPS (Evolved Packet System) architecture of the anticipated Future Mobile Internet (FMI). We discuss in particular the concept of *bearers*, which are the abstraction concept for data flows and the QoS support mechanism in LTE/EPS. Subsequently, we discuss new network concepts which might contribute to improved QoS and performance management in the FMI such as flow-based QoS enforcement in routers, Network Virtualization, and Network Federation. Based on this analysis, we develop in Sec. III an initial service for supporting improved flow management for video streaming service in the FMI. Finally, we outline in Sec. IV how flow management can improve the quality of videos, but also what happens if inappropriate flow switching mechanisms are used. Finally, Sec. V summarized the our findings.

II. Mechanisms and Architecture for Sustained Video and Multi Media Experience

A. Modern Video Codecs

General Concepts. One central feature of modern video codecs is the *hierarchical syntax* for describing the video stream and layered video coding as a stream of data with

978-1-4244-7445-5/10/$26.00 © 2010 IEEE

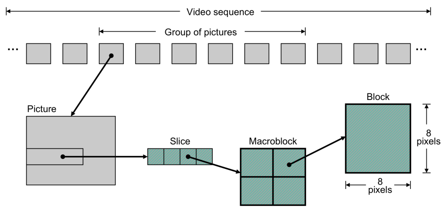

Fig. 1. Hierarchical Syntax for Video Sequences [6]

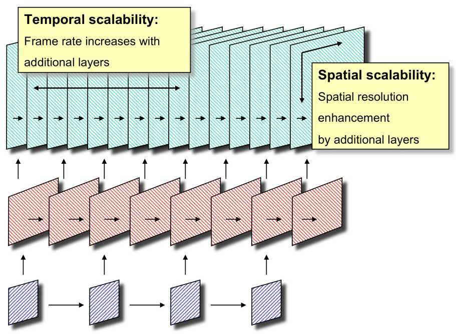

Fig. 2. Layered Video Coding [6]

defined starting points of objects in the stream, cf. Figure 1. These points are needed for decoding, inter-frame coding for higher coding efficiency (e.g. to support motion-compensated prediction), and for permitting adaptable picture resolutions. The *layered video coding concept* denotes the feature that a video sequence consists of multiple layers, cf. Figure 2. The availability of multiple layers at the decoder will improve the quality of the video, i.e. the more layers are available, the higher the quality is. In this way, multiple layers permit *scalability* in the video quality. Here, the term temporal scalability denotes the increase of the frame rate by additional layers, thus improving the motion perception of viewers. Additionally, the concept of spatial scalability describes the feature that the availability of multiple layers permits a picture with higher resolution.

Container Formats. Advances in modern multimedia content lead to the combination of different media types for a content. The content is wrapped in a file object called *container*. Containers comprise multiple types of different multimedia data. The container format specifies how data and meta-data is stored, rather than how it is coded. For example, a movie comprises a video track, but potentially multiple audio tracks, e.g. for different languages, and other types of information such as subtitles. Furthermore, the container may include meta-data such coding and accounting information.

H.264. Due to its capability to be applied in a large variety of scenarios, the H.264 video codec standard (in full denoted as H.264/AVC – Advanced Video Codec) [7], [8] is expected to become the most applied video codec. These scenarios range from professional high-definition (HD) video coding for digital production, HD and SD television, and even very low bit-rate applications such as MobileTV. Moreover, it is highly flexible the video resolutions and imposes relatively low computational requirements on the decoders. Thus, a number of Web2.0 video services now apply H.264, such as YouTube.

