Lehrstuhl für Kommunikationsnetze
der Technischen Universität München

Design and Optimization of Resilient Multipath Networks

Claus Günter Gruber

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor–Ingenieurs (Dr.–Ing.)

genehmigten Dissertation.

Vorsitzender:

Univ.–Prof. Dr.–Ing. Ulf Schlichtmann

Prüfer der Dissertation:

1. Univ.–Prof. Dr.–Ing. Jörg Eberspächer
2. Univ.–Prof. Dr.–Ing. Ralf Lehnert,
   Technische Universität Dresden

Die Dissertation wurde am 12.12.2006 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 22.6.2007 angenommen.

Design and Optimization of Resilient Multipath Networks

Claus Günter Gruber

Munich, Germany

To my parents and my sister.

Only if an optimal solution is known,

suboptimal solutions can be accepted.

Acknowledgments

This thesis was written during my work as researcher and teaching assistant at the Institute of Communication Networks (LKN) at Munich University of Technology (TUM). A time I will always remember as a very interesting moment of my life where helpful colleagues have surrounded me. Many of them became friends.

Foremost, I sincerely would like to thank Prof. Dr.-Ing. Jörg Eberspächer, the institute's head and my Ph.D. advisor for his constant support and guidance throughout the work. While giving considerable advice he gave me the freedom to develop own ideas. He created an environment where it was a pleasure to conduct research. Furthermore, I would like to thank Prof. Dr.-Ing. Ralf Lehnert for being second examiner and Prof. Dr.-Ing. Ulf Schlichtmann for presiding the dissertation committee.

At the institute, my thanks go to my former colleagues. Especially, I would like to thank Dr.-Ing. Dominic Schupke who was the supervisor of my diploma thesis and encouraged me to proceed the work at the institute. During many hours he was a counterpart for fruitful technical and not-so-technical discussions. I also had the pleasure to work with Thomas Schwabe, Matthias Scheffel, Robert Prinz, Dr.-Ing. Achim Autenrieth, Thomas Fischer, and Dr.-Ing. Marie Tromparent that were members of the research group 'Photonic Networks and Network Resilience Group' at LKN. This research group provided the forum for many discussions about the future of resilient networks and topics included in this thesis. A special thanks go to Hans-Martin Zimmermann and Andrea Bör for the interesting discussions about mobile networks, e-learning strategies and topics across all layers.

In addition, I would like to express my thanks to Jochen Frings who introduced me to the world of telecommunication and with whom I co-founded a company. In this respect, further thanks go to Prof. Dr.-Ing. Jörg Eberspächer, Dr.-Ing. Martin Maier, Maren Jopen, and Stefan Thallmaier, who supported us considerably during the founding process.

Furthermore, it was a pleasure to work with Dr.-Ing. Joachim Charzinski and all participants of the research project 'Key Components for the mobile Internet of Next Generation' (KING). Additionally, I would like to thank Prof. Dr. Thomas Stidsen at DTU and Dr. Roland Wessäly, Dr. Arie Koster, Sebastian Orlowski, and Adrian Zymolka at ZIB for many discussions and the exchange of experience and ideas concerning network optimization and resilience.

Other special thanks go to my graduate students. Especially, the work with Svetoslav Duhovnikov, Wolfgang Mühlbauer, Matthias Wimmer, Tilo Eissler and Arno Schmid-Egger contributed to this thesis. Very special thanks go to Moritz Kiese, who contributed twice to the thesis and who I had the pleasure to advise him on his outstanding diploma thesis.

Last but not least, I would like to thank my family and all of my friends for their encouragement, support and the enjoyable time spent together.

Munich, December 2006 Claus G. Gruber

Abstract

An efficient and reliable communication infrastructure has become an important fundamental of our society. To guarantee the smooth transportation of data, transport networks have to fullfill strict quality of service and resilience requirements. The choice of the used resilience mechanisms has a substantial influence on capital and operational expenditures and is an important criterion when designing telecommunication networks.

This thesis investigates the cost-efficient design and planning of resilient transport networks. We analyze the network planning process and present the *Resilience Classification Framework* (RCF). This framework enables the systematic description, comparison, and analysis of any resilience mechanism. We perform example classifications and present novel resilience approaches that are able to react dynamically and quickly to traffic load changes and network equipment failures.

In the second part of the thesis, we analyze and assess approaches for the planning of resilient multipath networks. We present mathematical formulations based on linear programming that enable the cost-efficient optimization of resilient transport networks. Apart from complete formulations of flow- and path-based equation systems for promising multipath resilience mechanisms, we apply a new mathematical decomposition approach called *Column Generation* that enhances the planning of resilient networks considerably. With this technique, even very large resilient transport networks can be planned efficiently that cannot be optimized using classical approaches.

In order to provide more insights in resilience mechanisms and cost-optimal topology and path-selection, this thesis furthermore evaluates five popular path-based protection and restoration mechanisms. Next to a mechanism comparison using the RCF, we perform case-study optimizations and analyze results to deduct quantitative capacity requirements. Furthermore, we present recovery-time analysis results for OSPF and MPLS networks. We analyze the influence of multipath routing on capacity requirements in order to provide guidelines for the development of faster algorithms and heuristics for the planning of resilient networks.

Contents

Chapter 1

Introduction

"By 2007, U.S. enterprises engaged in e-business will have lost more than US$ 50 million in potential revenue as a result of network failures.", J.E. Pultz, Gartner Inc., Management Update: The 'New' Telecom Manager Redefines the Mission, Jan. 2004.

Broadband access technologies and new services (e.g. peer-to-peer file-sharing applications and video on demand) have let to a tremendous increase of data rates in transport networks. The deregulation of telecommunication markets and global competition however, force network providers to cut costs in order to have advantages compared to their competitors. Therefore, transport networks have to be constantly monitored, extended, adapted, and optimized. Thus, network planning has changed from a rare task of extending the network to an almost daily task of thorough reoptimization. Case-study and what-if calculations have to be performed and reconfigurations of traffic flows combined with network extensions have to be planned frequently.

Transport networks are migrating towards a converged, flexible, and cost-efficient packet based infrastructure. The introduction of connection-oriented switching technologies, like *Multi Protocol Label Switching* (MPLS) [RVC01] or *Carrier Grade Ethernet* (CGE) [**KGRB06**], provide increased control, management, and traffic-engineering possibilities. These technologies enable an efficient and dynamic routing of traffic flows using multiple paths towards a destination.

In the past, the guarantee of traditional *Quality of Service* (QoS) characteristics like minimum bitrate, maximum delay, maximum delay variation (jitter), or maximum packet loss ratio were of main interest during the network planning process. However, since the society has become more and more dependent on telecommunication, the availability of a service has evolved as one of the main issues of network planning today.

Highly available networks can be obtained by adding redundancy and applying mechanisms that react in case of network element failures. In the past and even often today, telecommunication networks have been designed and optimized to perform well in failure free cases. Redundancy is added afterwards in a second design process. As an example, a recent tender for the deployment of a new nationwide connection-oriented Ethernet transport network mentions availability in a few sentences only and asks for a very bandwidth

inefficient type of redundancy (1+1 dedicated path protection). However, availability will be dearly bought if redundancy is added afterwards. The costs of highly available networks would drastically be reduced if availability deliberations and resilience mechanisms were already included at the beginning of the network design process and in parallel to traditional network planning tasks.

This thesis contributes to three significant research areas in the field of resilient network design and planning. These are:

- **Resilience Classification**: For the optimization of cost-efficient networks, resilience mechanisms have to be identified that are well suited for the intended purpose. A large number of resilience mechanisms exist today. This thesis presents the *Resilience Classification Framework* (RCF) that enables the systematic description, comparison, and analysis of any resilience mechanism. This analysis of resilience mechanisms facilitates the characterization, comparison of existing, and the development of new resilience mechanisms.

- **Resilient Network Optimization**: Network providers have to deploy cost-efficient network planning solutions that utilize multipath and traffic-engineering possibilities of new switching and routing technologies in order to be competitive in the market or even corner it. This thesis presents new optimization approaches for the planning of resilient multipath networks with that even very large resilient transport networks can be planned efficiently that cannot be optimized using classical approaches.

- **Resilience Mechanism Evaluation**: Resilience mechanisms have to be understood to provide the intended availability. Additionally, capacity requirements and recovery time are important characteristics that have to be investigated and compared in order to select suitable resilience mechanisms right from the beginning and to provide guidelines for the deployment of resilient telecommunication networks. This thesis evaluates five popular path-based protection and restoration mechanisms. We perform case-study optimizations and analyze results to deduct quantitative capacity requirements. Recovery-time analysis results for OSPF and MPLS networks are provided and the influence of multipath routing on capacity requirements is analyzed.

The remainder of this thesis is organized as follows:

Chapter 2 gives an overview about the traditional network planning process and discusses network design objectives. Consequently, we analyze resilience requirements and describe the benefits and drawbacks of multipath routing. We discuss the interdigitation of topology planning, multipath routing and dimensioning as well as the requirement for a joint optimization of failure-free and failure-affected network states.

Chapter 3 introduces a novel classification framework consisting of eight building blocks with which resilience mechanisms can be described independently of technology issues. The *Resilience Classification Framework* enables the finding of mechanism-characteristics, allows a mechanism comparison, and provides the basis for a decision towards the best fitting resilience mechanism. Furthermore, due to the decomposition into building blocks, dependencies and new combinations of characteristics can be identified and the design of novel resilience mechanisms is facilitated. To illustrate the advantages of the framework, we present example classifications and comparison of different resilience mechanisms and present a novel resilience mechanism called *Self Regulating Traffic Distribution* (SRTD) that is able to react dynamically and quickly to traffic load changes and network equipment failures.

Chapter 4 discusses network optimization approaches that exist in the literature. We show, however, that most of these optimization approaches do not provide information about the quality of the obtained solution. After introducing linear programming, we develop and present novel linear programming models for the resilient network design. In particular, we present complete models for five path-based resilience mechanisms. Furthermore, we apply the mathematical decomposition technique *Column Generation* in order to reduce optimization time and requirements on calculation ressources.

Chapter 5 provides an evaluation of selected resilience mechanisms and gives guidelines for the choice towards suitable resilience mechanisms. Based on optimization approaches of Chapter 4 we optimize case study networks and deduct quantitative capacity requirements. Furthermore, we present recovery time analysis for OSPF and MPLS networks and provide simulation results as well as theoretical recovery time formulas.

Finally, Chapter 6 comprises a conclusive summary and lists the research contributions of the thesis.

Appendix A defines important resilience terminologies. Appendix B provides background information about alternative meta-heuristic optimization approaches *Simulated Annealing* and *Genetic Algorithm*. Appendix C specifies sets, variables, and parameters that are used in the resilient network optimization formulas.

Chapter 2

Principles of Resilient Network Design and Planning

Optimal resilient network planning is the task to design a robust network that enables the transportation of data with strict Quality of Service requirements, lowest possible cost, and highest possible availability.

In this chapter, we investigate and analyze the network planning process in detail and present the individual tasks that have to be performed. We furthermore focus on the interdigitation of traditional network planning and resilience and present a combined resilient network design process in which failure-free as well as failure-affected network states are investigated and optimized in a joint manner.

The chapter is organized as follows: Section 2.1 introduces the traditional network design and planning process, discusses the objectives during these processes, and highlights the benefits of joint network planning and multipath routing. Following this, two types of network architectures are described in Section 2.2 that are used in the remaining chapters of this thesis. Section 2.3 investigates the requirements that arise for resilient network planning in more detail. Finally, Section 2.4 concludes this chapter and summarizes the key findings.

2.1 Network Design and Planning

The network design and planning process can be divided into three phases that are illustrated in Figure 2.1: Requirement Analysis, Network Design, and Network Planning.

The analysis and exact definition of requirements that the network should fulfill are one of the most important and difficult, however, often underestimated issues of the network design and planning process. Main objectives, technology, business or market related constraints, user demands, as well as performance parameters have to be determined and defined carefully. In this process, at least two perspectives have to be taken into account: The view of the network provider and the view of the network customer. Customers, as

opposed to providers, only care about those service characteristics they experience directly. Often, they are not concerned with and do not care about transport technology or network management issues. Instead, they are concerned with their perceived Quality of Service. Even worse, some characteristics (e.g. delays of some tens of milliseconds that cause difficulties for distributed gaming applications) might cause a fluctuation of customers, even if not initially required or defined in a Service Level Agreement (SLA). In addition, the perception of quality can vary significantly between different customers and can sometimes only be vaguely described in technical or legal terms.

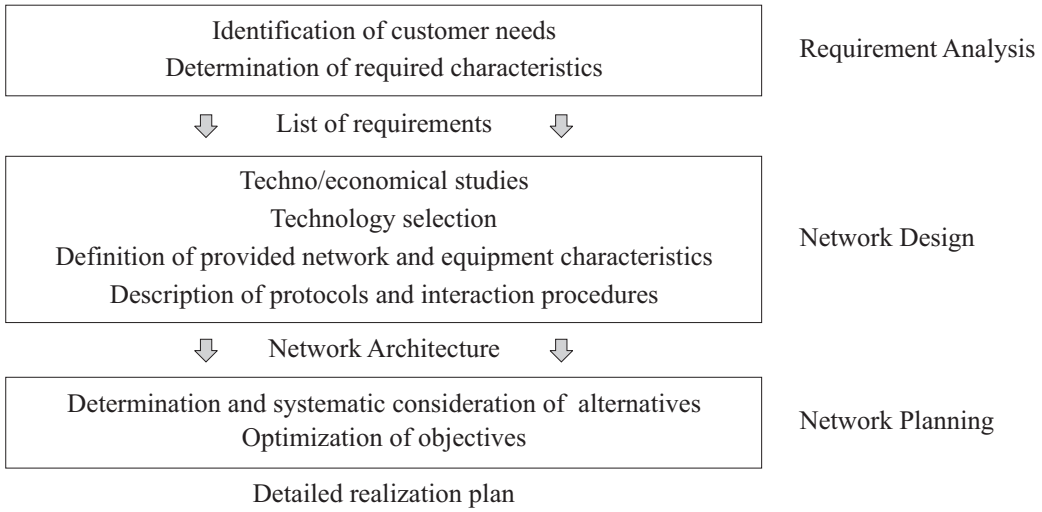


Figure 2.1: Overview of requirement analysis, network design, and network planning.

On the other hand, manageability, predictability, security, scalability, and the ability for real-time performance monitoring are key concerns of network providers. However, above all, the trade-off between cost, performance, and availability are the driving forces and have to be balanced thoroughly.

After the definition of requirements, the network design phase can be initiated in which the general network architecture is determined. Often, a large number of realization possibilities exist to fulfill the requirements. Next to an evaluation and selection of technologies or technology combinations by using sophisticated algorithms and simulations, techno-economical studies have to be performed to define suitable alternatives for the network architecture. Possible routing and resilience mechanisms have to be assessed and different network scenarios (topologies and traffic matrices) have to be analyzed. Furthermore, management, interoperability, and migration strategies have to be determined. Overall, network design defines alternatives and possibilities, selects technologies, and provides basic engineering rules for network planning.

Once possible network architectures are defined, network planning determines exact equipment choices and deployment plans for specific networks by performing a systematic consideration of the defined alternatives. Based on detailed equipment databases and prices sophisticated algorithms and optimization procedures are performed to determine an

optimal or good solution. Detailed equipment order-lists, complete equipment and protocol specific configurations, or even deployment schedules are outputs of network planning.

2.1.1 Objectives of Network Design and Planning

Often a trade-off between different contradicting objectives is aspired when designing and planning telecommunication networks. In the following we will summarize the most important objectives for network design which are used today by network providers and network equipment vendors.[1]

2.1.1.1 Cost

One of the main requirements when building communication networks is to reduce costs. Today's existing telecommunication networks have been mainly optimized to reduce the overall capital expenditures (CAPEX), i.e., expenditures used by a company to acquire or upgrade physical assets such as equipment, property, and industrial buildings [Inc06]. However, since rent of floor space, costs of electrical power consumption, and costs for well-trained human resources are rising, more and more emphasis is placed on operational expenditures (OPEX) today, i.e., on-going costs for running a product, business, or system [Inc06]. Table 2.1 lists typical components of CAPEX and OPEX. An overview and detailed analysis of CAPEX as well as OPEX cost factors can be found in [Mas06], [VCP+06] and [KIWP06].

Table 2.1: Typical components of capital (CAPEX) and operational expenditures (OPEX).

CAPEX	OPEX
Network equipment, e.g. routers and transmission equipment	Leased infrastructure, e.g floor space, rights of way
Cable/fiber	Power consumption
Civil works	Human resources and salaries
Customer premises equipment	Network management, service provisioning, and operational network planning
First time installation	Maintenance, inventory, and repair
Replacement equipment	Marketing
Licenses and permits	Administration and overhead

2.1.1.2 Quality of Service

Next to cost, the quality of a service is of main concern for network providers. Customers will note bad quality immediately and will change the network provider quickly, if they

[1]Next to network providers, also vendors of network equipment are called upon performing detailed design and planning studies in order to provide reasonable tenders based on their product portfolio.

experience (long lasting) bad network performance. Different definitions of QoS exist in the literature, e.g.:

Internet Engineering Task Force (IETF) [CNRS98]: "*Quality-of-Service: A set of service requirements to be met by the network while transporting a flow.*"
International Telecommunication Union (ITU-T) [IT94]: "*Quality of Service is the collective effect of service performances, which determine the degree of satisfaction of a user of the service.*"
European Communication Standards Institute (ETSI) [ETS03]: "*Totality of (network and non-network related) characteristics of a telecommunication service on its ability to satisfy stated or implied needs.*"
Alliance for Telecom Industry Solutions (ATIS) [All00]: "*1. The performance specification of a communications channel or system. Note: QoS may be quantitatively indicated by channel or system performance parameters, such as signal-to-noise ratio (S/N), bit error ratio (BER), message throughput rate, and call blocking probability. 2. A subjective rating of telephone communications quality in which listeners judge transmissions by qualifiers, such as excellent, good, fair, poor, or unsatisfactory.*"

In general however, two types of QoS can be distinguished: *Intrinsic* and *Perceived Quality of Service* [Har01a, IT01]. On the one hand, the intrinsic quality of service defines 'technical quality' including guarantees on bitrate, packet loss probability, end-to-end delay, and delay variations. On the other hand, the 'perceived quality of service' models the degree of satisfaction of the customer. Obviously, both types are of importance for network operators since both will have an effect on the level of acceptance by the customers. Thus, the chosen intrinsic quality of service parameters, that are often included in SLAs between customers and network providers, should suit the experienced 'believed' QoS by the customer. Acceptable intrinsic parameters are thus a matter of negotiation between a customer and a network provider and are dependent on technology and application.[2] Additionally, often today, different parameter-sets (Classes of Service, CoS) are used concurrently in a network and traffic has to be treated differently in order to achieve the desired QoS parameters. Table 2.2 shows classes of services recommended for IP networks by the International Telecommunication Union [IT06b].

[2]Guidelines to score customer's expectations are e.g. given in [IT01] and [ETS03].

Table 2.2: IP network QoS class definitions and network performance objectives defined in [IT06b]. Currently, QoS classes 6 and 7 are provisional and are not included in this table.

Nature of network performance	QoS Class 0	QoS Class 1	QoS Class 2	QoS Class 3	QoS Class 4	QoS Class 5
Upper bound on packet transmission delay (IPTD) [IT02]	100 millisecond	400 millisecond	100 millisecond	400 millisecond	1 second	n/a
Upper bound on the $1-10^{-3}$ quantile of IPTD minus the minimum IPTD	50 millisecond	50 millisecond	n/a	n/a	n/a	n/a
Upper bound on the packet loss probability	10^{-3}	10^{-3}	10^{-3}	10^{-3}	10^{-3}	n/a
Upper bound of packet error ratio	10^{-4}	10^{-4}	10^{-4}	10^{-4}	10^{-4}	n/a

2.1.1.3 Availability

Although it can be considered as a QoS characteristic as e.g. in [Aut02], [TFP+03] and [TCC+05], service availability and resilience mechanisms, i.e. countermeasures to sustain the impact of network element failures to maintain an acceptable level of functionality and structure, are not part of the 'classical' QoS requirements. However, the implications of unavailable service can be disastrous considering impeded emergency calls or delayed business communications. Table 2.3 gives an overview of estimated costs of downtime for different industries.

Table 2.3: Estimation of downtime costs. Taken from [Pat02] and [BS04] based on Contingency Planning Research and Gartner/Dataquest.

Business operation	Industry cost range per hour of downtime (US$)	Average cost per hour of downtime (US$)
Brokerage operations	5.6 to 7.3 million	6.5 million
Credit card / sales authorization	2.2 to 3.1 million	2.6 million
Pay-per-view television	67,000 to 230,000	150,000
Home shopping (TV)	87,000 to 140,000	113,000
Home catalog sales	60,000 to 120,000	90,000
Airline reservations	67,000 to 112,000	89,500
Tele-ticket sales	56,000 to 82,000	69,000
Package shipping	24,000 to 32,000	28,000
ATM fees	12,000 to 17,000	14,500

Downtime can cost several hundreds of thousand dollars per hour for e-commerce sites like amazon.com or eBay.com or can cost even up to some million dollars per hour for impeded stockbroker operations [BP01]. However, next to lost production hours, increased cost, lost revenue or profit, the unavailability of a service can even impose long-term implications considering unwanted press attention, employee dissatisfaction, or dissatisfied customers. This can even lead to the long-term loss of customers and brand erosion. Thus, providing high availability is - next to reducing cost and providing classical QoS characteristics - one of the main objectives of network design and planning.

2.1.1.4 Complexity and Manageability

The complexity of a system directly reflects the difficulty to run it. Therefore, devices, forwarding technologies, routing algorithms, or resilience mechanisms should be as simple as possible in order to facilitate the management of the system. Especially the task of reconfiguring traffic flows or the task of applying adaptations due to traffic growth will be troublesome if the complexity of the system can no longer be handled. Although many issues of failure management, performance management, configuration management, accounting/billing, or software management can be automated using sophisticated software, simple structures are preferred by network operators [Gro04].

