On Interconnection Models and Strategies

Oliver Heckmann[1], Jens Schmitt[1,2], and Ralf Steinmetz[1]
[1]Multimedia Communications (KOM) • Darmstadt University of Technology • Merckstr. 25 • 64283 Darmstadt • Germany
[2]Distributed Computer Systems Lab (DISCO) • Computer Science Department • University of Kaiserslautern • Germany
Email:{Oliver.Heckmann, Jens.Schmitt, Ralf.Steinmetz}@kom.tu-darmstadt.de

Abstract[1]

There are two basic types of interconnection agreements between providers in the Internet: peering and transit. A decision every Internet network service provider (INSP) has to make is which other peering/transit INSPs to connect with. The potential peering/transit partners differ (obviously) in the advertised routes and they may differ quite drastically in the amount and type of costs (line costs, exchange point related costs, settlement costs, administrative costs) as well as in reliability and quality of service aspects. In this work, we discuss and solve problems in this context:

The first problem is finding the optimal set of peering and transit partners for one INSP at one point in time given the routing information and the cost functions of the potential peering/transit partners; different types of costs and different cost functions are considered. Reliability issues are considered (for example enforcing enough spare capacity to absorb the complete failure of one provider) as well as quality of service constraints (e.g. enforcing a certain average AS-hop count). This problem is formally described and solved with an optimal algorithm and compared with heuristics.

Tariffs and traffic are in a permanent change, thus an INSP always has to rethink whether his current choice of peering/transit partners is still optimal for it or if it may be worthwile the administrative effort of changing some of its peering/transit agreements. The last part of this paper deals with this problem and adapts the algorithm from the first part for this setting.

1 Introduction

The Internet consists of a huge number of networks operated by independent providers. On the one hand, these providers compete for customers, on the other hand they have to interoperate and interconnect their networks to offer worldwide connectivity. Contrary to the situation on most telecommunication markets there is no central authority in the Internet enforcing cooperation.

Providers compete in a market with a nearly transparent product like IP forwarding and therefore are in deadly competition. They have to use existing potentials for optimization and cost savings in order to survive. One of the biggest cost factors for Internet network service providers (INSP) are interconnection costs. For the German research network DFN as an example they are the highest cost factor and much higher than e.g. the hardware costs.

In this paper, we use decision theory and mathematical programming methods to model the problem of finding the optimal set of peering and transit providers for an INSP. We consider costs, reliability issues, quality of service and the fact that traffic and tariffs are changing over time. We will show how the models can easily and exactly be solved and evaluate their performance in extensive simulations.

Before we proceed we first need to define some of the terms used continuously in this paper:

INSP (Internet Network Service Provider): INSPs are access providers that connect private persons and businesses to the Internet and backbone providers, that connect other access and backbone providers.

Interconnection: An interconnection describes the connection between the networks of two different INSPs. We distinguish interconnections by their type and their method (see Figure 1).

The *interconnection type* is determined by how routes are exchanged and by the financial settlement agreement. With the *transit* interconnection type one of the INSPs (customer INSP) pays the other INSP (transit INSP) for the access to all destinations in its routing table and for announcing the customer's networks entry in its routing table.

With the *peering* interconnection type the two INSPs mutually provide access to each other's customers, typically without settlement [9], [10], [14].

The *interconnection method* describes how the physical interconnection between the two providers is realized. There can be one or more direct connections between the two providers' networks (direct line method) or an Internet exchange point (IXP) can be used. An Internet exchange point is typically used by a larger number of INSPs that are connected to a central router (exchange router method), a central switch or LAN (exchange switch method) or a WAN (exchange WAN method). The exchange switch and exchange WAN methods are the ones typically found in large IXPs (LINX, DE-CIX, Parix).

1. This work is partly sponsored by the German research network provider DFN (Deutsches Forschungsnetz e.V., www.dfn.org) as part of the LETSQoS project (www.letsqos.de).
This work is also partly sponsored by the German academic foreign exchange office DAAD (Deutscher Akademischer Auslandsdienst, www.daad.de).

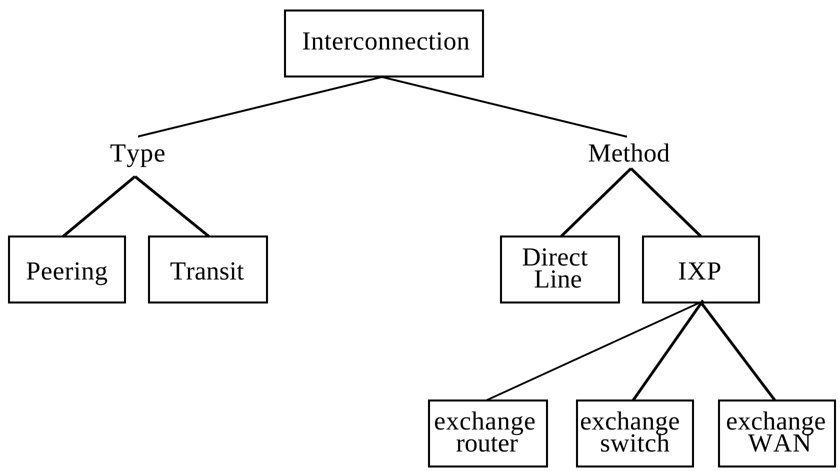

Figure 1: Interconnection Type and Method

After this introduction, we present mathematical programming models for the static optimization problem of finding the cost-minimal set of peering and transit partners at one point in time. We take different aspects into account like concave cost functions, reliability aspects (Section 3) and quality of service (QoS) requirements (Section 4). We show how the different problems can be solved exactly and evaluate the different models by extensive simulations.

In Section 5, we show how the previous models can be extended for the dynamic problem situation which is evaluating whether a given set of peering and transit partners is still optimal considering changes in the traffic mix or cost structure of the involved providers. Also considered are the administrative costs of changing peering and transit partners. Again, the models are evaluated by means of simulation.

Related work is discussed in section 6 and we conclude with a summary and brief outlook.

2 Static Models for Optimal Interconnection

In this section, we present two mathematical programming models for finding the optimal set of peering and transit partners for one INSP in the static case which means at a certain point in time. We start with a basic model that uses linear volume dependent cost functions for the transit providers and fixed costs for the peering providers. In section 2.2, the model is extended to all kinds of stepwise linear cost functions including the concave cost functions that are commonly used in reality [14]. The model can be solved by standard MIP solving techniques, we do so and use simulations to compare the results with heuristics that very much resemble what providers do today.

2.1 General Model

Finding the optimal transit and peering partners for one INSP is modelled by the following general optimization model. We assume that there are R different routes, the provider has a prognosis of the traffic for each route[1]. There are J transit providers offering transit service for all routes and I peering providers offering peering for some specific routes. The optimization model tries to minimize the costs which are fixed costs for peering partners and fixed costs plus volume dependent costs for the transit interconnections.

This problem is non-trivial because the selection of the best transit provider generally depends on the amount of traffic exchanged with this provider. At the same time the decision to peer with a peering provider or not depends on the price of the transit provider and at the same time affects the transit price itself. The problem can be modeled as a mathematical programming model.

The following indices, parameters and variables are used:

Indices[2]

$i = 1, ..., I$	peering provider i.
$j = 1, ..., J$	transit provider j.
$r = 1, ..., R$	route r.

Parameters

$\overline{x}_r$	traffic prognosis for route r.
l_i^P	fixed costs for an interconnection with peering provider i.
c_i^P	capacity of peering provider i.
$\Re_i$	set of routes offered by peering provider i.

1. Please note that a route in the context of this paper is a non-overlapping aggregation of BGP routes. Typically each peering provider has one route (its own network) and there is an (I+1)th route for the rest of the Internet.

2. Unless otherwise indicated the indices always run from the bounds presented here.

l_j^T	fixed costs for an interconnection with transit provider j.
$C_j\left(\sum_r x^T{}_{jr}\right)$	cost function of transit provider j, costs are a function for the traffic passing through provider j.
c_j^T	capacity of transit provider j.

Variables

x_{jr}^T	amount of traffic for route r passed through transit provider j.
x_{ir}^P	for $r \in \Re_i$, amount of traffic for route r passed through peering provider i.
y_i^P	binary variable, 1 if an interconnection to peering provider i is made and 0 otherwise.
y_j^T	binary variable, 1 if an interconnection to transit provider j is made and 0 otherwise.

Using a general cost function $C_j\left(\sum_r x_{jr}^T\right)$ the model is:

$$\text{Minimize } \sum_j C_j\left(\sum_r x_{jr}^T\right) + \sum_j l_j^T y_j^T + \sum_i l_i^P y_i^P \quad (1)$$

subject to

$$\sum_{i | i \in \Re_i} x_{ir}^P + \sum_j x_{jr}^T = \overline{x_r} \qquad \forall r \quad (2)$$

$$\sum_{r \in \Re_i} x_{ir}^P \le c_i^P y_i^P \qquad \forall i \quad (3)$$

$$\sum_r x_{jr}^T \le c_j^T y_j^T \qquad \forall j \quad (4)$$

$$x_{ir} \ge 0 \qquad \forall i \forall (r \in \Re_i) \quad (5)$$

$$x_{jr} \ge 0 \qquad \forall j \forall r \quad (6)$$

$$y_j^T \in \{0, 1\} \qquad \forall j \quad (7)$$

$$y_i^P \in \{0, 1\} \qquad \forall i \quad (8)$$

The target function (1) minimizes the fixed costs plus the volume depended costs. Constraint (2) makes sure that the complete traffic demand $\overline{x_r}$ is satisfied by the combination of peering and transit providers chosen. Constraints (3) and (4) are the capacity constraints for peering rsp. transit.

