

An Integer Programming Model for Multi-Layer Network Design

Sebastian Orłowski and Roland Wessäly*

November 2004

Abstract

We present an integer linear programming model for the design of multi-layer telecommunication networks which are based on connection-oriented routing protocols. The formulation integrates hardware, capacity, routing, and grooming decisions in *any* number of network layers. Practical hardware restrictions and cost can accurately be taken into account.

Keywords: multi-layer network design, integer programming, routing, hardware configuration

MSC classification (2000): 90C57, 90C11, 68M10

1 Introduction

In this article, we present an integer linear programming model for the *integrated* cost optimization of multi-layer telecommunication networks, covering

- routing over an arbitrary number of network layers,
- hardware configuration in all layers,
- topology and link configuration decisions in all layers, and
- end-to-end survivability.

The formulation can accurately reflect both the cost and the restrictions associated with the hardware present in multi-layer networks (e.g., routers, switches, multiplexers, cross-connects, repeaters, interface cards, ...), without being limited to specific technologies. Furthermore, the proposed routing model is independent of a particular technology and can be applied to all technologies based on connection-oriented routing protocols (explicit end-to-end paths), e.g., for MPLS-, ATM-, SDH-, ODU-, and optical networks. The formulation leads to a valid lower bound on the optimal network cost.

Over the last decade, many mathematical programming models have been developed for planning and optimization of single-layer networks. These models range from pure topology planning over integrated capacity and routing decisions for the failureless state to survivable network design using various protection and restoration mechanisms [1–13]. Sophisticated mathematical optimization techniques have been suggested, like branch-and-cut algorithms based on polyhedral methods, or column generation approaches to deal with an exponential number of routing paths. Using these methodologies, solutions with good quality guarantees can today be computed for practical single-layer network planning problems.

Multi-layer networks are more complex for several reasons. First, hardware of different technologies has to be taken into account. Second, a nested routing over several protocol layers has to be determined, where paths in a server layer are embedded into paths in a client layer as so-called *logical links*; this leads to paths in paths in paths. . . . Furthermore, traffic is routed in different granularities at different layers. All these issues make multi-layer planning problems hard to deal with both operationally and computationally.

To cope with this complexity, a common approach in practice is the subsequent planning of the different network layers, where the link capacities of one layer serve as communication demands for the next one.

*Konrad-Zuse-Zentrum für Informationstechnik Berlin, Takustr. 7, D-14195 Berlin, Germany, {orłowski,wessaely}@zib.de

This approach simplifies the planning process significantly, but does not take the interdependencies between decisions in the different layers into account. As serious consequences, (i) end-to-end survivability of communications demands is hard to implement, (ii) the end-to-end routing of a communication demand might multiply use a single physical resource such as an optical fiber, (iii) even if each single-layer problem can be solved optimally, the overall design may be far from being optimal. These issues are further discussed in Section 2.4.5.

The problems with the sequential approach strongly suggest an *integrated* planning of topology, hardware, capacities, routing, and grooming over several network layers. It is difficult to exactly model *all* practical requirements in a network with more than two layers, including all technology- and protocol-dependent restrictions and features. In fact, the most difficult part is not so much the hardware, but finding an exact model for a routing over several technologies and grooming in more than two layers.

All existing models for integrated multi-layer network design simplify the practical planning problem to a great extent, for instance by ignoring all hardware cost and restrictions, or by assuming a continuous routing which does not take the routing granularities at the different network layers into account (see Section 4.1). In this paper, we present an integer linear programming model which describes major (technology-independent) parts of the problem, while still being computationally tractable. It comprises a detailed description of hardware in different technologies, and it provides a generic routing formulation for multi-layer networks with any number of layers which is a good relaxation of practical routing constraints.

Some practical routing requirements whose exact modeling would make the formulation too complex can be incorporated by means of an oracle which tests a solution for feasibility with respect to missing constraints. Depending on the exact type of requirements, such an oracle could either just accept or reject a solution (maybe providing further information such as a cutting plane), or it could heuristically make a solution feasible with respect to missing restrictions. In particular, this approach

- leads to a lower bound on the optimal network cost and thus to a quality guarantee for solutions,
- is generic enough to be usable in different planning contexts,
- is specific enough to provide a good relaxation,
- can incorporate technology-specific restrictions in a flexible way, and
- has chances to be computationally tractable.

The structure of this article is as follows: in Section 2, the practical background of the multi-layer network planning problem is explained. Our model for the design of multi-layer networks is introduced and discussed both from a mathematical and a practical point of view in Section 3. In Section 4, our formulation is compared to previous approaches on multi-layer network planning. After a sketch of a possible algorithmic approach in Section 5, we conclude with Section 6.

2 Practical background

Modern communication networks are assembled by a vast variety of different pieces of equipment and protocols. A concise overview of all possible options to plug these together, and to plan, build and operate a complete communication network, is beyond the scope of this article. Instead, the main purpose of this section is to introduce the basic concepts and principles of multi-layer network planning problems from a practical perspective. Starting with a general overview on technologies of a communication network and an introductory example, the concepts of multiplexing, grooming, and network layers are explained. This is followed by a description of the hardware which is needed for the interworking of different technologies. As a step towards an abstract formulation of multi-layer network planning, a list of practical planning requirements which should be covered by a realistic mathematical model is presented and discussed.

2.1 Technologies of a communication network

Communication networks are composed of many subnetworks, which are stacked one on top of the other, as illustrated in Figure 1. Each network in the stack has its own technology, serves its own purposes, has its own devices, its own protocols, and is able to communicate with some (but usually not all) other subnetworks. Among others, some technological scenarios established today by network operators are

(i) ATM over SDH, (ii) MPLS over SDH, and (iii) MPLS/ATM over SDH over WDM, where *A over B* means that traffic which seems to be routed in technology A is actually routed through technology B. An additional scenario, where MPLS is directly routed over WDM, is currently in the development and standardization phase.

MPLS: Multi-Protocol Label Switching is used to route IP packets in a connection-oriented way using so-called Label-Switched Paths (LSPs).

ATM: Asynchronous Transfer Mode uses Virtual Paths (VPs) to provide Quality of Service in packet-oriented data networks, ranging from constant bit rate (e.g., video) to best-effort (e.g., e-mail) services.

SDH: Synchronous Digital Hierarchy (equivalent to SONET) uses standardized bandwidths, called Virtual Containers (VC-Ns), to transport MPLS-, ATM-, or other traffic.

WDM: Wavelength-Division-Multiplexing combines many lightpaths with different wavelengths into a single optical fiber to provide pipes with high capacity for MPLS-, ATM-, or SDH-demands.

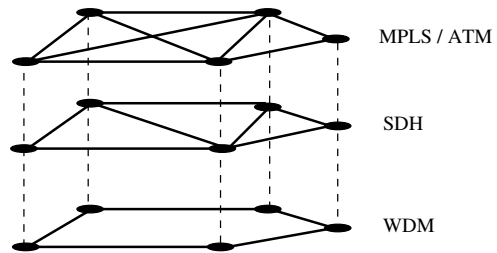


Figure 1: An MPLS/ATM over SDH over WDM network.

2.2 Routing in a communication network

Suppose Alice wants to send an e-mail to Bob. On its way, the e-mail passes through different technologies and protocols. At Alice's computer, the e-mail is wrapped into one or more IP packets. This provides a unified envelope, regardless of whether the e-mail consists of text, images, binary data, or a mixture of these. After Alice has established a modem connection to her local Internet Service Provider (ISP), these packets are sent through the regular telephone line, which is based on some other technology, such as ATM. The ISP encapsulates the ATM data into an SDH envelope and sends it to a nationwide backbone network. In the backbone, the SDH data is transmitted via an optical signal through optical fibers with high transmission capacities. At Bob's side of the backbone, the SDH signal is extracted from the optical one, the SDH and ATM envelope are removed one after another, and Bob finally extracts the e-mail from the IP packets. Parts of this wrapping and unwrapping procedure may even happen repeatedly on the path from Alice to Bob if Alice's ISP sends the data to Bob's one via several third-party ISPs.

2.2.1 Multiplexing and Grooming

At several nodes in the network, different data streams are combined into bigger units. For instance, Alice's ISP does not send every ATM packet individually to the backbone, but combines many of them into one SDH signal with coarser granularity which is sent to the backbone. This process is called *multiplexing*; the opposite process of extracting small granularity signals from a coarser one is known as *demultiplexing*.

Multiplexing is not only done at the border between two technologies, but can also happen within a technology. For instance, SDH provides so-called *Virtual Containers*, each of them corresponding to a specified bandwidth. MPLS or ATM data streams can be embedded into some of these containers, which are in turn multiplexed into other containers. Figure 2 illustrates this at the example of an ATM over SDH over WDM network: for instance, an ATM data stream of 2 Mbit/s can be embedded into a VC-12 container (2 Mbit/s). In a second step, 21 of these are combined into a VC-3 container (45 Mbit/s), three of which are multiplexed into one VC-4 container (140 Mbit/s), and so on. Finally, the data stream is multiplexed into a WDM wavelength with 2.5 Gbit/s.