2.1.1.5 Security and Misconfiguration

Finally yet importantly, a network has to be protected from misconfiguration and malicious attacks. Consequently, the network has to be designed in a way that simple or automated procedures can be applied by operational personnel. Additionally, security mechanisms and countermeasures have to be thoroughly tested and personnel have to be well trained in order to reduce the probability and negative effects of system misconfiguration and security threats.

2.1.2 Network Planning Cycle

Once requirements, expectations, the right balance between different planning objectives, and possible network architectures have been defined and user demands have been forecasted, the network planning process can be initiated. Traditionally and due to its complexity, the network planning process is divided into several steps. A typical network planning cycle is illustrated in Figure 2.2.

2.1.2.1 Technology Choice

At the beginning of the planning process, design concepts for traffic forwarding, configuration, and management have to be modeled in order to obtain feasible network constellations for the chosen architectural concepts and technologies. Often, this detailed modeling reveals additional restrictions and constraints for the following planning steps, as e.g. a maximum length of transparent routes and specific traffic transport granularity (2.5, 10,

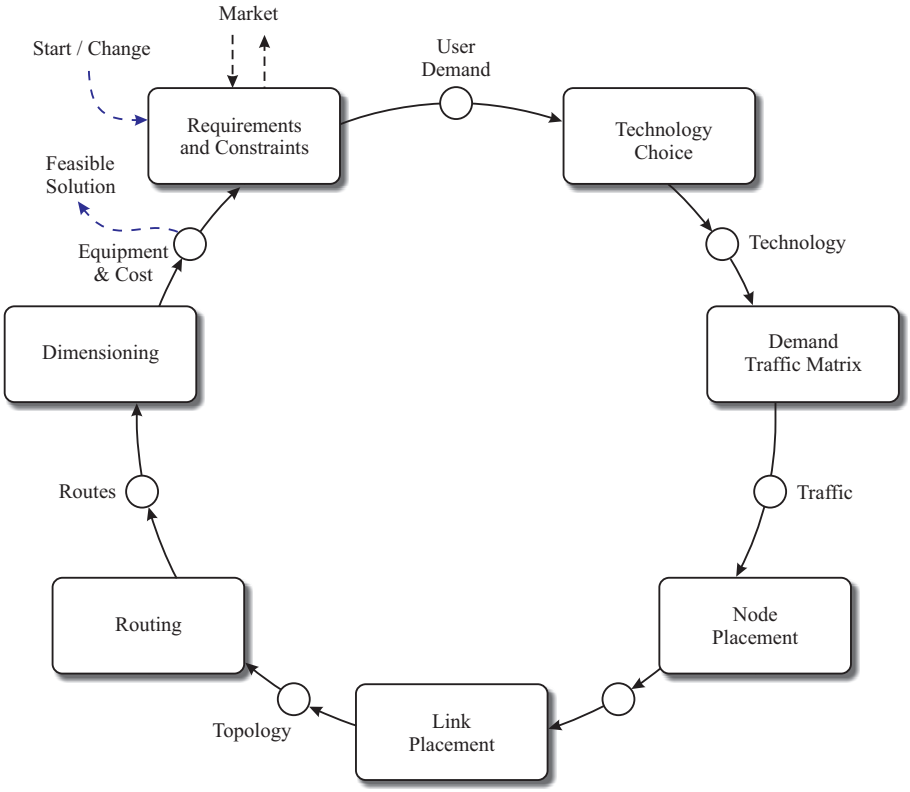


Figure 2.2: Network Planning Cycle.

or 40 Gbit/s) for optical *Dense Wavelength Division Multiplex* (DWDM) networks or a maximum label size and switching capacity for *Multi Protocol Label Switching* (MPLS) networks.

2.1.2.2 Demand Matrix Estimation

Following the selection of transport technologies, the demand matrix, i.e. a matrix indicating how much traffic is sourced at any location A and terminated at any other location B of the network, has to be determined. While an actual traffic matrix can be estimated using node or link load measurements [GJT04], providing accurate forecasts of future trends and applications is a discipline of its own. However, historic, economic, and demoscopic information can help to estimate user demands [HBB+04]. Additionally, characteristics of technologies (e.g. an increase of required demand due to packet overheads) and probability calculations to estimate peak bitrates or blocking probabilities have to be taken into account [Rie04]. Furthermore, if services have different requirements (e.g. strict delay requirements), several demand matrices can be obtained that can be used in the following routing and dimensioning process.

2.1.2.3 Network Topology

Subsequently, the network topology, i.e. possible nodes and links of the network, have to be determined. For transport networks, major interconnects and *Points of Presences* (PoPs) have to be determined that provide a backbone for the connection of regional and metro networks. Clustering algorithms that take user demand values and the distribution of potential customers over the service area into account [Rob99] are often used. In most cases, however, possible node locations are already pre-determined by existing real estate properties, main cities, or major network exchange points such as the *Deutscher Commercial Internet Exchange* (DECIX), located in Frankfurt/Main, Germany, that connects over 190 network providers with each other.

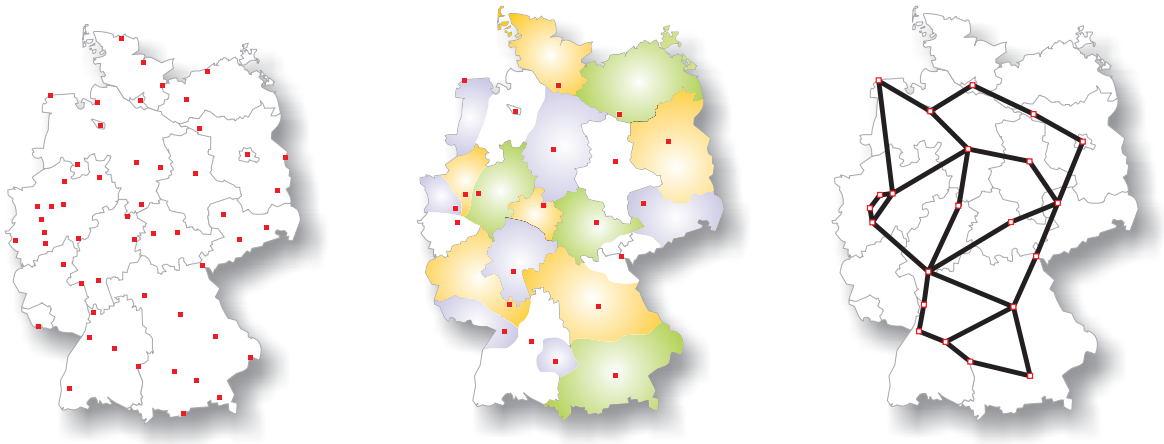


Figure 2.3: Example network topology. Left: Potential backbone nodes. Center: Clustering of nodes. Right: Chosen network topology.

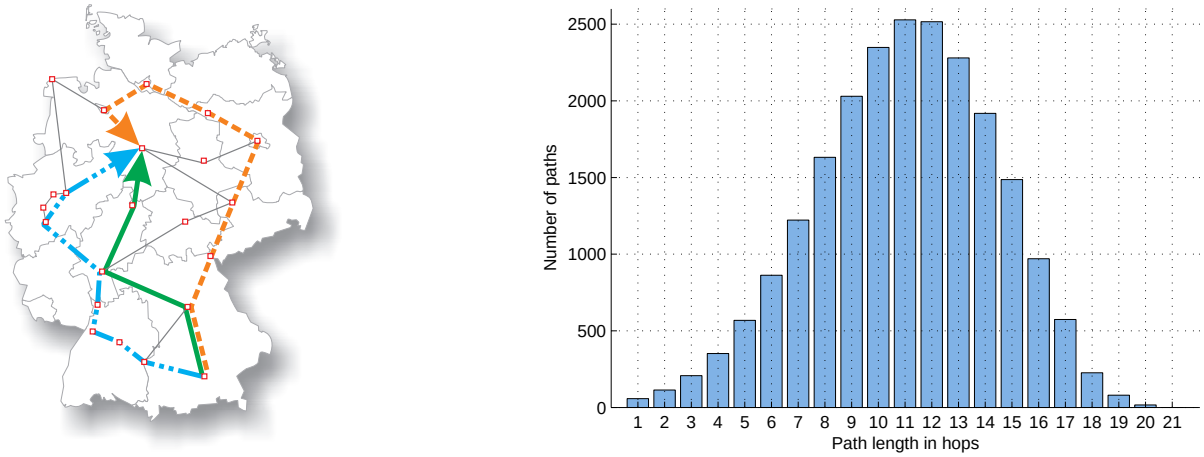
After backbone nodes have been selected, suitable interconnections of locations have to be determined. While star, tree, and ring structures are typically chosen in access networks [ZGL05], transport networks are nowadays often designed in a mesh-like manner to allow more flexibility for routing and resilience [CGLS01]. From a CAPEX point of view, i.e. digging costs and rights of way, the number of edges[3] should be small. However, concerning the interdigitation of topology with routing and dimensioning, one has to be careful not to build a too sparsely meshed network. Typical node degrees, i.e. the average number of edges connected to a node, vary between 2.3 in some north American networks to up to 4.5 in some European networks [SND06].

2.1.2.4 Routing

Once technology and demand matrices have been defined, one or more routes for each demand have to be determined. Since traffic can usually be routed via multiple transit nodes the number of possible routes is immense (see Figure 2.4). Additionally, the choice

[3]In this thesis we will refer to the term *edge* and *link* synonymously.

of routes is furthermore hampered by technology or protocol restrictions such as length limits of routes (e.g. maximum transparent optical reach) or dependencies between routes of different demands (e.g. by routing methods according to global link or port weights). However, choosing the right set of routes is one of the key factors for cost-efficient network planning.



(a) Three different paths.

(b) Number of total paths in the network.

Figure 2.4: Example network topology showing possible paths.

2.1.2.5 Dimensioning

In order to realize the required routes, equipment has to be installed and sufficient capacity has to be provided on edges and nodes. Thus, suitable equipment has to be chosen and has to be placed appropriately during the task of network dimensioning. Since equipment is often shipped in modules with different granularity, economy of scale effects have additionally to be taken into account in order to find cost-optimal solutions. Figure 2.5 illustrates a node model allowing different line cards and line technology combinations.

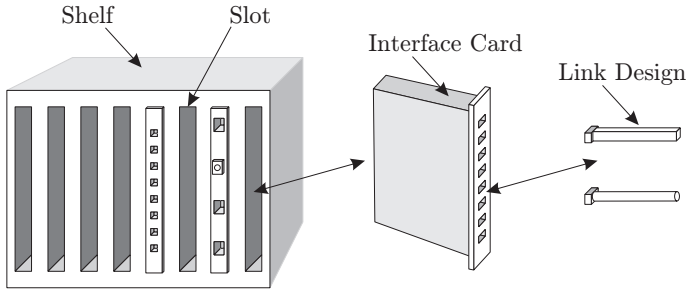


Figure 2.5: Illustration of a node model with different module combinations.

2.1.3 Joint Optimization

The individual steps of the network planning cycle of Section 2.1.2 can be performed separately from each other. However, if no feasible or good solution can be found in one step, changes in previous steps will be required (e.g. an adaptation of the topology if no feasible route can be found during *Routing*, or an adaptation of the technology if the demand cannot be routed with available equipment). Thus, often, planning steps have to be repeated or the network cycle has to be processed several times to find a feasible solution. Dependencies between individual planning steps and eventually required repetition of previous steps are illustrated in Figure 2.6 as dotted lines.

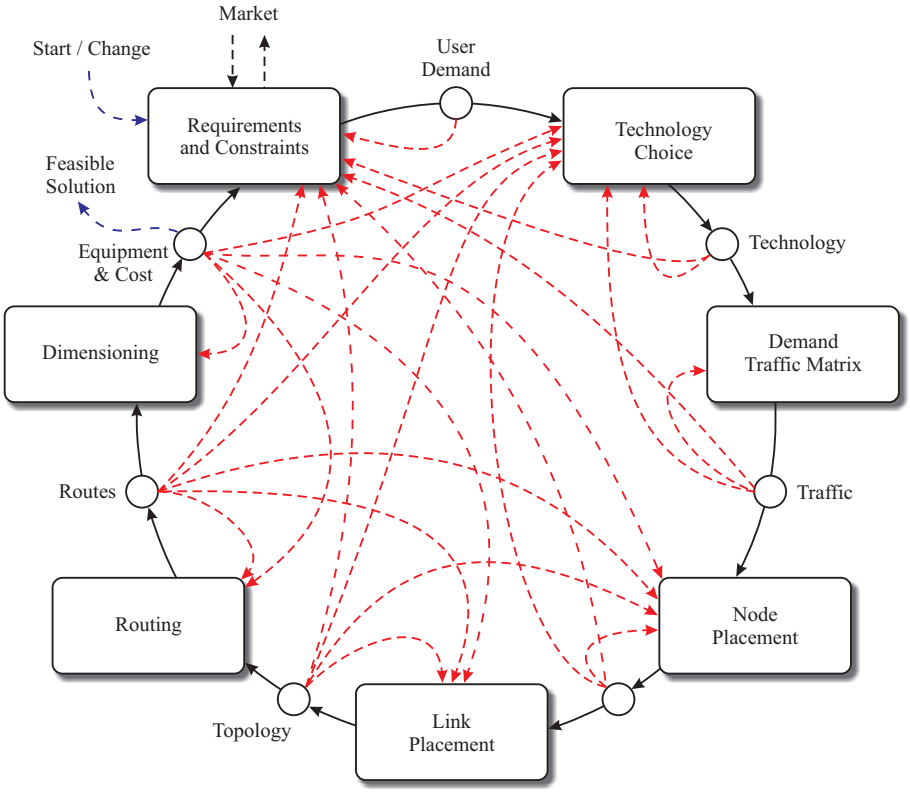
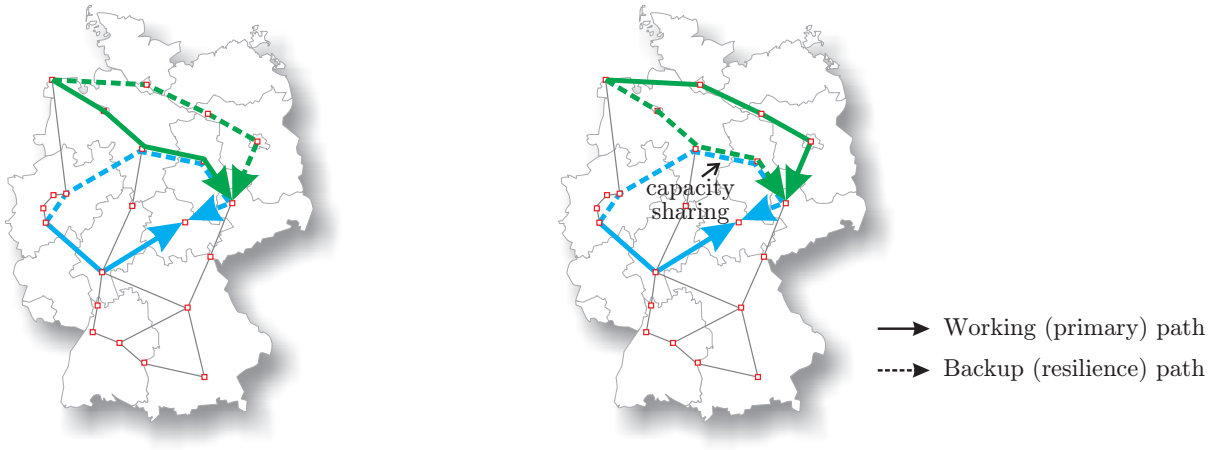


Figure 2.6: Typical network planning cycle. Iterative steps and interactions (dotted lines).

Furthermore, drastic improvements of the solution quality can be achieved by combining individual steps. Dimensioning, for example, is very much dependent on routing. Cheaper solutions will be found if information of equipment modules and their costs are already available during the routing process. The interdigitation of topology decision and routing is another example: The more edges, the more paths are available between two nodes. Thus, even though computationally complex, the individual steps of defining network topology, routing and dimensioning should be performed jointly, if possible.

As mentioned previously, in many network-planning procedures today the addition of resilience mechanisms, i.e. the determination of backup paths and the required dimensioning adaptation, is done in a separate design process. The network is designed and

dimensioned for the failure-free state while backup equipment and routes are added afterwards to survive the most probable failures. However, especially when using resilience mechanisms that are able to share backup paths, the joint optimization of failure-free and failure-affected scenarios is advantageous from a cost as well as from a performance point of view. Figure 2.7 depicts example results using an iterative and a joint optimization approach of failure-free and failure-affected network planning. The backup paths of different demands can be aligned and capacity can be shared with joint optimization. Therefore, in this thesis we present calculation approaches and network planning solutions in which both failure-free and failure-affected network states, routing and dimensioning are taken into account in a joint optimization process.



(a) Iterative planning. (b) Joint planning.

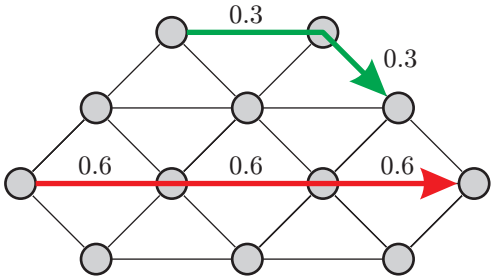
Figure 2.7: Example path constellation for iterative and joint network planning.

2.1.4 Multipath Routing

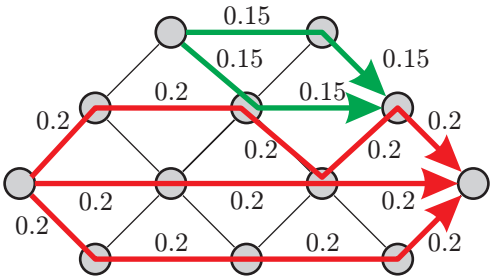
The evolution from ring-based networks towards mesh-like networks enlarges the number of possible paths between two locations significantly. To avoid overload situations of individual network elements, an intelligent combination of different routes (traffic-engineering) can be used to distribute the traffic evenly in the network. Especially, traffic demands with a large amount of traffic between two locations are difficult to route since enough capacity has to be provided on all components along the path. However, the number of paths with sufficient capacity can be increased by splitting a demand into several parts and by assigning each part a different route towards the destination node.

An example of multipath routing[4] is depicted in Figure 2.8. While some links are highly utilized without multipath routing (a), the traffic can be distributed evenly using multiple routes (b). Thus, if new demands have to be routed or demands increase, enough capacity

[4]Multipath routing is also called *inverse multiplexing* in the literature [Dun94].



(a) Without multipath routing. Maximum link load: 0.6; required capacity: 2.4 traffic units.

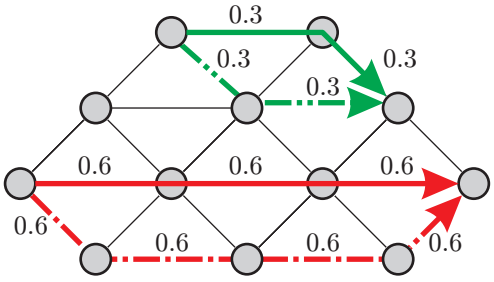


(b) With multipath routing. Maximum link load: 0.2; required capacity: 3.0 traffic units.

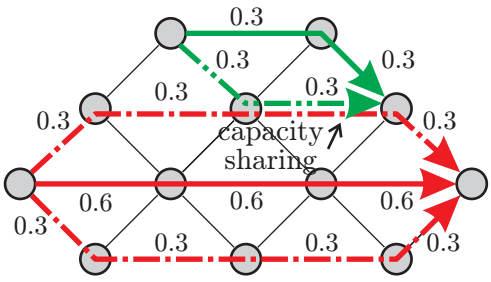
Figure 2.8: Example maximum link loads for multipath routing. All link capacities are 1.

will be available on all network elements. However, since some paths can be longer, the total amount of required capacity and thus cost can increase when using multipath routing (from 2.4 to 3.0 in the example).

While multipath routing is beneficial for traffic-engineering in failure-free networks, it is even more so for resilient network planning. Figure 2.9 depicts an example in which two demands are routed and protected against single link failures.[5]



(a) Without multipath routing. Maximum link load: 0.6; required capacity: 5.4 traffic units.



(b) With multipath routing. Maximum link load: 0.6; required capacity: 5.1 traffic units.

Figure 2.9: Example maximum link loads for resilient multipath routing. All link capacities are 1.

If different demands can be split into similar amounts, capacity will be efficiently shared between resilience paths of demands that are unlikely to fail simultaneously, i.e. which are routed disjoint in the failure free network. Thus, less capacity is required for multipath routing than for single path routing (reduction from 5.4 to 5.1 traffic units in the example).

[5]More details on the used protection scheme called '*Demandwise Shared Path Protection*' (DSPP) are given in Chapter 3.

However, multiple paths require more complex network nodes. Additional entries in forwarding[6] tables have to be applied and scheduling mechanisms have to be deployed that distribute traffic to multiple links.

When using multiple paths to transport a demand between two nodes, one has to be careful not to route traffic belonging to one flow via different paths. An example is illustrated in Figure 2.10. A sequence of incoming packets is distributed by node A to two different paths towards destination B. The first and third packet is forwarded along the lower path while the second packet is forwarded along the upper path in the example. Since the lower path is shorter, i.e. imposes less delay, the third packet overtakes the second packet. Since packet reordering severely degrades the throughput of transport layer protocols such as TCP [LG02, BA02], multipath routing is an issue for transport networks. Thus, sophisticated scheduling mechanisms have to be deployed to assign traffic of one flow to one path only. Simultaneously, however, incoming traffic of different flows has to be distributed on multiple paths.

R. Martin et al. [MMH06] provide a detailed overview, simulation studies, and a comparison of different load distribution mechanisms. Today, different load balancing mechanisms are deployed: Packet-based load-balancing, e.g. round robin, per-flow state load balancing, and hash-based flow balancing. However, dynamic adaptation procedures are required to achieve acceptable accuracy of traffic distribution, i.e. adapting the scheduling mechanism in order to achieve the defined traffic split/distribution ratios.

