Internet protocol network design with uncertain demand

Morten Riis*
Department of Operations Research
University of Aarhus, Building 530
Ny Munkegade
DK - 8000 Århus C
Denmark

Anders J.V. Skriver
TDC
Sletvej 30
DK - 8310 Tranbjerg
Denmark

Steen F. Møller
TDC
Sletvej 30
DK - 8310 Tranbjerg
Denmark

May 23, 2003

Abstract

This paper is a case study concerning the design and dimensioning of the IP (internet protocol) network of TDC, the largest Danish network operator. Due to historical reasons the number of IP POPs (points of presence) in the network has reached a level, believed to be too high. To point out potential IP POPs for dismantling, we consider a network planning problem concerning dimensioning of the IP POPs and capacity expansion of the transmission links of the network. This problem is formulated as a two-stage stochastic program using a finite number of scenarios to describe the uncertain outcome of future demand. The problem is then solved by an L-shaped algorithm, and we report results of our computational experiments.

Keywords: Internet Protocol Networks; Network Design; Stochastic Programming.

1 Introduction

The foundation of IP was laid in the late 1960s as the US Department of Defense sought to create a network resilient enough to withstand an enemy attack. The ARPANET (Advanced Research Projects Agency Network), initially connecting four US universities, has since then

*Corresponding author. Email: riis@imf.au.dk

grown to what is known today as the internet. The rapid growth of the internet and its use alone provides a constantly increasing source of traffic to be carried over the IP networks of today, and moreover, IP is expected to serve as a general platform for providing data and telecommunications services in the future. Hence the problem of constructing IP networks, providing sufficient capacity for the rising demand, in a cost-efficient way, is of great importance for network providers. For a brief introduction to the concepts and terms related to IP networks, we refer to Challinor [4].

In this paper we consider the IP network of TDC, the largest Danish network operator. The network basically consists of a large number of IP POPs interconnected by a number of transmission links in the form of optical fiber cables with SDH (synchronous digital hierarchy) equipment. With IP, data to be transmitted to some destination in the network is broken down into a number of small datagrams or packets, each of which is addressed with the destination before being passed into the network. The packets are sent from one IP POP to another through the network, with each IP POP examining the destination address to decide where to send the packet next. Hence, the IP POPs serve two main purposes — they handle the routing of traffic in the network and they serve as access points to the network for customers. During the period of time when the network was built up, forming its present structure, customers accessed the network by modem through the PSTN (public switched telephone network) and access was charged as regular telephone calls. Since at that time, telephone calls in Denmark where classified as either short-distance or long-distance and charged accordingly, it was felt by TDC that all customers should be able to access the internet at the lower short-distance rate. This policy has resulted in a network with a large number of IP POPs (approximately 200) distributed across the country. Today, however, a variety of internet products is offered to customers, providing alternative technologies for access as well as several different charging schemes, all of which are independent of the physical location of the IP POP providing access for the customer to the network. Moreover, all IP POPs in the network must be maintained and, more importantly, upgraded so that sufficient capacity to switch the increasing volume of IP traffic is available. Since the total amount of switching in the network (given a certain amount of traffic) increases with the number of IP POPs in the network, these considerations have lead TDC to believe that it may be economically and practically profitable to dismantle some of the IP POPs in outer, sparsely populated regions. To point out potential IP POPs for dismantling, we formulate the network design problem of TDC as a mathematical programming problem, taking into account the maintenance and upgrading of IP POPs, the connection of customers to the network and the capacity expansion of transmission links.

To plan the design of the IP network, it is essential to have a qualified estimate of the future number of customers as well as the future volume of IP traffic to be carried over the network. Bearing in mind the rapid growth of the internet and its use, and the fact that new services to be carried over IP networks frequently emerge, it is clear that such an estimate is not readily available. In other words, the assessment of future demand inevitably involves a large degree of uncertainty that should be taken into account in the formulation of the problem, so that the performance of the resulting network is not too sensitive with respect to the actual outcome of future demand. Therefore, we employ a stochastic programming approach, treating the future number of customers and the future volume of IP traffic as random variables. The network design problem under consideration here fits into the class of so-called two-stage stochastic programming problems with linear recourse, where the decisions are divided into two groups — a group of first-stage decisions that must be taken without certain knowledge about the outcome of random parameters, and a group of second-stage decisions that may be postponed until the actual outcome of random demand has been observed. Here, the first stage corresponds to the decisions on network design that must be planned some time ahead and hence have to be based solely on the estimates of future demand, whereas the second stage corresponds to the routing of IP traffic in the resulting network, which is naturally postponed until demand has actually occurred. For a general introduction to stochastic programming we refer to the textbooks by Birge and Louveaux [3], Kall and Wallace [11] and Prékopa [13], and for a collection of previous applications of stochastic programming in telecommunications we refer to the research papers by Dempster, Medova and Thompson [5], Medova [12], Riis and Andersen [14, 15], Riis and Lodahl [16], and Sen, Doverspike and Cosares [21].

The true distribution of the random variables describing future demand can at best be estimated from historical data combined with expert opinions on future development, and this distribution is most likely absolutely continuous with a multivariate distribution function. To handle the problem computationally, however, it is common practice to replace this absolutely continuous distribution with a discrete one, employing a scenario approach where the uncertain outcome of future demand is described by a finite number of scenarios with prescribed probabilities of occurrence. This approach is justified for the class of two-stage stochastic programs with linear recourse by stability results such as those presented in e.g. Dupačová [6, 7], Kall [10], Robinson and Wets [17], Römisch and Schultz [18, 19, 20], Shapiro [22], and Wang [24], in the sense that the optimal solution of a problem with an absolutely continuous distribution may be approximated to any given accuracy by optimal solutions of problems using only a finite number of scenarios. The scenario approach allows us to solve the problem using a modified version of the so-called L-shaped algorithm. This algorithm, which was originally introduced for linear two-stage stochastic programs by Van Slyke and Wets [23], is based on Benders decomposition principle, and solves the problem by adding cuts generated from linear subproblems to a mixed-integer master problem.

This paper is organized as follows. In Section 2 we go through a thorough description of the problem and present a two-stage stochastic programming formulation with mixed-integer first-stage and linear second-stage. Next, in Section 3 we discuss the basic L-shaped algorithm and its application to the problem at hand. The algorithm has been successfully implemented, and a number of computational experiments were performed on the IP network of TDC. In Section 4 we discuss this particular problem instance, and we present computational results. Finally, in Section 5 we give some concluding remarks.

2 Problem Formulation

We start off with a conceptual description of the current network, facilitating the formulation of the network design problem as a mathematical program. First of all the region serviced by the network is partitioned into a number of subregions corresponding to the service areas of current IP POPs, so that all customers in any subregion are currently connected to the network through the same particular IP POP. Next, we will distinguish between two different network segments — the core network and the distributed network. The core network is a meshed network interconnecting a number of large IP POPs using SDH STMs (synchronous transfer modules). The transmission rates are STM-1, running approximately 155 Mbit/s (equivalent to an OC3), STM-4, running approximately 622 Mbit/s (equivalent to an OC12), and STM-16, running approximately 2.5 Gbit/s (equivalent to an OC48). The distributed network, on the other hand, consists of a large number of smaller IP POPs, each of which is connected to the rest of the network by either ATM (asynchronous transfer mode) PVCs (permanent virtual circuits) or a number of E1 (2 Mbit/s) circuits. For now, we will assume that each IP POP in the distributed network is connected to the rest of the network by two alternatively conveyed links of equal type and capacity. (In reality things are a bit more complicated as discussed in Section 4.) A small sample IP network is illustrated in Figure 1.