Between the two nodes where traffic is multiplexed and demultiplexed, respectively, the aggregated data streams are routed together. This leads to the concept of *grooming paths*. A grooming path is similar to a highway without intermediate access or exit: all signals which are multiplexed into the grooming path at its beginning are bundled together and cannot be accessed individually until the end of the path, where they

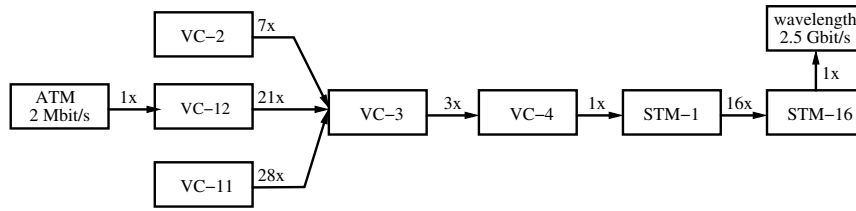


Figure 2: ATM over SDH over WDM: parts of the multiplexing hierarchy

are demultiplexed. The purpose of defining grooming paths which traverse several technologies is often the encapsulation of some data into another protocol, because many devices can only cope with one technology. Within a technology, grooming paths are used to speed up and simplify the routing process: interior nodes of a grooming path can route one coarse signal instead of routing many small units of data individually. In this paper, the term *grooming* refers to the process of defining a grooming path, multiplexing data at its beginning, and demultiplexing it at its end.

Grooming paths exist in many technologies; important examples are given in Table 1 for the technologies mentioned in Section 2.1. Some of these grooming paths always have the same bandwidth which can be seen from the table. Others, like LSPs and VPs, can be assigned any bandwidth within the given link capacities.

| Name | Techn. | Bandwidth |
|---------------------|--------|------------|
| Label Switched Path | MPLS | |
| Virtual Path | ATM | |
| VC-12-path | SDH | 2 Mbit/s |
| VC-3-path | SDH | 45 Mbit/s |
| VC-4-path | SDH | 140 Mbit/s |
| 2.5Gbit/s-lightpath | WDM | 2.5 Gbit/s |
| 10Gbit/s-lightpath | WDM | 10 Gbit/s |

Table 1: Important grooming paths

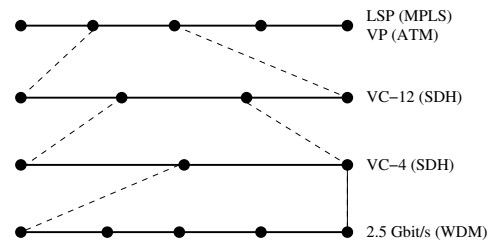


Figure 3: Layered view on the network

2.2.2 Layers

The characteristics of a grooming path are similar to those of a direct link: everything inserted at the beginning is routed until the end and cannot be accessed inside. This motivates a layered view on the network: a grooming path in a *server layer* defines a so-called *logical link* in a *client layer*. Thus, a logical link seems like a direct link but is actually a path in another layer. The links of a grooming path can themselves be logical links again, i.e., implemented by grooming paths in even another layer. This concept is illustrated in Figure 3, where MPLS/ATM is a client layer to VC-12, which is a client layer to VC-4, and so on.

Links which are not implemented by a path within the set of layers under consideration are called *physical links* (e.g., optical, copper fiber, or radio links). Notice that this term can be slightly misleading because a physical link might also be a leased line, for example. It is employed in this article nevertheless because it is the usual terminology.

Figure 3 also shows the most important characteristic of a multi-layer routing: it is *nested*, i.e., it consists of

paths in paths in paths in paths...

This property makes it hard to cope with multi-layer routings both operationally and computationally. The advantage of the layered view is that client nodes usually do not (and need not) care on which path and through which technologies they are connected; they only need to know that *there is* a connection. In

other words, they can just route the traffic on the “links” in their layer and ask the next lower layer to implement these links by a path. This approach reduces the nested multi-layer routing to a sequence of single-layer routings, which greatly simplifies routing decisions. However, this approach leads to other practical problems, as discussed in Section 2.4.5.

Notice that one technology may span several grooming layers, if multiplexing within the technology is supported; SDH, for instance, is inherently a multi-layer technology. This implies that networks with more than two layers are the rule, not the exception.

2.3 Hardware of a communication network

The hardware of a communication network determines both the physical capacities provided to the multi-layer routing and the investment cost of a network. Two basic functionalities have to be provided at the nodes in each layer: (i) multiplexing and demultiplexing at the end nodes of grooming paths, and (ii) switching (i.e., forwarding a signal from an input port to an output port) at inner nodes of a grooming path. In SDH, these tasks are performed by *add-drop-multiplexers* (ADMs) and *digital cross-connects* (DXCs). In MPLS and ATM networks, *routers* and *switches* are used. In optical networks, ADMs multiplex lower-rate traffic into wavelengths, *WDM systems* multiplex wavelengths into optical fibers, and *optical cross-connects* (OXC)s switch wavelengths. In the following, the term *network element* refers to any of these devices.

Typical network elements in MPLS, ATM, or SDH provide different types of slots for interface cards, to which the links are attached. The interface cards determine the bandwidth of a link and the protocol by which two nodes communicate. For instance, an MPLS router and an SDH add-drop-multiplexer may communicate via MPLS, Ethernet, or SDH, depending on their interface cards. Some protocols can be used on different kinds of links; for instance, two SDH add-drop-multiplexers may be connected by an optical fiber, by copper wire, or by a radio link. Of course, not all of these link types, cards, and network elements are compatible with each other, but it is beyond the scope of this paper to go into detail about the combinations which can be realized in practice.

Interface cards are responsible for multiplexing and demultiplexing, and thus determine where the grooming paths start and end. Figure 4 illustrates this at the example of an ATM/SDH network consisting of two locations. An interface card at the ATM switch on the left-hand side multiplexes ATM cells into SDH Virtual Containers and sends those to an SDH cross-connect. This cross-connect sends the Virtual Containers unchanged to the SDH add-drop-multiplexer on the right-hand side, whose interface card extracts the ATM cells and sends them as-is to the other ATM switch on an ATM link.

In optical networks, cost is incurred by slightly different hardware: optical fibers are needed to transport the wavelength signals. They are terminated by a WDM system on each side, which provides a certain number of ports (often 40 or 160) to the links leading to an SDH multiplexer or ATM switch, for instance. At the network elements where lightpaths are terminated, additional cost is incurred for *transmitters* (lasers) or *receivers* (photo detectors). To switch an optical signal from one link to another one, the optical fibers are directly attached to OXC)s; no interface cards are needed for these network elements. For long optical fibers, additional signal amplifiers and/or regenerators may be required in the middle of a link to avoid signal degradation. Furthermore, since two signals routed on the same fiber must have different wavelengths, wavelength converters may be needed at some places in the network.

Summarizing, cost is incurred by node hardware (e.g., network elements, wavelength converters, amplifiers, and regenerators), interface cards, transmitters and receivers, and physical links (e.g., fiber-optic or copper cables, radio links, or leased lines). In particular, logical links do not incur any cost, except for the interface cards at the end nodes.

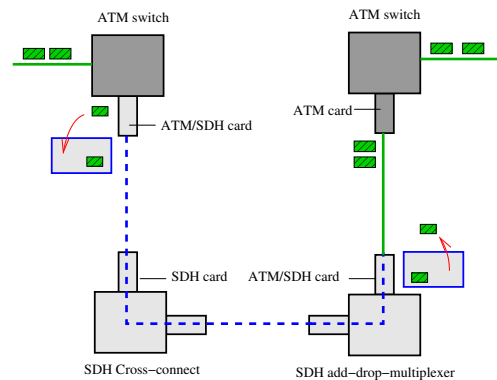


Figure 4: Two ATM/SDH locations

2.4 Multi-layer network planning

Planning a multi-layer network involves many interdependent decisions; the most important ones will be briefly discussed now.

2.4.1 Topology, node and link configurations

For all locations (e.g., cities) which should be part of the network and for all technologies under consideration, a hardware configuration has to be determined, together with inter- and intra-location connections. For a single location, this includes, but is not limited to, the following issues: which network elements are installed in which layers, how are they connected with each other and to the rest of the network, by which protocols do they communicate, and how many interface cards of which type are needed to achieve this? Furthermore, even for one type of network element, there may be several alternatives.

On the links, the network operator may have different configurations for choice: first of all, a link can be physical or logical. For logical links, an implementation by paths in another layer has to be determined. Depending on the employed technologies, a physical link could be implemented as an optical fiber, copper wire, radio link, or leased line, with different possible bandwidths. Eventually, the end nodes of the links have to be equipped with suitable hardware to realize the chosen link configuration.

Networks are usually not designed from scratch. Consequently, existing equipment, like network elements or interface cards, has to be taken into account in the planning process.

2.4.2 Demands, Routing and Grooming

A network is designed and dimensioned to carry communication traffic. Within the planning process, traffic requirements are usually considered in the form of estimated point-to-point demands, which may originate from different kinds of customers (e.g., home users, enterprises, other network providers,...). Hence, bandwidth may be requested in different regions and layers of the network, and the demands may exhibit a great variety in their requirements for the amount of bandwidth, transmission quality, and survivability.

For all these demands, an end-to-end routing through several network layers has to be realized. In particular, this requires defining appropriate grooming paths and deciding at which nodes which traffic is multiplexed together. Additionally, if an optical network is involved in the planning scenario, wavelengths have to be assigned to lightpaths. What type of routing is admissible depends on the layers, technologies, and routing protocols. For instance, in *connection-oriented* protocols, an end-to-end routing path has to be established before any data is sent. In contrast, *connectionless* protocols do not explicitly set up routing paths, but use local information such as artificial link weights to route the data. Routing paths might be restricted in the number of traversed network elements to satisfy end-to-end delay restrictions, or they may be required to respect a maximal length in kilometers to ensure sufficient signal quality. Similarly to existing hardware in the network, preconfigured routing paths might exist which should not be changed during the planning process.

