[HSchSt02] *Oliver Heckmann, Jens Schmitt, Ralf Steinmetz;* **Robust Bandwidth Allocation Strategies;** 2002 Tenth IEEE International Workshop on Quality of Service (IWQoS 2002), Miami, May 2002, S. 138-147.

Robust Bandwidth Allocation Strategies

Oliver Heckmann, Jens Schmitt, Ralf Steinmetz
Multimedia Communications Lab (KOM), Darmstadt University of Technology
Merckstr. 25 • D-64283 Darmstadt • Germany
{Heckmann, Schmitt, Steinmetz}@kom.tu-darmstadt.de

Abstract. **Allocating bandwidth for a certain period of time is an often encountered problem in networks offering some kind of quality of service (QoS) support. In particular, for aggregate demand the required bandwidth at each point in time may exhibit considerable fluctuations, random fluctuations as well as systematic fluctuations due to different activity at different times of day. In any case, there is a considerable amount of uncertainty to be dealt with by strategies for effectively allocating bandwidth. In this paper, we try to devise so-called robust strategies for bandwidth allocation under uncertainty. The notion of robustness here means that we look for strategies which perform well under most circumstances, but not necessarily best for a given situation. By simulations, we compare the different strategies we propose with respect to the robustness and performance they achieve in terms of (virtual) cost savings. We show that robustness and good performance need not be contradictory goals and furthermore that very good strategies need not be complex, either.**

Keywords. **Bandwidth management, demand uncertainty, VPN, robust algorithms.**

1. Introduction

Many decisions and optimizations in the areas of network design, traffic engineering and other resource allocation problems are based on uncertain data due to the relatively long timescales on which these mechanisms operate. In this paper, we argue that a decision maker is typically interested in robust solutions and we derive several fairly general strategies for a recurring sub-problem of the above areas - *bandwidth allocation.* The different strategies for the bandwidth allocation problem with renegotiations and reservation in advance between a customer and a network provider are implemented and their robustness and performance is tested in a series of numerical simulations.

In order to be able to guarantee a basic level of quality to a customer the provider has to know at least the upper limit of the customer's traffic, allowing him to provision the right amount of resources and perform admission control, independent of the quality of service architecture, e.g., IntServ [3] or DiffServ [2], used. In this paper, we look at a customer that needs a considerable, varying amount of network resources (e.g., bandwidth) over long timescales, for example for a provider provisioned virtual private network (see IETF working group ppvpn, [4, 9]), potentially in support of business-critical applications. The demand fluctuates heavily over the course of a day with peaks in the late morning and afternoon hours and far lower demand in the night hours as well as over the course of the week with ups on the weekdays and downs at the weekend.

Previous research work [11, 22, 35, 15] has shown that it is generally favourable for both customer and provider to allow renegotiation of bandwidth allocations. The customer saves costs during phases of low demand and the provider can make better use of the capacity of the network. Among other findings, the simulations in this paper confirm that without renegotiation the costs increase considerably (at least by a factor of 3 in our settings). A lot of research in the area of virtual private networks is done to increase the flexibility of VPNs [6, 21, 17, 18, 24], a trend which makes renegotiations easy.

On the downside, for business critical applications renegotiation can be a dangerous mechanism because customers are given no guarantees that they obtain the higher amount of bandwidth they need for their peak demands as the provider could run out of resources in such times leading to a rejection of the request.

This problem can be avoided if renegotiation is combined with reservations in advance. Customers can now request their increased bandwidth ahead of time. They can thus avoid the risk of running out of bandwidth for business critical applications. We will show in this paper that they will usually still save costs. So there are strong arguments for customers to use reservation in advance.

On the other hand with reservation in advance the provider has a better prognosis of the utilization of the network in advance which may allow him in turn to potentially allocate bandwidth more efficiently at further providers, yet the latter recursion is not in the scope of this paper. We assume that if there is not enough bandwidth for a reservation in advance that either the provider allocates the missing bandwidth at another provider or the customer changes providers.

In this paper, we take the viewpoint of a (e.g., VPN) customer mentioned above that reserves bandwidth (e.g., for one of the trunks of his VPN) in advance at a provider (e.g., offering a bandwidth-assured VPN service). The problem for the customer is that its demand forecast is necessarily uncertain.

We will use methods from stochastic programming. Stochastic programming deals with optimization under uncertainty and was introduced in 1955 by Dantzig [5]. Good overviews on stochastic programming are given in [19, 29, 33, 34]. A case study that uses stochastic programming for capacity planning in the semiconductor industry can be found in [20]. In [31], service provisioning for distributed communication networks with uncertain data is studied. Several service provisioning models are presented that account for several types of uncertainty. However, no efficient solution algorithms are presented and no simulations are carried out. Another related work is [7], here a service provider offers computational services and tries to maximize profits. In our work we consider a network service and take the perspective of the customer. Some of the methods presented in this paper were also successfully applied to a different problem domain, the planning of a production program [28].

As a remark, the mentioned problem can be considered as an instance of the MPRASE (Multi-Period Resource Allocation at System Edges) framework [12, 14]. This framework models the edge between two networks. Our recent work [27, 13, 15] has shown that many resource allocation problems at an edge are similar to a certain degree which makes it easy to reuse algorithms or to reduce problems to other, already solved ones. This background comes in handy when deriving solution algorithms in this paper. In terms of the MPRASE taxonomy [13], the bandwidth allocation problem dealt with here is "1 I 1 I 1 I 1 I FV I * I D_D" as it deals with an uncertain edge (discrete stochastic demand) between one customer and one provider, uses a one-dimensional resource model and a linear cost model with fixed and variable costs.

The paper is structured as follows: In the next section, the deterministic version of the bandwidth allocation model we use as application example is introduced and described. In the third section, uncertainty in planning problems is discussed. In particular, we show several ways of modeling uncertainty, define the robustness of a plan and show some general strategies that deal with uncertainty in model constraints. Those strategies are evaluated in the fourth section based on their robustness and general performance, before in the last section we summarize our findings and point towards future research directions in this area.

