# Final Report on the Subproject

# "Extrapolation"

**of the Technische Universität München**

**delivered to the**

**LETS QoS Project**

**of the Technische Universität Darmstadt,**

**Chair for Multimedia Communications**

**April $29^{th}$, 2004**

**Eike Jessen**
**Marco Hoffmann**

# Contents

# Chapter 1

# Extent of Work

The work began on November $15^{th}$, 2001, initially funded by the budget of TU München, as a contribution of 6 PM, until May $14^{th}$, 2002. The work ended after a cost-invariant extension of the subcontract on February $14^{th}$, 2004. The work was done by Dipl. Inform. Marco Hoffmann; he was assisted by Prof. Eike Jessen, Dr. Manfred Jobmann and Prof. Lester Lipsky (University of Connecticut). According to the project plan the contribution of TU München was, in person-months (PM):

| | | |
|---|---|---|
| Work Package I (QoS in the Network Layer) | 3 PM | |
| Work Package IV (System Integration) | 1 PM | |
| Work Package V (Extrapolation) | 20 PM | (full package) |

**Table 1.1**: Contribution of TU München

# Chapter 2

# Contribution to Work Package I (QoS in the Network Layer)

## 2.1 Analysis of Existing Work

The three most important concepts of QoS were evaluated in a form which is adequate for modeling (as a simulator as well as an analytical model) and documented, compare Milestone 1 [Schmitt et al. 02] and chapters 4.2, 4.3 and 4.4 of this report.

## 2.2 Observation of Technical Development during the Project

The uncertainties still existing in 2001 as to the exact definition of the formation of classes and of service strategies for Differentiated Services were cleared by the adoption of a priority mode, the Olympic DiffServ approach which had been designed similar to the approach in the DFN-Project Quasar (see [Burgstahler et al. 01]), and the weighted round robin mode called Default DiffServ approach, based on the IETF design. The further development of the DFN-Gigabit Research Network from 2001 to 2003 was observed and became the foundation of the extrapolation technique.

# Chapter 3

# Contribution to Work Package IV (System Integration)

This package comprises the integration of various methods and modules into the testbed operated by the TU Darmstadt. TU München made supplementary contributions:

Evaluation and implementation of the DiffServ variants, collaboration in defining load profiles, in particular for Voice over IP and Video Conferencing, and in system integration in Darmstadt.

# Chapter 4

# Work Package V (Extrapolation)

## 4.1 Requirement Analysis

References: [SCHMITT 01]

The forecast of the quality of service (throughput, delay, jitter, loss) for a given load, a given network and a given QoS-mechanism is rather difficult. The amount of work necessary and the uncertainty of such forecasts is high. Forecasts like these, however, are necessary for answering questions of the following kind:

Can a given load on a given network under a given QoS-mechanism be transported with sufficient quality? How far may one increase a load (and which type) on a given network, without violating the required service quality? Which QoS-mechanism is adequate? By which capacity of the network and which QoS-mechanism can one achieve the required quality of service for a given load?

The particular feature of the extrapolation approach of the LETS QoS-Project is that forecasts of this kind make use of a priori knowledge, which has been gained from the existing network, from a testbed or simulator and from existing or related load under one or several QoS-mechanisms. The extrapolation procedure begins at a start situation (network, load, QoS-mechanism, quality of service), out of which a conclusion for a target situation is drawn, by means of an extrapolation procedure. The target situation has similarities with the start situation. Within the project, the target situation is preferably one of the Gigabit-Wissenschaftsnetz. The extrapolation method may rely on a simulator or on analytical methods; as usual, it is typical that the analytical methods deliver results which are more easily accessible, but applicable only within a limited scope, if one is not satisfied by coarse results. The extrapolation methods have to be validated and - for the special extrapolation problem - calibrated.

Generalizing, one may differentiate eight extrapolation cases, of which the first one is trivial, as the start situation and the target situation are identical:

| Case | Different Network | Different Load | Different QoS-Mechanism |
|---|---|---|---|
| 1 | No | No | No |
| 2 | No | No | Yes |
| 3 | No | Yes | No |
| 4 | No | Yes | Yes |
| 5 | Yes | No | No |
| 6 | Yes | No | Yes |
| 7 | Yes | Yes | No |
| 8 | Yes | Yes | Yes |

**Table 4.1**: Extrapolation Cases

Clearly, the cases two, three, five are simpler than four, six, seven, eight; eight is the most difficult case.
The start situation may be generated within the testbed and be evaluated there, as well as in the Gigabit-Wissenschaftsnetz; neglecting the proper purpose of the project, any other network, the traffic of which is measurable, may render the start position. The testbed itself may be used as the extrapolation method; generally, however, a simulation model or an analytical model will be used. The models have to be calibrated within the start situation to fit the observed conditions. The consistency of the start situation in the testbed and the simulator is given by the scenario generator [Heckmann et al. 03-1] which defines topology and load for both, the testbed and the simulator. The models have to be validated within a parameter space which at least includes the start situation and the target situation. For the purpose of performance forecasts in the G-WiN, the simulator as well as the analytical method must be able to model at least large parts of it. For the testbed, this, of course, can only be true to a very limited extent. All methods, however, should be able to model the relevant load and the chosen QoS-mechanism; here the analytical methods are in trouble. For the simulator and the testbed a particular problem is the abstraction of the forecast out of the necessarily elementary event sequences, to gain technically manageable and relevant data.

## 4.2 Overprovisioning (OP)

References: [Allen 90, Whitt 88]

Based on its routing scheme, the network transports all load packet-wise by a first come - first served (FCFS) strategy without any priorities. Generally, this scheme is called "Best Effort" (BE). For low utilization, this regime leads to an acceptable service quality. As low utilization can be provided by excess bandwidth and router throughput capacity, overprovisioning is considered as a QoS strategy. Many examples show that in a network where the 5 min samples do not exceed an utilization of 20%, the QoS is very good, also for realtime voice and video transmission.

Overprovisioning (Best Effort) is included in the simulator as the standard FCFS strategy.

For solving simple extrapolation problems, as e.g. changing line bandwidth, router performance, packet size, throughput or load mixture, formulas deliver mean queueing time, leading to end-to-end delay. With the assumption of a Poisson arrival process (M/G/1), the Pollaczek-Khinchine formula is to be used, a good approximation for the G/G/1-case are the Krämer/Whitt formulas (see Formula 4.1 and 4.2). In many cases, the base formulas can be reduced in the proper extrapolation cases, where the mean waiting time in the start situation is known.

$$
\begin{aligned}
E[W] &\approx \frac{\rho \cdot E[B]}{2 \cdot (1-\rho)} \cdot (C_A^2 + C_B^2) \cdot f(\rho, C_A^2, C_B^2) \\
f(\rho, C_A^2, C_B^2) &= \begin{cases} e^{-(1-\rho) \cdot \frac{C_A^2 - 1}{C_A^2 + 4 \cdot C_B^2}}, & C_A^2 \geq 1 \\ e^{-\frac{2 \cdot (1-\rho) \cdot (1 - C_A^2)^2}{3 \cdot \rho \cdot (C_A^2 + C_B^2)}}, & C_A^2 < 1 \end{cases}
\end{aligned}
\tag{4.1}
$$

$$
\begin{aligned}
E[W] &\approx \frac{\rho \cdot E[B]}{2 \cdot (1-\rho)} \cdot (C_A^2 + C_B^2) \cdot f(\rho, C_A^2, C_B^2) \\
f(\rho, C_A^2, C_B^2) &= 1
\end{aligned}
\tag{4.2}
$$

For the more complex cases, where the topology of the target situation is different from that of the start situation, the throughput of the lines and routers has to be computed from the source/sink matrix of the network and the routing scheme, as well as the load parameters (packet size distribution), out of which the mean and the coefficient of variation of the service time can be derived, respecting line and router performance parameters. The constant delays for propagation time and the packet size invariant parts of router processing time can be respected easily.

## 4.3 Coarse Granularity (Class Based) QoS: Differentiated Service (DS)

References: [Burgstahler et al. 01, Burgstahler et al. 02, Burgstahler et al. 03, Burgstahler et al. 04, Burgstahler et al. 05, Burgstahler et al. 06, Blake et al. 98, Davie et al. 02, Heinanen et al. 99]

The Differentiated Service is based on domain-wide service-level-agreements (SLAs) for single flows. Flows of identical SLA form aggregates which are treated in an identical manner in each hop. The SLAs describe the quality of service for a flow of defined throughput. A policer checks the incoming flow, and if necessary, a shaper and / or dropper / or remarker will slow down or prune or downgrade the flow. For the nodes, a PHB (per hop behaviour) is defined, where a node comprises router and offgoing line. As already mentioned, two approaches have been modeled: IETF (Default) and Olympic.

In the IETF approach, there are two per hop behaviours, Expedited Forwarding (EF) or Premium service, which is prioritized and accepts a throughput which is at most as large as a fixed quota of the node throughput capacity, and Assured Forwarding, with three subclasses AF1, AF2 and AF3; A WRR (Weighted Round Robin) scheduler is used. The EF class has a very high weight compared to the AF classes, which have nearly equal weights, to have a kind of prioritization for the QoS sensitive EF class. for AF1 the admitted throughput is always less than the quota of the AF1 class, for AF2 at most two times the quota is admitted, whereas for AF3, which has a quota of its own, there are no admission bounds. In case that the incoming throughput exceeds the quota in AF1 or AF2, the excess traffic is downgraded to a higher drop precedence. In all other cases, excess traffic is lost if the queue flows over. If an admission control is used, traffic that is turned away by the AF 1 admission control is given to the AF 2 admission control and so on. So, EF, which in most cases is the class of interest from the point of view of quality, acts much like a single server system within its guaranteed partial throughput capacity, but with a mean residual service time which is determined by the whole load of the node.

In the Olympic approach of Differentiated Services, the first class, called Premium service, is treated like the IETF Expedited Forwarding, excess traffic, however, is not downgraded to the second level, but lost. The second level, Best Effort (BE), gets service when there is no waiting premium load or the premium quota is exhausted. Being second to the prioritized premium service, the second level is actually worse, working in a resource of fluctuating performance, than the Best Effort in the Overprovisioning Case. Should the two upper loads leave a part of the capacity unused, this may be taken by the third level, the so-called Scavenger Service. So with the exception of the quota for the first level, we have a simple non-preemptive priority regime.

If one accepts the simple M/G/1 model, it follows, that the mean waiting time for the premium / EF class is shorter than in a simple (OP) best effort system, where it has been assumed that the first and second moment of the service time distribution are equal between the EF load and the rest of the load and that the OP system would not be overloaded.

## 4.4 Fine Granularity QoS (integrated Services, IS)

References: [Braden et al. 94, Glasmann et al., Parekh & Galleger, White & Crowcroft, Zhang 95, Shenker et al. 97, Wroclawski 97, Wroclawski 97]

Integrated Services is a reservation regime within a best effort frame. Reservations are built up by RSVP. After successful reservation, the flow has a reserved channel with defined throughput, delay, jitter and loss of its own, so it can be considered as an independent subsystem. The reservation is not terminated explicitly but kept as a soft state which is cleared by a periodical check if it has not been used in the last period. So, we have simple conditions of quality during the reserved channel phase, a resource-consuming build-up phase of maybe complex structure, and a residual clearing period, which also occupies resources. Strictly speaking, during the reserved channel phase, temporarily unused but minimum a fixed capacity will be given to the best effort customers, by the weighted fair queuing strategy. Capacity which has not been reserved at all, is also given to a "best effort" class of service.

A simulator module for the build-up phase has been provided by the partners at Technische Universität Darmstadt and has afterwards been extended to the full functionality of Integrated Services.

There are no simple analytical results for Integrated Services, due to the build-up and clearing phase.

## 4.5 Development of the Computer-Based Procedure

References: [Fall & Varadhan 02, Greis 02, Heckmann et al. 03-1]

The architecture of the toolset of the computer based procedure is shown in Fig. 1. As already explained, the start situation may be realized in the testbed, the G-WiN (or any other measurable physical network), or the simulator. To solve one of the seven extrapolation cases, one may use the testbed itself, the simulator or the analytic approach. If the start situation is given in the testbed and the target is to be realized in the testbed, the extrapolation problem vanishes. Nevertheless, this is an important scenario, because the extrapolation technique by simulation or analytical means can be validated by variation of the situation in the testbed, which has been done. In general, however, the simulator will be used to realize the target situation. In this case, the scenario generator, which renders the topology (including all performance parameters, routing and QoS mechanisms) has to be fed with the start situation; if the start situation is given in the testbed, this is very simple, because the testbed is configured and put under load by the same scenario generator.

If the start situation is a major part of a large network, as the G-WiN e.g., the testbed will not do, of course. In this case the simulator is the adequate tool, by which the target situation is realized and evaluated. In principle, analytical methods

- M/G/1, by the P/K-formula
- G/G/1, by the Krämer/Whitt-formulas
- Best/worst case analysis

can be applied, with a very limited scope: min/max/mean waiting time, max jitter, for single subsystems, a router with subsequent line, e.g.; additionally, the character of the arrival process generally is not known in detail. If the waiting time is already measured in the start situation, the analytical approach may render simple transformations to a different utilization, due to a different arrival or service rate.

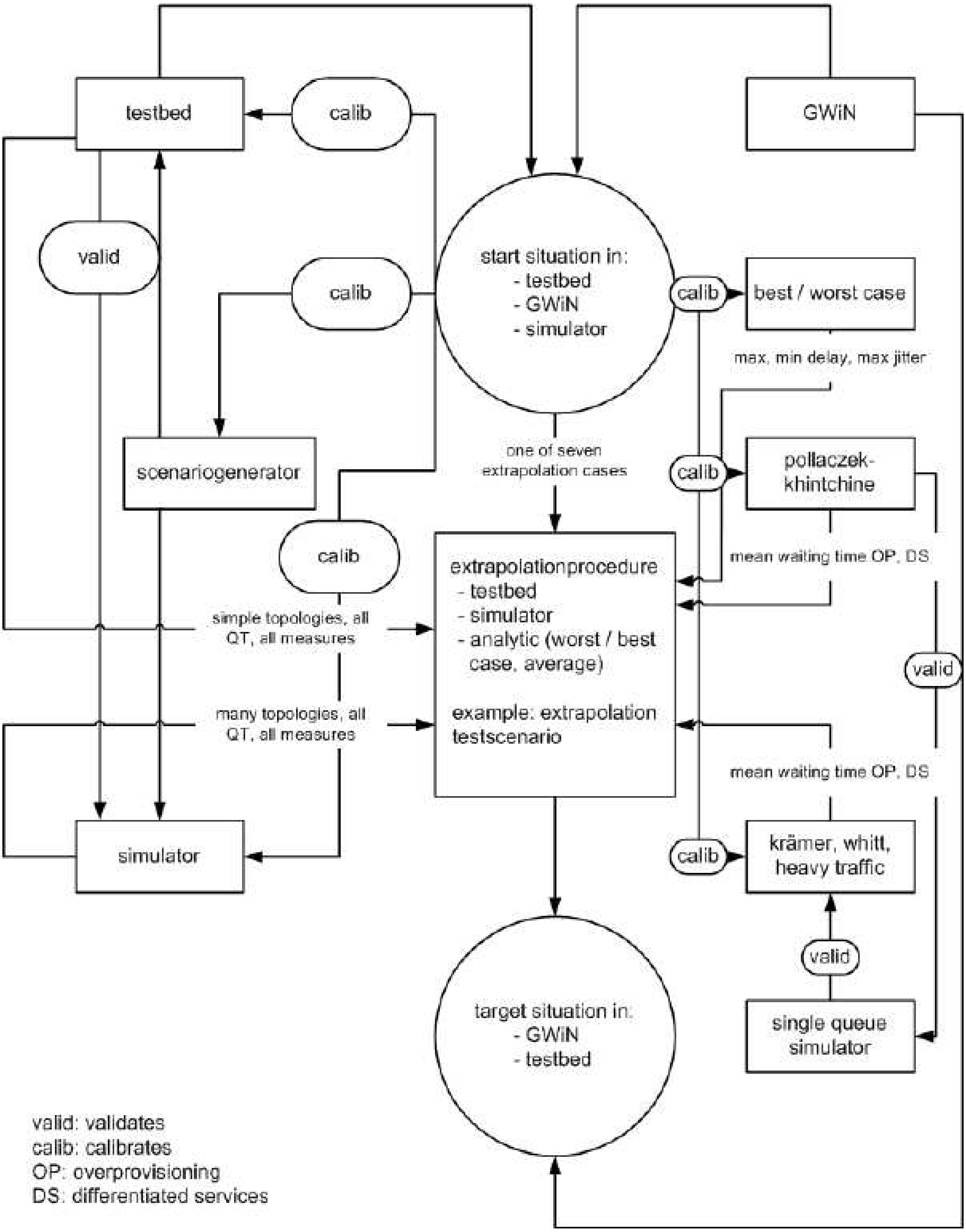



**Figure 4.1**: Extrapolationscheme

## 4.6 Validation of the Procedure

References: [Allen 90, Whitt 88, Lipsky et al.]

Validation demonstrates the correctness of a model, mostly in a statistical way and only to an acceptable degree. For the extrapolation case it has to be shown that the extrapolation tool renders results of adequate preciseness within a subset of the parameter space of the model used for extrapolation.

The first step of extrapolation is fitting the model (testbed for the network of interest, G-WiN e.g., simulator or analytical tools for the testbed or network of interest) to the start situation by setting the model parameters. This step is called calibration. If it is successful, the model behaves as the "real" start situation. If after setting the model parameters to the target situation, the model shows the behavior of the "real" target situation, the model has been validated for this point in parameter space.

For the purpose of the project (see Figure 4.1), the simulator has been validated against the testbed; a validation against the G-WiN had been desirable, but was impossible because the operational parameters of the network could not be set in an arbitrary way. The single server analytical tools (Krämer/Whitt formulas) have been validated against a single server queue simulator provided by Hans Peter Schwefel [Lipsky et al.], which, by its simplicity seems trustworthy, but was additionally validated against the M/G/1-P/K-formula.