2.4.3 Survivability

In a network, several components may be subject to a failure, e.g., network elements, interface cards, ports, single fibers, or whole links containing several fibers. For each of these failure types, some backup capacity in form of link capacities, free ports, etc., have to be provided such that affected traffic can be rerouted around the failure.

The easiest way to cope with all these failures at the same time is to use *1+1 end-to-end protection*. This means duplicating each demand at its source node, routing it on two paths which do not have any link (or network element) in common, and letting the target node choose the better signal. A demand routed on link-disjoint paths will survive any single fiber or cable cut as well as any port failure. With a routing which is disjoint with respect to network elements, it will also survive any single network element failure.

Notice that implementing survivability of any kind in multi-layer networks is a highly non-trivial task. It is not clear which layers should react to a failure at all, and how the actions of the different layers should be coordinated, especially if several management systems are involved. The questions how much each node in the network should know about the rest of the network, how many management systems the network

should have, and how these could be coordinated in case of a failure, is still discussed. An overview on the discussion and first results can be found in [14, 15].

2.4.4 Cost

In order to properly take installation cost into account, it has to be considered where it is actually incurred, i.e., at network elements, interface cards, and physical links. For logical links, it is very difficult to determine a realistic cost value which is a good approximation of the resources used in lower layers. This is particularly true if the realization of a logical link in lower layers is not known.

2.4.5 Planning approach in practice

Most practically relevant planning problems involve more than two layers: SDH is inherently a multi-layer technology by its VC- N container hierarchy; similarly, optical network planning is a multi-layer problem in itself with the lightpath layer and the fiber layer. Hence, in an ATM over SDH over WDM scenario, the network planner might easily end up with five layers or more, and even an MPLS over WDM network actually consists of three layers. Thus, being able to cope with networks with more than two layers will still be important for many years.

The traditional approach to multi-layer routing problems in practice is planning each layer individually in a top-down manner. More precisely, a routing on the highest layer is computed for the given demands, and logical capacities are derived from the routing. These capacities are then given as input to the planning problem of the next lower layer, as point-to-point demands between the end nodes of logical links.

On the one hand, this approach is attractive to network planners, since a series of single-layer problems can be tackled much easier computationally than one multi-layer problem. On the other hand, the sequential approach has several serious drawbacks:

- Survivability is hard to implement. Given that routing a demand over two logically link disjoint paths does not imply that the demand is actually routed over physically disjoint paths, computing a survivable routing requires taking into account all layers.
- It is difficult to impossible to assign a realistic cost value to a logical link, especially if its realization in lower layers is not known. This implies that cost minimization based on these costs is at least questionable. Furthermore, even with a good approximation of logical link cost, a sequential approach will probably lead to a more cost-intensive network than an integrated approach, since a logical layer designed for cost minimization may not be optimal for the whole network.
- If the routings in the different layers are not sufficiently coordinated, demands of higher layers are routed several times around the whole physical network. This leads to unnecessary capacity consumption on the links and to high transmission delays due to the high number of node traversals. Additionally, the more links and nodes are traversed by a demand, the higher the probability that the demand fails in case of a link or node failure.

The importance of these practical problems raises with the number of network layers. For all these reasons, an *integrated* planning approach of all layers is a must, from an operational as well as from a planning point of view.

3 Mathematical model

In this section, we present an integer programming model for multi-layer network planning, covering

- topology planning of several layers,
- hardware configuration at the nodes,
- capacities on the physical links,
- a routing over several layers,
- cost minimization,

- and survivability.

The model consists of a hardware part and a routing part, connected with each other by physical link capacities, as illustrated by Figure 5. These parts and the necessary preliminaries will be presented and discussed in detail in this section.

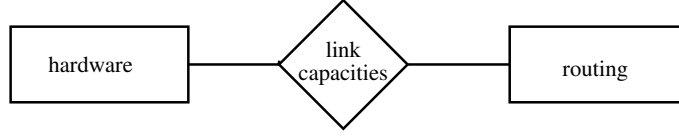


Figure 5: The different parts of the model

3.1 Layers and Network

3.1.1 Layers

Let \mathcal{L} be a set of *network layers*. With each layer $\ell \in \mathcal{L}$, a *routing unit* $r^\ell \in \mathbb{Z}_+$ is associated, which denotes the unit in which demands of layer ℓ are routed. All routing units are expressed relative to a common nonnegative integer *base unit* which can, w.l.o.g., be assumed to be 1. We assume that the routing units satisfy the *divisibility property* $\ell_1 \leq_L \ell_2 \implies r^{\ell_2}/r^{\ell_1} \in \mathbb{Z}_+$.

The way in which layers can be explicitly or implicitly embedded into each other is defined by a directed acyclic graph L on the node set \mathcal{L} . There is a directed link from layer ℓ_1 to layer ℓ_2 in L if and only if ℓ_1 can directly be multiplexed into ℓ_2 . The transitive closure of L defines a partial order \leq_L on the set of layers where $\ell_1 \leq_L \ell_2$ means that ℓ_1 can directly or indirectly be multiplexed into layer ℓ_2 .

As suggested by the Virtual Containers in SDH, a data stream with a given routing unit can be seen as a *container* of a certain size. Multiplexing low-rate signals into a high-rate signal can be interpreted as embedding small containers into a big one, as illustrated in Figure 6. The value r^{ℓ_2}/r^{ℓ_1} is the number of containers of type ℓ_1 which can be multiplexed into one container of type ℓ_2 .

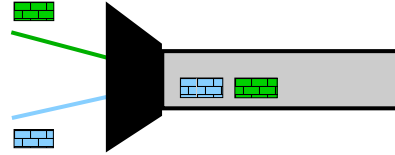


Figure 6: Multiplexing and containers

3.1.2 Potential physical network

The underlying network (V, E) is the *physical planning topology* from which nodes and links can be chosen. The node set V consists of all *potential* places where a network element (e.g., SDH multiplexer, cross-connect, ATM switch, IP router) may be installed. The set E comprises all *potential physical links* between nodes in V . Parallel links are allowed.

Nodes and links do not belong to a particular network layer. Instead, a set $\mathcal{L}_v \subseteq \mathcal{L}$ of *supported layers* is associated with each node $v \in V$. Grooming paths corresponding to these layers may be routed through v or may terminate at v . Likewise, each link $e \in E$ supports some set $\mathcal{L}_e \subseteq \mathcal{L}$ of layers which can be routed through it.

3.1.3 Discussion of layers and network

Layer vs. technology: There is no one-to-one correspondence between layers and technologies. In most practical applications, each considered technology will be represented by at least one layer, and may comprise several layers. For example, both the VC-12 and a VC-4 layer are part of the SDH technology. In particular, a node may support several layers (e.g., VC-12 and VC-4 in the case of an SDH multiplexer).

Nodes: The purpose of nodes is manifold: nodes (i) terminate physical links, (ii) are source and target of communication demands, and (iii) can be subject of failures as a whole. The most common practical application of a node is a particular network element, such as an ATM switch, an MPLS router, an SDH multiplexer, a WDM multiplexer, etc. However, nodes may also represent a complete network location comprising several pre-configured network elements, or a set of network elements from which an appropriate one has to be chosen.

Links: A physical link in the planning network may exist between *any pair of nodes*. In particular, there may be physical links in an upper layer. For instance, in an ATM over VC-4 scenario, two ATM switches may be connected by a direct physical link. Furthermore, physical links may bypass layers if this is technologically feasible. In an MPLS over ATM over SDH scenario, for example, an MPLS router could be directly connected to an SDH multiplexer, as well as to one or more ATM switches. Logical links do not incur any cost (apart from interface card cost, which is counted separately), and they are not really needed for routing. Consequently, instead of being included in the network, they are represented by grooming paths and considered in Section 3.5. All links in the planning graph are *physical links*, and unless otherwise stated, the term *link* always refers to a physical link in the sequel.

Routing units: If the routing units are defined according to the given multiplexing hierarchy L , such that r^{ℓ_2}/r^{ℓ_1} is the number of containers of type ℓ_1 which can be embedded into a container of type ℓ_2 , the divisibility assumption $\ell_1 \leq_L \ell_2 \implies r^{\ell_2}/r^{\ell_1} \in \mathbb{Z}_+$ is always satisfied. For instance, for the layers shown in Figure 2, the routing units could be defined as follows:

| base | ATM | VC-2 | VC-12 | VC-11 | VC-3 | VC-4 | STM-1 | STM-16 | WDM |
|------|-----|------|-------|-------|------|------|-------|--------|------|
| 1 | 4 | 12 | 4 | 3 | 84 | 252 | 252 | 4032 | 4032 |

Table 2: Examples of routing units

3.2 Hardware

The hardware model is based on [16]; its principle is illustrated in Figure 7. Node designs, which are installed at the nodes, provide slots of different types into which interface cards can be plugged. These interface cards provide ports to which link designs can be attached, which are in turn installed on the links and provide capacity to the routing.