At the same time constraints (3) and (4) force the according binary variables y_i^P and y_j^T to one if any amount of traffic is sent over the according peering/transit provider i/j.

Constraints (5) and (6) are the non-negativity and (7) and (8) the binary constraints for the variables.

If the cost function is a simple linear function with price k_j per unit of volume

$$C_j\left(\sum_r x^T{}_{jr}\right) = k_j \cdot \sum_r x^T{}_{jr} \quad (9)$$

then the model above can be easily solved with standard mixed integer programming techniques [8]. However, typical transit providers charge a stepwise decreasing volume dependent price as depicted in Figure 2. The model above can be extended to allow these functions. This is presented in the next section.

2.2 Model with Stepwise Linear Cost Functions

2.2.1 Description

The following models enhances the previous one by introducting stepwise cost functions. The problem here lies with concave cost functions, as the algorithm has to start using the lower parts of the cost functions first.[1]

1. Convex costs functions are unrealistic, we do not use them in this paper. However, they can be modeled with the very same optimization model without changes.

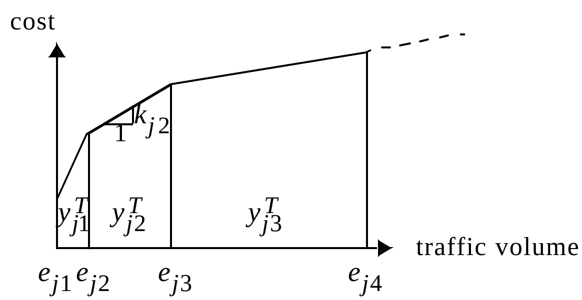

Figure 2: Stepwise decreasing cost function of transit

The following indices, parameters and variables are used:

Indices[1]

$i = 1, ..., I$	peering provider i.
$j = 1, ..., J$	transit provider j.
$r = 1, ..., R$	route r.
$m = 1,..., M_j$	part m of cost function of transit provider j.

Parameters

$\overline{x_r}$	traffic prognosis for route r.
l_i^P	fixed costs for an interconnection with peering provider i.
c_i^P	capacity of peering provider i.
$\Re_i$	set of routes offered by peering provider i.
l_j^T	fixed costs for an interconnection with transit provider j.
M_j	number of steps in the cost function of transit provider j.
$c_j^T = l_{j(M_j+1)}$	capacity of transit provider j.
l_{jm}	lower volume limit of step m of the cost function of transit provider j (see Figure 2).
k_{jm}	price per unit of volume in step m of the cost function of transit provider j.

Variables

x_{ir}^P	for $r \in \Re_i$, amount of traffic for route r passed through peering provider i.
y_i^P	binary variable, 1 if an interconnection to peering provider i is used and 0 otherwise.
x_{jr}^T	amount of traffic for route r passed through transit provider j.
$\tilde{x}_{jm}^T$	traffic volume in cost function segment m of transit provider j.
y_{jm}^T	binary variable, 1 if cost function segment m of transit provider j is used and 0 otherwise.

The problem can be described as the following mixed integer programming model

$$\text{Minimize} \sum_j \sum_{m \in M_j} k_{jm} \cdot \tilde{x}_{jm}^T + \sum_j l_j^T y_{j1}^T + \sum_i l_i^P y_i^P \qquad (10)$$

subject to

$$\sum_m \tilde{x}_{jm}^T = \sum_r x_{jr}^T \qquad \forall j \qquad (11)$$

$$\sum_{i \mid r \in \Re_i} x_{ir}^P + \sum_j x_{jr}^T = \overline{x_r} \qquad \forall r \qquad (12)$$

$$\tilde{x}_{jm}^T \le (e_{jm+1} - e_{jm})\, y_{jm}^T \qquad \forall j \forall m \qquad (13)$$

$$\tilde{x}_{jm}^T \ge (e_{jm+1} - e_{jm}) y_{jm+1}^T \qquad \forall j \forall m = 1, ..., M_j - 1 \qquad (14)$$

$$\sum_{r \in \Re_i} x_{ir}^P \le c_i^P y_i^P \qquad \forall i \qquad (15)$$

1. Unless otherwise indicated the indices always run from the bounds presented here.

$$x^P_{ir} \geq 0 \qquad \forall i \forall (r \in \Re_i) \qquad (16)$$

$$x^T_{jr} \geq 0 \qquad \forall j \forall r \qquad (17)$$

$$\tilde{x}^T_{jm} \geq 0 \qquad \forall j \forall m \qquad (18)$$

$$y^P_i \in \{0, 1\} \qquad \forall i \qquad (19)$$

$$y^T_{jm} \in \{0, 1\} \qquad \forall j \forall m \qquad (20)$$

Target function (10) minimizes the total costs. In this model compared to the basic model above, we use the additional variables $\tilde{x}^T_{jm}$ to keep track of how much of the traffic of provider j is in segment m of its cost function. Constraint (11) connects the variables $\tilde{x}^T_{jm}$ to the variables x^T_{jr} of the same transit provider j, the total amount of traffic that is divided among all routes has to be equal to the traffic in all segments of the cost function.

Constraints (12) and (13) make sure that the cost function segments are filled up correctly: (12) limits the amount of traffic in one segment to the segment size, for the highest segment (12) replaces the capacity constraint in the basic model for the transit provider. For concave cost functions the higher segments would be filled up first because of the lower volume costs and the minimizing target function. Therefore (13) is necessary, a higher segment of a cost function can only be used if the lower segment is completely full.

Constraint (11) is the traffic demand constraint (see basic model), constraint (15) is the capacity constraint for the peering providers. Constraints (16) to (18) form the non-negativity and (19) to (20) the binary constraints for the variables.

The exact solution for this problem can be found using standard MIP solution techniques like branch & bound with LP relaxation in combination with the simplex algorithm or interior point methods [8].

2.2.2 Simulative Evaluation

We now evaluate the model with two heuristics by a set of simulations.

Simulation Setup We evaluate different *scenarios*, a scenario is specified by a given number of peering providers, transit providers, routes, and an interval from which traffic and costs for these providers rsp. routes are drawn. A *scenario instance* is created by randomly creating cost functions and traffic demand vectors from the scenario specific parameter intervals. We create n=100 instances per scenario, solve each instance and evaluate the average of the 100 instances.

The parameter intervals for the basic set of scenarios are given in Table1 and Table 2. For the simulations, we assume that each peering provider offers one route and always has enough capacity for that route. The traffic demand for one route is drawn equally distributed from the “traffic demand for a peering provider’s route” interval. The BGP routes not covered by the peering providers’ routes are modeled with one additional larger route. The traffic for that route is determined by the “traffic demand for the rest of the world” parameter. The fixed costs for the peering providers are drawn from the “fixed peering costs” interval, the fixed transit costs from the “fixed transit costs” interval. The variable transit costs are drawn from the “variable costs” interval for the first step of the cost function and then decrease in each further step as specified by the “degression” interval. The transit capacity is drawn from the “transit capacity” interval and split up evenly upon the different segments of the cost function.

Description	Parameter Interval
Traffic Demand for a Peering Provider’s Route	[50, 1000]
Fixed Transit Costs	[0.05, 0.5] times the total traffic
Variable Transit Costs	[0.5, 2.0]
Variable Transit Costs Degression	[5%, 20%]
Number of Steps of the Transit Cost Function	5

Table 1: Parameter interval that are equal in all basic scenarios

Bit	Description	Parameter Interval A	Parameter Interval B
1	Number of Peering Providers	30	60
2	Number of Transit Providers	15	30
4	Capacity of a Transit Provider	[25%, 50%] of total traffic	[75%, 125%] of total traffic
8	Traffic Demand for Rest of the World	30 x average traffic demand of peering providers’ route	15 x average traffic demand of peering providers’ route
16	Fixed Peering Costs	[0.25, 2.5] times the traffic for routes of peering provider	[0.125, 1.25] times the traffic for routes of peering provider

Table 2: Parameter intervals that depend on the selected scenario

Table 2 lists the 5 scenario dependent parameter ranges, we will evaluate all possible 32 combinations of them. Each scenario has a number from 0 to 31. In scenario s the parameter A from Table2 is used if the according bit in s is not set, otherwise B is used. For scenario s=7 the parameter intervals A will be used for the number of peering and transit providers and the transit capacity (bits 1, 2, 4), parameter interval *B* will be used for the traffic demand of the rest of the world and the fixed peering costs (bits 8, 16).

We use the commercial MIP solver CPLEX [12] to calculate the exact solution for the "optimal interconnection model" (**OPT**) from Section2.2. We compare the solution obtained with two heuristics.

Description of the Heuristics The first heuristic (**H1**) describes an evolutionary approach that could describe how a real INSP found his interconnection partners: Go with the cheapest (or cheapest set of) transit providers first, then look at all peering possibilities individually and evaluate each of them, if the saved transit costs from a peering possibility are lower than the costs for peering itself, then peer, otherwise do not. The second heuristic (**H2**) is called the "peering slut" heuristic: Peer with everybody, select the cheapest transit provider (or set of transit providers) for the rest of the traffic.

