Julius-Maximilians-Universität Würzburg
Institut für Informatik
Lehrstuhl für Kommunikationsnetze
Prof. Dr. P. Tran-Gia

Performance Modeling of QoE-Aware Multipath Video Transmission in the Future Internet

Thomas Zinner

Würzburger Beiträge zur
Leistungsbewertung Verteilter Systeme

Bericht 3/12

Würzburger Beiträge zur
Leistungsbewertung Verteilter Systeme

Herausgeber

Prof. Dr. P. Tran-Gia
Universität Würzburg
Institut für Informatik
Lehrstuhl für Kommunikationsnetze
Am Hubland
D-97074 Würzburg

Tel.: +49-931-31-86630
Fax.: +49-931-31-86632
email: trangia@informatik.uni-wuerzburg.de

Satz

Reproduktionsfähige Vorlage vom Autor.
Gesetzt in LaTeX Computer Modern 9pt.

ISSN 1432-8801

Performance Modeling of QoE-Aware Multipath Video Transmission in the Future Internet

Dissertation zur Erlangung des
naturwissenschaftlichen Doktorgrades
der Julius–Maximilians–Universität Würzburg

vorgelegt von

Thomas Zinner

aus

Würzburg 2012

Eingereicht am: August 8, 2012
bei der Fakultät für Mathematik und Informatik
1. Gutachter: Prof. Dr.-Ing. P. Tran-Gia
2. Gutachter: Prof. Dr.-Ing. Andreas Timm-Giel
Tag der mündlichen Prüfung: 13.7.2012

Danksagung

Mehr als fünf Jahre wissenschaftlicher Arbeit liegen nun hinter mir und ich schaue voller Vorfreude und Zuversicht auf die neuen Herausforderungen in meiner wissenschaftlichen Laufbahn. Dass ich für diese Aufgaben so gut gewappnet bin, verdanke ich vielen Menschen.

Mein Dank gebührt in erster Linie meinem Doktorvater Prof. Dr.-Ing. Phuoc Tran-Gia, der mir mit seinem Wissen und wertvollen Tipps jederzeit zur Seite stand. Ich möchte ihm für das mir entgegengebrachte Vertrauen danken. Schon früh übertrug er mir Verantwortung, ermöglichte mir die Teilnahme an zahlreichen Projekttreffen und Konferenzen und legte damit den Grundstein für meine fachliche Entwicklung sowie die Zusammenarbeit mit anderen Wissenschaftlern. Die so entstandenen Diskussionen führten zu neuen Impulsen für meine Forschung und hatten maßgeblichen Einfluss auf meine Dissertation. Nicht zuletzt verhalf mir Professor Tran-Gia zu einem ganz besonderen Kontakt, nämlich den zu meiner Freundin Sabrina Flurschütz.

Ebenfalls möchte ich mich bei Prof. Dr.-Ing. Andreas Timm-Giel bedanken, der als Zweitgutachter meiner Doktorarbeit fungierte. Mein weiterer Dank gilt den Mitgliedern der Prüfungskommission Prof. Dr. rer.pol. Andreas Hotho und Prof. Dr.-Ing. Hakan Kayal.

Besonderen Dank möchte ich Dr. Tobias Hoßfeld und Prof. Dr. Kurt Tutschku aussprechen, die mich seit meinen ersten Schritten als Student und Diplomand am Lehrstuhl 3 begleitet und unterstützt haben und in dieser Zeit zu guten Freunden geworden sind. Bei der Anfertigung der Dissertation haben mir Dr. Tobias Hoßfeld, Dr. Rastin Pries, Matthias Hartmann und Frank Lehrieder, mit Anregungen und Kommentaren zur Seite gestanden, wofür ich mich bedanken möchte.

Für die konstruktive, gemeinschaftliche Zusammenarbeit in nationalen wie internationalen Projekten möchte ich ferner Prof. Markus Fiedler, Prof. Yuval Shavitt, Dr.-Ing. Tanja Zseby, Christian Henke, Albert Rafetseder, Carsten Schmoll und Osama Abboud danken.

An dieser Stelle möchte ich die freundschaftliche Atmosphäre unter den Kollegen und Kolleginnen am Lehrstuhl hervorheben. Neben der Zeit auf der Arbeit verbrachten wir viele Abende und Urlaube gemeinsam, und waren Geocachen, Kickern, Laufen, Radfahren oder Snowboarden. Für die schöne gemeinsame Zeit möchte ich mich bei allen Mitarbeitern des Lehrstuhls bedanken. Insbesondere bedanke ich mich bei Dr. Michael Duelli und Frank Lehrieder, mit denen ich mehrere Jahre ein Büro geteilt habe, bei Dr. Andreas Mäder, Dr. Simon Oechsner und Dr. Dirk Staehle für ihre Unterstützung während meines ersten wissenschaftlichen Projektes. Weiterhin möchte ich mich bei Dr. Barbara Staehle und Dr. Alexander Klein bedanken, mit denen ich bereits während meines Studiums zusammenarbeitete, bei Dominik Klein für seine Unterstützung in den vergangenen Monaten, und bei Dr. Robert Henjes, Dr. Daniel Schlosser, David Hock und Florian Wamser für die vielen fachlichen, manchmal hitzig geführten Diskussionen. Danke auch an Steffen Gebert der mir JSON näher gebracht hat, Matthias Hirth und Christian Schwartz für Ihren Support bei Server- und Programmierfragen, Michael Jarschel, der mir administrativ immer zur Seite stand, Matthias Hartmann für die Tipps zu Photoshop, Dr. Andreas Binzenhöfer für Spaß und gute Laune, Valentin Burger für die unzähligen privaten Kitestunden und Dr. Rastin Pries für meine neue Leidenschaft, dem Mountainbiken.

Weiterhin möchte ich Gisela Förster und Alison Wichmann für Ihre Unterstützung bei der Verwaltungsarbeit, insbesondere bei der Unterstützung bei den Projekten EuroFGI und Euro-NF, danken. Ein Dankeschön gebührt auch meinen Diplomanden Andreas Blenk, Michael Lang, Moritz Mohrmann, Jochen Prokopetz, Dirk Rauscher und Michael Seufert, meinem Bacheloranden Thomas Höhn, meinen Praktikanten Hans Henjes, Steffen Gebert und Christian Siebert sowie meinen Hiwis Bastian Blößl, Kathrin Borchert, Benjamin Kleine, Stanislav Lange, Sven Scheuring, Christian Siebert, Tristan Tzschichholz und Andreas

Wendl.

Schließlich möchte ich auch Dankesworte an meine Freunde und Familie richten, allen voran meinen Eltern Renate und Josef, die mir das Informatikstudium ermöglichten und mich in all den Jahren stets unterstützten. Meinen tiefsten Dank möchte ich meiner Partnerin Sabrina Flurschütz aussprechen für ihren Rückhalt und ihr Verständnis, wenn es mal wieder einen Termin einzuhalten galt.

Contents

1 Introduction

Internet applications are becoming more and more flexible to support diverge user demands and network conditions. This is reflected by technical concepts, which provide new adaptation mechanisms to allow fine grained adjustment of the application quality. A typical example for this is video streaming. It starts with a simple transmission of pictures in a row and gets more and more sophisticated, e.g., by temporal motion compensation techniques. Motion vectors, which describe the difference from previous pictures, are transmitted in order to reduce the required network resources. New video codecs allow to adjust the video quality with respect to the available network and device resources. Scalable video codecs consists of a base layer providing a basic video quality and optional enhancement layers which allow the seamless switch to better qualities, if these layers are available. Thus, a scalable video stream can easily be combined with new network technologies as for instance multipath transmissions.

However, these flexible applications are running on top of an Internet architecture that was not designed for their requirements. In particular, high quality video streaming or cloud gaming may have higher requirements than network user devices can provide. Thus, techniques are required which enable a flexible adaptation of the data stream to meet user, application, and network demands. Consequently, several research projects like G-Lab [27] or GENI [28] work towards new mechanisms and technologies for the future Internet.

Within network and transport, new technologies evolved during the last years providing a more flexible and efficient usage of data transport and network resources. One of the most promising technologies is *Network Virtualization* (NV) which is seen as an enabler to overcome the ossification of the Internet stack [29].

NV promises to overcome the limitations of the current Internet and its protocols [30]. It provides means to simultaneously operate multiple logical networks on a single physical substrate. Physical resources can be easily added to virtual networks to cope with high demands or removed to reduce operational expenditures during times of low workloads. This enables a flexible and efficient allocation of network resources based on the demands of specific virtual networks. Further, NV allows to create networks with their own application-specific naming, topology, routing, and resource management mechanisms. Accordingly, these virtual networks can be designed and operated with respect to application demands, that means, virtual networks can be designed as *application-aware networks*.

The underlying physical resources have to be operated in an efficient way. New transport mechanisms like multipath transmission on the network and transport layer aim at an efficient usage of available transport resources. In particular, this includes the transmission of data concurrently via different transport paths in different physical and virtual networks. Prominent examples of such techniques are discussed within the Celtic project MARCH [31], implemented by SKT [32], or standardized by the IETF, like Multipath TCP (Mptcp) and the concurrent multipath transfer extension for SCTP (SCTP-CMT). This thesis investigates multipath transmission beyond technical implementation details to understand general aspects of this technology.

However, even the efficient utilization of network resources and intelligent transport mechanisms using multiple connections concurrently may not provide enough resources to meet application requirements. To overcome these limitations, services have to adjust their demands to the available network resources in order to provide the best possible user perceived application quality, denoted to as *Quality-of-Experience* (QoE) [33]. In particular, this is of high importance for application with stringent requirements like high-quality video streaming or cloud gaming. Accordingly, such applications have to use the available resources efficiently with respect to the user perceived quality and to avoid an unnecessary waste of network resources. This requires a continuous QoE management of the

video streaming service including the monitoring of the current network state as well as control mechanisms to dynamically adapt the video system to deliver the optimal QoE. Typically, the required bandwidth of a video stream depends on the frame rate, the video quality, and its resolution. The dynamic adaption of these parameters at the involved devices and within the network can be achieved for instance by the scalable extension of the state of the art video codec H.264/AVC. Applications monitoring the network and adapting to the current state are called *network-aware applications*.

The combination of both approaches may be beneficial for both sides, the network and the application. On the one hand, network resources can be utilized more efficiently and on the other hand, applications can easily adapt their demands to the available resources. In this work we detail the management of application-network interaction for the case of multipath video streaming using the scalable video codec H.264/SVC and show the potential of such an interaction.

1.1 Scientific Contribution

This monograph details the cooperation between applications and networks in the future Internet. First, application-aware networking, in particular concurrent multipath transmissions, is addressed. Second, network-aware applications for the example of scalable video streaming services are detailed. Finally, we bring both aspects together and investigate the potential of application-network cooperation in the future Internet.

Figure 1.1 gives an overview of the contribution of this work. The individual research studies carried out during the course of this work are classified according to their used methodology. In particular they are classified with respect to practice oriented methods like measurements, objective testings, and proof-of-concept implementations, or theory-oriented methods like simulations, analysis, and design guidelines. The respective focus of the studies cover multipath transport, QoE management for video streaming and multipath video streaming &

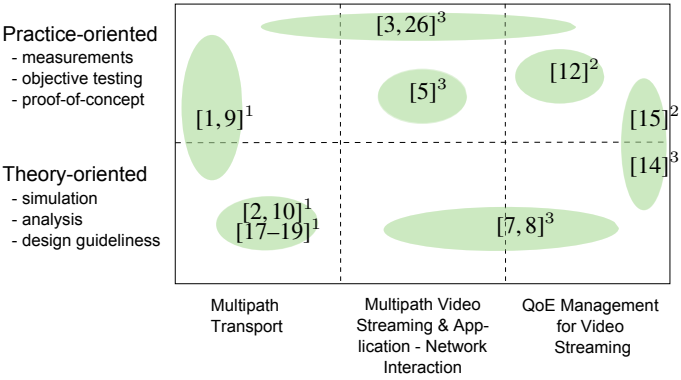


Figure 1.1: *Contribution of this work illustrated as a classification of the research studies conducted by the author. The notation $[x]^y$ indicates that the scientific publication or software demonstrator $[x]$ is discussed in Chapter y of this monograph.*

application-network interaction. The markers $[x]^y$ indicate the scientific publications and demonstrations $[x]$ which provide the basis for Chapter y.

The first part covers application-aware networks with focus on concurrent multipath transmissions, a technology that brings advantages to networks like higher throughput and increased resilience. However, multipath transmission inevitably introduces out-of-order packets due to different packet delays on the utilized paths. The reordering can be compensated by buffering at the destination, possibly leading to increased end-to-end delay. We study these effects by designing theoretical models for the buffer occupancy and the perceived delay. These models are validated by measurements in experimental facilities. Based on our findings, dimensioning guidelines for the re-sequencing buffer can be derived. Additionally, a pre-buffering mechanism is introduced and evaluated. It is shown that this mechanism reduces the buffer occupancy significantly while not increasing the perceived delay. This mechanisms illustrate how network resources can be optimized with respect to application requirements.

In the second part, a QoE control mechanism for video streaming based on scalable video coding is developed. Therefore, we investigate the video codec H.264/SVC, which permits an adaptation of the video quality to network conditions by reducing either frame rate, resolution, or image quality. We introduce a test method for video quality testing, which allows an investigation of different types of video distortion caused either by packet loss or video quality reduction and analyze different video clips. Our findings confirm that insufficient network resources, which lead to packet loss, have a severe influence on the video quality. Accordingly, we quantify this influence and derive thresholds how much packet loss can be tolerated. Further, it is shown that reducing the video quality of the application in a controlled way has a lower impact on the video quality than uncontrolled packet loss in the network. Our evaluations highlight that lowering the resolution leads to a better video quality compared to lowering the image qualities and the frame rate. In case of insufficient network resources, the resolution should be reduced first, then the image quality. A reduction of the frame rate should only be done if no other option remains. These guidelines can be implemented by a network-aware video streaming application.

The third part of this monograph presents the potential of a combination of both aspects, the cooperation of applications and networks. Consequently, it combines the findings presented in the first and second part. It presents the QoE-Provisioning Delivery Hysteresis (PDH), which describes fundamental relationships between QoE and Quality of Service (QoS) parameters, and investigates whether the hysteresis can be applied to video streaming using H.264/SVC. The hysteresis comprises the provisioning part, which describes the impact of the provisioned resources such as link capacity, and the delivery part which depicts the impact of delivery failures due to congestion, such as packet loss. Our findings reveal that video streaming using scalable video codecs follows the QoE-PDH. However, the implementation of the hysteresis without a closer application and network interaction is not enough for optimizing the user perceived quality and the network utilization. An application experiencing packet loss might interpret this as congestion and try to reduce the network load by limiting its demands. But

this does not necessarily reduce the packet loss rate which might be caused by other circumstances and thus does not increase the QoE. It might be better to react in a different way to the occurring packet loss, for instance by additional protection mechanisms like forward error correction. Based on the previous results, we illustrate the potential of a closer interaction of applications and networks in terms of user perceived video quality. We validate our findings by implementing a software prototype, which is integrated into the future Internet architecture of the COntrol and Management of COexisting Networks (COMCON) project. This demonstrates that the PDH principle is applicable to the future Internet and thus increases the significance of the presented results.

1.2 Outline of This Thesis

The organization of this monograph is shown in Figure 1.2. Each chapter contains a section that shows the background and related work of the covered aspects and summarizes the lessons learned. The three columns cover from left to right (1) the problems and challenges, (2) the algorithms and mechanisms to cope with them, and (3) the impact of the applied mechanisms on the performance and the results derived. The arrows between the sections show their relation, background and findings that are used in later sections. The section numbers of the building blocks are given in parentheses.

The remainder of this thesis is organized as follows. Chapter 2 covers multipath transmissions in general. First, we introduce a future Internet architecture based on scalable routing overlays. This architecture is able to pool available transport resources and to perform concurrent multipath transmissions. For investigating the concurrent multipath transmission and the amount of reordered packets, which inevitably occur, theoretical models are developed. We show the applicability as well as the limits of these models by measurements in experimental facilities and a comparison of practical and theoretical results. We take a look at the influence of different stochastic path characteristics and show their impact on the reordering issues. Based on these results, we propose and evaluate

mechanisms that reduce the amount of reordered packets.

Chapter 3 focuses on the scalable extension H.264/SVC of the state-of-the art codec H.264/AVC. In particular, we develop a QoE control mechanism for SVC video streaming, which is based on objective video quality metrics. We first investigate the impact of packet loss on video streaming. Based on the results, guidelines are derived how much packet loss can be tolerated without affecting the user perceived quality. In the second part of Chapter 3, the impact of the different video quality reduction possibilities of the SVC codec on the user perceived quality is analyzed. Finally, we introduce guidelines how to adjust the video quality if necessary.

In Chapter 4 we investigate the interaction of application and network to improve the user perceived quality of SVC based video streaming. We present the PDH, which describes fundamental relationships between Quality of Service (QoS) parameters and Quality of Experience (QoE). In addition, we show the applicability of the hysteresis to SVC video streaming. Based on the results we discuss a better interaction of application and networks in the future Internet and evaluate the gain in terms of user perceived quality. Additionally, we implement a multipath SVC video streaming application which is embedded in a future Internet architecture. This implementation can maximize the user perceived quality either by providing additional network resources if possible or reducing the service quality if necessary.

Finally, this monograph is concluded in Chapter 5 by a summary of the presented results and achievements.

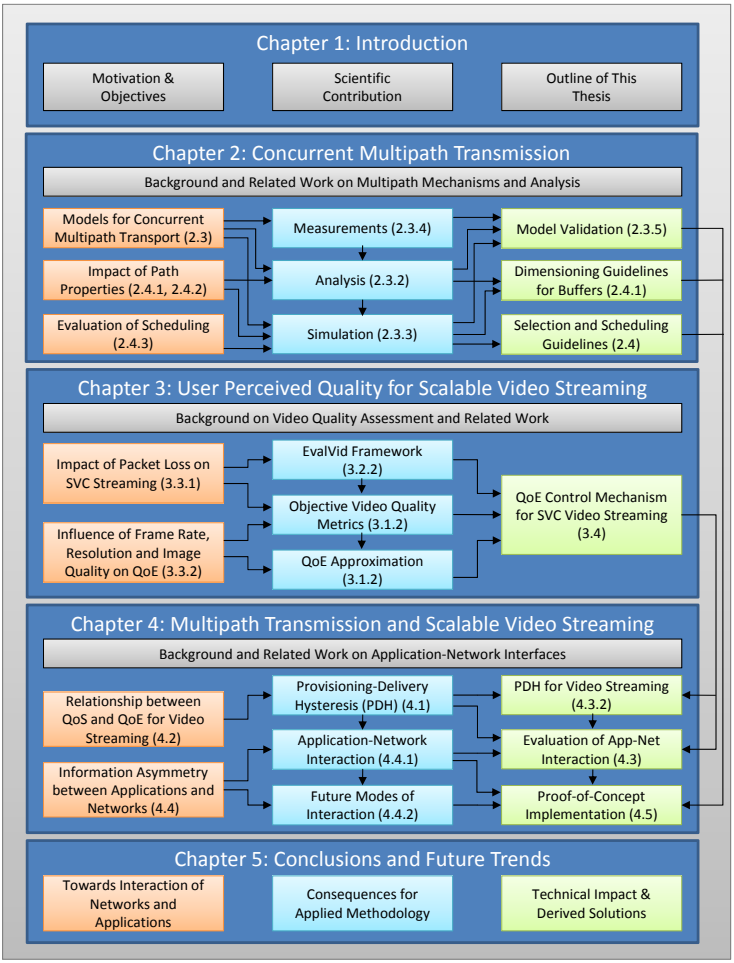


Figure 1.2: *Organization and contribution of this monograph according to problems and challenges, algorithms and methodologies, and impact of the applied mechanisms on the investigated systems.*

2 Concurrent Multipath Transmission

Multipath transmission techniques and protocols on transport layer have gained increasing interest in standardization and research. This trend is mostly driven by end user devices like smartphones, tablets, or netbooks which are able to connect to the Internet via different network technologies. The most popular protocols currently under development are Multipath-TCP (MPTCP) [34] and Concurrent Multipath Transmission-SCTP (CMT-SCTP) [35]. The main benefits of these state-of-the-art transport protocols are their potential for bandwidth aggregation, increased reliability, load balancing, and joint congestion control [36].

All these multipath transport concepts have one thing in common. They logically build a new, pooled transport resource out of several transport resources, typically with different resource characteristics like capacity or transmission delay. In future networks, such concepts might be transparent to network or transport layer protocols as proposed by the concept of Transport Virtualization (TV) [17] or possible extensions of MPLS [37].

However, the impact of the dissimilarity of diverse used resources on the pooled transport resource is a quite fundamental problem. Different latencies on the paths introduce packet reordering possibly leading to out-of-sequence arrivals at the destination. This can have a severe impact on applications and transport protocols. Applications which are not able to cope with this unordered packet stream experience information loss leading to a reduced service quality. Furthermore, transport protocols like TCP which implement congestion control mechanisms adapt the sending rate in case of sequence gaps in the packet stream. This

feature helps to ensure fairness in case of congested networks. However, in case of a multipath transmission this unnecessarily reduces the performance of the data transport. Also the selection of paths, in case of multiple available paths, may have a huge impact on the transport performance. Typically, a packet experiences a delay [38], which consists of propagation and transmission delay. Thus, the mean packet delay on a path is an initial candidate as selection criteria. However, the interaction between concurrent paths is expected to be complex and thus has to be investigated in detail.

A possible solution is to introduce a resequencing buffer,e.g., at the destination, which reconstructs the packet order and thus avoids the problems mentioned above. However, the impact of different packet delays on the involved paths on the system has to be investigated in order to deduce its influence on the buffer occupancy and the end-to-end delay. This motivates the design of performance models which can be used for a broad analysis of the system including the impact of different path characteristics or scheduling mechanisms on the multipath transport. Such studies permit a deep understanding of the system and enables the derivation of resource selection strategies parameters and dimensioning guidelines.

The reminder of this chapter is structured as follows. Related work like state-of-the-art multipath transport mechanisms and evaluation strategies for multipath transmissions are briefly discussed in Section 2.1. We detail our proposed multipath transport architecture based on Transport Virtualization in Section 2.2. For the transmission mechanisms we propose in Section 2.3 theoretical models, which are validated with a prototype implementation of the mechanism. Accordingly, we investigate the impact of different input parameters, e.g., scheduling mechanisms, on the system performance in Section 2.4. Section 2.5 details the lessons learned and discusses the impact of the results gained for Transport Virtualization on other multipath transport mechanisms.

2.1 Related Work

This section discusses related work on the analysis of concurrent multipath transport and mechanisms currently investigated by researchers and standardization bodies.

2.1.1 Analysis of Concurrent Multipath Transport

Disordering in networks can be investigated using typical queuing models like a $GI/GI/n-m$ system with first-in-first-out scheduling. Hereby, GI (general independent) are arbitrary arrival and service processes, n is the number of service entities and m the queue length. For $n > 1$ packet reordering may occur due to different processing times of the packets. Different realizations of this abstract queuing models have been investigated in literature. Xie et al. [39] for instance model the network as a $D/GI/\infty$ with random service time distributions. With the model the authors compute the waiting time and resequencing buffer occupancy for exponential and pareto delay distributions. Further approaches are for instance an $M/G/\infty$ model [40] or an $M/M/K$ model [41].

Due to the high abstraction level of the proposed models they can not be applied to our investigated system. One of the aims of our investigations is to investigate path selection strategies which means that we want to investigate the impact of diverse path delays on the described system. However, the models in literature assume equal path delays.

Recently, a framework for systematic evaluation of protocol performance with respect to out-of-sequence packet arrivals has been proposed [42]. This framework enables worst-case estimations on the amount of out-of-sequence arrivals with respect to a predefined network delay distribution. We choose a different approach since we evaluate the resequencing for arbitrary measured network delays and derive selection strategies. This allows a direct integration of the evaluation framework in the proposed architecture in Section 2.2. This architecture using conccurent multipath transmission can be easily adapted to varying path qualities

if the service parameters of the paths are monitored by a measurement entity like e.g. a passive multihop packet tracking tool [43].

2.1.2 Multipath Transport Mechanisms

This Section presents a short overview of different multipath transmission techniques and highlights the similarities of our proposed multipath mechanisms and other discussed approaches in literature.

Parallel and Multipath TCP Approaches

Extending TCP for the usage over multiple paths was proposed in research several times, cf. [44–47]. However, these proposals have not achieved practical relevance which might be due to the lack of an implementation running across the Internet. Currently, a new extension of TCP for the usage over multiple available paths is discussed within the IETF working group MPTCP [34]. It enables the usage of multiple paths and ensures fairness in case of shared bottlenecks. This is achieved by a coupled congestion control of the single TCP connections over each path, cf. [48]. Different implementations of Multipath TCP were discussed, for instance Multipath TCP (MPTCP) [49] or Multi-Connection TCP (MCTCP) [50]. However, these approaches mainly discuss implementation details on kernel level and do not change the coupled congestion control schema as presented in [51]. Recently, the first running kernel implementations were published, cf. [52, 53]. Performance questions how different path characteristics like bandwidth and packet end-to-end delay influence the overall performance are not discussed in detail yet.

However, the investigated models can not be used directly for answering performance question for the MPTCP case. The different implemented congestion control of MPTCP and packet retransmissions are not part of the model. Thus, further studies have to investigate to which part the proposed models can be used to investigate the performance of MPTCP and how the models have to be extended.

SCTP-CMT

The idea to use SCTP-multihoming for transmitting data concurrently via different paths was published in 2004 by Iyengar et al. [54]. An evaluation of the presented SCTP extension in [55] reveals problems occurring in case of multipath transmissions. In order to solve these problems, the authors introduce additional mechanisms like buffers at the sender. However, for the case of dissimilar path characteristics like latencies and available bandwidth, buffer blocking at the sending and receiving buffer occurs and has to be solved, as discussed in [56,57]. Currently, the standardization of SCTP-CMT is discussed within the IETF [35].

SCTP implementations can provide two data transmission modes, a reliable and a partially reliable transport [58]. For both mechanisms the same congestion control schemes are adopted. However for the partially reliable case no retransmission mechanisms is applied. As in the MPTCP case, the presented models can not be applied directly for investigating the performance of SCTP. Consequently, the application of the models has to be investigated in further studies and, if possibly, enhanced for the usage with SCTP.

Traffic Partitioning in a Multicast-Multipath MPLS Network

Hesselbach et. al [37, 59] introduce multipath load balancing for MPLS networks. The incoming traffic is split at the ingress node and sent via multiple label switched paths to the egress node. In order to avoid packet reordering due to different delays of the involved paths, the authors propose to introduce a buffer at the egress node and present dimensioning guidelines for constant path delays. According to the authors, pre-buffering at the ingress nodes could be used to reduce buffer utilization at the egress node.

Due to the multipath mechanisms beneath the IP layer, no interfering control loop as for the transport protocols exist. Thus, the presented models can easily be applied to MPLS load balancing and proper dimensioning guidelines for the resequencing buffer can be derived. Further, the results for pre-scheduling can also be applied.