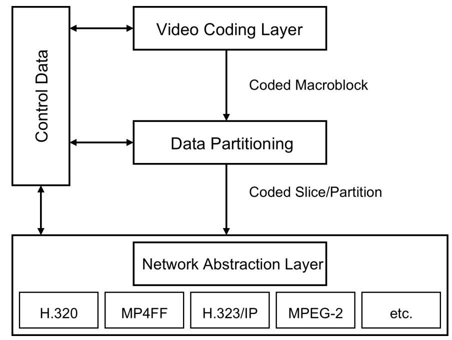

Fig. 3. Structure of H.264/AVC video encoder, cf. [8]

The structure of the H.264 codec is depicted in Figure 3. To address the need for flexibility and customizability, the H.264/AVC design covers a Video Coding Layer (VCL), which is designed to efficiently represent the video content, and a Network Adaptation Layer (NAL) described below.

The H.264 Network Adaptation Layer. The NAL formats the VCL representation of the video and provides header information in a manner appropriate for conveyance by particular transport layers (such as Real Time Transport Protocol, RTP) or storage media. It enables the mapping of VCL data to transport layer such as RTP/IP for any kind of real-time wire-line and wireless Internet services (conversational and streaming) or file formats, e.g. ISO MP4 for storage and multimedia messaging services, cf. [8]

In general, the NALs major subtasks within the mapping of the slice structure from the VCL to transport layer are the insertion of control and header information, and the framing, encapsulation and interleaving of slice data for increased robustness during transport. Other subtasks of the NAL are setup and closing of logical channels and timing and synchronization issues.

Flash Video is currently a pair of container file formats used for the transport of video for the Adobe Flash Player (versions 6-10). Flash Video content may also be embedded within Shockwave Flash (SWF) files [9]. Though the Flash Video container format itself is published, most of the compression formats used with it are patented, for example Flash Video FLV files contains material typically coded as Sorenson Spark or VP6 video compression formats. However, the most recent public releases of Flash Player also support H.264 video and HE-AAC audio.

The *HTML5 Video* tag is a markup language tag introduced in the HTML5 draft specification [10], adding support for embedding video. The current HTML5 draft specification, however, does not specify which video formats browsers should support in the video tag. User agents are free to support any video and container formats they feel are appropriate. After some discussions, H.264 now gets support for being the preferred video codec for HTML5 [11].

B. EPS Architecture and Challenges for the Bearer Concept

EPS. The Evolved Packet System (EPS) can be considered as an evolutionary step towards the notion of the future mobile

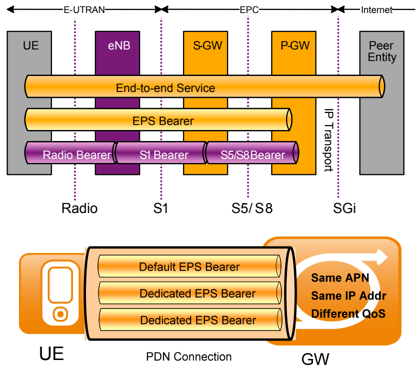

Fig. 4. Bearer concept in EPS

Internet. The term EPS defines a new radio interface for the cellular access named LTE long term evolution and a simplified and converged core network architecture EPC the evolved packet core. 3GPP has specified the EPS architecture to enable higher bandwidth for the cellular access at lower cost compared to UMTS due to higher spectral efficiency and less effort for operation and maintenance. The EPC is a packet-only core network, which can terminate diverse types of cellular accesses including legacy technology (e.g. UMTS, GSM) as well as non 3GPP access types and facilitate handover between these accesses. The architecture of EPC follows a split between control and data plane entities. The latter consists of two logical entities, the eNodeB (evolved NodeB) and the SAE-GW (Service Architecture Evolution GateWay). The eNodeB includes the radio interface to the end user, whereas the SAE-GW terminates the data plane services within the domain of the mobile access provider and represents the interface towards external networks, typically the Internet. This is a great simplification in comparison to legacy networks like UMTS, where parallel structures exist for circuit-switched and packet-switched data paths. The SAE-GW combines functions of two logical entities, the Serving Gateway (S-GW) and the Packet Data Network Gateway (PDN-GW or P-GW). The S-GW handles mobility when a UE moves between the same or different type of 3GPP access, whereas the PDN-GW, among other tasks, facilitates handover between 3GPP (e.g. LTE or UMTS) and non 3GPP (e.g. WIFI or WIMAX) networks.