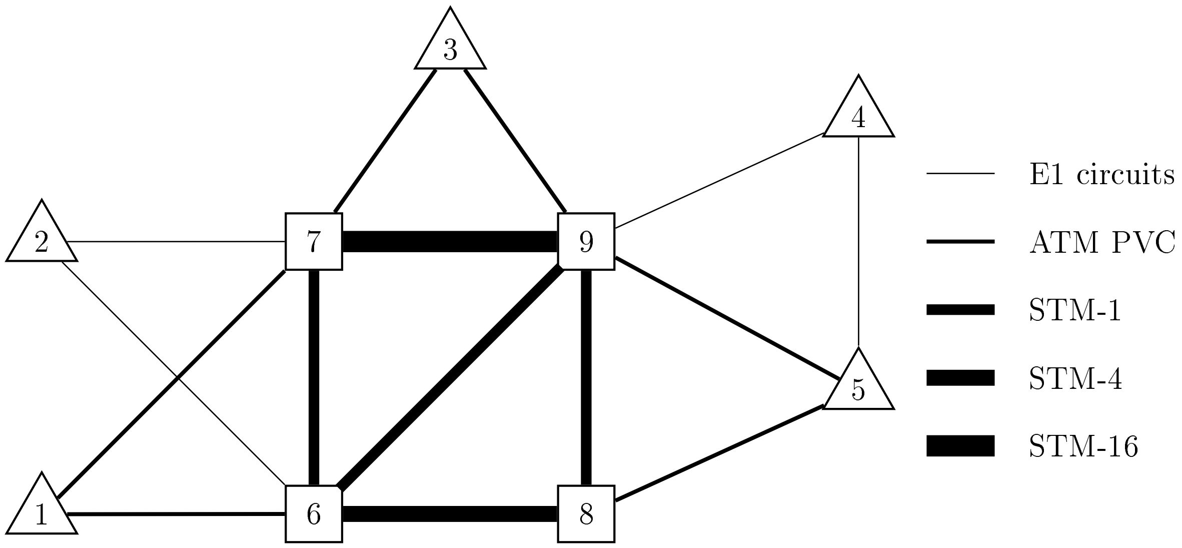

Figure 1: Illustration of a small IP network.

2.1 Network Representation

The network will be represented by a connected undirected graph $G = (V, E)$. Here the node set V represents the set of regions corresponding to current IP POPs, and hence a node $i \in V$ corresponds to a region in which all customers are currently connected to some particular IP POP. The edge set E, on the other hand, represents the set of transmission links in the current network, and hence each edge $\{i, j\} \in E$ corresponds to a transmission link between IP POPs in regions $i \in V$ and $j \in V$. The partition of the network into the core network and the distributed network is given by the following sets.

- V_1 The set of regions corresponding to IP POPs in the distributed network.
- V_2 The set of regions corresponding to IP POPs in the core network.
- $E_1(i)$ The set of transmission links connecting the IP POP in region i to the rest of the network ($i \in V_1$).
- E_2 The set of internal transmission links in the core network.

Note that $V = V_1 \cup V_2$, where the sets V_1 and V_2 are disjoint, and that $E = \big(\bigcup_{i \in V_1} E_1(i)\big) \cup E_2$, where the sets $E_1(i)$, $i \in V_1$, and E_2 are pairwise disjoint. Also note that each of the sets $E_1(i)$, $i \in V_1$, consists of just two edges. To ease the exposition, we will assume that any IP POP in the distributed network is eligible for dismantling. (The model is easily adjusted to account for the case that only some subset of the IP POPs are eligible for dismantling.) Furthermore, we will only allow an IP POP to be dismantled, if any other IP POP using it as a transit node is also dismantled.

If some IP POP is to be dismantled, the customers in the corresponding region must be connected to the network through an alternative IP POP. The following sets specify how customer connections may be transferred between IP POPs in neighboring regions.

- $N(i)$ The set of regions corresponding to IP POPs to which customers in region i can be connected if the IP POP in region i is dismantled ($i \in V$).
- $\overline{N}(i)$ The set of regions from which customers may be connected to the IP POP in region i ($i \in V$).

Note that $N(i), \overline{N}(i) \subseteq V$ and that we have $i \in \overline{N}(i)$ but $i \notin N(i)$ for all $i \in V$.

Example 1. The network in Figure 1 may be divided into a distributed network with node set $V_1 = \{1, 2, 3, 4, 5\}$ and a core network with node set $V_2 = \{6, 7, 8, 9\}$. The corresponding edge sets are for the distributed network $E_1(1) = \big\{\{1, 6\}, \{1, 7\}\big\}$, $E_1(2) = \big\{\{2, 6\}, \{2, 7\}\big\}$, $E_1(3) = \big\{\{3, 7\}, \{3, 9\}\big\}$, $E_1(4) = \big\{\{4, 5\}, \{4, 9\}\big\}$, and $E_1(5) = \big\{\{5, 8\}, \{5, 9\}\big\}$, and for the core network $E_2 = \big\{\{6, 7\}, \{6, 8\}, \{6, 9\}, \{7, 9\}, \{8, 9\}\big\}$. Clearly, the definition of the sets $N(i)$ and $\overline{N}(i)$, $i \in V$, depends on the practical possibilities to connect customers to IP POPs, and also on any specific preferences of the network operator. As mentioned above, we assume that the network operator considers all IP POPs in the distributed network eligible for dismantling. Suppose now that for practical reasons we have $N(1) = \{2, 6, 7\}$, $N(2) = \{1, 6, 7\}$, $N(3) = \{7, 9\}$, $N(4) = (5, 9)$, and $N(5) = \{4, 8\}$. Also, since the IP POPs in the core network cannot be dismantled, we let $N(6) = N(7) = N(8) = N(9) = \emptyset$. Now, the sets $\overline{N}(i)$, $i \in V$, should be defined consistently by $\overline{N}(1) = \overline{N}(2) = \{1, 2\}$, $\overline{N}(3) = \{3\}$, $\overline{N}(4) = \overline{N}(5) = \{4, 5\}$, $\overline{N}(6) = \{1, 2, 6\}$, $\overline{N}(7) = \{1, 2, 3, 7\}$, $\overline{N}(8) = \{5, 8\}$, and $\overline{N}(9) = \{3, 4, 9\}$. Finally, note that IP POP 5 is used as a transit node by IP POP 4, and hence it can only be dismantled if IP POP 4 is dismantled.

2.2 Decision Variables

The most important group of decisions to be made is whether each individual IP POP should be dismantled, or maintained and possibly upgraded. To this end we assume that a set H of different IP POP classes are available, each class $h \in H$ being characterized by a certain customer- and switch-capacity of the IP POP, and the class $0 \in H$ corresponding to dismantling of the IP POP. For each region $i \in V$ we denote by $H(i) \subseteq H$ the set of available IP POP classes that may be selected in region i. The dimensioning of IP POPs is now described by the variables

- $x_{ih} = \begin{cases} 1 & \text{if a class } h \text{ IP POP is selected in region } i \ (i \in V,\ h \in H(i)), \\ 0 & \text{otherwise.} \end{cases}$

The next group of decisions concern the connection of customers to the network. We assume that all customers in regions where the IP POP is maintained remain connected to the network through that particular IP POP. Customers in regions where the IP POP is to be dismantled, however, must be connected to the network through an alternative IP POP. The transfer of customer connections to alternative IP POPs clearly cannot be decided on an individual basis, and hence we divide the customers in any particular region into a number of groups, so that, if the IP POP in that region is to be dismantled, all customers in a given group must be connected to the network through the same alternative IP POP. For each region $i \in V$, we denote by $G(i)$ the set of customer groups in region i. Note that the sets $G(i)$, $i \in V$, are disjoint and hence form a partition of the set of all customer groups, $G = \bigcup_{i \in V} G(i)$. Now, the connection of customers to the network is given by the variables

- $y_{ig} = \begin{cases} 1 & \text{if group } g \text{ is connected to the IP POP in region } i \ \big(i \in V,\ g \in \bigcup_{j \in \overline{N}(i)} G(j)\big), \\ 0 & \text{otherwise.} \end{cases}$

The last group of decisions concerns the dimensioning of transmission links. As discussed above, all IP POPs in the distributed network are connected to the rest of the network through two alternatively conveyed transmission links of equal type and capacity, and hence we use just one variable to represent the dimensioning of these two connections for each IP POP. The transmission links in the distributed network currently use either E1 circuits or ATM PVCs, and we allow a future change of type for each IP POP. Also, the transmission links from an IP POP in the distributed network may be replaced by STM-1's (implying that the IP POP becomes part of the future core network). Thus the connections in the distributed network may be selected to be one of three types — E1 circuits (type 1), ATM PVCs (type 2), or STM-1 (type 3). If STM-1 connections are selected, the standard capacity of 155 Mbit/s is provided on each of the two connections. If, on the other hand, E1 circuits or ATM PVCs are selected, the capacity of the transmission links must be decided. Hence the variables concerning dimensioning of transmission links in the distributed network are

- $z_{il} = \begin{cases} 1 & \text{if type } l \text{ connections are selected from region } i \ (i \in V_1,\ l = 1, 2, 3), \\ 0 & \text{otherwise.} \end{cases}$
- v_i Number of E1 circuits to be added to both connections from region i ($i \in V_1$).
- w_i ATM PVC capacity to be added to both connections from region i ($i \in V_1$).