The results were:

- The testbed was run on a set of K points in the parameter space. A point which is described by (Network Configuration, QoS Technology, Load Spectrum, QoS Spectrum) was selected as the start situation, the simulator was calibrated and shown to be consistent with the behaviour of the testbed at this point. Its parameters were set to the K points in parameter space and the simulator delivered results which were consistent to those of the testbed within a small range. So the simulator extrapolation method was shown to be valid in the space defined by the test set. This is widely due to the scenario generator, which specifies the situation for the testbed and the simulator in an equal manner.
- The single server queue simulator behaved in the M/G/1 case as predicted by the P/K formula.
- By the single server queue simulator the analytical approaches for isolated nodes were validated at 215 points in a wide space:
  - the heavy traffic formula is inferior to the Krämer formula in all cases except some of the 95 % utilization cases. For small utilizations it is severely misleading and asymptotically incorrect for 0 % utilization
  - the Whitt formula is inferior to the Krämer formula, except for few cases. It is weak at low utilization.
  - the Krämer formula tends to underestimate the mean waiting time under low utilization and overestimate under high utilization. This is particularly striking if the coefficient of variation of the interarrival time is much larger than that of the service time, where the results may be wrong by more than 100 %. A large power tail truncation parameter $T$ (see [Lipsky et al.]) deteriorates the results.

– except these cases, the Krämer formula gives results which differ at most 10 % from those of simulation.

# Chapter 5

# Application Examples

## 5.1 Extrapolation Testscenario (ETS)

The Extrapolation Testscenario (ETS) is the start point of the extrapolation. It is specified based on the definition of an Extrapolation Scenario (see [Pandit et al. 03-2]) and consists of the following Network Configuration, Load Spectrum, QoS Technology and QoS Spectrum:

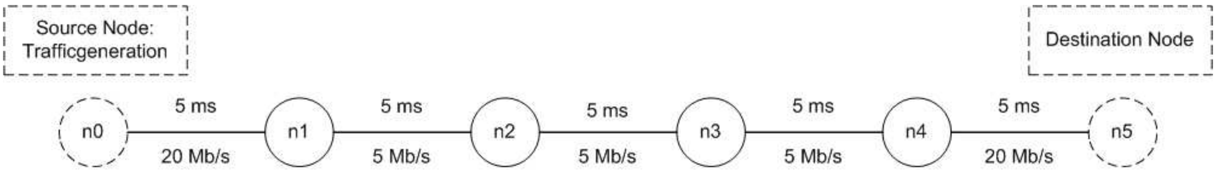



**Figure 5.1**: ETS Topology

Figure 5.1 shows the ETS topology, which is a result of the following **Network Configuration** $NC$:

$NC = \{T, B, LV, R\}$ consists of:

- the topology $T = \{ID, N, E\}$, with $ID = ETS$, $N = \{n_0, n_1, n_2, n_3, n_4, n_5\}$ nodes and $E = \{e_{0,1}, e_{1,2}, e_{2,3}, e_{3,4}, e_{4,5}\}$ edges
- the set of bandwidths $B = \{B_{0,1}, B_{1,2}, B_{2,3}, B_{3,4}, B_{4,5}\}$, with $B_{0,1} = B_{4,5} = 20Mb/s$ and $B_{1,2} = B_{2,3} = B_{3,4} = 5Mb/s$
- the set of link delays $LV = \{LV_{0,1}, LV_{1,2}, LV_{2,3}, LV_{3,4}, LV_{4,5}\}$, with $LV_{0,1} = LV_{1,2} = LV_{2,3} = LV_{3,4} = LV_{4,5} = 5ms$
- routing information $R = \{\text{IP routing}\}$

First a traffic mix of CBR and TCP traffic is used for the **Load Spectrum** ($LS$):
$LS = \{VLS_{CBR}, VLS_{TCP}\}$
$VLS_{CBR} = (SD_{CBR}, TT_{CBR})$
$SD_{CBR} = \{(n_0, n_5)\}$
$TT_{CBR} = \{\text{CBR generator}\}$
$VLS_{TCP} = (SD_{TCP}, TT_{TCP})$
$SD_{TCP} = \{(n_0, n_5)\}$
$TT_{TCP} = \{\text{background traffic, TCP generator}\}$

At this point no **QoS Technology** ($QT$) is selected. The BE service is used. All nodes use FIFO queues with a queue length of 50 packets independent of the packet size.
$QT = \{\}$

**QoS Spektrum** ($QS$)**:**
$QS = \{VQS_{VIDEO}\}$
$VQS_{VIDEO}$ = (0.1 %, 100 ms, 50 ms, -)

### 5.1.1 Sensitivity Analysis

A sensitivity analysis is the variation of single Extrapolation Testscenario parameters to find the exact influence of every parameter on the QoS factors (delay, jitter and loss). To investigate the seven extrapolation cases of interest, the traffic load, traffic proportions and traffic mix is varied beginning at the start point. Three different traffic mixes with the following proportion variation and Load Spectrum are simulated:

Traffic Mixes:

- CBR and TCP traffic:
  **Variation:**
  variation of the number of CBR flows:
  1 CBR flow, 10 CBR flows
  variation of the number of TCP flows for each number of CBR flows:
  1 TCP flow, 2 TCP flows, 3 TCP flows, 4 TCP flows, 5 TCP flows, 6 TCP flows, 7 TCP flows

- video and TCP traffic:
  **Load Spektrum** ($LS$)**:**
  $LS = \{VLS_{VIDEO}, VLS_{TCP}\}$
  $VLS_{VIDEO} = (SD_{VIDEO}, TT_{VIDEO})$
  $SD_{VIDEO} = \{(n_0, n_5)\}$
  $TT_{VIDEO}$ = {H.323 videoconferencing, traffic generated by trace files}
  $VLS_{TCP} = \{SD_{TCP}, TT_{TCP}\}$
  $SD_{TCP} = \{(n_0, n_5)\}$
  $TT_{TCP}$ = {background traffic, TCP generator}
  **Variation:**
  variation of the number of video traces:
  1 video trace, 10 video traces
  variation of the number of TCP flows for each number of video traces:
  1 TCP flow, 7 TCP flows

- CBR, video and TCP traffic:
  **Load Spektrum** ($LS$)**:**
  $LS = \{VLS_{VIDEO}, VLS_{CBR}, VLS_{TCP}\}$
  $VLS_{VIDEO} = (SD_{VIDEO}, TT_{VIDEO})$
  $SD_{VIDEO} = \{(n_0, n_5)\}$

$TT_{VIDEO} = \{$H.323 videoconferencing, traffic generated by trace files$\}$
$VLS_{CBR} = (SD_{CBR}, TT_{CBR})$
$SD_{CBR} = \{(n_0, n_5)\}$
$TT_{CBR} = \{$CBR generator$\}$
$VLS_{TCP} = \{SD_{TCP}, TT_{TCP}\}$
$SD_{TCP} = \{(n_0, n_5)\}$
$TT_{TCP} = \{$background traffic, TCP generator$\}$
**Variation:**
parallel variation of the number of CBR flows and video traces:
(1 video trace, 1 CBR flow), (5 video traces, 5 CBR flows)
variation of the number of TCP flows for each number of video traces:
1 TCP flow, 7 TCP flows

In this report we only show the video and TCP traffic mix simulations in detail, but the later described simulation results include also the results of the other traffic mixes. The simulation experiment is divided into parts with different QoS Technology:

- Simulations with Best Effort
- Simulations with IntServ
  The queue length for the IntServ reservations are calculated for every traffic mix.
- Simulations with DiffServ (Olympic)
  To realize DiffServ its functionality has to be implemented into the nodes of the DiffServ domain. Figure 5.2 shows the DiffServ domain with its edge nodes that execute policing, admission control (deactivated) and marking of packets. The queue lengths are selected as follows: EF = 10, BE = 40, QBSS = 50. The Best Effort queue length is divided into the EF and BE (DiffServ class) queue length. To minimize the delay and jitter the EF queue length is short. The QBSS queue is longer because the applications that selected the QBSS class are not delay and jitter sensitive, but have the lowest priority and can be loss sensitive.
- Simulations with DiffServ (Default)
  The queue lengths are selected as follows: EF = 10, AF 1 = 13, AF 2 = 13, Af 3 = 13.

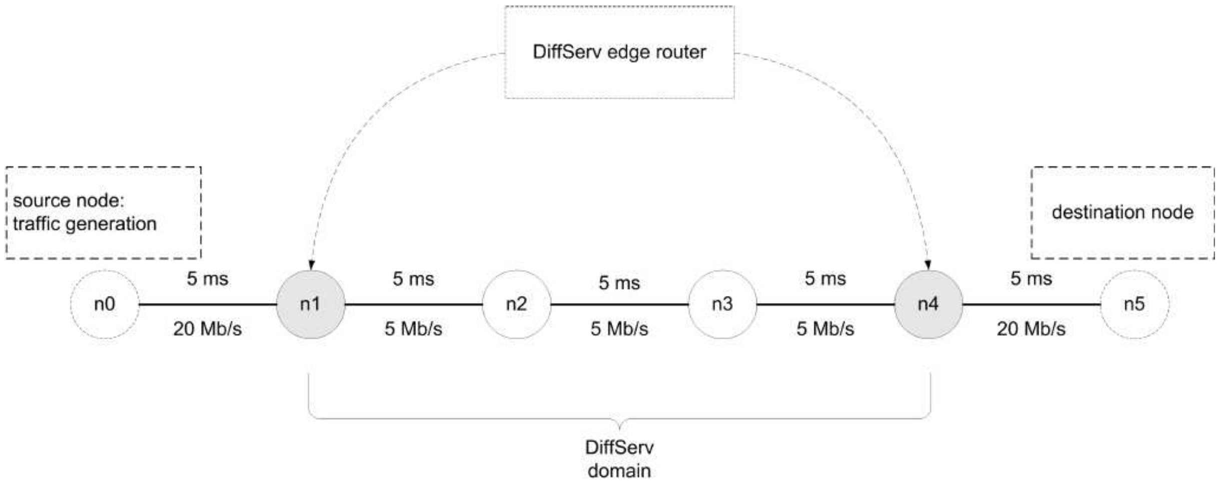
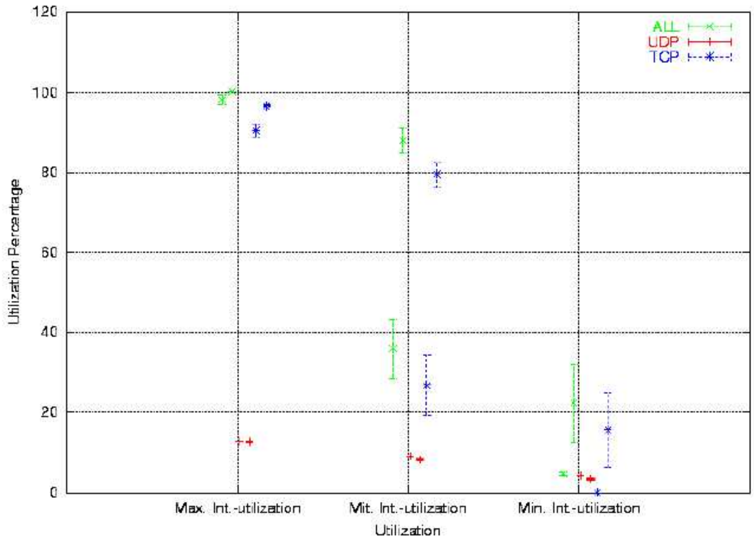



**Figure 5.2**: ETS DiffServ Topology

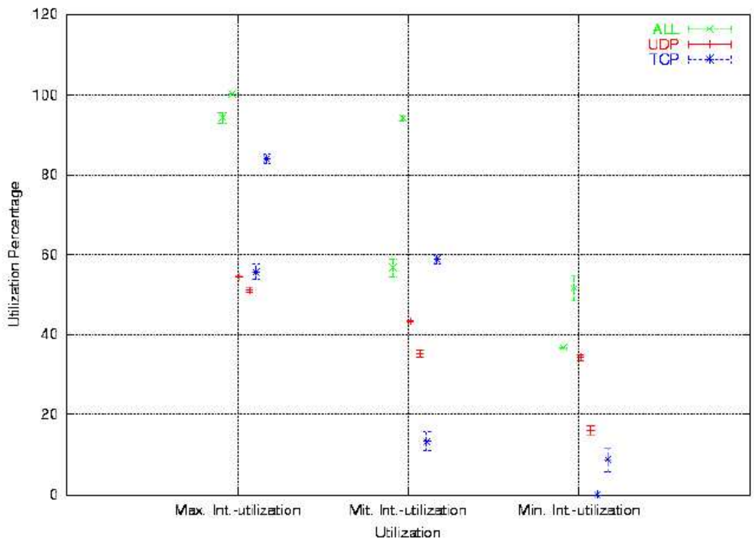
20 simulation runs with different seeds are executed for every experiment. The 90 % confidence intervals are plotted. Every single flow is conform to its policies. Only IntServ uses an admission control.

**Summary of Best Effort simulation results:**

Average utilizations from 20 % up to 94 % on the bottleneck link (n1-n2, see Figure 5.1) are investigated. The average utilization of the bottleneck link caused by QoS sensitive traffic reaches from 1 % to 43 % (see Figure 5.3). The maximum utilization is 100 %. Already low utilizations lead to inaceptable QoS values for the sensitive traffic. The maximum delay exceeds the 100 ms (see Figure 5.4), the maximum jitter the 50 ms (see Figure 5.5) and the loss increases up to 58 % (see Figure 5.6). The delay has an upper bound. The reason for this bound is the maximum queue length. If the queue is full and can not be emptied all incoming packets are dropped. This leads to the high loss percentage.

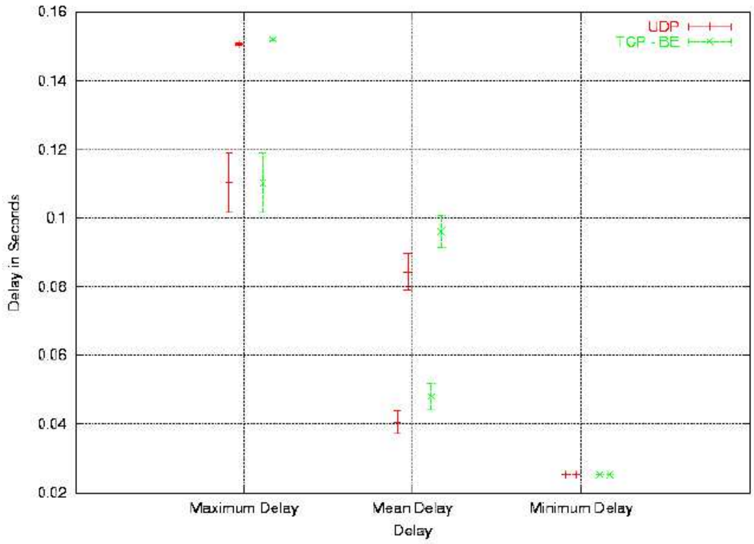



(a) Utilization of the bottleneck link caused by a video trace and a varying number of TCP flows

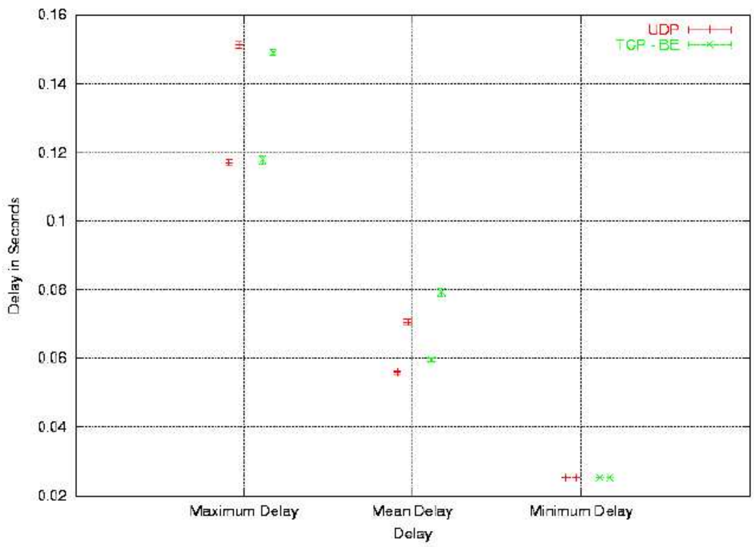



(b) Utilization of the bottleneck link caused by 10 video traces and a varying number of TCP flows