2. Application Example: Ordering a VPN Service

2.1 Bandwidth Allocation Model

In order to have a realistic background we use as application example a VPN for which bandwidth is reserved in advance. We assume that a customer requests bandwidth for a provider provisioned VPN for a longer period. The level of bandwidth r_t is flexible and can be changed (ahead of time). In order to give incentives not to change the level of bandwidth too often, fixed costs c_t^s which are incurred by each change in the level of bandwidth are introduced. These costs can be real costs or just calculatory fictive costs to account for the renegotiation overhead. Variable costs are incurred depending on the level of reserved (not necessarily used) bandwidth.

This bandwidth allocation model is formulated as a MIP (mixed integer programming [16]) problem in M1. M1 is a deterministic problem, all of the parameters are assumed to be known exactly - an obviously unrealistic assumption, which is why we introduce uncertainty in the next section.

The objective function (1) of M1 minimises total costs. (2) ensures that demand is fully satisfied in each period. Whenever r_t and r_{t-1} differ, i.e., a new bandwidth allocation takes place and s_t is forced to become *1*. This is expressed in (3) and (4). Note that s_t is set to 0 in all other cases automatically because of the non-negative entry c_t^s in the objective function.

M1 Deterministic Bandwidth Allocation Problem

Variables:

- r_t Amount of reserved capacity in period $t = 1,..., T$.
- s_t Binary variable, 1 if a (re)allocation is made at beginning of period $t = 1,...,T$ and 0 otherwise.

Parameters:

- b_t Demanded capacity in period $t = 1,...,T$. Demand is assumed to positive ($b_t > 0$).
- c_t^s Fixed allocation costs, costs per allocation. We assume positive costs ($c_t^s > 0$).
- c_t^r Variable allocation costs, costs per reserved capacity unit per period.
- r_0 Allocation level before the beginning of the first period.
- M M is a sufficiently high number (e.g., max $\{b_t\}$).

$$\text{Minimize } \sum_t c_t^s s_t + \sum_t c_t^r r_t \quad (1)$$

subject to

$$r_t \geq b_t \quad \forall t \quad (2)$$

$$r_t - r_{t-1} \leq M \cdot s_t \quad \forall t \quad (3)$$

$$r_{t-1} - r_t \leq M \cdot s_t \quad \forall t \quad (4)$$

$$s_t \in \{0, 1\} \quad \forall t \quad (5)$$

2.2 Solution Algorithms

In [14], several exact and heuristic algorithms for the problem above are presented and evaluated. In this paper, we use the cheapest exact algorithm from that work which is based on the dynamic programming paradigm [1] and has a complexity of $O(T^2)$. The function $C(t_1, t_2)$ is defined as the minimal costs for a single allocation between period t_1 and t_2. It can be calculated as

$$C(t_1, t_2) = c_{t_1}^s + \sum_{\tau = t_1}^{t_2} c_\tau^r \cdot max(b_t | t \in \{t_1, ..., t_2\}). \quad (6)$$

The algorithm exploits the structure of the problem which causes $C(t_y, t_x) \leq C(t_y, t_{x+1}) \quad \forall (t_y, t_x) | x > y$. The algorithm is depicted in Figure 1.

Preparation:
Prepare an empty array *cMin* and an empty array *pred*, each with T entries.
Start:
$cMin(1) = C(t_1, t_1)$
$pred(1) = 1$
Iteration $t = 2, ..., T$:
$cMin(t) = \min\{C(i, t) + cMin(i-1) \mid i = 1, ..., t\}$
$pred(t) = \text{argmin}\{C(i, t) + cMin(i-1) \mid i = 1, ..., t\}$
Result:
$cMin(T+1)$ contains the minimal costs while array *pred* stores the hops towards that solution.

Figure 1: Dynamic programming algorithm for the deterministic bandwidth allocation problem.

3. Bandwidth Allocation under Uncertainty

3.1 Modeling Uncertainty

If there is no uncertainty with regard to a parameter the value of that parameter is known at the time the decision is made. We then call that parameter deterministic. The deterministic case of the bandwidth allocation problem has been briefly presented in the previous section and is treated in detail in [14], where the basic problem is also advanced towards the case of multiple providers.

Types of Uncertainty. Parameters like future bandwidth demand which form the basis for a decision or optimization process can be and in practice often are uncertain. Several degrees of uncertainty for a parameter can be distinguished:

- Total uncertainty: Nothing is known about which values the parameter will take. The best thing one can do in this case is to try to react flexibly and learn from past values the parameter took. [27] deals with the single provider single customer bandwidth allocation problem under total uncertainty.
- Stochastic uncertainty: The exact value the parameter will take is not known but the decision maker knows the probability distribution of the parameter and can thus make some predictions about the parameter. [8] and [26] are typical works that deal with stochastic uncertainty for bandwidth allocation problems from a provider's point of view by assuming sources with on-off traffic.
- Discrete stochastic uncertainty: The parameter is drawn from a discrete set of values, each value has a certain probability. The set is typically modeled as a number of scenarios. This approach is discussed below in more detail as it is the approach taken in this paper.

Modeling Uncertainty with Scenarios. The idea of modeling uncertainty with scenarios has its roots in scenario analysis [25, 23]. Scenario analysis is a method for long-range planning under uncertainty. Conformant and plausible combinations of the realizations of all uncertain parameters yield a number of scenarios. These scenarios form the basis for the following decision process (e.g., a production plan is based on the assumption that one of the three scenarios will occur: "prices and demand go up", "prices fall slightly and demand remains equal", "demand goes back and prices fall heavily"). An application example and literature overview is given in [20].

However, describing uncertainty with a range of scenarios is also sensible for short- and mid-range planning and often used for stochastic programming [19, 5, 29] as it has some crucial advantages over using a parametrized probability distribution:

- It is easy and intuitive for the decision maker to create the scenarios, they could also be created automatically [10].
- Scenarios are easy to analyze, their plausibility can be approved easier than by creating a mathematical probabiilty distribution.
- Scenarios are flexible, every kind and number of possible events can be easily accounted for in the scenarios.
- Finally, scenarios can be used as a discretization of probability distributions for numerical algorithms.

Due to the advantages of the scenario method we apply it in this paper to model the uncertainty of the demand b_t for period $t = 1,\dots,T$. We assume that we have a number S of scenarios with the demand forecast b_{ts} for period t and scenario s, each scenario has a probability p_s with

$$\sum_{s=1}^{S} p_s = 1 \qquad (7)$$

3.2 Robustness

The notion of robust plans stems from decision theory [29]. Decision makers are typically evaluated ex post by how good their proposed plan performed in reality (i.e., in the scenario that actually occured). As they can loose their job and career when their plan performs badly in the occuring scenario and this typically outweighs the praise if the plan performs well, clever decision makers are risk-averse to a certain degree and biased towards robust plans. A robust plan is a plan that is judged positive in most of the scenarios and does not perform too badly in any of the scenarios.