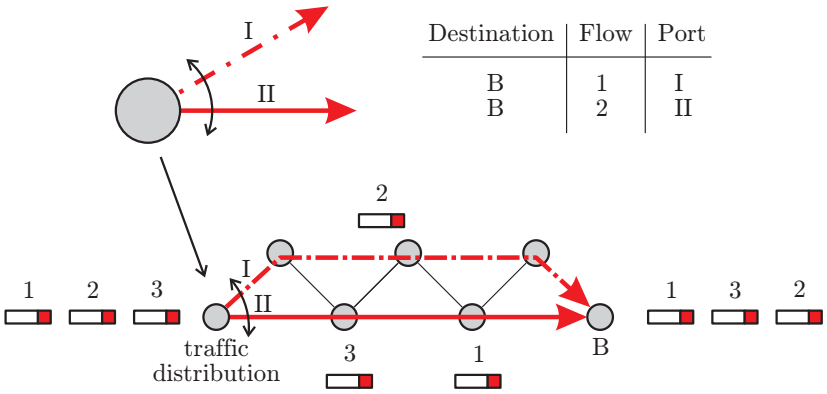


Figure 2.10: Example of packet reordering caused by multipath routing of one traffic flow.

In summary, multipath routing provides promising benefits for backup capacity reduction and traffic-engineering. However, many implications arise from the scheduling of packets and the distribution of traffic to multiple paths. Although many technologies are able to use multiple paths, little is known about the real benefits of this approach for resilient network planning. Thus, in the following chapters, we will investigate the benefits of multipath routing both in the failure-free as well as in failure affected case and will analyze the need for multipath routing from a network-planning point of view.

[6]In this document we will refer to the term *forwarding* as to the common operation that is either done by routers or switches: To forward a data-packet from one incoming port to one outgoing port according to information that is included in the packet header and/or node-internal forwarding rules.

2.2 Network Architectures

In the last two decades, pure connection-oriented voice networks were transformed to connection-less packet based data networks. However, due to the increased control, management, and traffic-engineering possibilities nowadays, connection-oriented forwarding is reintroduced and a mixture of both technologies is predominant in today's transport networks.

In the following paragraphs, we introduce and focus on the IP based layer 3 routing mechanism *Open Shortest Path First* (OSPF) and the layer 2 switching technology *Multi Protocol Label Switching* (MPLS) and will use them as example technologies in this thesis. Additionally, to understand convergence-time studies, that will be presented in Chapter 5, detailed information is given about failure detection and reaction possibilities of both OSPF and MPLS. However, the dominant part of the thesis can also be applied to other routing protocols such as IS-IS [Kat00]), optical or SDH/SONET architectures [IT00] and next generation connection-oriented Ethernet switching technologies like VLAN Cross-Connect [**KGRB06**, SBLL06] or Provider Backbone Transport (PBT) [**KGRB06**, FAS+06a, FAS+06b, SEKE06].

2.2.1 Connectionless Routing

2.2.1.1 OSPF

The *Open Shortest Path First* protocol is the mostly deployed *Interior Gateway Protocol* (IGP) today. It is a link state routing protocol designed to be run internally in a single *Autonomous System* (AS) or OSPF area. Its current version 2 is defined in IETF RFC 2328 [Moy98].[7] A link state protocol is based on a distributed map concept. Each OSPF router that belongs to the same area maintains an identical database describing the system's topology. Based on the topology-view each router individually calculates and constructs a shortest path tree yielding the shortest possible path(s) from the router towards any destination inside the Autonomous System and towards border gateway routers (and thus to remote destinations in other Autonomous Systems).

The OSPF protocol architecture can be divided into four parts:

- The detection of topology and topology changes,
- the distribution and storage of the network topology,
- the calculation of a shortest path tree,
- the configuration of the *Forwarding Information Base* (FIB).

[7] Additional extensions of the protocol, e.g. the support of QoS routing, the support of IPv6, the support of multicast, and traffic-engineering are standardized in RFC 2676 [WKG+99], RFC 2740 [CFM99], RFC 1584 [Moy94], and RFC 4203 [KR05] respectively.

Topology Detection: The *Hello Protocol* is responsible for establishing and maintaining relationships between adjacent routers. Each router periodically sends a Hello packet on its outgoing interfaces that are received by their adjacent routers. If no Hello packet is received within the configurable *Router Dead Interval* the link between the interfaces of the two routers is declared erroneous as illustrated in Figure 2.11. Concerning the crucial timing issues, the interval between the sending of Hello packets as well as the Router Dead Interval must be equal in the whole network [Moy98].

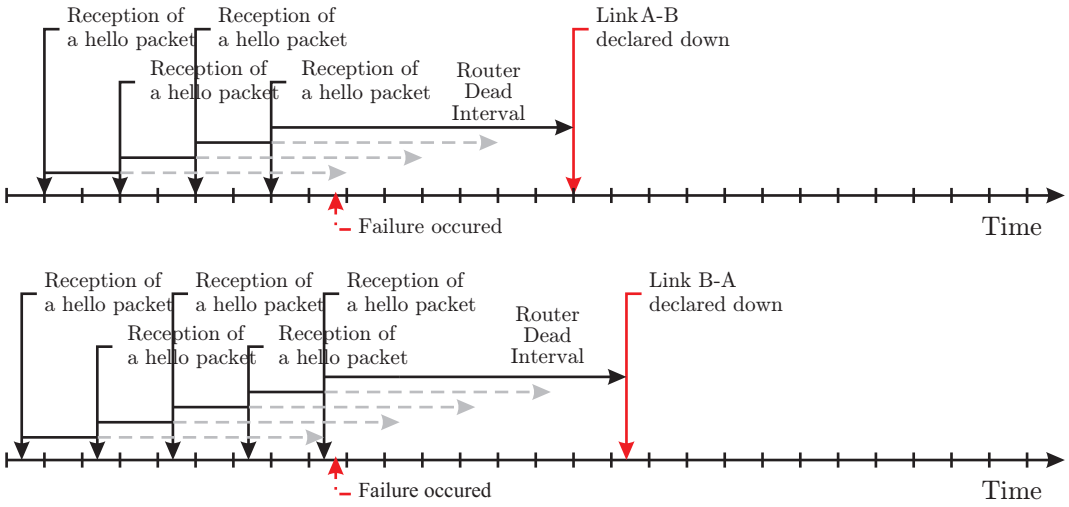


Figure 2.11: Hello timer and Router Dead Interval.

Distribution and Storage of the Network Topology: The topology information of an OSPF area is distributed between all routers by exchanging *Link State Advertisements* (LSAs) and is stored within each router individually in an LSA database. Each LSA represents one link of the network. Adjacent routers synchronize their databases via a reliable (i.e. acknowledged) exchange of *Link State Updates* (LSUs, bundles of LSAs). Thus, in a stable state, each router in an AS (or in the OSPF area) maintains an identical view of the topology.

Calculation of Shortest Path Trees: The view of the topology can be seen as a directed graph with nodes representing the routers and edges representing the links of the network. A cost value is associated with each router interface and is distributed using LSAs. On basis of this (weighted) graph, each router generates a shortest path tree with the router itself as root of the tree showing the shortest route towards all destinations inside the AS. Current router implementations use Dijkstra's algorithm [SG01] or similar algorithms based on the Bellmann-Ford algorithm [Hui00]. The OSPF standard [Moy98] also allows using multiple cost metrics for different routes depending on different possible service classes as well as to use paths with equal costs simultaneously (*Equal Cost Multi Path*, ECMP). With ECMP, the outgoing traffic is distributed equally on the resulting outgoing interfaces.

Forwarding Information Base: After the routing table has been computed, the *Forwarding Information Base* (FIB) has to be configured. The FIB is used to determine the outgoing interface(s) of the packet forwarding process. Modern routers have at least two different FIBs (primary and secondary) that enable a non-stop forwarding during software updates and FIB configurations [Hui00]. An example of the forwarding in an OSPF network is illustrated in Figure 2.12 and Table 2.4.

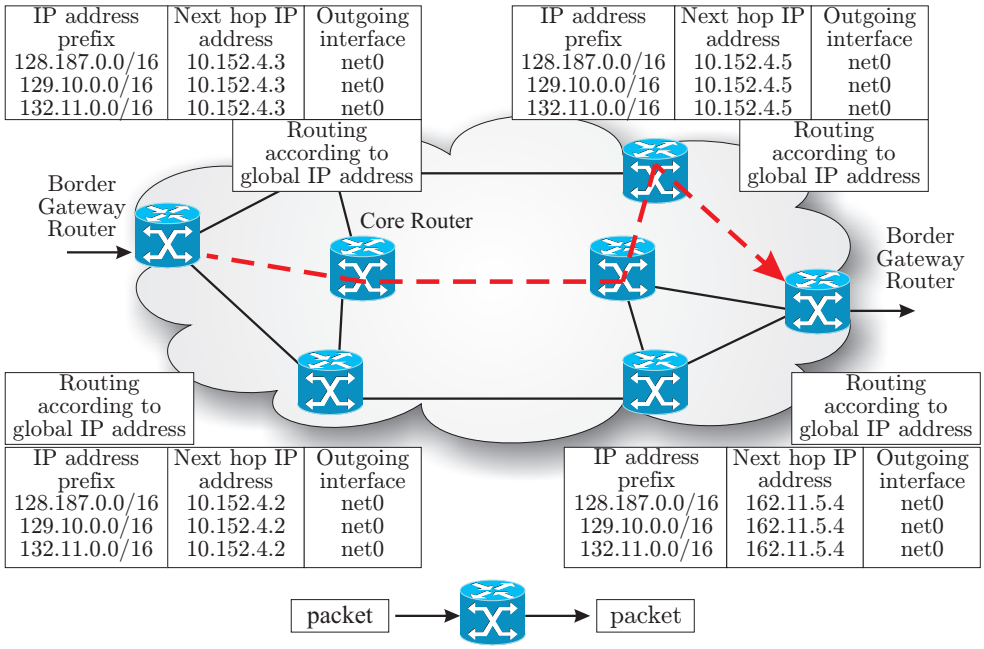


Figure 2.12: Schematic overview of the Open Shortest Path First forwarding.

Table 2.4: Example of a forwarding information base.

Entry	Destination Address	Network Mask	Gateway	Interface
1	178.155.0.0	255.255.0.0	170.18.121.1	net2
2	195.176.5.0	255.255.255.0	170.18.121.50	net1
3	195.176.5.0	255.255.255.0	170.18.121.50	net2
4	170.18.121.50	255.255.255.255	0.0.0.0	net0
5	0.0.0.0	0.0.0.0	170.18.121.13	net3

In general there has to exist at least one entry in the forwarding information base for each possible destination address. However, the number of entries can be reduced and entries towards different destinations can be combined using a network mask concept. Table 2.4 shows an example routing table of a forwarding information base including network masks and gateways, i.e. the border router of the area to which a packet towards a remote destination should be sent. The first row shows a 16 bit network mask that combines entries for destination addresses ranging from 178.155.0.0 to 178.155.255.255. These addresses can

be reached via gateway device (border router) 170.18.121.1 and should be routed towards interface net2. The second and third row show entries for a destination address using ECMP. Traffic towards destination addresses ranging from 195.176.5.0 to 195.176.5.255 is distributed evenly (i.e. 50:50) on the outgoing interfaces net1 and net2 towards border device 170.18.121.50. The fourth row shows an entry for a node with destination address 170.18.121.50 that is directly attached to the node. Finally, the last row defines a default route. All undefined destination network entries in this routing table will be mapped to the default route entry towards border device 170.18.121.13 along interface net3.

In summary, the link-state routing protocol OSPF provides simple mechanisms to find and store loop-free paths towards any destination. However, since large (tier1) backbone networks do not have a default route and addresses cannot be combined arbitrarily using the network mask concept, the number of entries in a forwarding information base is large [Hou01]. About 180.000 route prefixes are used in routing tables today [Smi06]. Additionally, due to the shortest path approach using only one metric per service class the possibilities for traffic-engineering is limited and the optimization of link weights is rather complicated as e.g. shown in [FT02].

2.2.2 Connection Oriented Routing

2.2.2.1 MPLS

Multi Protocol Label Switching (MPLS) was standardized by the IETF in RFC 3031 [RVC01] in January 2001. MPLS combines routing functionality of layer 3 and switching technologies of layer 2 in the ISO/OSI layering model. Due to this reason, it is often referred to as a layer 2.5 technology and can thus be seen as an intermediate layer between layer 2 and layer 3. The key idea, to establish virtual paths (*Label Switched Paths* - LSPs) inside a connection-less IP network, is an evolution from concepts known from *Asynchronous Transfer Mode* (ATM) [SM97] and several similar technologies that were invented in the mid 1990s. These approaches (notably *IP Switching* by Ipsilon [NLM96, NEH+96a, NEH+96b, NEH+96c], *Cell Switch Routing* by Toshiba [KNE97], Cisco's *Tag Switching* [RDK+97], and IBM's *Route-based IP Switching* [WVFB96]) proposed to use layer 3 IP addresses and Internet routing protocols such as OSPF and BGP to establish virtual paths that need not to be the shortest ones. Additionally, packets towards different destinations can be aggregated and the forwarding process can be simplified by using small path-identifiers that are independent of globally defined destination addresses. Today, MPLS is widely deployed in transport networks because of its possibilities of efficient traffic-engineering, resilience mechanisms, simplified establishment of *Virtual Private Networks* (VPNs), i.e. providing a private network over a shared infrastructure, and *Pseudowire Emulations*, i.e. providing an MPLS tunnel that appears to be a single hop for client traffic.

Figure 2.13 shows a schematic overview of Multi Protocol Label Switching forwarding. A *Label Edge Router* (LER), which is the beginning of a pre-established LSP, evaluates incoming IP packets and determines an LSP, i.e. a path, along that the IP packet should

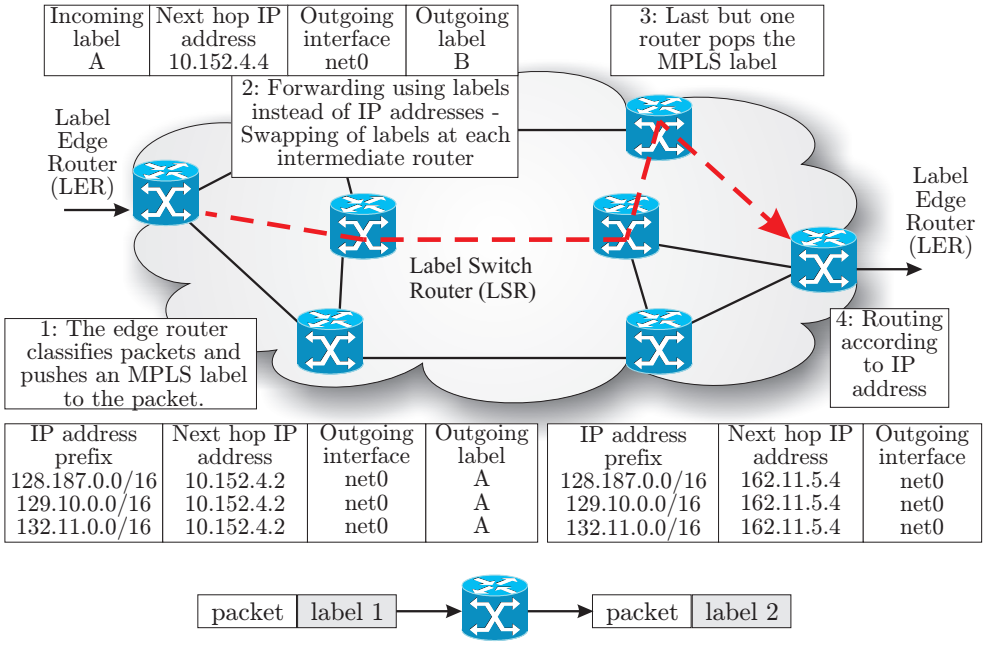


Figure 2.13: Schematic overview of Multi Protocol Label Switching forwarding.

be forwarded. To encode the chosen LSP, an additional MPLS header including an MPLS label, a short and fixed-length path-identifier, is stacked on top of the IP packet. Following this, *Label Switched Routers* (LSRs), routers in the middle of an LSP, forward the packet according to its MPLS label. Finally, the last but one LSR pops the MPLS header. The original IP packet is routed according to its IP destination address in the following.[8] In general, one label can be used to determine a path. However, to be able to reuse MPLS labels in a network without the need for signaling which label is already assigned in the network, the label can be swapped along the path. Labels have a local significance between two adjacent MPLS routers. Therefore, distinct forwarding tables may exist for each router interface.

Using this virtual-path concept, traffic towards different global IP destinations that will traverse a network via identical edge-routers can be aggregated and forwarded along the same LSP. Instead of using large IP routing tables small label switch tables can be used in the inner part of the network (see Figure 2.13). Traffic entering the network at the left most router and leaving the network at the right most router can take the indicated MPLS path independent of how many different IP destinations are addressed beyond that egress router. The routing tables at the edge of the network have one additional column compared to traditional IP routing to identify the label that should be pushed on top of

[8]The removing of the top-most MPLS header at the last but one router is called *Penultimate Hop Popping* (PHP) and can be used instead of removing the header at the last router along the path [IT04]. PHP is often not used in transport networks due to operational and management issues. Currently, an adaptation of the MPLS standard especially suited for Transport Networks (T-MPLS) therefore forbids the use of PHP [**KGRB06**, IT06a].

the packet. However, the aggregation of traffic towards the same egress router reduces the forwarding tables inside the core.

The MPLS protocol architecture can be divided into five parts:

- Information distribution (topology and capacity),
- label distribution,
- path calculation,
- path setup,
- forwarding traffic along an MPLS path.

Information Distribution: MPLS uses an *Interior Gateway Protocol* (IGP), e.g. OSPF [Moy98] or IS-IS [Kat00], to learn the network topology just as with IP routing. Additional information about available and used bandwidth as well as MPLS specific information (e.g. MPLS flags and MPLS weights) is provided by traffic-engineering extensions of the IGP protocols (OSPF-TE [KK03], IS-IS-TE [SL04]). With this additional information, traffic can be routed efficiently, individual bandwidth restrictions and reservations can be applied for each path, and traffic can be differentiated and separated using appropriate *Forwarding Equivalent Classes* (FECs). Additionally, link properties can be encoded using a 32-bit bitmap attribute flag for e.g. the differentiation of link types or delay properties. Furthermore, an MPLS administrative weight can be configured for each interface to allow simple routing algorithms independently of the used IGP weight.

Label Distribution: As mentioned above, the switching of packets is based on an MPLS label with a local meaning between two adjacent LSRs. However, these two routers must agree which label to use to identify the LSP. The *Label Distribution Protocol* (LDP, [ADF+01]) has four major functions: neighbor discovery, session establishment and maintenance, label advertisement, and label notification. For neighbor discovery and surveillance, two timers are standardized: A Hello timer which is used in combination with a hold timer to detect link failures (range in seconds - standard values: 5s, 15s) and a Keep-alive timer (range in seconds - standard value 60s) which is used to detect whether the neighbor is still functional.

Path Calculation: The route of an LSP can either be calculated centrally by a management tool and configured explicitly or can be calculated distributed via a *Path Computation Element* (PCE) with *Constraint Based Shortest Path* algorithms [FVA06]. The path is calculated according to IGP weights, available and used bandwidth, attribute flags, and administrative MPLS weights. Various possibilities are given to exclude or prefer interfaces or paths while calculating the best match for the constraints. In addition to this, each path has a priority and paths with higher priority can use resources of paths with lower priorities that are then automatically released [Cis01].

Path Setup: MPLS paths can be set up by explicit configuration of each device or by using the *Resource Reservation Protocol* (RSVP) as signaling protocol [BZB+97, ABG+01]. Therefore, the *Ingress Label Switch Router* (I-LSR), i.e. the head of an MPLS path, sends a PATH message along the calculated route using source routing, i.e. the route is configured in the packet. Each intermediate router checks if the required bandwidth is available and forwards the message towards the tail of the path (E-LSR). Once the message is received by the E-LSR a RESV message is sent back along the same path. On the way back, the resources are reserved and labels are selected and signaled to the upstream LSR. To release old paths and to check the status of a path or reservation, RSVP paths are refreshed periodically.

Resilience Mechanisms: Virtual paths that are not restricted to defined link weights or shortest paths enable the use of various resilience mechanisms. Some of the resilience mechanisms that can automatically configure backup paths for each established primary path are already supported in contemporary routers. As of today, Cisco proposes to use a local link or node protection mechanism called '*Fast Reroute*' [PSA05]. Furthermore, [AMA+99] defines a head-end (I-LSP) end-to-end rerouting mechanism for MPLS.[9]

Scalability: Since LSPs must be monitored and managed, the number of LSPs inside a network has to be limited. Cisco [OS03, p.409] gives the following guidelines for the maximum number of LSPs that should be used in a network to overcome performance degradation of other router parameters: 600 LSPs as a head-end (I-LSR), 10000 LSPs as a midpoint (LSR), 5000 LSPs as a tail (E-LSR).[10]

2.3 Requirements of Resilient Network Planning

In general, high availability, i.e. a high probability that a service is functional as specified, can be achieved by reliable design, i.e. using components and mechanisms that are unlikely to fail, or by including countermeasures (resilience mechanisms) that can react in case of failures. Although, in principle, a network should be designed as resilient as possible, both approaches imply redundancy and additional costs. Thus, resilience mechanisms have to be chosen carefully and have to be suited to provide the agreed service availability. Table 2.5 illustrates the relation between the service availability and the mean outage times. Typically required end-to-end availability values in backbone networks range between 0.999 (three nines) and 0.99999 (five nines) (e.g. an end-to-end availability of 0.9994 for Public Switched Telephone Networks (PSTNs) according to Telcordia [Cab00]).

In order to achieve these availability values, resilience mechanisms and enough spare capacity have to be provided for at least the most probable network element failures, i.e.

[9] We will present more details on possible path-based resilience mechanism for connection-oriented forwarding in Chapter 3.