**Figure 5.3**: Utilization of the bottleneck link caused by a video and TCP traffic mix using Best Effort

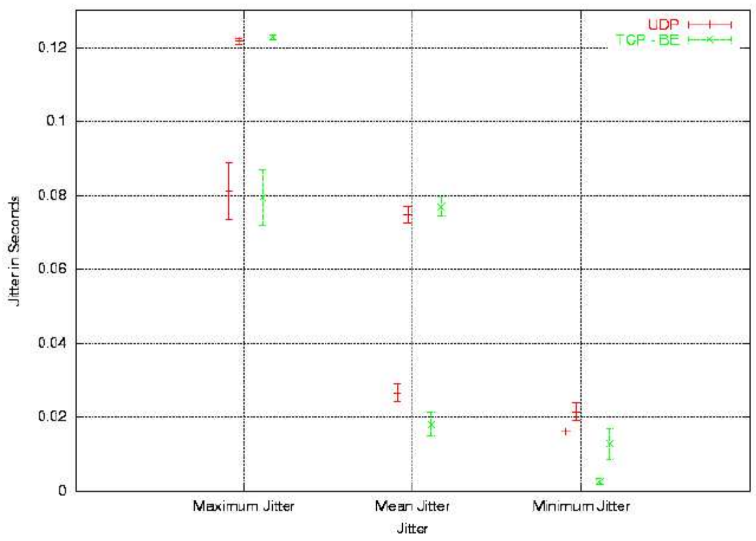



(a) End-to-End delay by a video trace and a varying number of TCP flows

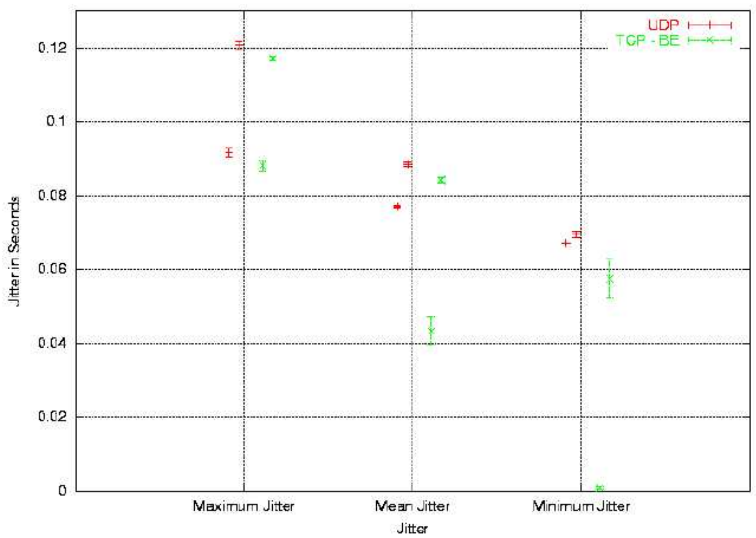



(b) End-to-End delay by 10 video traces and a varying number of TCP flows

**Figure 5.4**: End-to-End delay by a video and TCP traffic mix using Best Effort

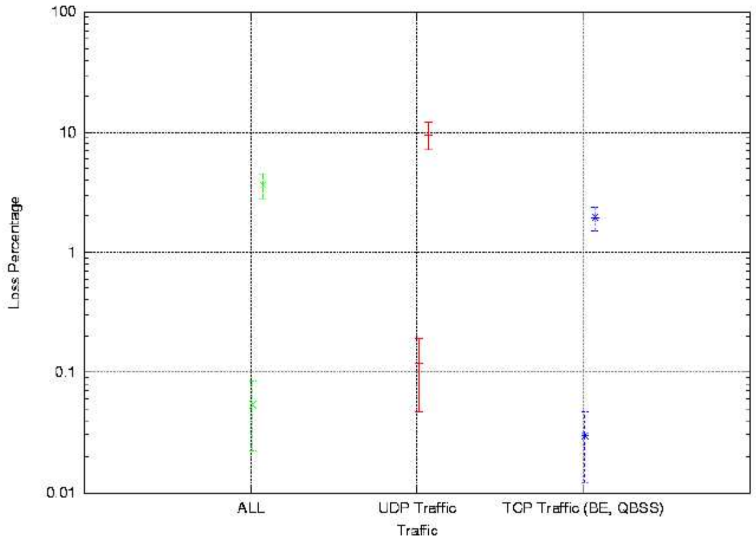



(a) Jitter caused by a video trace and a varying number of TCP flows

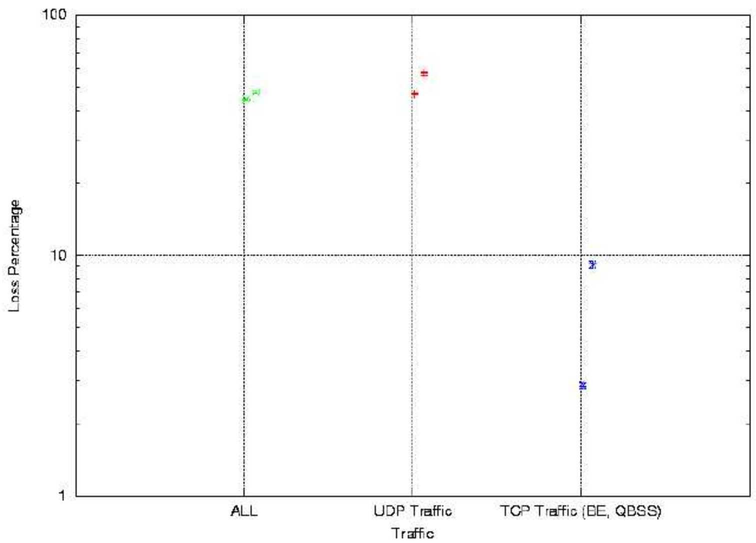



(b) Jitter caused by 10 video traces and a varying number of TCP flows

**Figure 5.5**: Jitter caused by a video and TCP traffic mix using Best Effort

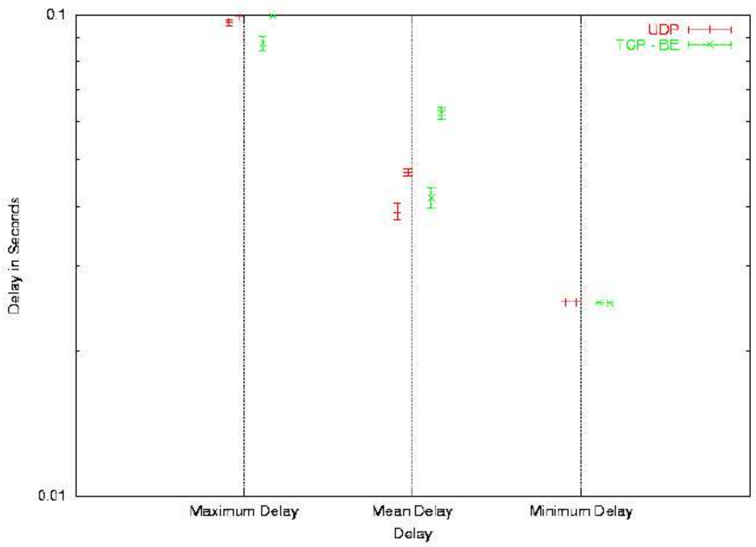



(a) Loss caused by a video trace and a varying number of TCP flows

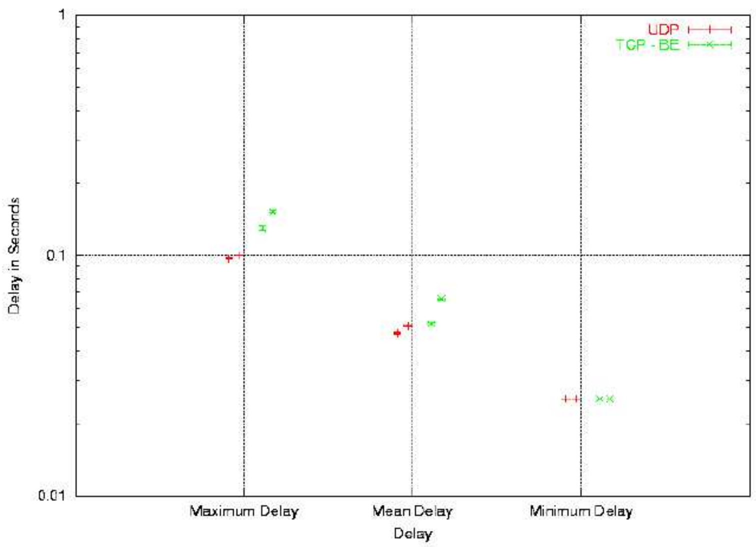



(b) Loss caused by 10 video traces and a varying number of TCP flows

**Figure 5.6**: Loss caused by video and TCP traffic mix using Best Effort

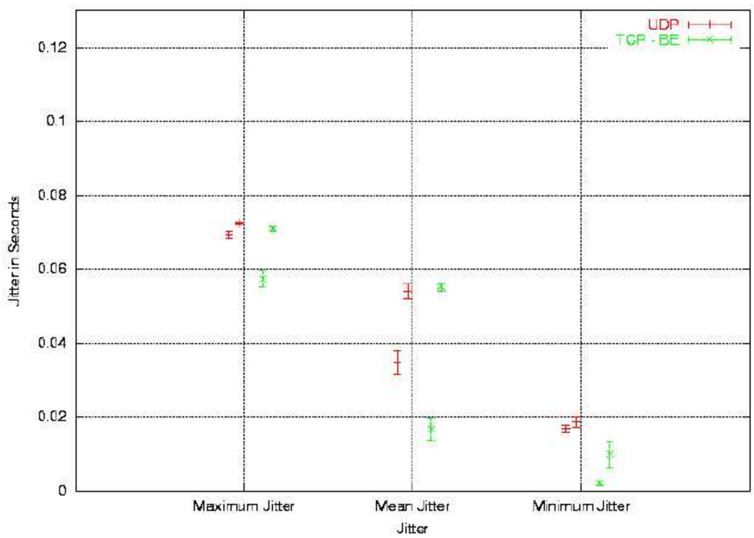



(a) End-to-End delay by a video trace and a varying number of TCP flows

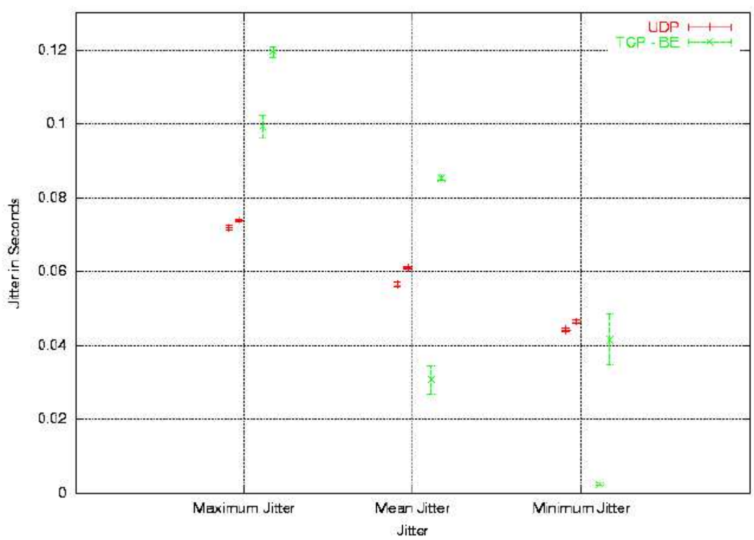



(b) End-to-End delay by 10 video traces and a varying number of TCP flows

**Figure 5.7**: End-to-End delay by video and TCP traffic mix using IntServ

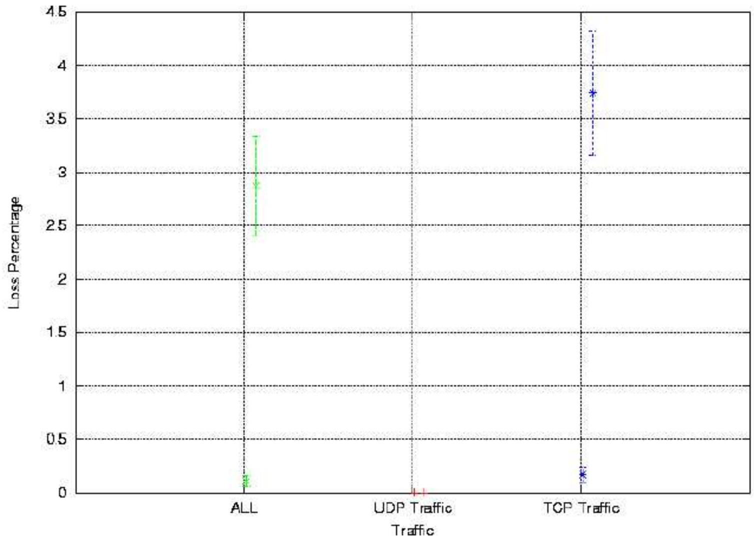



(a) Jitter caused by a video trace and a varying number of TCP flows

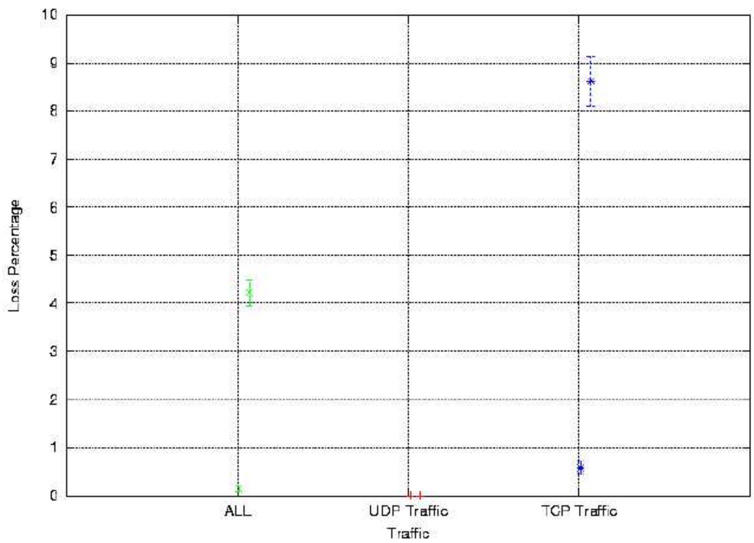



(b) Jitter caused by 10 video traces and a varying number of TCP flows

**Figure 5.8**: Jitter caused by video and TCP traffic mix using IntServ




(a) Loss caused by a video trace and a varying number of TCP flows




(b) Loss caused by 10 video traces and a varying number of TCP flows

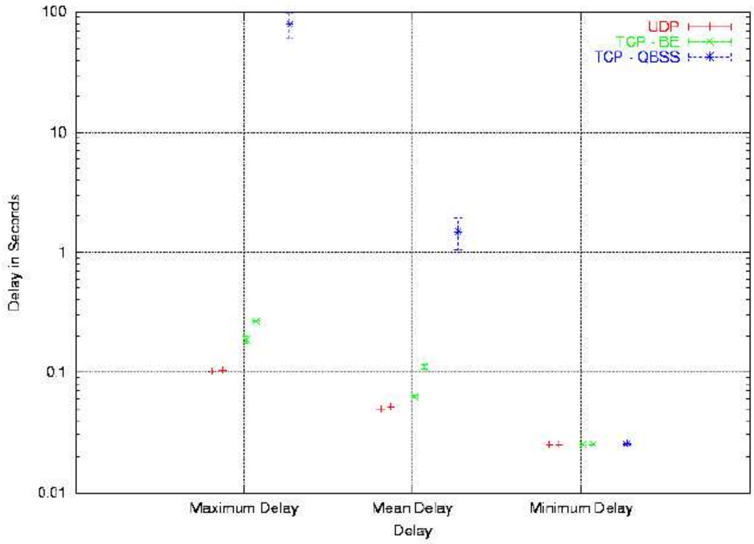
**Figure 5.9**: Loss caused by video and TCP traffic mix using IntServ

**Summary of IntServ simulation results:**

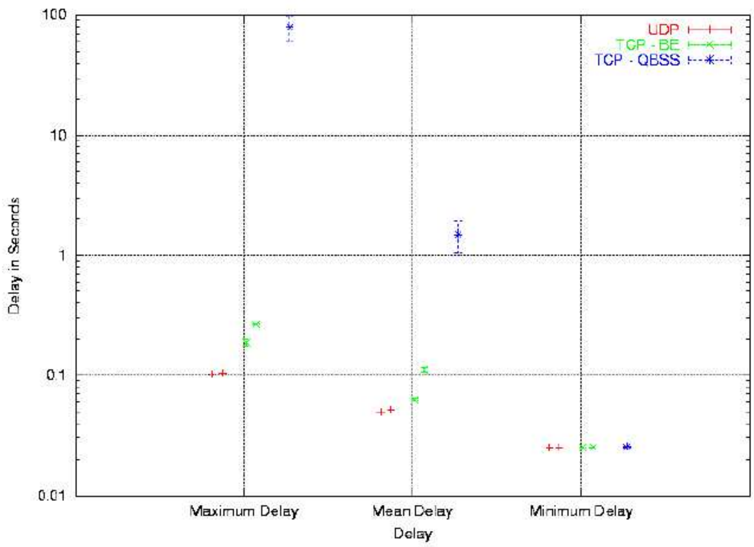
IntServ fulfills the guarantees that are promised. The QoS sensitive traffic does not exceed the specified delay bound of 100 ms (see Figure 5.7) and loss does not occur (see Figure 5.9). This can be garanteed because of the calculated and reserved buffer size and transmission rate. But the jitter exceeds the 50 ms (see Figure 5.8). IntServ gives no guarantee for jitter bounds. The problem can be solved by reducing the specified delay bound. Additionally, the part of reserved bandwidth in proportion to existing bandwidth should be tested, because too much loss makes TCP traffic useless (see Figure 5.9). Another possibility is the implementation of an admission control for TCP traffic.

**Summary of DiffServ (Olympic) simulation results:**

The maximum delay is about 100 ms at very high utilizations (see Figure 5.10). This is the upper limit fixed by the queue length of the QoS sensitive traffic class. If the queue is full the loss increases rapidly (see Figure 5.12). This makes an implementation of an admission control necessary. This bound is investigated in later simulation experiments (see Section 5.1.2). The jitter exceeds the 50 ms (see Figure 5.11). The jitter can be reduced by a delay reduction. This can be realized by a shorter queue which makes an earlier interference of the admission control necessary. The delay and jitter of the BE traffic is aceptable; the delay and jitter of the QBSS traffic is very high. But this is no problem because the QBSS traffic is not delay or jitter sensitive. The loss of both TCP classes (BE and QBSS) can get very high at high utilizations. An expansion of the BE and QBSS queues can reduce the loss but increase the delay and jitter. Another possibility is an admission control for BE and QBSS traffic. Both depends on the application demands.