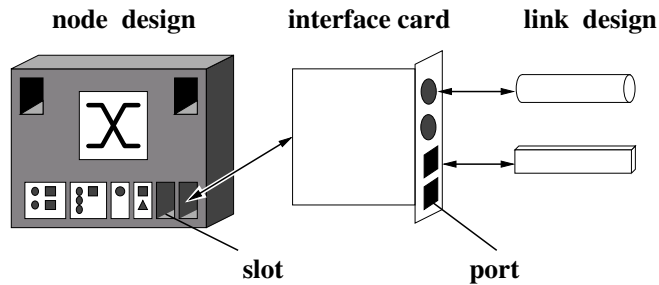


Figure 7: Hardware model

3.2.1 Parameters

This section explains the parameters of the hardware model; they are summarized in Table 3.

| parameter | index domains | description |
|---------------------------------|---|--|
| \mathcal{S} | | set of all slot types |
| \mathcal{P} | | set of all port types |
| \mathcal{I} | | set of all interface cards |
| \mathcal{D}_v | $v \in V$ | set of node designs installable at node v |
| \mathcal{D}_e | $e \in E$ | set of link designs installable on link e |
| \mathcal{I}_s | $s \in \mathcal{S}$ | set of interface cards requiring slots of type s |
| $L_v^d, U_v^d \in \mathbb{Z}_+$ | $v \in V, d \in \mathcal{D}_v$ | bounds on node design of type d installable at v |
| $L_e^d, U_e^d \in \mathbb{Z}_+$ | $e \in E, d \in \mathcal{D}_e$ | bounds on link design of type d installable on e |
| $L_v, U_v \in \mathbb{Z}_+$ | $v \in V$ | bounds on total number of node designs installable at v |
| $L_e, U_e \in \mathbb{Z}_+$ | $e \in E$ | bounds on total number of link designs installable on e |
| $S_s^d \in \mathbb{Z}_+$ | $s \in \mathcal{S}, v \in V, d \in \mathcal{D}_v$ | number of slots of type s provided by node design d |
| $S_s^i \in \mathbb{Z}_+$ | $s \in \mathcal{S}, i \in \mathcal{I}_s$ | number of slots of type s required by interface card i |
| $P_p^i \in \mathbb{Z}_+$ | $p \in \mathcal{P}, i \in \mathcal{I}$ | number of ports of type p provided by interface card i |
| $P_p^d \in \mathbb{Z}_+$ | $p \in \mathcal{P}, e \in E, d \in \mathcal{D}_e$ | number of ports of type p required by link design d |
| $C_e^d \in \mathbb{Z}_+$ | $e \in E, d \in \mathcal{D}_e$ | routing capacity provided by link design d on link e |
| $K_v^d \in \mathbb{Z}_+$ | $v \in V, d \in \mathcal{D}_v$ | cost of node design d at node v |
| $K_e^d \in \mathbb{Z}_+$ | $e \in E, d \in \mathcal{D}_e$ | cost of link design d on link e |
| $K^i \in \mathbb{Z}_+$ | $i \in \mathcal{I}$ | cost of interface card i |

Table 3: Parameters of the hardware model

Let \mathcal{S} be the set of *slot types* and \mathcal{P} the set of *port types*. With every node $v \in V$, a set \mathcal{D}_v of installable *node designs* is associated; similarly, for every link e , there is a set \mathcal{D}_e of installable *link designs*. The notation \mathcal{I}_s refers to the set of *interface cards* requiring slots of type $s \in \mathcal{S}$. The sets \mathcal{D}_v , \mathcal{D}_e , and \mathcal{I}_s are assumed to be disjoint among different nodes, links, and slot types, respectively. In particular, every interface card is assumed to need only slots of one particular type. For ease of notation, let

$$\mathcal{I} := \biguplus_{s \in \mathcal{S}} \mathcal{I}_s$$

be the set of *all* interface cards.

At any node $v \in V$, between L_v^d and U_v^d instances of node design $d \in \mathcal{D}_v$ may be installed. Altogether, between L_v and U_v instances of node designs may be installed at v . Likewise, the number of instances of link design $d \in \mathcal{D}_e$ installed on $e \in E$ must be between L_e^d and U_e^d , and between L_e and U_e instances of link designs may be installed on e in total. If a node or link is not equipped with any design, it is not included in the final topology.

Every node design $d \in \mathcal{D}_v$ offers $S_s^d \in \mathbb{Z}_+$ slots of type $s \in \mathcal{S}$. An interface card $i \in \mathcal{I}_s$ requires $S_s^i \in \mathbb{Z}_+$ slots of type $s \in \mathcal{S}$; it does not require slots of any other type. Furthermore, an interface card $i \in \mathcal{I}$ provides $P_p^i \in \mathbb{Z}_+$ ports of type $p \in \mathcal{P}$. Each link design $d \in \mathcal{D}_e$ provides a *routing capacity* $C_e^d \in \mathbb{Z}_+$ to the routing (given in the base unit), and requires $P_p^d \in \mathbb{Z}_+$ ports of type $p \in \mathcal{P}$; it may need ports of several types simultaneously.

3.2.2 Variables

Node designs: For every node $v \in V$ and every node design $d \in \mathcal{D}_v$, the variable $x_v^d \in \mathbb{Z}_+$ states how often d is installed at v .

Link designs: For every link $e \in E$ and every link design $d \in \mathcal{D}_e$, the variable $x_e^d \in \mathbb{Z}_+$ states how often d is installed at e .

Interface cards: For every node $v \in V$ and every interface card $i \in \mathcal{I}$, the variable $x_v^i \in \mathbb{Z}_+$ states how often i is installed at v .

3.2.3 Constraints

The following set of linear inequalities models the compatibility requirements of the hardware installed in the network:

$$L_v^d \leq x_v^d \leq U_v^d \quad \forall v \in V, \forall d \in \mathcal{D}_v \quad (1)$$

$$L_e^d \leq x_e^d \leq U_e^d \quad \forall e \in E, \forall d \in \mathcal{D}_e \quad (2)$$

$$L_v \leq \sum_{d \in \mathcal{D}_v} x_v^d \leq U_v \quad \forall v \in V \quad (3)$$

$$L_e \leq \sum_{d \in \mathcal{D}_e} x_e^d \leq U_e \quad \forall e \in E \quad (4)$$

$$\sum_{e \in \delta(v)} \sum_{d \in \mathcal{D}_e} P_p^d x_e^d \leq \sum_{i \in \mathcal{I}} P_p^i x_v^i \quad \forall v \in V, \forall p \in \mathcal{P} \quad (5)$$

$$\sum_{i \in \mathcal{I}_s} S_s^i x_v^i \leq \sum_{d \in \mathcal{D}_v} S_s^d x_v^d \quad \forall v \in V, \forall s \in \mathcal{S} \quad (6)$$

$$x_v^i, x_v^d, x_e^d \in \mathbb{Z}_+$$

With inequalities (1) and (2), the number of installable designs *of a specific type* at a node or link is bounded, while inequalities (3) and (4) restrict the *total number* of node and link designs which can be installed at a node or link, respectively. The port inequalities (5) ensure that for every node and every port type, the installed interface cards provide enough ports for the link designs on incident links. Eventually, the slot inequalities (6) state that for every node and every slot type, the number of slots required by the installed interface cards must not exceed the number of slots provided by the installed node designs.

3.2.4 Discussion of the hardware model

In this subsection, the hardware model is discussed from a practical perspective; examples show how it can be used to deal with particular practical planning tasks and hardware requirements.

Node designs: Different kinds of network elements can be modeled with node designs, e.g., MPLS routers, ATM switches, SDH multiplexers, WDM systems, optical cross-connects, etc. Node designs can also be used to distinguish between network elements of different vendors, or between a different number of shelves at the same network element.

Slot types: Network elements and therefore node designs in the model can provide different types of slots. As an example, an add-drop-multiplexer in SDH networks has two aggregate slots and a fixed number N of tributary slots. Cards fitting into tributary slots do not necessarily fit into aggregate slots. If the ADM is represented by a particular node design d , this situation can be modeled with two slot types s_A and s_T , for aggregates and tributaries, respectively. In this case, the number of provided slots of the node design must be set to $S_{s_A}^d = 2$ and $S_{s_T}^d = N$.

Interface cards: Interface cards provide ports to which link designs may be attached. For example, typical SDH-cards offer STM-1 ports or STM-4 ports; in optical networks, interface cards may represent transmitters and receivers. In most practical settings, an interface card requires exactly one slot and fits into exactly one slot type. If an interface card $i \in \mathcal{I}$ fits into more, say N , slot types, it can be replaced by N artificial cards which have the same properties as i but different slot type requirements.

Link designs: With link designs, different kinds of links can be modeled, e.g., leased lines, fiber-optic cables, copper cables, microwave connections, etc. Link designs can also be used to distinguish between different providers of link capacity, or between different configurations of a link, like STM-1, STM-4, or STM-16, or combinations of these.

Preinstalled hardware: The expansion of an existing network is more common in practice than building a new network from scratch. Since it can be cost-intensive to perform major changes in an operating network, it is very important that existing hardware can be taken into account.

In our model, preinstalled hardware can be respected by choosing the parameters appropriately. Existing network elements, for instance, can be taken into account by introducing both a node v and a node design d for this particular network element, and by setting $L_v = L_v^d = 1$. Existing interface cards can be respected by setting appropriate lower bounds on the variables x_v^i corresponding to these interface cards.

3.3 Objective

Our goal is to minimize total installation cost, which consists of cost for node designs, link designs, and interface cards. Hence, for every node $v \in V$ and every node design $d \in \mathcal{D}_v$, let $K_v^d \in \mathbb{Z}_+$ be the cost of node design d . Similarly, $K_e^d \in \mathbb{Z}_+$ denotes the cost of link design $d \in \mathcal{D}_e$, for some link $e \in E$. Eventually, every interface card $i \in \mathcal{I}$ incurs a cost of $K^i \in \mathbb{Z}_+$.