Performance Evaluation We first compare the solution obtained by our model with the solution obtained by the heuristics. Figure 3 shows the average costs for each scenario based on n=100 instances per scenario, each of the algorithms solved the same 100 instances per scenario. The costs are normalized to the costs of the OPT algorithm. For the heuristics the according 95% confidence

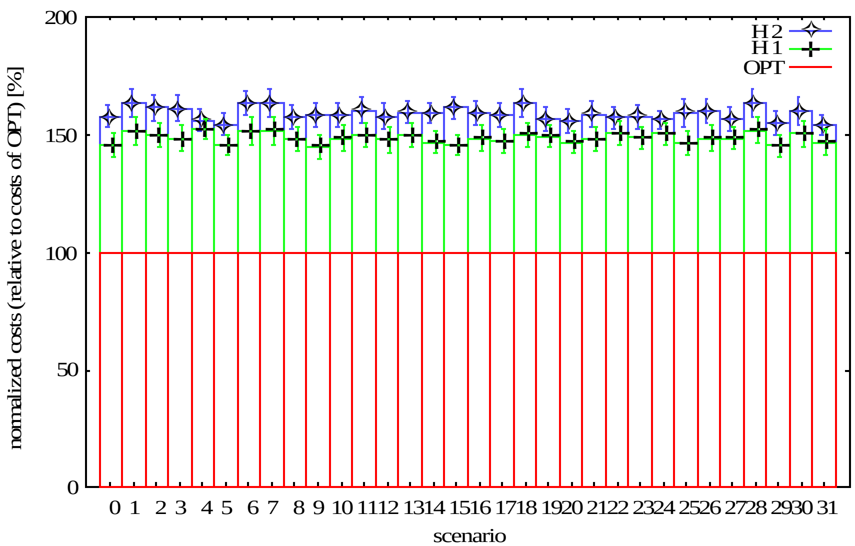

Figure 3: Normalized Costs

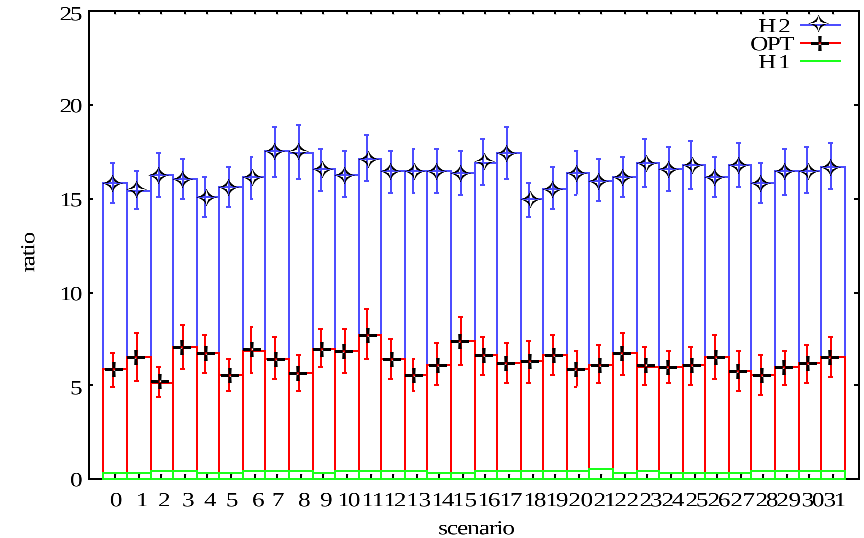

Figure 4: Peering/Transit Provider Ratio

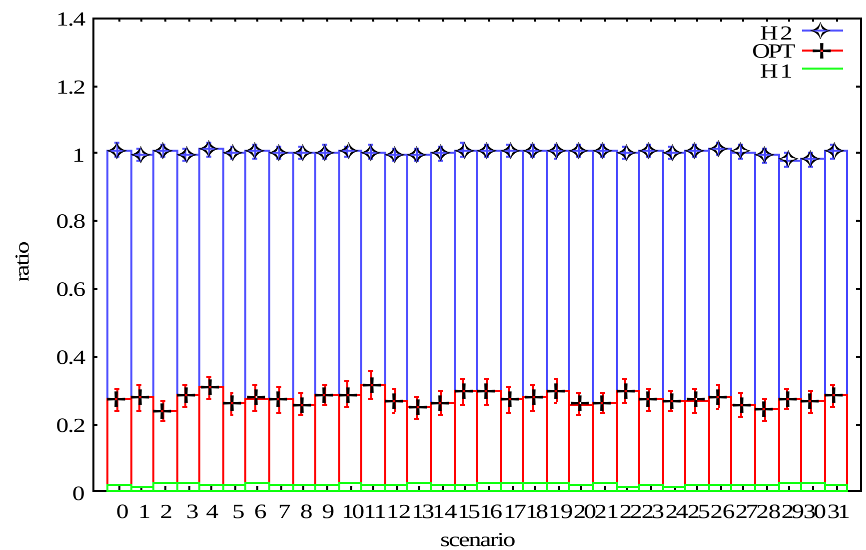

Figure 5: Peering/Transit Traffic Ratio

interval is also shown. It can be clearly seen that both heuristics lead to roughly 40 to 70% higher costs than OPT for all scenarios. H2 leads to a worse performance than H1. This holds true for all scenarios.

The reason for the bad performance can be seen in Figure 4 and Figure 5 which show the ratio of the number of peering providers per transit provider and the amount of peering traffic divided by the transit traffic.[1] H1 selects far too few peering partners compared to the optimal algorithm and H2 selects too many.

The results show that the OPT algorithm presented in this paper can save large amounts of interconnection costs for all the different scenarios when compared to two simple yet actually used real-world heuristics. The next question we investigate is whether the computational complexity of the OPT algorithm might be an obstacle for using it instead of the heuristics.

Evaluation of Computational Complexity. If we define $M = \frac{1}{J}\sum_j M_j$ and $S = \frac{1}{I}\sum_i \mathrm{Size}(\Re_i)$ then the model 2.2 needs

I(S+1) + J(2M+R) variables and (21)

I+2JM+R constraints. (22)

The time it took to solve an instance of scenario 0 on a machine with a 700 MHz Pentium 3 and 256 MB RAM is depicted in Figure 6. The number of peering providers I and transit providers J were increased as shown on the x-axis to increase the complexity of the problem. As Figure 6 shows, OPT can be solved in roughly 210 minutes for large problems with 900 providers. Given the fact that in the real-world the problem has to be solved only rarely the computational complexity is no obstacle for using OPT.

A further advantage of OPT is that it is based upon a MIP problem that can be further extended in different ways as shown in the next sections. Some of these changes would be very hard to incorporate into the heuristics.

3 Adding Reliability Measures

3.1 Policies

Reliability is an important issue for INSPs, the model 2.2 can be extended in several ways to also account for reliability. Reliability in this context is usually the protection against the failure of one or more interconnections. Looking at the

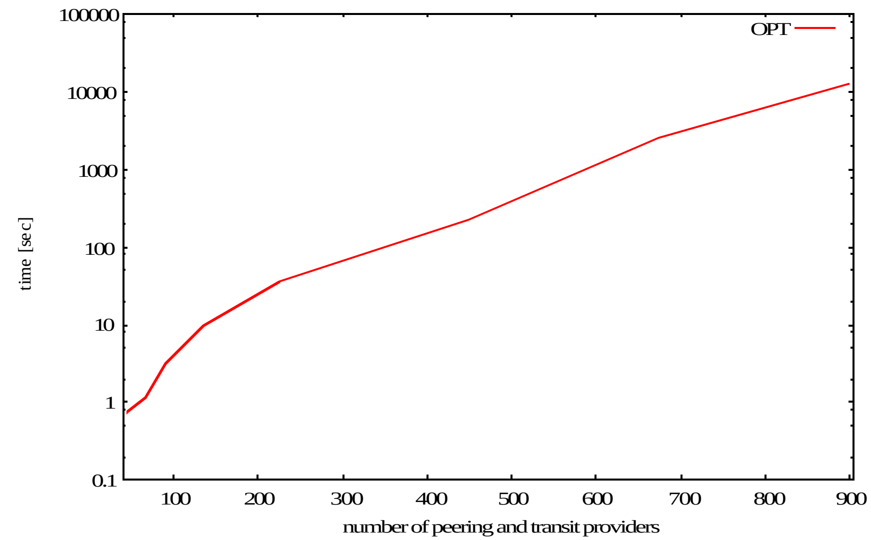

Figure 6: Time to Solve

simulation results above for the OPT algorithm if the biggest provider selected from that strategy fails, there is not enough free capacity available from the other interconnected transit providers to compensate the failure by routing the traffic destined for the failed provider. We therefore discuss several ways of extending the algorithm above.

Minimum Number of Transit Providers Policy One easy reliability policy is to interconnect with a minimum number $\underline{Y}$ of transit providers to reduce the dependency on each of them. This policy can be incorporated into the above model by adding the following constraint:

$$\sum_j y_{j1}^T \geq \underline{Y} \tag{23}$$

The advantage of this policy is its ease of use, the disadvantage is that it does not give any guarantees and fine-grained control.

Minimum Free Capacity Policy Another reliability policy is making sure that there is a minimum amount of free transit capacity available, e.g. a percentage Γ of the total traffic. The free transit capacity is the sum of all capacities of the transit providers minus the used capacities of these providers. This policy can be added to the above model 2.2 by introducing the following new parameter, vari-

[1] The 95% confidence interval is shown for H2 and OPT, for H1 it is so small it cannot be depicted.

ables and constraints[1]:

Parameter

Γ — the required fraction of free capacity from the total traffic.

Variables

f_j^T — measures the free capacity of transit provider j.

Constraints

$$f_j^T \le c_j^T - \sum_m \tilde{x}_{jm}^T \qquad \forall j \qquad (24)$$

$$f_j^T \le c_j^T \cdot y_{j1}^T \qquad \forall j \qquad (25)$$

$$f_j^T \ge 0 \qquad \forall j \qquad (26)$$

$$\sum_j f_j^T \ge \Gamma \cdot \sum_r x_r \qquad (27)$$

Constraint (24) limits variable f_j^T to the free capacity of transit provider j, (25) forces f_j^T to zero if there is no interconnection with transit provider j.

(27) enforces the minimum amount of free capacity. (26) is the non-negativity constraint for the new variables.

This policy gives the decision maker a fine-grained control over the free capacity. Its drawback is that if one interconnected provider who carries more than the fraction Γ of the traffic fails, there is not enough spare capacity. This is avoided by the next policy.