QoS in the EPS. In general, the QoS concept in the EPS for services [12] is based on the notion of *bearers* and was designed, amongst other, especially for supporting streaming applications, e.g streaming video. In 3GPP networks a bearer represents the data path between the UE and the PDN-GW, i.e. an end-to-end service between UE and any other node in the Internet depends on the resources associated with the bearer, cf. Figure 4. A default bearer is established once the UE attaches to the radio access. For this purpose control plane components of the EPC are involved. The MME (mobility management entity) supports the bearer setup by selecting appropriate gateway nodes (S-GW and P-GW). During the setup another control plane entity is involved to allocate the resources that should be assigned to the default bearer according to pre-defined policy rules. The PCRF (policy control and charging rules function) is responsible for this task. Once the resources are committed by EPC, the MME returns this information to the eNodeB, so that the radio scheduler can setup the correct behavior within the radio access. Effectively the EPS bearer is a virtual construct, which consists of the radio bearer and a further bearer between eNodeB and S-GW. Another bearer does only exist, if the SAE-GW is split into different physical nodes for S-GW and P-GW, cf. Figure 4. Each of these bearers is associated with a tunnel, i.e. the EPS bearer represents the concatenation of the underlying bearers with respective tunnel endpoint identifiers for a specific bearer. Each EPS bearer can carry a set of data flows, all assigned with the same QoS forwarding behavior. As a consequence, a user needs to setup an additional dedicated bearer, if an application requests QoS treatment, which cant be satisfied by the default bearer. Additional dedicated bearers may be established, if a user needs further differentiated QoS treatment in case this is supported by the operator policies. Up to the writing of this paper, the feature of dedicated bearer is not yet supported by most UEs and mobile networks. After this excursion into the specifics of the bearer concept, we describe remaining technical challenges for the bearer concept.

Handover and bearer modification: When a terminal changes access to a different cellular access the EPS bearer needs to be linked to a different path in the network, at least the radio bearer is affected by this change. In general a bearer modification procedure has to be initiated in order to make this change happen. However QoS might have to be adapted as well according to the changed conditions. As a consequence another bearer modification procedure needs to be triggered right after the first one. In order to reduce the amount of signaling and enable fast adaptation mechanisms to changed conditions (such as caused by handover), improvement of the existing procedure should be targeted.

Cell load aware traffic management: Mobile operators in many cases complain about unfair usage of resource by certain application, such as P2P. Congestion usually occurs within specific radio cells - the EPC is not necessarily the source of the problem. Due to the algorithm in the radio scheduler all flows in the cell are impaired in case of high traffic load, even the traffic belonging to QoS classes with higher priority. This is because the scheduler does not assign fixed bandwidth portions to all configured classes. On the other hand the current layout of the architecture does not enable QoS enforcement within the radio access. There is no access from this location to QoS policy information related user or application data. This information is provided by the PCRF entity. In contrast policy enforcement can be done at the gateway nodes, such as P-GW. But at this node there is no information related to high load situations in specific radio cells. Applying general priority schemes at P-GW for downgrading traffic related to certain users or applications is ineffective. Such approach would not just pre-empt low priority traffic in congested cells but in all other cells as well.

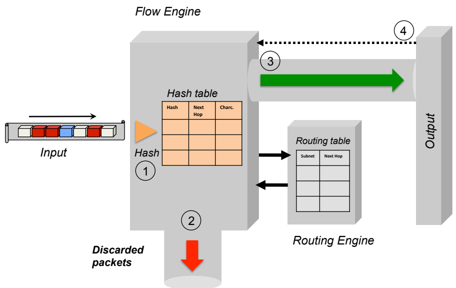

Fig. 5. A general flow-based forwarding architecture [13]

C. New Networking Concepts for the Future Mobile Internet

As outlined above, the upcoming large available access bandwidths in the MFI, e.g. due to the LTE radio interface technology, will require to refocus traffic and performance management mechanisms to core networks. Hence, we will outline next mechanisms which are applied this part of the EPS. In particular, we will start with the discussion of flow management concepts to similarity of the concepts of *flows* and *bearers*.