Regarding the connections in the core network, three types are available — STM-1 (type 1), STM-4 (type 2) or STM-16 (type 3) — and hence we only have one group of decisions,

- $u_{ijl} = \begin{cases} 1 & \text{if the connection } \{i, j\} \text{ is selected of type } l \ (\{i, j\} \in E_2,\ l = 1, 2, 3), \\ 0 & \text{otherwise.} \end{cases}$

2.3 Parameters

Associated with each group of decisions is a corresponding cost term. First, we have the cost associated with the selection of a certain IP POP class for each region and the cost of connecting customers to the future network.

- $\cdot\, p_{ih}$ Cost of selecting a class h IP POP in region i ($i \in V,\ h \in H(i)$). This parameter includes all costs associated with any potential upgrading of the IP POP in region $i \in V$, as well as the expected present value of future maintenance costs. The cost of upgrading an IP POP is given as the cost of new equipment minus the value of existing equipment. Thus we note that for all regions $i \in V$ we have $p_{i0} \leq 0$.
- $\cdot\, q_{ig}$ Cost of connecting group g to the IP POP in region i $\left(i \in V,\ g \in \bigcup_{j \in \overline{N}(i)} G(j)\right)$.

Next is the cost associated with capacity installments on the links of the network. For the transmission links in the distributed network, the cost structure is rather complicated since the three types of capacity — E1 circuits, ATM PVCs, or STM-1 — are completely different in nature. If E1 circuits are preferred from some particular IP POP, capacity is installed in lumps of 2 Mbit/s on each of the two links connecting this IP POP to the rest of the network. If, on the other hand, ATM PVCs are preferred, a fixed cost of the ATM equipment is incurred whereas the cost of increasing capacity on each of the two connections is assumed to be linear. (Obviously, capacity is also installed in lumps on ATM connections, but the assumption of linear cost in this model, is justified by the fact that ATM connections are shared by the IP network with a number of other services.) Finally, a fixed cost is incurred when connecting an IP POP to the rest of the network by two STM-1 connections, each providing the capacity of 155 Mbit/s. The structure of the capacity expansion cost for a transmission link, on which no capacity is currently installed, is illustrated in Figure 2.

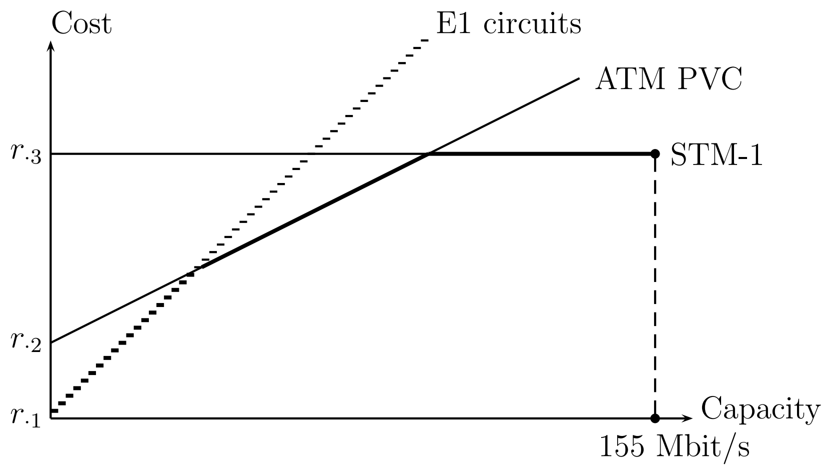

Figure 2: Cost structure for transmission links in the distributed network.

The matter is further complicated by the fact that some capacity, in the form of either E1 circuits or ATM PVCs, is already installed on all transmission links in the distributed network. If an IP POP in the distributed network is pointed out for dismantling or if a change of type of capacity on the connection from the IP POP to the rest of the network is decided, some of the currently installed equipment may be reused and hence represents a certain value. If ATM PVCs are currently used, the reusable "equipment" consists of ATM equipment installed in the IP POP as well as the ATM PVC capacity currently used on the connections from the IP POP to the rest of the network. If E1 circuits are currently used, the only reusable equipment is the actual circuits. All in all, the following parameters will be used to describe the capacity expansion of transmission links in the distributed network.

- $\cdot\ r_{il}$ Fixed cost incurred if type l connections are selected from the IP POP in region i ($i \in V_1,\ l = 1, 2, 3$). Note that if type l connections are currently used from the IP POP in region i, we have $r_{il} = 0$. If, on the other hand, some other type of connections are currently used, the parameter r_{il} represents the fixed cost associated with type l connections minus the value of existing equipment that may be reused.
- $\cdot\ a_i$ Cost of adding an E1 circuit on both connections from the IP POP in region i to the rest of the network ($i \in V_1$).
- $\cdot\ b_i$ Cost of increasing the ATM PVC capacity by 1 Mbit/s on both connections from the IP POP in region i to the rest of the network ($i \in V_1$).
- $\cdot\ \tilde{v}_i$ The number of E1 circuits currently installed on each of the two connections from the IP POP in region i to the rest of the network ($i \in V_1$).
- $\cdot\ \tilde{w}_i$ The current ATM PVC capacity of each of the two connections from the IP POP in region i to the rest of the network ($i \in V_1$).

For the internal links in the core network, the cost structure is significantly simpler, since a fixed cost is associated with the capacity provided by the preferred type of connection.

- $\cdot\ c_{ijl}$ Cost incurred if a type l connection is selected between the IP POPs in regions i and j ($\{i, j\} \in E_2,\ l = 1, 2, 3$). Again, the parameter is a net cost given as the cost of new equipment minus the value of existing equipment.
- $\cdot\ C_l$ Capacity of a type l connection in the core network ($l = 1, 2, 3$).

The next group of parameters concerns the dimensioning of IP POPs. As previously mentioned, each IP POP class is characterized by a certain customer- and switch-capacity of the IP POP. The customer-capacity restricts the number of customers that can be connected to the network through the IP POP, and is expressed as a number of sockets available for customer connections. The switch-capacity, on the other hand, restricts the amount of traffic that can be switched by the IP POP and is measured in Mbit/s.

- $\cdot\ M_h$ Customer-capacity of a class h IP POP ($h \in H$). Note that $M_0 = 0$.
- $\cdot\ N_h$ Switch-capacity of a class h IP POP ($h \in H$). Note that $N_0 = 0$.

The final group of parameters describes demand in the form of requests for customer connections and IP traffic. Demand in the form of IP traffic is modeled by means of a set K of commodities. Typically, such commodities may be defined in two different ways. A disaggregated formulation defines each commodity $k \in K$ as traffic from some customer group $o(k) \in G$ to another group $d(k) \in G$, thus resulting in a total of $O(|G|^2)$ commodities. An aggregated formulation, on the other hand, defines each commodity $k \in K$ as all traffic originating in a given group $o(k) \in G$, thus resulting in a total of only $O(|G|)$ commodities. In general, the disaggregated formulation provides a more detailed description of traffic, favorable for example when survivability requirements are to be formulated. The aggregated formulation, on the other hand, provides the advantage of reducing considerably the number of variables and constraints.

As pointed out in Section 1, the future demand is not known with certainty at the point in time when the network design problem is to be solved. We include this inherent uncertainty in the formulation using a scenario approach. Hence a number of scenarios is defined, each scenario $s = 1, \ldots, S$ corresponding to a possible future outcome of random demand.