Anticipating Failure Policy This policy is a modification of the last one and makes sure that there is enough spare transit capacity if a single transit/peering provider fails completely. It can be modeled by replacing constraint (27) with the following constraints[2]:

$$\sum_{k \mid k \ne j} f_k^T \ge \sum_m \tilde{x}_{jm}^T \qquad \forall j \qquad (28)$$

[1] And we now explicitly have to assume positive fixed costs for transit providers: $l_j^T > 0$

[2] If both policies are to be combined constraint (27) is kept.

$$\sum_j f_j^T \ge \sum_{r \in \Re_i} x_{ir}^P \qquad \forall i \qquad (29)$$

Constraint (28) anticipates the failure of transit provider j, (29) does the same for peering provider i.

3.2 Simulative Evaluation

In order to evaluate the reliability policies above we use again simulations. The results presented here are based on scenario 0 but they are not significantly different for the other scenarios.

In order to evaluate the reliability performance we calculate the free transit capacity of the solutions obtained by the different policies as percentage of the total traffic. The higher the free capacity, the more buffer remains if e.g. one provider fails. For each solution we also determine whether there would be enough free capacity to carry the traffic of the biggest (peering or transit) provider if it fails, we call this the robustness. The average results and the 95% confidence intervals are depicted in the following figures as are the average costs of the solutions obtained by the different policies. The figures also contain the reference reliability and cost measures of the solution obtained for the same problems by the unmodified OPT algorithm from above (0% robustness, 1.8% free capacity).

Again we generated n=100 problem instances that were solved by the “Minimum Number of Transit Providers Policy” (**MT**), the “Minimum Free Capacity Policy” (**MC**) and by the combination of the “Minimum Free Capacity Policy” and the “Anticipating Failure Policy” (**MCAF**). The results for the “Anticipating Failure Policy” (**AF**) alone are included in the results for MCAF with a minimum free capacity of 0%.

If we look at MT which has a parameter that can only be increased in steps of one it can be seen from Figure 7 that the costs increase very quickly if the minimum number of transit providers is increased. The cost increase of the MC and MCAF policies are much smoother and more controlled (Figure 9). Figure 9 (and again Figure 10 and Figure 11) also show that MC and MCAF lead to equal results if the minimum free capacity demanded is 60% and higher. This is the amount of free capacity necessary to be able to recover from a failure of the biggest provider in (almost) all instances so the constraints of AF no longer have a significant effect on the solution.

If we analyse the reliability measures, the robustness increases quickly for MT and MC; MCAF automatically leads to full robustness because of the AF constraints. The free capacity explodes for the MT strategy while it is obviously

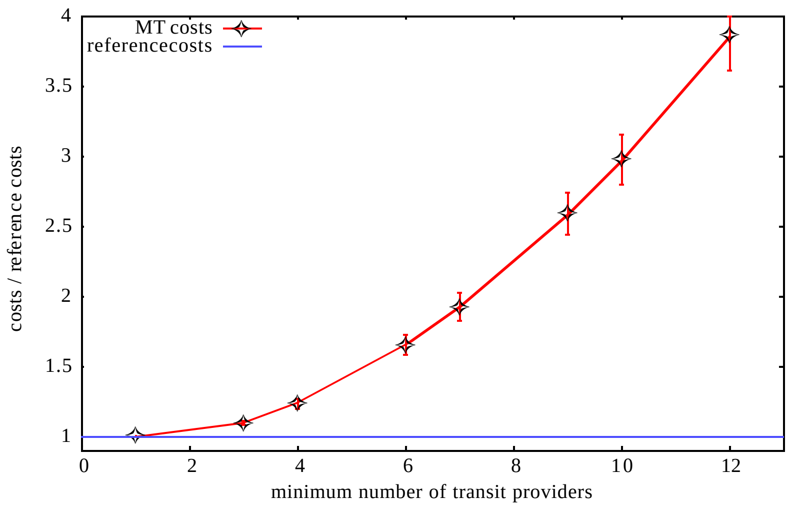

Figure 7: MT - Costs

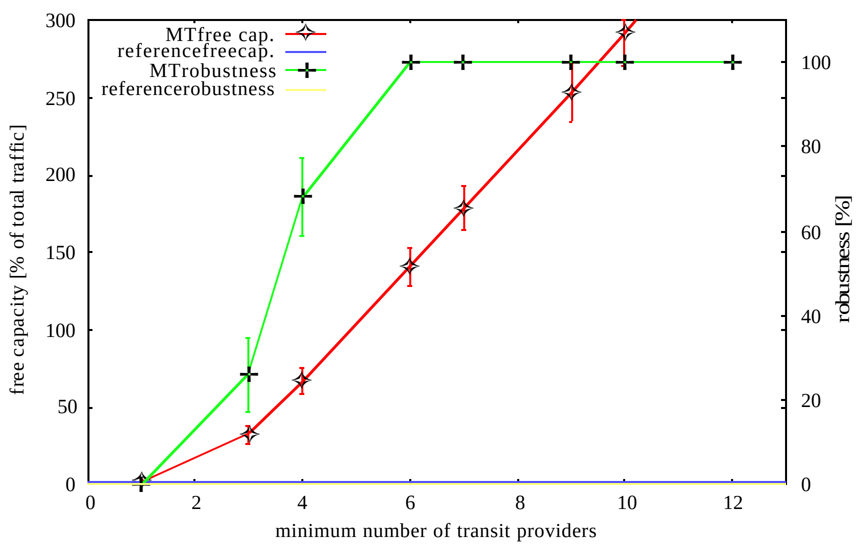

Figure 8: MT - Reliability

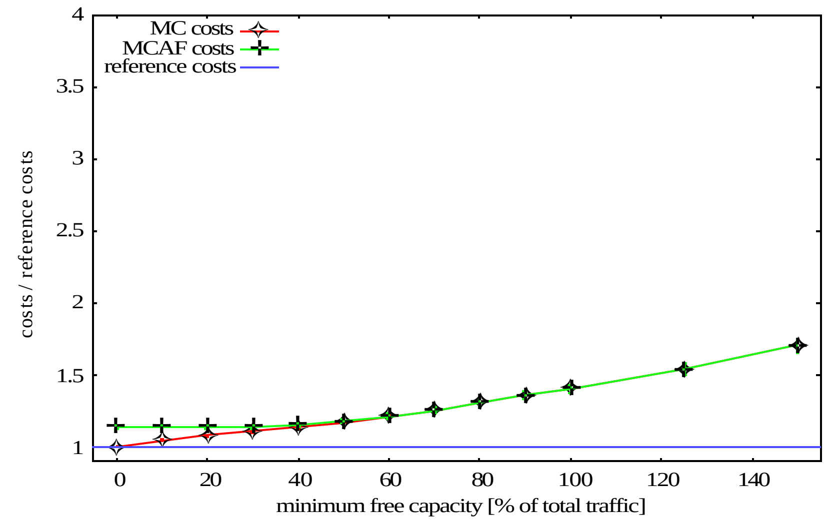

Figure 9: MC and MCAF - Costs

more controlled with the MC policy because the minimal free capacity is a parameter of that policy. Because of the AF constraints in MCAF the free capacity does not decrease for lower values of the minimum free capacity parameter.

The MT policy can be used to increase the reliability however as costs can explode and the policy parameter only indirectly influences reliability metrics like free capacity and robustness the MT policy cannot be recommended. The MCAF strategy seems to be the best choice, it offers full robustness and full control over the free capacity. Its parameter is the minimal free capacity which can be easily estimated by the decision maker. If the failure of the biggest provider is unlikely MC can also be used.

4 Awareness to Quality of Service

The quality of service achievable with its interconnections is also a typical parameter an INSP wants to optimize. In this context, quality of service can be mainly influenced by selecting interconnections such that the length of routes in terms of AS hops is kept low. Apart from that, peering or transit providers could be rated in some fashion with respect to the quality of service they usually offer and the solution could take those ratings into account. We will focus on the more objective measure of route lengths and now show several possibilities of extend-