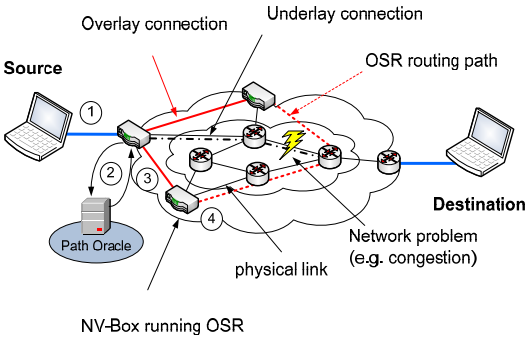


Figure 2.1: *Routing overlay using one-hop source routing.*

2.2 Multipath Transport Architecture Based on Transport Virtualization

This section introduces a Future Internet architecture based on scalable routing overlays. Routing overlays offer an appealing approach to enhance path selection and to work around slow BGP convergence. They are virtual network structures dedicated to forward arbitrary data and thus enhance regular Internet routing. Various architectures of routing overlays have been proposed recently [60, 61]. A highly promising approach is the concept of *one-hop source routing*. Hereby, the user data is forwarded to a specific intermediate node which then relays the traffic to its destination using ordinary IP routing. The dedicated forwarding can be achieved by establishing a tunnel to the intermediate node. The advantage of one-hop source routing is the easy control of performance by selecting an appropriate intermediate node while still being scalable. The question how to select such intermediate nodes and when to switch between different available nodes, i.e., how to control and improve the performance of the data transport, is discussed in this chapter.

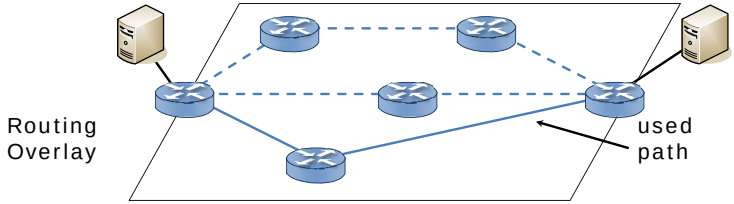


Figure 2.2: *Selection of one path out of several possible paths.*

2.2.1 A One-hop Source Routing Architecture

The discussed routing overlay architecture in this section is based on Network Virtualization (NV). This technique enables the simultaneous operation of multiple logical networks on a single physical platform. It permits distributed participants to almost instantly create their own network with application-specific naming, topology, routing, and resource management mechanisms. Based on these features, NV technologies are expected to be one of the major paradigms of the Future Internet as proposed by many international initiatives, e.g., GENI [62] or G-Lab [27]. In particular, NV can be used to create a one-hop source routing architecture as suggested in [38, 63, 64]. The architecture is depicted in Figure 2.1. It applies edge-based NV-boxes which can execute *virtual router software*. These software routers can accept incoming traffic from tunnels and forward this traffic to the destination using conventional IP routing protocols. When a source wants to send data with controlled performance, cf. Step 1 in Figure 2.1, it signals this to an NV-box running the *One-hop Source Router (OSR)* software. When an OSR router receives such a signal, it asks a *Path Oracle* to provide it with the address of an intermediate node which can forward this data in the required way, cf. Step 2 in Figure 2.1. Subsequently, the ingress OSR establishes a tunnel to the selected intermediate OSR, cf. Step 3 in Figure 2.1. Finally, the intermediate OSR inserts the traffic into the conventional IP routing process. This architecture shows a separation of the former monolithic IP system into two overlays, one for signaling

and one for data forwarding. These overlays can be structured and equipped with routing mechanisms according to their specific function. The architecture further allows the selection of one ore more paths out of several available paths and the pooling of these paths into on virtual transport resource. This issues are detailed in the next subsection.

2.2.2 Path Selection and Resource Pooling

First, let us consider ordinary path selection. In this case, a single path out of a set of candidate paths is selected, as depicted in Figure 2.2. The path characteristic has a direct impact on reliability, security or on QoS parameters between source and destination of the transmission. Therefore, a selection function has to exist which enables the appropriate choice of the used path. For instance, the different resources (i.e. paths) could be classified by path delay or the bandwidth. Moreover, the concurrent usage of different paths might have appealing advantages. The reliability of the end-to-end connection is increased and the overall throughput might be higher if concurrent paths are used for the data transmission. This is illustrated in Figure 2.3. In this example two paths are pooled and appear as one virtual link. However, there has to be a mechanism distributing the data among the used paths and selecting the appropriate set of concurrent paths. This feature will be discussed in Section 2.3.1. The set of used paths can consist of paths offered by different Internet or overlay providers. Consequently, the data transport is not bounded to one of these providers anymore and it is possible to choose the cheapest or fastest resources and pool them together to a high capacity pipe. The pooling of resources out of different networks is shown in Figure 2.4. Transport Virtualization achieves the pooling by a mechanism that is placed on nodes that have access to multiple paths or networks, e.g. end hosts which are multi-homed or routers running exterior gateway protocols. At the source of the transmission, the mechanism decides which resources, i.e. which paths, are selected, whether they are used preclusive or concurrently, and how the data is distributed among these paths. If necessary, the split data is recombined at the destination to a con-

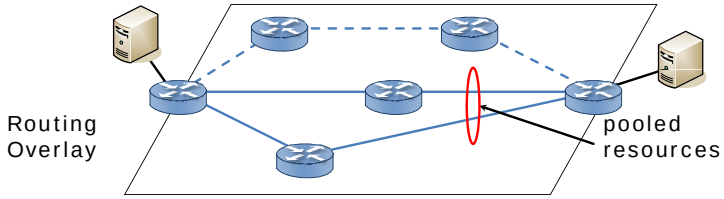


Figure 2.3: *Selection of a couple of paths out of several possible paths.*

sistent flow. Due to the ability to choose between different transmission options, TV achieves an independence from a specific physical transport resource. This independence increases if TV is used in combination with *Concurrent Multipath (CMP)* transmissions. Even if one of the used resources fails, TV using CMP may be able to continue the transmission service.

Other advantages of CMP transmissions are higher capacity due to parallelism, avoiding head-of-the-line blocking, spatial separation of control and data connections, differentiations between traffic types, and leveraging multiple processors in routers.On the downside, CMP transmissions introduce additional overhead and complexity. For instance, different path delays on the utilized paths may lead to packet reordering of the pooled resource. The reordering can be compensated by buffering at the destination, possibly leading to increased end-to-end delay but still being transparent to the data transport. In Section 2.3, we introduce a performance model for the multipath transmissions which allows us to investigate the impact of the packet delay on this buffer and on the perceived end-to-end delay.

2.2.3 Availability of Concurrent Paths

A major concern for the applicability of TV is the availability of different transport resources. Figure 2.5 depicts the topologies of four North-American Tier 1 network operators (AS3561, AS3967, AS3356, AS6467) on Point-of-Presence (POP) level [65]. The dashed lines indicate connections between routers, whereas

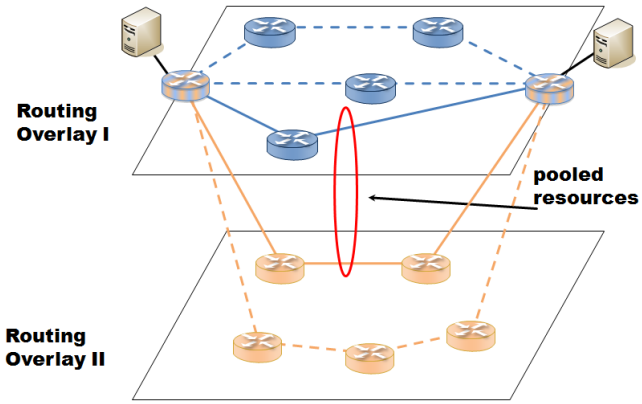


Figure 2.4: *Selection of a couple of paths out of several possible paths from different overlays.*

the horizontal lines show connections within an autonomous system. The vertical and diagonal lines indicate connections between ASes. The figure reveals that many locations have a high number of routes to arbitrary destinations. Thus, it is theoretically possible to choose among multiple routes, even from different providers. Additionally, this picture shows that a significant redundancy is present in the networks. Thus, a better exploitation of this characteristic might enhance the reliability of the system.

Moreover, measurements in PlanetLab [66] revealed that up to 25% of certain Internet routes may violate the *triangle inequality*. That means, a shorter delay can be experienced when another path through an intermediate node is used. Due to the flexibility provided by the presented source routing architecture, the routing can be adopted to use paths with lower delay. Further, it is theoretically possible to enhance the architecture and use concurrent CMP transmissions via different available paths. However, CMP will introduce additional complexity which has to be understood. CMP transmission will inevitably introduce out-of-order

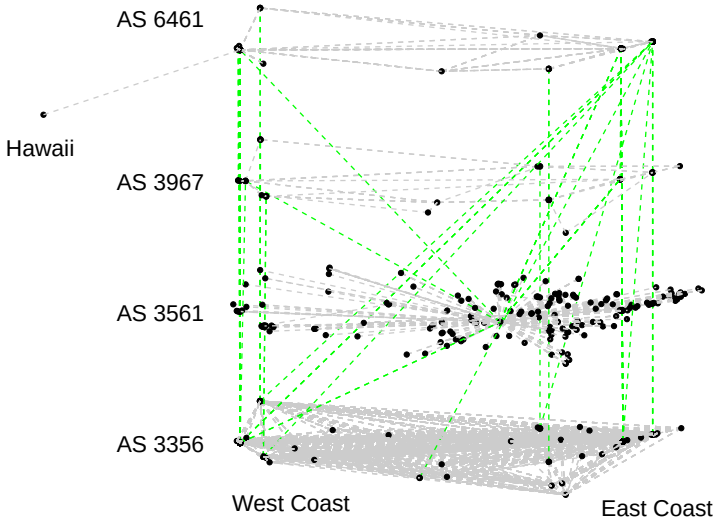


Figure 2.5: *Multiple Tier 1 Networks in the US.*

packets due to different stochastic packet delay characteristics on the paths. The reordering can be compensated by buffering at the destination, possibly leading to increased end-to-end delay. In order to evaluate this effects and decide whether concurrent multipath transmissions should be applied and which paths should be used, theoretical models are introduced in the next section.

2.3 Modeling of Concurrent Multipath Transport for Transport Virtualization

In this section, we provide models for evaluating the concurrent multipath transport system. We start with a description of the system including the influence parameters. Based on this description we explain our simulation approach and an

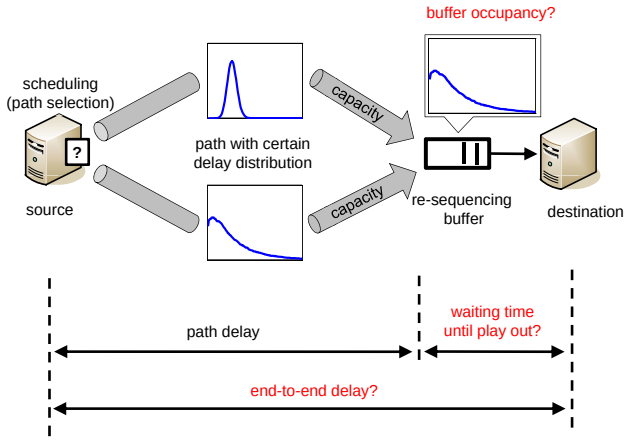


Figure 2.6: *System description.*

analytical model which allow the computation of the resequencing-buffer occupancy. Further, we validate the theoretical models with measurements in experimental facilities.

2.3.1 System Description

The system, together with the influence parameters and the performance metrics are depicted in Figure 2.6. The following input parameters are defined:

- **Capacity (in packets)** describes how many packets are transmitted per time unit over a path. In case of constant capacity, packets are transmitted e.g. every 10 milliseconds.

- **Scheduling** determines the mechanism how the packets are distributed over the different paths.
- **Path selection** describes the selection strategies on the system, e.g. take paths with low mean delay or low delay variation.
- **Path delay distribution** denotes the distribution of the delays each packet experiences. The delays on a path may vary and can be illustrated as path delay distribution / histogram.
- **Buffer size** is the size of the resequencing buffer. If the buffer is too small, packets may be dropped, an over-dimensioning leads to a waste of resources.

The above mentioned input parameters have an impact on the following performance metrics:

- **Resequencing-buffer occupancy** describes the number of packets which are stored in the resequencing-buffer.
- **Waiting time** describes the additional waiting time introduced by buffering.
- **End-to-end delay** denotes the perceived delay for each packet including its path delay and its waiting delay.

In the following, we present our evaluation methods, i.e., simulations and analytical modeling. Afterwards we validate simulation and analysis with the performed measurements and discuss the limitations of our models.

2.3.2 Analysis

Originally, the analytical model was introduced for the investigation of a many-to-one transmission scenario of parallel downloads for streaming applications by

Nebat and Sidi [67,68]. We adopted this model to the context of Transport Virtualization and validated this model with measurements in experimental facilities in Section 2.3.5. The considered model assumes a continuous data stream for the multipath transmissions over m concurrent paths. The delay on the paths is described by discrete delay distributions with a resolution of one time unit. We further consider paths with equal capacity, and that the transmission rate on each path is equal to one packet per time unit. Accordingly, control loops like the congestion control in TCP are not part of this model. The delay for different paths are independent and may follow different delay distributions. A detailed explanation of the mathematical model can be found in [67,68]. The used model ensures that no packet reordering on a single path can occur. This means that packets sent over one path can not overtake each other. To facilitate the explanation, the following notations are used.

The packets transmitted at time 0 over path $1, 2, ..., m$ are packets $1, 2, ..., m$ respectively. They are scheduled to the paths in a round-robin manner. At time t packets $1 + mt, 2 + mt, ..., m + mt$ are transmitted via the m involved paths. We further use the term *minimum valued packet (MVP)*, as introduced in [67], denoting the lowest indexed packet at time t that has not arrived at the destination by time t. For instance, if packets 1 through 5 arrived, but packet 6 did not, packet 6 is the MVP. Thus, the resequencing-buffer occupancy at time t is exactly the number of packets indexed higher than the MVP that have arrived by time t and no packet stored in the resequencing buffer was transmitted via the path of the MVP since packets transmitted on every path arrive in transmission order. We denote the index of the path of the MVP by s_n and $\delta_{X,t}$ as the time passed since the last packet received via path X was transmitted at time t. For brevity we refer to $\delta_{X,t}$ as δ_X in the following. With this notation the resequencing-buffer occupancy can be computed as

$$P(B = k) \quad = \quad \sum_{i=1}^{m} \sum_{x=0}^{\infty} P(B = k, s_n = i, \delta_i = x). \tag{2.1}$$

The right hand side of the equation denotes the buffer occupancy probability for each path transmitting the MVP and each possible value for the time passed since the last packet transmitted over this path was received. As discussed in [68], this yields to:

$$\begin{aligned} & P(B = k, s_n = i, \delta_i = x) \\ &= P\left(\sum_{j=1, j\neq i}^{m} \delta_j = (m-1)x + 1 - i - k, \delta_j < x \; \forall j < i, \delta_j \leq x \; \forall j > i \right) \\ &= \sum_{S_{i,x,k}} P_i(x) \prod_{j=1}^{i-1} P_j(l_j) \prod_{j=i+1}^{m} P_j(l_j), \end{aligned} \tag{2.2}$$

where $S_{i,x,k}$ defines the delay configuration on path i before the arrival of the MVP:

$$\begin{aligned} S_{i,x,k} = \{ & l_1, ..., l_{i-1}, l_{i+1}, ..., l_m : l_1 < x, ..., l_{i-1} < x, \\ & l_{i+1} \leq x, ..., l_m \leq x, \sum_{j=1, j\neq i}^{m} l_j = (m-1)x + 1 - i - k \}. \end{aligned}$$

With Equation 2.2, we can compute the resequencing-buffer occupancy in case of a transmission over m paths with equal transmission rate.

The analytical model has different limitations, e.g., it lacks the computation of the end-to-end delay or it does not support other scheduling mechanisms. However, the results can be computed fast and thus, it is applicable for an on-line computation and may enable an on-line path selection. To overcome the limitations of the model, we implemented a simulator which is described in the following.

2.3.3 Simulation

In this section we describe the abstract packet level simulation framework which was implemented to investigate scenarios where the analytical model is not applicable. We use a discrete time, event based simulation to study the behavior of the resequencing-buffer occupancy and the perceived end-to-end delay. Each path delay is characterized by its path delay distribution. Based on this, the packet delays are generated.

We assume that no re-ordering on a single path occuers. Thus, whenever a packet experiences a random delay on a path, the previous packet delay has to be considered. Hence, the current packet delay has always to be at least the previous packet delay minus the interim time between the previous and the current packet. Furthermore, the relative frequency of all delays on a path has to converge against the given delay distribution for that path. The packet delay generation method is described in the following.

Modeling Packet Delay

The applied packet delay method is an enhancement of the delay model introduced in [67] and extended to generate valid delay traces for the use in simulations. First, we describe the basic necessary conditions upon the delay distribution to ensure packets arrive at their destination in the same order they were transmitted. The complete derivation can be found in [67]. To facilitate the explanation, the following notations are used:

- d_i: the delay experienced by packet i.
- Δ_i^t: the inter-departure time for packets $i-1$ and i.

Consequently, there are only two restrictions for the experienced delay of packet i:

1. $d_i \geq 0$, the delay is always a positive value, and

2. $d_i \geq d_{i-1} - \Delta_i^t$, packet i can not overtake packet $i-1$. Nevertheless, they may arrive at the destination simultaneously.

Therefore, d_i can be any probabilistic function $f(d_{i-1}, \Delta_i^t)$ that satisfies

$$f(d_{i-1}, \Delta_i^t) \geq \begin{cases} d_{i-1} - \Delta_i^t, & d_{i-1} - \Delta_i^t \geq 0 \\ 0 & d_{i-1} - \Delta_i^t < 0 \end{cases}. \tag{2.3}$$

To ensure, the delay does not diverge, we need a stability constraint on $f(d_{i-1}, \Delta_i^t)$. For the special case where packets are transmitted at a constant rate, i.e., every k time units, we have a constant inter-departure time $\Delta_i^t = k \; \forall i$, and (2.3) becomes

$$f(d_{i-1}, \Delta_i^t) = f(d_{i-1}) \geq \begin{cases} d_{i-1} - k, & d_{i-1} \geq k \\ 0 & d_{i-1} < k \end{cases}.$$

The introduction of k is an extension to the model presented in [67] and enables us to adjust the path delay in a higher resolution. For simplicity, let us assume that the delay is an integer value expressed as multiples of a time unit. Consequently, $f(d_{i-1})$ can be any integer that satisfies

$$d_i = f(d_{i-1}) \geq \begin{cases} d_{i-1} - k, & d_{i-1} > k \\ 0 & d_{i-1} \leq k \end{cases}. \tag{2.4}$$

Any probabilistic function that corresponds to f in (2.4) can be represented by a Markov-chain, similar to Figure 2.7, which is an example for $k = 1$, i.e. packets are sent every time unit. Here, state i corresponds to delay of i time units and the arrows correspond to the transitions among states with the respective probabilities. This transition probabilities can be written as a transition probability matrix $\mathcal{P}$ consisting of the elements $p_{i,j}$. For a finite maximum delay d_n, the transition probability matrix can be written as depicted in Equation 2.5.

The matrix $\mathcal{P}$ consists of four parts. On the main diagonal are the probabilities to stay in current delay state. On the right side of the main diagonal are the

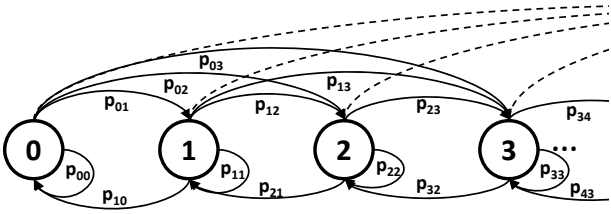


Figure 2.7: *Markov-chain representing the delay, $k = 1$.*

transition probabilities which increase the current delay state. The left side of the main diagonal illustrates a decrease in the delay state. The maximum decrease in the delay state depends on the inter-departure time.

$$\mathcal{P} = \begin{pmatrix} p_{0,0} & p_{0,1} & p_{0,2} & \cdots & p_{0,n-1} & p_{0,n} \\ p_{1,0} & p_{1,1} & p_{1,2} & \cdots & p_{1,n-1} & p_{1,n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ p_{k,0} & p_{k,1} & p_{k,2} & \cdots & p_{k,n-1} & p_{k,n} \\ 0 & p_{k+1,1} & p_{k+1,2} & \cdots & p_{k+1,n-1} & p_{k+1,n} \\ 0 & 0 & p_{k+2,2} & \cdots & p_{k+2,n-1} & p_{k+2,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & p_{n,n-1} & p_{n,n} \end{pmatrix}. \quad (2.5)$$

Thus, not all states smaller than the current state can be reached. This is expressed by $p_{i,j} = 0 \; \forall i < j - k - 1$. For a given delay distribution d we have to solve the fix-point equations

$$d = d \cdot \mathcal{P}, \quad (2.6)$$

so that the resulting transition probability matrix, and thus the represented Markov-chain is irreducible and aperiodic. That way, we ensure that the delay process is recurrent. The transition probability matrix $\mathcal{P}$ can be used in our simulation model to assure that 1) the packet delay follows the given delay distribution d and that 2) no packet reordering occurs by a transmission on a single path. Before we can use $\mathcal{P}$, we still have to solve Equation 2.6, which is described in the next subsection.

Computing the Transition Matrix by Using Linear Programming

In this section, we describe our approach to determine a transition probability matrix $\mathcal{P}$ which fulfills the fix-point Equation 2.6. This equation is underdetermined, i.e. we have to determine $\frac{1}{2} \cdot (n^2 + 2nk - k^2 + n - k)$ parameters with $n + 1$ equations, whereas $n + 1$ is the size of the delay state space d. For that, we model the problem as a Linear Program (LP) and solve this program with ILOG CPLEX [69]. In addition to the previously introduced variables, we define the following variables:

- c_1: vector with lower bounds for values on the main diagonal of matrix $\mathcal{P}$
- c_2: vector with upper bounds for values on the main diagonal of matrix $\mathcal{P}$

The LP depicted by Algorithm 1 aims to compute the transition probability matrix. This is expressed by Equation 2.7. The constraints on the variables are described in the following:

- Equation 2.8 describes the $n + 1$ fix point equations derived from Equation 2.6.
- The sum of each line in $\mathcal{P}$ has to satisfy the normalizing condition, i.e the sum over all probabilities is equal to 1, which is expressed by Equation 2.9.
- Equation 2.10 illustrates the transition probabilities in the lower left part of $\mathcal{P}$. These probabilities have to be zero, in order to ensure that the next

Algorithm 1 Determine the transition matrix.

Maximize

$$f(\mathcal{P}) = \sum_{i=0}^{n} \sum_{j=0}^{n} p_{i,j} \tag{2.7}$$

Subject to

$$\sum_{i=0}^{n} p_{i,j} \cdot x_i = x_j \ \ \forall j; \tag{2.8}$$

$$\sum_{j=0}^{n} p_{i,j} = 1 \ \ \forall i \tag{2.9}$$

$$p_{i,j} = 0, \ \ i < j - k - 1; \tag{2.10}$$

$$0 < p_{i,j} < 1, \ \ j - k < i < j \tag{2.11}$$

$$c_1 < p_{i,j} < c_2, \ \ i = j \tag{2.12}$$

$$0 < p_{i,j} < 1, \ \ i > j \tag{2.13}$$

delay value is not smaller than the current delay minus the inter-departure time at the source.

- The probabilities between the main diagonal and the zero values are depicted by Inequalities 2.11. These probabilities denote a slow-going delay decrease without causing packet reordering.
- The main diagonal of the matrix, depicted by the Inequalities 2.12, denote that the delay remains constant. In order to avoid the trivial solution of the problem, the identity matrix I, these probabilities have to be smaller than 1, which is expressed by c_2.
- An increase of the delay between the departure of two packets is illustrated by the Inequalities 2.13.

With the presented delay generation it is possible to generate arbitrary time series for given delay distributions. Thus, it is possible to compute the

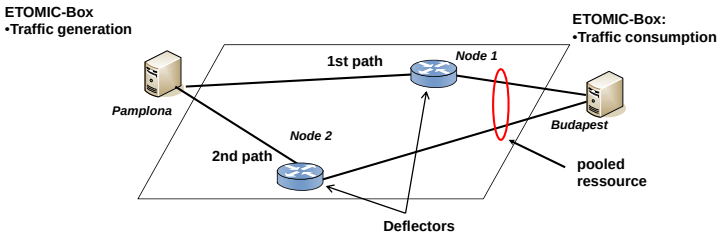


Figure 2.8: *Measurement setup.*

resequencing-buffer occupancy with the same input parameters for simulation and analysis. However, the simulations providers further features like the easy implementation of scheduling mechanisms or the investigation of the end-to-end delay.

2.3.4 Measurements

With the help of experimental facilities like Planetlab [70], Vini [71], or G-Lab [27] researchers can conduct large scale experiments. Our measurements are conducted within the European portion of the PlanetLab testbed, PlanetLab Europe (PLE). The federation of PLE with the ETOMIC [72] testbed provides access to high-precision, GPS-synchronized monitoring nodes which allow the measurement of one-way packet delays. The measurement setup is depicted in Figure 2.8 and consists of a source, a destination, and two paths between source and destination. The packet forwarding was implemented by application layer overlay routing. For measuring the one-way delays between source and destination, we relied on ETOMIC nodes with DAG cards. These nodes provide high precision GPS synchronized timestamps with an accuracy of nanoseconds, cf. [72]. As source and destination node we used the ETOMIC nodes located in Pamplona and Budapest. The application layer packet forwarding was realized on PlanetLab hosts located in Ireland, China, Canada, and Brazil.

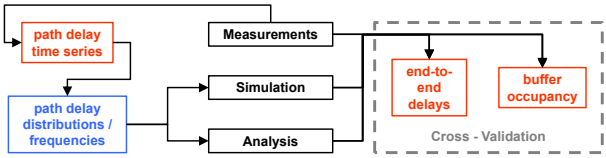


Figure 2.9: *Validation methodology.*

2.3.5 Model Validation

This section highlights how the theoretical models were validated. For that, we conduct measurements of one-way path delays and the resulting buffer occupancies in the Onelab facility as discussed in Section2.3.4.

Figure 2.9 shows the applied method. With a prototype implementation we measure the one-way delay for each packet and log the resequencing-buffer occupancy. We use the one-way delay timeseries to validate the simulator by computing the resequencing-buffer occupancy distributions and the perceived end-to-end delays. Further, we compute the path delay distributions, compute the resequencing-buffer occupancy with the analytical model, and compare the results with the measurements. We will use the terms probability function and cumulative probability function in the course of this work, although random variables are empirically measured.

We investigate three different multipath transmission scenarios, each with two paths.

- **Brazil-Ireland** respectively via nodes located at RNP - Para and Waterford Institute of Technology. The purpose of this experiment is to investigate the system behavior for a setup with a low and a high delay path, i.e. the path characteristics are very different.

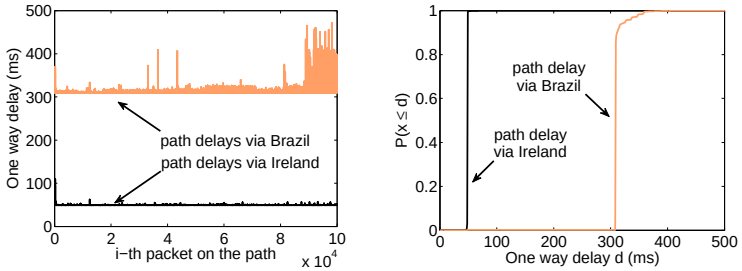
- **China-Canada** respectively via nodes located at PLA University of Science and Technology and University of Waterloo. We choose this setup

for evaluating the system with two paths with similar delays, i.e. a very homogeneous scenario.