[10] These small numbers may no longer be applicable for modern routers, e.g. Cisco's Carrier Routing System (CRS-1).

Table 2.5: Outage time dependent on end-to-end availability.

Availability		Unavailability	Outage time
0.9	(1 nine)	0.1	867.6 hours/year = 36.53 days/year
0.95		0.05	438.2 hours/year = 18.26 days/year
0.99	(2 nines)	0.01	87.66 hours/year = 3.65 days/year
0.995		0.005	43.83 hours/year = 1.83 days/year
0.999	(3 nines)	0.001	8.77 hours/year
0.9995		0.0005	4.38 hours/year
0.9999	(4 nines)	0.0001	52.60 minutes/year
0.99995		0.00005	26.30 minutes/year
0.99999	(5 nines)	0.00001	5.26 minutes/year
0.999995		0.000005	2.63 minutes/year
0.999999	(6 nines)	0.000001	0.53 minutes/year

failures of transmission trunks (edges) and switching/routing components (nodes). Thus, in the following sub-sections we will analyze the probability of failures and investigate the requirements on service interruption time.

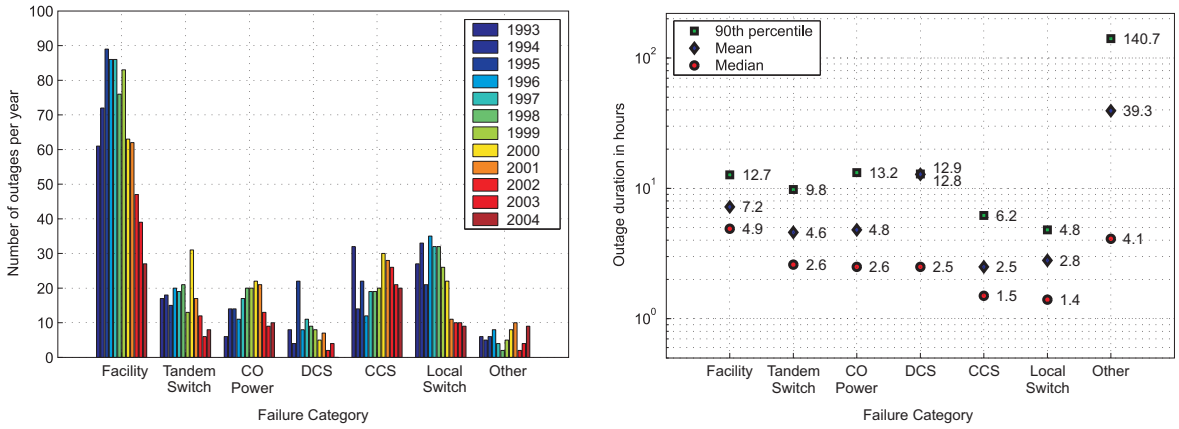
2.3.1 Probable Failure Patterns

Although almost every network component provides *Failures In Time* (FIT) values that reflect the possibility of a hardware failure, reliable information on common failure patterns in today's networks is not easily obtainable. This is because network operators are reluctant to reveal any availability deficiencies of their network that might then be used as an argument against them. However, some publications are available and neutral sources such as standards organizations (e.g. ETSI, ITU) or regulatory bodies provide some additional insights in probable failure patterns and failure reasons that have to be taken into account when designing resilient networks.

2.3.1.1 Failure Characteristics of Existing Networks

Figures 2.14(a) and 2.14(b) show information published by the Federal Communication Commission (FCC) of the U.S.A. [FCC06] and the Network Reliability Steering Committee (NRSC), under the auspices of the Alliance for Telecommunications Industry Solutions (ATIS) [Net05]. The information is based on reports provided by U.S. telecommunication providers that are legally obligated to report any network outage potentially affecting more than 30.000 customers and lasting longer than 30 minutes (Title 47, Code of Federal Regulations 63.100 [Com93]).

On an average 167 failures per year were reported to the FCC in the last 11 years. However, recently this number has reduced to values below one hundred (91 in 2003 and 87 in 2004) due to a larger number of installed resilient mechanisms in the networks. Facility



(a) Number of outages.

(b) Outage duration.

Figure 2.14: Annual number of reported outages and outage duration in the year 2004 reported to FCC [Net05].

failures, i.e. cable dig-ups and cable-electronic failures such as repeaters and multiplexers are the most prominent failures that contributed to 27 reported outages in the year 2004. The main part of additional 21 reported failures were caused by hardware and software failures in *Digital Cross Connects* (DCS), and local, and tandem switches. Finally, procedural errors were the most dominant root cause of reported *Common Channel Signaling* (CCS) and *Central Office Power* (CO Power) failures. Figure 2.14(b) illustrates the outage duration dependent on failure types for the year 2004. The outage duration is heavily influenced by its root cause. Hardware and software failures could be repaired relatively fast in approx. 4 hours whereas cable-dig up failures as well as failures caused by procedural errors took two to three times longer to repair.

Analyzing the data above, we note that fiber cuts amount to roughly half of the failures in the backbone. However, we must keep in mind that service providers have internal redundancy equipment. Thus, only those (multiple) failures were reported for which no countermeasure had been available and that persisted for more than 30 minutes. Thus, the number of failures in today's networks is much higher.

Another detailed failure statistics study based on failure logs and trouble tickets of a medium size regional backbone provider (Merit/MichNet [Mer06]) connecting education and commercial customers in 132 cities in Michigan, U.S.A. between November 1997 and November 1998 is summarized in Table 2.6. Routers connecting libraries or colleges often do not offer high availability or redundant equipment due to cost limitations. Additionally, field service personnel to repair outages quickly will often not be available. Thus, the majority of the outages listed in Table 2.6 are associated with individual customer sites rather than with backbone equipment. However, the numbers give reasonable insights in possible (single) failures in communication systems and outage causes. Furthermore, assuming redundant power supplies and disregarding the too vaguely defined outage categories 'Un-

Table 2.6: Recorded outages of the Merit network during November 1997 and November 1998 taken from [LAJ98].

Outage Category	Number of Occurrences	Percentage
Power Outage	273	16
Maintenance	272	16
Fiber Cut/Circuit/Carrier Problem	261	15
Unreachable	215	13
Hardware Problem	154	9
Interface Down	105	6
Routing Problems	104	6
Miscellaneous	86	6
Congestion/Sluggish	65	5
Unknown/Undetermined	32	5
Malicious Attack	26	2
Software Problem	23	1

reachable', 'Miscellaneous' and 'Unknown/Undetermined', 70 % of the remaining outages are caused by 'Fiber Cut/Circuit/Carrier Problem' and 'Interface Down'.

Next to information about failure causes, individual component availabilities are of importance for failure probability calculations. I.e. the ratio of (a) the total time the functional unit is capable of being used during a given interval and (b) the length of the interval [All00]. Two components are of interest for the given network architectures: Network nodes, i.e. routers or switches and network edges, i.e. connecting links. Although major router vendors (e.g. Cisco and Juniper) do not disclose exact values for their routers, *Mean Time Between Failures* (MTBF) values of more than 7 years and more than 200.000 hours are stated in product descriptions. Given an average *Mean Time To Repair* (MTTR) of four hours, the availability of the components can be calculated using the well known formula for availability approximation (Equation (2.1))[11], to be in the range between 0.9999 (four nines) and 0.99999 (five nines).

$$A = \frac{MTBF}{MTBF + MTTR} \tag{2.1}$$

Fiber cuts are the most dominant failures in today's telecommunication networks. MTBF values for real measurements of terrestrial optical fibers networks are reported to be in the range of 275 to 1000 years per kilometer [SAF01, VCD+05, Gro04, Cra93]. Since fiber-cuts have to be located and eventually excavated, MTTR values in the range of several hours are required. A study by Crawford [Cra93] reports an average mean time to complete a fiber repair to be 14.2 hours with an average service restoration time of

[11] More details on availability calculation and the method to calculate exact availability values can be found in [Rob99, Ise98].

5.2 hours. Nowadays, however, reparation times seem to be shorter as indicated in Figure 2.14(b). Assuming an average link length of 500 kilometers, an average MTBF of 275 years and an average MTTR of 8 hours, the mean availability of links can be assumed to be 0.999 (three nines). Analog values are given by standardization bodies in ETSI standard EN 300 416 [ETS98] and ITU-T recommendation G.827 [IT03a] to be in the order of 0.999 (three nines) for edges with less than 500 kilometer.

2.3.1.2 Failure Probability Calculations

Given individual unavailability values of network elements (q), the probability of f failures in a network area with X independent elements can be calculated by using the Binomial distribution (Equation (2.2)).

$$p_f(X) = \binom{X}{f} (1-q)^{X-f} \; q^f \tag{2.2}$$

The probability to have no failure (p_0), one failure (p_1), or two failures (p_2) in a network can be calculated using Equations (2.3), (2.4) and (2.5). The terms N and E denote the number of nodes and edges of the network while q_n denotes the failure probability of a node and q_e the failure probability of an edge.

$$p_0(N,E) = \binom{N}{0} (1-q_n)^N \; q_n^0 \binom{E}{0} (1-q_e)^E \; q_e^0 = (1-q_n)^N \; (1-q_e)^E \tag{2.3}$$

$$\begin{aligned} p_1(N,E) = & \binom{N}{1} (1-q_n)^{N-1} \; q_n^1 \binom{E}{0} (1-q_e)^E \; q_e^0 \; + \\ & + \binom{N}{0} (1-q_n)^N \; q_n^0 \binom{E}{1} (1-q_e)^{E-1} \; q_e^1 \end{aligned} \tag{2.4}$$

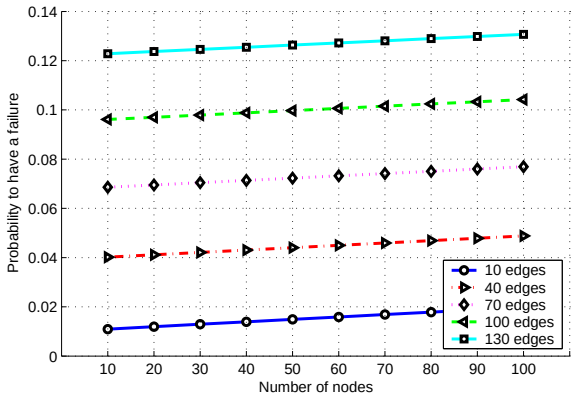
$$\begin{aligned} p_2(N,E) = & \binom{N}{2} (1-q_n)^{N-2} \; q_n^2 \binom{E}{0} (1-q_e)^E \; q_e^0 \; + \\ & + \binom{N}{1} (1-q_n)^{N-1} \; q_n^1 \binom{E}{1} (1-q_e)^{E-1} \; q_e^1 \; + \\ & + \binom{N}{0} (1-q_n)^N \; q_n^0 \binom{E}{2} (1-q_e)^{E-2} \; q_e^2 \end{aligned} \tag{2.5}$$

Consequently, the probability to have one or more failures ($p_{>0}$), to have more than one failure ($p_{>1}$), and to have more than two failures $p_{>2}$ can be calculated according to Equations (2.6) to (2.8). Figures 2.15(a) to 2.15(c) illustrate these values dependent on the network size for given failure probabilities of $q_n = 10^{-4}$ and $q_e = 10^{-3}$.

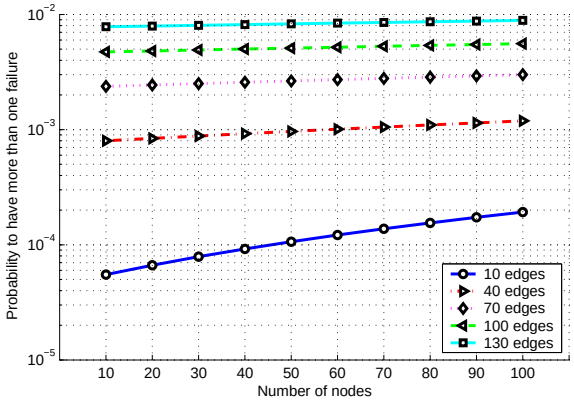
$$p_{>0}(N,E) = 1 - p_0(N,E) \tag{2.6}$$

$$p_{>1}(N,E) = 1 - [\; p_0(N,E) + p_1(N,E) \;] \tag{2.7}$$

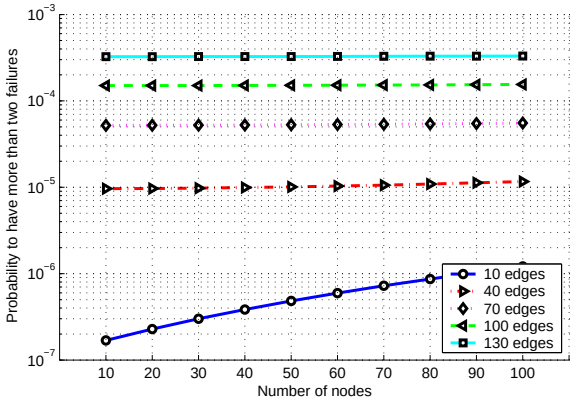
$$p_{>2}(N,E) = 1 - [\; p_0(N,E) + p_1(N,E) + p_2(N,E) \;] \tag{2.8}$$



(a) Probability to have a failure ($p_{>0}$).



(b) Probability to have more than one failure ($p_{>1}$).



(c) Probability to have more than two failures ($p_{>2}$).

Figure 2.15: Failure probabilities dependent on the number of nodes and edges. $q_n = 10^{-4}$, $q_e = 10^{-3}$. Please note, not all shown constellations can form a connected network.

The figures reveal that the probability to have multiple failures is quite high. Indeed, the probability is mainly dependent on edge failures due to their lower element availabilities compared to nodes. The numbers above, however, provide information about the probability of a network element failure anywhere in the network. While this is beneficial for network operators to e.g. plan and schedule their service personnel, some failures do not affect a demand and need thus not to be protected. Actually, a protected demand will be impaired, if not enough capacity is available on working and backup path(s) simultaneously. However, this makes the calculation of an end-to-end availability of a demand more complicated.

To illustrate this, Figure 2.16 depicts an example of a demand between node A and node B that will be routed along a working (primary) path or transported along a backup

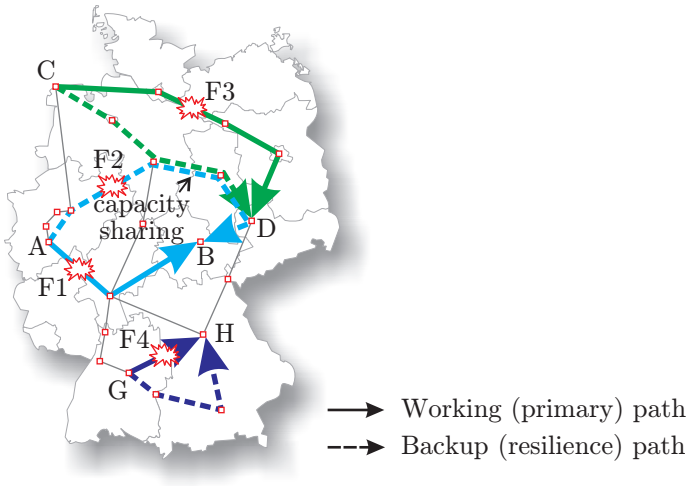


Figure 2.16: Example of different failure types that have different effects on traffic.

path, if a failure along the working path occurs. Without loss of generality, we can note the following:

1. As long as the working path is functional, i.e. no element along the working path failed, the demand is not impaired. This will also be the case if failures along the backup path (failure-type $F2$) or of other network elements occur (failure-type $F3$ or $F4$).

2. If we assume dedicated backup capacity, i.e. capacity is not shared between different backup paths, the demand will be impaired if the working path and the backup path are affected simultaneously (failure-type $F1$ and $F2$).

3. If we assume shared backup capacity, the demand will be impaired additionally, if elements of the working path have failed and another working path is affected that shares backup capacity with this demand (failure-type $F1$ and $F3$). Thus, not enough capacity is available on the backup paths for both demands. However, failures of network elements affecting paths that do not share capacity with the demand need not to be considered (failure-type $F1$ and $F4$).

We can asses the number of failures a demand should be protected against by distinguishing elements of the working paths from elements that can be used for backup path(s). The number of nodes and edges along the working paths are represented as N^W and E^W, respectively, while $N^R = N - N^W$ and $E^R = E - E^W$ denote the number of nodes and edges in the remaining part of the network. Consequently, the probability that a *demand* is affected by a failure, is affected by more than one, or more than two failures can be

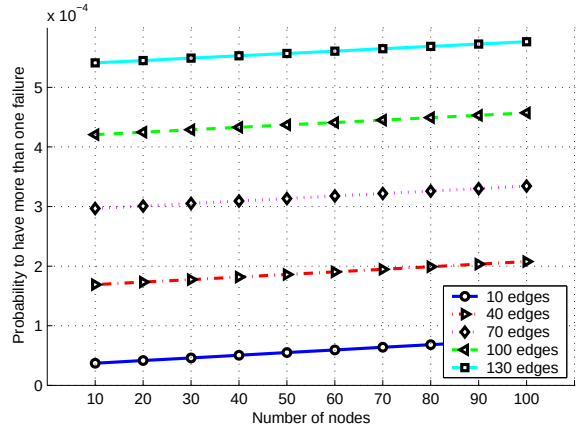
calculated according to Equations (2.9) to (2.11).

$$p'_{>0}(N^W, E^W) = 1 - p_0(N^W, E^W) \tag{2.9}$$

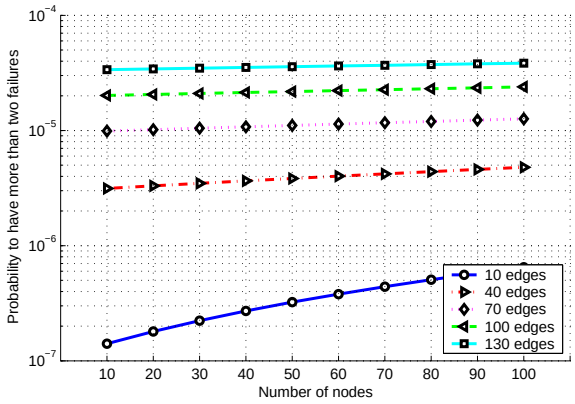
$$p'_{>1}(N^W, E^W) = 1 - [\ p_0(N^W, E^W) + p_1(N^W, E^W) \cdot p_0(N^R, E^R)\ ] \tag{2.10}$$

$$p'_{>2}(N^W, E^W, N^W, N^R) = 1 - [\ p_0(N^W, E^W) + p_1(N^W, E^W) \cdot p_0(N^R, E^R) + + p_1(N^W, E^W) \cdot p_1(N^R, E^R) + p_2(N^W, E^W) \cdot p_0(N^R, E^R)\ ] \tag{2.11}$$

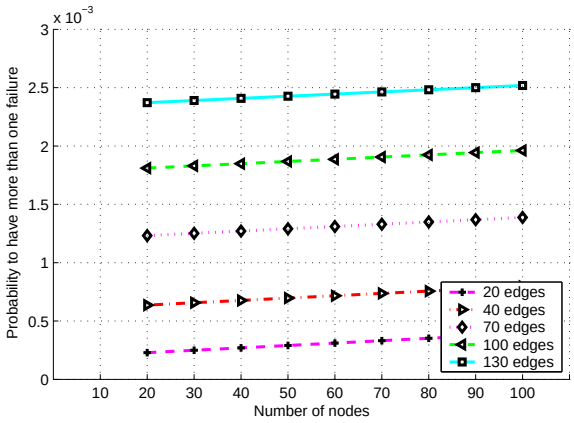
Figures 2.17(a) to 2.17(d) show the probability to have more than one failure and more than two failures for working path lengths of $N^W = 5$, $E^W = 4$ and $N^W = 20$, $E^W = 19$, dependent on the network size.



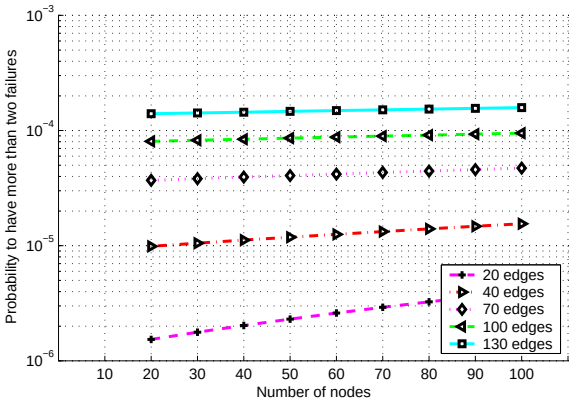
(a) Probability to have more than one failure for $N^W = 5$, $E^W = 4$.



(b) Probability to have more than two failures for $N^W = 5$, $E^W = 4$.



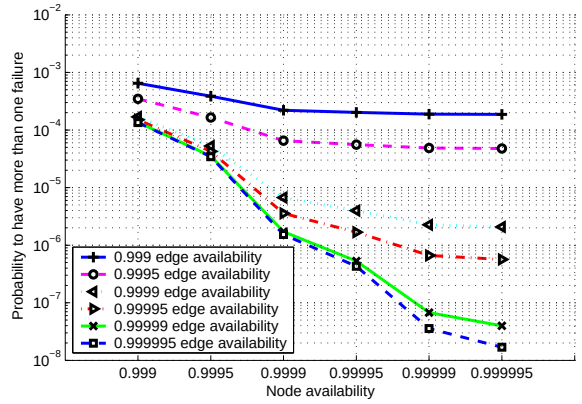
(c) Probability to have more than one failure for $N^W = 20$, $E^W = 19$.



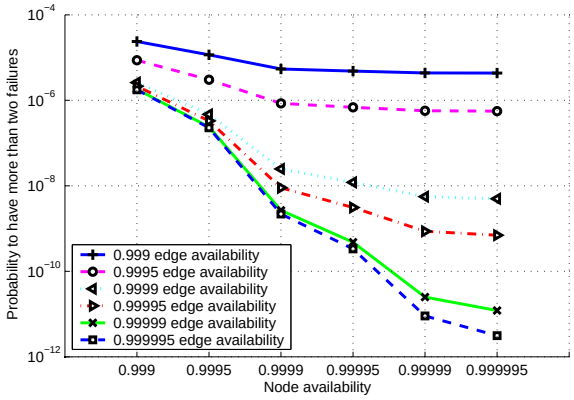
(d) Probability to have more than two failures for $N^W = 20$, $E^W = 19$.