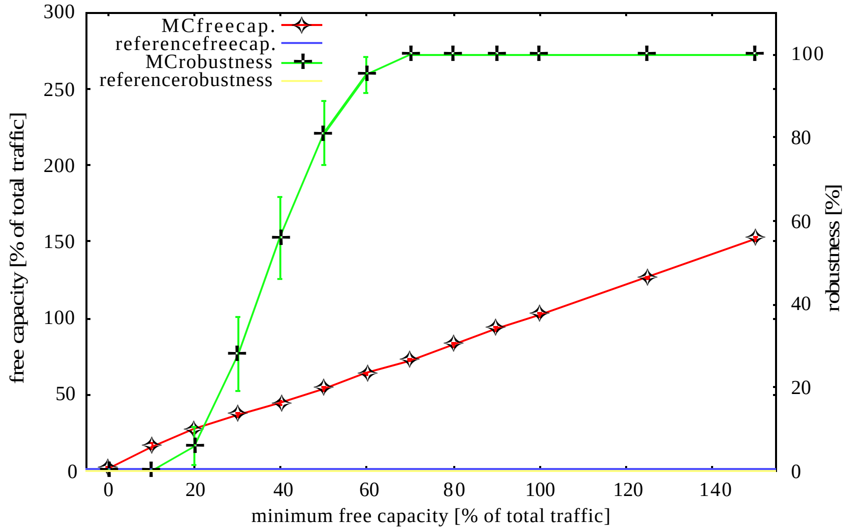

Figure 10: MC - Reliability

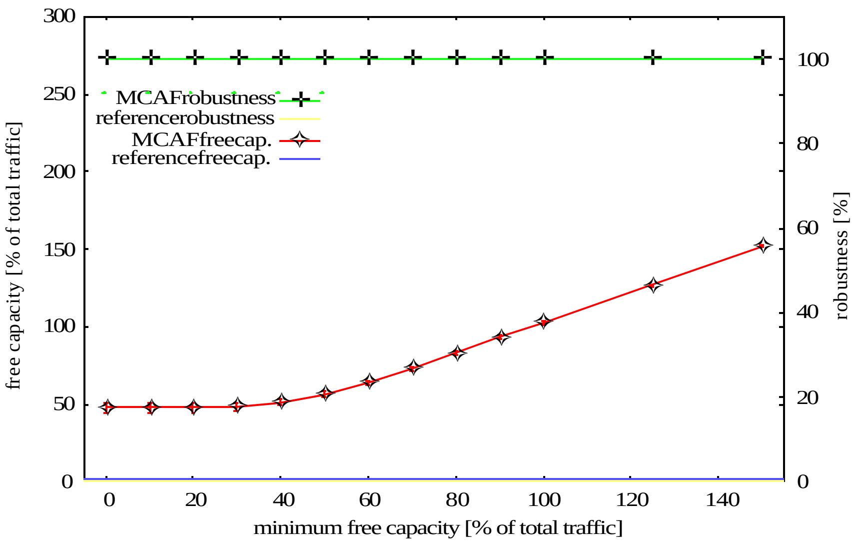

Figure 11: MCAF - Reliability

ing model 2.2 to be aware of the quality of service (QoS) that is achieved by the interconnection policy chosen. These extensions can be combined with the ones from Section3.

4.1 Strategies

The typical QoS metric used on the time-scale of interconnections is the average number of AS (autonomous system) hops for a route from the provider's network to the end-point. A lower number of hops correlates with lower delay and a lower loss probability for the packets and thus a higher utility for the customer/end-user. This is especially important for routes carrying traffic from realtime multimedia applications and network games. Peering interconnections usually offer a lower hop-count than transit interconnections because the traffic ends in the peering network. This is in fact the main reason why some larger INSPs accept peering with significantly smaller INSPs [14].

Peering Bonus The easiest way of taking the lower hop count of peering providers into account is giving peering providers with QoS sensitive routes a bonus b_i that reduces their fixed peering costs and thus makes peering with them more attractive.

This can be done by replacing the parameter l_i^P with the new parameter $\tilde{l}_i^P = l_i^P - b_i$ in model 2.2 above.

The advantage of this approach is its ease of use, the disadvantage is that the parameter b_i can be hard to estimate as it only indirectly influences the QoS.

Hop Constraint Another approach that gives the decision maker more control of the QoS parameter hop count is adding an additional constraint for the average hop count of the traffic. We introduce the following new parameters to model 2.2:

h_i^P	average hop count for traffic through peering provider i, this is typically 1 for a peering provider.
h_j^T	estimation of the average hop count for traffic through transit provider j.
q_r	delay sensitivity of the traffic on route r, routes known to carry delay sensitive traffic (e.g. to gaming sites) should obtain a higher than average q_r. q_r is used as a weight when determining the average hop count of the traffic.
$\bar{H}$	maximal average hop count allowed.

The average hop count is

$$H = \frac{1}{\sum_r \bar{x}_r q_r}\left(\sum_i h_i^P \sum_{r \in \Re_i} q_r x_{ir}^P + \sum_j h_j^T \sum_r q_r x_{jr}^T\right) \tag{30}$$

and we can add a new constraint to model 2.2 that limits H to $\bar{H}$:

$$\sum_i h_i^P \sum_{r \in \Re_r} q_r x_{ir}^P + \sum_j h_j^T \sum_r q_r x_{jr}^T \le \sum_r \bar{x}_r q_r \bar{H} \tag{31}$$

Apart from the AS hop count any other QoS metric can be modeled with this approach.

Instead of looking at the complete traffic this approach can be easily modified to take into account only a subset of the routes. For more fine-grained prediction of the hop count for transit providers h_j^T could be replaced by a route dependent prediction h_{jr}^T for route *r* through the network of transit provider *j*.

The advantage of this approach is that it gives the decision maker a finer control and with the maximum hop count an easy to understand design parameter. The disadvantages are the higher number of parameters and the slightly higher complexity of the optimization model with the additional constraint.

Hop Count Penalty Costs Strategy Decreasing the hop-count can lead to quickly increasing costs (as shown later in the simulations). The hop constraint strategy enforces a maximal hop count without respect for costs, the hop count penalty costs strategy is similar but does not enforce a maximum hop-count with a constraint but instead adds the hop-count with some penalty costs to the target function. This allows a trade-off between decreasing the hop count (which typically leads to increasing costs as we will see in the simulations) and decreasing the costs. It can be modelled with the parameters h_i^P , h_j^T , q_r from above and by adding (30) to target function (1) weighted with penalty costs Ψ .

4.2 Simulative Evaluation

In order to evaluate the QoS approaches we use simulations based on scenario 0 again, the results are not significantly different for the other scenarios. The hop count for peering providers is set to 1 and for the transit providers it is drawn equally distributed from the interval [3.0, 6.0].

The averages of *n*=100 problem instances and the 95% confidence intervals are shown for the “Peering Bonus” (**PB**), “Hop Constraint” (**HC**) and “Hop Count Penalty Costs” (**HP**) Strategies in Figure 12 to Figure 14. As reference the costs and the hop count from the plain OPT model from Section 2.2 without any QoS features are depicted, too.

With the PB strategy the costs increase only slightly and the hop count decreases only slightly even if the peering bonus is 100% of the average peering costs. The peering costs per traffic within on problem instance differ obviously quite strong so that even if they are all reduced by an average amount many of the peering providers are still not selected. This effect can also be seen in in Figure 4 and Figure 5 where even for the scenarios with lower peering costs (Bit 16 = 1) nearly as many peering providers remain unattractive as with higher average peering costs (Bit 16 = 0). The reason lies within the huge interval size for the traffic and for the peering costs.

The HC constraint offers direct control over the hop count which the other strategies do not. As can be seen in Figure 13 the costs increase quickly for lower hop counts. Decreasing the hop count by 36% to 2.4 costs roughly 38% more costs while decreasing the hop count by 25% only costs 15% more costs.

The HP strategy does not enforce a certain hop count but instead evaluates the value of the decreased hop count (expressed by the penalty costs) against the hop count. Therefore Figure 14 does not show the strong increase in costs as Figure