- **Brazil-Brazil** respectively via nodes located at RNP - Para and RNP - Rio de Janeiro. In this scenario we also expect very similar delays for both paths. However, since we sent packets from Europe via two nodes in Brazil the packets are sent partly via the same link. If this is a bottleneck link we can observe the influence of a shared bottleneck on the multipath transmission mechanism.

For each scenario, we conducted 14 experiments and transmitted 100.000 packets per path, that means 200.000 packets in total, with constant inter-departure times of 10 ms. Each packet contains a unique id. Thus, packets with odd ids were sent via the first path and packets with even ids via the second path. In case of lost packets on a path, e.g. packet 7 arrives after packet 3 on the first path and hence packet 5 is missing, we play out all packets in the buffer including the last successfully arrived packet, i.e., packet 7. This is a valid approach since we can confirm with our measurements that no packet reordering on a single path occurred.

We start with illustrating typical time series and typical path delay distributions of the first scenario, a multipath transmission via Brazil and Ireland. After that, we discuss the path delay distributions for the other scenarios. Results for the first scenario, Brazil-Ireland, are depicted in Figure 2.10. The experienced one-way delay for each packet, i.e. the delay time series, is depicted in Figure 2.10(a). It can be seen that the one-way delay of packets routed via Ireland is significantly lower than the one-way delay of packets routed via Brazil. Further, the delay variation on the path via Brazil is higher since packets experience delays between approximately 310 ms and 420 ms. The results are also depicted as cumulative frequencies in Figure 2.11(b). We see that about 90% of the packets on the path via Brazil experience a similar one-way delay of around 310 ms and that the rest of the packets experience higher delays. In contrast, the path via Ireland is rather constant and the delay gap between both paths is around 260 ms. Due to the



(a) One-way delay time series for each packet.

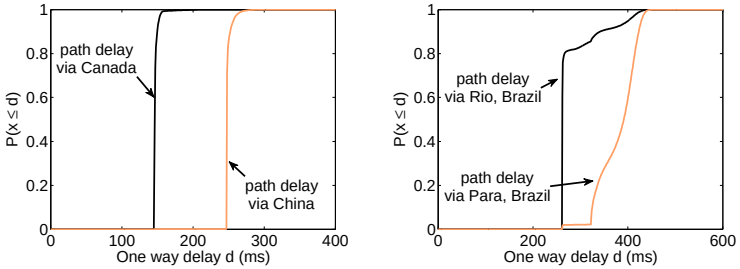
(b) One-way delay cumulative frequencies

Figure 2.10: *One-way delay measurements for Brazil - Ireland.*

constant inter-departure time of 10 ms of packets on each path, we expect that the buffer is always filled with more than 20 packets for this scenario. Detailed investigations of the buffer are shown later.

The cumulative one-way delay frequencies for the other scenarios are depicted in Figure 2.11. As explained, we chose the multipath scenario via China - Canada due to the expected similar one-way delays. The measurements are illustrated in Figure 2.11(a). It can be seen that the gap between the path delay frequencies is around 100 ms, smaller as in the Brazil - Ireland scenario. Thus, we expect a smaller minimum number of packets which are always stored in the resequencing buffer. Further, it can be seen that the delay variation is small for both paths. The one-way delay frequencies for the third scenario with both paths via Brazil are depicted in Figure 2.11(b). It can be seen that the one-way delays for the path via Rio are shorter than for the path via Para. Further, the delay variation on each of the utilized paths is higher than in the other scenarios.

Next, we investigate the resequencing-buffer occupancy for the different scenarios. In general experiments contain measurement errors, due to complex operation systems and hardware. For instance in our measurements packets are not always transmitted every 10 milliseconds. Every once in a while a packet is not



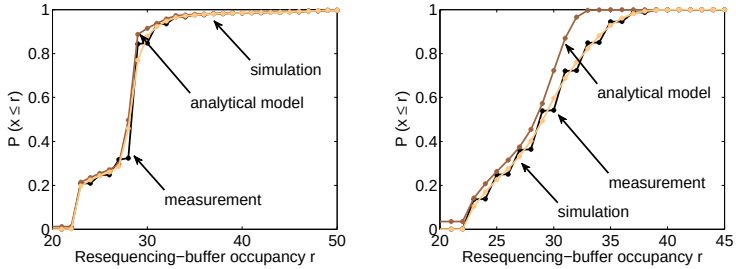
(a) China - Canada (b) Brazil - Brazil

Figure 2.11: *Measured cumulative one-way delay frequencies for the other scenarios.*

transmitted when it should, but two consecutive packets are sent at the next transmission time. To take this into account we show for each scenario an experiment with a close match of measurements, simulations and analysis and the experiment with the worst match, i.e. the biggest deviation between measurements, simulations and analysis. For describing the goodness of fit between measurement versus analysis and measurement versus simulation we computed the mean squared errors (MSE) between the curves.

Brazil - Ireland

The first scenario to discuss is for a path via Brazil and one path via Ireland, i.e. very dissimilar delays. The results for two experiments, one with a good match of the models and one with the worst match, are depicted in Figure 2.12. It can be seen that due to the high delay difference of the used paths, the resequencing buffer is always filled with at least 20 packets. The closest match between models and measurements according to MSE is depicted in Figure 2.12(a). The computed MSE between measurements and simulation $MSE_{sim} \approx 0.0004$ and between measurements and analysis $MSE_{ana} \approx 0.0007$. The differences be-



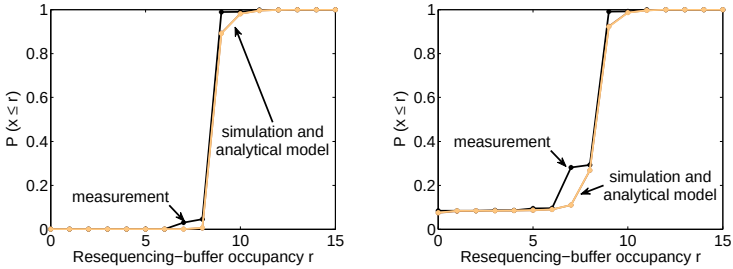
(a) good match between measurements and models.

(b) worst match between measurements and models

Figure 2.12: *Resequencing-buffer occupancy for measurements, simulation and analysis for the case Brazil-Ireland.*

tween measurements and simulations can be explained by the fact that for the measurements the inter-departure times for the packets on a specific path are not constant but varying slightly. This is not reflected by the simulation. Further, as explained above, the data transmission stalls several times: packets are not transmitted when they should, instead later piggy-backed with other packets. The analysis uses the histogram as input parameter and not the time series which constitutes a strong abstraction and thus, explains the gap between measurements and analysis.

The scenario yielding the highest MSE and thus to the highest difference between measurements and the models is depicted in Figure 2.12(b). However, the simulation still matches the measurements pretty good, and that analysis is also a good match for the measurements. The computed MSE between measurements and simulation $MSE_{sim} \approx 0.003$ and between measurements and analysis $MSE_{ana} \approx 0.004$.



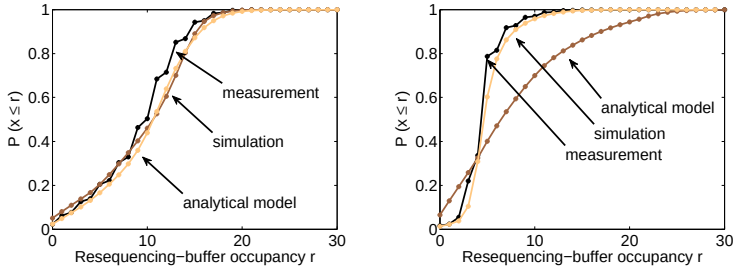
(a) good match between measurements and models.

(b) worst match between measurements and models

Figure 2.13: *Resequencing-buffer occupancy for measurements, simulation and analysis for the case China-Canada.*

China - Canada

In the second scenario, we investigate a path via China and a path via Canada, i.e. two disjoint paths with high delays. Again, we show the results for a scenario with a small MSE, i.e., a good match between measurements, simulation and analysis, and for a scenario with a high MSE, i.e., a worse match. The results are depicted in Figure 2.13. For this scenario, the buffer occupancy is lower than in the previous scenario. That is due to the smaller difference in path delays between both paths. The computed MSE between measurements and simulation $MSE_{sim} \approx 0.0002$ and between measurements and analysis $MSE_{ana} \approx 0.0002$ are illustrated in Figure 2.13(a). In contrast, the computed MSE between measurements and analysis or simulations is higher for the experiment depicted in Figure 2.13(b), respectively $MSE_{sim} \approx 0.001$ and $MSE_{ana} \approx 0.001$. This can again be explained by the effects occurring in measurements which are not reflected by the simulation and the analysis. However, the theoretical models are still very close to the measurements.



(a) good match between measurements and models (b) worst match between measurements and models

Figure 2.14: *Resequencing-buffer occupancy for measurements, simulation and analysis for the case Brazil-Brazil.*

Brazil - Brazil

In the third scenario, we examine two paths in Brazil: one via Rio de Janeiro, the other via Para, i.e. similar delays and a shared path. The results are depicted in Figure 2.14. As can be seen in both subfigures, the buffer occupancy is mostly very small which is due to the similar delay magnitudes on the paths. The experiment with the lowest MSE is illustrated in Figure 2.14(a), with $MSE_{sim} \approx 0.002$ and $MSE_{ana} \approx 0.001$. The MSE is higher as in the previous scenarios, but the models still fit the measurements very well. The experiment with the worst match for this scenario is depicted in Figure 2.14(b). Although the simulation is very close to the measurements, as also indicated by the $MSE_{sim} \approx 0.002$, the difference between measurements and analysis is very high. This is also reflected by the MSE which is much higher, $MSE_{ana} \approx 0.015$. Since this difference cannot be explained by minor modeling inaccuracies, we investigate the results in more detail.

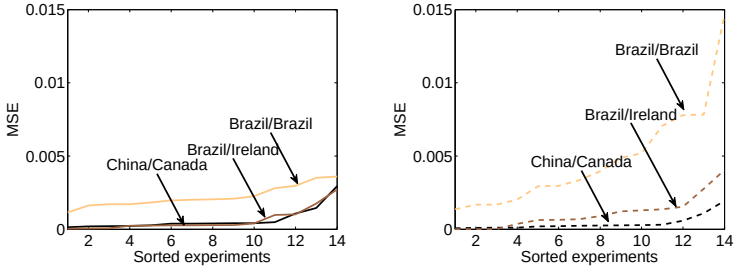
It turns out, that for this experiment the delays between the two paths are correlated. In case of a high delay value on path one it is very likely that the delay

value on path two is also very high and vice versa. In case of uncorrelated path delays, a high delay value on path one does not allow any prediction about the delay value on the second path. The analytical model represents the uncorrelated case which becomes very imprecise in case of path delay correlations. Due to the high delay correlation, packets on different paths experience similar delays. This means that it is very likely that missing packets arrive soon after their predecessor on the other path. Thus, the buffer occupancy is much lower as compared to the uncorrelated case.

The experiment shows the limitations of the analytical model. The path delay correlations are not considered and thus, the model becomes imprecise for high correlations. Since the simulation model is based on the path delay time series, it is much more accurate than the analytical model. The high correlations of the links are mainly due to the fact, that the paths are not disjunct, i.e., the multipath transmission is via a shared link.

Mean Squared Errors

Next, we discuss the MSE values for all experiments and show that analysis and simulation match the measurements in most of the cases. The results for both models are depicted in Figure 2.15. The results for the comparison of the measurements with the simulations are detailed in Figure 2.15(a). The MSE is always very low which means that the simulation is always very close to the measurement results. The experiments for the scenario with both paths via Brazil are worst in the comparison between the values. However, Figure 2.14 shows that the simulation model is still very accurate. The comparison between analytical model and measurements are depicted in Figure 2.15(b). Concerning the scenarios via China/Canada and Brazil/Ireland, the results of the model are always very close to the measurements. The scenario with a concurrent multipath transmission via two nodes in Brazil, yields five experiments with a MSE higher than 0.005, which is worse than the MSE of all other experiments. As discussed before, this is mainly due to the path delay correlation which is $\sigma_{path1,path2} > 0.1$



(a) MSE between measurements and simulations (b) MSE between measurements and analysis

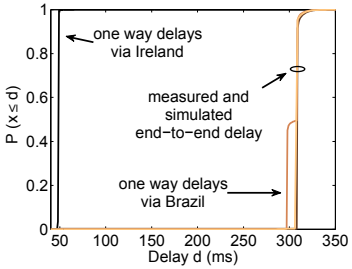
Figure 2.15: *MSE between measurements and simulation or analysis sfor the different scenarios.*

for these experiments. Thus, we can conclude that the analytical model cannot be applied to predict the behavior of the resequencing buffer in case of a significant delay correlation between the involved paths.

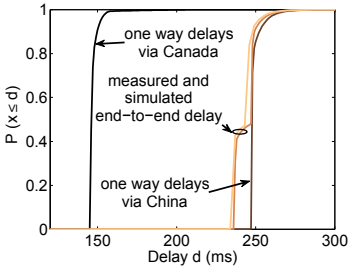
Impact on the End-to-End Delay

In this subsection, we investigate the impact of different path delays on the perceived end-to-end delay which comprises path delay of a packet and the corresponding waiting time in the resequencing buffer. The results are depicted as cumulative frequencies in Figure 2.16.

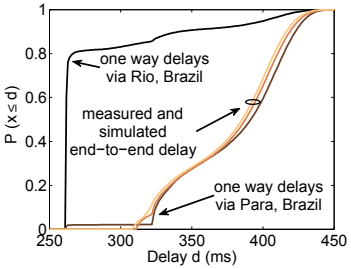
A typical result for the first scenario, a multipath transmission via Brazil and Ireland, is depicted in Figure 2.16(a). As previously discussed, the one-way delay via Ireland is much smaller than via Brazil leading to a buffer occupancy of 20 packets and more, mostly packets sent via Ireland. These packets have to be stored in the resequencing buffer and wait for missing packets sent via Brazil. Thus, the packets sent via Brazil are mostly in sequence and thus can be played out immediately. For these packets the end-to-end delay is similar to the experienced one-way delay since additional waiting times in the buffer can be



(a) Delays for the scenario Brazil - Ireland



(b) Delays for the scenario Canada - China



(c) Delays for the scenario Brazil-Brazil

Figure 2.16: *Measured one-way delays and end-to-end delays and simulated end-to-end delays for the investigated scenarios.*

neglected. Accordingly, a packet sent via Ireland can be played out after the previous packet sent via Brazil arrived. Hence, the packet experiences a similar end-to-end delay, which is dominated by the higher one-way delay path. However, the end-to-end delay is compound of the shorter one-way delay and the waiting delay. Thus, the overall experienced end-to-end delay is dominated by the path via Brazil. As a result, the cumulative end-to-end delay frequencies are quite similar to the cumulative delay frequencies of the path via Brazil. Further, it can be seen that simulated end-to-end delay frequencies closely match the measured

end-to-end delay. For the multipath transmission via Canada-China, depicted in Figure 2.16(b), and via Brazil-Brazil, illustrated in Figure 2.16(c) the same investigations apply. Thus, we can conclude that the simulation model predicts the perceived end-to-end delay very accurate and that it can be approximated fairly with the higher one-way delay distribution.

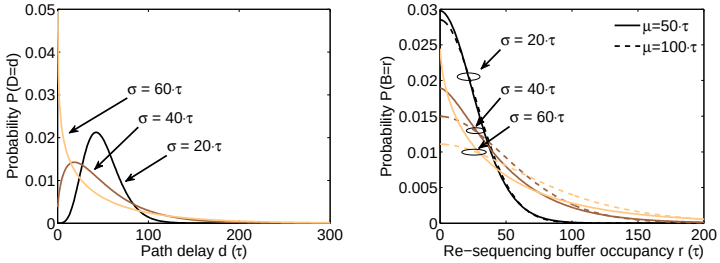
2.4 Evaluation of Multipath Transmission

This section provides results gained with the developed analytical and simulative models. First we discuss the impact of statistical path delays on the system in case of Concurrent Multipath Transmissions. The derived results can be used for a proper dimensioning of the resequencing-buffer or for designing path selection strategies. Subsequently, we investigate the impact of correlated path delays which is typically the case for shared bottlenecks. Last, we focus on improving the system performance by pre-buffering at the destination and investigate the optimization gain. The path delay is normalized with respect to the packet inter-departure time τ. Subsequently, path, scheduling and end-to-end delays are given as multiples of τ.

2.4.1 Influence of Statistical Path Delays on the Resequencing Buffer Occupancy

In this section, we highlight the impact of statistical path delays on the resequencing-buffer occupancy for multipath transmissions. The results can be used to derive appropriate dimensioning guidelines for the resequencing buffer. In a second step, we discuss towards path selection strategies in case of multiple available paths with different path characteristics.

Accordingly, we start with modeling the delay distributions of the involved paths with simple probabilistic distributions. We first investigate the impact of the first and second statistical moments on the resequencing buffer occupancy. For that, we choose negative-binomial (NEGBIN) distributed path delays, since



(a) NEGBIN distributed path delays with different standard deviations for a mean path delay $\mu = 50 \cdot \tau$

(b) Resequencing-buffer occupancy

Figure 2.17: *Impact of different parameters of NEGBIN distributed delays.*

this discrete distribution allows to adjust the mean delay and the delay variation. After that we show that the third statistical moment, the skewness has only a limited influence on the resequencing buffer.

Impact of Path Delay Distributions

We start with analyzing the impact of the path delay distributions of the multipath transmission on the resequencing buffer. For that we focus on the influence of the mean delay μ, the standard deviation σ, the coefficient of variation c_v, and the skewness ν of the delay distributions. The skewness is the third standardized moment and measures of the asymmetry of the probability distribution of a random variable. It is defined as

$$\nu = \frac{\mu_3}{\sigma^3},$$

whereas μ_3 is the third central moment and σ is the standard deviation. If not stated otherwise the results show the resequencing-buffer occupancy, depicted on the x-axis, with the corresponding probabilities, depicted on the y-axis. First, we investigate the impact of different mean delays μ and standard deviations σ on the

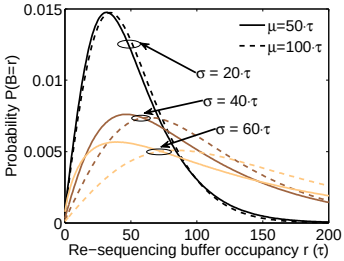
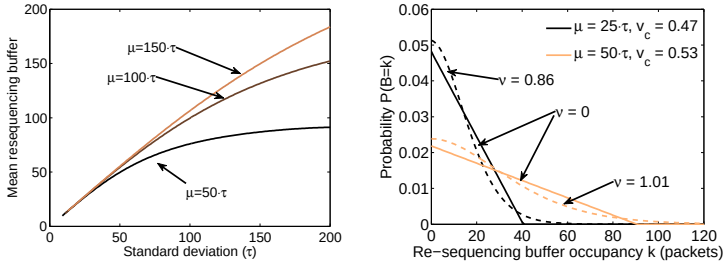


Figure 2.18: *Resequencing-buffer occupancy for three paths with NEGBIN distributed delays.*

resequencing-buffer occupancy for a multipath transmission via two paths with the analytical model. As delay distribution we use a NEGBIN distribution with $\mu = [50|100] \cdot \tau$ and $\sigma = [20|40|60] \cdot \tau$. The delay distributions for different standard deviations are depicted for $\mu = 50$ in Figure 2.17(a). The x-axis denotes the delay, the y-axis the corresponding probability. For $\sigma = 60 \cdot \tau$, the distribution is strictly decreasing, for the other cases it first increases before it decreases. The resequencing-buffer occupancy for mean path delays of $50 \cdot \tau$ and $100 \cdot \tau$ with different delay variations are illustrated in Figure 2.17(b). For $\sigma = 2 \cdot \tau 0$ the distributions for $\mu = 50 \cdot \tau$ and $\mu = 100 \cdot \tau$ are almost identical. In contrast, the distributions for different mean path delays and higher standard deviations are clearly different. For $\sigma = 40 \cdot \tau$ the shape of the buffer occupancy distributions is still similar for both mean delays, however this does not hold for $\sigma = 60 \cdot \tau$. This is due to the fact that for $\mu = 50 \cdot \tau$ and $\sigma = 60 \cdot \tau$ the shape of the NEGBIN distribution is strictly decreasing as illustrated in Figure 2.17(a). For $\mu = 100 \cdot \tau$ the path delay distribution is first increasing before it decreases resulting in the different shapes of the buffer occupancy distributions. For clarity reasons the path delay distributions for $\mu = 100 \cdot \tau$ are not included in Figure 2.17(a).

We can conclude that for path delay distributions of the same type, the delay variation has a higher impact on the resequencing-buffer occupancy than the



(a) Mean path delay and delay variation (b) Skewness

Figure 2.19: *Influence of different mean path delays and variations on the mean resequencing buffer and influence of skewness on the resequencing-buffer occupancy.*

mean delay. However, the impact of the mean delay increases for higher standard deviations. Next, we consider a concurrent transmission via three paths with the same parameters. Figure 2.18 depicts the resequencing-buffer occupancy for the case of a CMP transmission via three paths. the resequencing buffer for the investigated cases is higher compared to the results of a transmission via two paths. Further, it can be seen that the results are very similar for both mean values in case of $\sigma = 20 \cdot \tau$. For higher variations the shapes of the buffer occupancies disperse, i.e., the mean path delay has a significant influence for higher values of σ.

Next, we highlight the influence of different mean path delays $\mu = [50|100|150] \cdot \tau$ and different standard deviations $\sigma = [10, 200] \cdot \tau$ on the mean resequencing-buffer occupancy. For that we investigate a multipath transmission via two paths with equal path delay characteristics. Again, the path delay is negative binomial distributed. The results of this study are depicted in Figure 2.19(a). The x-axis depicts the standard deviation, the y-axis the mean resequencing-buffer occupancy. For small standard deviations $\sigma \leq 35 \cdot \tau$ the mean resequencing buffer the same for the investigated path delays. However, for increasing values

of σ the mean buffer occupancies disperse. Further, the mean buffer occupancy is smaller for lower mean path delays. We can conclude that for a low delay variation the mean path delay has a negligible impact on the mean buffer occupancy. However, for higher delay variations the mean path delay value can not be neglected, and the mean buffer occupancy is higher for higher path delays.

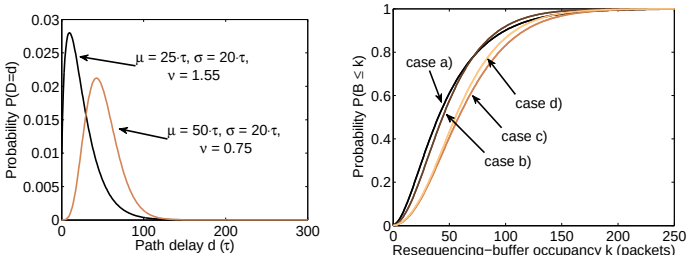
As the last part of our studies on the impact of the path delay distributions on the resequencing-buffer occupancy we analyze path delay distributions with varying skewness ν. For that we assume a concurrent multipath transmission via two paths with equal delay distributions. The path delays are either NEGBIN or equally distributed with equal mean delay $\mu = [25|50] \cdot \tau$ and coefficient of variation $v_c = [0.47|0.53]$. The different shape of the distributions yields to a different skewness ν. For both mean delays and coefficient of variations the skewness for the equal distributed mean delays is $\nu = 0$. For the NEGBIN distributed path delays the skewness is $\nu = [0.86|1.01]$ for mean delays $\mu = [25|50] \cdot \tau$ respectively.

The results of the study are depicted in Figure 2.19(b). The dashed lines illustrate the results for NEGBIN distributed path delays, the solid lines the results for equally distributed path delays. The results show that lower and higher buffer occupancies are more probable for NEGBIN distributed path delays than for equally distributed path delays. However, the differences for the resequencing-buffer occupancy are low.

We can conclude that beside the mean path delay μ and its standard deviation σ, the skewness of the distribution ν also has an impact on the resequencing-buffer occupancy. This effects have to be taken into account for an appropriate buffer dimensioning.

The Complexity of Path Selection

This part of our investigations deals with the complexity of path selection. In the following, we discuss concurrent multipath setting applying three concurrent paths. The number of three paths permits already the investigation of "what-if"



(a) Used negative-binomial distributions with equal standard deviation and different mean values

(b) Impact of different path combinations on the buffer occupancy for three paths

Figure 2.20: *Complexity of path selection.*

scenarios where two paths are already selected and a third path will complement the other paths.

The used distributions for this setting are again negative-binomial distributions with mean $\mu = E[D] = [25|50] \cdot \tau$, standard deviation $\sigma = 20 \cdot \tau$ and skewness $\nu = [1.55|0.75]$. The path delay distribution with $\mu = 50$ has a higher skewness. The distributions are depicted in Figure 2.20(a). First, we now consider four cases of paths combinations: *a)* all three path delay distributions have the same low mean delay $\mu = 25 \cdot \tau$, *b)* all three delay distributions have the same high mean $\mu = 50 \cdot \tau$, *c)* two paths have low mean delay $\mu = 25 \cdot \tau$ and one a high mean $\mu = 50 \cdot \tau$, and *d)* one path has a low mean delay $\mu = 25 \cdot \tau$ and two paths have a high mean delay of $\mu = 50 \cdot \tau$. Figure 2.20(b) shows the cumulative probabilities of the resequencing buffer occupancy for the four cases. The cases *a)* and *b)* are discussed in detail in Figure 2.18. For the used standard deviation, the buffer occupancies are very similar.

For the cases *c)* and *d)*, the latter case with two high path delays performs better in terms of buffer occupancy than case *c)*. Intuitively more paths with lower delay should result in a better performance. Let us consider case *c)* in greater

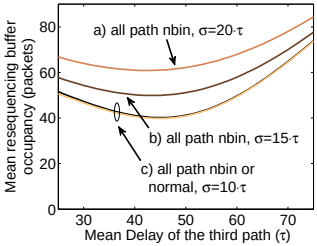


Figure 2.21: *Impact of the path characteristics of an arbitrary 3^{rd} path on the mean resequencing-buffer occupancy.*

detail. Therefore, we assume a single packet from the high delay path which is much overdue. Until the arrival of this overdue packet, the low delay paths can easily increase the occupancy of the resequencing buffer. Thus, the buffer is filled quickly by the low delay paths. This example shows that the high delay path becomes more dominant over low delays paths in terms of buffer occupancy. The selection of the paths should level the variation of the range of mean delays. As a result, it might be better in CMP to choose a path with a higher mean delay and lower delay variation in order to relieve the resequencing buffer and avoid packet loss.

Now, we investigate the impact of the mean delay and of the delay variation. In this setting, we assume that two paths are already selected. The selected paths are of negative-binomial type, one with a low mean delay $\mu = 25 \cdot \tau$ and one with a higher mean delay of $\mu = 50 \cdot \tau$. We consider three cases where these paths have a standard deviation of $\sigma = [10|15|20] \cdot \tau$. In case *a)*, the delay on the third path, which is completing the other two paths, is distributed according to a negative-binomial distribution with $\sigma = 20 \cdot \tau$ and a mean in the range of $[25, \ldots, 75] \cdot \tau$, in case *b)*the delay distribution is also negative-binomial but with $\sigma = 15 \cdot \tau$, and in case *c)* we use a normal or a negative-binomial distribution for all paths with low delay variance and $\sigma = 10 \cdot \tau$. This case should compare

the influence of the skewness, which is much lower for the Gaussian distribution.