Figure 2.17: Failure probabilities that impair a demand (dependent on working path length). $q_n = 10^{-4}$, $q_e = 10^{-3}$. Please note the different scaling of the y-axis. Not all shown constellations can form a connected network.

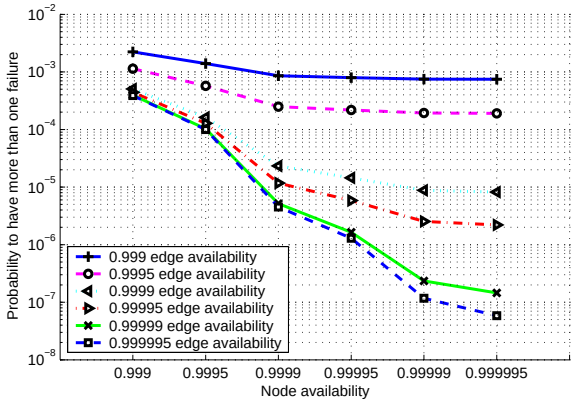
The probability of having a failure is increasing when working paths are getting longer. This is obvious since more elements can fail along the working path. However, even in medium-sized networks with around fifty nodes the failure probability of having more than two failures is less than 10^{-4}. In these networks, it is sufficient to install resilience mechanisms that provide backup paths for single and double failures only. Even more, for smaller networks with around ten to twenty nodes it is sufficient to have resilience mechanisms for single element failures only.



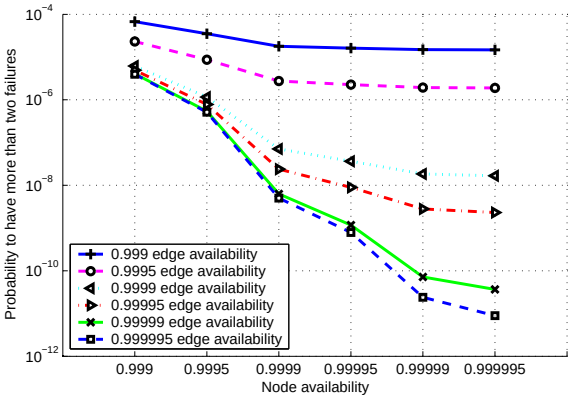
(a) Probability to have more than one failure for $N^W = 5$, $E^W = 4$.



(b) Probability to have more than two failures for $N^W = 5$, $E^W = 4$.



(c) Probability to have more than one failure for $N^W = 20$, $E^W = 19$.



(d) Probability to have more than two failures for $N^W = 20$, $E^W = 19$.

Figure 2.18: Failure probabilities that impair a demand (dependent on working path length). $N = 30$, $E = 50$.

The dependency of the failure probability on the availability ratios of the individual network elements is furthermore illustrated in Figures 2.18(a) to 2.18(d) for a network size of $N = 30$ and $E = 50$. The overall failure probability is mainly dominated by the network element with the lowest availability. I.e. failure calculations are almost independent on the exact value of the elements with higher availabilities. Overall, resilience mechanisms have

to be installed for the following failure patterns to achieve end-to-end availability values of around 0.9999 (four nines):

- single link failures,
- single node failures,
- double link failures

2.3.2 Recovery Time Requirements

The previous section revealed that failures of communication networks are quite frequent. Thus, mechanisms have to be deployed in order to reduce the perceived end-to-end outage times. However, in the telecommunication community there is a lot of dispute about the time requirements of resilience mechanisms, i.e. when a mechanism should react and fully restore the service (recovery time). Often 50 milliseconds are quoted, which is the specified speed to protect single link failures using *SONET Automatic Protection Switching* (APS): Approx. 20 milliseconds are required to detect a failure, 10 milliseconds are required for signaling, 10 milliseconds are added due to operational tail-end transfer-delay, and 10 milliseconds are added as spare time [Gro04].

Strict time requirement inhibit the deployment of some capacity efficient resilience mechanisms. Considering distributed rerouting mechanisms, for example, the required service restoration time of 50 milliseconds can hardly be reached because of signaling times between remote destinations.[12] Thus, the question remains whether 50 milliseconds are really required or, if the capability to do so with one resilience mechanism more or less "evolved to a requirement" [Gro04].

Several investigations of the impacts of different outage times on transported services have been carried out in the last years. Table 2.7 summarizes key findings of studies by S. Butenweg et al. [BS02], J. Sosnosky [Sos94] with additions by [Gro04], J. Schallenburg [Sch01] and ANSI standard T1.TR.68-2001 [ANS01].

Outage durations of less than 50 milliseconds certainly do interrupt a traffic flow. This can be perceived by customers as packet loss, or as a short 'click' in voice traffic. However, right from an early development phase, most applications and network protocols were designed to conceal or cope with negative effects of bitrate reductions or packet loss. Thus, if these clicks do not occur very often, today's networks and services are quite resistant to brief outages. When having outage times between 50 and 200 milliseconds, some streaming and real-time video codecs might require a reframing process [BS02]. However, the transmission of data is not jeopardized by this. Additionally, there is a possibility of less than 5% (at 200 milliseconds) that voiceband connections are disconnected (with old equipment) or are switched over to designated backup paths [Sch01, Sos94, Gro04]. Similarly, an ATM cell-rerouting process may be started with 200 milliseconds outage duration. These effects increase even more when having outage durations of 200 milliseconds to 1 second. More voiceband connections are dropped and session dependent applications (e.g. TCP) may

[12] E.g. one-way signaling between New York and San Diego (4500 kilometers) takes around 22.5 milliseconds.

Table 2.7: Classification of outage time impacts on service based on [Sch01, Gro04, Aut02].

Outage Duration	Main Effects / Characteristics
0 to $<$ 50 milliseconds	Service "hit" and packet loss; reframing required; packets are resent by TCP applications.
50 milliseconds to $<$ 200 milliseconds	Retransmissions by some streaming and real-time audio and video codecs; $<$ 5% voiceband disconnects by signaling system (SS7); SMDS (frame-relay) and ATM cell-rerouting may start.
200 milliseconds to $<$ 1 second	Dropping of switched connections on vintage equipment.
1 second to $<$ 10 seconds	Human being interaction; disconnection of all switched circuit services; potential X.25 disconnects; TCP/IP protocol may back off,
10 seconds to $<$ 5 minutes	Packet (X.25) disconnects; data session timeout; redials and reconnects by users.
more than 5 minutes	Social/business impact

begin a backoff process in the time range of 1 to 2 seconds [BS02, Gro04]. With outage durations of more than one second, the service interruption becomes far more serious and visible to human beings. Furthermore, virtually all voiceband connections are disconnected with outage times of around 2 seconds [BS02, Gro04]. Thus, from a time requirement point of view, the requirements of services on outage time can be separated into four different categories:[13]

- Outage duration of up to 50 milliseconds,
- outage duration between 50 and 200 milliseconds,
- outage duration between 200 milliseconds and 1 second,
- no maximum outage duration requirement.

In summary, all sources agree on the necessity to have fast resilience mechanisms. However, especially non real-time services that have the capability of detecting and performing packet retransmission are very robust against network outages. Streaming applications are often able to buffer information in advance and allow a retransmission of lost data during the depletion time of the buffer. Similarly, transport protocols like UDP, TCP and RTP/RTCP are insensitive to network outages up to some seconds during the connection establishment phase (if existing) and connection. However, when using real-time services with strict time constraints a network outage and data loss is conceivable by the user. Fortunately, however, reaction times of human beings are quite slow and outages that occur infrequently (once or twice a week) lasting less than 200 to 500 milliseconds are acceptable by most users [BS02].

[13]Classifications of resilience mechanisms based on similar categorizations are proposed by A. Autenrieth [Aut02], M. Tacca et al. [TFP+03] and proposed in Metro Ethernet Forum [Met04].

2.4 Chapter Summary

The network planning process requires the solution of several optimization problems. Due to their complexity, these tasks are often solved separately. However, since many of the tasks are dependent on each other, several iterations and repetitions of the network planning cycle have to be performed. Although a combined solving of several optimization problems is complex, drastic improvements in the solution quality can be achieved. Especially the combination of topology planning, multipath routing and dimensioning as well as the joint optimization of failure-free and failure affected network states is beneficial.

Studies of network failure statistics and in-depth availability calculations revealed that most failures are caused by link failures today. However, moderately sized transport networks (up to around 50 nodes) have to be protected against single link, single node, and double link-failures only to provide end-to-end availability values of around 0.9999 (four nines) for the routed demands. Additionally, outage time durations of up to 200 milliseconds are acceptable for most services.

Chapter 3

Resilience Classification Framework

The analysis of the network planning cycle of Chapter 2 showed that the choice of the resilience mechanism is an important issue when designing telecommunication networks and has substantial influence on capital expenditures (CAPEX) as well as operational expenditures (OPEX).

If impacts of failures and characteristics of resilience mechanisms are not well understood, network equipment failures will be able to cause catastrophic events and immense loss of transported data. Dependent on affected services, applications and service level agreements, penalty payments, injuries, or even deaths of human beings are possible when considering the dysfunction of emergency calls. However, disproportionate conservative planning of resilience mechanisms leads to increased OPEX and CAPEX that are not tolerable in a world of global competition between transport network providers. Thus, at the beginning of a network design process, resilience requirements have to be defined, mechanisms have to be designed, analyzed, and compared with each other in order to implant the best fitting resilience mechanism in the network. Unfortunately, a number of issues impede the characterization and comparison of existing and the development of new resilience mechanisms.

We will discuss these issues in Section 3.1 in detail and give an overview of related work in Section 3.2. Based on resilience terminology definitions of Section 3.3, we will present a novel classification framework consisting of eight building blocks in Section 3.4 with which resilience mechanisms can be described independently of technology issues. The *Resilience Classification Framework* (RCF) enables the finding of mechanism-characteristics, allows a mechanism comparison, and provides the basis for a decision towards the best fitting resilience mechanism. Furthermore, due to the decomposition into building blocks, dependencies and new combinations of characteristics can be found and the design of novel resilience mechanisms is facilitated.

To illustrate the advantages of the framework, we will perform example classifications of different resilience mechanisms and will perform a theoretical RCF-based comparison in Section 3.5. Additionally, in Section 3.6, we will present and discuss a novel resilience mechanisms called *Self Regulating Traffic Distribution* (SRTD) that has been identified by

applying a white-spot analysis on the RCF. Finally, we will summarize the key findings of this chapter in Section 3.7.

3.1 Need for a Resilience Classification Framework

Providing suitable robustness against network element failures was and still is one of the key issues during the design of transport networks. As seen in Chapter 2 there are various reasons for failures. Thus, almost every transport network architecture defines own resilience mechanisms to reduce the negative effects of network equipment failures. However, separate development has led to a number of issues when designing resilient networks:

- No common resilience terminology,
- difficult comparison of mechanism characteristics,
- hampered finding and development of novel resilience mechanisms.

3.1.1 Resilience Terminology

Resilient network design is an area in which approaches and results from different kinds of sciences (e.g. Mathematics, Informatics, Physics, and Engineering) are combined. Additionally, since telecommunication networks are of utmost importance for a large variety of business areas, reliability and proper work of these systems are addressed not only by network designers but also by users of the telecommunication network (e.g. investment banks, governments, or e-commerce companies). Thus, a mixture of terms and semantics from different areas and sciences were introduced gradually in the field of resilient network design.

The deregulation of telecommunication networks in Europe and the evolving Internet technologies has additionally resulted in a clash of two approaches and beliefs in the 1990s. This clash that is often denoted as *'War between Netheads vs. Bellheads'* [Ste96] addresses the discussions between the traditional telephone companies (Bellheads), that believed in rigorous quality control and standardization, and evolving computer-network providers (Netheads), that believed in *Rough Consensus and Running Code* [Har01b]. At that time, the number of people involved in designing, building and operating telecommunication networks increased rapidly and less time was invested to define a common resilience nomenclature.

While, most of the standardization bodies today (e.g. IETF, ITU-T, IEEE, ETSI, MEF) are more and more coordinating the standardization of technologies[1], competing companies often define new terms for similar approaches to distinguish their products and services from that of their competitors. A prominent example is the resilience mechanism *Fast Reroute* by Cisco Systems, Inc. [Cis06]. The mechanism that is described in IETF RFC 4090 [PSA05] uses a pre-configured *protection* path similar to the resilience mechanism

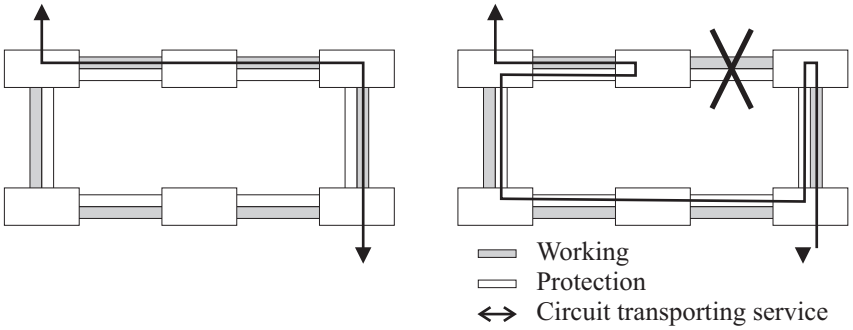
[1]See for example standardization documents ITU-T Y.1720 [IT03c], IETF RFC 3469 [VSFH03] and IETF RFC 4427 [EMDP06].

SLLPP that we will describe later on in Section 3.5. Although, Fast Reroute is a protection mechanism, the term *rerouting* in conjunction with *fast* has been used to highlight its applicability in their layer 3 routing devices. The term *routing* is even more misleading, since Fast Reroute is applicable in the MPLS *switching* part of the device.

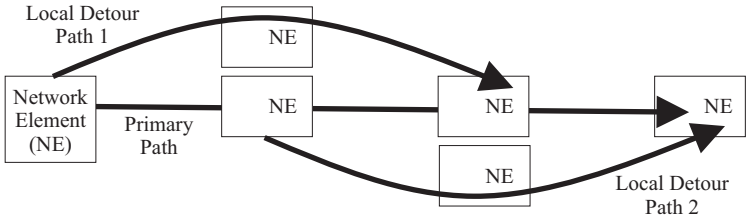
Consequently, even on expert conferences many different terms are used synonymously but are sometimes also used with different semantics. Example terms are *survivable, secure, resilient, robust, fault-tolerant, self-healing networks* or terms like *link, duct, edge, connection, span,* and *line.*

3.1.2 Comparison of Resilience Mechanisms

Obviously, there is a plethora of possibilities to describe a resilience mechanism. Dependent on the intention of the author, different emphasis is given to different parts of the resilience mechanism. Figure 3.1, for example, shows topology constellations for resilience mechanisms called *Multiplex Section Shared Protection Ring* (MSPRING) and *Aggregated Line and Node Protection* (ALNP) with the following excerpts from standardization documents:



(a) Example Multiplex Section Shared Protection Ring circuit routing in failure state for a ring switch. Taken from ITU-T G.841 [IT98]



(b) Example Aggregated Line and Node Protection. Taken from Metro Ethernet Forum, Technical Specification MEF 2 [Met04].

Figure 3.1: Example standardization figures.

MSPRING description taken from ITU-T G.841 [IT98]: "*...During a ring switch, normal traffic transmitted toward the failed span is switched at one switching node to the protection*

channels transmitted in the opposite direction (away from the failure). This bridged traffic travels the long way around the ring on the protection channels to the other switching node where the normal traffic from the protection channels is switched back onto the working channels. In the other direction, the normal traffic is bridged and switched in the normal manner..."
ALNP description taken from Metro Ethernet Forum, Technical Specification MEF2 [Met04]: "*...ALNP provides protection against local link and nodal failure by using local path detour mechanisms. In this case, local 'backup' or 'detour' paths are created along the primary pass that bypass the immediate downstream network element NE or the logical link and immediately merge back on the primary path. The detour path may provide 1:n protection or 1:1 protection of the primary paths in the network...*"

Obviously, using these definitions alone it is difficult to compare the characteristics of the resilience mechanism with each other concerning availability, capacity requirements, or recovery time. Some resilience mechanisms seem to differ substantially. Additionally, mechanisms or algorithms to find good topologies and or routes are often not part of the standardization description. Thus, experts for each resilience mechanism are required for a fair comparison of characteristics.

However, as we will present in Section 3.4, a resilience mechanism can be separated into different sub-categories. Thus, a structured description and a facilitated comparison is possible when using a resilience classification framework, as we will demonstrate in Section 3.5.

3.1.3 Development of Novel Resilience Mechanisms

As mentioned above, a large number of resilience mechanisms exist today that are standardized by different bodies. Thus, it is very difficult to summarize all - or at least the most prominent - approaches and mechanisms that exist today.

The analysis in Section 3.4, however, will reveal that a resilience mechanism can be considered as a combination of different characteristics. Thus, a structured description and a facilitated comparison is possible when using a resilience classification framework. Additionally, due to the structured separation, sub-categories and with it specific characteristics of different resilience mechanisms can be analyzed and recombined to form new resilience mechanisms. Thus, the finding of new resilience mechanisms is facilitated with a resilience classification framework, as we will demonstrate in Section 3.6.

3.2 Related Work

Certainly, a complete summary of all publications that describe and investigate resilience mechanisms is out of the scope of this document. However, we will mention the most important publications that actually standardize, compare, and categorize different resilience mechanisms in the following.

ITU-T standard Y.1720 [IT03c] provides a detailed description of 1+1, 1:1, shared mesh and packet 1+1 protection switching of the data-plane in MPLS networks. Conceptual figures are provided for all protection architectures; however, the resilience mechanisms are not compared with each other. Similarly, ITU-T G.841 [IT98] and G.842 [IT97] define SDH-based protection mechanisms such as Multiplex-Section Shared Protection Rings (MSPRING) and Sub-network Connection Protection (SNCP) rings. Architectures for the mechanisms are described in general terms including functional model illustrations. However, a comparison of mechanisms and a discussion of characteristics considering capacity requirement, recovery time, and complexity are not performed. ITU-T G.808.1 [IT03b] defines different protection architectures (1+1, 1:N, M:N, $(1:1)^n$) however, no comparison is performed. MEF document [Met04] considers Aggregated Line and Node Protection (ALNP), End-to-end Path Protection (EEPP) as well as multipoint-to-multipoint protection mechanisms and a link protection mechanism based on link aggregation. While similar descriptions of characteristics are used, no classification, comparison, and separation in building blocks is apparent. Document ETSI TR 101971 [ETS05] provides a network survivability framework that illustrates the relation between user perceived performance metrics, network performance metrics, service level agreements, and network requirements. The document discusses design considerations for the deployment of survivable networks and provides guidelines for operational measurement and improvement of the reliability/availability of IP based networks and services. Different failure causes and recovery times of IP-based routing protocols are discussed and metrics for the reliability/availability clauses in SLAs and a categorization of class of resilience are provided. However, individual resilience mechanisms and their characteristics are not in the scope of the document.

Next to standardization documents, a number of specific resilience related books were published recently that describe mechanisms and some characteristics of resilience mechanisms in detail. Very good overviews about existing resilience mechanisms are given by Vasseur [VPD04], Medard [ML06], and Grover [Gro04]. Additionally, a number of publications describe path-based resilience mechanisms (e.g. [RSM03, SP04, RM99]), perform comparison analysis (e.g. [ZD02, Aut02]) or provide first classifications of specific resilience mechanisms (e.g. [LI05] and [Sch05]).

3.3 Resilience Terminology Definition

In this section, we define the most common terms that are used in the resilience classification framework to define and characterize resilience mechanisms. A more detailed list of terminologies can be found in Appendix A.

Availability: *Availability is the probability that an item will be able to perform its designed functions at the stated performance level, within the stated conditions, and in the stated environment when called upon to do so.* [Kal02]. *When we assume constant component failure- and repair-rates the availability can be approximated using Mean Time Be-*

tween Failure (MTBF) and Mean Time To Repair (MTTR) values.[2]

$$A \approx \frac{MTBF}{MTBF + MTTR} \tag{3.1}$$

Backup Resources: *A resource, e.g. a path, that is used in fault condition to restore traffic of a working path. The recovery path can either be an equivalent recovery path and ensure no reduction in quality of service, or be a limited recovery path and thereby not guarantee the same quality of service (or some other criteria of performance) as the working path. Synonyms for a backup resource are: recovery resource, alternative resource, and protection resource.* [IT03c, VSFH03]

Dedicated Resources: *Reserved recovery resources that may be used to protect one working resource and cannot be shared.*

Demand: *The aggregation of flows between each pair of nodes on the transport network.* [Gro04]

Failure: *Termination of the capability to transfer user or OAM information due to an outage.* [IT03c]

Global Restoration: *A resilience mechanism in which new routes for all working paths are calculated, configured, and established dynamically after the detection of a fault.*

Mean Time Between Failure: *Mean Time Between Failure (MTBF) is the average time a device will function before failing.*

Mean Time To Repair: *Mean Time To Repair (MTTR) is the average time that it takes to repair a failure.*

Multipath: *Multiple resources that carry the traffic of a demand or working path based on a certain load splitting rule.*

Pre-configured: *A recovery resource that is prepared for establishment but needs to be activated. Variants include the case where an optical path or trail is configured, but no switches are set.*

Pre-established: *A recovery resource that is established prior to any failure on the working path.* [VSFH03]

[2]Details of the approximation can be found in [Ise99, Annex B]

Pre-reserved: *A recovery resource with reserved required resources on all hops along its route. The resources held by a set of recovery paths may be shared.* [VSFH03]

Protection: *A resilience mechanism that uses suited pre-planned, pre-configured, and pre-established backup resources.*

Reliability: *The probability of performing a specified function without failure under given conditions for a specified period of time.*

Rerouting: *Restoration in IP networks.*[3]

Resilience: *The capacity of a system exposed to threats to adapt by resisting or changing in order to reach and maintain an acceptable level of functioning and structure.*

Restoration: *A resilience mechanism in which backup resources for failure affected working paths are calculated, configured, and established dynamically after the detection of a fault.*

Shared Resources: *Reserved recovery resources that will be available to protect different working resources if the protected resources are not simultaneously subject to a failure.*

Working Resources: *A resource, e.g. a path, that is used in fault-free condition. Synonyms for a working resource are primary resource and active resource.* [IT03c, VSFH03]

3.4 Framework and Building Blocks

The previous sections revealed that a resilience classification framework facilitates the exact definition of resilience mechanisms and enables the comparison of existing and finding of new resilience mechanisms. Figure 3.2 shows the eight top-level building blocks of the novel *Resilience Classification Framework* (RCF). Each building block is described in detail in the following sub-chapters.