The decision making instance in the VPN application example is also interested in robust plans, no (corporate) customer runs high risks that there are insufficient resources in critical times just for saving some communication costs.

We now derive strategies that can deal with the uncertain parameters b_{ts} and evaluate their robustness later in simulations.

3.3 Strategies for Dealing with Uncertainty

In general, uncertain parameters can occur in the objective function and the constraints of an optimization problem. If the objective function is affected the decision maker runs the risk of not achieving optimal results because of the uncertainty. If, however, the constraints are affected the decision maker risks creating plans that are not valid or realizable in reality. Dealing with uncertainty in the constraints is usually harder and more complex, yet more important than dealing with uncertainty in the objective function [29]. In the bandwidth allocation problem constraint (2) is affected by the uncertain parameters b_{ts}. We now present some general strategies how to deal with problems that have uncertain constraints.

Deterministic Substitution Strategies. For the deterministic substitution strategies we substitute the uncertain (scenario dependent) parameter b_{ts} with a deterministic (scenario independent) parameter $\hat{b}_t$ and then solve the resulting deterministic problem M1 with the algorithm presented in Section 2.2.

Several substitutions can be used. An obvious one is to use the expected value

$$\hat{b}_t = \frac{1}{S}\sum_{s=1}^{S} p_s b_{ts} \qquad (8)$$

as substitute, we call this strategy DED (deterministic with expected demand). To avoid underestimating the demand a surcharge α can be added to the substitute. We call this strategy surcharge strategy (DSUα):

$$\hat{b}_t = (1+\alpha)\cdot\frac{1}{S}\sum_{s=1}^{S} p_s b_{ts} \qquad (9)$$

For the deterministic worst-case strategy DWC we use the highest value of all scenarios as substitute:

$$\hat{b}_t = max\{b_{ts} | \forall s\} \qquad (10)$$

A plan based on the worst case values yields a solution that satisfies all constraints for all scenarios, this is why such a strategy is also called fat solution strategy [19, 29].

Chance Constrained Strategies. The deterministic strategies have no real control over the chance that their plan violates the uncertain contraints with the exception of DWC which makes sure that the plan is valid for 100% of the scenarios. The chance constrained strategy CC allows finer control over the chance that a plan is valid by introducing a factor α and forcing the uncertain constraint to be satisfied in at least α percent of the scenarios.

M2 Chance Constrained Bandwidth Allocation (CC)

Variables see M1 and:

- ζ_s Binary Variable, 1 if all demand satisfied is satisfied for scenario s and 0 otherwise.

Parameters see M1 and:

- b_{ts} Demanded capacity in scenario s = 1,...,S for period t = 1,...,T.
- p_s Probability of scenario s = 1,...,S.
- α The probability that the plan is valid.

Minimize (1) (11)

subject to (3), (4), (5) and

$$r_t + M(1-\zeta_s) \ge b_{ts} \qquad \forall t, \forall s \qquad (12)$$

$$\sum_{s=1}^{S} p_s \zeta_s \ge \alpha \qquad (13)$$

$$\zeta_s \in \{0, 1\} \qquad \forall s \qquad (14)$$

The chance constrained strategy is much harder to implement than the deterministic substitution strategies, as can be seen from the complexity of the MIP model M2: The binary variable ζ_s is used to indicate if the demand is satisfied for *all* periods of scenario s (constraint (12)). (13) forces a number of scenarios to be satisfied with a chance of at least α.

An efficient algorithm to solve the chance constrained strategy CC is to reduce it to a number of deterministic problems: For all possible permutations of ζ_s for s=1,...,S we denote Ω the set of all scenarios s for which $\zeta_s = 1$. Now look at all Ω that satisfy (13) except those Ω that have a subset $\Omega' \subset \Omega$ that satisfies (13)[1]. A deterministic problem can be formulated with

$$\hat{b}_t = max\{b_{ts} | s \in \Omega\}. \qquad (15)$$

The deterministic problems can be solved with the algorithm from Section 2.2. For all the deterministic problems, select the one that yields least costs, its optimal solution is the optimal solution of the CC strategy. If all S scenarios have the same probability then the number of deterministic problems that have to be solved is

$$\binom{S}{\lfloor \alpha S \rfloor} \qquad (16)$$

For 20 scenarios and a chance α of 0.8 this, e.g., leads to 4845 deterministic problems. Because of the high complexity we also look at a modification of the idea behind the CC strategy which makes the calculation considerably easier. Instead of requiring that a plan is valid with a chance of α for all periods we require a plan to just account for the demand of α percent of the scenarios in each period. We call this strategy the *separated* chance constrained strategy SCC. It is quite easy to implement. Assume that $b'_{t\zeta}$ are the parameters b_{ts} sorted over all scenarios by increasing values and let p'_ζ be their probabilities. For the SCC strategy we pick

$$\hat{b}_t = min\left\{ b'_{t\zeta} \middle| \sum_{\nu=1}^{\zeta} p'_\nu \ge \alpha \right\} \qquad (17)$$

as substitute and can thus reduce the SCC problem to a single deterministic problem.

Recourse Strategies. The CC strategy controls the risk that a solution is invalid to some extent. Recourse strategies control the risk in a different way. In M3 a recourse strategy with expected recourse (RER) is given.

M3 Bandwidth Allocation with Expected Recourses (RER)

Variables see M1 and

- f_{ts} Recourse for scenario s = 1,...,S for period t = 1,...,T.

Parameters see M1 and

- c_t^f Recourse costs for scenario s = 1,...,S for period t = 1,...,T.
- b_{st} Demanded capacity in scenario s = 1,...,S for period t = 1,...,T.
- p_s Probability of scenario s = 1,...,S

$$\text{Minimize } \sum_t c_t^s s_t + \sum_t c_t^r r_t + \sum_t \sum_s p_s c_t^f f_{ts} \qquad (18)$$

subject to (3), (4), (5) and

$$r_t + f_{ts} \ge b_{ts} \qquad \forall t, \forall s \qquad (19)$$

$$f_{ts} \ge 0 \qquad \forall t, \forall s \qquad (20)$$

[1] These sets cannot yield better solution than their subset, this is why they do not have to be looked at.