- L_g^s Number of sockets required to connect group g to an IP POP under scenario s $(g \in G,\ s = 1, \ldots, S)$.
- D_{kg}^s Net demand for commodity k from group g under scenario s $(k \in K,\ g \in G,\ s = 1, \ldots, S)$. We emphasize that this parameter represents net demand, and hence it is given a sign so that for all $k \in K$ and $s \in \{1, \ldots, S\}$ we have $\sum_{g \in G} D_{kg}^s = 0$, $D_{kg}^s \geq 0$ for $g \in G \setminus \{o(k)\}$, and $D_{k,o(k)}^s < 0$.
- d_g^s Total amount of traffic terminating at group g under scenario s $(g \in G,\ s = 1, \ldots, S)$. This parameter includes traffic to group g from any other customer group under scenario s (i.e. $\sum_{k \in K: o(k) \neq g} D_{kg}^s$) as well as all internal traffic in group g under scenario s.

Remark 2.1. Employing the aggregated formulation of commodities, it is easily seen that for $s \in \{1, \ldots, S\}$, $k \in K$, and $g \in G \setminus \{o(k)\}$, the parameter D_{kg}^s in fact represents traffic from group $o(k)$ to group g, whereas the parameter $D_{k,o(k)}^s$ represents total traffic from group $o(k)$ to all other customer groups. Using the disaggregated formulation, on the other hand, we see that for $s \in \{1, \ldots, S\}$ and $k \in K$, the parameters $D_{k,o(k)}^s$ and $D_{k,d(k)}^s$ both represent traffic from group $o(k)$ to group $d(k)$, whereas $D_{kg}^s = 0$ for $g \in G \setminus \{o(k), d(k)\}$.

Finally, we will need to define the maximum aggregate traffic demand

- $D \;=\; \max_{s=1,\ldots,S} \sum_{k \in K} \sum_{g \in G \setminus \{o(k)\}} D_{kg}^s.$

2.4 Capacity Constraints

Capacity constraints will be imposed in two different contexts. The first group of capacity constraints concern the customer-capacity of each individual IP POP. Here, we find it convenient to introduce for $i \in V$ and $s \in \{1, \ldots, S\}$ the surplus variable λ_i^s, representing the shortage of sockets for customer connections to the IP POP in region i under scenario s. Hence for each scenario $s \in \{1, \ldots, S\}$ the constraints are formulated as

$$\sum_{h \in H(i)} M_h x_{ih} + \lambda_i^s \geq \sum_{j \in \overline{N}(i)} \sum_{g \in G(j)} L_g^s y_{ig}, \quad i \in V. \tag{2.1a}$$

The remaining capacity constraints concern the restrictions on the flow of IP traffic in the network. To formulate these constraints, we need to determine the traffic flow in the network under any scenario. To this end we define for each $s = 1, \ldots, S$, $\{i, j\} \in E$ and $k \in K$ the variables f_{ijk}^s and f_{jik}^s representing the flow under scenario s of commodity k on the edge $\{i, j\}$ in direction from i to j and j to i, respectively. The flow of traffic is now determined by the following flow conservation constraints, stating that the net flow of a commodity into some IP POP should equal the net demand for the commodity from customers connected to that particular IP POP. Hence for each scenario $s \in \{1, \ldots, S\}$ we impose the constraints

$$\sum_{j:\{i,j\} \in E} f_{jik}^s - \sum_{j:\{i,j\} \in E} f_{ijk}^s = \sum_{j \in \overline{N}(i)} \sum_{g \in G(j)} D_{kg}^s y_{ig} \quad i \in V,\ k \in K. \tag{2.1b}$$

Remark 2.2. Note that the flow conservation constraints (2.1b) correspond to those of a standard multicommodity flow problem, and in particular that the possibility of free flow distribution is implicitly assumed. In other words, we implicitly assume that the flow of traffic between any pair of nodes may be divided arbitrarily among a number of different paths. This is not, however, in accordance with the facts of IP routing cf. the discussion in Holmberg and Yuan [9]. Still, computational results presented by Holmberg and Yuan show that the standard multicommodity flow constraints provide a reasonable approximation of a much more complex model for IP routing.

The following constraints concern the switch-capacity of the IP POPs. (Here, the amount of traffic switched by an IP POP is determined as the total flow out of the IP POP — that is, the sum of traffic terminating at customers connected to the IP POP and traffic sent on through the network.) Again we define for $i \in V$ and $s \in \{1, \ldots, S\}$ the surplus variable γ_i^s representing the shortage of switch-capacity of the IP POP in region i under scenario s. Hence for $s \in \{1, \ldots, S\}$ the constraints are formulated as

$$\sum_{h \in H(i)} N_h x_{ih} + \gamma_i^s \geq \sum_{j \in \overline{N}(i)} \sum_{g \in G(j)} d_g^s y_{ig} + \sum_{k \in K} \sum_{j:\{i,j\} \in E} f_{ijk}^s, \quad i \in V. \tag{2.1c}$$

For the transmission links in the distributed network we require not only that the total traffic in either direction on a link should not exceed the capacity of that link, but also that each of the two alternative connections from an IP POP in the distributed network to the rest of the network, have enough capacity to carry 60% of the total traffic into and out of the IP POP. Here we define for $i \in V_1$ and $s \in \{1, \ldots, S\}$ the surplus variables τ_i^s representing the shortage of capacity under scenario s on the two transmission links from the IP POP in region i to the rest of the network. Now, for $s \in \{1, \ldots, S\}$ the constraints are,

$$2(\tilde{v}_i z_{i1} + v_i) + \tilde{w}_i z_{i2} + w_i + C_1 z_{i3} + \tau_i^s \geq \sum_{k \in K} f_{ijk}^s, \quad i \in V_1, \ \{i,j\} \in E_1(i), \tag{2.1d}$$

$$2(\tilde{v}_i z_{i1} + v_i) + \tilde{w}_i z_{i2} + w_i + C_1 z_{i3} + \tau_i^s \geq \sum_{k \in K} f_{jik}^s, \quad i \in V_1, \ \{i,j\} \in E_1(i), \tag{2.1e}$$

$$2(\tilde{v}_i z_{i1} + v_i) + \tilde{w}_i z_{i2} + w_i + C_1 z_{i3} + \tau_i^s \geq 0.6 \sum_{k \in K} \sum_{j:\{i,j\} \in E_1(i)} f_{ijk}^s, \quad i \in V_1, \tag{2.1f}$$

$$2(\tilde{v}_i z_{i1} + v_i) + \tilde{w}_i z_{i2} + w_i + C_1 z_{i3} + \tau_i^s \geq 0.6 \sum_{k \in K} \sum_{j:\{i,j\} \in E_1(i)} f_{jik}^s, \quad i \in V_1. \tag{2.1g}$$

Remark 2.3. Note that since we are considering an IP network using optical transmission systems, each link in the network can carry flow in either direction and, more importantly, these flows do not interfere.

Finally, for the transmission links in the core network, the only capacity constraints are for each scenario $s \in \{1, \ldots, S\}$ that

$$\sum_{l=1}^{3} C_l u_{ijl} + \sigma_{ij}^s \geq \sum_{k \in K} f_{ijk}^s, \quad \{i,j\} \in E_2, \tag{2.1h}$$

$$\sum_{l=1}^{3} C_l u_{ijl} + \sigma_{ij}^s \geq \sum_{k \in K} f_{jik}^s, \quad \{i,j\} \in E_2, \tag{2.1i}$$

where σ_{ij}^s denotes the shortage of capacity on the transmission link $\{i,j\} \in E_2$ under scenario s.