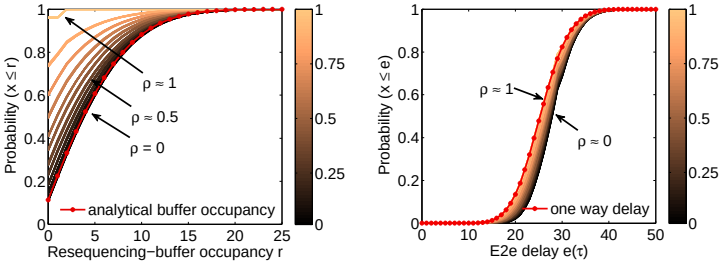
Figure 2.21 shows that the mean delay of the third path has a significant impact on buffer occupancy. If the mean path delay of the third path increases beyond the one with the higher mean delay, the mean buffer occupancy increases.. However, if the mean path delay of the third path is between the other two mean values, a minimum exists. This effect is taking more shape when σ is low (case *b*) and *c*)). We also conclude that a high delay variance degrades the CMP performance. Case *c* shows that the skewness has only little impact on the mean resequencing-buffer occupancy.

In summary, the above presented studies show that a path selection strategy should also consider the second moment of path delay. For a in depth analysis of the resequencing-buffer occupancy beyond a mean value analysis, the skewness might also have an impact.

2.4.2 Impact of Packet Delay Correlation on the Utilized Paths

In this subsection, we investigate the impact of packet delay correlation of packets on the different paths. For that we conducted simulation studies for a concurrent multipath transmission via two paths. We consider the delay distributions for both paths to be either of negative-binomial type, of *Poisson* type or of an approximated *NORMAL* type. The different type of distributions allow a study of equal mean path delays with different delay variations. The mean delay is set to $\mu = 25 \cdot \tau$ with standard deviations of $\sigma_{norm} = 12 \cdot \tau$, $\sigma_{pois} = 5 \cdot \tau$ and $\sigma_{nbin} = 10 \cdot \tau$. For our investigations we vary the correlation coefficient ρ of the packet delays of both paths $\rho \in [0, 1]$.

In the following the results are illustrated as cumulative probability distributions of the buffer occupancy or the end-to-end delay, which is depicted on the x-axis. Further, the results of the analytical models are included in the diagrams as they illustrate the resequencing-buffer occupancy for uncorrelated path delays.



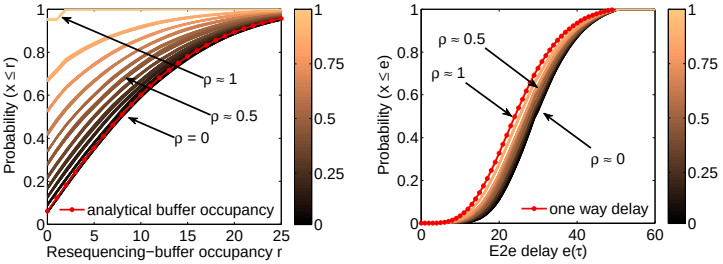
(a) Resequencing-buffer occupancy

(b) End-to-end delay

Figure 2.22: *Impact of ρ on buffer occupancy and end-to-end delay for Poisson distributed packet delays.*

The results for Poisson distributed path delays are depicted in Figure 2.22. In Figure 2.22(a), the resequencing-buffer occupancy is depicted for different correlation coefficients. For the case of a full correlation the buffer is nearly empty. In this case, consecutive packets sent via different paths experience almost the same path delay and thus arrive at the destination simultaneously. The packets can be played out instantly and do not have to be stored in the resequencing buffer. For lower correlation coefficients, the resequencing-buffer occupancy increases continuously. In case of no correlation, the simulated resequencing-buffer occupancy reaches the analytically computed occupancy. This due to the fact, that the analytical model assumes uncorrelated packet delays on the paths.

The perceived end-to-end delay is illustrated in Figure 2.22(b). It can be seen, that the impact of the correlation coefficient is lower than compared to impact on the resequencing-buffer occupancy. However, for an decreasing coefficient of correlation, the perceived end-to-end delay increases. This is due to the additional waiting time in the buffer. The next scenario highlights the impact of negative-binomial distributed path delays on the buffer occupancy and the end-to-end delay, as illustrated in Figure 2.23. The results are similar to the case of Poisson distributed delays. However, the impact of ρ on the resequencing buffer



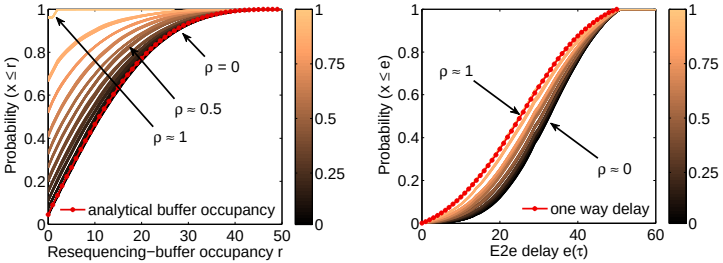
(a) Resequencing-buffer occupancy

(b) End-to-end delay

Figure 2.23: *Impact of ρ on buffer occupancy and end-to-end delay for negative-binomial distributed packet delays.*

and on the end-to-end delay is higher. This is due to the higher standard deviation and therewith a higher variation of the path delay distributions in case of negative-binomial distributed delays.

The last scenario we focus on is a concurrent multipath transmission via two paths with approximately NORMAL delay distribution. The resequencing-buffer occupancy is shown in Figure 2.24(a). The impact of the coefficient of correlation on the resequencing-buffer occupancy is similar to the other cases. However, the buffer occupancy itself is higher compared to the negative-binomial and the Poisson case. The reason for this is the higher variation of the path delay distributions. This also has a significant impact on the end-to-end delay, as depicted in Figure 2.24(b). It can be seen that with decreasing ρ the end-to-end delay increases much more compared to the other cases. The reason for this is the higher variation of the path delay distribution as compared to the previous cases. The higher buffer occupancy implies higher waiting times for the stored packets. In the following, we investigate the impact of the correlation coefficient on the mean buffer occupancy and the mean end-to-end delay for all three scenarios, the Poisson scenario denoted as case a), the NEGBIN scenario as case b) and the NORMAL scenario as case c). The results are shown in Figure 2.25. First, we focus on the



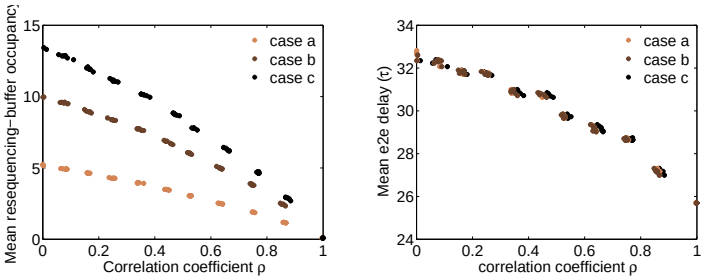
(a) Resequencing-buffer occupancy (b) End-to-end delay

Figure 2.24: *Impact of ρ on buffer occupancy and end-to-end delay for NORMAL distributed packet delays.*

mean resequencing-buffer occupancy for varying correlation coefficients ρ, as illustrated in Figure 2.25(a). It can be seen that the mean resequencing buffer is maximal for the uncorrelated case, $\rho = 0$, and that the buffer is empty for the correlated case $\rho = 1$. Further, it can be seen that for decreasing ρ the mean buffer occupancy decreases slowly at first. For small values of ρ it decreases faster. Last but not least, the higher the path delay variation, the higher the mean buffer occupancy.

We can conclude that low path delay correlations have a negligible impact on the buffer occupancy. Nevertheless, higher path delay correlations have a significant impact on the buffer occupancy and thus on the system.

The impact of the delay correlation on the end-to-end delay is illustrated in Figure 2.25(b). The x-axis depicts the correlation, the y-axis the mean end-to-end delay. It can be seen that for the investigated scenarios the difference between the type of path delay distributions can be neglected. For the uncorrelated case the mean value is maximal, for the correlated case the mean end-to-end delay is equal to the mean path delay, $\mu = 25$. Again, the end-to-end delay decreases with a smaller gradient for higher values of ρ as for lower values of ρ.



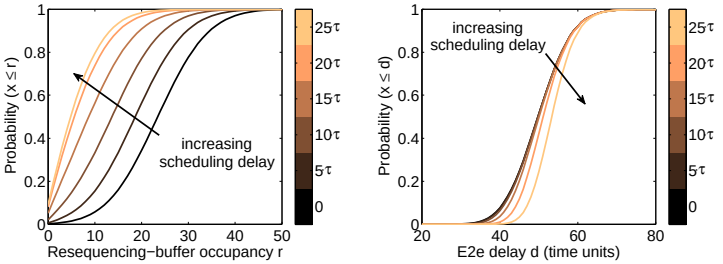
(a) Resequencing-buffer occupancy

(b) End-to-end delay

Figure 2.25: *Scatter plot for the investigated scenarios and varying coefficients of correlation.*

2.4.3 Impact of Scheduling

As shown in Section 2.3.5, high differences between the path delay distributions lead to a high minimal resequencing-buffer occupancy. Further, it was discussed that this is due to the fact that packets experiencing a low delay have to wait at the destination for packets experiencing higher delays on the other path. Thus, delaying faster packets with techniques like pre-buffering might have a positive impact on the resequencing-buffer occupancy. In this subsection, we investigate the impact of scheduling by pre-buffering at the source and highlight the impact on the resequencing-buffer occupancy and the perceived end-to-end delay. For this purpose we conduct simulation studies for a concurrent multipath transmission via two paths. We consider the delay distributions for both paths to be either of Poisson type, case a), or *n*egative-binomial, case b). The mean delay for one path is set to $\mu = 25 \cdot \tau$, the mean delay for the second path to $\mu = 50 \cdot \tau$. For the faster path we vary the scheduling delay $d_s \in [0, 5, 10, 15, 20, 25] \cdot \tau$. If not stated otherwise the results show the resequencing-buffer occupancy or the end-to-end delay, depicted on the x-axis, with the corresponding cumulative probability distributions, depicted on the y-axis.



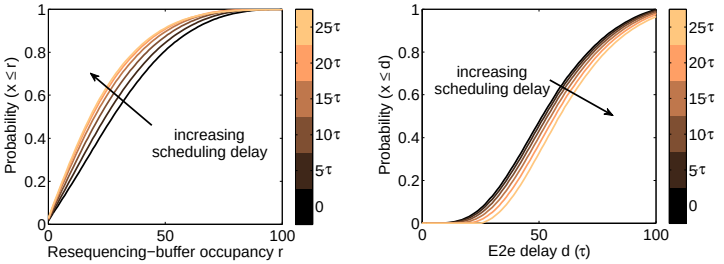
(a) Resequencing buffer (b) End-to-end delay

Figure 2.26: *Impact of scheduling mechanisms for the Poisson distributed path delays.*

The results for the Poisson distributed path delays are shown in Figure 2.26. Hereby, Figure 2.26(a) depicts the cumulative distribution function of the resequencing-buffer occupancy. It can be seen that with increasing scheduling delay the resequencing-buffer occupancy is reduced. The occupancy decreases rapidly for scheduling delays $d_s \in [5, 20]$. However, the additional gain in terms of buffer occupancy between $d_s = 20 \cdot \tau$ and $d_s = 25 \cdot \tau$ is negligible.

The perceived end-to-end delay distribution for the packets are depicted in Figure 2.26(b). The x-axis denotes the end-to-end delay whereas the y-axis depicts the cumulative probabilities. It can be seen that for increasing scheduling delays the end-to-end delay increases. However, this increase is nearly imperceptible for $d_s \leq 20 \cdot \tau$. Thus, an additional scheduling delay for the path with the lower mean delay could tune the system significantly: the resequencing buffer is balanced while the end-to-end delay is almost not increased.

The second scenario, the NEGBIN path delay case, is illustrated in Figure 2.27. The impact of an increasing scheduling delay on the resequencing-buffer occupancy is shown in Figure 2.27(a). The y-axis denotes the cumulative probabilities. Again, the buffer occupancy can be reduced for increasing scheduling delays. However, the gain in terms of buffer occupancy is lower as in case a).



(a) Resequencing-buffer occupancy (b) End-to-end delay

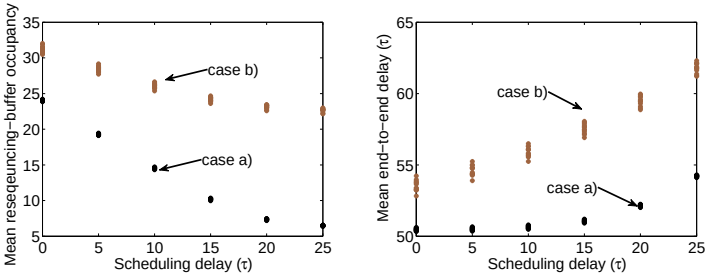
Figure 2.27: *Impact of scheduling mechanisms for the NEGBIN distributed path delays.*

Further, the additional gain in case of $25 \cdot \tau$ is, compared to the $20 \cdot \tau$ scheduling delay, nearly imperceptible. For the end-to-end delay, depicted in Figure 2.27(b) the results are similar to the results for the Poisson case. At first the increased scheduling delay has a minor impact on the perceived end-to-end delay. However, the effect increases steadily for higher values of d_s and become more and more significant.

In the following, we investigate the impact of the scheduling delay on the mean buffer occupancy and the mean end-to-end delay for both scenarios. The results are shown in Figure 2.28.

The mean resequencing-buffer occupancy for varying scheduling delays d_s is depicted in Figure 2.28(a). The mean resequencing buffer is maximal in case of no scheduling delay $d_s = 0$. It decreased with increasing scheduling delays, whereas the decreasing gradient is lower for higher scheduling delays. Again the results show, that the higher the path delay variation, the higher the mean buffer occupancy.

The impact of the scheduling delay on the end-to-end delay is illustrated in Figure 2.28(b). The x-axis depicts the scheduling delay, the y-axis the mean end-to-end delay. The results show, that the mean end-to-end delay is higher for



(a) Resequencing-buffer occupancy (b) Perceived end-to-end delay

Figure 2.28: *Scatter plot for the investigated scenarios for different pre-buffering delays.*

NEGBIN distributed path delays. Further, for higher scheduling delays the mean end-to-end delay increases, in case a) moderate, in case b) more significantly.

We can conclude that it is possible to influence the resequencing buffer and the end-to-end with pre-buffering at the source. Further, the effects highly depend on the path delay distributions. In case of path delays with low variation it is for instance possible to relieve the resequencing buffer without increasing the end-to-end delay significantly. The implemented simulation model allows an investigation of the system and thus a proper dimensioning of the scheduling delay with respect to the output parameters.

2.5 Lessons Learned

The objective of this chapter was to investigate performance issues of multipath transport mechanisms in the Future Internet. Therefore, theoretical models for a multipath data transmission are proposed and validated by a prototype implementation and measurement studies in experimental facilities. In a broad range of scenarios we apply the simulative and analytical models and investigate the im-

pact of shared bottlenecks, path selection strategies, and scheduling mechanisms on the system performance.

The implemented simulator allows a detailed investigation of the system and of the performance metrics. Due to the high operation times of the simulation it suits for an off-line investigation of the system. However, it allows a broad investigation of parameters and mechanisms which can not be investigated by the presented analytical models. These are for instance the perceived end-to-end delay, the impact of scheduling mechanisms, and the influence of correlated path delays on the re-sequencing buffer occupancy. Correlated path delays may occur in case of transmissions via partly shared paths.

The analytical model on the other hand allows in general a fast computation of the resequencing buffer but lacks the computation of the end-to-end delay. Nevertheless, it can be used for comparing different path combinations with respect to their influence on the resequencing buffer. This allows proper buffer dimensioning of the resequencing buffer and an approximation of the packet loss probability. Further, the analytical model allows the selection of path combinations so that the buffer occupancy is minimized.

In detail we investigated the impact of different path delay distributions characterized by their mean value, standard deviation, and skewness on the resequencing buffer occupancy. We investigated that in case of equal path delay distributions, the delay variation has a higher impact on the resequencing buffer occupancy than the mean delay. We showed that also the skewness of the path delay distributions has a perceptible influence on the resequencing buffer occupancy.

In a next step, we derived insights into path selection strategies by investigating multipath transmissions via three paths. Two paths were fixed and the third was chosen out of a set of paths with different path delay characteristics. It turned out that in such a case the third path should have similar characteristics, at least a similar mean delay as the fixed paths with the higher delay. This leads in general to a lower buffer occupancy.

Further, the impact of packet delay correlation was investigated. For that, different delay distributions with equal mean values are chosen and the coefficient

of correlation ρ is varied. It was shown how different positive values of ρ influence the resequencing buffer occupancy and the perceived end-to-end delay: increasing path delay correlations lead to a lower utilization of the buffer and to a lower end-to-end delay. Further, it was shown that the analytical model is a lower bound approximation of the resequencing buffer occupancy which is still accurate for correlation values $\rho \leq 0.3$.

The last study in this chapter highlighted the impact of pre-buffering at the source. For that we choose equal path delay distributions for the two paths with different mean values. A scheduling delay d_s for the faster path was introduced and varied and the influence on the buffer occupancy and the end-to-end delay studied. It turned out that an additional scheduling delay relieves the buffer occupancy and increases the end-to-end delay. However, it was shown that for appropriate values of d_s the impact on the end-to-end delay is negligible while the buffer savings are significant.

It can be summarized from this chapter that the derived models enable an online path selection out of different available path combinations for the proposed system . Additionally, the resequencing buffer can be leveraged during runtime by pre-buffering of packets at the source. By a careful tuning of the pre-buffering delay, the impact on the perceived end-to-end delay can be neglected. As discussed, the results can be partly transfered to SCTP-CMT and may also enable an optimization of the data transmission for this transport protocol.

3 Quantification of the User Perceived Quality for Scalable Video Streaming

In the previous chapter, we investigated concurrent multipath transmissions, a technique, which enables the pooling of transport links and thus may increase the available capacity for data transport. This high-capacity transport link can be used, e.g., to provide a high quality video streaming service. However, in order to operate such an application, it is necessary to understand the influence of network transport on it. In particular, the impact of insufficient network resources on the user perceived quality, the Quality of Experience (QoE) are studied for the case of scalable video streaming using the state-of-the-art codec H.264/SVC. Network applications can either adopt to insufficient resources by reducing their requirements or they accept that the network is not able to provide the necessary resources, which results in waiting times and packet loss. A controlled reduction of the required network resources, the first case, is called *controlled quality reduction* throughout this work. The latter case leads to a network-caused quality reduction which can not be controlled by the service. Thus it is called *uncontrollable quality reduction* throughout this work.

Uncontrollable influences due to limited network resources disturbs the user's video streaming experience. In case of YouTube-like TCP-based streaming this disturbances are additional waiting times where the video is pre-buffered before or during play out. For UDP-based streaming this may even lead to to a loss

of video information through packet loss. In general, this loss leads to distorted frames, video freezing, and stream starvation. On the other hand, the video quality and therewith the required bandwidth can be reduced in a controlled way allowing an adaptation to the network conditions. This can be achieved by reducing the frame rate, the resolution, and the image quality of the streamed video.

In this chapter, we discuss the impact of such controllable and uncontrollable quality degradations in case of UDP-based video streaming using the video codec H.264/SVC. For that, we derive guidelines how much packet loss can be tolerated. Further, we investigate how the three control knobs provided by SVC, frame rate, resolution, and image quality, can be used to reduce the required bandwidth in a way that the video quality is almost not affected. For that, the acceptable areas, i.e, the frame rates, image qualities, and resolutions which are required to satisfy the user are evaluated together with their potential to reduce the necessary streaming bandwidth.

Figure 3.1 illustrates these areas of QoE control knob settings for SVC in a spider plot. The different axes denote the influence of the control knob settings on the user-perceived quality. For each of these scalabilities, we investigate the impact on the user-perceived quality and define thresholds where the user-perceived quality is acceptable. The results gained allow for an on-line management of a SVC-based video streaming service with respect to the user-perceived quality.

The chapter is structured as follows. In Section 3.1, we introduce the video codec H.264/SVC, discuss its capability to adopt the video streaming to network conditions, and summarize methods for evaluating the video quality. Section 3.2 presents the methods used to assess the impact of controlled and uncontrolled distortions on video streaming. Accordingly, Section 3.3 highlights the derived results and discusses their importance for managing the QoE of video streaming. Related work is discussed in Section 3.1.3 and the lessons learned are summarized in Section 3.4.

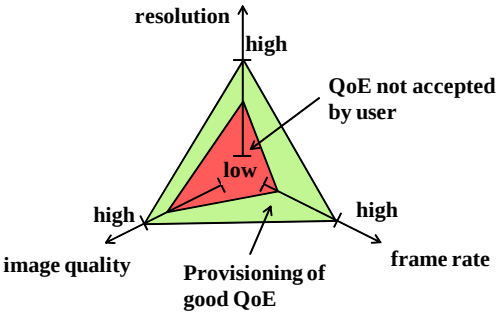


Figure 3.1: *Acceptable area of QoE control knob settings.*

3.1 Background on Video Quality Assessment and Related Work

In this section, we introduce video coding and discuss the video codec H.264/SVC in detail. We provide insights into the different scalabilities of this video codec, which allows an easy quality adaption with respect to the network conditions. In the second part of this section, we discuss methods for assessing the impact of video quality distortions on the Quality-of-Experience (QoE). We briefly discuss methods for subjective QoE (*sQoE*) assessments with surveys and full reference metrics, which compare the original video clip with the disturbed video clip. Thus, the grade of distortion can be computed and approximations of its influence on the user perceived quality, the objective QoE (*oQoE*) can be derived.

3.1.1 Scalable Video Coding with H.264/SVC

The scalable video codec H.264/SVC is an extension of the widely used H.264/AVC codec. The standard is specified by the Telecommunication Standardization Sector of the International Telecommunications Union (ITU-T) in the ITU-T Recommendation H.264 [73].

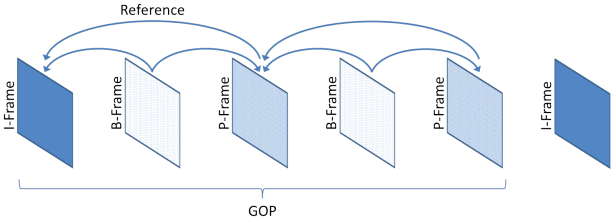


Figure 3.2: *Predictive relationship of frame types.*

Like regular video on a film roll, digital video is encoded on a per-frame basis. A sequence of frames shown in rapid succession, starting at about 24 pictures per second, is perceived by the human eye as fluid motion. Instead of encoding many full pictures per second, digital video uses redundancies in preceding pictures to reconstruct the following frames from small parts of image information, such as motion information. This is called picture prediction. In this context, different types of frames are used to encode a video sequence. These are I-Frames, P-Frames and B-Frames.

I-Frame (Intra-coded picture) These frames contain pictures with full image information, just like a static image (e.g., JPEG).

P-Frame (Predictive-coded picture) These frames contain information about the change from a preceding frame. They use an I-Frame or another P-Frame as reference to reconstruct the picture, and cannot be decoded fully or in parts without their reference frame. For example, in case of an object moving in front of a static background, only the motion of the object is encoded. Thus, the required network bandwidth can be reduced.

B-Frame (Bi-predictive-coded picture) Like P-Frames, these frames use redundant data and prediction to encode a picture, but use both preceding and

(a) CIF (b) QCIF

Figure 3.3: *Comparison of CIF and QCIF resolution.*

following frames as a reference. This allows a higher compression of the video and thus higher network bandwidth savings.

The relationships between the types of frames are illustrated in Figure 3.2. The frames are depicted as squares, while the arrows show the direction of the prediction, and point to the frame that gets referenced. This shows the basic difference between P- and B-Frames. P-Frames reference one preceding frame, while B-Frames reference a preceding and a succeeding frame. A sequence of an I-Frame with all other frames that depend on it is called Group Of Pictures (GOP).

There are three different scalabilities for scalable video coding, namely temporal scalability (frame rate), spatial scalability (resolution), and quality scalability (image quality), which affect the required bandwidth and the perceived quality of a video. They are described in the following.

Resolution is the pixel count in the x and y dimension of the video image. The higher the resolution, the more detail an image can contain. Common res-

(a) High quality (b) Low Quality

Figure 3.4: *Comparison of different quality encodings.*

olutions for videos in various applications are for instance CIF (352x288) or HD (1280x720 and 1920x1080) resolutions. These standardized resolutions [73] are the basis for the resolutions chosen for the sample sequences in the investigations later in this work. Figure 3.3 illustrates the scale difference of two resolutions. In Figure 3.3(a), a frame of a video in CIF resolution is shown, while in Figure 3.3(b) the same frame is shown in QCIF (176x144) resolution. As is shown later in this work, resolution has the highest impact on the video bitrate and file size respectively, since doubling the resolution of an image quadruples its pixel count.

Frame rate is the rate at which single image frames of the video are displayed, measured in Frames Per Second (FPS). Common frame rates are 24 fps (NTSC TV), 25 fps (PAL TV), or 30 fps.

Quality depicts the quality of the single frame pictures measured in Signal to Noise Ratio (SNR), similar to quality levels in image encoding, for example JPEG. In SVC, a quality layer is characterized by the so-called Quantization Parameter (QP). The QP determines, how much spatial detail is saved, or how strong it is compressed. A lower value for QP denotes more

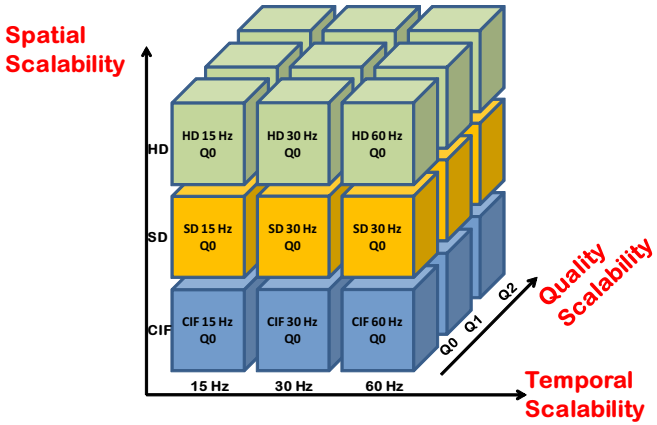


Figure 3.5: *SVC Cube, illustrating the possible scalability dimensions for a video file.*

displayed detail. This leads to lower compression and therefore a higher bitrate. The quality layer can be seen as a special case of spatial scalability, but with pictures of same size encoded with different fidelities and bandwidths. Figure 3.4 illustrates the difference in two exemplary quality encodings. Figure 3.4(a) shows a frame in high quality encoding, while in Figure 3.4(b) the same frame is shown with a lower quality encoding at the same resolution.