Flow-based QoS Enforcement. Sophisticated flow management is suggested to overcome disadvantages of DiffServ-like QoS architectures in routers [13]. The major ideas are that *a)* each packet contains the full information to assign it to a flow it belongs to, and *b)* that flow handling and packet forwarding are reversed. Reversing means in this context that flow information can be derived either from the header of the packet (e.g. the source/destination IP address, source/destination port, protocol) or in conjunction with the payload (which is identifiable by deep packet inspection). Sophisticated flow management identifies and imposes QoS actions before the packets of a flow are forwarded.

Figure 5 shows a general flow-based forwarding architecture derived from [13]. In Step 1, the Flow Engine computes a hash value for each packet's header. If there is a corresponding entry in the hash table then the packet either goes directly to the output port (Step 3), or might be discarded (Step 2). Due to the hash table, the Routing Engine is only consulted when the flow is seen for the first time. Then, the hash value and flow characteristics (such as header values, next hop, etc.) are stored there. The Flow Engine keeps track of each flow's characteristics, e.g. its duration or throughput for dedicated QoS enforcement. Additionally, the output can provide feedback to the Flow Engine (Step 4), e.g. on queue lengths or available bandwidth. Hence, flow-based forwarding may enforce QoS policies similar to DiffServ architecture. However, this architecture can perform the task in a more fine grained way since it can use the whole flow information for QoS enforcement decisions.

Network Virtualization. *Virtualization* of operating systems has become recently very popular due to its capability to consolidate multiple virtual servers into a single physical machine [14], [15]. Virtualization can be used to consolidate physical resources to reduce power consumption, maintenance and management costs. In addition, by loosening the binding of services to the physical resources providing those services, virtualization increases reconfiguration flexibility. The potential of virtualization technology to increase reliability, availability, and serviceability.

In *sharing* mode, the virtualization mechanism makes up multiple virtual resources from a single physical one and provides them to the upper layer, cf. Figure 6. Hereby, the sharing is coordinated and controlled by the *virtual machine monitor (VMM)* (or *hypervisor*). For example, the VMM cuts the CPU time into slices and assigns them to *virtual machines (VMs)* as virtual CPUs. Isolation and safe partitioning are key features of this type of virtualization at the hardware level.

Network Virtualization (NV) can be viewed as a transition of the operating system virtualization into the networking area. NV technologies cut the physical network into virtual network slices in parallel to cutting CPUs into slice, e.g. time slices. Thus, NV allows the simultaneous operation of multiple logical networks, often denoted as as overlays, on a single physical platform. NV permits distributed participants to create almost instantly their own network slice with application-specific naming, routing, and resource management mechanisms. Hence, network virtualization recently received tremendous attention since it is expected to be one of the major paradigms for the Future Internet as proposed by numerous international initiatives on future networks.

Network Federation. A solution to overcome shortcomings of todays Internet in the FI is to the concept of *Network Federation (NF)* [16], [4]. NF will facilitate new ways of collaborations of network resources. NF means the voluntary contribution of network resources and cooperation of networks for a limited lifetime, e.g. like resources provided by peers in an overlay. In detail, NF permits the interconnection of independently-owned and autonomously administered NV infrastructures in a way that permits their owners to define resource usage and allocation policies for the infrastructure under their control, operators to manage their infrastructures, and researchers to create and populate overlays, allocate resources to them, and run, currently *experiment-specific*, software in them. Figure 7 outlines the concept of federation. An experiment, later an application, combines resources which like to collaborate and thus, specifies a slice span across multiple providers, technologies, and even other countries. Due