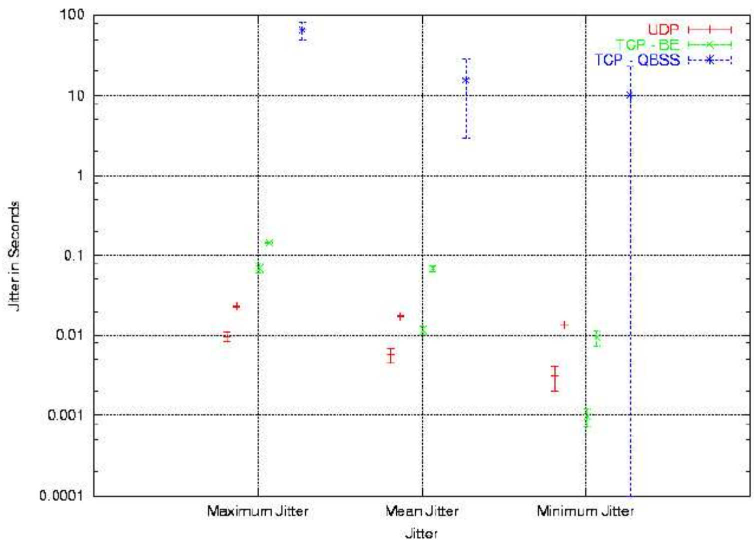



(a) End-to-End delay by a video trace and a varying number of TCP flows

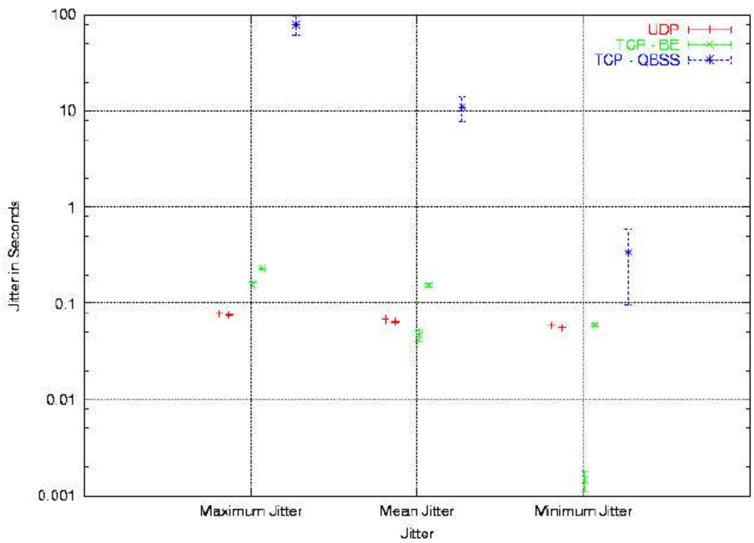



(b) End-to-End delay by 10 video traces and a varying number of TCP flows

**Figure 5.10**: End-to-End delay by video and TCP traffic mix using DiffServ (Olympic)

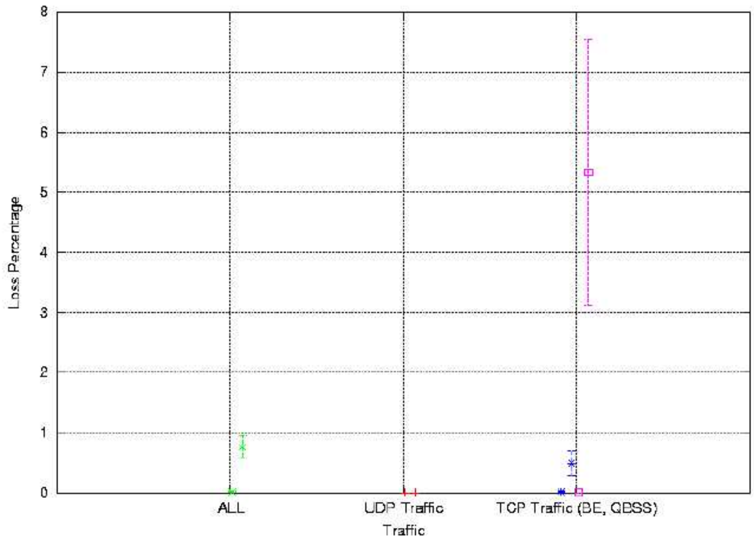



(a) Jitter caused by a video trace and a varying number of TCP flows

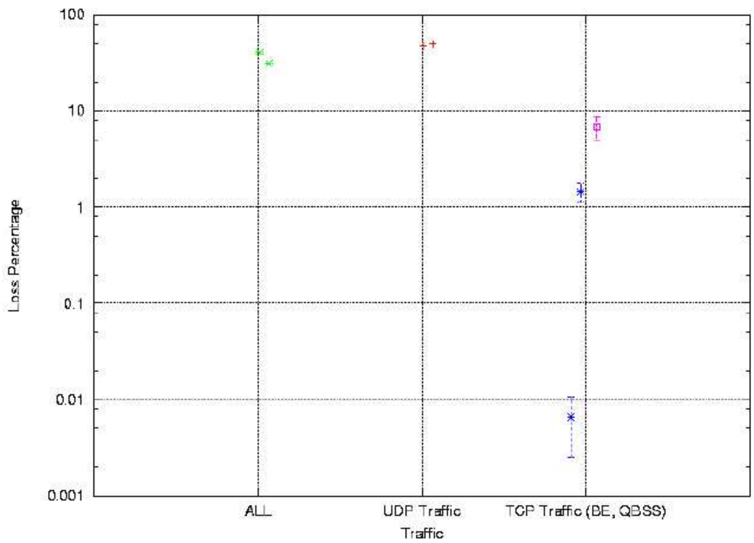



(b) Jitter caused by 10 video traces and a varying number of TCP flows

**Figure 5.11**: Jitter caused by video and TCP traffic mix using DiffServ (Olympic)

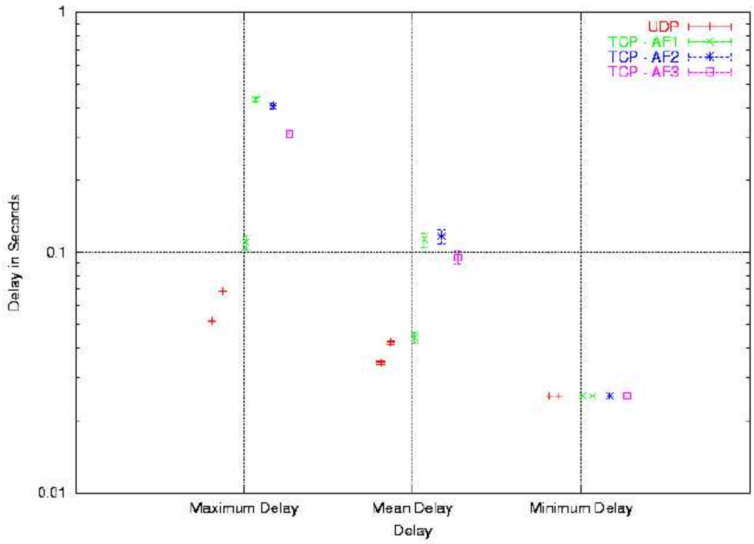



(a) Loss caused by a video trace and a varying number of TCP flows

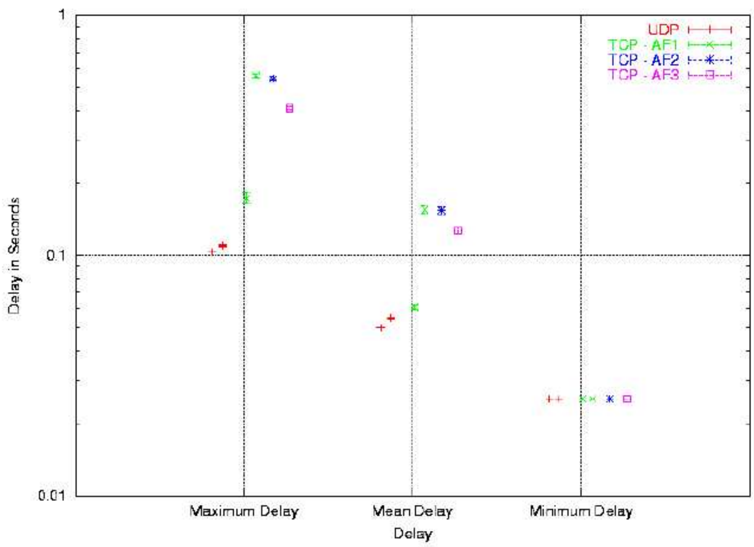



(b) Loss caused by 10 video traces and a varying number of TCP flows

**Figure 5.12**: Loss caused by video and TCP traffic mix using DiffServ (Olympic)

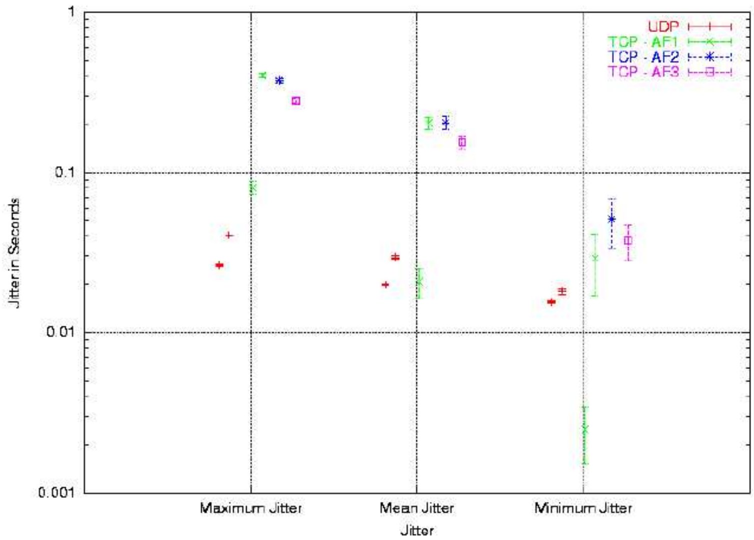



(a) End-to-End delay by a video trace and a varying number of TCP flows

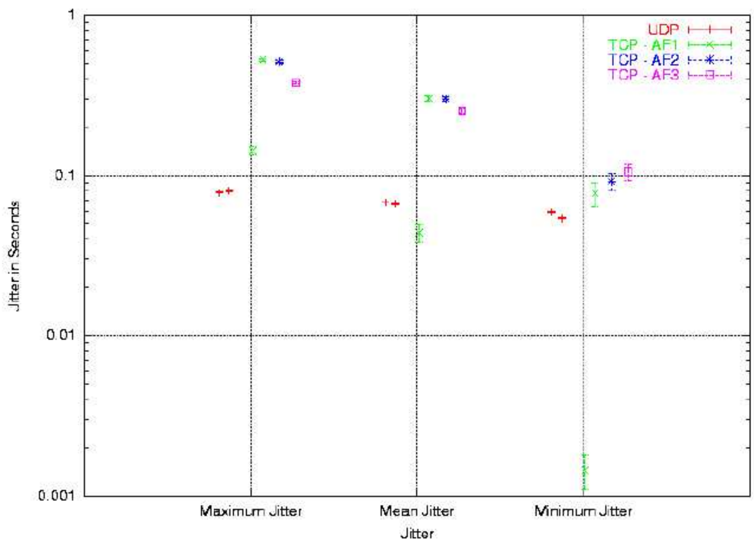



(b) End-to-End delay by 10 video traces and a varying number of TCP flows

**Figure 5.13**: End-to-End delay by video and TCP traffic mix using DiffServ (Default)

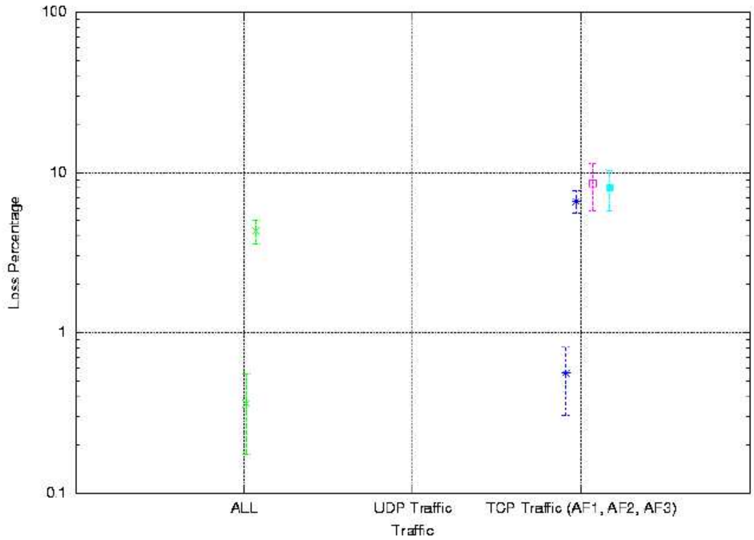



(a) Jitter caused by a video trace and a varying number of TCP flows

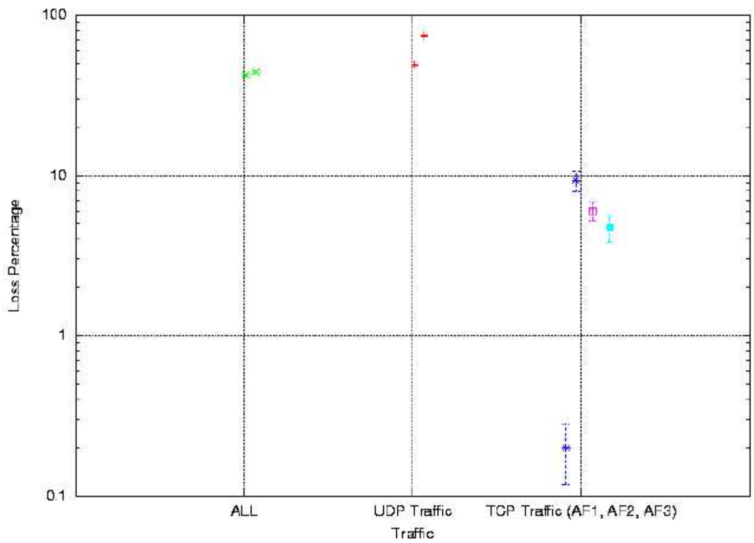



(b) Jitter caused by 10 video traces and a varying number of TCP flows

**Figure 5.14**: Jitter caused by video and TCP traffic mix using DiffServ (Default)




(a) Loss caused by a video trace and a varying number of TCP flows



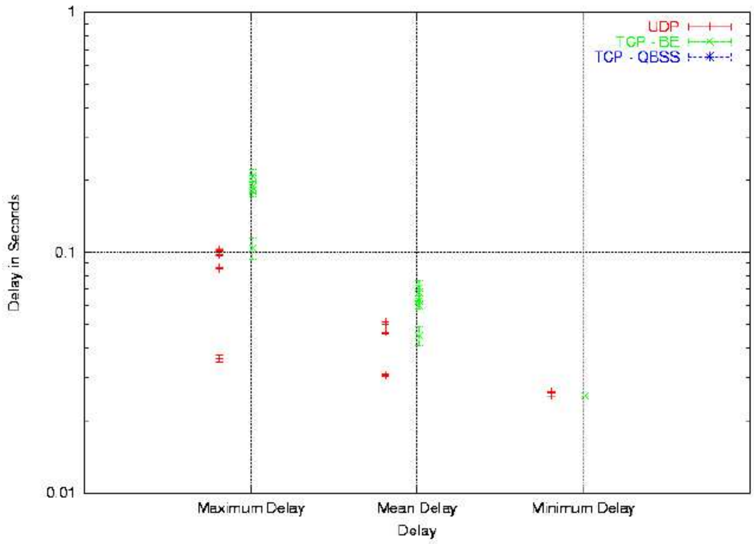

(b) Loss caused by 10 video traces and a varying number of TCP flows

**Figure 5.15**: Loss caused by video and TCP traffic mix using DiffServ (Default)

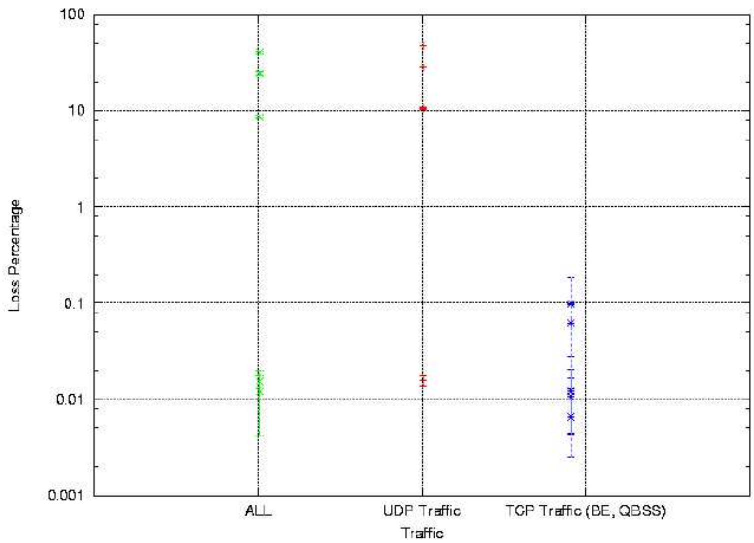
**Summary of DiffServ (Default) simulation results:**
The maximum delay is about 100 ms by very high utilizations (see Figure 5.13). The TCP delay of all three AF classes is always under 600 ms. The jitter is once more too high but can be reduced as mentioned before. The loss increases up to 50 % because of full queues which can be prevented by an admission control. Loss occurs earlier compared to the Olympic approach because of the used WRR algorithm (see [Hoffmann]). The loss of the AF classes goes up to 10 % which can be reduced by queue expansion or admission control.

### 5.1.2 Admission Control Limit

As the simulations showed, DiffServ approaches can work without admission control until a limit that depends on the queue size of the EF class. If the queue is full and can not be emptied, the loss increases extremely. As an example, we try to find this limit for DiffServ (Olympic) by increasing the number of generated video traces. The number of video traces starts from 1 and goes up to 10. Additionally 1 TCP session is simulated as background traffic.
Figure 5.16 shows that the limt lies between 3 and 4 simulated video traces. At this point the loss increases from 0.015 % up to 10.690 %. The limit corresponds to an average utilization of the bottleneck link caused by video traffic on the bottleneck link between 27.3 % and 33.8 %. If this value is topped, an implementation of an admission control is inevitable. But this value is only correct for this special kind of generated traffic and topology and is not valid in general. For the other traffic mixes different limits are found.