One or more of these scalability parameters can be varied for a video encoded in H.264/SVC. In the following, we investigate an SVC video with temporal, quality and spatial layers in more detail. The basic video quality, i.e, the lowest resolution, frame rate, and image quality of the video, is encoded as base layer. Based upon this base layer, extension layers enhance the quality with respect to the three scalabilities. Figure 3.5 gives an example of different possible scalability parameters for a video file. The depicted scalable video file can be watched in three different temporal resolutions (15Hz, 30Hz, 60Hz), three different spatial

resolutions (CIF, SD, HD), and three different quality resolutions (Q0, Q1, Q2). The left bottom "subcube", CIF resolution with 15 Hz and quality Q0, is the base layer which is necessary to play back the video file. Based on this layer, different enhancement layers permit a better video experience with a higher resolution, better SNR or higher frame rate, respectively. The more subcubes along any of the three axes are available, the higher the quality. If all subcubes are available, the video can be played back in highest quality. If all subcubes within quality Q0 are available, the video can be played back in HD-resolution with 60 Hz, but only with a low SNR.

In this context, the question occurs, how the end-user perceives the actual quality of the delivered video. Is a user more satisfied with i) a low resolution, a low picture quality but a smooth video play out or ii) a high resolution with high picture quality at the cost of quality degradations due to packet loss in the network or a lower frame rate? The following part of this section discusses methods which allow us to answer this research question.

3.1.2 Quality of Experience Assessment for Video Streaming

In this section, we discuss methods for assessing the Quality-of-Experience for video streaming. QoE is in its nature a subjective measure of the quality a user experiences when using a service such as Voice-over-IP, video streaming, or browsing the web. It combines non-technical parameters such as user perception, experience, and expectations with technical parameters on application and network level.

The QoE of an application is essentially a characteristic that is highly subjective to a user's perception, experience and expectations. Thus, the obvious way to measure it is to vary the network Quality of Service (QoS) and investigate its impact on the user perception by asking the users themselves. Based on the user ratings the QoE can be calculated as, for instance, the average user rating, the Mean Opinion Score (MOS). The QoE assessment based on user surveys is

called *subjective* QoE (sQoE) throughout this work.

On the other hand, it is possible to compare the original video clip with the received video clip based on signal and frame processing techniques. Thus, the distortion of the received video clip can be evaluated and the received video quality calculated. Based on the video quality, an approximation of the user perception can be calculated. Such an approximation of the user-perceived quality based on such computational metrics is called *objective* QoE (oQoE) throughout this work.

Subjective User Surveys and Mean Opinion Scores

Quality of Experience is defined as the subjectively perceived acceptability of a service [74]. The perceived quality can be investigated in subjective tests, where presented stimuli—such as impaired video sequences—are rated by subjects under controlled conditions. However, the assessment of results from these tests is difficult, since the individual scoring depends on daily form and mood of the human subject, as well as on generally overly pessimistic or overly optimistic users. For the purpose of having comparable and uniform tests, the International Telecommunication Union has created a recommendation for standardized user tests concerning video and image quality in ITU-R BT.500-11 "Methodology for the subjective assessment of the quality of television pictures" [75]. The recommendation gives guidelines to normalize the viewing conditions for a subjective study, in terms of room illumination, display position and parameters (resolution, contrast, brightness) and hardware to use, the selection of test materials and the length of test sessions. Typical methods for assessing the sQoE are Double-Stimulus Continuous Quality Scale (DSCQS), Double-Stimulus Impairment Scale (DSIS) and Singe-Stimulus Continuous Scale Quality Evaluation (SSCQE).

DSCQS The double-stimulus continuous quality-scale method is used to measure the quality of a distorted video relative to an undistorted reference. A test subject looks at pairs of pictures or videos, one of which is impaired, and one is the unimpaired reference. The subject then scores both pictures

on a continuous quality scale, which is divided into equal length intervals called "Bad", "Poor", "Fair", "Good", and "Excellent".

DSIS The double-stimulus impairment scale method is used to measure the failure characteristics of a video. Similar to DSCQS, the user rates reference and impaired videos, but not at the same time. First, the unimpaired reference is shown, followed by the same sequence impaired. The subject then votes on the impairments of the video on a five-grade impairment scale with the grades "Very annoying", "Annoying", "Slightly annoying", "Perceptible but not annoying" and "Imperceptible".

SSCQE The single-stimulus continuous quality evaluation method measures the overall quality. The test subject is shown only an impaired version of a sequence, and then votes on the continuous quality scale.

The grades of the scale are mapped to numerical values from 1 (bad quality) to 5 (excellent) and the mean of the scores, the MOS value, is obtained for each test condition. The obtained rating expresses the subjective Quality of Experience (sQoE).

The presented methods differ in their objectives. Whereas DSCQS is used to measure the remaining quality of an impaired video relative to the reference video, DSIS focuses on the distortions and rates impairments. SSCQE measures the overall satisfaction of the user with his experience regardless of a reference. Another criterion for the selection of a specific method is available equipment, budget, and time.

The results of such surveys reflect the user's perception and thus have a high significance. However, due to different quality judgment of human observers, multiple subjects are required to participate in a subjective study [76]. According to [75], at least 15 observers should asses a stimuli in order to gain significant results. Tests are conducted manually in a controlled environment which is time-consuming and costly. Thus, it should be used as base data for objective video quality algorithms which automatically predict the visual quality of a video clip.

Objective Video Quality Metrics

Objective video quality metrics can be classified into three categories by the required amount of reference information [77]: *Full-Reference* (FR) metrics are based on frame-by-frame comparison between a reference video and the video to be evaluated; *No-Reference* (NR) metrics have to make assumptions about the video content and distortions, e.g. by evaluating the blockiness of a frame, as a common artifact in block-based compression algorithms such as MPEG; *Reduced-Reference* (RR) metrics evaluate the test video based on a subset of features previously extracted from the reference video. Based on the complex nature of cognitive aspects and the human visual system, objective quality metrics do not capture its entire complexity and focus on aspects, which have been shown to correlate well with human perception in subjective tests. For our studies we use publicly available full reference metrics, i.e., the Peak Signal to Noise Ratio (PSNR), Structural Similarity Index Metric (SSIM), and Video Quality Metric (VQM). These mechanisms are briefly described in the following.

PSNR Peak Signal to Noise Ratio is the function of the mean squared error of luminance and chroma values on a per-pixel basis between a reference image and an impaired image. For every pixel, the difference between the expected luminance $I(i,j)$ and the received luminance $J(i,j)$ is calculated and the Mean Squared Error (MSE) is computed according to Equation 3.1. The PSNR is computed with respect to the maximum signal strength (luminance) MAX_I in the reference image I and the MSE, cf. Equation 3.2. PSNR is measured in decibel (dB), with higher values indicating a less impaired image. Calculating PSNR on a per-frame basis and computing the mean value gives a quality assessment for video sequences. PSNR is the most widely used metric for assessing video quality, since it is easy and fast to calculate. However, it has been shown that it does not correlate well with human perception [78] in case of video quality assess-

ment.

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i,j) - J(i,j)]^2, \tag{3.1}$$

$$PSNR = 20 \cdot \log_{10} \left(\frac{MAX_I}{\sqrt{MSE}} \right). \tag{3.2}$$

SSIM Structural Similarity Index Metric is also a full-reference metric, which assesses image quality by means that come closer to human vision and ability to see structures in an image [79]. Luminance and contrast of image pixels with respect to the other pixels around are compared and the image structure is analyzed. The SSIM value is computed with respect to the mean values μ, the standard deviation σ and the cross-correlation $\sigma_{x,y}$ of the luminance, contrast and structure of the image pairs x, y, cf. Equation 3.3. C_1 and C_2 are constants used to stabilize the results in case of values close to zero. The value range for the resulting SSIM value is between 1 (no perceptible impairment) and 0 (maximum perceptible impairment). The SSIM metric is still easy to calculate and correlates better to human perception as PSNR [76, 79].

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}. \tag{3.3}$$

VQM Video Quality Metric is the most in-depth and complicated full-reference metric presented here. It has been standardized by the ITU in ITU-R BT.1683 [80] and has been shown to predict the outcome of a subjective DSCQS survey well. Both the reference and impaired video are first sampled, i.e. they are converted to a component video containing the three signals luminance, blue color difference, and red color difference. Further, the video frames are indexed and divided into rectangular sub-regions. After that, the sampled videos are calibrated, i.e., they are aligned with respect to spatial and temporal registration as well as brightness and con-

trast. Differences in spatial, temporal, and chrominance properties are calculated and combined into the VQM model. The resulting VQM value ranges from 0 (no perceptible impairment) to 1 (maximum perceptible impairment).

Several studies showed the high correlation of VQM with human perception [76, 81]. However, this comes to the costs of a high and complex computation.

The presented objective video quality metrics PSNR, SSIM, and VQM allow the quantification of video distortions and allow the comparison of, e.g., different packet loss patterns on the video streaming quality. In order to evaluate video quality, the image quality metrics PSNR and SSIM compute the image quality frame by frame and calculate the average image quality.

As discussed above, video quality metrics and user surveys can be used to evaluate the grade of distortion between a reference video clip and a distorted video clip. While user surveys reflect the user perception, video quality metrics foremost allow a comparison of distortions with respect to the chosen metric. However, the metric values can be mapped to, e.g., a MOS scale by using the sQoE values as ground data. This results in an objectively computed QoE indicatior, the oQoE. This process is depicted in Figure 3.6. Based on results obtained for still images in [79] and for videos in [81], we introduce a mapping of PSNR, SSIM, and VQM to a nominal 5-point MOS scale according to Table 3.1 for expressing the objectively computed Quality of Experience, the oQoE. These mappings can be used to estimate the impact of controlled and uncontrolled video distortion on the user-perceived quality. For our investigations we rely on objective metrics instead of performing user surveys. That is mainly since subjective tests are cumbersome and expensive. If we want to conduct a user survey for five different random packet loss rates and three different video clips this yields to 15 stimuli per user. If an average user requires 2 minutes for each stimuli the whole test lasts for 30 minutes. According to the ITU standard at least 15 observers should rate the content quality. Thus, the entire tests take 7.5 hours, without preparation and

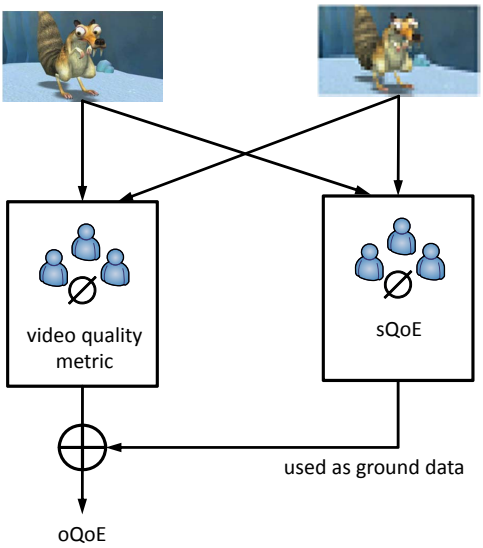


Figure 3.6: *Relation between video quality metrics, sQoE, and oQoE.*

setup of the equipment. Additionally, our studies aim at designing a QoE management mechanism. For such a mechanisms it is sufficient to provide guidelines how the user-perceived quality is affected the least in case of insufficient network conditions. In particular we want to study whether it is better to either to reduce frame rate, resolutions, or image qualities, or to keep the video quality and risk broken frames. The results can be used to design an on-line quality management mechanism for a H.264/SVC-based streaming application.

MOS	PSNR	**SSIM**	**VQM**
5 (excellent)	> 45	> 0.99	< 0.2
4 (good)	≥ 33 & < 45	≥ 0.95 & < 0.99	≥ 0.2 & < 0.4
3 (fair)	≥ 27.4 & < 33	≥ 0.88 & < 0.95	≥ 0.4 & < 0.6
2 (poor)	≥ 18.7 & < 27.4	≥ 0.50 & < 0.88	≥ 0.6 & < 0.8
1 (bad)	< 18.7	< 0.50	> 0.8

Table 3.1: *Mapping of quality metrics to objective QoE measures.*

3.1.3 Related Work

This sections details related work on the impact of packet loss and different video qualities on the user-perceived quality. These investigations differ from prior work which focused on modeling video traffic based on MPEG-2 [82], instead they focus on understanding the impact of the network on video streaming, in particular on H.264.

Investigations on Uncontrolled QoE distortions

In the last years, many subjective studies have been carried out to investigate the impact of packet loss on the user-perceived quality [44, 83–85]. In [85] the authors discuss the impact of packet loss and different content for the case of high quality IPTV services using H.264/SVC. A user survey with ninety participants for different packet loss rates and packet loss patterns is conducted. The authors discover, that for the same packet loss rate the burst length of lost packets has a significant impact on the subjective video quality. The higher the burst length, the lower is the decrease of perceived quality with increasing loss rate. Thus, random packet loss has a more severe impact on the user-perceived quality as burst packet loss. Further, it was shown, that packet loss rates between 0.05% and 0.5% yield a MoS rating of 2, i.e., a rather annoying video quality. Similar investigations have been made in [83]. The authors conducted user surveys for investigating the impact of packet loss rates on the user-perceived quality for HD and SD IPTV streaming using H.264/AVC. The results indicate a strong influence

of packet loss on the user-perceived quality. Accordingly, a fast decrease in user-perceived quality can be seen between 0% and 1% packet loss rate. The authors also show that user do not accept video streaming in case of more than 1% packet loss rate. Recently, first tests on packet loss for H.264/SVC streaming have been conducted by [84]. The authors performed user surveys for different packet loss rates affecting base layer and enhancement layers with respect to quality scalability for a resolution of 704x576 pixels. The results show that packet loss affecting the enhancement layers has a lower impact on the perceived QoE than packet loss affecting the base layer. The authors use H.264/SVC and conduct subjective user studies. They also show that packet loss rates less than 1% have a huge impact on the user experience and lead to poor video streaming experience. These subjective studies have been taken out during the course of this work. The results of the presented studies are in-line with our studies and support our results.

Investigations on Controlled QoE distortions

First, we detail related work concerning video quality adaptations on other video codecs. After that, we investigate recent work discussing H.264/SVC.

In [86] Buchinger et al. describe the interconnection between the compression rate and the frame rate in mobile environments. It is shown that frame rate has a huge impact on the user-perceived quality. However, this influence depends strongly on the amount of color changes within a picture and changes over time. The authors conclude that providing a minimum image quality is most important and has to be ensured. Further, it was shown that optimal frame rates may be as low as 10 or 5 frames per second. Other investigations have been carried out by McCarthy et al. [87]. The conducted experiments focus on test videos for desktop computers and palmtops in two different resolutions, 352x244 for the desktop experiments and 176x144 for the palmtop experiments. The conducted surveys confirm that users tend to neglect a reduction of the frame rate, but that a decrease of picture quality leads to dissatisfied users.

Yamagishi et al. [85] also discusses the influence of the coded bit rate on the video quality. In their experiments, different image qualities are rated by users

and minimum bit rates needed for HD video transmission are identified. Recently, a subjective evaluation of scalable video coding for content distribution was published [88]. The authors investigated the impact of the different scalability parameters with user surveys. They used three different video clips and compared different quality configurations for the same bitrate. For the JSVM H.264/SVC codec they found out that users prefer lower resolutions instead of lower frame rates for high quality content. Further, as long as a certain image quality threshold is guaranteed a higher resolution is more important than image quality. The presented work was conducted during the course of this work and the presented results are in line with our investigations illustrated in Section 3.3.

The discussed work focuses on different aspects of our work, but lacks to combine the results for a QoE management of a SVC-based video streaming service. There is work on the impact of packet loss on video streaming [83, 84], but the authors do not include investigations on different video qualities. Yamagishi et al. [85] investigates the impact of packet loss and video quality in the H.264/AVC context, but does not include other scalabilities like resolution and frame rate.

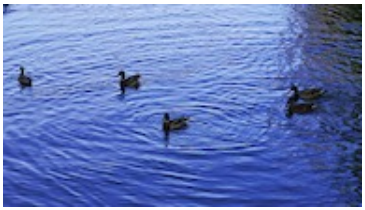
Other work on video qualities discusses the trade off between either quality and frame rate or resolution and frame rate [86,87], but investigates videos in low resolution. The optimal SVC configurations for a given bandwidth are provided by [88]. However, the results are not applicable for a QoE management, since such a mechanisms has to adopt the given configuration to the network conditions instead of the other way round.

3.2 Evaluation Methods

This section highlights the methods used to evaluate the user-perceived quality. First, we introduce the different video clips and classify the clips with respect to their spatial and temporal information. Subsequently, the methods for assessing the impact of controlled and uncontrolled disturbances on the video quality are described.

(a) blue sky

(b) crowd run



(c) ducks take off

(d) life

(e) old town cross

(f) park joy

Figure 3.7: *Screenshots of the test sequences.*

3.2.1 Investigated Video Clips

First, we introduce the investigated video clips and classify the clips with respect to their Temporal Information (TI),i.e., the amount of motion between two successive pictures, and Spatial Information (SI), i.e. the amount of detail per pictures, as introduced in ITU-T Recommendation P.910 [89].

Test Sequences

Our studies were performed using free high definition test video sequences, available at the Xiph.org Test Media website [90]. Since content types are expected to have a diverging influence on the perceived video quality, we choose scenes with varying types of content such as high or low motion and high or low detail. These scenes are *blue sky*, *crowd run*, *ducks take off*, *life*, *old town cross*, and *park joy*. Figure 3.7 depicts screenshots of all used scenes.

blue sky presents a treetop reach into the blue sky from left to right. The scene is filmed from the ground, the camera is slowly panning (cf. Figure 3.7(a)).

crowd run is a twofold scene. The lower part shows a running marathon crowd, the upper part a static background with spectators and trees (cf. Figure 3.7(b)).

ducks take off shows ducks swimming on the water. After a while they take off from water, causing water ripples (cf. Figure 3.7(c)).

life is a high contrast scene from a computer generated animated film. It starts with showing a bar with a bartender and customers in front. Then, the scene is switched to a tree with two static characters standing below it (cf. Figure 3.7(d)).

old town cross is a panning view of a big city. It is filmed from a tilted aerial view (cf. Figure 3.7(e)).

park joy shows a group of people run through a park, with trees passing by the camera in the foreground (see Figure 3.7(f)).

The SI and TI values for the used video clips are depicted as scatter plot in Figure 3.8. The figure highlights that the video clips cover a broad spectrum of temporal and spatial information. The scene *life* has a much higher TI value than the others. This is due to a scene cut in the middle of this video clip. For the other

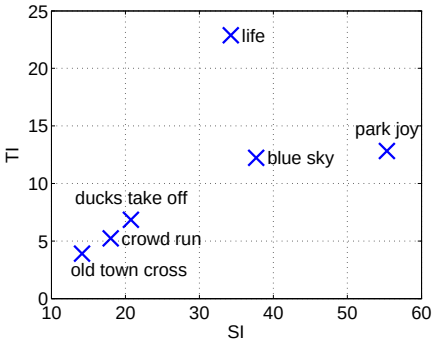


Figure 3.8: *Scatterplot of temporal and spatial information.*

scenes a higher temporal information yields also a higher spatial information. Next, the methods for generating disturbed video clips are presented.

3.2.2 Evaluation Methodology for Uncontrolled Distortions

In case of video live streaming with UDP as transport protocol, insufficient network conditions lead to a loss of data packets and thus a loss of video information. This results in frame artifacts, video freezing, and in case of a high packet loss, to a starvation of the video stream. We evaluate the impact of packet loss on the video quality using the EvalVid framework [91], a framework which ensures frame synchronization between a reference video clip and a distorted video clip. This allows us to use video quality metrics like PSNR and SSIM for evaluating the oQoE. The video streaming setup with the EvalVid framework is shown in Figure 3.9.

The framework transmits the H.264/AVC encoded video clip from the server to the client and creates an EvalVid trace file with details of the video transmis-

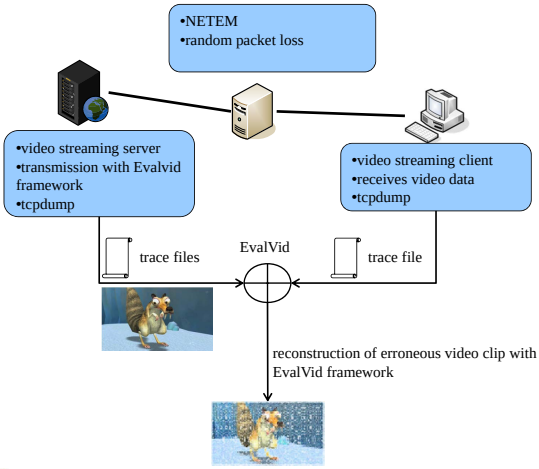


Figure 3.9: *EvalVid framework.*

sion. Additionally, *tcpdump* traces are recorded at server and client. Based on this traces and the video clip, the lost packets and frames can be identified and the erroneous video file can be reconstructed. In case of completely lost frames, the framework inserts the last displayed frame for every lost frame and thus creates a synchronized video clip which can be compared to the original video clip with video quality metrics. The EvalVid framework enables reproducible measurements for the influence of packet loss on H.264/AVC encoded video streaming. Consequently, we used this codec to emulate the behavior of packet loss on scalable video streaming. As discussed in Section 3.1.3, the results are quite similar. The encoding of the video clips to H.264/AVC was done with AutoX264 [92], an application mainly using mencoder and the x264 [93] codec. For the encoding process, we assured, that every 30th frame is an I-frame. Since we use a frame rate of 30 fps we assure that image failures spread over one second at maximum. The resolution of the video clips is similar to 720p HD. For the conducted mea-

surements we used the setup depicted in Figure 3.9. As operation systems Debian Sid with Kernel 2.6.26-2-686 is used for all three hosts. One host, a Pentium IV equipped with a 2.4 GHz processor and 1 GB RAM acted as video streaming server, and another, a Dual Pentium III with 2x 1.2 GHz and 512 MB RAM acted as client. The experiments were traced using tcpdump [94]. For investigating the influence of network conditions, i.e. packet loss, on the video degradation we used a software based network emulation approach. This is performed by the network entity located in the middle of our testbed, a Pentium III with a 0.8 GHz processor and 256 MB RAM, running Netem [95] as network emulation software. On this entity, we adjusted the desired random packet loss $p_l \in [0, 5\%]$. As models for evaluating the user-perceived quality for the disturbed video sequences we used PSNR and SSIM metric. An efficient implementation of these metrics is provided by the MSU Video Quality Measurement Tool [96].

3.2.3 Evaluation Methods for Controlled Video Distortions

The scalable video coding standard H.264/SVC allows an easy adjustment of an encoded video stream with respect to frame rate, resolution, or image quality. Thereby, applications or network elements are able to reduce the video quality and therewith the required bandwidth. As a result, network congestion and packet loss might be reduced or avoided. In order to estimate the impact of worse video quality on the user, we emulate typical SVC configurations with H.264/AVC clips in various resolutions, frame rates and SNR qualities.

For the spatial layers we use different resolutions for the video clips with respect to standardized formats. In particular we investigate the following resolutions: 1920x1080, 1280x720, 960x540, 640x360, and 480x270. For the investigation of the impact of different resolutions on the user-perceived quality we used a constant Quantization Parameter (QP) of 30 whereas QP denotes the accuracy of the transform coefficients resulting from the used Discrete Cosine Transform [97]. All sequences are encoded with a maximum frame rate of 30

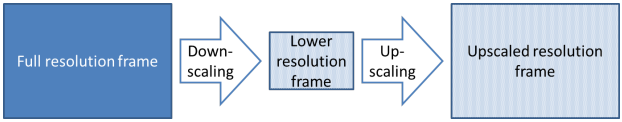


Figure 3.10: *Method of resolution comparison by upscaling.*

frames per second (fps) and a GOP size of 16 frames. Since the switching of temporal layers is achieved by removing every second frame to halve the frames per second, these settings allow for the following temporal layers: 30 fps, 15 fps, 7.5 fps and 3.75 fps. For the quality layers (in terms of QP) to be comparable, SVC files of only one spatial layer (resolution) have been created, spanning various QP values. The QP values range from 30 to 38 in steps of 2.

Since currently, no objective Video Quality Metrics exist that support the SVC codec format, the scenes are encoded in two ways. First, each scene is encoded in H.264/SVC with the various spatial, temporal and quality layers described above and the mean bit rates for each quality configuration are calculated. Then, each of the layer is encoded in H.264/AVC and the resulting video clips are compared to the reference clip in best quality.

Investigation of Different Resolutions

The scalable video coding standard enables video clips with spatial scalability, i.e., different resolutions. However, in order to compare video clips in different resolutions with objective video quality metrics, lower resolutions have to be upscaled to the reference resolutions. This is in-line with surveys indicating that users scale up the video clip to full screen, if possible.

Figure 3.10 illustrates the method used for resolution comparison. The full resolution video is first downscaled to the target resolution using different downscaling mechanisms. The lower resolution is transmitted to the user, where it is upscaled to the reference resolution again and then displayed.

When upscaling an image, empty space of unknown pixels relative to the scal-

ing factor is created between existing pixels. This space has to be filled with new pixels that correspond to their surrounding neighbors. There are different methods of interpolating the color and luminance values for those in-between pixels. Here, the different upscaling methods used in the investigation are described briefly. These are Nearest Neighbor interpolation, Bicubic interpolation, and Lanczos resampling.

Nearest Neighbor The nearest neighbor algorithm is the simplest upscaling mechanism. An interpolated pixel gets the color and luminance values of its nearest neighbor in one or two dimensions (x- and/or y-axis). For example, if the image size is doubled in both dimensions, each pixel of the original image is replaced by four pixels of the same color and luminance. This results in a conservation of all the original image detail, but leads to jagged and pixelated edges, so-called interpolation artifacts.

Bicubic This method considers the closest 4x4 neighborhood of known pixels. From those 16 pixels, a weighted average for color and luminance of the unknown pixel is calculated. Again, the weight of each pixel is determined by its distance to the unknown pixel. This method produces sharper images than the nearest neighbor algorithm, and does not suffer from the most prominent interpolation artifacts. Since this method is considered to be the best trade-off between resulting image quality and processing time, it is the most widely used upscaling mechanism. It is found in common applications such as digital zoom for compact cameras, in image manipulation software, and most video playback programs.

Lanczos is a sophisticated upscaling method which works very similar to Bicubic interpolation. The difference is in the weighting of the distance to the neighboring known pixels, and the sample size of pixels. For Lanczos upscaling, the distance is weighted by the normalized sinc function over a sample window defined by size a. This results in an $a \times a$ neighborhood, called the Lanczos window. This window is a horizontally stretched sinc

(a) Reference (b) Nearest Neighbour

(c) Bicubic (d) Lanczos

Figure 3.11: *Comparison of upscaling algorithms.*

function, $sinc(x/a)$ for $-a \leq x \leq a$. Based on the original image's pixel intensities $I(x, y)$, the unknown pixel $\hat{I}(x_0, y_0)$ is calculated with respect to the Lanczos window.

The described upscaling algorithms differ in complexity and quality of results. Figure 3.11 depicts the behavior of the upscaling mechanisms. Figure 3.11(a) illustrates the reference image. In Figure 3.11(b) the result of the least complex Nearest Neighbor upscaling is shown. Coarse pixels are clearly visible, producing roughly jagged edges. The result for Bicubic (Figure 3.11(c)) upscaling shows a smoother interpolation of pixels and edges, and differ mainly in blurriness of the

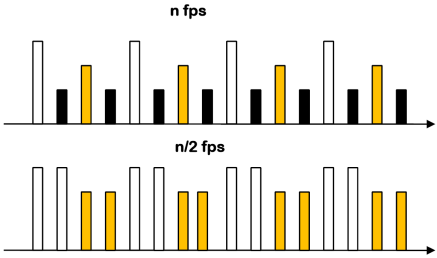


Figure 3.12: *Video with different frame rates.*

overall picture. The better quality comes to the costs of more complex upscaling mechanism. The most sophistic algorithm is Lanczos upscaling. Figure 3.11(d) shows that it results in much smoother edges and looks less blurry, presenting the best quality of all methods. Thus, we investigate the influence of the presented upscaling mechanisms on the user-perceived quality in order to provide a worst-, average-, and best-case approximation.