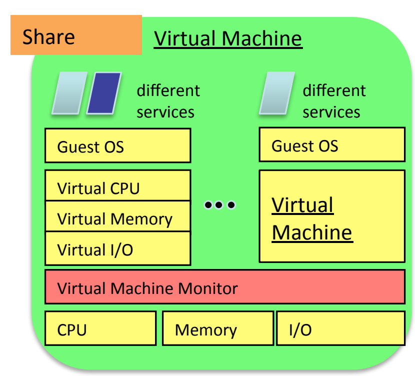

Fig. 6. Virtulization for sharing the access to resource [15]

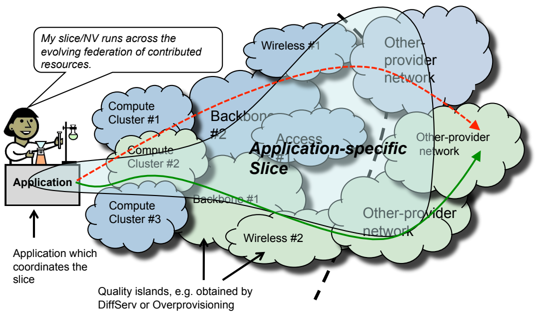

Fig. 7. Concept of Federation, after [17]

to the willingness of the resources to cooperate, NF is inherently suited for inter-domain performance management. This feature gives raise to the hope that NV enables long-desired features such inter-domain QoS and resilience. In Figure 7, for example, the brighter networks mark the cooperating system which provide for QoS, e.g. by some of them using flow-routing techniques (see above). Since the slice contains only willing networks, a data path doesn't pass through systems which are not able or incline supporting QoS.

D. Summary

Network Federation combined with Network Virtualization might provide the required adaptivity and integration of resources for the FI. Hereby, NV permits *vertical integration* that means the parallel operation of multiple networks in a single and basic infrastructure. NF facilitates *horizontal integration*, i.e. the integration of spatially distributed networks that are willing to collaborate. Moreover, if flow-routing or flow-switching concepts are applied then a fine-grained support of end-to-end QoS, in particular for video stream, appears to be possible. The flexibility of NV-based systems, the integration and cooperation features of NF and the use of the flow concept for traffic management, are expect to provide for a sustained multi media experience in the FMI. However, the above outlined technologies still miss an important part, which is the collaboration of the different layers. This feature will be addressed next.

III. The Resource Information Service

The previous discussion underlined that traffic management for overlays in the EPC cannot be accomplished by a single mechanism only. Traffic management is apparently most efficient when it is carried out on the virtual topology of overlays, i.e. when it interferes with the selection of nodes and the set-up of virtual connections. It should also exploit the parallels between bearers and virtual flows, e.g. by smart flow handling in routers or by flow management in the core network. The traffic management should also consider the quality and availability of physical resources in the transport system. Thus, the application overlay topology is made to match the resources provided by the underlay in the best possible way. A cooperation between the layers is needed so that the application overlay can optimally utilize the physical resources, cf. [18].

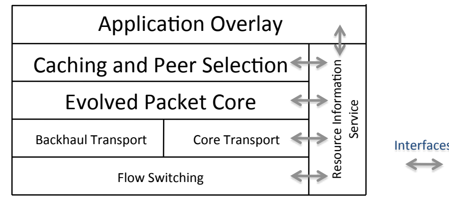

Fig. 8. Integrated, layer-cooperative traffic management

The RIS Concept

In order to enable cooperation between the EPS and the application overlay in an ordered way, we suggest the introduction of a *Resource Information Service (RIS)*. The RIS offers its capabilities as a service, thus allowing for different types of cooperation between applications and the EPS. It will collect information about the EPS, the quality of application overlay nodes (respectively of application overlay elements) in the EPS, and, if possible, also in the PDN. Finally, it and may provide this information to other overlay elements for peer selection. The structure of the RIS service is depicted in Figure 8. The RIS might mediate information between the application overlay and traffic management mechanisms on different layers such as caching, peer selection, or flow switching.