In constraint (19) the new variable f_{ts} measures by which amount the demand remains unsatisfied in scenario s for the resulting planned allocation in period t, r_t. The CC strategy only takes into account that demand is unsatisfied or not, the recourse strategy also takes into account how much demand is unsatisfied in a given scenario.

The recourse f_{ts} has to be penalized in the objective function. The RER does this by weighting f_{ts} with c_t^f and adding the expected value over all scenarios to the objective function(18).

In order to implement the recourse strategy the algorithm of Section 2.2 can be reapplied with some modifications.. The modified algorithm is presented in Figure 2.

Preparation:
  Prepare empty arrays *cMin, niveau* and *pred*, each with T entries.
Start:
  $niveau(1) = r_{opt}(1, 1)$
  $cMin(1) = C_{opt}(1, 1)$
  $pred(1) = 1$
Iteration $t = 2, ..., T$:
  $cMin(t) = \min\{C_{opt}(i, t) + cMin(i\text{-}1) \mid i = 1, ..., t\}$
  $pred(t) = \text{argmin}\{C(niveau(i)\ i, t) + cMin(i\text{-}1) \mid i = 1, ..., t\}$
  $niveau(t) = r_{opt}(pred(t), t)$
Result:
  cMin(T+1) contains the minimal costs while
  array *pred* stores the hops towards that solution. and
  array *niveau* the optimal reservation niveaus.

Figure 2: Dynamic programming algorithm for the recourse strategies.

It uses as cost function

$$C(r_{12}, t_1, t_2) = c_{t_1}^s + \sum_{t=t_1}^{t_2} c_t^r r_{12} + \sum_{t=t_1}^{t_2}\sum_{s=1}^{S} p_s c_t^f f_{ts}(r_{12}, t_1, t_2) \quad (21)$$

(22)

the optimal rate r_{opt} (that leads to minimal costs $C_{opt}(t_1, t_2) = C(r_{opt}(t_1, t_2), t_1, t_2)$ between t_1 and t_2)

$$r_{opt}(t_1, t_2) = r \mid C(r, t_1, t_2) = \min\{C(r, t_1, t_2) \mid \forall r \in [0, max\{b_t \mid t \in [t_1, t_2]\}]\} \quad (23)$$

and the recourse $f_{ts}(r, t_1, t_2)$ which is defined as

$$f_{ts}(r, t_1, t_2) = max\{0, b_{ts} - r\} \quad (24)$$

As c_{t1} is fixed, the minimum costs $C(r_{12}, t_1, t_2)$ from (21) can also be written as

$$\tilde{C}(r, t_1, t_2) = \sum_{t=t_1}^{t_2} c_t^r r + \sum_{t=t_1}^{t_2}\sum_{s=1}^{S} p_s c_t^f max\{0, b_{ts} - r\} \quad (25)$$

which can be rewritten as

$$\tilde{C}(r, t_1, t_2) = \left(\sum_{t=t_1}^{t_2} c_t^r\right) r - \sum_{t=t_1}^{t_2}\sum_{s=1}^{S} p_s c_t^f min\{0, r - b_{ts}\} \quad (26)$$

$$\tilde{C}(r, t_1, t_2) = \tilde{C}_1 - \tilde{C}_2 \quad (27)$$

Function $\tilde{C}_1 = \left(\sum_{t=t_1}^{t_2} c_t^r\right) r$ (28)

is a linear strictly monotonic increasing function of r.

Function $\tilde{C}_2 = \sum_{t=t_1}^{t_2}\sum_{s=1}^{S} p_s c_t^f min\{0, r - b_{ts}\}$ (29)

is a wide-sense increasing piecewise linear function that starts with negative values. Its slope is decreasing and becomes zero for all $r > max\ \{b_{ts} \mid s = 1,...,S,\ t \in [t_1, t_2]\}$. For a local minimum the slope of the difference of these two functions $\tilde{C}$ has to be zero[2]. As the slope of $\tilde{C}$ is the difference between the constant positive slope of $\tilde{C}_1$ and the decreasing slope of $\tilde{C}_2$ it is zero only for a single point t_a or a single interval $[t_a, t_b]$. $\tilde{C}$ therefore only has one local minimum which is then at the same time the global minimum. If there is only a single minimum it can be easily found with a binary search over all $r = b_{ts}$ with $t \in [t_1, t_2]$ and $s = 1,...,S$. This results in a worst-case complexity of $O(T^2 log(TS))$.

4. Simulations

A simulative comparison is used to assess the merits of the different strategies presented above. First in this section, the simulation setup and the generation of the scenarios are described. After that the robustness of the strategies is examined. Yet, robustness is not the only important criterion, the average performance of the strategies is also very important, therefore, it is evaluated in a second series of simulation runs. After that some further results from other simulations are presented shortly.

4.1 Setup

In order to generate realistic demand patterns for the scenarios the following method is used to generate a basic demand pattern for one day: A day is divided into 48 periods of 30 minutes each. A curve with peaks in the late morning and afternoon and downs during the night in accordance with [30] and [32] is used to describe empirically found traffic patterns. The average demand is 170 bandwidth units.

Based on this curve random fluctuations of up to +/- 20% are generated for all periods. This is done for every day in the week, saturdays are decreased by 60% and sundays by 80% to reflect decreased business activity during those days. The result is a basic demand pattern which is then mutated to create the different scenarios for the problem instance. The following mutations are made independently for each generated scenario: With a probability of 80% the demand of 1 to 4 whole days is scaled up or down by up to 20%, with a probability of 80% representing busy or calm days. The same is done for the whole week with a chance