Investigation of Different Frame Rates

The video codec H.264/SVC allow also the adjustment of frame rates to reduce the required bandwidth. In particular, it is possible to remove a temporal layer and thus to halve the frame rate. In order to compare different frame rates with objective video quality metrics, we emulated the lower frame rate by removing a temporal layer and refilling the missing frames with predecessor frames. An exemplary video with a frame rate of n fps is depicted in Figure 3.12. The black frames contribute a frame rate of $n/2$ fps. For the same video clip with a frame rate of $n/2$, half of the frames are not displayed. The other frames are displayed longer on the screen, and the duration of the clip remains constant. This allows an evaluation of the video quality with objective metrics. Although low frame rates lower than 5 FPS surely denote a bad QoE, we aim at quantifying the impact of such frame rates and their potential for bandwidth saving.

Investigation of Different Image Qualities

For investigating the impact of quality scalability on the video quality we investigate video qualities for a quantization parameter ranging from 30 to 38, the higher this parameter, the lower the image quality. These are typical values used for the quality parameter [98, 99].

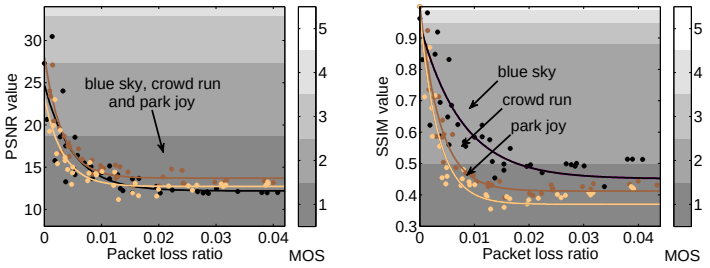
SVC Bandwidth Estimation

We encoded the videos in H.264/SVC with the JSVM software Version 9.15 in different spatial, temporal and quality layers. The base layer comprises a resolution of 480x270 pixels with a frame rate of 1.875 frames per second. This layer is extended by several temporal extension layers with 3.75, 7.5, 15, and 30 fps and by the spatial enhancement layers increasing the resolution up to 1920x1280 pixels. Further, we encode several video clips with maximum resolution and frame rate with varying QP parameters ranging from 30 to 38.

The maximum video quality of 1920x1280 pixels with 30 fps and a QP parameter of 30 is achieved if all layers are available. Based on the JSVM software, we computed the average bandwidth requirements for each of the different qualities. Based on these results, we discuss the trade-off between bandwidth and estimated video quality in the next section.

3.3 Results and Discussions

Our investigations indicated a minor influence of content on the video quality metrics in case of controllable and uncontrollable quality distortions. Thus, we focus on the results of our investigation for the video clips *blue sky*,*crowd run* and *park joy* in this section, since these clips cover all relevant findings.



(a) PSNR (b) SSIM

Figure 3.13: *Objective comparison for different packet loss ratios.*

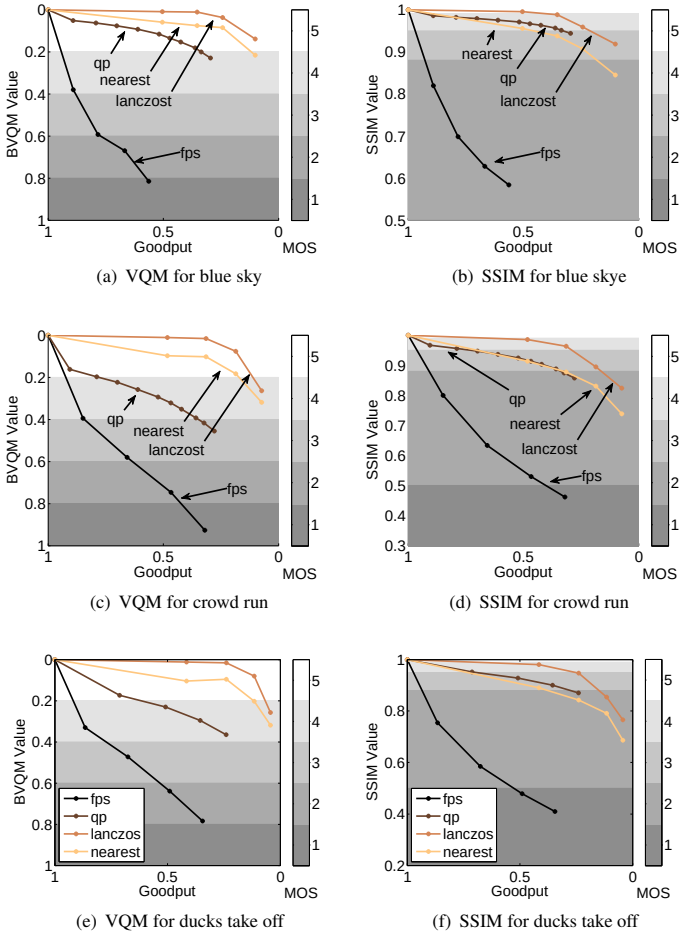
3.3.1 Impact of Uncontrollable Distortion

This subsection deals with the influence of packet loss on video transmission. For the conducted studies we rely on SSIM and PSNR as video quality metrics, since they provide an easy and fast computation of the video quality. An evaluation of the quality with VQM lasts more than 8 hours compared to less than 5 minutes with SSIM and PSNR. We conduct experiments with different content types and varying packet loss ratios, as explained in Section 3.4. We focused on random packet which affects the quality more severe than burst loss, cf. [83]. The results for the objective metrics and the corresponding subjective MOS values are depicted in Figure 3.13. In both figures, the y-axis denotes the objective QoE indicators, while the different MOS values are illustrated by the areas with different gray colors. The PSNR value, depicted in Fig. 3.13(a), decreases with increasing packet loss ratio. The same behavior is investigated for the SSIM metric, shown in Fig. 3.13(b). It can be seen that the PSNR values decrease equally for different content. Hence, PSNR can not differ between different content, which is due to the fact that it is computed as ratio between the maximal error and the average error. This inability of PSNR are in accordance with the results obtained by Huynh-Thu et al. [100]. The authors also show that PSNR cannot be a reliable

method for assessing the video quality across different video contents. On the other hand, SSIM is able to distinguish different content as can be seen in Fig. 3.13(b). This is mainly due to the fact that SSIM is comparing the video content with respect to structures. Here, blue sky is affected least by packet loss, followed by the crowd run sequence. We can conclude that the gradient of the SSIM curves for increasing packet loss differs for the investigated content types, i.e. QoE control has to take the type of content into account. Both metrics indicate that already a small packet loss probability decreases the user perceived quality drastically. Referring to SSIM, a packet loss rate $p_l = 0.7\%$ yields to a MOS value of 3, and a MOS value lower than 2 is obtained for $p_l < 0.5\%$. Thus, we can conclude that for the given encoded video clips and scenarios, packet loss should be avoided in any case.

3.3.2 Impact of Controlled Distortion

This section highlights the results of our investigation on controlled video distortion, i.e., bandwidth and quality reduction with respect to resolution, QP parameter and frame rate. The results presented in the following are computed with SSIM and BVQM, a software for computing the VQM value. For both metrics the results are shown for each video clip. The reference video is encoded in 1920x1280 pixels with 30 fps and a QP parameter of 30. Thus, the maximum score for each metric is achieved if all layers are available. Based on these results, we discuss the trade-off between bandwidth and estimated video quality. On the x-axis the required throughput, i.e., the *goodput*, is depicted, on the y-axis the resulting metric values. The figures contain again areas with different gray colors which illustrate the corresponding MOS values. For the upscaling mechanisms we show results for Nearest Neighbor and Lanczos upscaling, i.e., for the simplest and the most sophisticated mechanism. The results for the three video clips are depicted in Figure 3.14. First, we investigate the video clip *blue sky* in detail. Figure 3.14(a) highlights the results for VQM, Figure 3.14(b) for SSIM respectively. It can be seen that for a bandwidth reduction by reducing the resolution the VQM value



(a) VQM for blue sky

(b) SSIM for blue skye

(c) VQM for crowd run

(d) SSIM for crowd run

(e) VQM for ducks take off

(f) SSIM for ducks take off

Figure 3.14: *Results for the video clips blue sky, crowd run, and park joy.*

and the corresponding MOS value are affected the least. Lower image qualities and the corresponding bandwidth savings lead to a higher decrease of video quality as compared to the case where the resolution was reduced. The video quality is disturbed most in case of a frame rate reduction. Similar results are obtained when using SSIM metric, as illustrated in Figure 3.14(b). The results only differ with respect to the QP parameter. SSIM indicates that upscaling with the nearest neighbor algorithm leads to a lower video quality as varying the QP parameter. By taking a closer look at the estimated user perceived quality on a MOS scale we see that the lowest resolution of 480x270 pixels still yields to a MOS values larger than 3, independent of the upscaling method. This seems to be a best-case approximation of the MOS value, since the corresponding MOS calculated with SSIM is $2-3$, depending on the upscaling method. Both video quality metrics show a high impact of the frame rate on the user perceived quality. For the lowest frame rate, 1.875 frames per second, the estimated MOS values is 1 for both metrics indicating the worst possible user rating. Similar observations can be derived for the other video clips. For all investigated cases spatial scalability performs better or similar than quality scalability in case of SSIM and VQM. Further, temporal scalability performs much worse than the other two scalabilities. Similar results are obtained for *crowd run*, depicted in Figure 3.14(c) and Figure 3.14(d) and *park joy*, as illustrated in Figure 3.14(e) and Figure 3.14(f).

3.4 Lessons Learned

The objective of this chapter was to provide insights for a QoE management mechanism for a future video streaming service based on scalable video coding. Therefore, we introduced the video codec H.264/SVC and discussed its capabilities to adapt the video quality to network conditions by adapting either frame rate, resolution, or image quality. These three degrees of freedom are called temporal, spatial, and quality scalability. Further, we introduced a test methodology which allows an investigation of different types of video distortion caused either by packet loss or the video scalabilities and analyzed different video clips.

As expected, insufficient network conditions, which lead to packet loss, have a severe influence on the video quality. This is already the case for packet loss rates lower than 0.1% the video quality and therewith the user perceptions is affected strongly. Depending on the type of content, packet loss rates between 0.5% and 2% a MOS value of 1, i.e., the worst possible video experience.

Further, it was shown that reducing the video quality in a controlled way has a lower impact on the video quality than for the case of packet loss. Our evaluations show that lower resolutions lead to a better video quality than lower image qualities and frame rates. Further, video quality decreases fastest in case of lower frame rates. These results are in-line with subjective evaluations for H.264/SVC, which were published after our investigations.

We can conclude that packet loss has a huge impact on user perceived quality for the case of HD video streaming. For packet loss rates larger than 0.5% the video streaming experience is strongly disturbed and users will not accept this service quality. Secondly, we have seen that bandwidth savings can be achieved best by reducing the video resolution in case of HD video streaming. Lower image qualites and frame rates lead to a lower perceived quality and lower possible bandwidth savings.

4 Multipath Transmission and Scalable Video Streaming

This chapter combines the investigations from the previous chapters on concurrent multipath data transmission and the evaluation of the user perceived quality for video streaming with scalable video codecs and discusses the interaction of both presented mechanisms.

In case of insufficient available network resources for the video streaming service, different solutions are possible. On the one hand, the network can provide additional resources using mechanisms like multipath transmissions as discussed in Chapter 2, if possible. If such additional resources are not available, techniques like H.264/SVC, discussed in Chapter 3 allow the easy adaptation of the video quality to the available resource, either at the edge or within the network.

In this chapter the application of the QoE Provisioning-Delivery Hysteresis (PDH) for the case of video streaming is shown and the implications on service delivery discussed. We show how the overall quality of a service can be improved if applications and network interact and discuss interfaces allowing for this interaction.

This chapter is structured as follows. Section 4.1 introduces the Provisioning-Delivery Hysteresis. The application of the PDH for video is discussed in Section 4.2. This is followed by a detailed discussion on network and service adaptation for a scalable video streaming service and a flexible network capable of multipath transport in Section 4.3. Subsequently, current and future modes of network and service interaction are discussed in Section 4.4. This is followed by a presentation of our proof-of-concept implementation of a multipath video streaming service

using SVC in Section 4.5. The lessons learned are summarized in Section 4.6.

4.1 Provisioning-Delivery Hysteresis

This section summarizes the Provisioning-Delivery Hysteresis and its underlying differential equations [101], which are the background for our investigations in Section 4.2.

4.1.1 Notions

First, we introduce a couple of notions of importance for the remainder of this work. Subjective measures of QoE that grow with the degree of user satisfaction are called *satisfaction rating*. A typical example is the mean opinion score as measure for the overall quality of, e.g., a web service [102]. Regarding the QoS measures, we need to distinguish between different sensitivities as follows:

- A *failure measure* $\bar{Q}oS_{\mathrm{f}}$ grows with degree of failures. Most QoS parameters are of this kind and related to packet delivery problems. Prominent examples are *loss ratios* L, jitter measures, reordering ratios or waiting times.
- A *success measure* QoS_{s} is the opposite of a failure measure; it grows with the degree of success using a resource. Examples are availability measures, often given by the numbers of nines (e.g. four nines denote 99.99 % availability), or the *packet delivery ratio* $D = 1 - L$.
- A *resource measure* QoS_{r} is similar to the success measure in the sense that it grows as conditions improve. However, the reason for the improvement is rather found in an increased amount of resources than in better success in using the resources. A typical resource measure is the provisioned *throughput*.

Further, we distinguish between two different kinds of quality degradation,

- *uncontrolled quality degradation*, i.e. an uncontrolled adaptation to a bottleneck capacity, typically resulting in packet loss;
- *controlled quality degradation*, i.e. a controlled adaptation to a bottleneck capacity either through protocols like TCP, or content adaptation like codecs with lower bandwidth.

These impacts are considered separately in the following.

4.1.2 QoE Delivery – Impact of Uncontrollable Quality Degradation

This subsection highlights the impact of $\bar{Q}oS_{\mathrm{f}}$ and QoS_{s} on the QoE with the help of the partial differential equations presented in [101]. The relationship between *satisfaction rating* and *failure measure* can be described as either exponential or logarithmic behavior:

$$QoE(\bar{Q}oS_{\mathrm{f}}) = \alpha + \beta \exp(-\gamma \bar{Q}oS_{\mathrm{f}}) , \tag{4.1}$$

$$QoE(\bar{Q}oS_{\mathrm{f}}) = \alpha - \beta \left|\log(\bar{Q}oS_{\mathrm{f}})\right| , \tag{4.2}$$

where α, β and γ are non-negative values. Although these coefficients are used in all formula (4.1−**??**), their value may differ. Both functions are of convex nature and have similar shapes. For the sake of simplicity, we use the exponential relation between *satisfaction rating* and *failure measure* in the following. Nevertheless, the results and discussions also apply for the logarithmic relationship.

Equation 4.1 and **??** are extended to a relationship between *satisfaction rating* and *success measure*:

$$QoE(QoS_{\mathrm{s}}) = \alpha + \beta \exp(\gamma QoS_{\mathrm{s}}) , \tag{4.3}$$

$$QoE(QoS_{\mathrm{s}}) = \alpha + \beta \left|\log(\bar{Q}oS_{\mathrm{f}})\right| , \tag{4.4}$$

indicating a relation between the packet delivery ratio and the user-perceived quality.

4.1.3 QoE Provisioning – Impact of Controllable Quality Degradation

This subsection highlights the impact of QoS_r on the user perceived QoE, as investigated in [101]. Here, again an exponential and a logarithmic relationship were provided by resolving partial differential equations presented in [101]:

$$QoE(QoS_\mathrm{r}) = \alpha - \beta \exp(-\gamma QoS_\mathrm{r}) \,, \tag{4.5}$$

$$QoE(QoS_\mathrm{r}) = \alpha + \beta \left|\log(QoS_\mathrm{r})\right| \,. \tag{4.6}$$

The functions presented in Equation 4.5 and Equation 4.6 are of concave nature and have similar shapes. This behavior for providing better quality by increasing the required bandwidth was investigated by [15, 85, 103] for instance. We use the exponential relationship between the *satisfaction rating* and the *resource measure* in the following. However, results and discussions also apply for the corresponding logarithmic relationship.

4.1.4 The QoE Provisioning-Delivery-Hysteresis

In this section, we highlight the different impacts that provisioning and (lack of) success of packet delivery have on QoE. We consider the *goodput* Φ, i.e., the application perceived throughput, as joint parameter. Furthermore, the *goodput ratio* x is defined as the relative goodput as compared to an optimal value C for which the QoE is maximized, i.e. $x = {}^{\Phi}/_{C}$. For $x = 1$, the observed QoE is maximal and cannot be improved by higher goodput.

Goodput and goodput ratio can be affected by:

- controllable quality distortion through resource allocation $QoS_\mathrm{r} < C$, yielding $x = QoS_\mathrm{r}/C$;

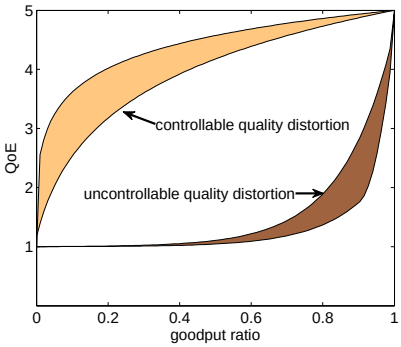


Figure 4.1: *Illustration of the QoE Provisoning-Delivery Hysteresis.*

- uncontrollable quality distortion through data loss $\bar{Q}oS_{\mathrm{f}}$, yielding $QoS_{\mathrm{r}} = 1 - \bar{Q}oS_{\mathrm{f}}$ and thus $x = QoS_{\mathrm{s}}/C = (1 - \bar{Q}oS_{\mathrm{f}})/C$.

Figure 4.1 sketches the postulated hysteresis as a set of functions of the goodput ratio. While specific relationships between QoE and goodput ratio depend amongst others on application and context, we observe two fundamentally different areas. Controllable quality distortion allows for keeping the QoE rather high in view of considerable savings, i.e. goodput ratios much smaller than one. However, this may not hold for any kind of provisioning, as we will see in Section 4.2 for frame rate reduction in case of video streaming. Uncontrollable quality distortion, however, yields a completely different behaviour. Small decreases in the goodput ratio imply large decreases in the QoE values, while this decrease flattens out at the lower edge of the QoE scale as the goodput ratio sinks. This implies that, in order to ensure good QoE, controlled actions are superior over problems appearing in an uncontrolled way. In [101] the QoE-PDH was shown for VoIP and web browsing, in the following we apply it for live video streaming.

4.2 The Provisioning-Delivery Hysteresis for the Case of Video

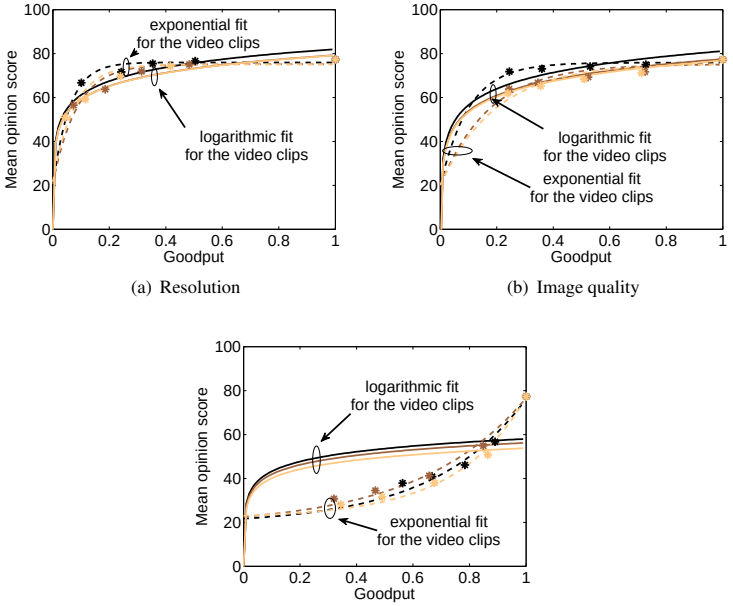
This section investigates if the PDH can be applied for the case of video streaming. Accordingly, the results from Chapter 3 are plotted with respect to the application layer perceived goodput, as introduced in Section 4.1.4.

4.2.1 Methods used

To estimate the influence of controlled and uncontrolled video distortion, we rely on the results observed with SSIM as discussed in Chapter 3. To estimate the user perceived quality, we use an SSIM to MOS mapping function presented in [104]. This exponential fitting function $f(x) = 13.91 \cdot e^{1.715 \cdot x}$ is intended to map SSIM values, ranging from 0 to 1, to a mean opinion score ranging from 0 to 100. However, it has to be noted that the used mapping function allows for MOS values $MOS \in [f(0); f(1)] = [13.91; 77.29]$. Accordingly, the results are presented on a scale of 1 to 100 instead of 1 to 5.

4.2.2 Provisioning Curves

First, we briefly discuss the impact of throughput reduction on the perceived user quality. We investigate different resolutions, image qualities and frame rates and their required average capacity for three different video clips. In particular we use the resolutions 1920x1080, 1280x720, 960x540, 640x360, and 480x270, the QP parameters from 30 to 38 in steps of 2 and the frame rates 30, 15, 7.5 and 3.75 fps for the video clips *blue-sky*, *crowd-run* and *park-joy*. To be able to compare different resolutions, we used the bicubic upscaling mechanism as introduced in Chapter 3. The results as well as exponential and logarithmic fitting functions are depicted in Figure 4.2. The x-axis shows the goodput, i.e. the average throughput which is required to provide the quality depicted on the y-axis as mean opinion score. For the case of quality adaptation by decreasing the resolution of a



(a) Resolution

(b) Image quality

(c) Frames per second

Figure 4.2: *Provision curves for different resolutions, image qualities and frame rates.*

video clip, which is depicted in Figure 4.2(a), high bandwidth savings are possible whereas the impact on the user perceived quality is negligable. According to our evaluation, a bandwidth reduction of 50% still yields to user perceived quality of more than 90% of the maximum QoE. The provisioning curves can be fitted logarithmical or exponential, which is illustrated by the solid and the dashed lines. The parameters of the fitting functions are illustrated in Table 4.1, as well as the Mean Squared Error (MSE) as estimator for the goodness of the fit. With respect to the investigated scenario, the logarithmical fitting is sometimes better

Table 4.1: *Fitting functions for resolution reduction.*

Video	MOS	MSE
blue_sky	$9.064 \cdot \ln(8487.5 \cdot x)$	0.682
blue_sky	$56.935 \cdot \exp(-17.114 \cdot x) + 75.957$	0.153
crowd_run	$8.762 \cdot \ln(8487.5 \cdot x)$	0.044
crowd_run	$54.678 \cdot \exp(-11.875 \cdot x) + 75.126$	0.145
park_joy	$8.738 \cdot \ln(8588.8.8 \cdot x)$	0.036
park_joy	$53.554 \cdot \exp(-13.333 \cdot x) + 75.004$	0.112

Table 4.2: *Fitting functions for image quality reduction.*

Video	MOS	MSE
blue_sky	$8.810 \cdot \ln(8366.5 \cdot x)$	1.612
blue_sky	$56.372 \cdot \exp(-10.014 \cdot x) + 75.823$	0.334
crowd_run	$8.406 \cdot \ln(8365.8 \cdot x)$	0.044
crowd_run	$55.182 \cdot \exp(-5.734 \cdot x) + 75.133$	0.209
park_joy	$8.324 \cdot \ln(8365.8 \cdot x)$	0.065
park_joy	$55.309 \cdot \exp(-5.078 \cdot x) + 75.4142$	0.176

than the exponential fitting. Similar observations hold for the quality adaptation by reducing the image quality which is depicted in Figure 4.2(b). The parameters for the fitting functions as well as the MSE are shown in Table 4.2. Similar to the fitting functions for resolution, the goodness of the fittings varies for the investigated video clips. Whereas the degradation of resolution and quality yield to concave curve, decreasing the frame rate yields to convex curves as illustrated in Figure 4.2(c). Accordingly, a logarithmical function, as defined in Equation **??** seems inappropriate. The parameters for the computed fitting functions are given in Table 4.3.

Delivery Curves

If the network is not able to fulfill the service requirements, e.g., bandwidth requirements, jitter, or an in-order transmission, delivery problems occur. Thus,

Table 4.3: *Fitting functions for frame rate reduction.*

Video	MOS	MSE
blue_sky	$6.423 \cdot \ln(8364.9 \cdot x)$	0.754
blue_sky	$-4.390 \cdot \exp(2.611 \cdot x) + 15.616$	0.031
crowd_run	$6.222 \cdot \ln(8364.8 \cdot x)$	0.684
crowd_run	$-5.419 \cdot \exp(2.403 \cdot x) + 16.359$	0.012
park_joy	$5.955 \cdot \ln(8364.7 \cdot x)$	0.722
park_joy	$-1.813 \cdot \exp(3.431 \cdot x) + 19.964$	0.016

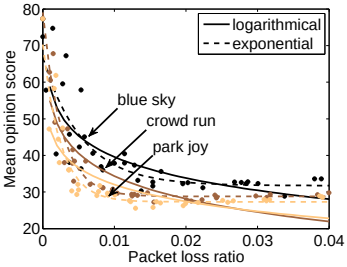


Figure 4.3: *Impact of packet loss on the perceived quality.*

packets and information are dropped or delayed, which affects the user perceived quality to a high extend. The results for uncontrollable video streaming distortions from Chapter 3 for the video clips *blue-sky*, *crowd-run* and *park-joy* are taken as reference. The results are illustrated in Figure 4.3. The figure indicates that an already very small amount of packet loss $p_l < 1\%$ has a severe impact on the MOS values and thus on the user perceieved quality. Exponential fitting functions are depicted in Table 4.4, together with the MSE. Note that the packet loss value zero was excluded from the logarithmic investigation due to the singularity of the log function. For the investigated use cases, an exponential fitting performs slightly better than a logarithmical in terms of MSE.