(a) End-to-End delay by a varying number of video traces and a TCP flow




(b) Loss caused by a varying number of video traces and a TCP flow

**Figure 5.16**: Utilization limit

### 5.1.3 Case Study

The case study is a summary of the simulation results of the QoS sensitive traffic. The results are presented in relation to the QoS factor limits for video conferencing. Only loss is given by its exact percentage.

| BE | | | |
|---|---|---|---|
| Sensitive Parameter Variation | Delay | Jitter | Loss |
| CBR, TCP | | | |
| Number of CBR Flows 1 Number of TCP Flows increases from 1 to 7 | > 100 ms | > 50 ms | -7 % |
| Number of CBR Flows 10 Number of TCP Flows increases from 1 to 7 | > 100 ms | > 50 ms | -10 % |
| Video, TCP | | | |
| Number of Video Traces 1 Number of TCP Flows increases from 1 to 7 | > 100 ms | > 50 ms | -10 % |
| Number of Video Traces 10 Number of TCP Flows increases from 1 to 7 | > 100 ms | > 50 ms | -57 % |
| CBR, Video, TCP | | | |
| Number of CBR Flows 1 Number of Video Traces 1 Number of TCP Flows increases from 1 to 7 | > 100 ms | > 50 ms | -5 % (CBR) / -12 % (Video) |
| Number of CBR Flows 5 Number of Video Traces 5 Number of TCP Flows increases from 1 to 7 | > 100 ms | > 50 ms | -41 % (CBR) / -47 % (Video) |

**Table 5.1**: ETS case study BE

| IntServ | | | |
|---|---|---|---|
| Sensitive Parameter Variation | Delay | Jitter | Loss |
| CBR, TCP | | | |
| Number of CBR Flows 1 Number of TCP Flows increases from 1 to 7 | $< 100$ ms | $> 50$ ms | 0 % |
| Number of CBR Flows 10 Number of TCP Flows increases from 1 to 7 | $< 100$ ms | $> 50$ ms | 0 % |
| Video, TCP | | | |
| Number of Video Traces 1 Number of TCP Flows increases from 1 to 7 | $< 100$ ms | $> 50$ ms | 0 % |
| Number of Video Traces 10 Number of TCP Flows increases from 1 to 7 | $< 100$ ms | $> 50$ ms | 0 % |
| CBR, Video, TCP | | | |
| Number of CBR Flows 1 Number of Video Traces 1 Number of TCP Flows increases from 1 to 7 | $< 100$ ms | $> 50$ ms | 0 % (CBR) / 0 % (Video) |
| Number of CBR Flows 5 Number of Video Traces 5 Number of TCP Flows increases from 1 to 7 | $< 100$ ms | $> 50$ ms | 0 % (CBR) / 0 % (Video) |

**Table 5.2**: ETS case study IntServ

| DiffServ (Olympic) | | | |
|---|---|---|---|
| Sensitive Parameter Variation | Delay | Jitter | Loss |
| CBR, TCP | | | |
| Number of CBR Flows 1<br>Number of TCP Flows increases from 1 to 7 | $< 100$ ms | $< 50$ ms | 0 % |
| Number of CBR Flows 10<br>Number of TCP Flows increases from 1 to 7 | $< 100$ ms | $< 50$ ms | 0 % |
| Video, TCP | | | |
| Number of Video Traces 1<br>Number of TCP Flows increases from 1 to 7 | $< 100$ ms | $< 50$ ms | 0 % |
| Number of Video Traces 10<br>Number of TCP Flows increases from 1 to 7 | $\leq 100$ ms | $> 50$ ms | -50 % |
| CBR, Video, TCP | | | |
| Number of CBR Flows 1<br>Number of Video Traces 1<br>Number of TCP Flows increases from 1 to 7 | $< 100$ ms | $< 50$ ms | 0 % (CBR) / 0 % (Video) |
| Number of CBR Flows 5<br>Number of Video Traces 5<br>Number of TCP Flows increases from 1 to 7 | $\leq 100$ ms | $> 50$ ms | -67 % (CBR) / -26 % (Video) |

**Table 5.3**: ETS case study DiffServ (Olympic)

| DiffServ (Default) | | | |
|---|---|---|---|
| Sensitive Parameter Variation | Delay | Jitter | Loss |
| CBR, TCP | | | |
| Number of CBR Flows 1 Number of TCP Flows increases from 1 to 7 | < 100 ms | < 50 ms | 0 % |
| Number of CBR Flows 10 Number of TCP Flows increases from 1 to 7 | < 100 ms | < 50 ms | 0 % |
| Video, TCP | | | |
| Number of Video Traces 1 Number of TCP Flows increases from 1 to 7 | < 100 ms | < 50 ms | 0 % |
| Number of Video Traces 10 Number of TCP Flows increases from 1 to 7 | > 100 ms | > 50 ms | -74 % |
| CBR, Video, TCP | | | |
| Number of CBR Flows 1 Number of Video Traces 1 Number of TCP Flows increases from 1 to 7 | < 100 ms | < 50 ms | 0 % (CBR) / 0 % (Video) |
| Number of CBR Flows 5 Number of Video Traces 5 Number of TCP Flows increases from 1 to 7 | ≤ 100 ms | > 50 ms | -67 % (CBR) / -60 % (Video) |

**Table 5.4**: ETS case study DiffServ (Default)

### 5.1.4 Best / Worst Case Analysis

The worst case analysis compares:

- average end-to-end delay at different utilizations of the bottleneck link, simulated
- best case end-to-end delay, calculated based on QoS architecture
- worst case end-to-end delay, calculated based on QoS architecture
- average jitter at different utilizations of the bottleneck link, simulated
- worst case jitter, calculated based on QoS architecture

for the QoS sensitive traffic of the three described traffic mixes for each QoS Technology (not IntServ, delay bound is given as input see [HOFFMANN]). Here only the calculation for video and

TCP traffic with the Olympic technology is shown. But the summary of the results includes also the calculation of the other traffic mixes. The Olympic architecture (see [Hoffmann]) leads to the following worst case delay and jitter formula. The inserted parameter values are based on the ETS Topology parameters (see Section 5.1):

$$\begin{aligned} D_{e2e} \leq\; & \left(D_{prop,0} + \frac{(Q_{max,0}-1)*MTU_0}{\gamma_0} + \frac{MTU_0}{\gamma_0} + \frac{M}{\gamma_0}\right) + \\ & \left(D_{prop,1} + \frac{(Q_{max,1}-1)*MTU_{EF,1}}{\gamma_1} + \frac{MTU_1}{\gamma_1} + \frac{M}{\gamma_1}\right) + \\ & \left(D_{prop,2} + \frac{(Q_{max,2}-1)*MTU_{EF,2}}{\gamma_2} + \frac{MTU_2}{\gamma_2} + \frac{M}{\gamma_2}\right) + \\ & \left(D_{prop,3} + \frac{(Q_{max,3}-1)*MTU_{EF,3}}{\gamma_3} + \frac{MTU_3}{\gamma_3} + \frac{M}{\gamma_3}\right) + \\ & \left(D_{prop,4} + \frac{(Q_{max,4}-1)*MTU_4}{\gamma_4} + \frac{MTU_4}{\gamma_4} + \frac{M}{\gamma_4}\right) \end{aligned} \tag{5.1}$$

$$\begin{aligned} D_{e2e} \leq\; & \left(0.005s + \frac{49*MTU_0}{20Mbit/s} + \frac{MTU_0}{20Mbit/s} + \frac{M}{20Mbit/s}\right) + \\ & \left(0.005s + \frac{9*MTU_{EF,1}}{5Mbit/s} + \frac{MTU_1}{5Mbit/s} + \frac{M}{5Mbit/s}\right) + \\ & \left(0.005s + \frac{9*MTU_{EF,2}}{5Mbit/s} + \frac{MTU_2}{5Mbit/s} + \frac{M}{5Mbit/s}\right) + \\ & \left(0.005s + \frac{9*MTU_{EF,3}}{5Mbit/s} + \frac{MTU_3}{5Mbit/s} + \frac{M}{5Mbit/s}\right) + \\ & \left(0.005s + \frac{49*MTU_4}{20Mbit/s} + \frac{MTU_4}{20Mbit/s} + \frac{M}{20Mbit/s}\right) \end{aligned} \tag{5.2}$$

$$\begin{aligned} J \leq\; & \left(\frac{(Q_{max,0}-1)*MTU_0}{\gamma_0} + \frac{MTU_0}{\gamma_0} + \frac{\Delta M_0}{\gamma_0}\right) + \\ & \left(\frac{(Q_{max,1}-1)*MTU_{EF,1}}{\gamma_1} + \frac{MTU_1}{\gamma_1} + \frac{\Delta M_1}{\gamma_1}\right) + \\ & \left(\frac{(Q_{max,2}-1)*MTU_{EF,2}}{\gamma_2} + \frac{MTU_2}{\gamma_2} + \frac{\Delta M_2}{\gamma_2}\right) + \\ & \left(\frac{(Q_{max,3}-1)*MTU_{EF,3}}{\gamma_3} + \frac{MTU_3}{\gamma_3} + \frac{\Delta M_3}{\gamma_3}\right) + \\ & \left(\frac{(Q_{max,4}-1)*MTU_4}{\gamma_4} + \frac{MTU_4}{\gamma_4} + \frac{\Delta M_4}{\gamma_4}\right) \end{aligned} \tag{5.3}$$

$$
\begin{aligned}
J \quad \leq \quad & \left(\frac{49 * MTU_0}{20Mbit/s} + \frac{MTU_0}{20Mbit/s} + \frac{\Delta M_0}{20Mbit/s}\right) + \\
& \left(\frac{9 * MTU_{EF,1}}{5Mbit/s} + \frac{MTU_1}{5Mbit/s} + \frac{\Delta M_1}{5Mbit/s}\right) + \\
& \left(\frac{9 * MTU_{EF,2}}{5Mbit/s} + \frac{MTU_2}{5Mbit/s} + \frac{\Delta M_2}{5Mbit/s}\right) + \\
& \left(\frac{9 * MTU_{EF,3}}{5Mbit/s} + \frac{MTU_3}{5Mbit/s} + \frac{\Delta M_3}{5Mbit/s}\right) + \\
& \left(\frac{49 * MTU_4}{20Mbit/s} + \frac{MTU_4}{20Mbit/s} + \frac{\Delta M_4}{20Mbit/s}\right)
\end{aligned} \tag{5.4}
$$

Video and TCP traffic lead to the following parameter values (see table 5.5), that deliver, inserted into the formulas, the results of table 5.6 and 5.7.

| traffic specific parameters |
|---|
| $MTU_0$ = 1500 Bytes |
| $MTU_1$ = 1500 Bytes |
| $MTU_2$ = 1500 Bytes |
| $MTU_3$ = 1500 Bytes |
| $MTU_4$ = 1500 Bytes |
| average $M$ = 782 Bytes |
| $\Delta M_0$ = 1344 Bytes |
| $\Delta M_1$ = 1344 Bytes |
| $\Delta M_2$ = 1344 Bytes |
| $\Delta M_3$ = 1344 Bytes |
| $\Delta M_4$ = 1344 Bytes |
| $MTU_{EF,1}$ = 1375 Bytes |
| $MTU_{EF,2}$ = 1375 Bytes |
| $MTU_{EF,3}$ = 1375 Bytes |

**Table 5.5**: Traffic specific parameters

| Delay Type | Delay [s] | Traffic Type |
|---|---|---|
| average Delay at 1 Video Trace, 1 TCP Flow | 0.030786 | Video Trace |
| maximum Delay at 1 Video Trace, 1 TCP Flow | 0.036178 | Video Trace |
| average Delay at 1 Video Trace, 7 TCP Flow | 0.036231 | Video Trace |
| maximum Delay at 1 Video Trace, 7 TCP Flow | 0.049383 | Video Trace |
| average Delay at 10 Video Traces, 1 TCP Flow | 0.049915 | Video Trace |
| maximum Delay at 10 Video Traces, 1 TCP Flow | 0.102970 | Video Trace |
| average Delay at 10 Video Traces, 7 TCP Flow | 0.051584 | Video Trace |
| maximum Delay at 10 Video Traces, 7 TCP Flow | 0.104516 | Video Trace |
| calculated Best Case Delay | 0.029379 | Video Trace |
| calculated Worst Case Delay | 0.155979 | Video Trace |

**Table 5.6**: Comparison of calculated and simulated delay values

| Jitter Type | Jitter [s] | Traffic Type |
|---|---|---|
| average Jitter at 1 Video Trace, 1 TCP Flow | 0.005758 | Video Trace |
| maximum Jitter at 1 Video Trace, 1 TCP Flow | 0.009751 | Video Trace |
| average Jitter at 1 Video Trace, 7 TCP Flow | 0.017237 | Video Trace |
| maximum Jitter at 1 Video Trace, 7 TCP Flow | 0.023111 | Video Trace |
| average Jitter at 10 Video Traces, 1 TCP Flow | 0.067853 | Video Trace |
| maximum Jitter at 10 Video Traces, 1 TCP Flow | 0.077723 | Video Trace |
| average Jitter at 10 Video Traces, 7 TCP Flow | 0.064454 | Video Trace |
| maximum Jitter at 10 Video Traces, 7 TCP Flow | 0.075449 | Video Trace |
| calculated Worst Case Jitter | 0.134126 | Video Trace |

**Table 5.7**: Comparison of calculated and simulated jitter values

**Summary of best / worst case calculation results:**
The minimum or best case end-to-end delay is the minimum time a packet needs for transmission from the source to the destination. This delay includes no waiting times in the router queues. It is necessary for calculating the worst case jitter. The maximum or worst case delay represents the highest delay that is theoretically possible with all router queues filled up to their maximum. The investigations show that this approach can be useful for technologies with short router queues, traffic with small packets and for high utilizations. For services with long router queues, e.g. BE, the calculations are misleading because never all queues are filled up to the maximum

at the same time as the calculation supposes. This means the longer the queues the higher the difference between realistic delays and the worst case calculation. Another factor is the packet size because the queue length often is measured in the number of packets and not in bits or bytes. This means the higher the packet size the higher the calculation error (CBR traffic better then video traffic). The last point is the utilization forced by the simulated traffic. A worst case calculation of a link is of course much nearer to a high utilized simulation result then a minimal utilized. But in some cases its enough to have a rough estimation of the worst delay, which can be done by the worst case calculation.

### 5.1.5 Average Delay Calculation

Additional to worst case calculations average delay calculations for a target scenario are made based on existing measurements of a start scenario.
The average delay caused by one network element (router + outgoing link) in the target scenario $E[D_t]$ is calculated for the different QoS Technologies BE, DiffServ (Olympic) and DiffServ (Default) (the priority formulas for G/G/1 are derived from the M/G/1 formulas). BE is the service of our start situation, this means the target scenario parameters are identical to the start scenario parameters because nothing is changed. For the other QoS Topologies, DiffServ (Olympic and Default), the residual service times (RST) are different for the target scenario end-to-end delay calculation, but the calculation is independent of the target scenario parameter. The meaning of the parameters can be seen in Appendix B.

$$\begin{aligned} E[D_t]_{be} &= E[W_t] + E[S_t] + D_{prop,t} \\ &= \frac{\rho_t}{1-\rho_t} \cdot E[S_t] \cdot \frac{C_{A,t}^2 + C_{S,t}^2}{2} + E[S_t] + D_{prop,t} \end{aligned} \tag{5.5}$$

$$\begin{aligned} E[RST_t]_{be} &= E[S_s] \cdot \frac{C_{A,s}^2 + C_{S,s}^2}{2} = \frac{1-\rho_s}{\rho_s} \cdot (E[D_s]_{be} - D_{prop,s} - E[S_s] \\ E[RST_t]_{olympic} &= \rho_{s,ef} \cdot \frac{1-\rho_{s,tcp}}{\rho_{s,tcp}} \cdot (E[D_{s,tcp}]_{be} - D_{prop,s} - E[S_{s,tcp}]) \\ &\quad + (\rho_{s,be} + \rho_{s,qbss}) \cdot \frac{1-\rho_{s,ef}}{\rho_{s,ef}} \cdot (E[D_{s,ef}]_{be} - D_{prop,s} - E[S_{s,ef}]) \\ E[RST_t]_{default} &= (\rho_{s,ef} \cdot w_{ef}) \cdot \frac{1-\rho_{s,tcp}}{\rho_{s,tcp}} \cdot (E[D_{s,tcp}]_{be} - D_{prop,s} - E[S_{s,tcp}]) \\ &\quad + ((\rho_{s,af1} \cdot w_{af1}) + (\rho_{s,af2} \cdot w_{af2}) + (\rho_{s,af3} \cdot w_{af3})) \\ &\quad \cdot \frac{1-\rho_{s,ef}}{\rho_{s,ef}} \cdot (E[D_{s,ef}]_{be} - D_{prop,s} - E[S_{s,ef}]) \end{aligned} \tag{5.6}$$

Then the target scenario specific parameters can be replaced by varied start scenario parameters in the formula for the QoS Technology that is investigated. The variation depends on the kind of extrapolation from the start to the target scenario. We distinguish a different link capacity (see 5.7) and a changed traffic intensity (see 5.8). The target parameters are replaced in the following way by the varied start parameters:

$$\begin{aligned} E[S_t] &= \frac{1}{k} \cdot E[S_s] \\ \rho_t &= \frac{1}{k} \cdot \rho_s \\ C^2_{S,t} &= C^2_{S,s} \\ C^2_{A,t} &= C^2_{A,s} \end{aligned} \tag{5.7}$$

$$\begin{aligned} E[S_t] &= E[S_s] \\ \rho_t &= \frac{1}{n} \cdot \rho_s \\ C^2_{S,t} &= C^2_{S,s} \\ C^2_{A,t} &\neq C^2_{A,s} \end{aligned} \tag{5.8}$$

For a changing link capacity we show the derivation of the Best Effort target scenario average end-to-end calculation in detail. It results in a formula that is independent of the target scenario parameters. The link capacity is changed by the factor $k$.

$$\begin{aligned} E[D_t]_{be} &= E[W_t] + E[S_t] + D_{prop,t} \\ &= \frac{\rho_t}{1-\rho_t} \cdot E[S_t] \cdot \frac{C^2_{A,t} + C^2_{S,t}}{2} + E[S_t] + D_{prop,t} \\ &= \frac{\frac{\rho_s}{k}}{1-\frac{\rho_s}{k}} \cdot \frac{1}{k} \cdot E[S_s] \cdot \frac{C^2_{A,s} + C^2_{S,s}}{2} + \frac{1}{k} \cdot E[S_s] + D_{prop,t} \\ &= \frac{\rho_s}{k-\rho_s} \cdot \frac{1}{k} \cdot E[S_s] \cdot \frac{C^2_{A,s} + C^2_{S,s}}{2} + \frac{1}{k} \cdot E[S_s] + D_{prop,t} \\ &= \frac{\frac{\rho_s}{k-\rho_s}}{\frac{\rho_s}{1-\rho_s}} \cdot \frac{1}{k} \cdot \frac{\rho_s}{1-\rho_s} \cdot E[S_s] \cdot \frac{C^2_{A,s} + C^2_{S,s}}{2} + \frac{1}{k} \cdot E[S_s] + D_{prop,t} \end{aligned} \tag{5.9}$$

Because of the following equation the calculation can be done based on the simulation results of the start scenario.

$$\frac{\rho_s}{1-\rho_s} \cdot E[S_s] \cdot \frac{C^2_{A,s} + C^2_{S,s}}{2} = E[D_s]_{be} - E[S_s] - D_{prop,s} \tag{5.10}$$

The results of the average delay calculation are nearly identical to the simulation results if the target scenario has new link capacities. These results are independent of the traffic mix. This extrapolation case can be used to calculate or test an Overprovisioning factor. If the traffic intensity is changed (mostly increased) the results deviate from the simulation results if video conferences are investigated. The percentage of the deviation of the 90 % confidence intervals of the simulation results increases up to 55 %. The results of CBR traffic only deviates about 1%. The reason for the deviation difference lies in the different coefficient of variation of the interarrival time values.

## 5.2 G-WiN

Because a major task of this project is the extrapolation of results gained in the ETS on the G-WiN and the extrapolation from a G-WiN start point on a different G-WiN target point, the G-WiN is described in general also as its topology, its load situation, the investigated scenario and the extrapolation. The G-WiN was described in Milestone1 [Schmitt et al. 02] , but the topology changed significantly which is the reason for a new description:

The G-WiN is, at the moment, one of the largest science networks of the world with respect to the number of access points and throughput and, for example in November 2003, an estimated number of users of 2 million at 600 access points and an exported volume of data of about 1.2 PBytes/month. The G-WiN consists of the core network with 27 routers (as of 31.11.2003, see Figure 5.17, [Adler 02])). The core network is extended with 79 access routers connected to access lines by interconnect bandwidths between 128 Kbit/s and 622 Mbit/s. Accordingly routers of type Cisco 7507 up to 155 Mbit/s and of type Cisco 12008 for 155 Mbit/s are deployed.