IV. Performance Impacts on Video Streams

As a proof of concept for path selection by a RIS instance, a prototype routing overlay was implemented in the *Seattle* Internet testbed [19]. Seattle provides an isolated virtual environment in which programs (called *experiments*) written in a restricted subset of the Python scripting language run. Seattle nodes generally use resources on donated machines, ranging from desktop PC's to smart phones. For this reason, care is taken that the virtual environment consumes minute amounts of the physical resources only, e.g. no more than 10kBps of network bandwidth, a small number of sockets, 5% of the CPU, etc. Furthermore, the experiments are not permitted to read or write anything other than their own files on the local system. What makes Seattle interesting for the purpose of prototyping a RIS is the distribution of Seattle nodes on the Internet (as opposed to a clean-room lab implementation with known conditions on the links, or a university-centric testbed such as Planetlab), and its well-designed restricted Python programming environment that offers a lot of abstraction for easy networking.

The principal demo scenario implemented for preliminary performance evaluation is seen in Figure 9. The source and destination, as well as the virtual routers (of which only one is shown) are Seattle nodes and participate in the routing overlay. In our setup, the experiment controller is on the same LAN as the video source and destination, but is not a Seattle node. The controller determines the overlay route through which the RTP/UDP-based video stream from the source to the destination is transmitted. By sending appropriate commands, it can reroute the stream on the fly. As indicated before, the

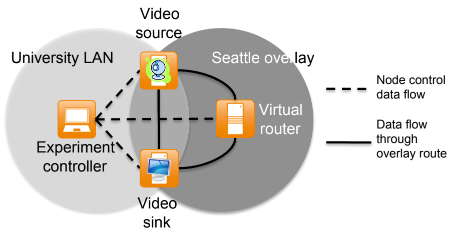

Fig. 9. Demo scenario

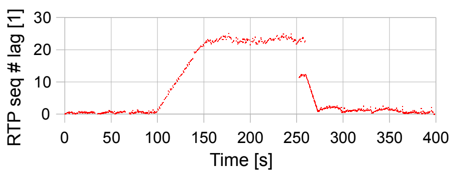

Fig. 10. RTP sequence number lag at the receiver

Seattle nodes the virtual routers run on are located on the public Internet at arbitrary locations.

As a measure of performance for the current route of the video stream through the overlay, we look at the RTP sequence number lag at the receiver, i.e. which RTP sequence number has already been played out by the source, but has not yet been received at the sink. For the first 100 seconds of the experiment, we notice little lag, although the route is changed at 50 seconds to include an additional hop.

At 100 seconds, we increase the network load on one overlay router by forcing the stream to go through it two times. Since Seattle limits the bandwidth per experiment, we would at first expect some kind of loss happening since UDP has no way of adapting the sender bandwidth to the receiver's capability. Interestingly, rather than dropping packets right away, the virtual router is seen to introduce an increasing amount of delay on the packets. It is only after additional 50 seconds that the lag between source and destination RTP sequence number stabilizes. At this point in time, the virtual router finally does drop packets. When, at 250 seconds, the overload condition is removed, some highly delayed packets and some packets with less delay are seen. The lag finally decreases to levels similar to the start of the experiment.

The reason for this buildup of lag lies in the way UDP socket buffer management works between an operating system (OS) and an application. The OS provides a certain amount of receive buffer space per socket. It will fill that space at the rate the sender sends packets, and drop packets when the buffer is full, implementing a tail-drop queue. The application can adjust its readout rate from the other side of the buffer, but that does not imply the OS is forced to receive packets more slowly. If, as in our case, the application feeds the output of its receive buffer to the input, every packet experiences additional delay, and has a higher probability of loss due to the buffer being full. In our experiment, the maximum delay was 15 seconds, with the loss reaching 27%.