2. The slope in a local minimum or maximum is zero. The difference function here obviously has no maximum.

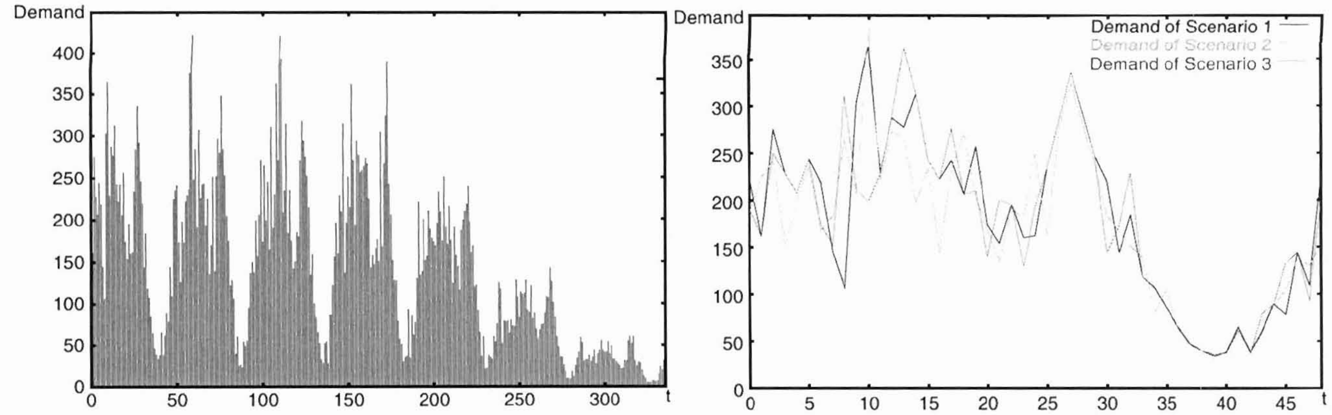

Figure 3: Demand of one scenario for one week (left) and demand of three scenarios for one day (right).

of 75%. In addition, 15% to 35% of the demand of 8 to 12 periods is shifted 1 period earlier or later, representing a slight shift in working schedules (e.g., a videoconference half an hour later as usual).

For each simulation run 20 scenarios were generated based on the basic demand. Each scenario was assigned the same probability p_s. The bandwidth demand of one sample scenario is depicted for a whole week in Figure 3 (on the left). In the same figure three example scenarios are depicted for a single day (on the right).

The fixed costs were drawn from a uniform distribution between 700 and 1000 and are equal for all periods, the variable costs were set to 5 for all periods. The strategies that were tested are listed in Table 1.

Abbrev.	Strategy
CERT	*Solution of the deterministic bandwidth allocation model M1 for the bandwidth allocation problem without uncertainty*
DED	*Deterministic strategy with expected demand*
DSUα	*Deterministic with surcharge* α*=0.05, 0.1, 0.2, 0.3, 0.4*
DWC	*Worst-Case strategy*
CCα	*Chance-Constrained strategy with chance* α*=0.8, 0.85 and 0.9*
SCCα	*Separated Chance-Constrained Strategy with chance* α*=0.8, 0.85 and 0.9*
RER c	*Recourse strategy with recourse costs c=38, 50 and 75 (the recourse costs are in the same order of magnitude as the penalty costs below)*

Table 1: Overview of the tested strategies.

4.2 Evaluating the Robustness

In order zu evaluate the robustness we have to evaluate the performance of a plan for disadvantageous scenarios. In order to do so we evaluate the plans resulting from the different strategies for each scenario. It is possible that a plan does not allocate sufficient bandwidth for the demand of some periods for a given scenario. See, e.g., Figure 4 for the allocation of the RER_{50} strategy and the demand of a certain scenario.

To account for such failures of the bandwidth allocation strategies the unsatisfied demand is penalized with penalty costs that are 10 times as high as the variable costs.

For comparison the deterministic problem without uncertainty, denoted CERT, is solved (based on the actual demand) - it naturally always leads to the best results. A strategy is good if it comes close to the costs of CERT, as a measurement we use the relative deviation $dev_X = \frac{(\text{costs}_X - \text{costs}_{CERT})}{\text{costs}_{CERT}}$ for each scenario. In order to evaluate the robustness the maximum relative deviation must not be too large. Table 2 shows the aggregated plan and penalty costs for the different strategies, averaged over 10 simulation runs (10 different problem instances). The ranking of the strategies is also listed, based on the maximum relative deviation.

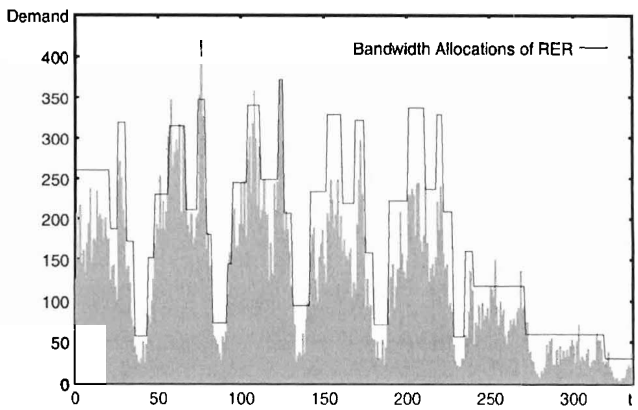

Figure 4: Demand for one week of one scenario

algorithm	av. costs	av. dev_x	min. dev_x	max. dev_x	rank
CERT	*321'023*	-	-	-	-
DED	*422'733*	*31.68%*	*17.19%*	*57.27%*	*16*
$DSU_{0.05}$	*402'780*	*25.47%*	*17.23%*	*45.97%*	*13*
$DSU_{0.1}$	*389'400*	*21.30%*	*14.47%*	*36.32%*	*9*
$DSU_{0.2}$	*380'749*	*18.60%*	*13.67%*	*27.12%*	*5*
$DSU_{0.3}$	*388'634*	*21.06%*	*14.99%*	*32.29%*	*8*
$DSU_{0.4}$	*404'725*	*26.07%*	*19.98%*	*39.16%*	*10*
DWC	*432'299*	*34.66%*	*22.82%*	*51.49%*	*15*
$SCC_{0.8}$	*377'575*	*17.62%*	*12.59%*	*26.45%*	*4*
$SCC_{0.85}$	*379'130*	*18.10%*	*12.90%*	*28.16%*	*6*
$SCC_{0.9}$	*382'857*	*19.26%*	*14.61%*	*31.43%*	*7*
$CC_{0.8}$	*413'255*	*28.73%*	*21.71%*	*43.53%*	*11*
$CC_{0.85}$	*416'191*	*29.65%*	*22.66%*	*44.74%*	*12*
$CC_{0.9}$	*420'141*	*30.88%*	*22.28%*	*46.48%*	*14*
RER_{38}	*370'638*	*15.46%*	*9.92%*	*24.06%*	*2*
RER_{50}	*368'441*	*14.77%*	*10.31%*	*22.62%*	*1*
RER_{75}	*371'997*	*15.88%*	*10.56%*	*25.75%*	*3*

Table 2: Aggregated Plan and Penalty Costs.