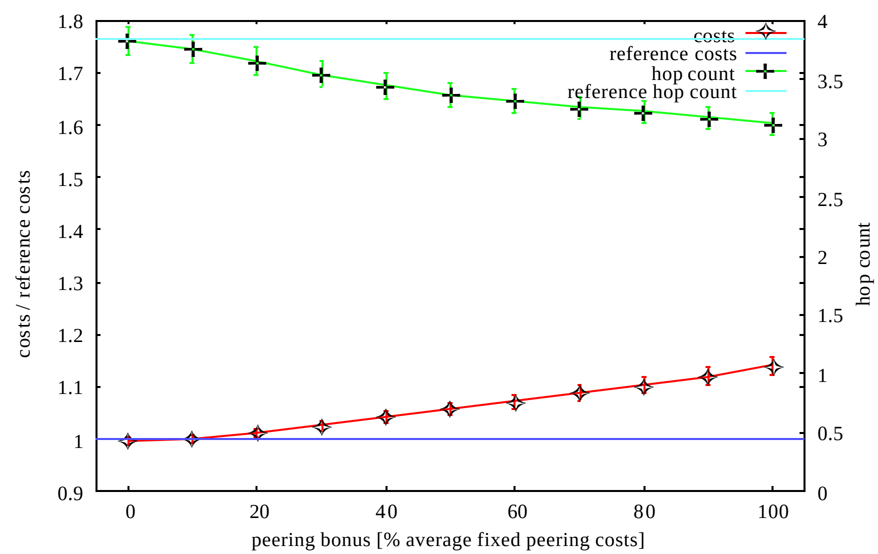

Figure 12: Peering Bonus Strategy - Costs & Hop Count

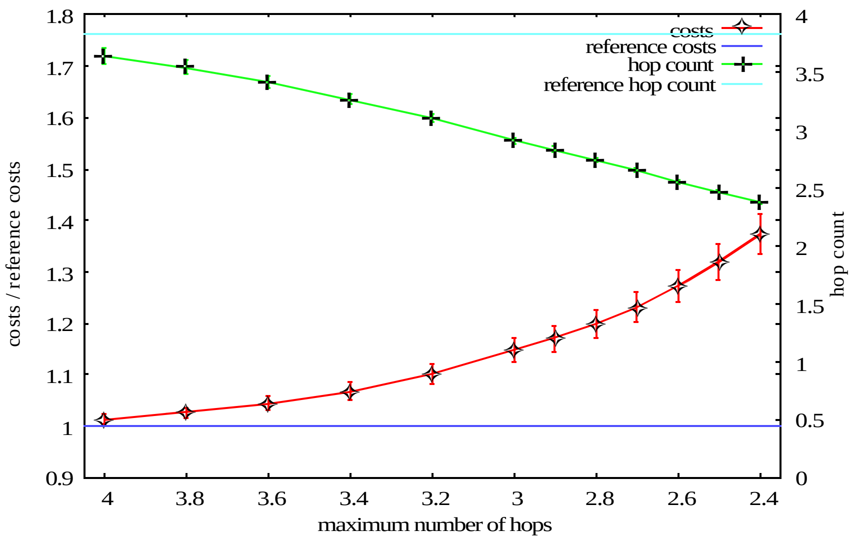

Figure 13: Hop Constraint Strategy - Costs & Hop Count

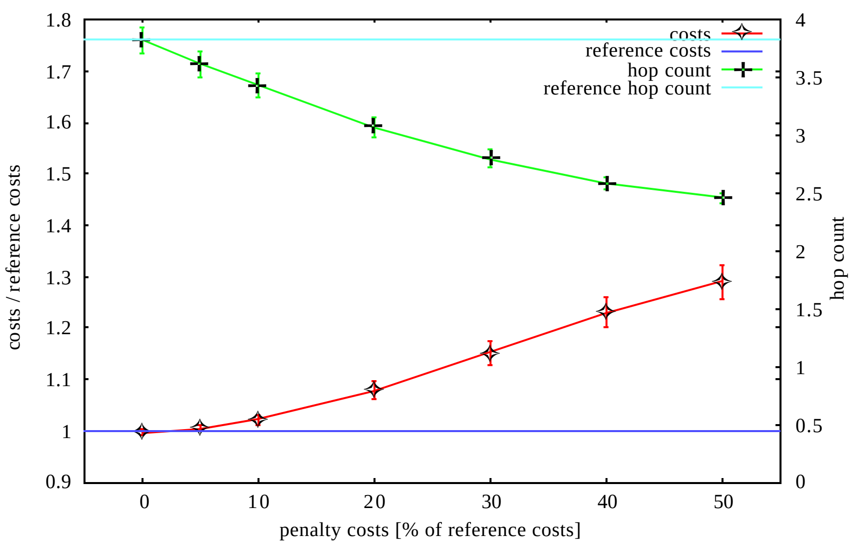

Figure 14: Hop Count Penalty Costs Strategy - Costs & Hop Count

13 while at the same time the hop count decreases nearly as much as in Figure 13.

To conclude, the influence of the PB parameter on the hop count is only an indirect one and not strong, we do not recommend this strategy. If a certain maximum hop count is strictly given, the HC strategy has to be used, otherwise if there is flexibility on the hop-count the HP offers the best way of modelling this trade-off. HC and HP can also be combined, HC could be used to ensure that a certain (higher) hop count is not exceeded while HP is used to further decrease the hop-count without ignoring the cost-increase.

5 Dynamic Model for Optimal Interconnection

The static models of section 2 to 4 can be used to calculate the optimal set of peering and transit providers for one INSP at one point in time. This is useful for a new INSP entering the market. An INSP that already has interconnections with a number of peering and transit providers faces a slightly different problem: Is the current set of peering and transit providers still optimal or is it worth changing interconnections considering the technical and administrative costs for establishing a new interconnection or cancelling an existing one?

We call this the dynamic problem and now show that the static models can be easily extended for the dynamic case. Again, we evaluate our models by simulations.

5.1 Adjusting the Static Models for the Dynamic Problem

For the dynamic case we now assume that there are interconnections to a set Θ of the I peering providers and to a set θ of the J transit providers. As the traffic requirements and the cost functions of the providers change, the dynamic problem is solved every period in order to find the new optimal set of providers.

There is typically some technical and adminstrative effort necessary for establishing a new interconnection that can be expressed by a cost term (transaction costs). Also cancelling an existing interconnection typically involves some effort that can be expressed by a cost term.

Penalty Costs Policy The costs for establishing a new interconnection can be expressed as penalty costs per period by dividing them by the number of periods an interconnection is expected to last or by a typical amortization or planning horizon. These penalty costs can be added to the fixed costs of the providers not in set Θ rsp. θ. Similarly, the costs for canceling an existing interconnection can be transformed into bonus costs per period that are substracted from the fixed costs for the providers in set Θ rsp. θ. This gives an incentive to stick with the current set of providers, we call this the penalty costs policy, model 2.2 is formally extended the following way:

Parameters

Θ	The set of peering providers that an interconnection exists with in the beginning of the current period.
θ	The set of transit providers that an interconnection exists with in the beginning of the current period.
s_i^P	for all $i \notin \Theta$, the (per period) penalty costs for establishing a new interconnection with peering provider i.
b_i^P	for all $i \in \Theta$, the (per period) bonus for not canceling an existing interconnection with peering provider i.
s_j^T	for all $j \notin \theta$, the (per period) penalty costs for establishing a new interconnection with transit provider j.
b_j^T	for all $j \in \theta$, the (per period) bonus for not canceling an existing interconnection with transit provider j.

Parameters l_i^P and l_j^T are replaced by $\tilde{l}_i^P$ rsp. $\tilde{l}_j^T$ which are defined as

$$\tilde{l}_i^P = l_i^P + s_i^P \text{ for } i \notin \Theta \text{ and } \tilde{l}_i^P = l_i^P - b_i^P \text{ for } i \in \Theta$$

$$\tilde{l}_j^T = l_j^T + s_j^T \text{ for } j \notin \theta \text{ and } \tilde{l}_j^T = l_j^T - b_j^T \text{ for } j \in \theta$$

The advantage of this policy is that the static models are easily extended this way and the involved cost terms can typically be estimated quickly and easily.

Limiting Change Policy Another policy for dealing with the dynamic problem would be limiting the amount of change (new interconnections and canceled interconnections) per period reflecting the limited technical capacities for these changes in a period or the risk of change the provider is ready to take. We call this policy "limiting change policy", it is more complicated to add to model 2.2:

Parameters

Θ, θ	see above.
W	maximum allowed number of new and cancelled interconnection agreements in this period.

Additional Constraints

$$\sum_{j \in \theta} (1 - y_{j1}^T) + \sum_{i \in \Theta} (1 - y_i^P) + \sum_{j \notin \theta} y_{j1}^T + \sum_{i \notin \Theta} y_i^P \leq W \qquad (32)$$

Constraint (32) limits the allowed number of changes. The left hand side of constraint (32) counts the binary y-variables that are 1 if an interconnection to provider i/j is made for all providers i/j that no previous interconnection agreement existed with and adds all cancellations of interconnection agreements by counting the zeroes in the binary y-variables of the providers i/j with which an interconnection agreement existed with in the last period.

5.2 Simulative Evaluation

For the simulative evaluation we create n=100 problem instances. To simulate the dynamic environment we simulate p periods per instance, to the beginning of each period the amount of traffic and the capacity of the providers growth and the fixed and variable costs vary. The range of the changes is shown in Table3 and Table 4. As in Section 2.2 we analyse different scenarios were either option A or B from Table 4 is used. If option "All Providers Available at Beginning" is used, all the providers are available for an interconnection agreement at period 0, the only change in that simulation is the traffic, capacity and cost change. If this option is not chosen, 25% of the providers are not available in period 0 and become available in a random period of the simulation (each period has the same probability).

Description	Parameter Interval
Growth of Traffic per Route per Period	[15%, 25%]
Growth of Capacity per Period	[15%, 25%]

Table 3: Parameters for all scenarios

Bit	Description	A	B
1	Number of Periods p	20	40
2	Change of Fixed Peering Costs / Period Change of Fixed Transit Costs / Period Change of Variable Costs / Period	[-20%, +5%]	[-10%, 0%]
4	All Providers Available at Beginning	Yes	No

Table 4: Scenario dependent Parameters

We now first evaluate the dependency of the results of each policy on the parameters of the policy for scenario 7 and then compare all of the policies for each scenario.

Dependency on policy parameters We start with analysing the average number of changed interconnectios and the probability of a period without any changes. These change metrics are depicted in Figure 16 for different parameter *W* that limit the number of allowed changes per period for the limiting change policy (**LC**).[1] We can see that the probability that no change occurs in a period remains low independent of *W*. The LC policy allows a number of changes each period and thus equally distributes the amount of changes over all periods. This leads to the low probability as seen in the figure. The amount of changed interconnections per period obviously decrease with *W*. The costs of LC are shown in Figure 15 and increase by only 6% if *W* is decreased from 6 to 1.

For the penalty cost policy (**PC**) the penalty costs were calculated as a constant percentage of the fixed peering rsp. transit costs for establishing a new or cancelling an existing interconnection. For penalty costs of up to 100% the probability that no change occurs increases almost linearly while at the same time the amount of changes per period decreases (see Figure 18). At the same time the costs increase slightly (see Figure 17). This is a nice result, the PC can influence the amount of change better than LC. However, if penalty costs reach 100% the amount of change is no longer decreased even for very high penalty costs. The conclusion is the the amount of change seen for high penalty costs is the change that is necessary (e.g. choosing a new transit provider) because traffic demand exceeds the capacity of the existing interconnections. There is a strange effect in the cost function Figure 17 which has a maxima at 80%. We have no explanation for this phenomenon.

Evaluation of the different Scenarios Figure 19 shows the average costs of the unmodified algorithm from 2.2, the PC policy with 50% penalty costs, the LC policy with *W*=2 and the combination of PC and LC for all scenarios. The costs can differ up to 20% between the policies. No policy leads to clearly lower or higher costs than another in all scenarios. As shown in Figure 20 the combined policy leads to the fewest changes, followed by PC and LC. The unmodified algorithm does not control change and thus leads to the highest change rate. The probability of a period without any changes is generally lowest for the LC policy and the unmodified algorithm (see Figure 21). For the other two policies the probability is significantly higher.

To conclude we can recommend using PC or the combination of PC and LC. Using LC alone or no policy at all leads to higher changes and not necessarily lower costs. The combination of PC and LC seems to be the most robust policy.