Further connections to other telecommunication networks exist:

- A 2.5 Gbit/s line to the European GÉANT Network
- From the GÉANT Network two 2.5 Gbit/s lines to North Amerika
- Two 2.5 Gbit/s lines into commercial networks (provider: Global Crossing / TELIA)

As mentioned before, the core network consists of 27 routers. According to their capacities these routers are divided into 10 so called level-1 and 17 level-2 routers. The level-1 routers are Cisco 12016 routers, the level-2 routers are Cisco 12008 routers (more information about the router parameters can be found in [Cisco Systems 12008, Cisco Systems 12016]). A table of router places, levels and types can be found in Appendix A.
A typical transmission technology for powerful networks is IP over SDH/WDM. SDH (Synchronous Digital Hierarchy) splits an optical channel into single channels of a fixed bandwidth. Additionally WDM (Wavelength Division Multiplexing) is used to realize several optical channels on one fiber. That way bandwidths from 622 Mbit/s up to 10 Gbit/s are available in the core network (as of 31.11.2003).

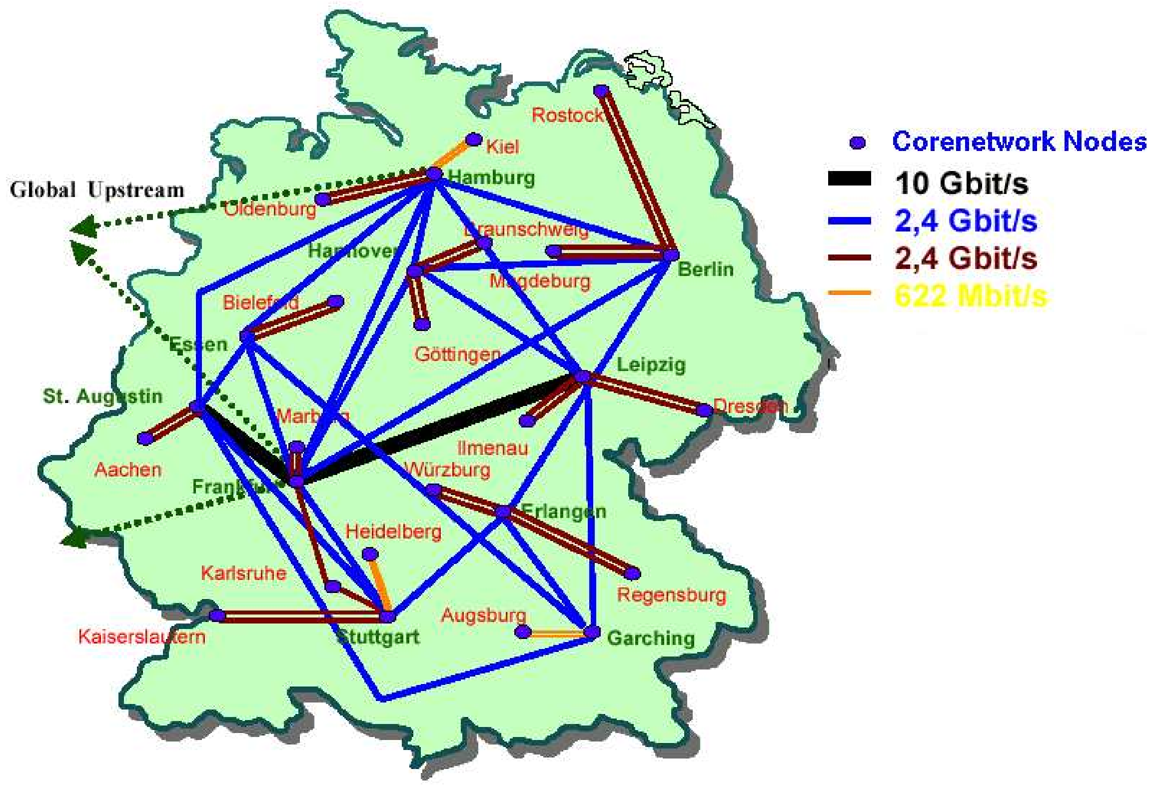



**Figure 5.17**: G-WiN Core Network Topology (: 31.11.2003)

As a summary we conclude a total number of 59 connections splitted into the following numbers with respect to the type of connections (see Figure 5.17):

Connections between level-1 routers:

- 20 x 2.4 Gbit/s
- 2 x 10 Gbit/s

Connections between level-1 and level-2 routers:

- 27 x 2.4 Gbit/s
- 6 x 622 Mbit/s

Each of the 17 level-2 routers is connected with two lines to its only one level-1 router (see table A.3 and A.4), making up a total of 34 level-1 to level-2 router connections.

### 5.2.1 G-WiN Load Situation

In the following we look at the load situation of the G-WiN to decide based on this situation which QoS technology can be used in the G-WiN. We distinguish between the core network and the access links.

#### 5.2.1.1 Core Network Utilization

In the core network the average and maximum utilization of every link is measured over one month. The measurements are executed by the CNM (Customer Network Management) System of the DFN. The measurement interval is 15 minutes. Shorter intervals are not possible because of an arising amount of router workload.
The tables A.2, A.3 und A.4 in Appendix A show that for no link the average utilization exceeds the 10 % mark. In most cases the utilization is much lower. The highest utilizations are found on links to the level-1 router Frankfurt, because it is the connection to the GÉANT network and a commercial network. Additionally, delay measurements are made by the DFN that confirm the utilization measurements. On no link in the core network the delay exceeds 10 ms. On single links in periods of only one day utilizations up to 40 % were found, but these have their reason in single conferences e.g., and represent no typical network situation. Again, most values are much lower. Figure 5.18 shows an example of a core network link with utilizations measured in 15 minute intervals in both directions.
The routers delay that can influence the end-to-end delay can be neglected because of low utilization. If the routers would be the bottlenecks of the network an update of linecards would improve the situation and the transmission of the packets.

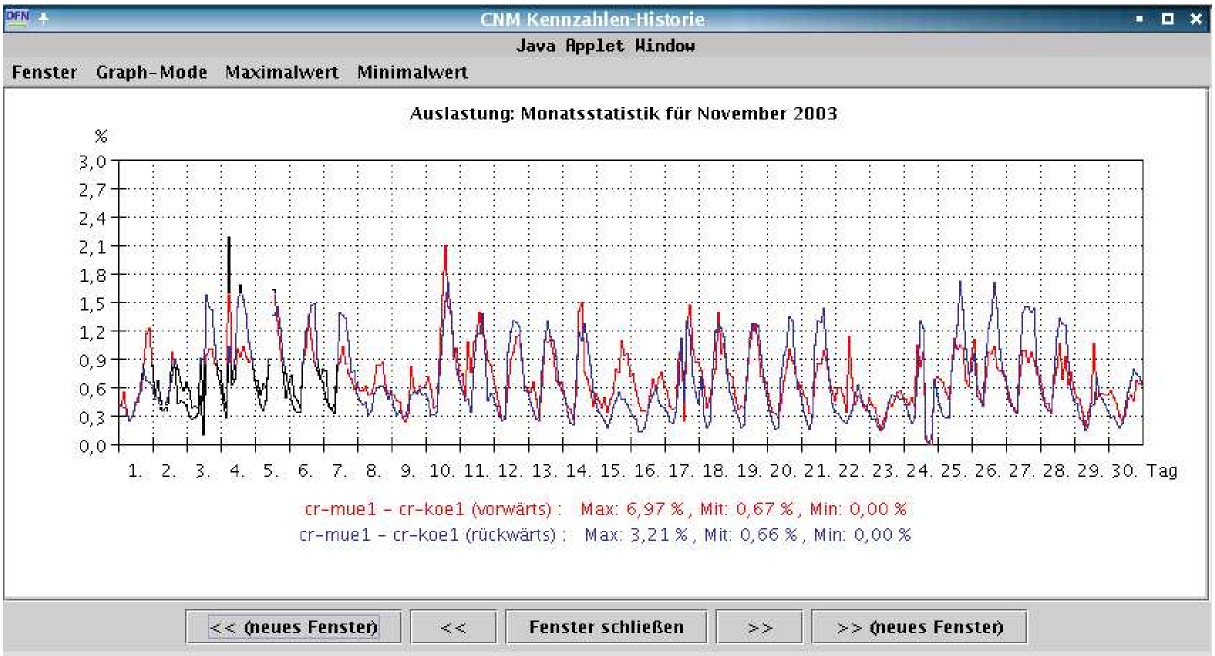



**Figure 5.18**: G-WiN core network utilization 15 minute interval - average over one month (as of 31.11.2003)

#### 5.2.1.2 Access Link Utilization

The utilization of access links is only demonstrated on an example access link with a high utilization, because only the existence of a highly utilized link and the utilization level is of interest. One more reason is that more access link specific information can not be offered because this information is confidental. Figures 5.19 and 5.20 show the utilization of an access link to a core network router in both directions measured over one day (Figure 5.19) and one week

(Figure 5.20). The average and maximum utilizations were measured in 5 minute intervals. The measurements show that a utilization over 90 % is possible.

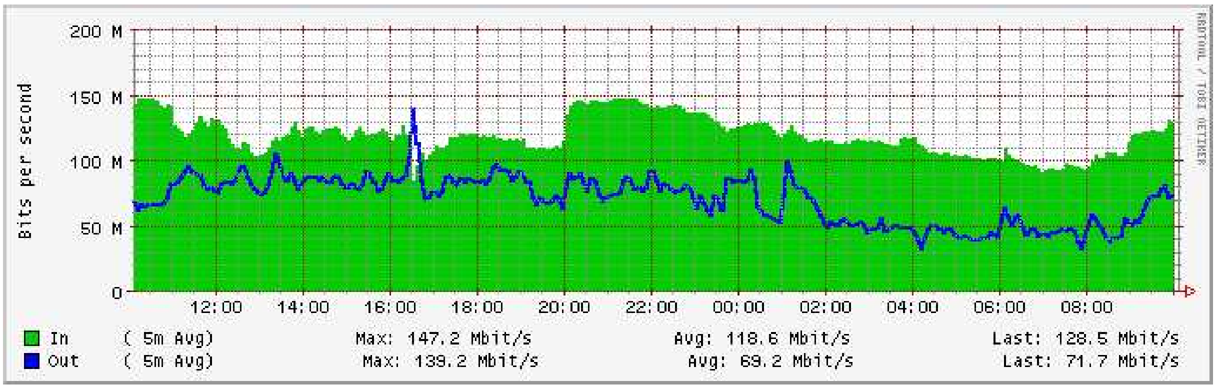



**Figure 5.19**: Throughput of a 155 Mbit/s - G-WiN access link 5 minute interval - average over one day (as of 31.11.2003)

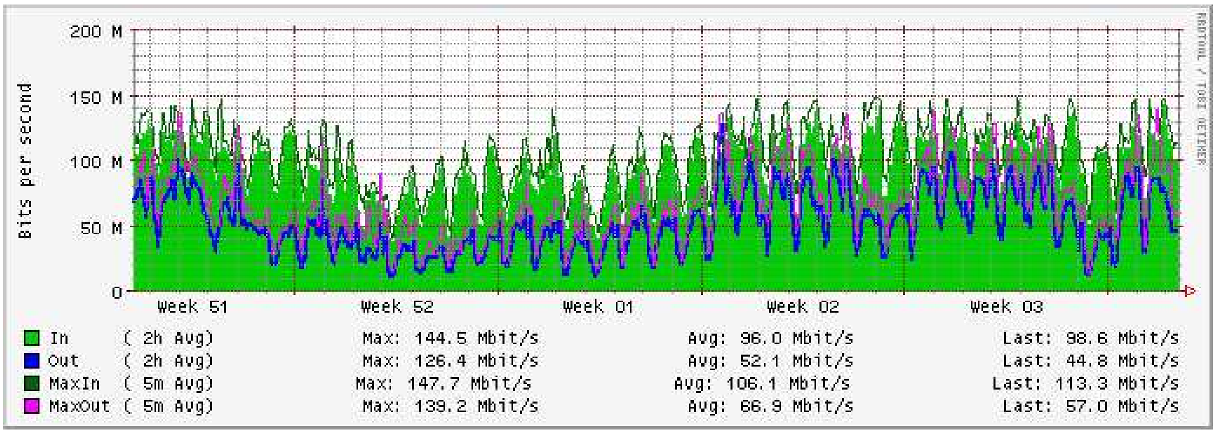



**Figure 5.20**: Throughput of a 155 Mbit/s - G-WiN access link 15 minute interval - average over one month (as of 31.11.2003)

### 5.2.2 Traffic Mix

Information about the typical G-WiN traffic mix in the core network and on the access links could not be collected because the data are confidential.

### 5.2.3 G-WiN Extrapolationscenario

The former described load situation leads to a prefered QoS technology approach, a composition of DiffServ on the access links and Overprovisioning in the core network. A reason for Overprovisioning in the core network is the very low link and router utilization level. DiffServ is the prefered technology on the access links because it can not only be used end-to-end and the access links need a technology that improves the transmission of QoS sensitive packets to meet the desired QoS. Additionally DiffServ minimally increases the core network edge routers load,

because the edge routers only forward the incoming packets according to the DSCP. The core network routers are tunneling the packets. If the utilization on a transmission path is low, no admission control or bandwidth broker is needed [Hoffmann]; if it is higher a bandwidth broker has to be integrated (see [Hoffmann, Heckmann et al. 04]). Then the DiffServ classes of the access links have to be uniform or converted on the transmission path. At the moment, we prefer a simple DiffServ approach with only two or three classes. The PHBs of these classes are uniform in the whole G-WiN. IntServ is not suitable for the G-WiN because of the high expenditure of implementation and signalling that is a result of the expansion of the G-WiN. Additionally, IntServ has to be integrated end-to-end in the whole G-WiN.
The investigation of the G-WiN core network and all access links is not possible in the simulator. The traffic load that is needed to utilize the 10 Gbit/s links is too high. Also the memory that is necessary to evaluate the simulations exceeds our capacities. Therefore we only investigate an example path through the G-WiN with a maximum number of hops in the core network and highly utilized access links (see Figure 5.21).

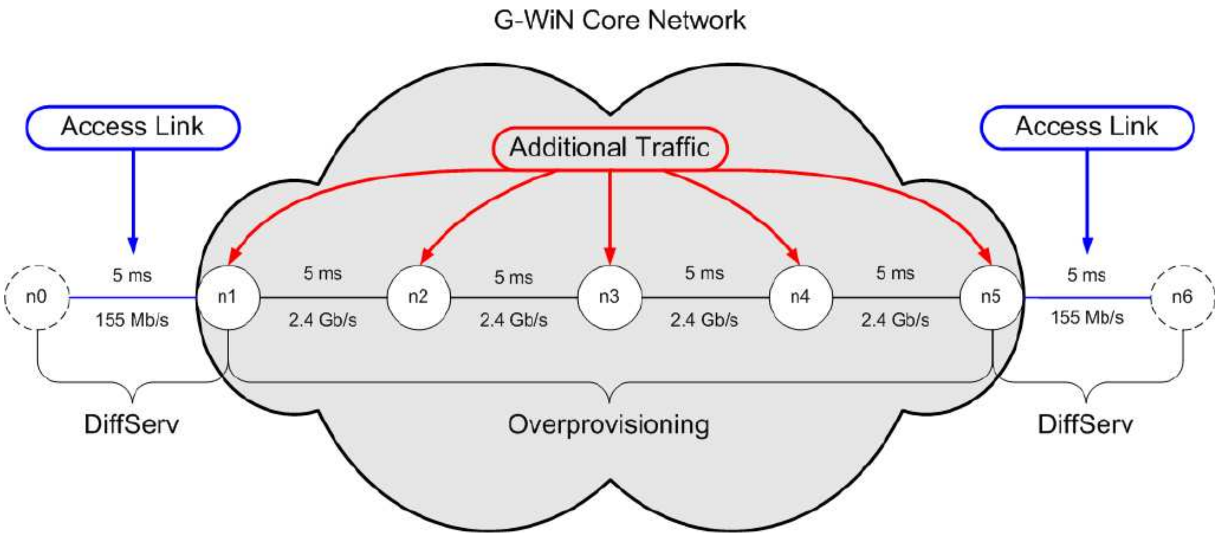



**Figure 5.21**: G-WiN Path

The following start scenario is used for the next investigations:

**Network Configuration** ($NK$):

- topology $T = \{ID, N, E\}$, with $ID = ETS$, $N = \{n_0, n_1, n_2, n_3, n_4, n_5, n_6, n_7\}$ nodes and $E = \{e_{0,1}, e_{1,2}, e_{2,3}, e_{3,4}, e_{4,5}, e_{5,6}, e_{6,7}\}$ edges
- set of bandwidths $B = \{B_{0,1}, B_{1,2}, B_{2,3}, B_{3,4}, B_{4,5}, B_{5,6}, B_{6,7}\}$,
  with $B_{0,1} = 155Mb/s$, $B_{1,2} = B_{2,3} = B_{3,4} = B_{4,5} = 2.4Mb/s$, $B_{5,6} = 1.0Gb/s$ and $B_{6,7} = 34.0Mb/s$
- set of link delays $LV = \{LV_{0,1}, LV_{1,2}, LV_{2,3}, LV_{3,4}, LV_{4,5}, LV_{5,6}, LV_{6,7}\}$,
  mit $LV_{0,1} = LV_{1,2} = LV_{2,3} = LV_{3,4} = LV_{4,5} = LV_{5,6} = LV_{6,7} = 1.5ms$
- routing information $R = \{\text{IP routing}\}$

**Load Spectrum** ($LS$) with a traffic mix of video and TCP traffic:
$LS = \{VLS_{VIDEO}, VLS_{WWW}\}$
$VLS_{VIDEO} = (SD_{VIDEO}, TT_{VIDEO})$
$SD_{VIDEO} = \{(n_0, n_7)\}$
$TT_{VIDEO} =$ {H.323 videoconference, generated from trace files}
$VLS_{WWW} = (SD_{WWW}, TT_{WWW})$
$SD_{WWW} = \{(n_0, n_7)\}$
$TT_{WWW} =$ {background traffic,WWW generator}

### 5.2.4 Sensitivity Analysis

The application with the highest QoS requirements is videoconferencing. Therefore and because of the interest of the DFN in videoconferencing we investigate video traffic with TCP background traffic as example. The combination approach of Overprovisioning and DiffServ (Olympic) is compared to the Best Effort service used at the moment in the G-WiN.