The objective is to minimize the sum of all these cost values:

$$\min \sum_{v \in V} \left(\underbrace{\sum_{d \in \mathcal{D}_v} K_v^d x_v^d}_{\text{node design cost}} + \underbrace{\sum_{i \in \mathcal{I}} K^i x_v^i}_{\text{card cost}} \right) + \sum_{e \in E} \underbrace{\sum_{d \in \mathcal{D}_e} K_e^d x_e^d}_{\text{link design cost}} \quad (7)$$

Notice that this objective function does not use any estimated logical link cost. Instead, cost is considered where it is actually incurred, namely at network elements, interface cards, and physical links.

3.4 Link capacities

Many capacity models have been proposed. For instance,

- capacities may be continuous,
- any nonnegative integer value may be allowed,
- a small set of allowed capacities may be given via link designs as described in Section 3.2,
- or it may be admissible to install arbitrary combinations of a small set of base capacities.

From a routing point of view, it does not matter which capacity model is actually used, as long as each link $e \in E$ has some *routing capacity* y_e which provides an upper bound on the amount of flow on that link. This abstraction naturally leads to the separation of the model into a hardware part and a routing part which are connected by the link capacities, as illustrated in Figure 5. With the modular hardware model described in Section 3.2, the *routing capacity*

$$y_e := \sum_{d \in \mathcal{D}_e} C_e^d x_e^d.$$

of link $e \in E$ is the total capacity provided by the installed link designs.

3.5 Routing

The routing model introduced in this section divides each end-to-end *routing path* P into a sequence of subpaths (Q_i, ℓ_i) , which represent *grooming paths*. Examples of grooming paths are WDM lightpaths, VC- N -paths in SDH, Virtual Paths in ATM, or Label Switched Paths in MPLS. The interpretation of routing and grooming paths is that traffic is *physically* routed on path

$$P = ((Q_1, \ell_1), (Q_2, \ell_2), \dots, (Q_{n_P}, \ell_{n_P})),$$

and that it is multiplexed into containers of layer ℓ_i at the beginning of grooming path Q_i , and demultiplexed at its end.

3.5.1 Parameters

Demands: Let \mathcal{K}^ℓ be the set of point-to-point communication demands of layer $\ell \in \mathcal{L}$. For each demand $k \in \mathcal{K}^\ell$, let $d_k \in \mathbb{Z}_+$ denote its demand value in units of r^ℓ , that is, a total traffic of $d_k \cdot r^\ell$ in the base unit has to be routed through the network in units of r^ℓ . The end nodes s_k and t_k of a demand $k \in \mathcal{K}^\ell$ must support layer ℓ , i.e., $\ell \in \mathcal{L}_{s_k} \cap \mathcal{L}_{t_k}$. Parallel demands in the same layer are allowed. For the purpose of this article, the underlying network as well as communication demands are assumed to be undirected.

Grooming paths: For each layer $\ell \in \mathcal{L}$, a set \mathcal{Q}^ℓ of admissible *grooming paths* is given. Grooming paths (Q, ℓ) correspond to loopless (sub-)paths $Q \in \mathcal{Q}^\ell$ in the physical network, such that traffic routed through this subpath is multiplexed into containers of type ℓ at the beginning of Q , demultiplexed at its end, and not accessed at intermediate nodes. All nodes and all links in Q must support layer ℓ , i.e., $\ell \in \mathcal{L}_v$ and $\ell \in \mathcal{L}_e$ must hold for all $v \in Q$ and $e \in Q$, respectively. Clearly, a link can only be contained in grooming paths of layers which are supported by both end nodes. Consequently, $\mathcal{L}_e \subseteq \mathcal{L}_u \cap \mathcal{L}_v$ can be assumed for all links $e = uv \in E$.

Routing paths: For every layer $\ell \in \mathcal{L}$ and every demand $k \in \mathcal{K}^\ell$, the set \mathcal{P}_k defines the set of admissible end-to-end *routing paths* on which traffic for demand k may be routed. A routing path $P = ((Q_1, \ell_1), (Q_2, \ell_2), \dots, (Q_{n_P}, \ell_{n_P}))$, where $\ell \leq_L \ell_i$ for all $i = 1, \dots, n_P$, is a sequence of compatible grooming paths which connects source and target of the demand. A grooming path can also be interpreted as a logical link, and an end-to-end routing path P as a path on logical links.

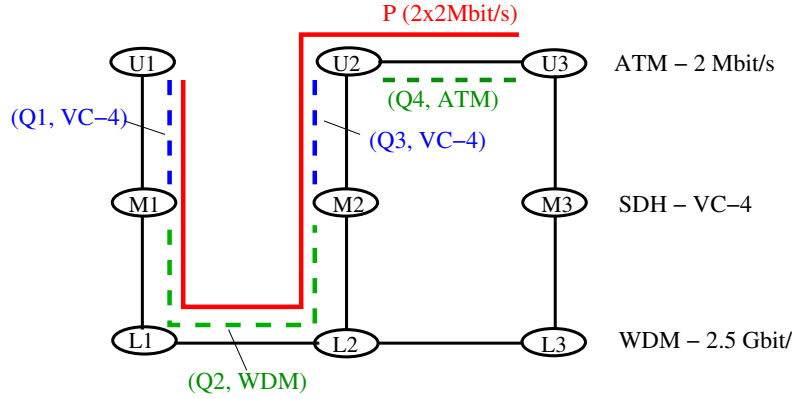


Figure 8: Routing example

Example: In Figure 8, an ATM demand is routed end-to-end using four (dashed) grooming paths: first, it is embedded into a VC-4-path $(Q_1, \text{VC-4})$ at node U_1 , which is then embedded into a WDM lightpath (Q_2, WDM) at node M_1 . At the end node M_2 of the lightpath, the VC-4 signal is demultiplexed and routed on a VC-4-path $(Q_3, \text{VC-4})$ to node U_2 . This ATM switch finally demultiplexes the ATM cells and routes them in units of 2 Mbit/s to the destination node U_3 on the ATM grooming path (Q_4, ATM) , which consists of a direct physical connection between two ATM switches. That is, the layer ℓ_i in the notation (Q_i, ℓ_i) denotes the biggest container type into which the traffic is wrapped on subpath Q_i . The (solid) routing path P consists of these four grooming paths.

3.5.2 Variables

The model contains two types of flow variables:

Flow on grooming paths: For each layer $\ell \in \mathcal{L}$ and each grooming path $Q \in \mathcal{Q}^\ell$, the variable $f^\ell(Q) \in \mathbb{Z}_+$ denotes the number of containers of type ℓ routed on Q .

Flow on routing paths: For each layer $\ell \in \mathcal{L}$, each demand $k \in \mathcal{K}^\ell$, and each end-to-end routing path $P \in \mathcal{P}_k$, the variable $f_k(P) \in \mathbb{Z}_+$ denotes the flow routed on path P for demand k in units of r^ℓ . This corresponds to a flow value of $f_k(P) \cdot r^\ell$ in the base unit.

Example (cont.): A possible variable assignment for the routing considered in Figure 8 is as follows:

- $f_{U_1, U_3}(P) = 2$, because both ATM flow units are routed over routing path P .
- $f^{\text{VC-4}}(Q_1) = f^{\text{VC-4}}(Q_3) = 1$, because only one VC-4 container is needed on the grooming paths $(Q_1, \text{VC-4})$ and $(Q_3, \text{VC-4})$, respectively, to accommodate both ATM flow units.
- $f^{\text{WDM}}(Q_2) = 1$, because only one WDM wavelength is needed on (Q_2, WDM) .
- $f^{\text{ATM}}(Q_4) = 2$, because two ATM containers are needed on the grooming path (Q_4, ATM) for the two ATM flow units.

3.5.3 Constraints

In practice, traffic for one layer is embedded into a next higher layer in the multiplexing hierarchy L , which is then in turn embedded into a higher layer, and so on. In contrast, the following formulation routes containers of different layers side by side in a bigger container, which is then routed on a physical link. This is discussed in the next section.

The following formulation models an undirected integer multicommodity flow in multi-layer networks:

$$\sum_{P \in \mathcal{P}_k} f_k(P) = d_k \quad \forall \ell \in \mathcal{L}, \forall k \in \mathcal{K}^\ell \quad (8)$$

$$\sum_{\ell' \in \mathcal{L}: \ell' \leq_L \ell} r^{\ell'} \sum_{k \in \mathcal{K}^{\ell'}} \sum_{\substack{P \in \mathcal{P}_k: \\ (Q, \ell) \in P}} f_k(P) \leq r^\ell f^\ell(Q) \quad \forall \ell \in \mathcal{L}, \forall Q \in \mathcal{Q}^\ell \quad (9)$$

$$\sum_{\ell \in \mathcal{L}_e} \sum_{\substack{Q \in \mathcal{Q}^\ell: \\ e \in Q}} r^\ell f^\ell(Q) \leq y_e \quad \forall e \in E \quad (10)$$

$$f_k(P), f^\ell(Q) \in \mathbb{Z}_+$$

The demand constraints (8) state that all end-to-end demands must be satisfied. The grooming constraints (9) ensure that on every grooming path (Q, ℓ) , enough containers are routed such that the total flow on routing paths P containing (Q, ℓ) as a subpath can be accommodated. This inequality can be seen as a capacity constraint on the logical links defined by grooming paths. Finally, the physical link capacity constraints (10) make sure that the total flow on all grooming paths using a physical link $e \in E$ does not exceed its capacity.