As one can see from Table 2, the RER strategies show the best worst-case behavior, followed by SCC and $DSU_{0.2}$. DED and DSU with lower or higher surplus perform very badly, as does DWC and CC. Those strategies cannot be considered robust. The RER and SCC strategies are more robust concerning the variation of their parameters α respectively c.

Instead of penalizing unsatisfied demand the customer could also try to short-term allocate the missing bandwidth if this on-demand renegotiation feature is supported by the provider.

Table 3 shows the planned costs plus the adaptation costs if short-term allocation is allowed (and the provider always has enough free capacity). The fixed and variable costs for short-term allocations are set to twice the costs for reservation in advance.

Note that even if short-term allocation in that fashion is possible, it is still better to use reservation in advance. First of all, it avoids the risk that there might be no short-term resources left and second reservation in advance is still cheaper, because to completely rely on short-term reservations cannot be better than twice the costs of the comparison strategy without uncertainty CERT (642'046) and these costs are far higher even than the worst strategy with reservation in advance.

algorithm	av. costs	av. dev_x	min. dev_x	max. dev_x	rank
CERT	*321'023*	-	-	-	-
DED	*409'361*	*27.52%*	*23.23%*	*34.70%*	*10*
$DSU_{0.05}$	*403'219*	*25.60%*	*21.16%*	*33.14%*	*9*
$DSU_{0.1}$	*395'743*	*23.28%*	*17.36%*	*30.11%*	*8*
$DSU_{0.2}$	*391'714*	*22.02%*	*16.65%*	*30.01%*	*7*
$DSU_{0.3}$	*397'267*	*23.75%*	*18.09%*	*35.75%*	*11*
$DSU_{0.4}$	*410'928*	*28.01%*	*20.94%*	*41.66%*	*12*
DWC	*432'299*	*34.66%*	*22.82%*	*51.49%*	*16*
$SCC_{0.8}$	*379'568*	*18.24%*	*14.37%*	*24.94%*	*1*
$SCC_{0.85}$	*381'429*	*18.82%*	*14.52%*	*25.48%*	*2*
$SCC_{0.9}$	*385'637*	*20.13%*	*16.38%*	*29.56%*	*6*
$CC_{0.8}$	*413'688*	*28.87%*	*21.71%*	*43.53%*	*13*
$CC_{0.85}$	*416'271*	*29.67%*	*22.74%*	*44.74%*	*14*
$CC_{0.9}$	*420'486*	*30.98%*	*24.21%*	*46.48%*	*15*
RER_{38}	*388'297*	*20.96%*	*15.93%*	*27.95%*	*5*
RER_{50}	*384'607*	*19.81%*	*16.32%*	*27.42%*	*4*
RER_{75}	*383'122*	*19.34%*	*15.07%*	*26.90%*	*3*

Table 3: Aggregated Plan and Adaptation Costs.

Looking at the aggregated plan plus adaptation costs, again SCC and RER lead to robust results while DWC, CC and $DSU_{0.4}$ cannot be considered robust. Interestingly the $SCC_{0.8}$ and $SCC_{0.85}$ strategies now perform better than the RER strategies. This can be explained by looking at the objective function of the RER model M3. The recourse costs which are penalized are similar to the penalty costs in Table 2. If the demand of a single period in a scenario is high the objective function assigns rather low costs to the risk of underfulfilling the demand in that single period. If, however, later this scenario occurs and a short term allocation has to be made, the costs will be relatively high since high fixed costs are incurred for only a single period. The SCC strategy on the other side would base its calculations on the α quantile of the demand in that period, running less risk of being forced to reallocate for a single period.

Summarizing, the DWC strategy is not robust and in both cases leads to very bad results. Although the DWC strategy never leads to penalty or adaptation costs, its basic plan, based on the worst case demand of all scenarios, is still much more expensive than the combination of penalty or adaptation and the planned costs of the other strategies. Only when the penalty costs are set higher than 100 times the variable costs the DWC strategy performs acceptably. Thus the DWC strategy cannot be recommended for a wide range of parameter sets of the bandwidth allocation problem.

DED and DSU with low surplus factor are also not robust. Only if the surplus factor of DSU is set correctly its performance is acceptable; it can thus not really be considered robust.

The chance-constrained strategy CC also performs badly and is dominated in performance and complexity by SCC.

SCC and RER can be considered robust. SCC bases its calculations on quantiles of the demand distribution and thus uses more information from the demand distribution than the surplus strategies DSU which explains the better performance.

RER performs very good, obviously the fine-grained control over the risk makes it more robust than the deterministic strategies. If unsatisfied demand leads to penalty costs the best results are obviously achieved if the recourse costs are to equal the penalty costs (see RER_{50}). For short-term allocations the influence of the recourse costs is not that significant, they should be set to slightly higher values (RER_{75}).

4.3 Evaluating the General Performance

So far only the robustness of the strategies has been evaluated. In a second more complex, but also more realistic series of simulations we try to evaluate the general/average performance of the different algorithms. The scenario creation is modified to reflect a greater uncertainty in the planning process. The scenarios are created and every strategy creates a plan based on the scenarios. Then one scenario is selected to occur in reality and the plans are evaluated by their performance with the demand of that scenario. The occuring scenario, however, is not part of the set of scenarios, it is just similar to one of those scenarios. We create it by selecting one of the scenarios and changing the demand of each period by +/- 2%. This reflects that the scenarios the decision making instance bases its decision on are kind of fuzzy, as they would be in reality.

The average costs, the average deviation, and its standard deviation over 20 problem instances for the aggregated plan and penalty as well as plan and adaptation costs can be found in Table 4. The ranking ist based on the average deviation. The ranking in performance is quite similar to the ranking regarding robustness in Section 4.2. The RER and SCC strategies perform best and can be recommended. RER again is better suited for the case with penalty costs (lost demand) while SCC is better suited for short-term allocations (resulting in adaptation costs).