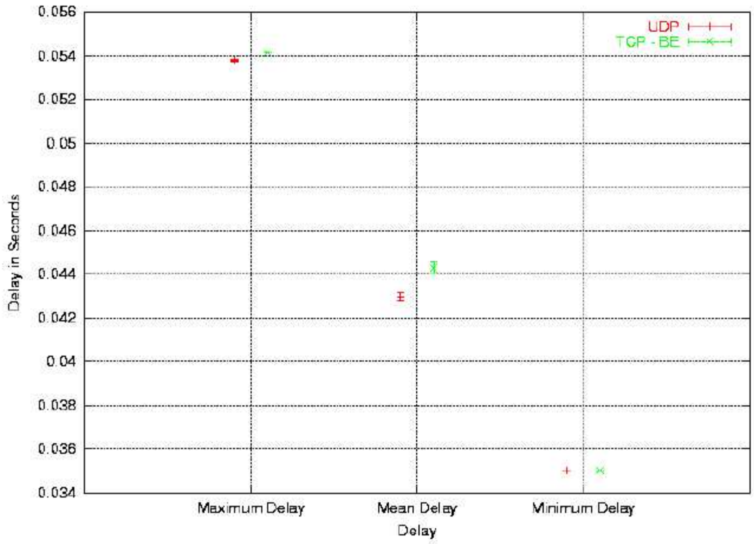
**Traffic Mix:**
20 video traces
49 TCP flows

This traffic mix rebuilds the utilization conditions in the G-WiN (at the moment) that are described above.

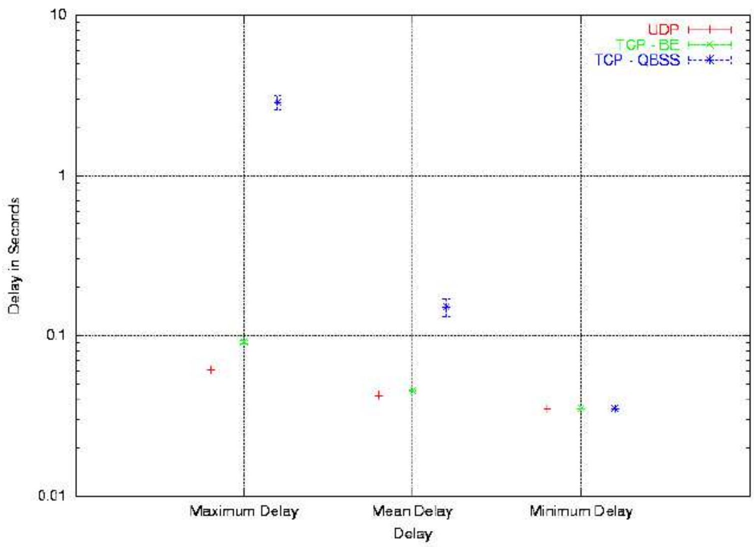
- Simulations with Best Effort:

- Simulations with combination approach:
  As **QoS Technology** DiffServ (Olympic) is used on the access links and Overprovisioning in the core network:
  $QT = \{EQT_{DiffServQ}\}$
  $EQT_{DiffServQ} = \{NI, EM\}$
  $NI$ = {($n_0$, (FIFO, (10, 40, 50)), PQ), ($n_1$, (FIFO, (10, 40, 50)), PQ), ($n_5$, (FIFO, (10, 40, 50)), PQ), ($n_6$, (FIFO, (10, 40, 50)), PQ), ($n_7$, (FIFO, (10, 40, 50)), PQ))}
  $EM$ =(packet, per hop / aggregate / deterministic / loss / delay, n classes, complex, router, edge, proactive, yes, good, network)

  **QoS Spectrum** ($QS$)**:**
  $QS = \{VQS_{VIDEO}\}$
  $VQS_{VIDEO} =$ (0.1 %, 100 ms, 50 ms, -)

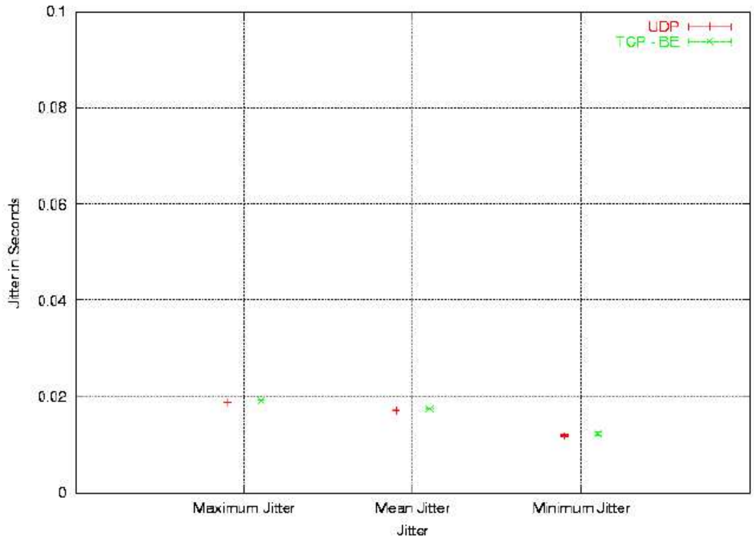



(a) End-to-End delay by 10 video traces and 49 TCP flows using Best Effort

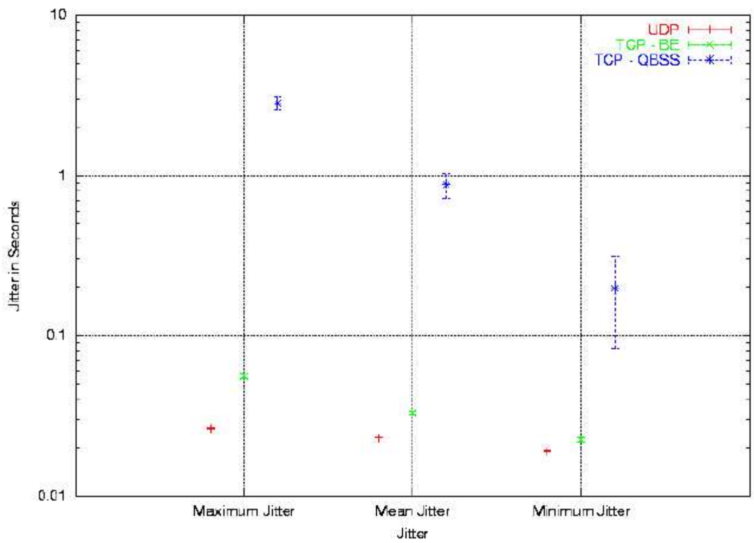



(b) End-to-End delay by 10 video traces and 49 TCP flows using DiffServ (Olympic)

**Figure 5.22**: End-to-End delay by 10 video traces and 49 TCP flows

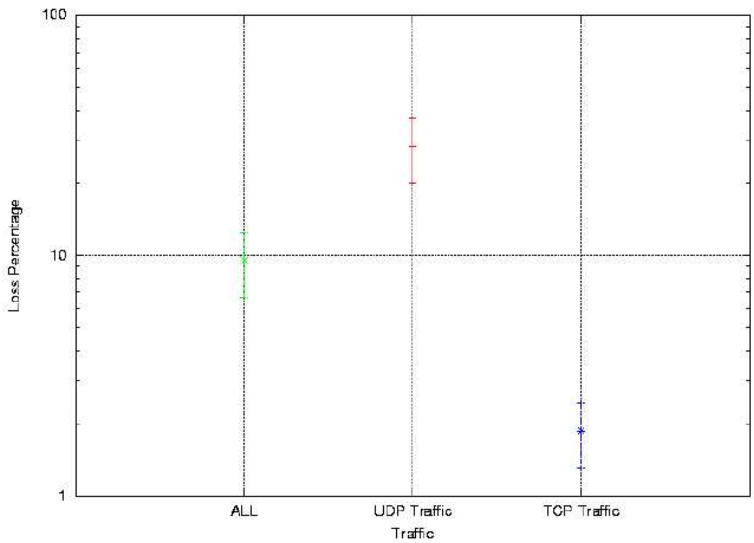



(a) Jitter caused by 10 video traces and 49 TCP flows using Best Effort

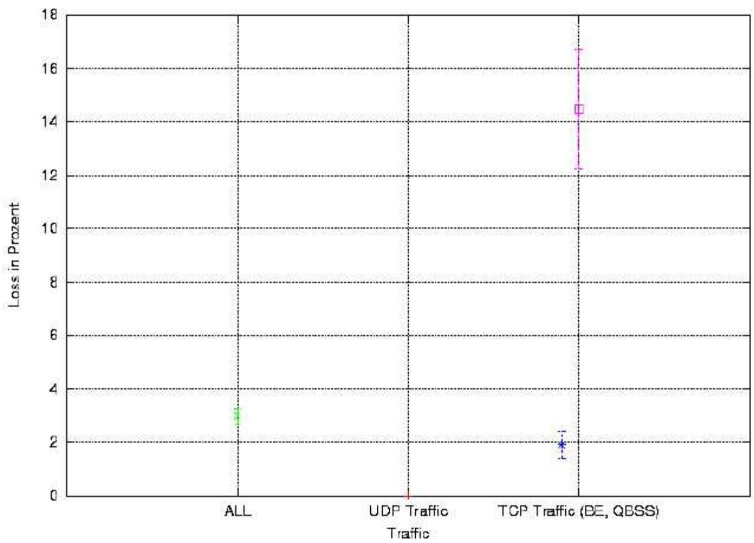



(b) Jitter caused by 10 video traces and 49 TCP flows using DiffServ (Olympic)

**Figure 5.23**: Jitter caused by 10 video traces and 49 TCP flows




(a) Loss caused by 10 video traces and 49 TCP flows using Best Effort




(b) Loss caused by 10 video traces and 49 TCP flows using DiffServ (Olympic)

**Figure 5.24**: Loss caused by 10 video traces and 49 TCP flows

### 5.2.5 Worst Case Analysis

| Delay Type | Delay [s] | Traffic Type |
|---|---|---|
| average Delay at 10 Video Traces, 49 TCP Flow | 0.042332 | Video Trace |
| maximum Delay at 10 Video Traces, 49 TCP Flow | 0.061226 | Video Trace |
| calculated Best Case Delay | 0.035241 | Video Trace |
| calculated Worst Case Delay | 0.290576 | Video Trace |

**Table 5.8**: Comparison of calculated and simulated delay values

| Jitter Type | Jitter [s] | Traffic Type |
|---|---|---|
| average Jitter at 10 Video Traces, 49 TCP Flow | 0.026203 | Video Trace |
| maximum Jitter at 10 Video Traces, 49 TCP Flow | 0.022789 | Video Trace |
| calculated Worst Case Jitterwert | 0.255749 | Video Trace |

**Table 5.9**: Comparison of calculated and simulated jitter values

The worst case calculation results are misleading because of the low utilization in the core network.

### 5.2.6 Average Delay Calculation

The average delay calculation can not be used in this example for an acceptable calculation because the traffic intensity is changed and not the link capacities. It only could be used to test Overprovisioning results. More detailed information about the average delay calculation can be found in [Hoffmann].

### 5.2.7 Results

The simulation results show that the used QoS Technology approach reduces the very high loss rate of video traffic of the BE simulations to a value of 0. The maximum delay and jitter is not reduced for video traffic but this was not necessary. The delay, jitter and loss of the BE class are within aceptable ranges. The delay, jitter and loss of the QBSS class is very high. High values for delay and jitter are no problem for QBSS traffic but the high value for loss. Therefore the queue length of the QBSS class has to be increased or an admission control has to be implemented for QBSS traffic. But the main task of offering an adequate QoS to QoS sensitive applications is possible to manage with this approach in the G-WiN. In contrast to BE the QoS values for videoconferencing are very good. The single parameters for optimizing TCP traffic have to be validated in reality depending on the application demands.
The worst case calculation is in the case of the G-WiN a bad approximation because of the low core network utilization.

# Chapter 6

# Conclusion and Open Problems

The project has demonstrated that quality of service insight that has been gained in the testbed or in the simulator can be extrapolated to a scenario with different configuration, load and quality of service mechanisms. For the extrapolation, preferable the simulator is used; for Overprovisioning and Differentiated services / Expedited Forwarding, also G/G/1 models (Krämer / Whitt formulas) can be used for approximate results and simple load and simple configuration, as a path through G-WiN e.g.. The extrapolation technique and its tools are described in detail, validated and demonstrated in examples.
Looking back to the experiences of the project, it would have been promoted by a more intense exchange of concepts and results between the parallel QoS projects and a closer cooperation with the G-WiN-Laboratory and the Network Operation Center, a visit of which, e.g., could not be realized.

At the end of the project it is worthwhile to envisage further challenges in the QoS area, some of which could be attacked on the basis of the LETS QoS Project:
In future networks, burst and path switching will be the prefered technique for many communication tasks, as large file transfers or media transport. The paths can be of defined quality, and for most types of usage short set-up times will be critical and difficult to achieve.
Some networks (commercial networks as well as research and engineering networks) provide QoS technology, the GÉANT network e.g.. Their operational results may be compared with LETS QoS forecasts.
Finally, the realization of service quality by means of QoS mechanisms is a design option the evaluation of which should be integrated into the topology design, as, e.g., done by the DFN project at the Konrad-Zuse-Zentrum für Informationstechnik in Berlin.

# Chapter 7
# Documentation

- Project Proposal:
  - [Steinmetz et al. 01]
- Project Milestones:
  - [Schmitt et al. 02]
  - [Heckmann et al. 03-1]
  - [Pandit et al. 03-2]

# Appendix A

# G-WiN Core Network Topology

| Place | Level | Cisco Router Type |
|---|---|---|
| Berlin | 1 | 12016 |
| Erlangen | 1 | 12016 |
| Essen | 1 | 12016 |
| Frankfurt | 1 | 12016 |
| Hamburg | 1 | 12016 |
| Hannover | 1 | 12016 |
| Köln (St. Augustine) | 1 | 12016 |
| Leipzig | 1 | 12016 |
| München (Garching) | 1 | 12016 |
| Stuttgart | 1 | 12016 |
| Aachen | 2 | 12008 |
| Augsburg | 2 | 12008 |
| Bielefeld | 2 | 12008 |
| Braunschweig | 2 | 12008 |
| Dresden | 2 | 12008 |
| Göttingen | 2 | 12008 |
| Heidelberg | 2 | 12008 |
| Illmenau | 2 | 12008 |
| Kaiserslautern | 2 | 12008 |
| Karlsruhe | 2 | 12008 |
| Kiel | 2 | 12008 |
| Magdeburg | 2 | 12008 |
| Marburg | 2 | 12008 |
| Oldenburg | 2 | 12008 |
| Regensburg | 2 | 12008 |
| Rostock | 2 | 12008 |
| Würzburg | 2 | 12008 |

**Table A.1**: Place, Level, und Routertype

| **Connection** | **Capacity** | **Average Utilization bidirectional** | **maximum Utilization bidirectional** |
|---|---|---|---|
| Berlin - Erlangen | 2.4 Gbit/s | 2.25 % / 0.96 % | 10.92 % / 10.51 % |
| Berlin - Frankfurt | 2.4 Gbit/s | 5.37 % / 7.67 % | 13.11 % / 24.65 % |
| Berlin - Hannover | 2.4 Gbit/s | 3.44 % / 1.59 % | 12.62 % / 21.08 % |
| Berlin - Hamburg | 2.4 Gbit/s | 4.47 % / 2.52 % | 8.44 % / 8.19 % |
| Erlangen - München | 2.4 Gbit/s | 0.97 % / 2.19 % | 18.28 % / 20.12 % |
| Erlangen - Stuttgart | 2.4 Gbit/s | 5.45 % / 3.72 % | 29.14 % / 30.49 % |
| Essen - Frankfurt | 2.4 Gbit/s | 6.51 % / 8.53 % | 12.17 % / 29.45 % |
| Essen - Hamburg | 2.4 Gbit/s | 5.58 % / 2.32 % | 10.17 % / 5.67 % |
| Essen - München | 2.4 Gbit/s | 0.51 % / 2.30 % | 8.22 % / 10.22 % |
| Frankfurt - Hamburg | 2.4 Gbit/s | 6.11 % / 4.96 % | 15.88 % / 10.94 % |
| Frankfurt - Hannover | 2.4 Gbit/s | 5.74 % / 7.58 % | 26.88 % / 16.00 % |
| Frankfurt - Köln | 2.4 Gbit/s | 6.25 % / 7.35 % | 14.42 % / 28.21 % |
| Frankfurt - Leipzig | 10 Gbit/s | 3.22 % / 4.13 % | 8.39 % / 7.72 % |
| Frankfurt - Stuttgart | 2.4 Gbit/s | 9.15 % / 7.29 % | 19.47 % / 24.94 % |
| Hamburg - Hannover | 2.4 Gbit/s | 4.62 % / 1.89 % | 9.33 % / 7.77 % |
| Hamburg - Leipzig | 2.4 Gbit/s | 1.65 % / 1.30 % | 6.11 % / 5.09 % |
| Hamburg - Köln | 2.4 Gbit/s | 2.74 % / 2.18 % | 6.63 % / 12.67 % |
| Hannover - Leipzig | 2.4 Gbit/s | 1.26 % / 3.15 % | 5.46 % / 7.68 % |
| Köln - München | 2.4 Gbit/s | 0.66 % / 0.67 % | 3.21 % / 6.97 % |
| Köln - Stuttgart | 2.4 Gbit/s | 4.74 % / 1.02 % | 10.96 % / 8.56 % |
| Leipzig - München | 2.4 Gbit/s | 4.44 % / 8.20 % | 11.65 % / 19.25 % |