3.5.4 Discussion of the routing model

Type of routing: For the purpose of this paper, undirected links and demands are assumed, as well as an integer multicommodity flow, where the flow for each demand can be split among several paths. Obviously, these assumptions do not apply to all practical planning scenarios. For instance, the flow might also be fractional, a single-path routing might be required, or the demands and links might be directed or bidirected. For all these alternatives, the corresponding changes in the formulation are rather straightforward and not discussed here in detail.

As the routing model contains explicit end-to-end routing paths, it is only applicable to connection-oriented protocols, like MPLS, ATM with Virtual Paths, SDH, ODU, or WDM. Shortest path routing protocols based on link weights, such as IS-IS, OSPF, or PNNI, are not covered.

Admissible path sets: Besides restrictions imposed by supported layers of links and nodes, the sets of admissible grooming paths are not explicitly restricted. In a practical planning scenario, there might be additional constraints, e.g., on the length of grooming paths in kilometers or in the number of hops. Similarly, the length of routing paths might be restricted in the number of logical hops or in kilometers.

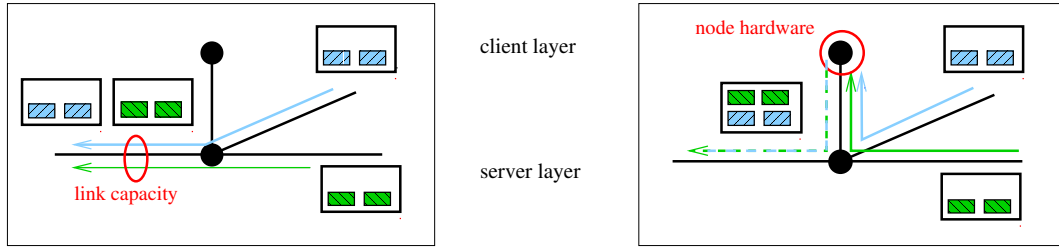


Figure 9: Link capacity vs. node hardware trade-off associated with grooming

The grooming trade-off: When routing a demand over two or more grooming paths, cost is incurred at the end nodes of each grooming path, because suitable hardware has to be installed to multiplex and demultiplex the traffic. Hence, if it is technologically feasible to route a demand on only one grooming path between the end nodes of a demand, the question arises why one should ever do otherwise. In fact, if the routing unit r^{ℓ_1} in which the demand is routed equals the routing unit r^{ℓ_2} of the used grooming path (e.g., in an ATM-2Mbit/s over VC-12 scenario), terminating the grooming path at an intermediate node and starting a new one cannot reduce the incurred cost.

If, in contrast, $r^{\ell_1} < r^{\ell_2}$ holds (e.g., in a VC-12 over VC-4 scenario), demultiplexing the flow at an intermediate node and combining it with other flow of layer ℓ_1 might reduce the incurred cost. The trade-off associated with grooming is illustrated in Figure 9: two grooming paths of layer ℓ_2 whose containers are only half-filled arrive at a server network element; the traffic of both will be routed together to the next server node. There are two possibilities: on the left-hand side, both grooming paths are continued to the next node and the containers are left unchanged. On the right-hand side, both signals are sent to the client node (which can access the small containers of layer ℓ_1), where the grooming paths are terminated. Both signals are then recombined into a new signal which is sent to the next node on a new grooming path. There is a trade-off between the additional link capacity needed in the first case and the expensive node hardware needed in the second case.

Grooming in more than two layers: In the presented formulation, different containers are routed *side by side* through a link and are directly embedded into a link design, rather than being embedded one into the other, which enables us to model grooming in three or more layers at least approximately. Under certain conditions, the needed capacity is modeled exactly: consider a grooming path (Q, ℓ) through which traffic for some layer ℓ' with smaller granularity is routed, i.e., (Q, ℓ) is contained in a routing path $P \in \mathcal{P}_{\ell'}$, for some layer $\ell' \leq L$. If the end nodes u and v of Q support layer ℓ' , i.e., $\ell' \in \mathcal{L}_u \cap \mathcal{L}_v$, they can demultiplex the signal down to layer ℓ' and recombine it with traffic arriving on other routing paths. If u and v support all layers of routing paths containing (Q, ℓ) as a logical link, the grooming constraints (9) lead to the correct number of containers $f^\ell(Q)$ on Q .

From a practical perspective, the described assumption is satisfied, for instance, if all grooming paths for a demand of layer ℓ' are required to start and end in layer ℓ' (as also explicitly or tacitly assumed in all previous models which consider grooming, see Section 4). This means, for example, that in an SDH over WDM network, all lightpaths terminate at SDH multiplexers.

If, on the other hand, the end nodes u and v of Q can only access coarser signals than those of layer ℓ' , the number of containers of type ℓ on Q is underestimated (and thus possibly the needed capacity as well). In this case, the proposed formulation relaxes the practical planning problem and thus leads to a lower bound on the optimal network cost.

Relation to peer and overlay model: The routing model contains explicit end-to-end routing paths over several layers. At first glance, this suggests the assumption of a peer network model, where all nodes have complete information about the whole network. However, such a routing can easily be decomposed into one routing per layer. Hence, the model is also applicable to overlay networks, where the knowledge of a node about the rest of the network is limited.

Preconfigured routing: Network operators are reluctant to change an existing routing configuration. It is therefore indispensable that an existing (partial) routing can be respected. In the model, such a preconfigured routing can easily be realized by setting lower bounds on end-to-end path flow variables and on grooming path variables.

3.6 Survivability

Survivability is of utmost importance to network operators and has to be addressed by any network planning model which claims to be realistic. Which survivability concept is used, however, depends on the considered technologies. Particularly in multi-layer networks, an important question is to which extent the different layers have information on each other (peer vs. overlay model), and whether there is a central management system or one per technology. Closely related to these decisions is the question whether backup paths (preconfigured or not) can be established and controlled over several layers, or whether several intra-layer management systems have to be coordinated in case of a failure, and how this could be done. All these issues are currently discussed (see for instance [14, 15, 17, 18]), but there are no standards yet, and it is not yet decided which survivability mechanisms will be technically possible and operationally desirable.

Out of the mathematical models for multi-layer network design (see Section 4.1), only two take survivability into account: Kubilinskas et al. [19] assume complete reconfiguration of the network in case of a failure, whereas Hu and Leida [20] consider 1+1 dedicated path protection in a two-layer network. In practice, it is unlikely that a complete reconfiguration of the routing is applicable. In fact, this would imply the interruption of all ongoing connections, regardless of whether they are directly affected by the failure or not. In contrast, 1+1 path protection over several layers is a serious option in an Automatically Switched Optical Network (ASON) [18]. Other practical situations where this approach is suitable are SDH over ODU, SDH over WDM, or pure SDH multi-layer network planning.

Suppose that the network is to be protected against some set $S \subseteq V \cup E$ of failures (e.g., all single node or physical link failures). The following model extension aims at modeling 1+1 dedicated path protection approximately with full end-to-end protection of all demands:

$$\sum_{P \in \mathcal{P}_k} f_k(P) = 2d_k \quad \forall \ell \in \mathcal{L}, \forall k \in \mathcal{K}^\ell \quad (11)$$

$$\sum_{\substack{P \in \mathcal{P}_k: \\ P \text{ fails in } s}} f_k(P) \leq d_k \quad \forall \ell \in \mathcal{L}, \forall k \in \mathcal{K}^\ell, \forall s \in S \quad (12)$$

Constraints (11) replace the original demand constraints (8) since the value of each demand has to be doubled. In addition, Constraints (12) ensure that in each of the considered failure situations, at least the original demand value d_k survives.

This formulation is exact for all demands k with original demand value $d_k = 1$. Every such demand is routed on exactly two disjoint paths, each of which carries one unit of flow. Demands with higher demand value are routed on a path system which may contain more than two paths. Whenever this path system can be decomposed into pairs of paths with equal flow value which are disjoint with respect to S , this path system leads to a feasible 1+1 protection routing for those demands. Sometimes, the set of routing paths for a demand cannot be decomposed into pairs of disjoint paths (for example, consider a system of four paths in which every pair of paths shares a common link), which means that the routing is not directly feasible for 1+1 protection. On the other hand, the above formulation still defines a relaxation of 1+1 protection since every routing for the latter satisfies the above constraints. In our practical experiences with single-layer models and the more general survivability concept *diversification* [12, 21], the described phenomenon rarely occurred, i. e., solutions of the relaxation are almost always feasible for 1+1 dedicated path protection as well.

From an algorithmic point of view, the described formulation is suitable for a column generation approach as long as only single physical link or node failures are considered. In this case, missing routing paths can be identified by solving a shortest path problem on the physical links with respect to weights derived from the dual solution of the routing LP [12].

Notice that several survivability mechanisms which aim at protecting links or ports, such as MSP or EQP, can be incorporated in our formulation by defining link designs with appropriate port requirements and capacities. Other survivability concepts like SNCP, where the amount of routed flow changes in the middle of a path, can be approximately incorporated into the model by means of the above formulation for diversification in combination with a feasibility algorithm which makes a diversification solution feasible for SNCP, or rejects a solution if this is not possible. Restoration or protection within a single layer, however, are not covered by our formulation because explicit control over paths within a layer would be needed. Our formulation does not distinguish between the paths which traverse different layers and those which do not.

3.7 Extensions of the model

Although the mathematical formulation presented in the previous sections already covers a lot of practical planning issues, additional side constraints might be required from a practical point of view. This section briefly presents possible extensions to the model which could be useful in some planning scenarios.