DSU again only performs well if the surplus factor is set correctly. DED and DWC as well as CC perform relatively badly and cannot be recommended.

The conclusions from the experiments are that the RER

	Plan and Penalty Costs				Plan and Adaptation Costs			
algorithm	av. costs	av. dev_x	stddev	rank	av. costs	av. dev_x	stddev	rank
CERT	*325'511*	-	-	-	*325'511*	-	-	
DED	*431'138*	*32.45%*	*10.66%*	*15*	*415'374*	*27.61%*	*4.11%*	*12*
$DSU_{0.05}$	*409'841*	*25.91%*	*7.90%*	*11*	*405'740*	*24.65%*	*3.53%*	*10*
$DSU_{0.1}$	*396'783*	*21.90%*	*5.55%*	*9*	*401'172*	*23.24%*	*3.38%*	*9*
$DSU_{0.2}$	*386'182*	*18.64%*	*3.23%*	*6*	*394'881*	*21.31%*	*2.37%*	*7*
$DSU_{0.3}$	*392'220*	*20.49%*	*3.44%*	*8*	*399'371*	*22.69%*	*3.31%*	*8*
$DSU_{0.4}$	*407'271*	*25.12%*	*4.29%*	*10*	*413'036*	*26.89%*	*4.13%*	*11*
DWC	*431'587*	*32.59%*	*6.17%*	*16*	*431'587*	*32.59%*	*6.17%*	*16*
$SCC_{0.8}$	*381'367*	*17.16%*	*3.17%*	*4*	*383'746*	*17.89%*	*2.31%*	*2*
$SCC_{0.85}$	*381'770*	*17.28%*	*3.17%*	*5*	*383'678*	*17.87%*	*2.94%*	*1*
$SCC_{0.9}$	*386'829*	*18.84%*	*3.32%*	*7*	*389'246*	*19.58%*	*3.15%*	*5*
$CC_{0.8}$	*416'042*	*27.81%*	*4.73%*	*12*	*416'662*	*28.00%*	*4.55%*	*13*
$CC_{0.85}$	*418'068*	*28.43%*	*5.61%*	*13*	*418'326*	*28.51%*	*5.66%*	*14*
$CC_{0.9}$	*421'794*	*29.58%*	*5.91%*	*14*	*422'006*	*29.64%*	*5.95%*	*15*
RER_{38}	*375'203*	*15.27%*	*4.03%*	*3*	*390'696*	*20.03%*	*2.98%*	*6*
RER_{50}	*371'850*	*14.24%*	*3.32%*	*1*	*387'000*	*18.89%*	*3.02%*	*4*
RER_{75}	*374'424*	*15.03%*	*3.36%*	*2*	*385'422*	*18.41%*	*2.62%*	*3*

Table 4: Performance Evaluation

strategy should be used if no short-term allocations are made, it is robust and performs best for that situation. The recourse costs should be set similar to the estimated (calculatory) penalty costs of unsatisfied demand for best performance. However, the strategy is robust against a wrong setting of the recourse costs, it still performs very good as long as the recourse costs are in the same order of magnitude as the penalty costs.

For short-term allocations SCC should be preferred. However, its parameter α should not be set too high. If it is set too high the strategy approaches the DWC strategy which performed extremely bad. α=0.8 was the best choice in our simulations.

The other strategies are either not robust or perform too badly to be recommended. In practice one would intuitively often base the calculations on the expected demand (DED strategy) or on the worst-case demand (DWC). Both approaches lead to very bad results.

4.4 Further Results

In further simulations the fixed costs were varied, the uncertainty increased and the number of scenarios varied. In all cases the general conclusions from above and the general ranking of the strategies remained unaltered in principle. For the problem instances of Section 4.2 and Section 4.3 allocating resources once per week without renegotiation leads to about 3 times higher costs than those yielded by RER or SCC. This shows again that renegotiation can save a considerable amount of costs. We have explained why reservation in advance is vital to avoid the risk of not getting enough bandwidth in peak periods. Even if that is not the case reservation in advance can be better that short-term reservations: Short term reservations will be priced higher because they leave the provider with a much higher planning uncertainty and the risk of underutilizing his resources. The results show that if short-term allocations are priced even only 15 to 20% higher than long-term reservations the latter combined with a robust algorithm are cheaper than the optimal short-term allocations.

5. Summary and Outlook

In this paper, we have devised several strategies for bandwidth allocation under uncertainty. We have put emphasis on robust strategies which from a decision-theoretic viewpoint are generally desirable. By simulations we have examined our proposed strategies with respect to robustness as well as performance in terms of cost minimization. Some of the more clever strategies showed excellent robustness and performance characteristics whereas others, mainly the most simple and straightforward ones but also a fairly sophisticated one (CC), exhibited deficiencies. While we are aware that our simulation settings are quite arbitrary (due to lack of empirical data for such services) we believe that the principle lessons from these experiments are very general and that scenarios capture uncertainty in the bandwidth allocation problem very well.

As future work we perceive the investigation of more sophisticated resource models than just simple (one-dimensional) bandwidth capacities, e.g., based on controlled burstiness as for example captured by simple token buckets as has been done for deterministic demand in [15]. Furthermore, it would be interesting to extend the model towards multiple providers as has been done for the deterministic case in [14], which, however, will certainly be much more difficult for uncertain demand. Another issue which to us seems worthwhile further investigation is a rigorous comparison of systems based on reservation in advance of variable capacities vs. systems based on on-demand renegotiation under different demand situations which could quantitatively justify our assumption that for critical demand the latter system bears too many risks.

6. References

[1] R. Bellmann and S. Dreyfus. *Applied Dynamic Programming*, Princeton University Press, Princeton, N.J., 1962.

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

[3] R. Braden, D. Clark, and S. Shenker. *Integrated Services in the Internet Architecture: an Overview.* Informational RFC 1633, June 1994.

[4] R. Callon, M. Suzuki, J. De Clercq, B. Gleeson, A. Malis, K. Muthukrishnan, E. Rosen, C. Sargor, and J. J. Yu. *A Framework for Provider Provisioned Virtual Private Networks.* Internet Draft, draft-ietf-ppvpn-framework-03.txt, January 2002.

[5] G.B. Dantzig. *Linear programming under uncertainty.* Management Science 1, pp. 197-206, 1955.