Table 4.4: *Fitting functions packet loss.*

Video	MOS	MSE
blue_sky	$35.007 \cdot \exp(-181.8174 \cdot x) + 31.7335$	0.189
blue_sky	$-8.332 \cdot \ln(1.278 \cdot x) + 3.271$	0.382
crowd_run	$50.26 \cdot \exp(-382.509 \cdot x) + 28.795$	0.033
crowd_run	$-8.056 \cdot \ln(3.659 \cdot x) + 4.727$	0.277
park_joy	$42.980 \cdot \exp(-396.213 \cdot x) + 27.305$	0.080
park_joy	$-6.643 \cdot \ln(2.1911 \cdot x) + 6.640$	0.42

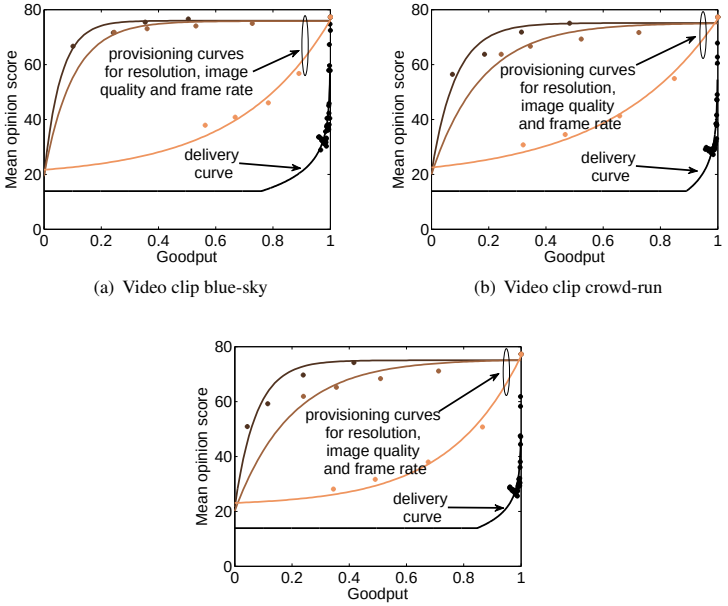
Provisioning-Delivery Hysteresis for Video Streaming

Now, we combine the results for the provisioning and the delivery curves. For this purpose, we investigate the curves with respect to the effective goodput for the application. The results are plotted in Figure 4.4, showing MOS values as function of the goodput. We can recognize two sets of curves:

1. the upper ones, emanating from variations of the resources, namely of video quality with respect to resolution, image quality and frame rate. While the curves for resolution and quality reduction are both concave ones, the curve for the frame rate is convex.

2. the lower, convex one, emanating from goodput reduction due to losses.

The variation of resolution and image quality is similar to the upper, concave curve of the generic hysteresis discussed in Section 4.1, the provisioning part of the hysteresis. The quality reduction due to losses is similar to the shape of the delivery part of the hysteresis. However, the variation of frame rate does not really fit to the hysteresis due to its convex shape. As discussed, the shape of a variation of resources should yield to a concave behavior. However, frame rate variation still outperforms quality degradations due to packet loss as illustrated in Figure 4.4. Although this kind of quality degradation has a high impact on the user perceived quality, and thus the other possibilities should be used to reduce service quality, it is still better than getting hit by uncontrollable packet loss. Thus, it is



(a) Video clip blue-sky

(b) Video clip crowd-run

(c) Video clip park-joy

Figure 4.4: *Provision-Delivery Hysteresis for the video clips blue-sky, crowd-run and park-joy.*

also important to know how to provision the service in order to maximize the user perceived quality in case of insufficient network resources.

As can be seen, a reduction of the quality in order to adapt the video bandwidth to insufficient network resources in terms of end-to-end throughput results in still a high good QoE. However, as soon as packet loss appears, the video quality is strongly impaired. As illustrated in Figure 4.4, the QoE-PDH is demonstrated for video streaming. The QoE-PDH for video streaming can be extended by, e.g, taking Forward Error Correction (FEC) mechanisms into account. Such mechanisms

enable a protection of the video stream against packet loss, but require additional bandwidth. Thus, they increase the bandwidth requirements, which amplifies the congestion in the network. Although the quality of the video service might be enhanced, other services suffer more and may try to use similar mechanisms to improve the user perceived quality. In the resulting arms race, no service and also no user will finally benefit. In order to prevent such additional bandwidth requirements, such mechanisms could be combined with provisioning mechanisms. In such a way it would be possible to maximize the QoE for given packet loss and bandwidth tuples. The QoE-PDH does not take these prospects into account and its extension should be part of further research in this area.

4.3 Evaluating Application-Network Interaction for Multipath Video Streaming with H.264/SVC

The previous sections presented the QoE-PDH for video streaming based on H.264/SVC and showed how a video streaming service can adapt to network conditions. Furthermore, in Chapter 2, we discussed towards a multipath mechanisms which allows for bundling capacity within the network, i.e. providing more network resources. If network and application are flexible and can adapt to application demands or network state, the question arises how such an adaption should be done. In the following, we discuss this question for the example of a video streaming service based on H.264/SVC and an exemplary network setup with two available paths, as illustrated in Figure 4.5. Each path is able to provide resources for a number of x video streams, e.g., for the the equivalent of 1.5 video streams. We assume that the capacity of both paths can be bundled by a multipath transmission as disccused in Chapter 2. Further, the quality for each customer can be decreased seamlessly, i.e. each customer gets the same share of bandwidth and the video can adapt to this share. We can now investigate the impact of different interaction policies on the quality per customer. In particular, we study four different policies, (1) no service or network interaction in case

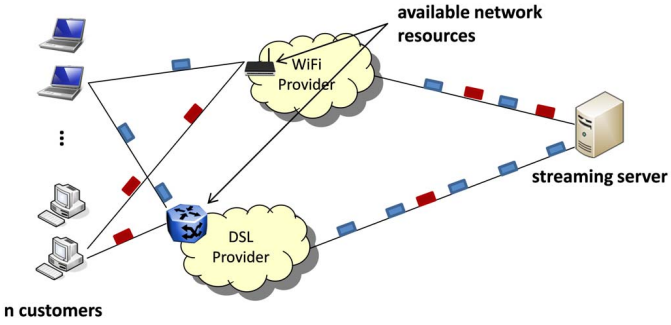


Figure 4.5: *Scenario set-up.*

of insufficient resources, (2) the network reacts to the bottleneck by increasing the capacity with a concurrent transmission via a second path, (3) the service reacts by reducing the quality of the video stream, (4) first the network reacts by providing additional capacity, then the service decreases its demands.

For this study we use the clip *crowd-run* as reference with the fitting functions as described in Section 4.2. For the delivery curve, i.e. for the case of packet loss we use the corresponding logarithmical fitting, for the provisioning part we use the exponential fitting for the quality degradation by reducing the resolution. The results of the study for a link providing network resources for the equivalent of 1.5 video streams are depicted in Figure 4.6. The x-axis denotes the number of customers using the video streaming service, the y-axis the average quality per customer.

For the case of no interaction between applications and network, the application quality is sufficient for one customer, but already for two customers, the quality per customer drops rapidly. The network resources are insufficient and neither the application nor the network adapt to the congestion state. Thus, a huge amount of packets are lost, resulting in low service quality. If the network reacts to the congestion by providing additional resources, the application can be

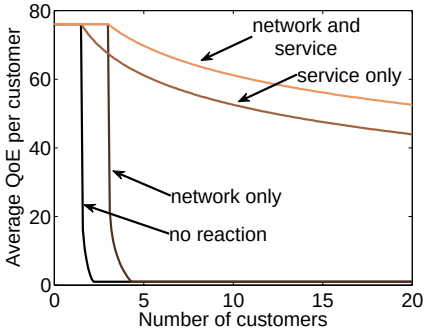


Figure 4.6: *Results of the evaluation.*

provided to more customers without impairments. However, for more than three customers, the service quality drops rapidly. Again, the available network resources are not sufficient, congestion and packet loss occur resulting in a low service quality per customer. Otherwise, the service can adapt to the congestion by reducing the service requirements, i.e., reducing the quality of the video stream. Thus, the network can be relieved and recover from the congestion. Accordingly, no packets are lost and the huge impact of packet loss on video streaming can be avoided. However, in case of service quality adaptation, the quality of the video stream is reduced due to the adaption to the available resources. As illustrated by Figure 4.6, the best quality for customers can be achieved if both, service and network, adapt to the rising number of customers. In this case the network reacts to the congestion at foremost ensuring that the service quality can be kept to the maximum. However, if further customers join the system no additional network resources can additionally be provided. The network may signal this to the service which then reduces its requirements, i.e., the service quality. Thus, the average service quality per customer can be maximized.

The results indicate a high potential to increase the user perceived quality for the interaction between networks and applications. How this interaction is real-

ized in the current Internet, and efforts towards future methods of interaction are discussed in the following section.

4.4 Implementation of Application and Network Interaction

First, we briefly present two applications in the current Internet and discuss how these applications try to overcome the missing application-network interaction. Subsequently, we discuss approaches defining future interfaces enabling the close cooperation between applications and networks. This also includes two future Internet architectures which are based on the tight coupling of applications and networks.

4.4.1 Interaction in the Current Internet

In the current Internet, information asymmetry [105] prevails due to the missing information exchange between network and service providers as illustrated in Figure 4.7. The service does not take the underlying physical infrastructure into account and the network infrastructure does not know the requirements of the service. For the case of a streaming service, the server could reduce the video bitrate, if there is a bottleneck within the network, in order to cope with congestion. This could be achieved by signaling the video streaming server *before* congestion occurs. Vice versa, if the application communicates its service requirements, e.g. in terms of average throughput, the network provider could try to provision the service requirements. Two examples trying to overcome this information asymmetry in the current Internet, the Skype application and P2P networks, are briefly discussed in the following.

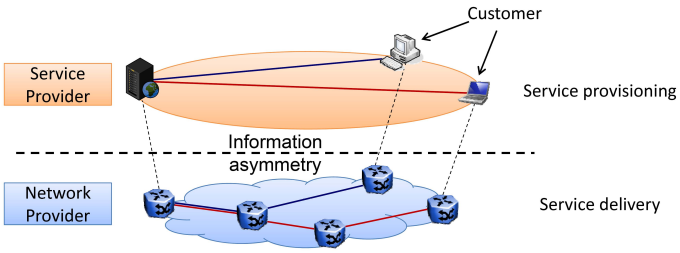


Figure 4.7: *Information asymmetry between service and network provider.*

The Skype Application

Streaming multimedia applications face the problem that their predominant transport protocol UDP does not take into account any feedback from the network. Consequently, any quality control and adaptation has to be done by the application itself at the edge of the network. The network providers have to cope with the fact that these edge-based applications dynamically determine the amount of consumed bandwidth. In particular, applications such as Skype which is mainly used for VoIP or video conferencing do their own network quality measurements and react to quality changes to keep their users satisfied. In other words, edge-based applications perform QoE management and traffic control on application layer. Thus, Skype distinguishes between packet losses and network congestion and reacts either by protecting the data streams with Forward Error Correction (FEC) mechanisms or by switching codecs and codec configurations to reduce the required bandwidth [106, 107]. However, this QoE management comes at the costs of sophisticated network measurements which are implemented by the Skype application.

P2P Networks

Another prominent example currently discussed in literature is the information asymmetry emerging in P2P networks [105, 108, 109]. The overlay connections used by these networks are up to now generally network-agnostic and therefore wasteful with resources. For such a network it does not matter if the required data are downloaded from a peer located in the same domain or in the domain of another Internet Service Provider. However, the network provider would prefer communication with peers in his own domain since the usage of transit links between Internet Service Providers is costly in general.

Therefore, it is desired that the underlay provides some kind of information to the overlay application. The aim is to support traffic management of the overlay application and to prevent any negative effects on both parties caused by the information asymmetry, i.e. increased costs and reduced QoE. In any case, an information exchange must be able to lead to a "win-win" scenario for all parties involved. The prioritization is the result of an economic decision function which takes into account both requirements: reduction of provider costs and improvement on users' QoE.

4.4.2 Application and Network Interaction in Future Networks

In this section we briefly discuss research towards new interfaces allowing a better interaction between applications and networks, as well as two future network approaches implementing a better interconnection between applications and networks.

A Future-Proof Application-to-Network Interface

A Future-Proof Application-to-Network Interface enhances the drawbacks of the current socket API by providing a requirement-based application interface [110]. Application programmers need much networking know-how since they have to

specify the right transport protocol, have to perform name-to-address resolution and have to implement application-layer protocols. This hinders adaption of new network and transport protocols. Accordingly, applications should simply state their requirements, e.g., minimal bandwidth, to the network stack. The necessary functionalities like the transport protocol selection are pushed down below this API. Thus, new protocols, name-to-address resolution mechanisms or other functionalities can be introduced without modifying the application.

A G-Lab Application-to-Network Interface

The G-Lab API (GAPI) [111] is very similar to the Future-Proof Application to Network Interface described in [110]. In particular it extends this API to integrate three different Future Internet architectures, namely Sonate [112], Forwarding on Gates [113] and Nena [114]. The G-Lab API enables the network to push measurement information to the applications and enables the application to specify functional and non-functional requirements, like "bandwidth >= 150kbit/s and loss rate < 2%". If these requirements cannot be fulfilled (e.g. indicated by a network observer, which is situated in the stack) the network will inform the applications. Thus, the application service may adapt to the network conditions without implementing active and passive measurement mechanisms.

AquareYoum - *A*pplication and *Qua*lity of Experience-Aware *Re*source Management for *You*Tube in Wireless *M*esh Networks

AquareYoum is an example for YouTube application and QoE-aware resource management in wireless mesh networks [115] where multiple links and gateways to the Internet exist. An additional browser plugin enables the approximation of the playout buffer and provides these information to the wireless network, in particular to the wireless mesh controller entity. This entity monitors the network and knows about the resource utilization of the available links. If the browser plugin detects a problem on service level, e.g., the playout buffer decreases and falls under a certain threshold, a notification is forwarded to the mesh controller.

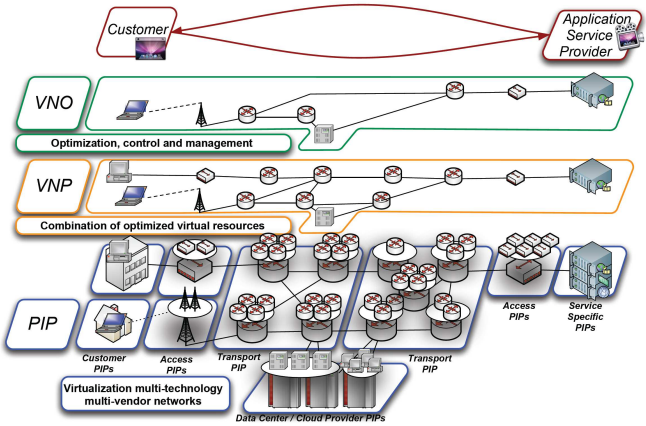


Figure 4.8: *An overview of the roles and architecture considered in the COMCON project [116].*

The controller now tries to reroute the YouTube flow via a less congested link, either via the same or another gateway. Subsequently, the YouTube video can be prevented to stall which reduces the user perceived quality.

COMCON Architecture

In the G-Lab [27] phase 2 project COMCON, we enhance the role model of the Architecture and Design for the Future Internet (4WARD) project and identify key functions in a potential vertical market based on these concepts. The different roles and their interaction assumed in the COMCON architecture are illustrated in Figure 4.8. The role model comprises Physical Infrastructure Providers (PIPs), Virtual Network Providers (VNPs), Virtual Network Operators (VNOs), and Application Service Providers (ASPs). A PIP is the owner of its physical resources

and uses virtualization techniques to rent shares of these physical resources to VNPs. These virtual topologies provided to a VNP include links as well as virtual nodes providing storage and computation resources. The PIP also provides means to configure and control these virtual nodes. A VNP is a broker and an aggregator that typically acquires resources from multiple PIPs while delivering a single contiguous Virtual Network (VN) to a VNO. A VNO designs the VN, defining its topology and the functionality to be provided by each node. During operation, the VNO configures the functionality of the nodes in a VN using the control means forwarded from the PIPs. Thus, the VNO defines the shape as well as the inner configuration of the VN and, thereby, tailors the network to the needs of the service envisaged by the ASP. The clear definition of these four roles allows to identify potential and necessary interaction and cooperation. Nevertheless, the modeled roles do not need to be actually represented by different players, in real scenarios several roles may also consolidate in one entity. To set up a new VN, the roles interact as follows. The ASP formulates its demands and mandates the VNO that makes the best and most suitable offer to realize the requested service. In turn, the VNO requests a VNP who contacts a PIPs. As we consider automation to be a key enabler for future networks, all communication for the set-up as well as the operation of a VN is assumed to use standardized interfaces and corresponding protocols. Accordingly, the ASP and the network providers can use these interfaces to exchange application and network information.

One of the benefits of such an architecture based on Network Virtualization (NV) is that it allows networks to dynamically grow or shrink. Thus, the ASP can easily extend its network if more and more customers join the service.

4.5 Proof-of-Concept Implementation of Multipath SVC Streaming

To show the capabilities of multipath video streaming based on H.264/SVC we implemented a software framework enabling multipath functionality either at the edge or within the network.

Multipath transmission implemented in the network requires additional functional blocks, namely a flow splitter and a flow assembler. The splitter uses scheduling techniques like round-robin to distribute the packets via the different paths, the assembler reassembles the different subflows to the previous flow. We implemented these functionalities and also mechanisms to cope with out-of-order packets and assure a valid byte stream for the video player. Further, the capabilities of an SVC stream to split the flow into base layer and enhancement layers were used to transmit different layers via different paths. This extends pure round-robin packet scheduling techniques as investigated in Chapter 2. However, we had to extend the flow assembler to be able to identify base and enhancement layers in order to reassemble the complete stream or forward only a part of the stream if parts are missing. Due to SVC dependencies, we further had to assure that the quality within a GOP does not oscillate since quality changes within a GOP may lead to distortions during play back. Thus, we had to provide a larger buffer and therewith additional delay at the assembler than compared to the case of round-robin packet scheduling. We integrated the implemented multipath streaming framework into the demonstration of the COMCON project at the G-Lab Status Meeting during the Euroview 2011 event [117]. During the course of the demonstration, the capabilities of a dynamic Network Virtualization Infrastructure based on GMPLS and of an elaborated service provisioning architecture, the multipath SVC streaming architecture, were shown. This included the automatic network management and resource control for virtual networks for involved roles as described in Section 4.4.2. As a starting point the physical components were virtualized by the PIP and offered to the VNP and VNO. Further, an ASP using SVC video streaming requests the VNO to design the network and deliver

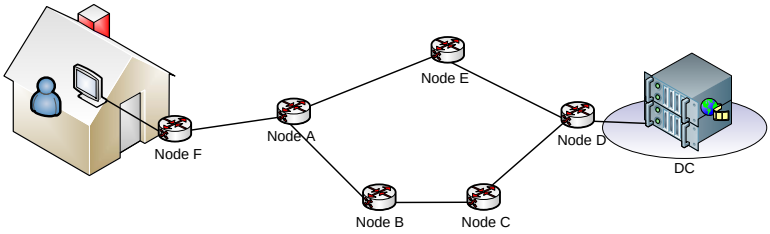


Figure 4.9: *Physical network*

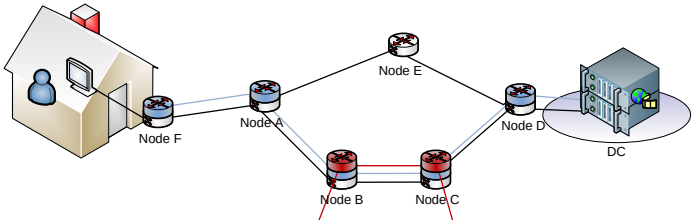


Figure 4.10: *Initial setup of the VN and setup of a second VN.*

the service. Figure 4.9 depicts the underlying physical network which consists of a ring formed by the nodes A to E as well as node F, the peering point to the users and the connection to the data center via node D. After receiving the request to set up the service, the PIP's network management determines the realization of the virtual network and defines the path A-B-C-D as the network for the service. The information is signaled using RSVP-TE and establishes a Label Switched Path (LSP) with the requested bandwidth, as illustrated in Figure 4.10.

Customers start to join the service and the utilization of the video streaming network increases. At the same time the PIP assigns the remaining resources of the link between nodes B and C to another VN. This is indicated by the additional virtual routers and the link between them in Figure 4.10. In our scenario, more

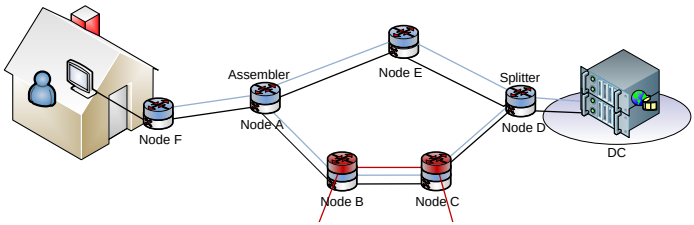


Figure 4.11: *Enhancement layers successfully transferred to second path.*

and more customers join the video service inducing an increasing bandwidth consumption. Thus, the VNO's network management system reacts by requesting additional bandwidth. As there is no remaining capacity on the link B-C, the bandwidth of the link cannot be increased. Thus, the PIP sets up a second virtual link via node E, maybe even with different QoS guarantees. Therefore, the VNO configures his components, i.e., the multipath streaming framework, to transmit the base layer for all users on the previous link and the less important enhancement layer via the second link. As a result, there is no quality degradation of the video streams that are delivered to the customers. The resulting changes in traffic on the two paths can be seen in the lower part of Figure 4.11. Thus, the demonstration showed the interaction between network and service providers using multipath transmission to increase the available network resources. Accordingly, the video stream was distributed via the involved paths with respect to the dependencies of SVC. Although it was not shown in the demonstration, a graceful adaptation of the video streaming service in case of insufficient network resources by dropping enhancement layers is possible, if necessary.

4.6 Lessons Learned

This chapter introduced the QoE-PDH and investigated whether the hysteresis can be applied to video streaming using H.264/SVC. The hysteresis comprises

the provisioning part and the delivery part. The provisioning part describes the impact of the provisioned resources such as link capacity, and the delivery part describes the impact of delivery failures due to congestion, such as packet loss. We showed that video streaming fulfills the delivery part of the hysteresis in case of packet losses. The provisioning part is also fulfilled if resolution or image quality are reduced. However, it became apparent that frame rate reduction better matches the delivery part of the hysteresis, although it performs much better in terms of user perception than a degradation due to packet loss. Thus, a wrong provisioning of a service may cause the wrong part of the hysteresis. Again, frame rate reduction still outperforms distortion caused by packet loss with respect to user-perceived quality.

Next, we investigated the potential of network and service interaction and derived guidelines how to react in case of insufficient network resources. If the network adapts to the congestion state by providing additional resources, more customer can be supported. However, if no more additional resources can be provided, congestion and packet loss occur resulting in a strong impairment of the service quality. If the service adspts to the congestion in the network by reducing its requirements, i.e. implementing the provisioning part of the QoE-PDH, the service quality is reduced gracefully. Thus, the quality of a customer is reduced compared to the maximum available service quality, but still much higher than in case of delivery failures. Further, we have seen that the best service quality can be provided if both, network and service, react together.

Current discussion for future networks wants to solve this information asymmetry by providing unified interfaces between applications and networks. Thus we summarized candidate approaches like the G-Lab API and showed possible application scenarios. To show the capabilities of multipath SVC streaming, we implemented a software prototype and integrated it into the COMCON demonstration during the Euroview 2011 event [117]. Accordingly, we showed that the interaction between applications and networks is feasible and enhances their capabilities.

5 Conclusion

The future Internet should enable a better interaction between applications and networks. This interaction may be beneficial for both sides since network resources can be utilized more efficiently and applications can adapt their demands to the available resources. Depending on the interaction and the resulting reactions, the user perceived quality can be improved.

In this monograph, we studied the interaction between applications and networks for the prominent use case of a scalable video streaming application and multipath data transport. We showed that the perceived video quality can be increased significantly if both parties cooperate.

Video streaming has become the dominating application in the current Internet, responsible for the highest amount of traffic. It is consumed on a wide range of devices from smart TVs to mobile smartphones by users that are connected to the Internet via heterogeneous access technologies like cable, DSL, FTTH, UMTS, or WiFi. Thus, it becomes necessary to use video streaming techniques that enable a flexible adaptation of the video stream to meet user, application, and network requirements. These demands are supported by the scalable extension of the video codec H.264/AVC, which allows an adaptation of the video quality with respect to resolution, image quality, and frame rate.

To operate such a scalable video streaming service, a proper QoE management is necessary, which takes into account these adaptation mechanisms as well as the impact of network distortion and congestion. To achieve this, we introduce methods based on objective QoE metrics that allow an investigation of video distortions caused by packet loss or a reduction of the video quality. We perform experiments in a local testbed and show that packet loss has a severe impact on the

user perceived quality and thus should be avoided if possible. Instead, intelligent mechanisms should lower the data rate of the video stream, which prevents packet loss and thus results in a better perceived quality. We investigate different options to reduce the data rate and evaluate their impact on the video quality. Based on these results, an appropriate QoE control for scalable video streaming is derived.

Not only the applications are more and more flexible and able to adapt their demands, but also the networks. Technologies like network virtualization and multipath transport allow a more flexible and efficient utilization of the network resources. Virtual network operators can add physical resources to their virtual networks to cope with high demands or remove them to reduce operational expenditures during times of low workloads. Multipath techniques allow the pooling of transport resources from different network operators or network technologies and thus increase the resilience and capacity of the data transport. However, these techniques add additional complexity, which has to be understood, so that multipath mechanisms can be operated efficiently. Multipath transmissions inevitably introduce out-of-order packets at the destination, due to different, varying delays on the involved paths. This reordering can be compensated by an additional re-sequencing buffer, which might entail additional buffering delays.

We propose theoretical models which allow the computation of the resequencing buffer occupancy based on the characteristics of the involved paths. These models are validated with measurements in experimental facilities and their limitations are explored. The developed models allow an a priori calculation of the resequencing-buffer occupancy and thus enable an appropriate dimensioning of the buffer. Additionally, the developed simulation framework allows also an investigation of scheduling approaches and of the perceived end-to-end delay. Based on these results, we propose a pre-buffering mechanism that reduces the buffer occupancy significantly while the perceived packet delay is not increased. This mechanism illustrates how network resources can be optimized without deteriorating the application performance.

An application experiencing packet loss might interpret this as congestion and try to reduce the network load by limiting its demands. But this does not necessar-

ily reduce the packet loss rate, which might be caused by other circumstances and thus does not increase the QoE. It might be better to react in a different way to the occurring packet loss, for instance by additional protection mechanisms like forward error correction. If information about application capabilities and network state are exchanged, the involved parties can adapt their behavior specifically to the current situation. Thus, a closer interaction of applications and networks is required in the future Internet since it is not enough to either adapt the application or the network without a communication in between.

We present the QoE-Provisioning-Delivery Hysteresis, which describes fundamental relationships between QoE and Quality of Service (QoS) parameters. These relationships allow to evaluate the benefit of application-network cooperation for the user perceived quality. The hysteresis comprises the provisioning part, which describes the impact of the provisioned resources such as link capacity, and the delivery part, which depicts the impact of delivery failures for instance due to congestion or packet loss. Our findings show that the Provisioning-Delivery Hysteresis can be applied to video streaming. Accordingly, we illustrate the potential of interaction between a video streaming application and the network by analysis and implementation.

In the course of this monograph, we evaluated the cooperation of future resource and service management for the example of multipath video streaming using H.264/SVC. In particular, we described how resource management within the network can be implemented for the use-case of concurrent multipath transport and presented a proper QoE management for scalable video streaming. We showed the potential of close interaction between network and application in terms of user perceived quality. The investigated mechanisms, which enable a better network and application flexibility, constitute important building blocks for the future Internet.

Beyond that other questions dealing with technical, architectural and performance issues have to be answered. The technical ones comprise the impact of other resource management mechanisms on the application performance, and how other kinds of applications can adapt their demands to network capabilities.

Second, architectural questions as decision unit placement and the corresponding flow of information have to be investigated. Third, performance questions have to be answered. Information exchange required for intelligent future network control implies additional overhead and therewith costs in terms of bandwidth and processing. Therefore, it is important to determine the amount of information that is necessary to improve user perceived quality without degrading network performance by excessive signaling. The solution of these problems will result in implementation instructions, the specification of new communication interfaces, and operation guidelines, which will be of high importance for network and application interaction in the future Internet. This work provides a first step towards the understanding of the potential and the operation of this new network paradigm.