3.4.1 Internal Redundancy

The building block *Internal Redundancy* is divided further into three sub-categories as illustrated in Figure 3.3.

[3]Compare the definition of rerouting in RFC 3469 [VSFH03]: "*A recovery mechanism in which the recovery path or path segments are created dynamically after the detection of a fault on the working path. In other words, a recovery mechanism in which the recovery path is not pre-established.*"

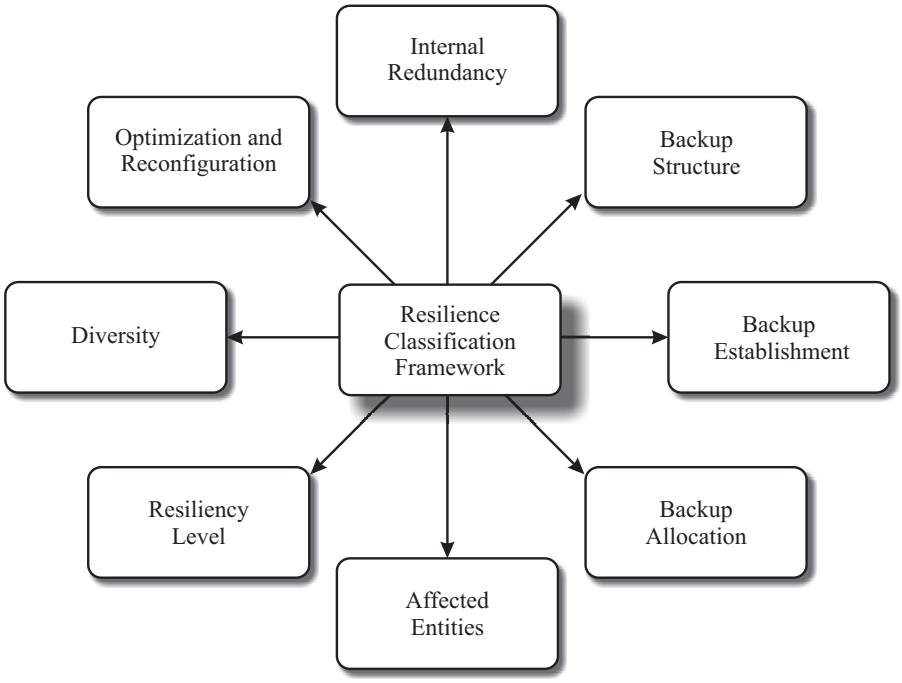


Figure 3.2: Top-level building blocks of the *Resilience Classification Framework*.

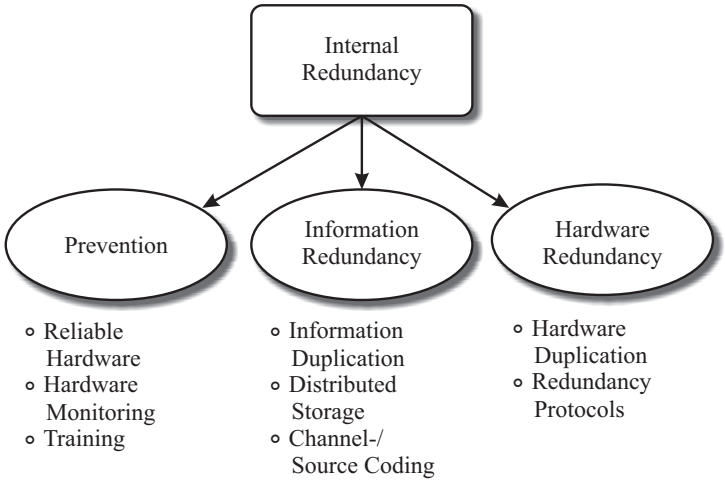


Figure 3.3: Sub-categories of building block *Internal Redundancy*.

3.4.1.1 Prevention of Failures

The best technique to reduce negative effects of network element failures is to prevent failures at all. Section 2.3.1 summarized the main reasons for network outages. Based on this information, counter-mechanisms can be installed that reduce the probability of element failures. Obviously, the *Mean Time Between Failures* (MTBF) can be increased by using equipment components that are well-tested and unlikely to fail. Similarly, a constant monitoring of equipment as well as the replacement of older equipment helps to reduce the probability of failures. Additionally, countermeasures can be installed that prevent failures that are caused by external influences. The proper marking of cable locations and the use of metal-shielded fibers, for example, can prevent cable-cuts by excavators. Last, but not least, one important, however often underestimated, issue is the training of personnel. Many failures are due to human error as shown in detailed analysis by Kuhn [Kuh97] and Crawford [Cra93].

3.4.1.2 Information Redundancy

If a failure in a communication network has occurred, transported data might be lost. Thus, in order to reduce the negative effects of data loss, information redundancy can be deployed. Data can be duplicated and transmitted along different paths (e.g. done with 1+1 protection) or procedures can be deployed that automatically retransmit data in case of a transmission failure. The *Transmission Control Protocol* (TCP) [PA00] for example monitors the delivery of data via an acknowledged transfer and starts a retransmission of lost data automatically. Furthermore, the use of source or channel coding, i.e. applying coding algorithms prior to the transmission via a channel (and reverse algorithms at the receiver), can improve the reliable transport of data.

3.4.1.3 Hardware Redundancy

Next to the prevention of failures and information redundancy, built-in hardware redundancy can increase the reliability of an element. If each component is unlikely to fail, the probability of a failure along a transmission path will be low. Thus, each component should have adequate MTBF values (see Section 2.3.1). For this, important components, like power supply, cooling fans, and forwarding equipment can be deployed in a redundant way inside the network element. Additionally, separate hardware devices can be coordinated to fulfill the tasks of the partner component in case of a failure. An example hardware redundancy protocol for IP networks is the *Virtual Redundancy Router Protocol* (VRRP) [RH04]. An automatic switch to a separate backup router is performed in case of a primary router failure.

3.4.2 Backup Structure

If a failure occurs and no internal redundancy was deployed, alternative transmission paths have to be found. Dependent on the used forwarding technology and approach either rout-

ing information (e.g. forwarding table entries), routing decisions (e.g. routing algorithms), or complete routes (e.g. MPLS paths) have to be changed. The building block *Backup Structure* can further be categorized into three different sub-categories as illustrated in Figure 3.4.

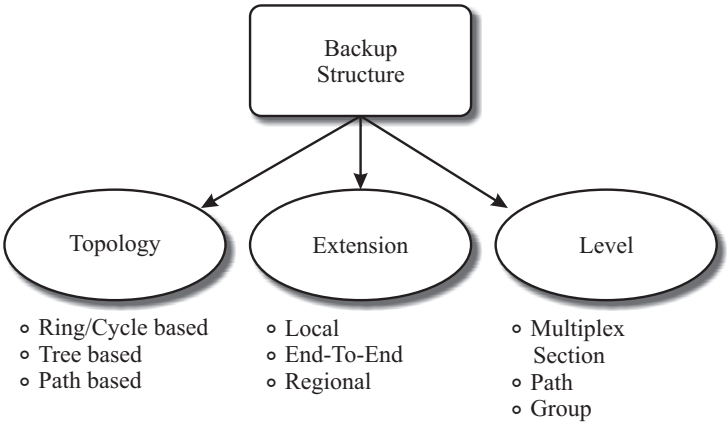
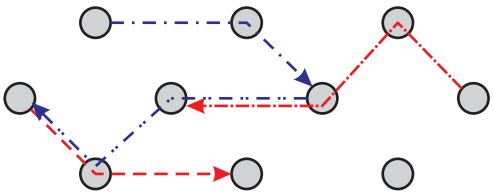


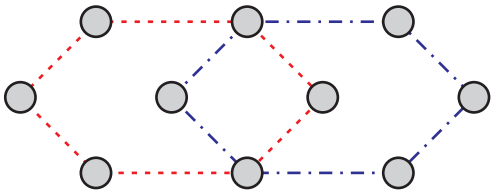
Figure 3.4: Sub-categories of building block *Backup Structure*.

3.4.2.1 Topology

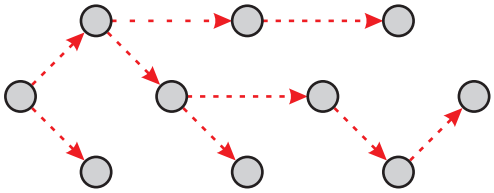
The most prominent characteristic of a backup structure is its topology. In order to divert traffic around failing regions traffic must be shifted along backup routes.



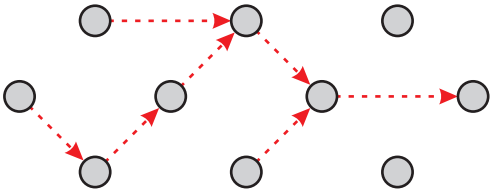
(a) Path Structure.



(b) Ring Structure.



(c) Source Tree Structure.



(d) Sink Tree Structure.

Figure 3.5: Example backup structure topologies.

Figure 3.4.2.1 depicts four examples of backup topologies: Figure 3.5(a) shows path backup structures. Figure 3.5(b) shows a backup structures in which traffic can be detoured

along ring-structures. Finally, Figures 3.5(c) and 3.5(d) show structures forming a source and sink tree, respectively.

Path Structures Paths are the most general form of backup structures. Any backup structure (e.g. ring, tree, and trail) can be built by a combination of backup paths. Backup paths can be assigned to individual working paths or groups of working paths. Since path-based resilience mechanisms provide the largest flexibility for network planning, we will present more details on path-based resilience mechanisms in Section 3.5.1.

Ring Structures Rings are simple structures that inherently provide a disjoint route between all nodes of the ring. Equipment that switches traffic to backup paths need no complicated forwarding configuration. Therefore, ring schemes have been used quite often in telecommunication systems in the past (e.g. SONET BLSR [ANS95]). A detailed overview of ring schemes is provided in [Gro04] and [VPD04].

Tree Structures Tree structures are often one of the first choices when using destination based forwarding. These structures are widely deployed as shortest path trees (e.g. in OSPF) or spanning trees (e.g. in Ethernet). Especially, when using connection-oriented technologies that do not include routing information in their data (e.g. conventional optical networks), backup sink trees can be used to route a group of working paths towards the same destination. A detailed study on backup trees can be found in [GCM+03]. Additionally, redundant backup trees can be used to provide protected multicast [FCGC01].

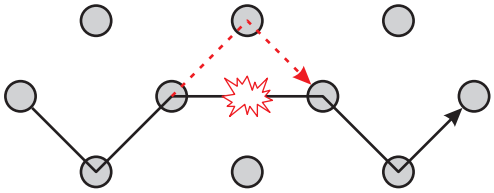
3.4.2.2 Extension

Next to backup structure, the locations of start- and end-points of traffic detours around a failing region are important properties of a resilience mechanism. If traffic has to be diverted at nodes that are not able to detect a failure, signaling mechanisms will be required.

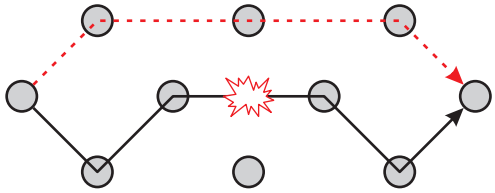
Figure 3.6 illustrates three backup structure extensions. Local detours start immediately in front of the failure and end at the opposite side of the failing region (Figure 3.6(a)). There, traffic is reverted to the original working path. End-to-end detours, in contrary to that, detour traffic at the original source of the demand and route the traffic towards the sink of the demand as illustrated in Figure 3.6(b). More generally, regional mechanisms are able to detour regions or segments of the affected path. Dependent on start and end location of the detour, local, end-to-end, and any constellation in-between can be formed.

3.4.2.3 Level

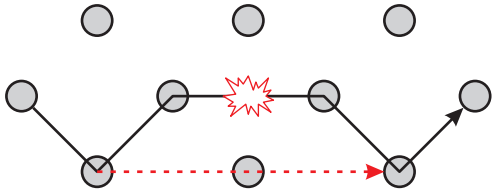
In addition to the backup path structure, a protection level can be defined. In case of a network element failure, usually more than one working path is affected. Thus, in principle, backup paths for all affected paths have to be found. However, often one backup path is



(a) Local.



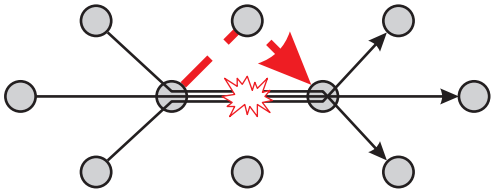
(b) End-to-end.



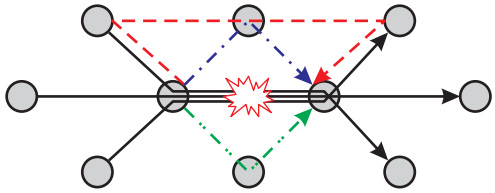
(c) Regional/Segment.

Figure 3.6: Example backup structure extensions.

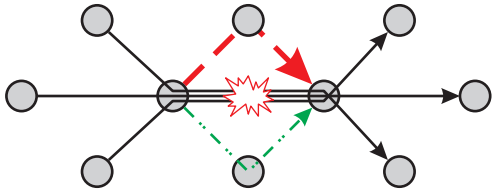
able to protect multiple working paths. Thus, it is possible to group working paths and protect them using one backup path only. Obviously, dependent on the level of grouping the number of backup paths can be reduced drastically. However, when grouping paths, larger capacity amounts have to be detoured and backup paths with suited capacity have to be found. Thus, longer paths and with that more capacity might be required.



(a) Multiplex Section Level.



(b) Path Level.



(c) Individual Group Level.

Figure 3.7: Example backup structure levels.

Figure 3.7 shows three types of backup structure levels. In Figure 3.7(a) only one backup path is established to detour the affected traffic for the whole failing multiplex-section. In contrast to that, backup paths can be assigned individually to each affected

path when using path level as shown in Figure 3.7(b). Finally, Figure 3.7(c) shows an example of a mixture of different backup structure levels in which some paths are grouped.

3.4.3 Backup Establishment

Backup resources can be established as a reaction to a failure (*reactive resilience mechanisms*) or can be prepared in advance of (probable) failures (*proactive resilience mechanisms*). Three main events have to be taken into account:

- Calculation: Which backup resources should be used?
- Configuration: At which time are the backup resources configured?
- Activation: At which time can the backup resources be used?

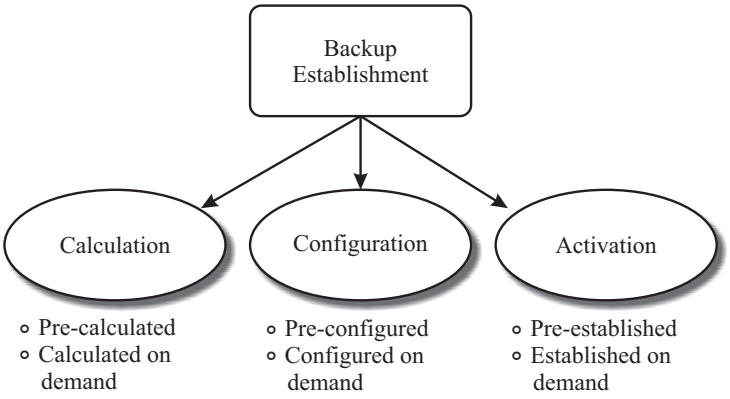


Figure 3.8: Sub-categories of building block *Backup Establishment*.

3.4.3.1 Calculation

As seen in Section 2.1 the choice of backup paths has a significant influence on the characteristics of the resilience mechanism. Backup paths can be calculated in advance of any failure or can be calculated after the occurrence of a failure. Obviously, pre-calculation helps to speed up a reaction upon failures. However, since the number of failures is immense, pre-calculations are usually performed for the most probable failures only.

3.4.3.2 Configuration

After appropriate backup resources have been calculated, they can be configured. Similarly, the configuration can be performed in advance of a failure or as a reaction to a failure. Optical cross connects along a backup path, for example, can be pre-configured in order to facilitate the switching of mirrors in case of a failure.

3.4.3.3 Activation

Finally, backup resources can be activated in order to establish the backup resource. Again, the activation of backup resources can be performed in advance of a failure or as a reaction to a failure. After activation, the path is established and can be used.

Table 3.1 summarizes proactive and reactive backup establishment possibilities and corresponding terminology.

Table 3.1: Proactive and reactive possibilities for *Backup Establishment*.

Name	Calculation		Configuration		Activation	
	proactive	reactive	proactive	reactive	proactive	reactive
Protection	x		x		x	
Pre-Configured Restoration	x		x			x
Pre-Calculated Restoration	x			x		x
Restoration		x		x		x

3.4.4 Backup Allocation

The building block *Backup Allocation* determines the affiliation and utilization of the backup resources. Figure 3.9 illustrates the two sub-categories of backup allocation: *Sharing* and *Usage*.

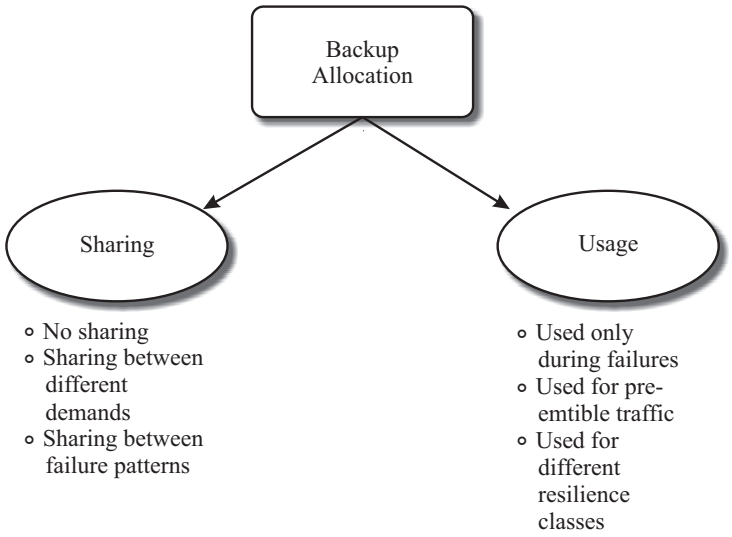
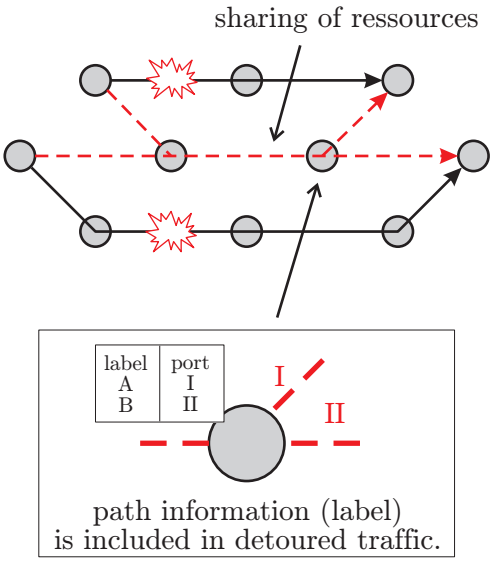


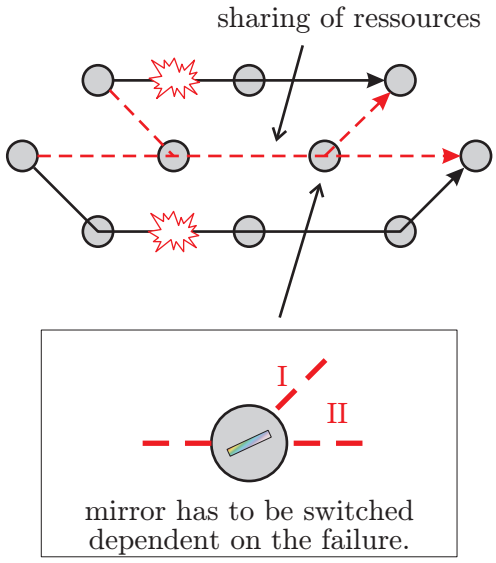
Figure 3.9: Sub-categories of building block *Backup Allocation*.

3.4.4.1 Sharing of Resources

Backup resources can be shared between different working resources (*shared usage*) or can be used for one working resource only (*dedicated usage*). Similarly, backup resources can be used in case of different failure patterns. Note however, dependent on the used technology, a sharing of backup resources may prevent the proactive establishment of backup paths. An example using MPLS and DWDM technology is illustrated in Figure 3.10. With MPLS, the switching decision is determined by information that is included in the MPLS header (label): A proactive establishment of both backup paths is possible. With WDM technology however, only one mirror position is possible at a given time. Thus, the appropriate mirror needs to be adapted accordingly after the occurrence of a failure.



(a) MPLS - The establishment of a backup path with shared capacity can be performed proactively.



(b) WDM - The establishment of a backup path with shared capacity requires a reactive establishment.

Figure 3.10: Examples of capacity sharing and influences on proactive and reactive establishment possibilities.

3.4.4.2 Usage of Resources

Backup resources might also be used to transport low priority traffic if no failure has occurred. Although the transport of additional unprotected traffic is advantageous from a network operators revenue point of view, additional mechanisms have to be deployed to assure that the extra traffic is pre-emptied in case of a failure.