[6] N. G. Duffield, P. Goyal, A. Greenberg, K. K. Ramakrishnan, and J. E. van der Merwe. *A flexible model for resource management in virtual private networks.* Proceedings of SIGCOMM, Aug. 1999.

[7] S. Dye. *The Stochastic Single Node Service Provision Problem.* http://citeseer.nj.nec.com/26408.html.

[8] M. Falkner, M. Devetsikiotis, and I. Lambadaris. *Minimum Cost Traffic Shaping: A User's Perspective on Connection Admission Control*, IEEE Communications Letters, pp. 257-259, Volume 3, September 1999.

[9] B. Gleeson, A. Lin, J. Heinanen, G. Armitage, and A. Malis. *A Framework for IP Based Virtual Private Networks.* Informational RFC 2764. draft-gleeson-vpn-framework-03.txt. February 2000.

[10] T.J. Gordon and H. Haywood. Initial experiments with the cross-impact matrix method of forecasting. In: Futures 1, pp. 100-116, 1968.

[11] M. Grossglauser, S. Keshav, and D. Tse. *RCBR: A simple and efficient service for multiple time-scale traffic.* Proceedings of SIGCOMM'95, pp. 219-230, Boston, MA, September 1995.

[12] O. Heckmann and J. Schmitt. *Multi-Period Resource Allocation at System Edges (MPRASE).* Technical Report TR-KOM-2000-05, Darmstadt University of Technology, ftp://ftp.kom.e-technik.tu-darmstadt.de/pub/papers/HS00-2-paper.pdf, October 2000.

[13] O. Heckmann and J. Schmitt. *A Taxonomy for Multi-Period Resource Allocation Problems at System Edges (MPRASE Taxonomy).* Technical Report TR-KOM-2001-09, ftp://ftp.kom.e-technik.tu-darmstadt.de/pub/papers/HS01-1-paper.pdf, Darmstadt University of Technology, September 2001, currently under submission.

[14] O. Heckmann, J. Schmitt, and 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 2001), pp. 416-425, November 2001.

[15] O. Heckmann, F. Rohmer, and J. Schmitt. *The Token Bucket Allocation and Reallocation Problems (MPRASE Token Bucket).* Technical Report TR-KOM-2001-12, Multimedia Communications (KOM), ftp://ftp.kom.e-technik.tu-darmstadt.de/pub/papers/HRS01-1-paper.pdf, December 2001, currently under submission.

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

[17] R. Isaacs and I. Leslie. *Support for Resource-Assured and Dynamic Virtual Private Networks.* IEEE Journal on Selected Areas in Communications (JSAC) 19(3), Special Issue on Active and Programmable Networks, 2001.

[18] R. Isaacs. *Lightweight, Dynamic and Programmable Virtual Private Networks.* IEEE OPENARCH, pp. 3-12, March 2000.

[19] P. Kall and S.W. Wallace. *Stochastic Programming.* Wiley, New York, 1994.

[20] S. Karabuk and S.D. Wu. *Strategic Capacity Planning in the Semiconductor Industry: A Stochastic Programming Approach.* Lehigh University, Department of Industrial and Manufacturing Systems Engineering, Report No. 99T-12. http://citeseer.nj.nec.com/karabuk99strategic.html, 1999.

[21] I. Khalil and T. Braun. *Implementation of a Bandwidth Broker for Dynamic End-to-End Resource Reservation in Outsourced Virtual Private Networks.* Proceedings of IEEE Conference on Local Computer Networks (LCN), November 9-10 2000.

[22] E. Knightly and H. Zhang. *Connection Admission Control for RED-VBR, a Renegotiation-Based Service.* Proceedings of IWQoS'96, Paris, France, 1996.

[23] D. Meadows and J. Randers, *The Limits to Growth.* Universe books, New York, 1972.

[24] D. Mitra, J. A. Morrison, and K. G. Ramakrishnan. *VPN DESIGNER: a tool for design of multiservice virtual private networks.* Bell Labs Technical Journal October-December 1998, pp. 15-31.

[25] K. Nair and R.K. Sarin. *Generating Future Scenarios - Their Use in Strategic Planning.* In Long Range Planning (LRP) 12(3), pp. 57-66, June 1979.

[26] G. Procissi, M. Gerla, J. Kim, S. S. Lee, and M. Y. Sanadidi. *On Long Range Dependence and Token Buckets.* Proceedings of SPECTS 2001, Orlando, Florida, pp. 66-73, Jul. 2001.

[27] J. Schmitt, O. Heckmann, M. Karsten, and R. Steinmetz. *Decoupling Different Time Scales of Network QoS Systems.* In Proceedings of the 2001 International Symposium on Performance Evaluation of Computer and Telecommunication Systems, pages 428–435. Society for Modelling and Simulation International, July 2001.

[28] A. Scholl and O. Heckmann. *Rollierende robuste Planung von Produktionsprogrammen.* Schriften zur Quantitativen Betriebswirtschaftslehre, 02/00 (ISSN 1432-6671), March 2000.

[29] A. Scholl. *Robuste Planung und Optimierung: Grundlagen – Konzepte und Methoden – Experimentelle Untersuchungen.* Physica, Heidelberg, 2001.

[30] J. Roberts. *Traffic Theory and the Internet.* IEEE Communications, pp 94-99. January 2001.

[31] A. Tomasgard, S. Dye, S.W. Wallace, J.A. Audestad, L. Stougie, and M.H. van der Vlerk. *Stochastic optimization models for distributed communication networks.* Working paper #3-97, Department of industrial economics and technology management, Norwegian University of Science and Technology, N- 7034 Trondheim, Norway, 1997.

[32] Traces of the *Internet Traffic Archive.* http://ita.ee.lbl.gov/html/traces.html.

[33] H. Vladimirou, S.A. Zenios and R.J-B. Wets (editors). *Models for planning under uncertainty.* Annals of Operations Research, Vol. 59, J.C. Baltzer AG Scientific Publishers, 1995.

[34] W. K. Haneveld and M. H. van der Vlerk. *Stochastic integer programming: general models and algorithms.* Annals of Operations Research (85), pp. 39-57, 1999.

[35] H. Zhang and E. W. Knightly. *RED-VBR: A renegotiation-based approach to support delay-sensitive VBR video.* Multimedia Systems, 5(3):164–176, 1997.