1. Again the average over n=100 problem instances and the 95% confidence interval are shown.

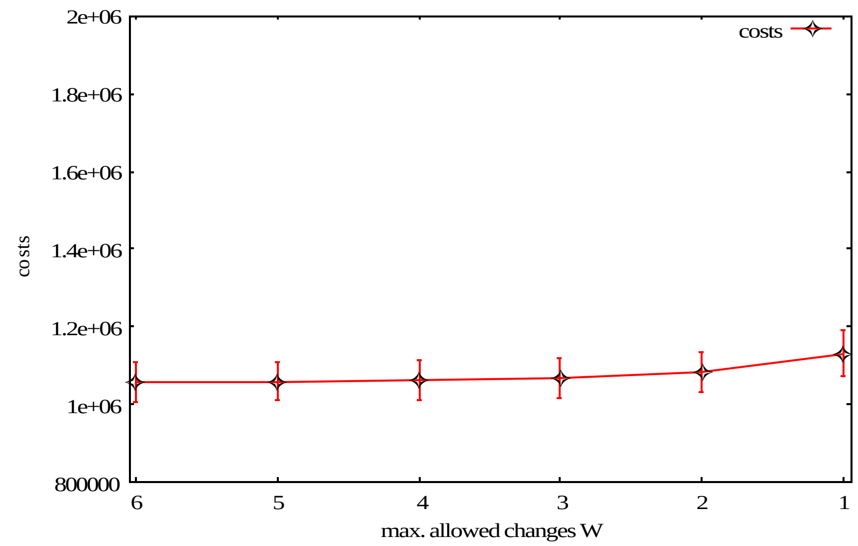

Figure 15: Limited Changes Policy (LC) - Costs

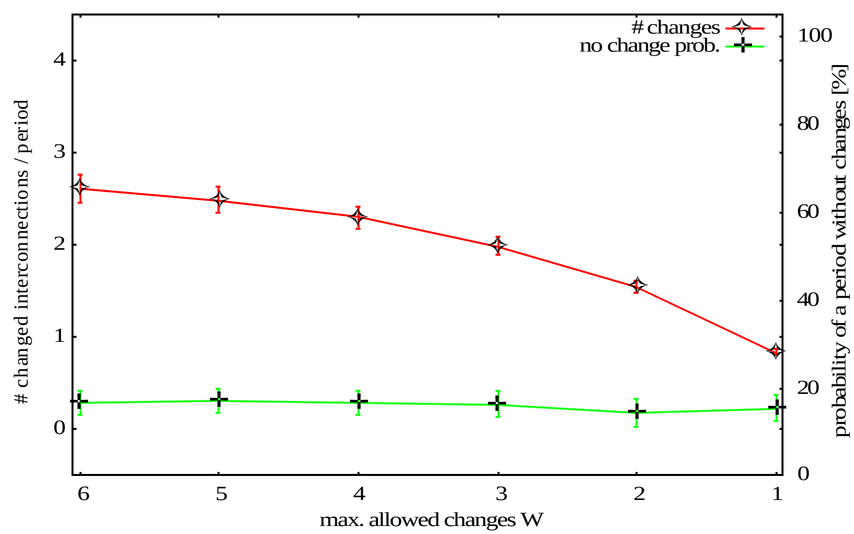

Figure 16: Limited Changes Policy (LC) - Change Metrics

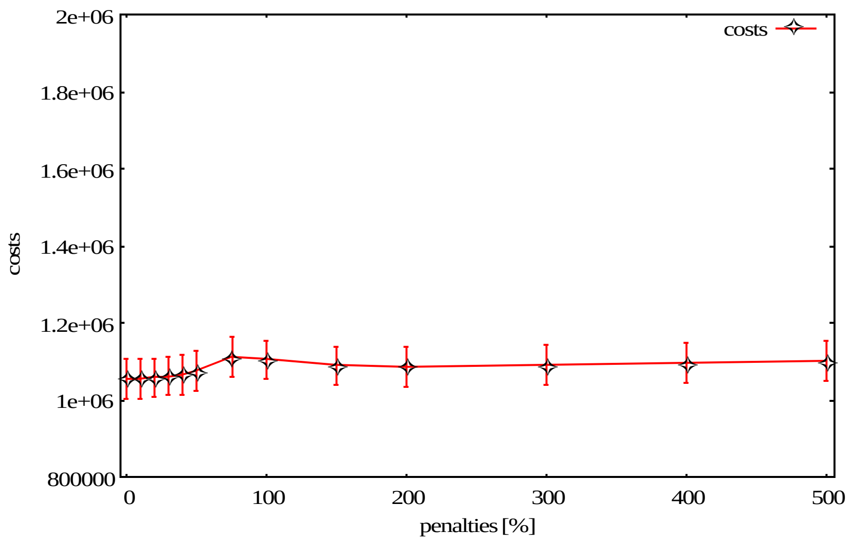

Figure 17: Penalty Cost Policy (PC) - Costs

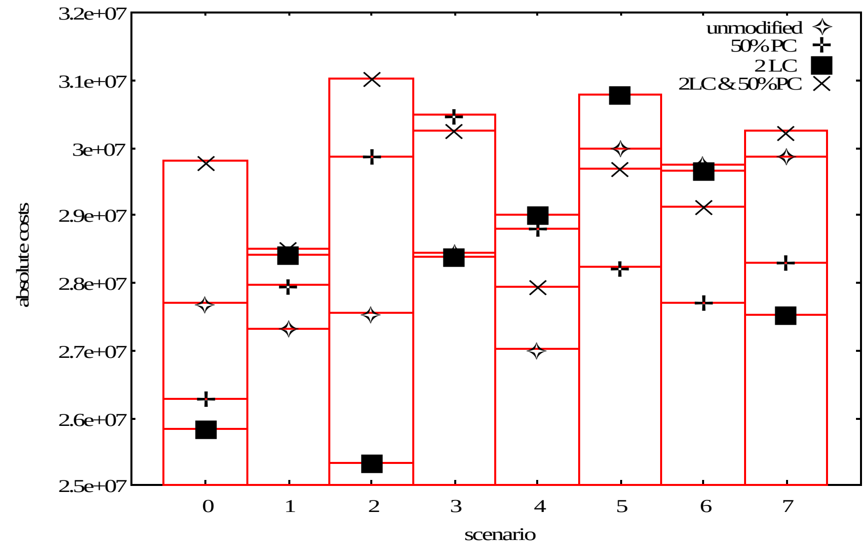

Figure 19: Average Costs (absolute)

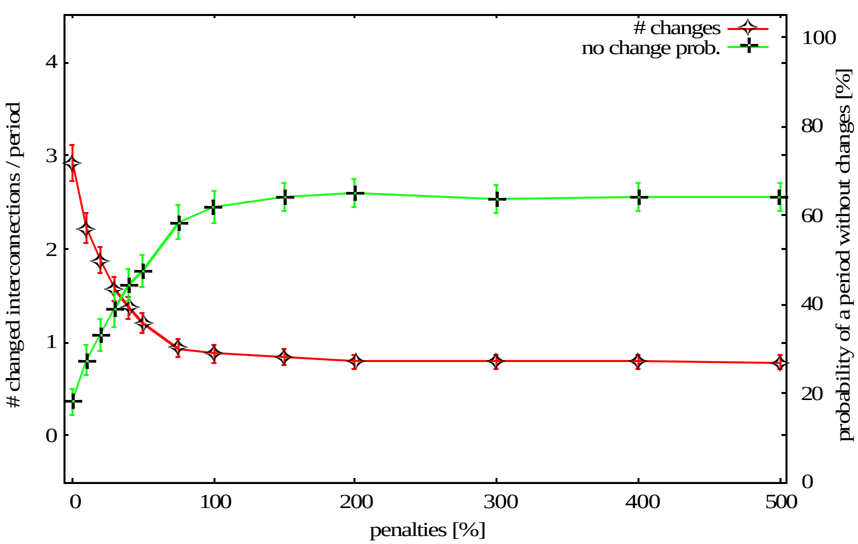

Figure 18: Penalty Cost Policy (PC) - Change Metrics

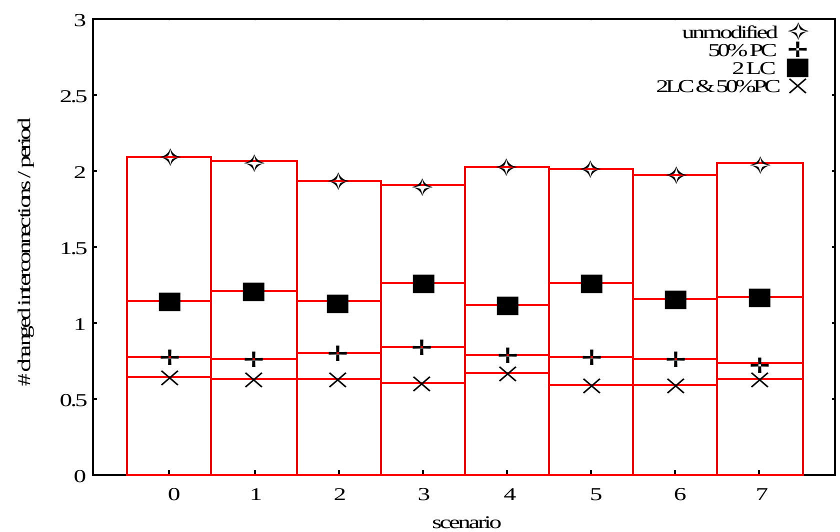

Figure 20: Number of Changed Interconnections

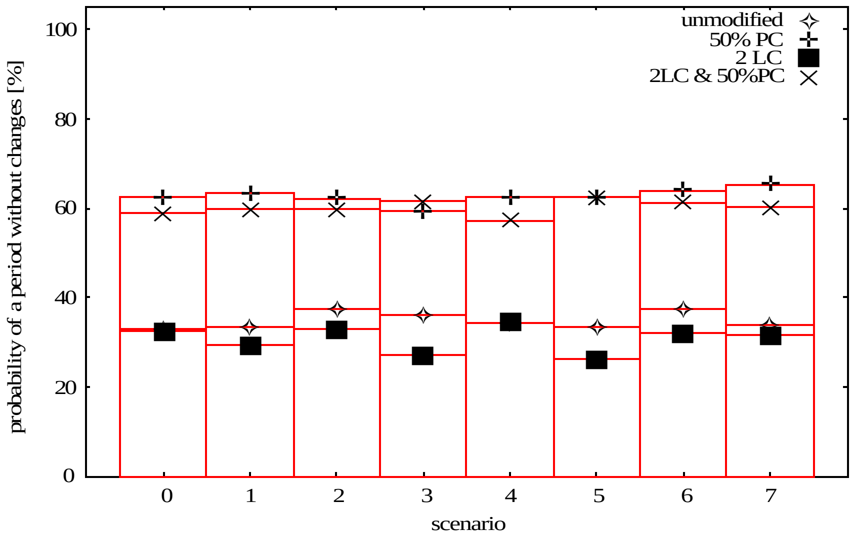

Figure 21: Probability of Period without Change

6 Related Work

The are two basic types of work about interconnections, game theoretic and decision theoretic works. This paper is an example of decision theoretic work, the optimal decision of one INSPs is analysed under a ceteris paribus constraint which effectively means that it does not anticipate possible reactions of the other parties involved. The game theoretic works focus on the anticipation of possible reactions of competing INSPs and typically model the optimization problem itself in less detail.