3.4.5 Affected Functional Units

The building block *Affected Functional Units* models the required action of units that are used for the recovery process. Figure 3.11 depicts the three sub-categories of this building block.

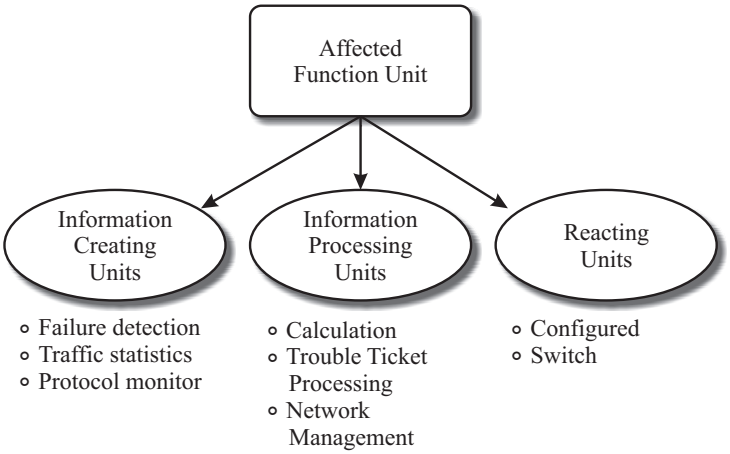


Figure 3.11: Sub-categories of building block *Affected Functional Units*.

3.4.5.1 Information Generating Entities

The first sub-category models the units of the network that are used to collect and generate information that are required for resilience purposes. Generated information ranges from failure indication signals to traffic or protocol monitoring data.

3.4.5.2 Information Processing Entities

The second sub-category models units that process the generated information in order to find or activate a suitable resilience mechanism. *Path Computation Elements* (PCEs) or a centralized operation and management center are examples for these kinds of network entities.

3.4.5.3 Reacting Entities

The third sub-category models units that react upon failures (e.g. perform switching operations). Entities that generate information, process information, and finally react are identical in some resilience mechanisms.

3.4.6 Resilience Level

Figure 3.12 depicts the building block *Resilience Level* that can furthermore be separated into two sub-categories: *Granularity* and *Survivability*.

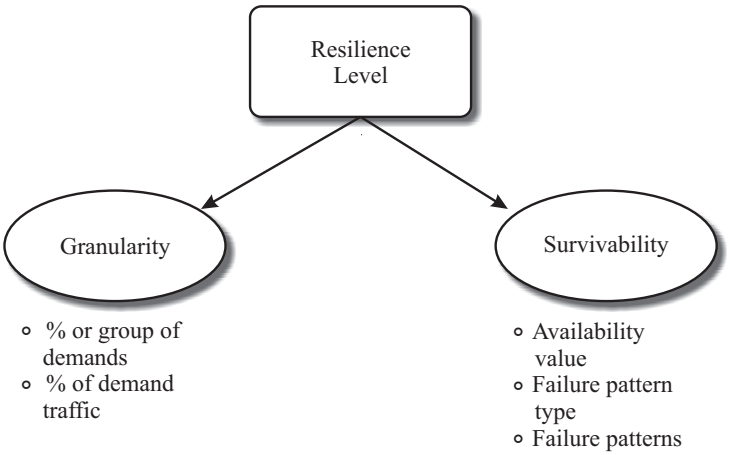


Figure 3.12: Sub-categories of building block *Resilience Level*.

3.4.6.1 Granularity

Sub-category *Granularity* models the amount of traffic that should be protected. In many cases, backup resources should be dimensioned to provide enough capacity to detour all traffic of a demand, path, or link. However, a design that assures appropriate backup resources for a fraction of traffic only is also conceivable.[4]

3.4.6.2 Survivability

The level of survivability is modeled by sub-category *Survivability*. While end-to-end availability values directly reflect a survivability value of a demand, a resilient network design that guarantees to provide enough resources for given failure pattern types or individual failure patterns is conceivable.

3.4.7 Diversity

The possibility to split a traffic demand into several parts and route them along different paths was mentioned in Section 2.1.4 already. Figure 3.13 illustrates the two sub-categories of building block *Diversity*: *Multipath* and *Traffic Distribution*.

3.4.7.1 Multipath

Sub-category *Multipath* models the possibilities and restrictions of the number of working and resilience path splits. While a large number of demand splits are possible in some technologies, e.g. MPLS, no splits or a reduced number of splits might be desirable to reduce the complexity of the resilience mechanism.

[4] A similar definition for this sub-category is given in RFC 3469 [VSFH03] that defines *Recovery Granularity* as "*the amount of traffic requiring protection. This may range from a fraction of a path to a bundle of paths.*"

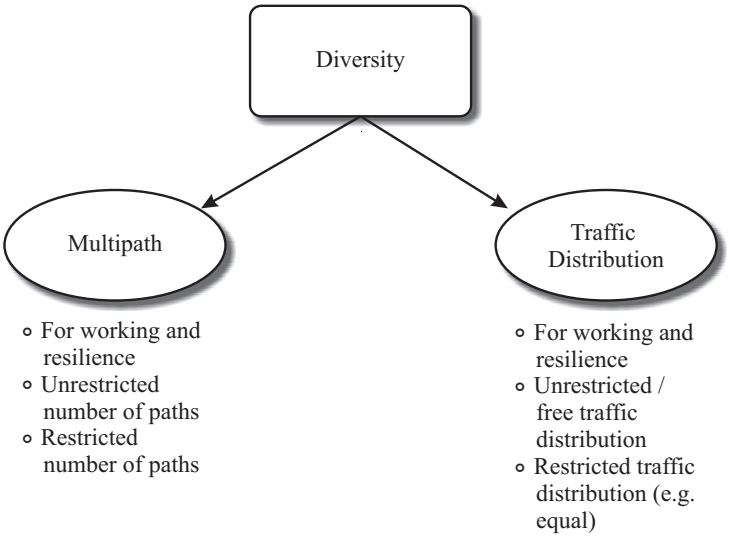


Figure 3.13: Sub-categories of building block *Diversity*.

3.4.7.2 Traffic Distribution

Similarly, the characteristics and possibilities of traffic distribution algorithms can be modeled by sub-category *Traffic Distribution* for working and backup traffic. Obviously, if multipath routing is not allowed, traffic distribution characteristics will not be applicable.

3.4.8 Optimization and Reconfiguration

Additional characteristics, timing issues, configuration, and optimization characteristics can be described in building block *Optimization and Reconfiguration*. The sub-categories are illustrated in Figure 3.14.

3.4.8.1 Optimization Approach, Target, and Objective

Sub-categories *Optimization Approach*, *Optimization Target*, and *Objective* define the used algorithms with which backup resources are selected. Considering dynamic networks in which demands are added, removed, and changed frequently, iterative additions or small changes of the network can be beneficial from a route-stability point of view. However, a reoptimization of all existing routes can be performed to reduce the amount of required resources.

3.4.8.2 Location and Information

The location of the optimization process, if centralized or distributed, as well as the available information will have a significant impact on the overall characteristics of a resilience mechanism. While a centralized approach facilitates a coordination of different resilience reactions, distributed approaches are usually more robust against multiple failures.

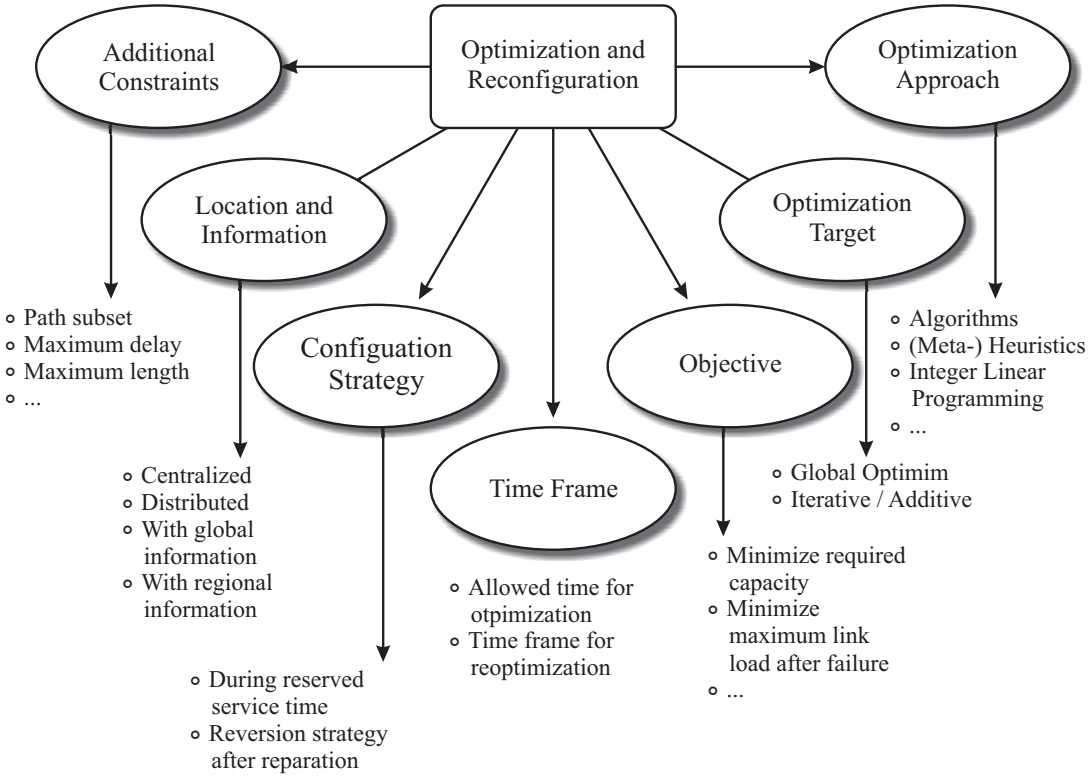


Figure 3.14: Sub-categories of building block *Optimization and Reconfiguration*.

3.4.8.3 Configuration Strategy and Time Frame

Similarly, different reversion strategies are conceivable, i.e. strategies whether and when to reactivate a working path after the successful reparation of a failure. Furthermore, the intervals in which reoptimizations are performed and the maximum allowed time for optimization can be modeled in sub-category *Time Frame*.

3.4.8.4 Additional Constraints

Finally, and dependent on network operator preferences or technology issues, the number and characteristics of backup resources can be restricted further. A large number of constraints are possible such as a length restriction of working and backup paths to guarantee a maximum delay for traffic with strict QoS requirements.

3.5 Example Classifications and Comparison

In the following sub-sections, we present example classifications of common resilience mechanisms that guarantee to survive single link failures. Following this, we perform a comparison of the mechanisms based on the RCF description.

3.5.1 Path Protection Mechanisms

3.5.1.1 Shared End-to-End Path Protection

Shared End-to-End Path Protection (SE2EPP) is a path-based protection mechanism that uses one or several pre-configured backup paths that are able to protect one or several working paths.[5] While start and end nodes of working and backup paths are identical, the backup paths are routed disjoint to protected equipment of the working path. The capacity on backup paths is pre-reserved and can be shared to protect different working paths (of the same demand or other demands) that cannot fail simultaneously (respective to the considered failure pattern). If a failure of a network element along a working path occurs, the source node will be informed about the failure via a signaling mechanism.[6] After activation of appropriate backup path(s) the traffic that would originally traverse the erroneous path(s) are detoured along the backup path(s) towards the destination node. Figure 3.15 depicts an example in which two working paths (w_1 and w_2) are protected with SE2EPP. Table 3.2 shows the classification according to the RCF.

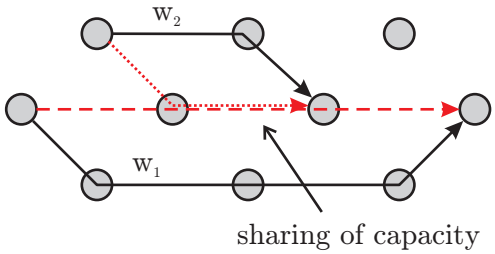


Figure 3.15: Illustration of *Shared End-to-End Path Protection*.

3.5.1.2 Demandwise Shared Path Protection

Demandwise Shared Path Protection (DSPP) [**GKO+05**, **GKZ+05**] is a path-based protection mechanism that uses one or several pre-configured backup paths that are able to protect one or several working paths. While start and end nodes of working and backup paths are identical, the backup paths are routed disjoint to protected equipment of the working path. The capacity on backup paths is pre-reserved and can be shared to protect

[5] SE2EPP is also denoted Shared Backup Path Protection (SBPP) in [Gro04].

[6] The detection of network element failures as well as the used signaling protocol may depend on the technology. As an example hardware detection (e.g. loss of signal) or failure detection mechanisms similar to *Bidirectional Forwarding Detection* (BFD) [KW06] are applicable.

Table 3.2: Classification of *Shared End-to-End Path Protection*.

Internal Redundancy	Prevention	N/A
	Information Redundancy	N/A
	Hardware Redundancy	N/A
Backup Structure	Topology	Path
	Extension	End-to-end
	Level	Path
Backup Establishment	Calculation	Pre-calculated
	Configuration	Pre-configured
	Activation	Pre-established
Backup Allocation	Sharing	Shared between working paths of all demands that cannot fail simultaneously
	Usage	In case of a failure
Affected Functional Units	Information Creation	Adjacent to failure. May be generated by demand end-nodes.
	Information Processing	Demand end-nodes
	Reacting	Demand end-nodes
Resiliency Level	Granularity	100% traffic should be protected
	Survivability	At least single link failures
Diversity	Multipath	Possible
	Traffic distribution	Unequal distribution possible
Optimization	Constraints	Dependent on technology and network operator's choice. Path length restrictions are often used as well as a reduction in the number of candidate paths in order to reduce the complexity of the optimization.
	Location	Centralized. In some variants also distributed in end-nodes of the path with global information (e.g. in MPLS [Cis06]).
	Configuration Strategy and Time Frame	Dependent on the operator's choice. Optimization approaches that provide (near) optimal constellations are often used. Their running time varies between few seconds and some hours.
	Objective	Different objectives are conceivable and are dependent on the operator's choice. Often minimization of overall CAPEX is demanded.
	Target	Different targets are conceivable. Often a number of probable failure patterns have to be survived while end-to-end QoS requirements have to be fulfilled.
	Approach	Algorithms, meta-heuristics, or other optimization approaches.

different working paths of the same demand that cannot fail simultaneously (respective to the considered failure pattern). If a failure of a network element along a working path occurs, the source node will be informed about the failure via a signaling mechanism similar to SE2EPP. After activation of appropriate backup paths, traffic that would originally traverse the erroneous path is detoured along a backup path towards the destination node. Figure 3.16 illustrates an example in which one demand with two working paths w_1 and w_2 is protected with DSPP. Table 3.3 shows the classification according to the RCF.

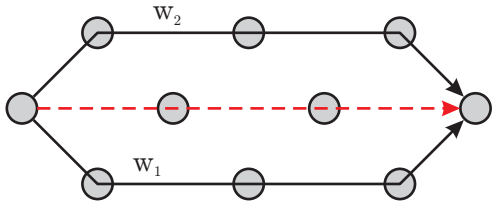


Figure 3.16: Illustration of *Demandwise Shared Path Protection*.

3.5.1.3 Shared Regional Path Protection

Shared Regional Path Protection (SRPP) is a path-based protection mechanism that uses one or several pre-configured backup paths that are able to protect a region of one or several working paths. Each backup path starts and ends at the path-region that is protected. The capacity on the backup paths is pre-reserved and can be shared between different backup paths that do not fail simultaneously (respective to the considered failure pattern). If a failure of a network element along a working path occurs, the source-switching node that is in between a configurable interval in front of the failure will be informed about the failure via a signaling mechanism similar to SE2EPP. After activation of appropriate backup paths, the traffic that would originally traverse the erroneous path is detoured along the backup path towards the destination-switching node. Figure 3.17 depicts an example in which two demands w_1 and w_2 are protected with SRPP. More details on SRPP will be given in Chapter 4. Table 3.4 shows the classification according to the RCF.

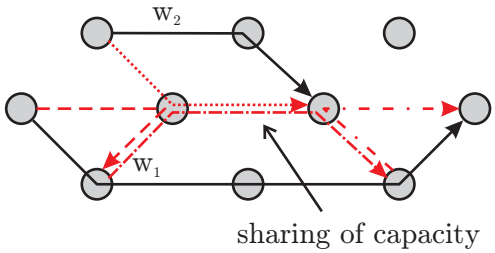


Figure 3.17: Illustration of *Shared Regional Path Protection*.

Table 3.3: Classification of *Demandwise Shared Path Protection*.

Internal Redundancy	Prevention	N/A
	Information Redundancy	N/A
	Hardware Redundancy	N/A
Backup Structure	Topology	Path
	Extension	End-to-end
	Level	Path, Group or Multiplex Section
Backup Establishment	Calculation	Pre-calculated
	Configuration	Pre-configured
	Activation	Pre-established
Backup Allocation	Sharing	Shared between disjoint working paths of the same demand that cannot fail simultaneously
	Usage	In case of a failure
Affected Functional Units	Information Creation	Adjacent to failure. May be generated by demand end-nodes.
	Information Processing	Demand end-nodes
	Reacting	Demand end-nodes
Resiliency Level	Granularity	100% traffic should be protected
	Survivability	At least single link failures
Diversity	Multipath	Possible
	Traffic distribution	Unequal distribution possible
Optimization	See Table 3.2	

3.5.1.4 Shared Local Link Path Protection

Shared Local Link Path Protection (SLLPP) is a path-based protection mechanism that uses one or several pre-configured backup paths that are able to protect one common link of one or several working paths. Each backup path starts and ends at the link that is protected but does not use the link itself. The capacity on backup paths is pre-reserved and can be shared between different backup paths that cannot be used simultaneously (respective to the considered failure pattern). After the detection of a link-failure by failure adjacent nodes the appropriate backup paths are activated and traffic that would originally traverse the erroneous link is detoured to the backup paths towards the other end of the failed link. Figure 3.18 depicts an example in which two demands w_1 and w_2 are protected with SLLPP. Table 3.5 shows the classification according to the RCF.

Table 3.4: Classification of *Shared Regional Path Protection*.

Internal Redundancy	Prevention	N/A
	Information Redundancy	N/A
	Hardware Redundancy	N/A
Backup Structure	Topology	Path
	Extension	Regional. The start and end-nodes of the detour can be configured.
	Level	Path, Group or Multiplex Section
Backup Establishment	Calculation	Pre-calculated
	Configuration	Pre-configured
	Activation	Pre-established
Backup Allocation	Sharing	Shared between working paths of the all demands that cannot fail simultaneously
	Usage	In case of a failure
Affected Functional Units	Information Creation	Adjacent to failure. May be generated by region end-nodes.
	Information Processing	Region end-nodes
	Reacting	Region end-nodes
Resiliency Level	Guarantee	Guaranteed for single failures
	Survivability	Failure pattern driven
Diversity	Multipath	Possible
	Traffic distribution	Unequal distribution possible
Optimization	See Table 3.2	

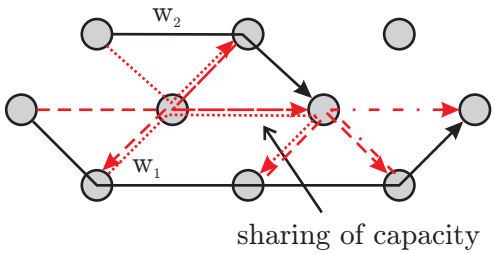


Figure 3.18: Illustration of *Shared Local Link Path Protection*.

Table 3.5: Classification of *Shared Local Link Path Protection*.

Internal Redundancy	Prevention	N/A
	Information Redundancy	N/A
	Hardware Redundancy	N/A
Backup Structure	Topology	Path
	Extension	Local
	Level	Path, Group or Multiplex Section
Backup Establishment	Calculation	Pre-calculated
	Configuration	Pre-configured
	Activation	Pre-established
Backup Allocation	Sharing	Shared between disjoint working paths of all demands that cannot fail simultaneously
	Usage	In case of a failure
Affected Functional Units	Information Creation	Adjacent to failure.
	Information Processing	Link end-nodes
	Reacting	Link end-nodes
Resiliency Level	Granularity	100% traffic should be protected
	Survivability	At least single link failures
Diversity	Multipath	Possible
	Traffic distribution	Unequal distribution possible
Optimization	See Table 3.2	

3.5.2 Pre-configured Protection Cycles (*p*-Cycles)

The *p*-Cycle concept [Gro04, Sch05, **SGA02**, **GS02**, **Gru03b**] is a ring-based protection mechanisms that uses one or several pre-configured backup rings. Two basic types of *p*-Cycles exist. Link *p*-Cycles protect individual channels within a link. Node-encircling *p*-Cycles are routed through all adjacent neighbor nodes of a node to be protected but exclude the protected node itself. Thus, all connections traversing the node are protected. In this document we focus on link *p*-Cycles.

Figure 3.5.2 depicts an example network with one link *p*-Cycle. The *p*-Cycle is able to protect on-cycle links as shown in Figure 3.19(a) by providing one detour along the cycle. Additionally, the same *p*-Cycle is able to protect so-called 'straddling' links. Straddling links are links whose endpoints are on-cycle nodes of one *p*-Cycle but do not belong to the cycle itself. Since two detour directions are possible along the cycle the working capacity of a straddling link can be divided into two parts. The *p*-Cycle of Figure 3.5.2 is therefore

(a) On-cycle failure. (b) Straddling failure.

Figure 3.19: Illustration of p-Cycle protection.

able to protect 9 on-cycle and 7 straddling links. Each backup path starts and ends at the link that is protected but does not use the link itself. The capacity on the cycle is pre-reserved and can be shared. Table 3.6 shows the classification according to the RCF.

3.5.3 Theoretical Comparison

The presented resilience mechanisms seem to differ substantially. Especially, the ring-based protection mechanism p-Cycle seems to be conceptually different. As mentioned above, even if resilience mechanisms are defined with similar sentences and illustration, a comparison of mechanisms and the finding of characteristics is difficult. However, when separating the characteristics of a mechanism into smaller building blocks an individual analysis and comparison is facilitated. By performing a comparison of Tables 3.2 to 3.6 differences appear easily. Table 3.7 summarizes the different properties of the five investigated resilience mechanisms.