Coloring a WDM routing: In purely optical networks, every lightpath has a wavelength, and two lightpaths with the same wavelength must not share a common fiber. This can easily be incorporated into our formulation since lightpaths are grooming paths. Using one layer (or container type) for each wavelength, a class of constraints has to be added, stating that for any link and any wavelength, at most one grooming path corresponding to the given wavelength should traverse the link.

From a model point of view, this solves the problem. However, it is unlikely that this approach still yields satisfying results. Given that the network dimensioning, grooming, and routing problem as well as the wavelength assignment problem are already very hard if treated separately, it is probably just pointless trying to solve them together. Instead, we propose to decompose the problem into a dimensioning, grooming, and routing part and a wavelength assignment part [22].

Maximum switching capacity: Network elements often have a maximum switching capacity which limits the traffic through a node. In SDH networks, these restrictions are usually satisfied whenever the port and slot constraints are fulfilled. On the other hand, optical cross-connects can switch only a limited number of lightpaths, but no restrictions are imposed by means of interface cards or link capacities. Furthermore, an artificial switching capacity can be used to prohibit full usage of installed equipment.

Such restrictions can easily be added to the model by assigning a maximum switching capacity to every node design. For every node $v \in V$, an additional constraint states that the total switching capacity of the node designs installed at v limits either the total routing capacity y_e of the edges e incident to v , or the total flow on routing paths P traversing v , or the number of grooming paths (Q, ℓ) which contain v .

4 How the model adds to the literature

This section gives an overview on the available literature on multi-layer network design, and summarizes the pros and cons of the presented formulation in relation to previous models.

4.1 Previous approaches

A huge amount of literature deals with grooming in physical ring topologies. The following literature review is restricted, however, to papers in which models for arbitrary network topologies are presented, since this is the more general approach. An extensive overview on ring grooming literature can be found in [23]. Furthermore, only those formulations are considered which address a nested routing (i.e., the routing of logical links on physical links), since this is the key issue which makes a multi-layer planning problem difficult, and the main criterion to distinguish it from a single-layer planning problem.

Basically, the publications on multi-layer network design can be divided into two groups. The first group [19, 24] is based on the traditional top-down planning approach where the capacities of one layer define the demands for the next one, except that the whole network is considered in an integrated step. These

models or straightforward extensions thereof are applicable to any number of layers, but the formulations do not take grooming into account. The other group [20, 25, 26] discusses routing models considering a nested routing in two layers. While these formulations are restricted to two layers, they can be used to describe grooming correctly. One publication [27] does not fit into either of these two groups because grooming is not modeled, although the model is restricted to two layers. This paper is interesting for other reasons, as discussed later on.

Knippel and Lardeux [24] propose an integer program where the capacities of one layer define the demands of the next lower one. The problem is formulated by means of metric inequalities for the failureless state, with discrete capacities. The objective is to minimize the sum of some step-wise increasing discrete functions over the link-flows. In the computational results, both the physical and the logical topology are given in advance, and the routing of the logical layer in the physical one has to be determined. To achieve this, the authors solve the LP relaxation of the presented integer program. Results on networks up to 10 nodes and 18 links in the physical layer are presented (the logical layer is a complete graph). The authors compare the results of an integrated planning of both layers to sequential planning and find out that the cost difference can be substantial and is not predictable.

Kubilinskas, Pióro, and Nilson [19] present three models for two-layer network planning, all based on the top-down approach. These models consider complete reconfiguration in case of a failure in the upper, in the lower, or in both layers. In one of their models, the objective is to maximize a weighted sum of the logarithms of upper layer link throughput, while the flow is constrained by a global budget constraint on the total capacity in the network. The upper layer demand values are not fixed but can be freely chosen within a given range. In the other two models, a revenue vector is lexicographically maximized. These problems are approximated using an iterative algorithm for convex lexicographical maximization, combined with a piecewise linearization of the logarithmic function, which reduces the problem to solving linear programs. The results obtained with the three formulations are then compared for different input parameters.

Banerjee and Mukherjee [25] propose an integer programming formulation for two-layer networks in the failureless state. The model routes the client demands on lightpaths and the lightpaths in the physical network. As an unlimited number of converters is assumed to be available at zero cost, lightpath blocking and wavelength continuity constraints are omitted. The logical degree of a node is bounded by the number of available transmitters and receivers at the end nodes of a lightpath. The objective is to minimize a weighted average of the number of logical hops of all routing paths, to reflect the number of lightpaths needed. In the numerical computations on one test network, each demand may be routed on at most two (shortest) lightpaths, and integrality conditions are omitted, such that a linear program can be solved.

Dutta and Rouskas [26] present a model for the failureless state which is similar to the one in [25], with some additional constraints. These are the number of available wavelengths per fiber, lightpath blocking constraints and hop limits for the lightpaths. The number of available transmitters and receivers is modeled by logical degree bounds. Whereas the potential logical links form a complete graph, the potential physical topology may be restricted to a subset of links. The objective is to minimize the maximum load on any lightpath. No computational experiments are given.

Hu and Leida [20] present a mixed-integer programming formulation for the *Grooming, Routing and Wavelength Assignment* (GRWA) problem, with and without 1+1 end-to-end protection. The objective is to minimize the number of lightpaths and thus the number of needed transponders, which cause most of the network cost. The authors propose the decomposition of the problem into the *Grooming and Routing* (GR) problem and the subsequent coloring of the routing, called the *Wavelength Assignment* (WA) problem; for both problems, integer programming models are presented. Computational results are given for the failureless state; the test instances are sparse networks with up to 144 nodes but with a very small number of admissible lightpaths (two to three times the number of nodes). The problem GR is solved by relaxing the integer constraints, solving an LP and rounding up the values. The problem WA is solved by a greedy heuristic (with an unlimited number of available wavelengths). In the few cases where solutions have been found with the integrated approach, they are compared to the results of the heuristic approach.

Belotti and Malucelli [27] present an integer programming formulation for two network layers where end-to-end routing paths are divided into subpaths. Arbitrary integer capacities can be installed on physical links and subpaths, and a weighted sum of these capacities is minimized. Neither hardware nor grooming are considered in the model. Contrary to all other models considered above, the authors explicitly distinguish between nodes at the lower and at the upper layer. In addition, they sketch the general principles

of the multi-layer pricing problem, and present some computational results obtained by solving the LP relaxation and rounding up the obtained values.

4.2 Contributions of our formulation

The most important features of our model are:

- The formulation allows for an *integrated* planning of physical and logical topologies, node hardware (network elements, interface cards), link capacities, and a routing over several layers, intra-location planning, and survivability.
- The generic hardware model is very flexible and provides a detailed description of the hardware employed in different technologies. Furthermore, hardware cost is considered where it incurs: at network elements, interface cards, and physical links. This allows to take cost more precisely into account than by using estimated logical link costs or no hardware costs at all.
- Any number of layers can be reduced to just two path layers: paths on physical links and paths on logical links. We expect that this feature will make planning problems with more than two layers computationally tractable even if hardware restrictions, different routing granularities, and grooming are taken into account.
- Preinstalled hardware and a preconfigured routing can easily be respected by setting lower variable bounds appropriately, as discussed in Sections 3.2.4 and 3.5.4.

In fact, the formulations presented in [20, 25, 26] are special cases of our model: these formulations comprise two layers, and a routing of end-to-end demands on one type of grooming paths (lightpaths).

5 Sketch of an algorithmic approach

The model presented in Section 3 provides a detailed view on requirements of multi-layer network planning in practice. However, in order to be of practical use for network operators, the model must be suitable to compute satisfactory solutions for realistic problem instances.

Why do we believe that this formulation can be used for solving practical planning problems, even though it is more complex than all previous models for multi-layer network planning? Actually, we have been solving various single-layer network planning problems over many years. The corresponding models and algorithms (see [5, 12, 13, 22, 28], for instance) have been integrated into our network planning tool DISCNET [29] and work well for single-layer networks. As the model presented in this article is based on these developments, we have reason to believe that the used algorithmic approaches can also be used for solving multi-layer network design problems.

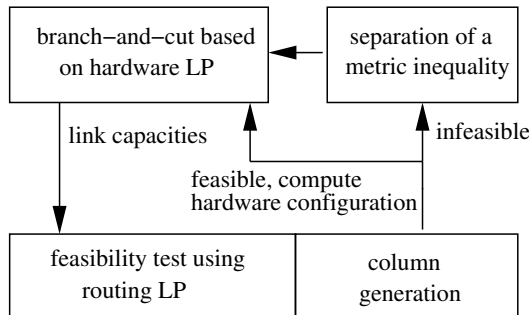


Figure 10: Algorithmic idea

The algorithmic idea, outlined in Figure 10, is similar to Benders decomposition [30]. At its core is a branch-and-cut algorithm (see [31] for a detailed description) based on the LP relaxation of the hardware model. No routing variables are used in this relaxation. Every time an integer capacity vector y is identified during the branch-and-bound process, it is tested for feasibility with respect to the missing routing constraints. In other words, the routing LP is solved to determine a feasible routing within the link capacities y if possible.

If such a routing exists, a hardware configuration based on y can be computed to obtain a feasible solution. In this step, practical requirements which have been relaxed in the model have to be taken into account, either in the form of an oracle which simply accepts or rejects a solution of the relaxation, or using a mechanism which transforms the solution of the relaxation into a feasible solution for the whole problem and/or provides some additional information in the case of infeasibility (such as a separation algorithm).