2.5 A Two-Stage Formulation

As previously discussed, the decisions concerning network design must be made so as to minimize total cost incurred while ensuring that enough capacity is installed to accommodate any future demand scenario. The latter restriction is formulated implicitly using functions $\mathcal{Q}^s$, defined for each scenario $s \in \{1,\dots,S\}$ by

$$\begin{aligned}\mathcal{Q}^s(x,y,z,v,w,u) = \min & \sum_{i\in V}\big(\lambda_i^s + \gamma_i^s\big) + \sum_{i\in V_1}\tau_i^s + \sum_{\{i,j\}\in E_2}\sigma_{ij}^s \\ \text{s.t. } & (2.1a) - (2.1i), \\ & f^s \in \mathbb{R}_+^{2|E||K|},\ \lambda^s,\gamma^s \in \mathbb{R}_+^{|V|},\ \tau^s \in \mathbb{R}_+^{|V_1|},\ \sigma^s \in \mathbb{R}_+^{|E_2|}.\end{aligned} \tag{2.2}$$

Obviously, sufficient capacity to accommodate demand scenario $s \in \{1,\dots,S\}$ is installed if and only if the decisions concerning the network design are such that $\mathcal{Q}^s(x,y,z,v,w,u) = 0$. Hence the network design problem may be formulated as

$$\min \sum_{i\in V}\sum_{h\in H(i)} p_{ih}x_{ih} + \sum_{i\in V}\sum_{j\in\overline{N}(i)}\sum_{g\in G(j)} q_{ig}y_{ig}$$

$$+ \sum_{i\in V_1}\Big(\sum_{l=1}^{3} r_{il}z_{il} + a_i v_i + b_i w_i\Big) + \sum_{\{i,j\}\in E_2}\sum_{l=1}^{3} c_{ijl}u_{ijl} \tag{2.3a}$$

$$\text{s.t. } \sum_{h\in H(i)} x_{ih} = 1 \qquad i \in V \tag{2.3b}$$

$$x_{j0} \le x_{i0} \qquad i,j \in V_1,\ \{i,j\} \in E_1(i) \tag{2.3c}$$

$$\sum_{j\in N(i)} y_{jg} = x_{i0} \qquad i \in V_1,\ g \in G(i) \tag{2.3d}$$

$$y_{ig} = 1 - x_{i0} \qquad i \in V_1,\ g \in G(i) \tag{2.3e}$$

$$y_{ig} = 1 \qquad i \in V_2,\ g \in G(i) \tag{2.3f}$$

$$\sum_{l=1}^{3} z_{il} = 1 - x_{i0} \qquad i \in V_1 \tag{2.3g}$$

$$v_i \le Dz_{i1} \qquad i \in V_1 \tag{2.3h}$$

$$w_i \le Dz_{i2} \qquad i \in V_1 \tag{2.3i}$$

$$\sum_{l=1}^{3} u_{ijl} = 1 \qquad \{i,j\} \in E_2 \tag{2.3j}$$

$$\mathcal{Q}^s(x,y,z,v,w,u) = 0 \qquad s = 1,\dots,S \tag{2.3k}$$

$$x \in \mathbb{B}^X,\ y \in \mathbb{B}^Y,\ z \in \mathbb{B}^{3|V_1|},\ u \in \mathbb{B}^{3|E_2|},\ v \in \mathbb{Z}_+^{|V_1|},\ w \in \mathbb{R}_+^{|V_1|}. \tag{2.3l}$$

Here $X = \sum_{i\in V}|H(i)|$ and $Y = \sum_{i\in V}\sum_{j\in\overline{N}(i)}|G(j)|$. The objective function (2.3a) consists of four terms, corresponding to installment of IP POPs, connection of customers to the network, capacity expansion of transmission links in the distributed network, and capacity expansion of transmission links in the core network, respectively. According to (2.3b), one IP POP class is selected for each region, and (2.3c) ensures that an IP POP is only dismantled if any other IP POP using it as a transit node is also dismantled. If some IP POP in the distributed network is to be dismantled ($x_{\cdot 0} = 1$), the customers in the corresponding region should be connected to the network through an alternative IP POP. If, on the other hand, the IP POP is maintained ($x_{\cdot 0} = 0$), the customers in the corresponding region remain connected to the network through this particular IP POP. This is achieved by the constraints (2.3d) and (2.3e). For an IP POP in the core network, on the other hand, all customers in the corresponding region remain connected to the network through this particular IP POP cf. (2.3f). Next, a type of connection should be selected from each IP POP in the distributed network that is maintained ($x_{\cdot 0} = 0$) cf. (2.3g). Also, for each transmission link connecting an IP POP in the distributed network to the rest of the network, (2.3h) and (2.3i) ensure that capacity in the form of E1 circuits or ATM PVCs may be expanded if and only if this particular type of capacity is selected. (Note that we use the maximum aggregate traffic demand D as a "big-M coefficient".) Next, the constraint (2.3j) requires a type of connection to be selected for each transmission link connecting a pair of IP POPs in the core network. Finally, (2.3k) ensures that enough capacity is installed to accommodate all demand scenarios.

Remark 2.4. Problem (2.3) involves only the first-stage decisions concerning network design, that must be taken without certain knowledge about future demand. Problem (2.2), on the other hand, is the second-stage problem, involving only the routing of traffic, that is clearly postponed until the actual outcome of demand is observed.

Remark 2.5. Clearly, no customers should be connected to the network through an IP POP that is pointed out for dismantling. Hence any feasible solution of the network design problem should satisfy the constraints

$$y_{ig} \leq 1 - x_{i0} \quad i \in V_1,\ g \in \textstyle\bigcup_{j \in \overline{N}(i)} G(j).$$

These constraints are implied, however, by e.g. the constraints (2.1a) and (2.3k), and hence we do not include them in the formulation.

Remark 2.6. The second-stage constraint (2.1c) ensures that no flow of traffic can occur out of an IP POP that is pointed out for dismantling. Hence if the IP POP in some region is to be dismantled, the constraint (2.1b) ensures that no flow of traffic can occur into this IP POP since no customer groups are connected to it cf. Remark 2.5. Thus, no flow of traffic can occur into or out of IP POPs that are to be dismantled.

3 Solution Procedure

To ease the exposition we find it convenient in the following to simplify the notation, writing the aggregate first-stage solution vector (x, y, z, u, v, w) simply as $\tilde{x}$, and for each scenario $s = 1, \ldots, S$ writing the vector of second-stage surplus variables $(\lambda^s, \gamma^s, \tau^s, \sigma^s)$ simply as ρ^s. Also we will consider the second-stage problem (2.2) only in a conceptual form, written for $\tilde{x} \in \mathbb{R}^{n_1}$ and $s = 1, \ldots, S$ as

$$\mathcal{Q}^s(\tilde{x}) = \min\big\{ e\rho^s \mid W^s f^s + \tilde{W}^s \rho^s \geq h^s - T^s \tilde{x},\ f^s \in \mathbb{R}_+^{n_2},\ \rho^s \in \mathbb{R}_+^{n_2'} \big\}, \tag{3.1}$$

where $e = (1, \ldots, 1) \in \mathbb{R}^{n_2'}$, and $h^s \in \mathbb{R}^{m_2}$, $W^s \in \mathbb{R}^{m_2 \times n_2}$, $\tilde{W}^s \in \mathbb{R}^{m_2 \times n_2'}$, and $T^s \in \mathbb{R}^{m_2 \times n_1}$ represent the capacity constraints (2.1a)-(2.1i). Likewise, the first-stage problem (2.3) is considered only in the conceptual form,

$$
\begin{aligned}
\min\ & c\tilde{x} \\
\text{s.t. } & A\tilde{x} = b, \\
& \mathcal{Q}^s(\tilde{x}) = 0, \quad s = 1, \ldots, S, \\
& \tilde{x} \in \tilde{X},
\end{aligned}
\tag{3.2}
$$

where $c \in \mathbb{R}^{n_1}$ represents the first-stage objective (2.3a), $b \in \mathbb{R}^{m_1}$ and $A \in \mathbb{R}^{m_1 \times n_1}$ represent the first-stage constraints (2.3b)-(2.3k), and $\tilde{X} \subseteq \mathbb{R}^{n_1}_{+}$ is a subset, restricting the appropriate components of $\tilde{x}$ to be either binary, integer or real numbers.