Game theoretic works are [2], [4], [5]. The rationales behind peering decisions for commercial INSPs and for academic research networks are analysed in [2], the focus lies on analysing competition and business stealing effects.

[4] and [5] concentrate on the economics of direct line interconnections assuming that IXs are congested and there are thus incentives to move away from them and that INSPs differentiate based on connected content providers. [4] discusses direct line interconnection agreements between INSPs that compete for customers in the same area while [5] discusses the same for INSPs that do not compete for customers in the same area.

An interesting work related to the game theoretic works is the “Peering Simulation Game” [15] where the participants play providers and negotiate interconnections.

Decision theoretic works are [6], [11], [1] and [13]. [6] is part of MPRASE (Multi-Period Resource Allocation at System Edges) [7], a mathematical framework that describes and solves all kind of resource allocation problems at the edge between two networks. [6] discusses (among other things) the selection of the cheapost provider or the cheapest combination of providers from the customer of an INSPs point of view (which could be another INSP). Similarly to Section 5 a dynamic problem with multiple periods is investigated. The approach however is fundamentally different to this paper. [6] makes one decision in the first period about the combination of providers used for the rest of the planning horizon while in Section 5 a decision is made at the beginning of each period. Also the models [6] contain less complex cost functions and no reliability and QoS issues.

While this paper discusses an interconnection problem in the current best-effort Internet, [11] presents an interconnection problem for a future QoS supporting Internet, where DiffServ [3] is used as QoS mechanism. The paper studies how the cost of quality for different QoS networks characterizes the optimal resource allocation strategies of the DiffServ bandwidth broker.

[1] presents a MIP model for finding the cost-minimal placement of a given number of interconnection points within the topology of an INSP once the decision to interconnect is made. Similar and also taking the switch/router placement (network design) problem into account is [13].

7 Summary and Outlook

In this paper we presented several optimization models for interconnections between providers. We started by presenting models and solution algorithms for the static interconnection problem which is finding the cost-optimal set of peering and transit partners for one provider. In simulations we showed that our approach is far superior to typical real-world heuristic approaches.

Next we presented and discussed several ways of extending our models to take reliability issues into account. Besides reliability, quality of service can also an important aspect for a provider. We presented and discussed several quality of service strategies for the models. In the last part of the paper we showed how to extend the static models to the dynamic problem which is evaluating whether a given set of peering and transit partners is still optimal considering changes in the traffic mix or cost structure of the involved providers. We also considered the administrative costs of changing peering and transit partners and evaluated different approaches in simulations.

The results of this paper are models, algorithms and recommendations for different reliability policies and QoS strategies for interconnection decisions from the point of view of one provider. We plan to extend this work with a case study based on real data from one provider.

Acknowledgements

The authors would like to thank John Souter from LINX (London Internet Exchange, www.linx.net) and Prof. Jessen from the DFN (German research network, www.dfn.de) for their input and fruitful discussions.

References

[1] D. Awduche, J. Agogbua, J. McManus. *An Approach to Optimal Peering Between Autonomous Systems in the Internet*. In Proceedings of IC3N 1998, Lafayette, USA, October 1998.

[2] P. Baake, T. Wichmann. *On the Economics of Internet Peering.* Berlecon Research Papers, http://www.berlecon.de/tw/peering.pdf, January 1998.

[3] D. Black, S. Blake, M. Carlson, E. Davies, Z. Wang, W. Weiss. *An Architecture for Differentiated Services*. Informational RFC 2475, Dezember 1998.

[4] R. Dewan, M. Freimer, P. Gundepudi. *Interconnection Agreements between Competing Internet Service Providers.* In Proceedings of the 33rd Hawaii International Conference of System Sciences, Hawaii, USA, January 2000.

[5] R. Dewan, M. Freimer, P. Gundepudi. *Evolution of Internet Infrastructure in the 21st Century: The Role of Private Interconnection Agreements.* In Proceedings of International Conference on Information Systems (ICIS), Charlotte, USA, December 1999.

[6] O. Heckmann, J. Schmitt, R. Steinmetz. *Multi-Period Resource Allocation at System Edges - Capacity Management in a Multi-Provider Multi-Service Internet.* In Proceedings of IEEE Conference on Local Computer Networks (LCN), Tampa, USA, November 2001.

[7] O. Heckmann, J. Schmitt, R. Steinmetz. *Multi-Period Resource Allocation at System Edges.* In Proceedings of the 10th International Conference on Telecommunication Systems Modelling and Analysis (ICTSM), Monterey, USA, October 2002.

[8] F. S. Hillier, G. J. Lieberman. *Operations Research.* McGraw-Hill, 1995.

[9] G. Huston. *Interconnection Peering and Settlement - Part I*. In Internet Protocol Journal, Vol. 2, No. 1, March 1999.

[10] G. Huston. *Interconnection Peering and Settlement - Part II*. In Internet Protocol Journal, Vol. 2, No. 2, June 1999.

[11] J. Hwang, M. B. H. Weiss. *Market-Based QoS Interconnection Economy in the Next-Generation Internet*. Journal of Information Economics and Policy, December 2000.

[12] Ilog Cplex, Commercial Mixed Integer Problem Solver, http://www.ilog.com/products/cplex/.

[13] Z. Liu, Y. Gu, D. Medhi. *On Optimal Location of Switches/Routers and Interconnection.* Technical Report, Computer Science Telecommunications, University of Missouri-KansasCity,http://www.sice.umkc.edu/~dmedhi/abs/lgm_tr_20_98_abs.html, November 1998.

[14] W. B. Norton. *A Business Case for ISP Peering.* White Paper, http://www.equinix.com, February 2002.

[15] W. B. Norton. *The Peering Simulation Game.* http://www.equinix.com.

Oliver Heckmann. Oliver Heckmann finished his degree in economics and electrical engineering (Wirtschaftsingenieur) in March 2000 at the Darmstadt University of Technology. During his studies he focused on hardware verification, optimization techniques and electronic commerce.

Since June 2000 he is a researcher in the Multimedia Distribution & Networking group of the Multimedia Communications Lab (KOM) of Prof. Steinmetz at the Darmstadt University of Technology. His research area is the Internet Service Provider in a Multiservice Internet. Within that area he is interested especially in quality of service, robust network design, traffic engineering, network simulation and the influence of peer-to-peer networking. He was working in the M3I project (www.m3i.org) and is now working in the LETSQoS project (www.letsqos.de).

Jens Schmitt. Jens Schmitt is Professor in the Computer Science Department at the University of Kaiserslautern where he is heading the Distributed Computer Systems Lab (DISCO). Before, he was research group leader of the Multimedia Distribution & Networking group at the Multimedia Communications Lab (KOM) of the Darmstadt University of Technology. He works in the fields of Quality of Service (QoS) provisioning in distributed systems, in particular in heterogeneous network scenarios, QoS for mobile communications, and scalable distribution of multimedia content with an emphasis on high availability systems. Further research interests are in network traffic modelling, real-time scheduling, and evolutionary programming.

Jens Schmitt received his Master degree (Dipl.) from the University of Mannheim in Joint Business and Computer Sciences in 1996. In 1994, during a stay at the University of Wales, Swansea, he also did a European Master of Business Sciences degree. In 2000, he received his Ph.D. degree (Dr.-Ing.) from Darmstadt University of Technology, his thesis was about “Heterogeneous Network Quality of Service Systems”.

Ralf Steinmetz. Ralf Steinmetz received the M.Sc. (Dipl.-Ing.) degree and the Ph.D. (Dr.-Ing.) degree, working in the area of Petri nets and concurrent programming languages with focus on communications, from the Darmstadt University of Technology, both in electrical engineering, in 1982 and 1986, respectively.

He was research engineer at Philips and IBM. He became responsible for the establishment, definition and realization of multimedia projects at IBM, Heidelberg. In January 1993 he was nominated as IBM "Chief Designer" for IBM Germany. From 1997 to 2001 he has been the Director of the GMD (German - National Research Center for Information Technology) institute IPSI (Integrated Publications- and Information Systems Institute) in Darmstadt. In 1999 he founded the httc (Hessian Teleme-

dia Technology Competence Center) with focus on applied networked multimedia services being applied to education projects.

He is an ICCC Governor, IEEE Fellow, and ACM Fellow.

Since 1996 Ralf Steinmetz is Professor at the Departments of "Electrical Engineering and Information Technology" and "Computer Science" at the Darmstadt University of Technology and head of the "Multimedia Communications" (KOM) Lab. His research interests are the areas of multimedia distribution, multimedia networking and multimedia semantics.