Bibliography and References

— Bibliography of the Author —

— Journals and Book Chapters —

[1] T. Zseby, T. Zinner, K. Tutschku, Y. Shavitt, P. Tran-Gia, C. Schwartz, A. Rafetseder, C. Henke, and C. Schmoll, "Multipath routing slice experiments in federated testbeds," in *Future Internet Assembly (FIA) Book Future Internet: Achievements and Promising Technology*. Springer, May 2011.

[2] T. Zinner, K. Tutschku, A. Nakao, and P. Tran-Gia, "Performance evaluation of packet re-ordering on concurrent multipath transmissions for transport virtualization," *JCNDS Special Issue on: Network Virtualization - Concepts and Performance Aspects DOI: 10.1504/IJCNDS.2011.039538*, vol. 6, May 2011.

— Conference Papers —

[3] M. Duelli, S. Meier, D. Wagner, T. Zinner, M. Schmid, M. Hoffmann, and W. Kies, "Experimental demonstration of network virtualization and resource flexibility in the COMCON project," in *8th International ICST Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities*, Thessaloniki, June 2012.

[4] J. Rückert, O. Abboud, T. Zinner, D. Hausheer, and R. Steinmetz, "Quality adaptation in P2P video streaming based on objective QoE assessment," in *IFIP/TC6 NETWORKING 2012*, Prague, Czech Republic, May 2012.

[5] T. Zinner, F. Liers, T. Hoßfeld, D. Rauscher, B. Reuther, D. Guenther, T. Volkert, and M. Fiedler, "Prospects for realizing user-centric network orchestration: FEC-protected SVC streaming," in *7. GI/ITG KuVS Fachgespraech 'Future Internet'*, Muenchen, January 2012.

[6] 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 - Modeling, Evaluation, and Directions (MQoE 2011)*, Dana Point, CA, USA, December 2011.

[7] T. Hoßfeld, M. Fiedler, and T. Zinner, "The QoE provisioning-delivery-hysteresis and its importance for service provisioning in the future internet," in *NGI 2011, 7thEURO-NGI Conference on Next Generation Internet Networks*, Kaiserslautern, Germany, June 2011.

[8] R. Stankiewicz, P. Cholda, J. Domzal, R. Wojcik, T. Hoßfeld, T. Zinner, S. Oechsner, and F. Lehrieder, "Influence of traffic management solutions on quality of experience for prevailing overlay applications," in *NGI 2011, 7thEURO-NGI Conference on Next Generation Internet Networks*, Kaiserslautern, Germany, June 2011.

[9] T. Zinner, D. Klein, K. Tutschku, T. Zseby, P. Tran-Gia, and Y. Shavitt, "Performance of concurrent multipath transmissions - measurements and model validation," in *NGI 2011, 7thEURO-NGI Conference on Next Generation Internet Networks*, Kaiserslautern, Germany, June 2011.

[10] T. Zinner, K. Tutschku, and T. Zseby, "MultiNext - measuring concurrent multipath transmissions in an experimental facility," in *NGI 2011, 7thEURO-NGI Conference on Next Generation Internet Networks*, Kaiserslautern, Germany, June 2011.

[11] O. Abboud, T. Zinner, K. Pussep, and R. Steinmetz, "On the impact of quality adaptation in SVC-based p2p video-on-demand systems," in *ACM Multimedia Systems*, San Jose, CA, USA, February 2011.

[12] T. Zinner, O. Abboud, O. Hohlfeld, T. Hoßfeld, and P. Tran-Gia, "Towards QoE management for scalable video streaming," in *21st ITC Specialist Seminar on Multimedia Applications - Traffic, Performance and QoE*, Miyazaki, Japan, March 2010.

[13] S. Oechsner, T. Zinner, J. Prokopetz, and T. Hoßfeld, "Supporting scalable video codecs in a P2P video-on-demand streaming system," in *21th ITC Specialist Seminar on Multimedia Applications - Traffic, Performance and QoE*, Miyazaki, Japan, March 2010.

[14] T. Zinner, T. Hoßfeld, T. N. Minhas, and M. Fiedler, "Controlled vs. uncontrolled degradations of QoE - the provisioning-delivery hysteresis in case of video," in *QOEMCS 2010 - New Dimensions in the Assessment and Support of Quality of Experience (QoE) for Multimedia Applications*, Tampere, Finland, Jun. 2010.

[15] T. Zinner, O. Hohlfeld, O. Abboud, and T. Hoßfeld, "Impact of frame rate and resolution on objective QoE metrics," in *QOMEX 2010 - International Workshop on Quality of Multimedia Experience, 2010*, Trondheim, Norway, Jun. 2010.

[16] O. Abboud, T. Zinner, E. Lidanski, K. Pussep, and R. Steinmetz, "Streamsocial: A P2P streaming system with social incentives," in *IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks*, Montreal, Canada, May 2010.

[17] T. Zinner, K. Tutschku, A. Nakao, and P. Tran-Gia, "Using concurrent multipath transmission for transport virtualization: Analyzing path selection," in *22nd International Teletraffic Congress (ITC)*, Amsterdam, Netherlands, September 2010.

[18] K. Tutschku, T. Zinner, A. Nakao, and P. Tran-Gia, "Network virtualization: Implementation steps towards the future internet," in *Electronic Communications of the EASST, Volume 17: Kommunikation in Verteilten Systemen 2009*, Kassel, Germany, March 2009.

[19] ——, "Re-sequencing buffer occupancy of a concurrent multipath transmission mechanism for transport system virtualization," in *KIVS 2009, 16. Fachtagung Kommunikation in Verteilten Systemen*, Kassel, Germany, March 2009.

[20] T. Zinner, K. Tutschku, A. Nakao, and P. Tran-Gia, "Performance evaluation of packet re-ordering on concurrent multipath transmissions for transport virtualization," in *20th ITC Specialist Seminar on Network Virtualization - Concept and Performance Aspects*, Hoi An, Viet nam, May 2009.

[21] T. Zinner, D. Staehle, P. Tran-Gia, A. Maeder, and K. Tutschku, "Performance evaluation of the information sink in a multi-probe statistical anomaly detection system," in *ATNAC 2008, Australasian Telecommunication Networks and Applications Conference*, Adelaide, Australia, December 2008.

[22] T. Zinner, T. Hoßfeld, S. Oechsner, and P. Tran-Gia, "On the trade-off between efficiency and congestion in location-aware overlay networks - example of a vertical handover support system," in *P2P 2008, 8th International Conference on Peer-to-Peer Computing*, Aachen, Germany, September 2008.

— Software Demonstrations —

[23] T. Zinner, F. Lehrieder, V. Burger, B. Kleine, D. Rauscher, V. Singeorzan, S. Oechsner, and T. Hossfeld, "Utilizing multi-layer codecs for QoE control in P2P video streaming," in *Euroview 2009, 9th Wuerzburg Workshop on IP:Joint ITG and Euro-NF Workshop "Visions of Future Generation Networks"*, Wuerzburg, July 2009.

[24] O. Abboud, K. Pussep, R. Steinmetz, T. Zinner, S. Oechsner, and T. Hossfeld, "A QoE-aware P2P streaming system based on scalable video coding," in *Euroview 2010, 10th Wuerzburg Workshop on IP:Joint ITG and Euro-NF Workshop "Visions of Future Generation Networks" & G-Lab Status Meeting*, Wuerzburg, August 2010.

[25] T. Zinner, "SIG multipath/ASP5: Multipath video streaming demonstration," in *G-Lab Project Meeting*, Neuburg am Inn, Germany, February 2011.

[26] D. Schlosser, M. Duelli, T. Zinner, S. Meier, D. Wagner, M. Barisch, M. Hoffmann, W. Kellerer, and M. Schmid, "Service component mobility enabled by network virtualization," in *Euroview 2011, 11th Wuerzburg Workshop on IP:Joint ITG and Euro-NF Workshop "Visions of Future Generation Networks" & G-Lab Status Meeting*, Wuerzburg, Germany, August 2011.

— General References —

[27] P. Tran-Gia, "G-lab: A future generation internet research platform," 2008.

[28] GENI Consortium, "GENI - global environment for network innovations," 2006, information available at http://www.geni.net/.

[29] A. Feldmann, M. Kind, O. Maennel, G. Schaffrath, and C. Werle, "Network virtualization - an enabler for overcoming ossification," *ERCIM News*, no. 77, pp. 21–22, April 2009, invited article; Special theme: Future Internet Technology. [Online]. Available: http://ercim-news.ercim.org/content/view/574/763/

[30] J.-P. L. Daniel Kofman, "First update of the Euro-NF vision regarding the network of the future - D.SEA.10.1.1," Available at http://www.euronf.org, 2009.

[31] "Multilink architecture for multiplay services (MARCH)," 2011, information available at http://projects.celtic-initiative.org/MARCH/march/.

[32] Y. Ja-young, "Sk telecom develops network integration solution," *The Korea Times*, 2012, http://www.koreatimes.co.kr/www/news/nation/2012/01/182_102253.html.

[33] T. Hoßfeld, D. Hock, K. Tutschku, and M. Fiedler, "Testing the IQX hypothesis for exponential interdependency between qos and qoe for voice codecs ilbc and g.711," in *18th ITC Specialist Seminar on Quality of Experience*, Karlskrona, Sweden, May 2008.

[34] Y. Nishida and P. Eardley, "Charter of the multipath tcp (MPTCP) working group."

[35] M. Becke, T. Dreibholz, J. Iyengar, P. Natarajan, and M. Tuexen, "Draft load sharing for the stream control transmission protocol (SCTP)," January 2011, information available at http://tools.ietf.org/html/draft-tuexen-tsvwg-sctp-multipath-01.

[36] 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.

[37] X. Hesselbach, R. Fabregat, and C. Kolias, "The impact over the packets sequence at the output interface in load balancing strategies," in *Icton 2006, 8th International Conference on Transparent Optical Networks*, vol. 3. Nottingham, United Kingdom: IEEE, June 2006, pp. 263–266.

[38] J. R. Lane and A. Nakao, "Best-effort network layer packet reordering in support of multipath overlay packet dispersion," in *Globecom 2008, IEEE Global Communications Conference*, New Orleans, LA, USA, November/December 2008.

[39] Y. Xia and D. Tse, "Analysis on packet resequencing for reliable network protocols," in *IEEE INFOCOM 2003, 22nd Annual IEEE International Conference on Computer Communications*, San Francisco, USA, 2003.

[40] G. Harrus and B. Plateau, "Queueing analysis of a reordering issue," *IEEE Transactions on Software Engineering*, vol. 8, pp. 113–123, March 1982. [Online]. Available: http://dx.doi.org/10.1109/TSE.1982.234954

[41] T.-S. Yum and T.-Y. Ngai, "Resequencing of messages in communication networks," *IEEE Transactions on Communications*, vol. 34, pp. 143–149, February 1986. [Online]. Available: http://dx.doi.org/10.1109/TCOM.1986.1096505

[42] S. Gunreben, "Framework for systematic evaluation of protocol performance with respect to out-of-sequence packet arrivals," Ph.D. dissertation, University of Stuttgart, December 2010.

[43] T. Santos, C. Henke, C. Schmoll, and T. Zseby, "Multi-hop packet tracking for experimental facilities," in *ACM Sigcomm 2010, 2010 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications Conference*.

[44] M. Zhang, J. Lai, A. Krishnamurthy, L. Peterson, and R. Wang, "A transport layer approach for improving end-to-end performance and robustness using redundant paths," in *Proceedings of the annual conference on USENIX Annual Technical Conference*, ser. ATEC '04. Berkeley, CA, USA: USENIX Association, 2004, pp. 8–8. [Online]. Available: http://dl.acm.org/citation.cfm?id=1247415.1247423

[45] L. Magalhaes and R. H. Kravets, "Transport level mechanisms for bandwidth aggregation on mobile hosts," in *ICNP 2001, 9th International Conference on Network Protocols*, November 2001, pp. 165–171.

[46] K. Rojviboonchai, T. Osuga, and H. Aida, "R-m/tcp: Protocol for reliable multi-path transport over the internet," *AINA 2005, 19th International Conference on Advanced Information Networking and Applications*, vol. 1, pp. 801–806, March 2005.

[47] H.-Y. Hsieh and R. Sivakumar, "pTCP: An end-to-end transport layer protocol for striped connections," in *ICNP 2002, 10th IEEE International Conference on Network Protocols*, Paris, France, November 2002.

[48] D. Wischik, M. Handley, and C. Raiciu, "Control of multipath tcp and optimization of multipath routing in the internet," in *NET-COOP 2009, 3rd Euro-NF Conference on Network Control and Optimization*, ser. NET-COOP '09. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 204–218. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-10406-0_14

[49] A. Ford, C. Raiciu, and M. Handley, "Draft TCP extensions for multipath operation with multiple addresses," January 2011, information available at http://tools.ietf.org/html/draft-ietf-mptcp-multiaddressed-02.

[50] M. Scharf, "Draft multi-connection tcp (MCTCP) transport," January 2011, information available at http://tools.ietf.org/html/draft-scharf-mptcp-mctcp-01.

[51] C. Raiciu, M. Handley, and D. Wischik, "Draft coupled congestion control for multipath transport protocols," January 2011, information available at http://tools.ietf.org/html/draft-ietf-mptcp-congestion-01.

[52] S. Barre, C. Raiciu, O. Bonaventure, and M. Handley, "Experimenting with multipath TCP," in *ACM Sigcomm 2010, 2010 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications Conference*, September 2010, http://conferences.sigcomm.org/sigcomm/2010/papers/sigcomm/p443.pdf.

[53] S. Barre, C. Paasch, and O. Bonaventure, "Multipath tcp: From theory to practice," in *IFIP/TC6 Networking*, Valencia, Spain, May 2011.

[54] J. Iyengar, P. Amer, and R. Stewart, "Concurrent multipath transfer using SCTP multihoming," in *SPECTS 2004, 2004 International Symposium on Performance Evaluation of Computer and Telecommunication Systems*, San Jose, CA, USA, May 2004.

[55] ——, "Concurrent multipath transfer using SCTP multihoming over independent end-to-end paths," *IEEE/ACM Transactions on Networking (TON)*, vol. 14, no. 5, pp. 951–964, 2006.

[56] H. Adhari, T. Dreibholz, M. Becke, and E. Rathgeb, "Evaluation of concurrent multipath transfer over dissimilar paths," in *PAMS 2011, 1st International Workshop on Protocols and Applications with Multi-Homing Support*, Singapore, 2011.

[57] T. Dreibholz, M. Becke, E. Rathgeb, and M. Tüxen, "On the use of concurrent multipath transfer over asymmetric paths," in *GLOBECOM 2010, IEEE Global Communications Conference*, Miami, Florida/U.S.A., December 2010, ISBN 978-1-4244-5637-6. [Online]. Available: http://www.tdr.wiwi.uni-due.de/fileadmin/fileupload/I-TDR/SCTP/Paper/Globecom2010.pdf

[58] R. Stewart, M. Ramalho, Q. Xie, M. Tuexen, and P. Conrad, "IETF RFC 3578stream control transmission protocol (SCTP) partial reliability extension," May 2004, information available at http://www.ietf.org/rfc/rfc3758.txt.

[59] X. Hesselbach, R. Fabregat, B. Baran, Y. Donoso, F. Solano, and M. Huerta, "Hashing based traffic partitioning in a multicast-multipath MPLS network model," in *LANC 2005, 3rd international IFIP/ACM Latin American Conference on Networking*, ser. LANC '05. New York, NY, USA: ACM, 2005, pp. 65–71. [Online]. Available: http://doi.acm.org/10.1145/1168117.1168127

[60] A. Nakao, L. Peterson, and A. Bavier, "A routing underlay for overlay networks," in *ACM Sigcomm 2003, 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications Conference*, Karlsruhe, Germany, August 2003.

[61] K. P. Gummadi and H. V. Madhyastha, "Improving the reliability of internet paths with one-hop source routing," in *OSDI 2004, 6th Symposium on Operating Systems Design & Implementation*, San Francisco, CA, USA, December 2004.

[62] GENI Group, "GENI design principles," *IEEE Computer*, Septemper 2006.

[63] J. R. Lane and A. Nakao, "SORA: A shared overlay routing architecture," in *ROADS 2007, 2nd International Workshop on Real Overlays And Distributed Systems*, Warsaw, Poland, July 2007.

[64] S. Khor and A. Nakao, "AI-RON-E: Prophecy of one-hop source routers," in *GLOBECOM 2008, IEEE Global Communications Conference*, New Orleans, LA, USA, November/December 2008.

[65] M. Liljenstam, J. Liu, and D. Nichol, "Development of an internet backbone topology for large-scale network simulation," in *Proceedings of the 2003 Winter Simulation Conference*, New Orleans, December 2003.

[66] S. Banerjee, T. Griffin, and M. Pias, "The interdomain connectivity of planetlab nodes," in *PAM2004, 5th Passive and Active Measurement Workshop*, Antibes Juan-les-Pins, France, April 2004.

[67] Y. Nebat and M. Sidi., "Resequencing considerations in parallel downloads," in *IEEE INFOCOM 2002, 21st Annual Joint Conference of the IEEE Computer and Communications Societies*, 2002.

[68] Y. Nebat and M. Sidi, "Parallel downloads for streaming applications – a resequencing analysis," *Performance Evaluation*, vol. 63, no. 1, pp. 15–35, January 2006.

[69] "Ilog cplex," http://www.ilog.com/products/cplex/. [Online]. Available: http://www.ilog.com/products/cplex/

[70] B. Chun, D. Culler, T. Roscoe, A. Bavier, L. Peterson, M. Wawrzoniak, and M. Bowman, "Planetlab: an overlay testbed for broad-coverage services," *ACM SIGCOMM Computer Communication Review*, vol. 33, pp. 3–12, July 2003. [Online]. Available: http://doi.acm.org/10.1145/956993.956995

[71] A. Bavier, N. Feamster, M. Huang, L. Peterson, and J. Rexford, "In VINI veritas: Realistic and controlled network experimentation," in *ACM Sigcomm 2006, 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications Conference*, Pisa, Italy, September 2006.

[72] D. Morato, E. Magana, M. Izal, J. Aracil, F. Naranjo, F. Astiz, U. Alonso, I. Csabai, P. Haga, G. Simon, J. Steger, and G. Vattay, "The european traffic observatory measurement infraestructure (ETOMIC): A testbed for universal active and passive measurements," in *Tridentcom 2005, 1st International Conference on Testbeds and Research Infrastructures for the DEvelopment of NeTworks and COMmunities*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 283–289. [Online]. Available: http://portal.acm.org/citation.cfm?id=1042447.1043725

[73] *H.264: Advanced video coding for generic audiovisual services (03/2010)*, Telecommunication Standardization Sector Std.

[74] I. D. C. D.197, "Definition of quality of experience," 2004.

[75] "Recommendation 500-10: Methodology for the subjective assessment of the quality of television pictures," ITU-R Rec. BT.500, 2000.

[76] K. Seshadrinathan, R. Soundararajan, A. C. Bovik, and L. K. Cormack, "Study of subjective and objective quality assessment of video," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1427–1441, Jun. 2010. [Online]. Available: http://dx.doi.org/10.1109/TIP.2010.2042111

[77] S. Winkler, "Video quality and beyond," in *Eusipco 2007, 15th European Signal Processing Conference*, Poznan, Poland, September 2007.

[78] B. Girod, "What's wrong with mean-squared error?" *Digital images and human vision*, pp. 207–220, 1993.

[79] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, pp. 600–612, 2004.

[80] I. R. BT.1683, "Objective perceptual video quality measurement techniques for standard definition digital broadcast television in the presence of a full reference," 2004.

[81] M. H. Pinson and S. Wolf, "A new standardized method for objectively measuring video quality," *IEEE Transactions on Broadcasting*, vol. 50, no. 3, pp. 312–322, September 2004.

[82] O. Rose, "Traffic modeling of variable bit rate MPEG video and its impacts on ATM networks," Ph.D. dissertation, University of Wuerzburg, March 1997.

[83] M.-N. Garcia and A. Raake, "Parametric packet-layer video quality model for IPTV," in *ISSPA 2010, 10th International Conference on Information Science, Signal Processing and their Applications*, May 2010, pp. 349–352.

[84] K. D. Singh, A. Ksentini, and B. Marienval, "Quality of experience measurement tool for SVC video coding," in *ICC 2011, IEEE International Conference on Communications*, Kyoto, Japan, June 2011.

[85] K. Yamagishi and T. Hayashi, "Parametric packet-layer model for monitoring video quality of IPTV services," in *ICC 2008, IEEE International Conference on Communications*, 2008, pp. 110–114.

[86] S. Buchinger and H. Hlavacs, "Subjective quality of mobile mpeg-4 videos with different frame rates," *Journal of Mobile Multimedia (JMM)*, vol. 1, no. 4, pp. 327–341, 2006.

[87] J. McCarthy, M. Sasse, and D. Miras, "Sharp or smooth?: Comparing the effects of quantization vs. frame rate for streamed video," in *CHI 2004,*

SIGCHI conference on Human Factors in Computing Systems. New York, NY, USA: ACM, 2004, pp. 535–542.

[88] J. S. Lee, F. De Simone, N. Ramzan, Z. Zhao, E. Kurutepe, T. Sikora, J. Ostermann, E. Izquierdo, and T. Ebrahimi, "Subjective evaluation of scalable video coding for content distribution," in *ACM International Conference on Multimedia*, ser. MM '10. New York, NY, USA: ACM, 2010, pp. 65–72. [Online]. Available: http://dx.doi.org/10.1145/1873951.1873981

[89] I. T. Union, "ITU-T Recommendation P.910: Subjective video quality assessment methods for multimedia applications," 2004.

[90] "Xiph.org test media," http://media.xiph.org/video/derf/.

[91] J. Klaue, B. Rathke, and A. Wolisz, "Evalvid - a framework for video transmission and quality evaluation," in *13th International Conference on Modelling Techniques and Tools for Computer Performance Evaluation*, Urbana, IL, USA, Sep. 2003.

[92] "Autox264," http://www.videohelp.com/tools/AutoX264.

[93] "x264 - h264/avc encoder," http://www.videolan.org/developers/x264.html.

[94] "tcpdump/libcap," http://www.tcpdump.org/.

[95] The Linux Foundation, "Compression project," http://www.compression.ru/video/index.htm.

[96] Graphics and M. S. U. Media Lab, CMC department, "Msu video quality measurement tool."

[97] Vcodex, "4x4 transform and quantization in h.264/avc," *White Paper*, November 2010.

[98] B.-Y. Chen and S.-H. Yang, "Using h.264 coded block patterns for fast inter-mode selection," *ICME 2008, IEEE International Conference on Multimedia and Expo*, 2008.

[99] S.-H. Yang and K.-H. Wang, "An improved h.264 fast inter-mode decision algorithm," *Internet and Multimedia Systems and Applications*, 2009.

[100] Q. Huynh-Thu and M. Ghanbari, "Scope of validity of psnr in image/video quality assessment," *Electronics Letters*, vol. 44, no. 13, 2008.

[101] M. Fiedler and T. Hoßfeld, "Quality of experience-related differential equations and provisioning-delivery hysteresis," in *21st ITC Specialist Seminar on Multimedia Applications - Traffic, Performance and QoE*, Miyazaki, Japan, Mar. 2010.

[102] J. Shaikh, M. Fiedler, and D. Collange, "Quality of experience from user and network perspectives," *Annales des Télécommunications*, vol. 65, no. 1-2, pp. 47–57, 2010.

[103] P. Reichl, S. Egger, R. Schatz, and A. D'Alconzo, "The logarithmic nature of qoe and the role of the weber-fechner law in qoe assessment," in *ICC 2010, International Conference on Communications*, Cape Town, South Africa, May 2010.

[104] U. Engelke, M. Kusuma, H.-J. Zepernick, and M. Caldera, "Reduced-reference metric design for objective perceptual quality assessment in wireless imaging," *Signal Processing-Image Communication*, pp. 525–547, 2009.

[105] T. S. Consortium, "Requirements and application classes and traffic characteristic," in *Deliverable D1.1: Requirements and Application Classes and Traffic Characteristic*, v4.0 ed., T. Hoßfeld and D. Staehle, Eds. European Seventh Framework Project SmoothIT FP7-2008-ICT-216259-STREP, 2009, available at http://www.smoothit.org.

[106] D. Bonfiglio, M. Mellia, M. Meo, and D. Rossi, "Detailed analysis of skype traffic," *IEEE Transactions on Multimedia*, vol. 11, no. 1, pp. 117–127, 2009.

[107] T. Hoßfeld and A. Binzenhöfer, "Analysis of skype voIP traffic in UMTS: End-to-end qos and qoe measurements," *Computer Networks, online http://dx.doi.org/10.1016/j.comnet.2007.10.008*, vol. 52, no. 3, February 2008.

[108] J. Peterson, V. Gurbani, and E. Marocci, "Charter of the application-layer traffic optimization (ALTO) working group," March 2009.

[109] V. Aggarwal, O. Akonjang, and A. Feldmann, "Improving user and ISP experience through ISP-aided P2P locality," in *GI 2008, 11th IEEE Global Internet Symposium*. Washington, DC, USA: IEEE Computer Society, April 2008.

[110] D. Martin, H. Wippel, and H. Backhaus, "A future-proof application-to-network interface," in *NoF 2011, 2nd International Conference on the Network of the Future*. Paris, France: IEEE, November 2011.

[111] F. Liers, T. Volkert, D. Martin, H. Backhaus, H. Wippel, E. M. Veith, A. A. Siddiqui, and R. Khondoker, "GAPI: A G-Lab application-to-network interface," in *EuroView2011, 11th Wuerzburg Workshop on IP:Joint ITG and Euro-NF Workshop "Visions of Future Generation Networks"*, Wuerzburg Germany, August 2011.

[112] B. Reuther and D. Henrici, "A model for service-oriented communication systems," *Journal of Systems Architecture: the EUROMICRO Journal*, vol. 54, pp. 594–606, June 2008. [Online]. Available: http://dl.acm.org/citation.cfm?id=1374864.1375269

[113] F. Liers, T. Volkert, and A. Mitschele-Thiel, "Forwarding on gates: A clean-slate future internet approach within the g-lab project," in *Euroview*

2009, 9th Wuerzburg Workshop on IP:Joint ITG and Euro-NF Workshop "Visions of Future Generation Networks", Wuerzburg, July 2009.

[114] D. Martin, L. Völker, and M. Zitterbart, "A flexible framework for future internet design, assessment, and operation," *Computer Networks*, vol. 55, pp. 910–918, March 2011. [Online]. Available: http://dx.doi.org/10.1016/j.comnet.2010.12.015

[115] B. Staehle, F. Wamser, M. Hirth, D. Stezenbach, and D. Staehle, "AquareYoum: Application and quality of experience-aware resource management for youtube in wireless mesh networks," *Praxis der Informationsverarbeitung und Kommunikation*, vol. 34, no. 3, pp. 144–148, 2011.

[116] "COMCON - control and management of coexisting networks," 2011, information available at http://www.german-lab.de/phase-2/comcon//.

[117] 11th Würzburg Workshop on IP: Joint ITG and Euro-NF Workshop "Visions of Future Generation Networks", *Conference Web Page*, Aug. 2011, http://www.euroview2011.com/.

ISSN 1432-8801