The fundamental idea in the cutting plane method for problem (3.2) presented below, is to relax the constraints $\mathcal{Q}^s(\tilde{x}) = 0$, $s = 1, \ldots, S$, and iteratively re-enforce them by means of so-called feasibility cuts. Hence, we start with a relaxation of problem (3.2), referred to as the master problem, in which the constraints $\mathcal{Q}^s(\tilde{x}) = 0$, $s = 1, \ldots, S$, have been removed. Given a solution $\tilde{x}^\nu \in \mathbb{R}^{n_1}$ of the master problem in some iteration ν, the feasibility cuts are derived from the second-stage problem (3.1) with $\tilde{x} = \tilde{x}^\nu$. Specifically, we consider the corresponding dual problem, defined for $s = 1, \ldots, S$ by

$$
\mathcal{Q}^s(\tilde{x}) = \max\bigl\{(h^s - T^s\tilde{x}^\nu)\pi^s \mid \pi^s W^s \leq 0,\ \pi^s \tilde{W}^s \leq e,\ \pi^s \in \mathbb{R}^{m_2}_{+}\bigr\}. \tag{3.3}
$$

Obviously, problems (3.1) and (3.3) are both feasible, and hence they are both solvable and their optimal values are identical and clearly non-negative. In particular, if $\tilde{x}^\nu \in \mathbb{R}^{n_1}$ is such that $\mathcal{Q}^s(\tilde{x}^\nu) > 0$ for some scenario $s \in \{1, \ldots, S\}$, then a feasible solution $\pi^{s,\nu}$ of the dual problem (3.3) exists such that $(h^s - T^s\tilde{x}^\nu)\pi^{s,\nu} > 0$. Moreover, since $\pi^{s,\nu}$ is feasible for the dual problem (3.3), we see that for all $\tilde{x} \in \mathbb{R}^{n_1}$ with $\mathcal{Q}^s(\tilde{x}) = 0$ we have

$$
(h^s - T^s\tilde{x})\pi^{s,\nu} \leq 0. \tag{3.4}
$$

The constraint (3.4) is referred to as a feasibility cut, and as described above it can be used to cut off the current solution of the master problem $\tilde{x}^\nu \in \mathbb{R}^{n_1}$ whenever $\mathcal{Q}^s(\tilde{x}^\nu) > 0$ for some scenario $s \in \{1, \ldots, S\}$.

The algorithm progresses by sequentially solving a master problem and adding violated feasibility cuts that are generated through the solution of subproblems (3.1) and (3.3).

Algorithm 1

Step 1 (*Initialization*) Set $\nu = 0$, and let the current master problem be defined by $\min\bigl\{c\tilde{x} \mid A\tilde{x} = b,\ \tilde{x} \in \tilde{X}\bigr\}$.

Step 2 (*Solve master problem*) Set $\nu = \nu + 1$. Solve the current master problem and let $\tilde{x}^\nu$ be an optimal solution vector.

Step 3 (*Solve subproblems*) For each scenario $s = 1, \ldots, S$ solve the second-stage problem (3.1) with $\tilde{x} = \tilde{x}^\nu$, and let $\pi^{s,\nu}$ be a corresponding optimal dual solution. If $\pi^{s,\nu}(h^s - T^s x^\nu) > 0$ for some $s \in \{1, \ldots, S\}$, add a feasibility cut (3.4) to the master problem and return to Step 2. Otherwise, stop; the current solution $\tilde{x}^\nu$ is optimal.

Proposition 1. *If problem (3.2) is feasible, then Algorithm 1 terminates with an optimal solution of the problem in a finite number of iterations.*

Proof. Finite convergence is an immediate consequence of the fact that only a finite number of extreme points of the feasible region in (3.3) exist. □

Remark 3.1. The requirement of feasibility of problem (3.1) in Proposition 1 is certainly not unreasonable, since it should always be possible to install sufficient capacity to accommodate demand. Clearly, though, problem (2.3) becomes infeasible if demand rises beyond a certain level. Therefore, in such a case, the formulation is not appropriate, and one may have to allow for the placement of new IP POPs as well as for the installment of several facilities (STM-1, STM-4 or STM-16) on connections in the future core network. For the IP network of TDC considered in this case study, however, the formulation presented here was found appropriate.

Remark 3.2. Note that Algorithm 1 employs a mixed-integer programming formulation of the master problem in each iteration. The generation of feasibility cuts, however, may as well be carried out for fractional solutions cf. our discussion above, and hence it seems natural to assume that it is not worthwhile to put a lot of effort into finding integral first-stage solutions in early iterations of the algorithm. In fact, Riis and Andersen [15] formulated the capacitated netweork design problem as a two-stage stochastic program with linear recourse, and proposed a solution method that is similar in vein to Algorithm 1, but in which integer requirements are initially removed in the master problem. This algorithm then proceeds to restore integrality and feasibility simultaneously through a branch-and-cut scheme, with feasibility cuts being generated at all nodes of the branching tree. Furthermore, Albareda-Sambola, van der Vlerk and Ferńandez [1], compared different versions of a similar algorithm for a class of stochastic generalized assignment problems, and concluded that such a branch-and-cut scheme performed superior to a branch-first-cut-second scheme such as Algorithm 1. We did in fact also try a branch-and-cut algorithm for problem (3.2). It turned out, however, that for the particular instance considered here, the major effort lies in solving the second-stage problems, whereas solving even a mixed-integer formulation of the master problem is relatively easily done with the CPLEX Mixed Integer optimizer. This means that generating cuts via the solution of the second-stage problems throughout the branching process is simply too time consuming, and very little movement in the lower bound was observed for the branch-and-cut algorithm. Hence for this problem, the branch-first-cut-second scheme actually proved superior, and therefore this version of the algorithm was used for the computational experiments discussed in the following section.

4 Computational Experiments

The algorithm described in the previous section was implemented in C++ using procedures from the callable library of CPLEX 6.6. In particular, the mixed-integer master problem was solved with the CPLEX Mixed Integer optimizer cf. Remark 3.2. A series of computational experiments was performed on the IP network of TDC, the largest Danish network operator. In this section we discuss the application of the model presented in Section 2, and present results of our computational experiments.

4.1 Application of the Model

Let us first consider the IP network of TDC. Here the core network consists of 39 IP POPs interconnected by a total of 70 transmission links. The distributed network, on the other hand, consists of 155 IP POPs, most of which are connected to the rest of the network by two alternatively conveyed links of equal type and capacity as assumed in the model. Some exceptions from the idealized network structure of the model presented in Section 2 had to be dealt with, however. First, for some IP POPs in the distributed network, the two alternatively conveyed links, connecting the IP POP to the rest of the network, does not presently have equal capacities. In these cases, we simply used the average of the two as the existing capacity for the model input. Second, for specific reasons, a few IP POPs in the distributed network actually have an extra STM-1 link to the rest of the network. These extra links were included in the model, but no upgrading of the connections were allowed. Finally, a few IP POPs in the distributed network are connected to the rest of the network through "hoops" of two IP POPs. This is best illustrated by a small example.

Example 2. Figure 3 illustrates a "hoop", connecting IP POPs 1 and 2 to the rest of the network through IP POPs A and B.

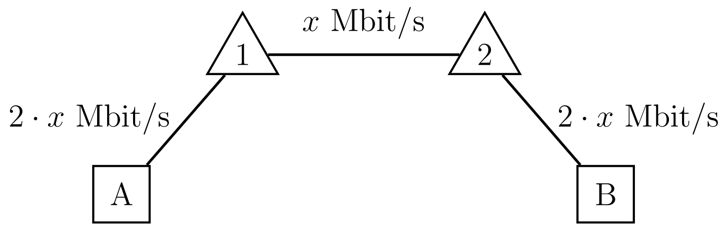

Figure 3: Illustration of a "hoop".

Clearly, it is possible to accurately represent such a hoop within the integer programming formulation of the model. We did not find the improved accuracy of such a formulation sufficient to justify the increased model complexity, though, and hence we simply chose to treat IP POPs such as 1 and 2 in Figure 3 as if they both had a link with capacity x to IP POP A and a link with capacity x to IP POP B.

A total of seven different IP POP classes were defined (including the class '0'), with at most five potential IP POP classes available for selection in any particular region. Also, for all regions corresponding to IP POPs that may be dismantled, the customers were divided in up to four groups, and up to three potential alternative IP POPs for the customers were specified. (For IP POPs that are not eligible for dismantling, it obviously does not make sense to divide customers into more than one group, or to specify alternative IP POPs.)

All in all, we ended up with a two-stage stochastic program with recourse, containing in the first stage a total of 1960 variables, most of which are binary, and 1137 constraints at initialization. Moreover, when the algorithm progresses, the number of constraints increases as cuts are imposed to re-inforce (2.3k). Clearly, though, the special structure of constraints such as e.g. (2.3e) and (2.3f) allowed CPLEX MIP Presolve to reduce the size of the problem considerably, removing a priori a large number of first-stage variables and constraints. Given a first-stage solution and a particular scenario, the second-stage problem, on the other hand, is a linear programming problem with 206737 continuous variables and 53268 constraints.