V. Conclusion

In this contribution, we first outline the ways and requirements of video streaming in the MFI, as wellthe capabilities of supporting QoS in EPS by the use of the bearer concept. It became evident that flow management will a major performance management mechanism in the MFI. Such flow management can be achieved by flow-based routing/switching and flow-supporting NV and NF. The coordination of the different flow management mechanisms can be achieved by a *Resource Information Service*. Flow management might improve the QoS, however, significant care should be applied when implementing flow switching mechanism. Flow management will impose significant signaling on the control elements of the core network. Hence, we expect that novel load models for such elements will be needed.

References

[1] M. Donahoo and K. Calvert, *TCP/IP Sockets in C Pratical Guide for Programmers*, 2nd ed., Morgan Kaufmann, Burlington, Ma., 2009.

[2] L. Frankland. (2009) Scrutinizing the business case for lte. nokia siemens networks, unite magazine. [Online]. Available: http://unite.nokiasiemensnetworks.com/adwords/article/view/id/145

[3] European Commission, "Future Internet Assembly 2009 Conference Report," DG Information Society and Media Directorate for Converged Networks and Services, Nov. 2009.

[4] GENI Project Office, "Home page of the GENI Project; available at www.geni.net," 2008.

[5] A. Galis, H. Abramowicz, and M. Brunner, "Management and Service-aware Networking Architectures (MANA) for the Future Internet System Functions, Capabilities and Requirements, Position Paper, Version V6.0," available at http://www.future-internet.eu/fileadmin/documents/prague_documents/MANA_PositionPaper-Final.pdf, May 2009.

[6] B. Girod, "Image Communication II," Class Notes, Stanford University, available at: www.stanford.edu/class/ee398b, 2007.

[7] T. Wiegand and G. Sullivan, "The H.264/AVC Video Coding Standard," *IEEE Signal Processing Magazine*, vol. 148, Mar. 2007.

[8] T. Wiegand, G. J. Sullican, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC Video Coding Standard," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 13, no. 7, Jul. 2003.

[9] Anonymous, "Adobe Flash," Available online at http://en.wikipedia.org/wiki/Adobe_Flash.

[10] ——, "HTML5 video," Available online at http://en.wikipedia.org/wiki/HTML5_video.

[11] Apple Inc., "HTML5 and web standards," http://www.apple.com/html5/, 2010.

[12] H. Ekström, "QoS Control in the 3GPP Evolved Packet System," *IEEE Communications Magazine*, vol. 37, no. 2, Feb. 2009.

[13] L. Roberts, "A radical new router," *IEEE spectrum*, vol. 46, no. 7, pp. 34–39, 2009.

[14] R. Daley and J. Dennis, "Virtual Memory, Processes, and Sharing in MULTICS," *Communications of the ACM*, vol. 11, no. 5, May 1968.

[15] SCOPE Alliance, "I/O Virtualization: A NEP Perspective," Technical Report available at http://www.scope-alliance.org/, Feb. 2009.

[16] T. Spyropoulos, S. Fdida, and S. Kirkpatrick, "Future Internet Fundamentals and Measurements," *ACM SIGCOMM Computer Communication Review*, vol. 37, no. 2, Apr. 2007.

[17] L. Landweber and A. Falk, "A GENI Use-Case," Invited Talk at Future Internet Assembly, Madrid, Spain; available at http://www.ict-fireworks.eu/fileadmin/events/FIA-Madrid/Larry_Landweber.pdf, Dec. 2008.

[18] V. Gurbani, V. Hilt, I. Rimac, M. Tomsu, and E. Marocco, "A survey of research on the application-layer traffic optimization problem and the need for layer cooperation," *IEEE Communications Magazine*, vol. 47, no. 8, pp. 107–112, 2009.

[19] J. Cappos, I. Beschastnikh, A. Krishnamurthy, and T. Anderson, "Seattle: a platform for educational cloud computing," *ACM SIGCSE Bulletin*, vol. 41, no. 1, pp. 111–115, 2009.