**Table A.2**: Level1 - Level1 connections in the G-WiN core network with capacities, average and maximum utilizations measured in 15 minute intervalls over November 2003

| **Connection** | **Capacity** | **Average Utilization bidirectional** | **maximum Utilization bidirectional** |
|---|---|---|---|
| Aachen - Köln | 2.4 Gbit/s | 3.72 % / 1.92 % | 24.45 % / 6.06 % |
| Augsburg - München | 622 Mbit/s | 2.28 % / 1.29 % | 10.79 % / 6.63 % |
| | 622 Mbit/s | 2.14 % / 1.23 % | 9.87 % / 5.84 % |
| Berlin - Magdeburg | 2.4 Gbit/s | 0.94 % / 0.84 % | 2.35 % / 2.31 % |
| | 2.4 Gbit/s | 0.93 % / 0.86 % | 4.51 % / 2.47 % |
| Berlin - Rostock | 2.4 Gbit/s | 1.62 % / 2.18 % | 4.46 % / 3.86 % |
| | 2.4 Gbit/s | 1.73 % / 2.16 % | 6.50 % / 5.87 % |
| Bielefeld - Essen | 2.4 Gbit/s | 3.77 % / 2.80 % | 7.56 % / 24.10 % |
| | 2.4 Gbit/s | 3.97 % / 2.47 % | 7.64 % / 6.23 % |
| Braunschweig - Hannover | 2.4 Gbit/s | 1.87 % / 1.20 % | 3.57 % / 3.57 % |
| | 2.4 Gbit/s | 1.90 % / 1.06 % | 3.75 % / 3.08 % |
| Dresden - Leipzig | 2.4 Gbit/s | 5.65 % / 2.87 % | 15.32 % / 10.25 % |
| | 2.4 Gbit/s | 5.58 % / 2.77 % | 10.40 % / 6.65 % |
| Erlangen - Regensburg | 2.4 Gbit/s | 0.77 % / 0.42 % | 3.08 % / 1.75 % |
| | 2.4 Gbit/s | 0.77 % / 0.42 % | 3.00 % / 1.36 % |
| Erlangen - Würzburg | 2.4 Gbit/s | 0.33 % / 0.25 % | 2.29 % / 1.78 % |
| | 2.4 Gbit/s | 0.35 % / 0.23 % | 2.20 % / 1.78 % |
| Frankfurt - Karlsruhe | 2.4 Gbit/s | 1.11 % / 0.34 % | 40.44 % / 29.78 % |
| Frankfurt - Marburg | 2.4 Gbit/s | 0.79 % / 0.56 % | 12.64 % / 6.20 % |
| | 2.4 Gbit/s | 0.74 % / 0.57 % | 2.64 % / 1.85 % |
| Göttingen - Hannover | 2.4 Gbit/s | 0.00 % / 0.00 % | 1.07 % / 0.73 % |
| | 2.4 Gbit/s | 5.31 % / 3.04 % | 7.24 % / 6.93 % |
| Hamburg - Kiel | 622 Mbit/s | 1.64 % / 1.06 % | 11.33 % / 5.12 % |
| | 622 Mbit/s | 1.56 % / 1.25 % | 9.96 % / 6.39 % |
| Hamburg - Oldenburg | 2.4 Gbit/s | 2.23 % / 3.76 % | 5.22 % / 5.58 % |
| | 2.4 Gbit/s | 2.24 % / 3.80 % | 5.23 % / 6.07 % |

**Table A.3**: Level1 - Level2 connections in the G-WiN core network with capacities, average and maximum utilizations measured in 15 minute intervalls over November 2003

| **Connection** | **Capacity** | **Average Utilization bidirectional** | **maximum Utilization bidirectional** |
|---|---|---|---|
| Heidelberg - Stuttgart | 622 Mbit/s | 0.49 % / 0.80 % | 4.00 % / 5.25 % |
| | 622 Mbit/s | 0.49 % / 0.61 % | 2.84 % / 4.56 % |
| Illmenau - Leipzig | 2.4 Gbit/s | 2.79 % / 1.74 % | 6.19 % / 5.50 % |
| | 2.4 Gbit/s | 2.61 % / 1.78 % | 5.14 % / 6.02 % |
| Kaiserslautern - Stuttgart | 2.4 Gbit/s | 1.24 % / 0.62 % | 2.04 % / 1.73 % |
| | 2.4 Gbit/s | 1.96 % / 0.85 % | 3.88 % / 2.76 % |
| Karlsruhe - Stuttgart | 2.4 Gbit/s | 0.11 % / 0.08 % | 40.37 % / 31.97 % |

**Table A.4**: Level1 - Level2 connections in the G-WiN core network with capacities, average and maximum utilizations measured in 15 minute intervalls over November 2003

# Appendix B

# Average Delay Parameters

| Parameters | |
|---|---|
| $E[D_t]_{be}$ | end-to-end delay in the target scenario using BE |
| $E[W_t]$ | average waiting time in the target scenario |
| $E[S_t]$ | average service time in the target scenario |
| $D_{prop,t}$ | propagation delay in the target scenario |
| $\rho_t$ | utilization of the link in the target scenario |
| $C^2_{S,t}$ | coefficient of variation of the service time in the target scenario |
| $C^2_{A,t}$ | coefficient of variation of the interarrival time in the target scenario |
| $E[RST_t]_{be}$ | average residual service time using best effort service |
| $E[S_s]$ | average service time of the start scenario |
| $C^2_{A,s}$ | coefficient of variation of the service time in the start scenario |
| $C^2_{S,s}$ | coefficient of variation of the interarrival time in the start scenario |
| $\rho_s$ | utilization of the link in the start scenario |
| $E[D_s]_{be}$ | end-to-end delay in the start system using BE |
| $D_{prop,s}$ | propagation delay in the start scenario |
| $E[S_s]$ | average service time in the start scenario |

**Table B.1**: Average delay calculation parameter

| Parameters | |
|---|---|
| $E[RST_t]_{olympic}$ | average residual service time using Olympic service |
| $\rho_{s,ef}$ | utilization of the link in the start scenario caused by the EF class |
| $\rho_{s,tcp}$ | utilization of the link in the start scenario caused by the TCP classes (BE, QBSS) |
| $E[D_{s,tcp}]_{be}$ | end-to-end delay of TCP traffic (BE , QBSS) in the start system using BE |
| $E[S_{s,tcp}]$ | average service time of TCP traffic (BE , QBSS) in the start scenario |
| $\rho_{s,be}$ | utilization of the link in the start scenario caused by the BE class |
| $\rho_{s,qbss}$ | utilization of the link in the start scenario caused by the BE class |
| $E[D_{s,ef}]_{be}$ | end-to-end delay of EF traffic in the start system using BE |
| $E[S_{s,ef}]$ | average service time of EF traffic in the start scenario |
| $E[RST_t]_{default}$ | average residual service time using Default service |
| $w_{ef}$ | weight of the EF class in the WRR scheduling algorithm |
| $\rho_{s,af1}$ | utilization of the link in the start scenario caused by the AF1 class |
| $w_{af1}$ | weight of the AF1 class in the WRR scheduling algorithm |
| $\rho_{s,af2}$ | utilization of the link in the start scenario caused by the AF2 class |
| $w_{af2}$ | weight of the AF2 class in the WRR scheduling algorithm |
| $\rho_{s,af3}$ | utilization of the link in the start scenario caused by the AF3 class |
| $w_{af3}$ | weight of the AF3 class in the WRR scheduling algorithm |

**Table B.2**: Average delay calculation parameter

# List of Figures

# List of Tables

# References

[Ferguson & Huston 98] Paul Ferguson and Geoff Huston: *Quality of Service - Delivering QoS on the Internet and in Corporate Networks.* WILEY COMPUTER PUBLISHING, 1998.

[Wroclawski 97] J. Wroclawski: *The Use of RSVP with IETF Integrated Services.* Network Working Group, Category: Standards Track, RFC 2210, September 1997.

[Wroclawski 97] J. Wroclawski: *Specification of Controlled-Load Network Element Service.* Network Working Group, Category: Standards Track, RFC 2211, September 1997.

[Shenker et al. 97] S. Shenker, C. Partridge and R. Guerin: *Specification of Guaranteed Quality of Service.* Network Working Group, Category: Standards Track, RFC 2212, September 1997.

[Braden et al. 94] R. Braden, D. Clark and S. Shenker: *Integrated Services in the Internet Architecture: an Overview.* Network Working Group, Category: Informational, RFC 1633, June 1994.

[Blake et al. 98] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang and W. Weiss: *An Architecture for Differentiated Services.* Network Working Group, Category: Informational, RFC 2475, December 1998.

[Davie et al. 02] B. Davie et al.: *An Expedited Forwarding PHB.* Network Working Group, Category: Standards Track, RFC 3246, March 2002.

[Heinanen et al. 99] J. Heinanen et al: *Assured Forwarding PHB Group.* Network Working Group, Category: Standard Track, RFC 2597, June 1999.

[Burgstahler et al. 01] L. Burgstahler, J. Jaehnert, L. Mark, J. Tiemann and R. Roth: *QUASAR - Quality of Service Architekturen - Meilenstein 1: Technologieuebersicht.* DFN Projekt, Start 2001.

[Burgstahler et al. 02] L. Burgstahler et al.: *QUASAR - Quality of Service Architekturen - Meilenstein 2: Quasar QoS Reference Model.* DFN-Projekt, Start 2001.

[Burgstahler et al. 03] L. Burgstahler et al.: *QUASAR - Quality of Service Architekturen - Meilenstein 3: Preliminary Quasar QoS Architecture.* DFN-Projekt, Start 2001.

[Burgstahler et al. 04] L. Burgstahler et al.: *QUASAR - Quality of Service Architekturen - Meilenstein 4: First Test Results.* DFN-Projekt, Start 2001.

[Burgstahler et al. 05] L. Burgstahler et al.: *QUASAR - Quality of Service Architekturen - Meilenstein 5: QUASAR QoS-Architektur.* DFN-Projekt, Start 2001.

[Burgstahler et al. 06] L. Burgstahler et al.: *QUASAR - Quality of Service Architekturen - Meilenstein 6: Projektabschluss mit Abschlussbericht.*DFN-Projekt, Start 2001.

[Stardust.com, Inc. 99/1] Stardust.com, Inc.: *White Paper - The Need for QoS.* July 1999.

[Stardust.com, Inc. 99/2] Stardust.com, Inc.: *White Paper - QoS protocols and architectures.* July 1999.

[Stardust.com, Inc. 99/3] Stardust.com, Inc.: *White Paper - Introduction to QoS Policies.* July 1999.

[Baumgartner et al. 96] f. Baumgartner, T. Braun and Pascal Habegger: *Differentiated Services: A New Approach for Quality of Service in the Internet.* Chapman & Hall, 1996.

[Braden et al. 97] R. Braden, L. Zang, S. Berson, S. Herzog and S. Jamin: *Resource ReSerVation Protocol (RSVP) - Version 1 Functional Specification.* Network Working Group, Category: Standards Track, RFC2205, September 1997.

[van der Zee 99] M. van der Zee and G. Heijenk: *Quality of Service Routing - State of the Art Report.* ERICSSON, Open REPORT, August 1999.

[Teitelbaum 01] B. Teitelbaum: *Future Priorities for Internet2 Qo.S* October 2001.

[Kleinrock 75] L. Kleinrock: *Queueing Systems - Volume I: Theory.* Wiley-Interscience, 1975.

[Kleinrock 76] L. Kleinrock: *Queueing Systems - Volume 2: Computer Applications.* Wiley-Interscience, 1976.

[Tran-Gia 96] P. Tran-Gia: *Analytische Leistungsbewertung verteilter Systeme - Eine Einfuehrung.* Springer, 1996.

[Jaiswal 68] N. K. Jaiswal: *Priority Queues.* ACADEMIC PRESS, 1968.

[Jessen 01] E. Jessen: *Vorlesung Sommersemester 2001 - Rechnernetze II.* Lehrstuhl fuer Informatik, Technische Universitaet Muenchen, 2001.

[Jobmann 01] M. Jobmann: *Quantitative Modelle fuer Rechensysteme - Ausarbeitung der Folien zur Vorlesung WS 01/02.* Lehrstuhl fuer Informatik, Technische Universitaet Muenchen, 2001.

[Steinmetz 99] R. Steinmetz: *Multimedia-Technologie - Grundlagen, Komponenten und Systeme.* Springer, 1999.

[Greiner & Tinhofer 96] M. Greiner and G. Tinhofer: *Stochastik fuer Studienanfaenger der Informatik.* Carl Hanser, 1996.

[Schmitt 01] J. B. Schmitt: *Heterogeneous Network Quality of Service Systems.* Kluwer Academic Publisher, 2001.

[Davie et al. 02] B. Davie, A. Charny, F. Baker, J. Bennet, K. Benson, A. Chiu, S. Davari, C. Kalmanek, D. Stiliadis, J. Le Boudec, W. Courtney, V. Firoiu and K. Ramakrishnam: *An Expedited Forwarding PHB.* Network Working Group, Internet Draft, March 2002.

[Heinanen et al. 99] J. Heinanen, F. Baker, W. Weiss and J. Wroclawski: *Assured Forwarding PHB Group.* Network Working Group, Category: Standards Track, RFC2597, June 1999.

[Cisco Systems 12008] Cisco Systems, Inc.: *Product Overview.*
< URL : http : //www.cisco.com/univercd/cc/td/doc/product/core/cis12000/cis12008 /icg/product1.htm >, 2001

[Cisco Systems 12016] Cisco Systems, Inc.: *Product Overview.* < *URL* : *http* : *//www.cisco.com/univercd/cc/td/doc/product/core/cis*12000*/cis*12016 */icg/hfricgpo.htm* >, 2001

[Adler 02] H.-M. Adler: *Neues im G-WiN.* 36.Betriebstagung, 2002

[DFN 01] DFN: *Gigabit-Wissenschaftsnetz G-WiN.* < *URL* : *http* : *//www.dfn.de/win* >, 2001

[Jessen 02] E. Jessen: *Rechnernetze II.* Lehrstuhl für Rechnerkommunikation und maschinelle Deduktion, Institut der Informatik, TU-München, 2002

[Fall & Varadhan 02] K. Fall und K. Varadhan: *The ns Manual.* < *URL* : *http* : *//www.isi.edu/nsnam/ns/ns* − *documentation.html* >, 2002

[Greis 02] M. Greis: *Tutorial for the Network Simulator ns.* < *URL* : *http* : *//www.isi.edu/nsnam/ns/tutorial/index.html* >, 2002

[Welch 97] B. B. Welch: *Practical Programming in Tcl and Tk.* Prentice Hall Ptr, 1997

[Ferguson & Huston 98] P. Ferguson and G. Huston: *Quality of Service in the Internet: Fact, Fiction, or Compromise?* < *URL* : *http* : *//www.employees.org/ ferguson/inet_qos.htm* >, Cisco Systems, Inc., 1998

[Constable & Price 01] G. Constable and D. Price: *Options for Quality of Service Mechanisms, Network Quality of Service and Resource Issues, H.323 IP Videoconferencing demonstrator.* University of Wales, Aberystwyth, 3/2001

[Schmitt et al. 02] J. Schmitt et al.: *Network QoS An Up-to-Date Overview.* DFN LETSQoS Project, Milestone 1, 2002

[Varschney et al. 02] U. Varschney et al.: *VOICE OVER IP COMMUNICATIONS OF THE ACM.*, Vol. 45, 2002

[ITU 02] International Telecommunication Union: *List of ITU-T Recommendations.* < *URL* : *http* : *//www.itu.int/publibase/itu* − *t/ItutDetailsOnSubject.asp?serie* = *g*& *subjectid* = 2100 >, 2002

[Cisco Systems 02] Cisco Systems, Inc.: *Implementing QoS Solutions for H.323 Video Conferencing Over IP.* < *URL* : *http* : *//www.cisco.com/warp/public/*105*/video* − *qos.html* >, 2002

[Hu & Low] J. Hu and N. Low: *Investigation of Telephonic Voice Traffic Network Performance on Integrated Wireless LAN and Wired Internet.* Royal Melbourne Institute of Technology, Melbourne

[Cho 01] K. Cho: *The Design and Implementation of the AltQ Traffic Management System.* Doctoral Dissertation, Keio University, Jannuary 2001

[Karsten 03] M. Karsten: *KOM RSVP Engine .* < *URL* : *http* : *//www.kom.e* − *technik.tu* − *darmstadt.de/rsvp/* >, 2003

[Heckmann et al. 03-1] O. Heckmann et al.: *Scenario Generation.* DFN LETSQoS Project, Milestone 2, 2003

[Pandit et al. 03-2] O. Pandit et al.: *Experiment and Extrapolation Plan.* DFN LETSQoS Project, Milestone 3, 2003

[Zhang 95] H. Zhang: *Service Disciplines for Guaranteed Performance Service in Packet Switching Networks.* Proc. IEEE,S. 1373-1396, Oktober 1995

[Parekh & Galleger] A.K. Parekh and R.G. Galleger: *A generalized processor sharing approach to flow control in integrated services networks: The single node case.* IEEE/ACM Trans. on Networking, 2(2): 137-150, April 1994

[White & Crowcroft] P. White and J. Crowcroft: *The Integrated Services in the Internet: State of Art.* Department of Computer Science, University College London

[Glasmann et al.] J. Glasmann et al.: *Estimation of Token Bucket Parameters for Videoconferencing Systems in Corporate Networks.* Lehrstuhl für Kommunikationsnetze, Technische Universität München

[Tanenbaum 98] A.S. Tanenbaum: *Computernetzwerke.* Prentice Hall, 1998

[Allen 90] A.O. Allen: *Probability, Statistics, and Queueing Theory with Computer Science Applications.* Academic Press, Inc., 1990

[Stoyan 83] D. Stoyan: *Comparison Methods for Queues and Other Stochastic Models.* John Wiley & Sons: 1983

[Whitt 88] W. Whitt and S. Suresh: *Arranging Queues in Series.* 1988

[ITU 00] ITU-T:*Internet Protocol Communication Service - IP Performance and Availability Objectives and Allocations.* Draft ITU-T Recommendation Y.1541, November 2000

[GW 00] GW:*Perceived quantitative quality of applications* $< URL : http : www. >$

[Holleczek et al.] P. Holleczek et al.: *Bestimmung der Dienstqualität im G-WiN mittels IPPM.* Vortrag DFN-Mitgliederversammlung, Dezember 2003

[Lipsky et al.] L. Lipsky et al.: *Simulationstechniken und Module.* Bericht zum Ergänzungspaket 1 des Projektes - Pufferdimensionierung in ATM-Weitverkehrsnetzen mit selbstähnlichen Verkehrsströmen unter besonderer Berücksichtigung des transienten Verhaltens, Juni 1999

[Granges 00] E. A. Granges: *Bandwidth Broker for Differentiated Services.* Diplomarbeit, Forschungsgruppe Rechnernetze und Verteilte Systeme (RVS), Institut für Informatik und angewandte Mathematik, Universität Bern, 2000

[Nichols 99] K. Nichols et al.: *A Two-bit Differentiated Services Architecture for the Internet.* Network Working Group, Category: Standards Track, RFC 2638, July 1999

[Keshav 98] S. Keshav: *An Engineering Approach to Computer Networking. ATM Networks, the Internet, and the Telephone Network.* Addison-Wesley, 1998

[Steinmetz et al. 01] R. Steinmetz et al.: *DFN-Projektvorschlag - LETS QoS.* TU Darmstadt, TU München, 2001

[Glasmann 03] J. Glasmann: *Ressourcenmanagement für Echtzeitverkehre in Intranets.* Dissertation, Lehrstuhl für Kommunikationsnetze, TU München, 2003

[HOFFMANN] M. Hoffmann.: *Verfahren zur Übertragung von Dienstgüteaussagen.* Dissertation, Lehrstuhl VIII, Institut für Informatik, TU München, 2004

[HECKMANN ET AL. 04] O. Heckmann et al.: *Final Report* DFN LETSQoS Project, Milestone 4, 2004