4.2 Generation of Scenarios

The only available demand input for the model was the current groupwise demand for customer connections, denoted here by L_g, $g \in G$, and the current regionwise demand for IP traffic, expressed as the total amount of IP traffic terminating at each IP POP and denoted here by T_i, $i \in V$. Using this data we had to estimate the groupwise demand for IP traffic, and generate a number of future demand scenarios. This was done as follows. First of all, for $s = 1, \dots, S$ the future groupwise demand for customer connections under scenario s was calculated as

$$L_g^s = \mu^s \cdot \rho_g^s \cdot L_g, \quad g \in G,$$

where μ^s is a parameter reflecting the average growth in demand for customer connections, and ρ_g^s, $g \in G$, are parameters reflecting regional fluctuations from this average growth. Now, to calculate an estimate of the groupwise demand for IP traffic, we used the estimated future demand for customer connections to split the regionwise demand T_i, $i \in V$, among groups. Hence, for $s = 1, \dots, S$ the future groupwise demand for IP traffic under scenario s, expressed as the total volume of IP traffic terminating at each group, was calculated as

$$d_g^s = \lambda^s \cdot \gamma_g^s \cdot \frac{L_g^s}{\sum_{g \in G(i)} L_g^s} \cdot T_i, \quad i \in V,\ g \in G(i),$$

where λ^s is a parameter reflecting the average growth in demand for IP traffic, and γ_g^s, $g \in G$, are parameters reflecting regional fluctuations from this average growth. The growth factors were all generated by random sampling from appropriate uniform distributions.

Remark 4.1. To capture the correlation between growth in demand for customer connections and growth in demand for IP traffic, we actually independently generated parameters μ^s, $\tilde{\lambda}^s$, ρ_g^s and $\tilde{\gamma}_g^s$ for all $g \in G$ and $s = 1, \dots, S$, and then defined $\lambda^s = \mu^s \cdot \tilde{\lambda}^s$ and $\gamma_g^s = \rho_g^s \cdot \tilde{\gamma}_g^s$ for $g \in G$ and $s = 1, \dots, S$.

Finally, we used an aggregated formulation of the commodities, defining a commodity for each customer group, i.e. $K = G$, so that commodity $k \in K$ corresponds to IP traffic originating at group k. The commodity demand was then calculated by gravitation, using the estimates of the future volume of IP traffic terminating at each group. Hence for $s = 1, \dots, S$ the commodity demand for IP traffic was calculated as

$$D_{kg}^s = \frac{d_g^s \cdot d_k^s}{\sum_{g' \in G} d_{g'}^s}, \quad k \in K,\ g \in G \setminus \{k\},$$

and

$$D_{kk}^s = -\sum_{g \in G \setminus \{k\}} D_{kg}^s, \quad k \in K.$$

4.3 Implementational Details

As previously pointed out, for the IP network of TDC the main effort in solving problem (2.3) using Algorithm 1, lay in solving the second-stage problem (2.2), whereas solving the master problem (3.2) was relatively easy. Furthermore, since no capacity constraints are initially present in the master problem, a direct application of Algorithm 1 as it was presented in Section 3 would require a large number of feasibility cuts to be imposed to properly reflect the capacity requirements in the second stage. In fact we tried a direct application of Algorithm 1, and observed that a vast amount of time was spent solving second-stage problems to generate feasibility cuts, achieving only very little movement in the optimal value of the master problem. Therefore, to improve performance of the algorithm we determined a large number of capacity constraints that could be generated a priori without solving any second-stage problems. First of all, it is obvious that the constraints concerning the customer-capacity of each individual IP POP can be used directly in the master problem, since they are independent of the routing of traffic in the second stage cf. (2.1a). Hence we used the constraints,

$$\sum_{h\in H(i)} M_h x_{ih} \geq \sum_{j\in\overline{N}(i)} \sum_{g\in G(j)} L_g^s y_{ig}, \quad i\in V,\ s=1,\ldots,S. \tag{4.1}$$

The constraints concerning the switch-capacity of each individual IP POP, on the other hand, clearly depend on the routing of traffic in the second stage cf. (2.1c), and hence they can not be used directly in the master problem. Instead, we considered the following alternative constraints,

$$\sum_{h\in H(i)} N_h x_{ih} \geq \sum_{j\in\overline{N}(i)} \sum_{g\in G(j)} \left(d_g^s - \sum_{j'\in\overline{N}(i)} \sum_{g'\in G(j')} D_{gg'}^s \right) y_{ig}, \quad i\in V,\ s=1,\ldots,S. \tag{4.2}$$

To see that these are in fact valid inequalities, we note that for any $s\in\{1,\ldots,S\}$, $i\in V$, $j\in\overline{N}(i)$, and $g\in G(j)$ the first term in the parentheses on the right-hand side, $d_g^s\geq 0$, gives the total amount of traffic that terminates at group g under scenario s, whereas the second term, $-\sum_{j'\in\overline{N}(i)}\sum_{g'\in G(j')} D_{gg'}^s \geq 0$, gives the total amount of traffic originating at group g under scenario s (i.e. $-D_{gg}^s$) minus the part of this traffic that terminates at groups that may be connected to the network through the IP POP in region i $\big($i.e. $\sum_{j'\in\overline{N}(i)}\sum_{g'\in G(j')\setminus\{g\}} D_{gg'}^s\big)$. Hence, for any possible allocation of customer groups to IP POPs, the right-hand side of (4.2) provides a lower bound on the total amount of traffic that must be switched by the IP POP in region $i\in V$ under scenario $s\in\{1,\ldots,S\}$, and hence it is a valid inequality. In general, the lower bound on the required switch-capacity provided by (4.2) is not tight, but since the switch-capacities of the different IP POP classes are generally far apart, the constraints turned out to be quite effective.

To generate cuts for the required capacity on links in the distributed network, our starting point was the constraints (2.1f) and (2.1g), stating that each of the two alternative connections from an IP POP in the distributed network to the rest of the network, is required to have enough capacity to carry 60% of the total traffic into and out of the IP POP. Here we used the constraints,

$$\begin{aligned} 2(\tilde{v}_i z_{i1} + v_i) + \tilde{w}_i z_{i2} + w_i + C_1 z_{i3} \geq 0.6 \sum_{j\in\overline{N}(i)} \sum_{g\in G(j)} \left(\sum_{j'\in V_2} \sum_{g'\in G(j')} D_{gg'}^s \right) y_{ig}, \\ i\in V_1,\ s=1,\ldots,S. \end{aligned} \tag{4.3}$$

$$2(\tilde{v}_i z_{i1} + v_i) + \tilde{w}_i z_{i2} + w_i + C_1 z_{i3} \geq 0.6 \sum_{j \in \overline{N}(i)} \sum_{g \in G(j)} \left(\sum_{j' \in V_2} \sum_{g' \in G(j')} D^s_{g'g} \right) y_{ig}, \qquad i \in V_1,\ s = 1, \dots, S. \tag{4.4}$$

To see that (4.3) is a valid inequality, we note that for any $s \in \{1, \dots, S\}$, $i \in V$, $j \in \overline{N}(i)$, and $g \in G(j)$, the term in the parentheses on the right-hand side gives the total amount of traffic that must be routed from group g to a customer group in a region in the core network under scenario s. Hence for any $i \in V$ and $s \in \{1, \dots, S\}$, the right-hand side in (4.3) is clearly a lower bound on the amount of traffic from region i to regions in the core network under scenario s and hence the inequality is valid. Obviously, a similar observation goes for (4.4) only with the direction of traffic reversed. Again we note, that the lower bounds provided by (4.3) and (4.4) are obviously not tight, but since traffic between a region $i \in V_1$ and other regions in the distributed network is typically negligible compared to the traffic between region i and the core network, the inequalities turned out quite useful.