If no feasible routing exists, dual information from the routing LP can be used to separate generalizations [12, 28] of metric inequalities [32, 33] which describe the space of feasible capacity vectors y . After transformation to link design variables, these metric inequalities can be added to the core LP relaxation to cut off the infeasible capacities. Additional cutting planes can be separated to strengthen the LP relaxation, and heuristics can be used to identify further feasible solutions during the branch-and-cut process.

To solve the routing LP, one has to cope with the exponential number of variables associated with routing and grooming paths. It is definitely impractical to enumerate these variables. As a remedy, one can either artificially restrict the set of admissible paths (probably leading to suboptimal solutions [34]), or apply column generation to generate routing path variables on demand (see [12, 21, 28], for instance). The latter approach has proven to perform well if there are either no restrictions on the admissible path set, or if these restrictions are “nice” (e.g., hop limits, or nodes and links which a path must not employ).

The described approach is generic and a priori independent of the exact form of the routing constraints. The algorithm reflects the separation of our model into a hardware part and a routing part, linked to each other by the capacities. The most significant change compared to single-layer network design problems is the column generation procedure which has to be adapted to a multi-layer routing. Without going into the details here, the principal challenge of the multi-layer pricing problem is that end-to-end routing paths have to be priced out together with a division into sub-paths.

6 Conclusions

In this article, an integer linear programming model for the integrated cost optimization of multi-layer telecommunication networks has been presented. The model can be applied to different multi-layer planning contexts and technologies, provided that connection-oriented routing protocols are used and that realistic divisibility assumptions are satisfied by the routing units of the considered layers. It allows an integrated planning of topology, hardware, link capacities, routing, and survivability over an arbitrary number of layers, which is new in the literature. The various parts of the model have been discussed in detail, and it has been compared to previous approaches on multi-layer network planning. Finally, an outline of a possible algorithmic approach has been given.

Future research will be done based on the model in order to be able to solve practical multi-layer planning scenarios within acceptable computation times. In addition to studying the column generation problem for routing and grooming path variables, this includes the development of suitable heuristics to compute feasible solutions, and investigations on the polyhedral structure of the problem in order to identify further cutting planes which can be used within a branch-and-cut framework.

Acknowledgments

This work has been supported by the DFG Research Center “Mathematics for key technologies” (MATHEON), Berlin, <http://www.matheon.de>. We would like to thank our cooperation partners Peter Jonas (e-plus, Düsseldorf), Fritz-Joachim Westphal (T-Systems, Berlin), and their colleagues for fruitful discussions about technical requirements and modeling issues in multi-layer networks.

References

- [1] M. Grötschel, C.L. Monma, and M. Stoer. *Design of Survivable Networks*, volume Networks of *Handbooks in Operations Research and Management Science*, chapter 10, pages 617–672. North-Holland, 1993.
- [2] T.L. Magnanti, P. Mirchandani, and R. Vachani. Modeling and solving the two-facility capacitated network loading problem. *Operations Research*, 43(1):142–157, 1995.
- [3] D. Bienstock and O. Günlük. Capacitated network design – polyhedral structure and computation. *INFORMS Journal on Computing*, 8(3):243–259, 1996.
- [4] K. Murakami. *Survivable Network Management for High-Speed ATM Networks*. PhD thesis, Carnegie Mellon University, September 1995.
- [5] D. Alevras, M. Grötschel, and R. Wessäly. Cost efficient network synthesis from leased lines. *Annals of Operations Research*, 76:1–20, 1998.
- [6] D. Bienstock and G. Muratore. Strong inequalities for capacitated survivable network design problems. Technical report, Combinatorial Optimization Research Center, Columbia University, December 1997.
- [7] Y. Xiong and L. Mason. Restoration strategies and spare capacity requirements in self-healing ATM networks. In *Proceedings of the 16th IEEE Infocom 1997*, Kobe, Japan, pages 353–360, 1997.
- [8] G. Dahl and M. Stoer. A cutting plane algorithm for multicommodity survivable network design problems. *INFORMS Journal on Computing*, 10(1):1–11, 1998.
- [9] F. Poppe and P. Demeester. Economic allocation of spare capacity in mesh-restorable networks: Models and algorithms. In *Proceedings of the 6th International Conference on Telecommunication Systems (ICTSM 1998), Modeling and Analysis*, Nashville, USA, pages 77–86, March 1998.
- [10] J. Doucette and W. Grover. Influence of modularity and economy-of-scale effects on design of mesh-restorable DWDM networks. *IEEE Journal on Selected Areas in Communications*, 18(10):1912–1923, October 2000.
- [11] A. Balakrishnan, T. Magnanti, J. Sokol, and Y. Wang. Telecommunication link restoration planning with multiple facility types. *Annals of Operations Research*, 106:127–154, 2000.
- [12] R. Wessäly. *Dimensioning Survivable Capacitated NETWORKS*. PhD thesis, Technische Universität Berlin, April 2000.
- [13] S. Orłowski and R. Wessäly. Comparing restoration concepts using optimal network configurations with integrated hardware and routing decisions. In *Proceedings of the Fourth International Workshop on the Design of Reliable Communication Networks (DRCN 2003)*, Banff, Alberta, Canada, pages 15–22, October 2003.
- [14] P. Demeester, M. Gryseels, K. Van Doorselaere, A. Autenrieth, C. Brianza, G. Signorelli, R. Clemente, M. Ravera, A. Jajszczyk, D. Janukowicz, G. Kalbe, Y. Harada, S. Ohta, and A.G. Rhissa. Resilience in a multi-layer network. In *Proceedings of the First International Workshop on the Design of Reliable Communication Networks (DRCN 1998)*, Brugge, Belgium, May 1998.
- [15] A. Authenrieth. Differentiated multilayer resilience in IP over optical networks. In *Scuola Superiore G. Reiss Romoli, L’Aquila, Italy, TU München*, July 2002.
- [16] A. Kröller. Network optimization: Integration of hardware configuration and capacity dimensioning. Diploma thesis, Technische Universität Berlin, 2003. Available at <http://www.math.tu-bs.de/~ali/>.

- [17] W. Lai and D. McDysan. *Network Hierarchy and Multilayer Survivability*. IETF, November 2002. Internet RFC 3386, <http://www.ietf.org/rfc/rfc3386.html>.
- [18] Y. Xue. *Carrier Optical Service Requirements*. IETF, November 2002. Internet draft, <http://www.ietf.org/internet-drafts/draft-ietf-ipo-carrier-requirements%04.txt>.
- [19] E. Kubilinskas, M. Pióro, and P. Nilsson. Design models for robust multi-layer next generation internet core networks, carrying elastic traffic. In *Proceedings of the Fourth International Workshop on the Design of Reliable Communication Networks (DRCN 2003)*, Banff, Alberta, Canada, pages 61–68, October 2003.
- [20] J. Hu and B. Leida. Traffic grooming, routing, and wavelength assignment in optical WDM mesh networks. In *Proceedings of 6th INFORMS Telecommunications Conference*, Boca Raton, Florida, USA, April 2002.
- [21] G. Dahl and M. Stoer. A polyhedral approach to multicommodity survivable network design. *Numerische Mathematik*, 68(1):149–167, 1994.
- [22] A. Koster, R. Wessály, and A. Zymolka. Transparent optical network design with sparse wavelength conversion. In *Proceedings of the 7th Conference on Optical Networks Design and Modeling (ONDM 2003)*, Budapest, Hungary, pages 61–80, 2003.
- [23] R. Dutta and G.N. Rouskas. Traffic grooming in WDM networks: Past and future. *IEEE Network*, 16(6):46–56, 2002.
- [24] A. Knippel and B. Lardeux. The multi-layered network design problem. Submitted to EURJOR, Elsevier Science, March 2003.
- [25] D. Banerjee and B. Mukherjee. Wavelength-routed optical networks: Linear formulation, resource budgeting tradeoffs, and a reconfiguration study. In *Proceedings of the 16th IEEE Infocom 1997*, Kobe, Japan, pages 269–276, April 1997.
- [26] R. Dutta and G.N. Rouskas. A survey of virtual topology design algorithms for wavelength routed optical networks. *Optical Networks Magazine*, 1(1):77–89, January 2000.
- [27] P. Belotti and F. Malucelli. Network design with grooming constraints. In *Cologne-Twente Workshop 2004*, Milano, Italy, pages 51–55, June 2004.
- [28] S. Orłowski. Local and global restoration of node and link failures in telecommunication networks. diploma thesis, Technische Universität Berlin, February 2003. Available at <http://www.zib.de/orłowski>.
- [29] atesio GmbH, Rubensstr. 126, D-12157 Berlin, Germany. DISCNET, 2000–2004. Information available at <http://www.atesio.de>.
- [30] J. Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4(1):238–252, 1962.
- [31] M. Padberg and G. Rinaldi. A branch and cut algorithm for the resolution of large-scale symmetric traveling salesman problems. *SIAM Review*, 33:60–100, 1991.
- [32] M. Iri. On an extension of the maximum-flow minimum-cut theorem to multicommodity flows. *Journal of the Operations Research Society of Japan*, 13(3):129–135, 1971.
- [33] K. Onaga and O. Kakusho. On feasibility conditions of multicommodity flows in networks. *Transactions on circuit theory*, 18(4):425–429, 1971.
- [34] S. Orłowski and R. Wessály. The effect of hop limits on optimal cost in survivable network design. In G. Anandalingam and S. Raghavan, editors, *Telecommunications Network Design and Economics and Management*, 2004. To appear.