Finally, to generate cuts for the required capacity on links in the core network, we used a generalization of the well-known cutset inequalities, employed for example for the capacitated network design problem by e.g. Bienstock and Günlük [2], Günlük [8], and Riis and Andersen [15]. To this end, for some $U \subseteq V_2$ we consider the partition $\pi = (U, V_2 \setminus U)$ of V_2, and let $E_\pi = \big\{\{i,j\} \in E_2 : |\{i,j\} \cap U| = 1\big\}$ be the corresponding cutset. Also, we recall that only one customer group was defined for all regions corresponding to IP POPs in the core network, and hence for ease of notation we let $G(i) = \{g_i\}$ for $i \in V_2$. Now, we used the following constraints,

$$\sum_{\{i,j\} \in E_\pi} \sum_{l=1}^{3} C_l u_{ijl} \geq \max_{s \in \{1, \dots, S\}} \max \left\{ \sum_{i \in U} \sum_{j \in V_2 \setminus U} D^s_{g_i g_j},\ \sum_{i \in U} \sum_{j \in V_2 \setminus U} D^s_{g_j g_i} \right\}. \tag{4.5}$$

It is easily seen that this is in fact a valid inequality since the right-hand side of (4.5) is a lower bound on the amount of traffic that must be routed across the cutset E_π. Furthermore, cf. our discussion of the inequalities (4.3) and (4.4) above, we note that the traffic between regions in the core network and regions in the distributed network is typically negligible compared to the interregional traffic in the core network, and hence the inequalities proved quite useful.

Remark 4.2. Note that since the commodity demands for the IP network of TDC was generated by a gravitational model, the traffic matrix was in fact symmetric, and hence in practice we did not have to consider traffic in both directions for the link-capacity constraints as stated here by (4.3), (4.4), and (4.5).

Obviously, a potentially large number of constraints may be generated a priori from (4.1)-(4.5). This is true in particular for the generalized cutset inequalities (4.5), and hence we chose to consider only those cutsets corresponding to subsets $U \subseteq V_2$ consisting of up to three IP POPs. Moreover, to control the size of the master problem, we chose to generate all cuts from (4.1)-(4.5) at initialization of the algorithm and store them in a cutpool. Then, in each iteration of the algorithm, before the second-stage problems are solved to possibly generate violated feasibility cuts, this cutpool is scanned to search for violated capacity constraints. If any violated constraints are found they are included in the master problem (at most 10 at a time), and the master problem is re-solved.

4.4 Computational Results

A series of computational experiments were performed on the IP network of TDC. We generated instances of the problem with 1, 5, 10, 50, and 100 scenarios, and solved the problems using Algorithm 1 as described in the previous section. The instance with only one scenario was generated by replacing all random parameters by their expected values, and hence it will be referred to as the expected value problem (EVP). At termination of each run we recorded the number of iterations performed, the total number of generated cuts, the number of cuts in the master problem (referred to as *active cuts*), and the CPU time spent by the procedure. Results are reported in Table 1.

Table 1: Computational Results

S	Iterations	Total cuts	Active cuts	CPU time
(EVP) 1	16	403	152	2:16
5	15	1637	714	4:28
10	24	3245	1506	7:56
50	21	15725	2027	26:15
100	21	31322	8026	48:31

The optimal solution of the two instances with 50 and 100 scenarios, respectively, suggested dismantling of the same particular 17 IP POPs. The optimal solution of the instances with 5 and 10 scenarios only disaggreed with this suggestion for one and three IP POPs, respectively. The optimal solution of the expected value problem, on the other hand, suggested dismantling of 11 IP POPs, one of which was not suggested for dismantling in any of the other solutions. To investigate the effect of using a stochastic programming model with multiple scenarios, we fixed the dismantling of IP POPs suggested by the solution of the expected value problem, and subsequently solved the stochastic programming problem with the same 100 scenarios as before. The resulting total cost turned out to be 3.5% larger than the minimum cost determined in the previous run. Hence, given the size of the total installment cost, the saving obtained by solving the stochastic programming problem rather than the expected value problem, is considerable.

5 Conclusions

In this paper we have set up a model to point out IP POPs in the internet protocol network of TDC that are eligible for dismantling. The model takes into account the cost of upgrading IP POPs, the cost of transferring customers to alternative IP POPs, and the cost of expanding capacity on transmission links in the network. In order to take due account of the inherent uncertainty involved in the assessment of future demand, the problem was formulated as a two-stage stochastic program. This problem was solved by an L-shaped algorithm, which was made practicable by the inclusion of a large number of capacity constraints, that can be generated a priori without considering the actual routing of traffic. The algorithm was implemented in C++ and a series of computational experiments were carried out. The experiments demonstrate that the solution procedure is practicable, and indicate that a superior solution is obtained by solving the stochastic programming problem rather than basing decisions simply on the expected value of random parameters.

References

[1] M. Albareda-Sambola, M.H. van der Vlerk, and E. Fernández. Exact solutions to a class of stochastic generalized assignment problems. SOM Research Report 02A11, University of Groningen, Department of Econometrics & OR, 2002.

[2] D. Bienstock and O. Günlük. Capacitated network design. Polyhedral structure and computation. *INFORMS Journal on Computing*, 8(3):243–259, 1996.

[3] J.R. Birge and F.V. Louveaux. *Introduction to Stochastic Programming.* Springer-Verlag, New York, 1997.

[4] S. Challinor. An introduction to IP networks. *BT Technology Journal*, 18(3), 2000.

[5] M.A.H. Dempster, E.A. Medova, and R.T. Thompson. A stochastic programming approach to network planning. *Teletraffic Contributions for the Information Age. Proceedings of the 15th International Teletraffic Congress - ITC 15*, 1:329–339, 1997.

[6] J. Dupačová. Stochastic programming with incomplete information: A survey of results on postoptimization and sensitivity analysis. *Optimization*, 18:507–532, 1987.

[7] J. Dupačová. Stability and sensitivity analysis for stochastic programming. *Annals of Operations Research*, 27:115–142, 1990.

[8] O. Günlük. A branch-and-cut algorithm for capacitated network design. *Mathematical Programming*, 86:17–39, 1999.

[9] K. Holmberg and D. Yuan. Optimization of internet protocol network design and routing. Research Report LiTH-MAT-R-2001-07, Linkoping Institute of Technology, Department of Mathematics, 2001.

[10] P. Kall. On approximations and stability in stochastic programming. In J. Guddat, H.Th. Jongen, B. Kummer, and F. Nožička, editors, *Parametric Optimization and Related Topics*, pages 387–407. Akademie Verlag, Berlin, 1987.

[11] P. Kall and S.W. Wallace. *Stochastic Programming.* John Wiley and Sons, Chichester, UK, 1994.

[12] E.A. Medova. Chance-constrained stochastic programming for integrated services network management. *Annals of Operations Research*, 81:213–229, 1998.

[13] A. Prékopa. *Stochastic Programming.* Kluwer Academic Publishers, Dordrecht, Netherlands, 1995.

[14] M. Riis and K.A. Andersen. Multiperiod capacity expansion of a telecommunications connection with uncertain demand. *Computers & Operations Research* (to appear).

[15] M. Riis and K.A. Andersen. Capacitated network design with uncertain demand. *INFORMS Journal on Computing*, 14(3):247–260, 2002.

[16] M. Riis and J. Lodahl. A bicriteria stochastic programming model for capacity expansion in telecommunications. *Mathematical Methods of Operations Research*, 56(1):83–100, 2002.

[17] S.M. Robinson and R.J-B Wets. Stability in two-stage stochastic programming. *SIAM Journal on Control and Optimization*, 25:1409–1416, 1987.

[18] W. Römisch and R. Schultz. Distribution sensitivity in stochastic programming. *Mathematical Programming*, 50:197–226, 1991.

[19] W. Römisch and R. Schultz. Stability analysis for stochastic programs. *Annals of Operations Research*, 30:241–266, 1991.

[20] W. Römisch and R. Schultz. Stability of solutions for stochastic programs with complete recourse. *Mathematics of Operations Research*, 18:590–609, 1993.

[21] S. Sen, R.D. Doverspike, and S. Cosares. Network planning with random demand. *Telecommunication Systems*, 3:11–30, 1994.

[22] A. Shapiro. Quantitative stability in stochastic programming. *Mathematical Programming*, 67:99–108, 1994.

[23] R.M. Van Slyke and R.J-B Wets. L-shaped linear programs with applications to optimal control and stochastic linear programming. *SIAM Journal of Applied Mathematics*, 17:638–663, 1969.

[24] J. Wang. Distribution sensitivity analysis for stochastic programs with complete recourse. *Mathematical Programming*, 31:286–297, 1985.