Although the resilience mechanisms seemed to be quite different, the comparison results of Table 3.7 reveal that the mechanisms differ in very few characteristics only that are analyzed in the following.

Comparison of SE2EPP, SRPP and SLLPP:
Shared end-to-end path protection, shared regional path protection, and shared local link path protection differ in extension, processing, and reacting functional units. However, in principal, every path constellation of SLLPP can be achieved by SRPP as shown in Figure 3.20(a). Similarly, every SRPP constellation can be achieved by SE2EPP (Figure 3.20(b)).

Thus, the simple comparison reveals the following characteristics from a capacity requirement point of view:

Capacity requirements:

$$\text{SE2EPP} \leq \text{SRPP} \leq \text{SLLPP}$$

Since signaling time rises with an increased extension between information creating, processing, and reacting functional units, the recovery times of the three resilience mechanisms

Table 3.6: Classification of *p-Cycle Protection*.

Internal Redundancy	Prevention	N/A
	Information Redundancy	N/A
	Hardware Redundancy	N/A
Backup Structure	Topology	Ring
	Extension	Local
	Level	Path, Group or Multiplex Section
Backup Establishment	Calculation	Pre-calculated
	Configuration	Pre-configured
	Activation	Pre-established
Backup Allocation	Sharing	Shared between working paths of the same demand that cannot fail simultaneously
	Usage	In case of a failure
Affected Functional Units	Information Creation	Adjacent to failure
	Information Processing	Adjacent to failure
	Reacting	Adjacent to failure
Resiliency Level	Guarantee	Guaranteed for single failures
	Survivability	Failure pattern driven
Diversity	Multipath	Possible
	Traffic distribution	Unequal distribution possible
Optimization	See Table 3.2	

(a) All SLLPP paths can be formed by SRPP.

(b) All SRPP paths can be formed by SE2EPP.

Figure 3.20: Example constellation for capacity comparison of SE2EPP, SRPP, and SLLPP.

Table 3.7: Comparison of resilience classifications.

Building Block	Sub-Category	SE2EPP	DSPP	SRPP	SLLPP	p-Cycle
Backup Structure	Topology	Path	Path	Path	Path	Cycle
	Extension	End-to-end	End-to-end	Regional	Local	Local
Backup Allocation	Sharing	... all demands ...	... same demand ...	... all demands ...	... all demands ...	... all demands ...
Affected Functional Units	Information Processing	Demand end-nodes	Demand end-nodes	Region end-nodes	Link end-nodes	Link end-nodes
	Reacting	Demand end-nodes	Demand end-nodes	Region end-nodes	Link end-nodes	Link end-nodes

will be different. No signalization is required with local protection, and thus, from a recovery time point of view we can categorize as follows:[7]

Recovery time:

$$\text{SLLPP} \leq \text{SRPP} \leq \text{SE2EPP}$$

Comparison of SE2EPP and DSPP:
Similarly, shared end-to-end path protection and demandwise shared path protection differ in their sharing characteristic only. All DSPP constellations can be achieved with SE2EPP. However, some sharing potential cannot be exploited when using DSPP. Thus, from a capacity requirement point of view SE2EPP wins favor, while both will have similar recovery times.

Capacity requirements:

$$\text{SE2EPP} \leq \text{DSPP}$$

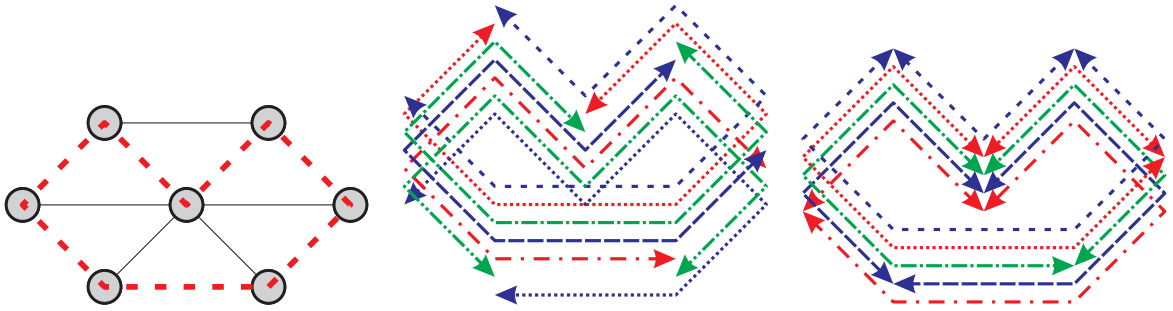
Recovery time:

$$\text{SE2EPP} = \text{DSPP}$$

Comparison of SLLPP and *p*-Cycle:
Furthermore, the comparison of characteristics of SLLPP and p-Cycle protection of Table 3.7 shows that the two mechanisms differ in the backup topology only. In fact, as mentioned earlier, any ring structure can be formed by combining path structures.

Figure 3.5.3 illustrates the combination of SLLPP paths that are required to protect on-cycle and straddling links similar to the p-Cycle concept. Thus, one can conclude,

[7]Neglecting the number of paths that are switched at the node. A detailed analysis of recovery times is presented in Section 5.3.



(a) Example p-Cycle constellation with 7 on-cycle and 5 straddling edges.

(b) Corresponding 7 on-cycle edge-protecting SLLPP paths.

(c) Corresponding 10 straddling edge-protecting SLLPP paths.

Figure 3.21: Example constellation for capacity comparison of p-Cycle and SLLPP protection.

that p-Cycle protection is a special type of SLLPP. Since the property to form cycle-like structures is a further constraint to SLLPP, we can furthermore categorize:

Capacity requirements:	
	SLLPP $\leq$ p-Cycle
Recovery time:	
	SLLPP $=$ p-Cycle

3.6 New Resilience Mechanisms

The resilience classification framework provides a unit assembly system with which new resilience mechanisms can be developed. A classification of popular resilience mechanisms has revealed that many building block combinations form suitable resilience mechanisms that wait patiently to be discovered. As an example, we will describe a novel resilience mechanism called *Self Regulating Traffic Distribution* (SRTD) that was developed by using a 'white-spot analysis' on the RCF.

3.6.1 Self Regulating Traffic Distribution

The presented resilience mechanisms of Section 3.5 are all based on pre-configured backup structures. These structures can be chosen and optimized according to traffic demand values and network topology information. In case of dynamic traffic patterns however, information about traffic demand has to be updated frequently in order to adapt working and backup resources. Thus, large quantities of signaling messages to optimization instances are required. Reactive actions are prolonged due to signalization requirements.

As shown in Chapter 2, the mesh structure of communication networks, however, already provides multiple loop-free paths from one node to any other node. Several loop-free paths that have equal lengths according to link weights can, for example, be calculated

with the OSPF routing protocol extension *Equal Cost Multipath Path* (ECMP). A similar routing concept called O_2 [SCK+03] is able to calculate and use several loop-free paths that need not to have equal length. Since paths do not change frequently in wired networks, the use of pre-calculated multiple paths in combination with a smart and fast traffic distribution mechanism could prevent the overload of network elements and could provide backup paths in case of network element failures.

The resilience mechanism *Self Regulating Traffic Distribution* (SRTD) [**GLS05**, **GSB06**] distributes traffic at a node autonomously and automatically towards outgoing links. To reduce signalization the mechanism uses local utilization information only. The concept can be split into three components that are described in the following:

- Local Traffic Distribution
- Upstream Signaling
- Demand Adaptation

Local Traffic Distribution Learned by multipath capable routing mechanisms each node maintains one or more routes, i.e. outgoing links, towards all known destinations. Figure 3.22 depicts an example configuration of an SRTD node. For simplicity and without loss of generality two outgoing links are possible for each destination (X,Y,Z).

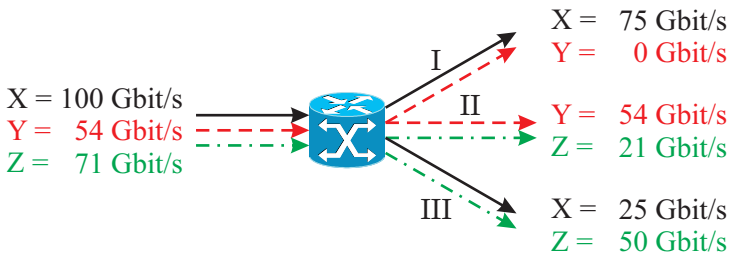


Figure 3.22: Model of an SRTD node.

The main aim of the *Local Traffic Distribution* concept is to prevent the overload of network elements. Thus, in a first phase, traffic has to be distributed evenly on outgoing links. Therefore, the maximum load of outgoing links should be minimized while an even distribution of traffic towards one destination is less important. Possible traffic distribution ratios for the example node are depicted in Table 3.8.

A node has to measure the amount of transit traffic towards different destinations.[8] Consequently, the distribution ratios have to be calculated and configured to load-balance the traffic. As can be anticipated from Figure 3.22 the task to find optimal traffic distribution ratios for all traversing demands will be difficult if routes overlap and the amount of incoming traffic towards different destinations varies. However, algorithmic and Integer Linear Programming formulations were developed and presented in [**GLS05**]. The use of traffic statistic collection in combination with traffic value thresholds can furthermore reduce the number of reconfigurations in the node.

[8]This measurement of through traffic is already implemented in modern hardware [Cis06].

Table 3.8: Routing table of the example SRTD node.

Destination	Outgoing Interface	Distribution Ration
X	I	0.75
	III	0.25
Y	I	0.0
	II	1.0
Z	II	0.296
	III	0.704

Upstream Signaling: Although the concept of local reaction and traffic distribution alone works well, the lack of global load information may cause an overload of downstream equipment. In some cases, congestion and traffic loss will possibly be reduced if upstream nodes have knowledge about downstream distribution problems and are able to shift traffic from edges towards the congested area towards other parts of the network. Thus, an additional upstream signaling concept has been developed. A node that is receiving too much traffic, i.e. the node has difficulties to find traffic distribution ratios so that no traffic is lost, can send traffic reduction request messages to upstream nodes. These requests are furthermore integrated in the traffic distribution ratio calculation of the upstream node. If possible, traffic ratios will be adapted in order to divert traffic away from the congested downstream node.

Demand Adaptation: If traffic cannot be shifted to other areas by the upstream node itself, further reduction messages will be initiated in upstream directions. However, if no further upstream node exist, e.g. the node is a border node of the network, or the limitation of incoming traffic was not accomplished after a configurable period of time, the amount of traffic that is locally generated at the node, will be reduced. Thus, traffic is blocked at the border of the network to prevent overload situations in the core. Limits imposed on distribution ratios and incoming traffic have to be canceled after a period of time to readapt to new traffic values. To prevent overload of network elements again, a slow start mechanism that slowly reduces limits has been proposed and investigated in [**GLS05**]. Figure 3.23 depicts an example network before and after redistribution of traffic with SRTD.

Resilience with SRTD: The concept of SRTD enables the dynamic control of traffic flow in the network. The same concept can be used as a reaction to network element failures. If a failure is detected using hardware detection or protocols such as bidirectional forwarding detection [KW06], a node upstream to the failure will readapt its traffic loads in order to shift failure affected traffic towards other outgoing links. If not enough capacity is available or no other outgoing link is provided, traffic reduction request messages will be sent to upstream nodes. Thus, behavior and reaction time are similar to a congestion and

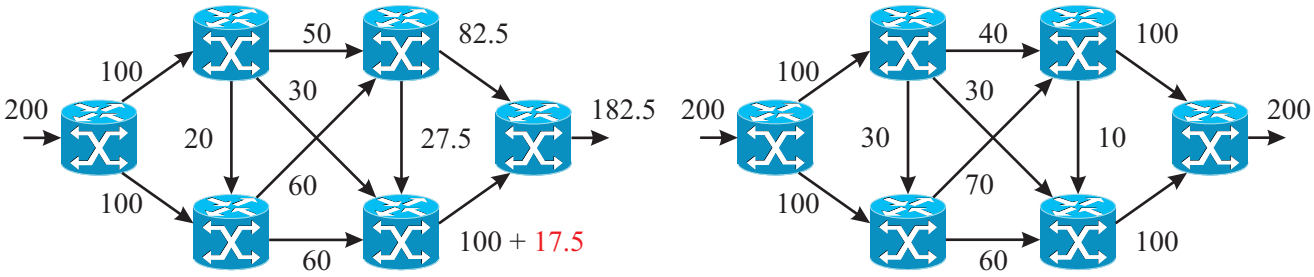


Figure 3.23: Example traffic distribution of an SRTD network.

traffic-engineering reaction. Case study results for recovery and dynamic traffic engineering reactions are discussed in detail in [**GLS05**].

In summary, SRTD provides a new dynamic distributed approach for traffic engineering and resilience. In contrast to traffic-engineering and resilience mechanisms that are based on global information, the reaction of SRTD is mainly based on local information. Few messages to upstream nodes are sufficient to reroute and detour traffic around congested areas. An overload of network elements can be prevented. Inherently, a mechanism to reduce the negative effects of any network failure is provided. Furthermore, since a local redistribution of traffic can be performed independently by all network elements, the concept provides no single point of failure and can be deployed step by step in today's networks. Table 3.9 summarizes the classification of SRTD according to the resilience classification framework.

3.7 Chapter Summary

In this chapter, we addressed the difficulty to classify and compare resilience mechanisms with each other. The introduction of a resilience classification framework and the decomposition of resilience mechanism characteristics into building blocks showed that a theoretical comparison of resilience mechanisms is facilitated. Classifications of five popular resilience mechanisms furthermore revealed that the differences between these resilience mechanisms are rather small. Additionally, we showed that the classification enables a theoretical analysis and categorization of capacity requirements and recovery times of different resilience mechanisms. Finally, the RCF based classification facilitates the design of novel resilience mechanisms since different characteristics can be combined and white-spot analysis reveal new resilience mechanisms.

Table 3.9: Classification of *Self-Regulating Traffic Distribution*.

Internal Redundancy	Prevention	N/A
	Information Redundancy	N/A
	Hardware Redundancy	N/A
Backup Structure	Topology	(multiple) Paths (Hammock Structure)
	Extension	Local
	Level	Path
Backup Establishment	Calculation	Paths pre-calculated. Distribution weights calculated on demand.
	Configuration	Configured on demand
	Activation	Activated on demand
Backup Allocation	Sharing	Shared
	Usage	In case of a failure
Affected Functional Units	Information Creation	Adjacent to failure
	Information Processing	Adjacent to failure
	Reacting	Adjacent to failure
Resiliency Level	Guarantee	No guarantee without traffic admission control.
	Survivability	Failure pattern driven
Diversity	Multipath	Required
	Traffic distribution	Unequal distribution possible
Optimization	Constraints	Sub-set of routes that are determined by a multipath capable routing algorithm. The routes may be chosen for example to fulfill end-to-end delay constraints.
	Location	Distributed in every SRTD node.
	Configuration Strategy and Time Frame	Time Frame: Fast calculation and very fast reconfiguration.
	Objective	Different objectives are conceivable and are dependent on the operator's choice, e.g. minimization of maximum outgoing link load.
	Target	Different targets are conceivable. E.g. the possibility to survive single outgoing link failures.
	Approach	Algorithms, meta-heuristics, or other optimization approaches.

Chapter 4

Resilient Network Optimization

There exist various possibilities to extend or design new networks (green-field planning). Network providers using simple mechanisms like shortest path routing in combination with a simple protection mechanism scheme are certainly able to design a resilient network. However, the investigation of the network planning cycle of Chapter 2 has revealed that the individual tasks of network planning are dependent on each other. Especially the joint optimization of topology selection, routing, and dimensioning in failure-free and failure-affected network states can increase network performance and can reduce expenditures significantly. Cost savings of more than 100% will be possible if more emphasis is put on the right choice of the used resilience mechanism, the optimization of failure-free routes, and the reaction in case of failures.[1] Thus, optimization approaches for the planning of resilient networks are required.

It is an ongoing discussion in which time frame these network optimization approaches should be performed, i.e. how long it should take between the start of a network optimization program and its solution. Certainly, algorithms that run close to real time are required to analyze small network changes or to adapt quickly to changing demands (e.g. a path set-up). However, these small changes of the network, need not to be optimal from a cost point of view and can be performed with rather simple approaches. In contrast to that, resilient network planning procedures that decide about large investments and the performance of the network for the next years require a thorough planning. Overnight run times or run times in the range of some days will be acceptable if the solution can cut down on costs or improve the quality of the network.

In this chapter, we discuss network optimization approaches that exist in the general literature. We will show, however, that most of these optimization approaches do not provide information about the quality of the obtained solution (Section 4.1). Therefore, we will focus on optimization approaches based on *Linear Programming* (LP). After introducing linear programming (Section 4.1.2), we will develop and present novel linear programming models for the resilient network design in Section 4.2. In particular, we will present complete models for two different formulation approaches (flow and path approach) and for

[1] For example by using shared path protection instead of dedicated 1+1 protection mechanisms.

five path-based resilience mechanisms. Furthermore, in Section 4.2.4 we will discuss the mathematical theory of duality in linear programming and present a new approach that enhances the planning of resilient networks considerably by using a mathematical technique called *'Column Generation'*. Finally, Section 4.3 will summarize the main contributions and conclude the chapter.

4.1 Introduction

4.1.1 Optimization Approaches and Quality of a Solution

A large number of optimization approaches have been developed in the last decades to plan resilient networks, to determine working and backup paths, and to dimension network elements. While simple algorithmic approaches like shortest path calculations and shortest disjoint path algorithms have been used to provide working and backup paths in commercial planning tools in the past (e.g. by [Bha99]), more and more sophisticated heuristic approaches are used today (e.g. in '*OnePlan*' by VPI Systems [VPI06], '*NPAT*' by Wandl [Wan] or '*NetWorks*' by Detecon [Det06]). Especially, heuristic approaches based on *Simulated Annealing* (e.g. [DSS03]), *Genetic Algorithms* (e.g. [Rie04]), *Tabu Search* (e.g. [ZGL05]) and recently *Particle Swarm Optimization* (e.g. [ZDL06]) are being constantly improved in order to provide quicker and better solutions.[2]

Finding feasible solutions to the problem, however, is only part of the objective of resilient network planning. From a network planner's point of view, either the (cost-) optimal solution should be obtained or at least some information about the quality of the solution should be provided. In other words, the difference between the current solution and the (unknown) optimal solution (optimality gap) should be available. Additionally, if the optimality gap could be obtained during the optimization process, it would be possible to stop difficult problems with long-lasting optimization times earlier, if sub-optimal solutions with small optimality gaps (e.g. $< 5\%$) were acceptable.

Unfortunately, heuristic approaches do not provide information about the quality of the solution and provide no guarantee to obtain the optimal solution. In addition, theoretical lower bounds on the capacity requirements for resilient networks are not very strict.[3] Thus, little statements can be given to the estimated cost of an optimal solution at the beginning.

In contrast to that, optimization approaches that are based on linear programming are able to calculate the mathematical optimal solution. In addition, lower bounds on the possible solutions are provided during the calculation process implicitly.

[2] A detailed overview of many optimization approaches for network planning can be found in [Rob99]. A short description of the principle of Simulated Annealing and Genetic Algorithms can be found in Appendix B.

[3] The currently best lower bound of the overall required capacity used for protection purpose in relation to the required working capacity is $\frac{1}{\bar{d}-1}$, with $\bar{d}$ being the average node degree. A detailed analysis can be found in [DG01] and [Sch05].

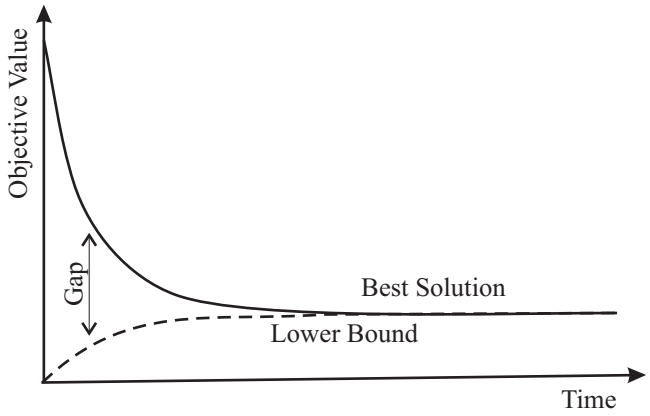


Figure 4.1: Example LP optimization run. The gap between the current solution and the minimum obtainable result (lower bound) is known during the solution process.

Figure 4.1 depicts an example linear programming optimization run. By relaxing the problem, i.e. neglecting some constraints[4], lower bounds are provided implicitly during the optimization. When reducing the number of relaxations during the optimization process, the lower bound increases while the upper bound decreases. Finally, the solutions for both formulations will converge, if the optimal solution is found.

Especially in the last 10 years, optimization approaches that are based on solving linear equation systems (linear programming) have evolved from a shadow existence and generated more interest in the optimization community since computer power, software programming, and equation solving approaches evolved dramatically. Therefore, we will provide formulations for resilient network planning in the following.

4.1.2 Linear Programming Fundamentals

A linear program is an optimization problem with a linear or piecewise linear and convex objective function and linear constraints [BT97]. The most general form of a *Mixed Integer Linear Program* (MILP) can be written as:

$$\text{minimize} \quad (\mathbf{c}^T\mathbf{x} + \mathbf{d}^T\mathbf{y}) \tag{4.1a}$$

$$\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{y} \leq \mathbf{b} \tag{4.1b}$$

$$\mathbf{x} \geq 0, \quad \textit{real} \tag{4.1c}$$

$$\mathbf{y} \geq 0, \quad \textit{integer} \tag{4.1d}$$

where $\mathbf{A}$ is an m by n matrix, $\mathbf{B}$ is an m by p matrix, $\mathbf{c}$ and $\mathbf{d}$ are n-dimensional and p-dimensional row vectors, $\mathbf{b}$ is an m-dimensional column vector and $\mathbf{x}$ and $\mathbf{y}$ are n-dimensional and p-dimensional column vectors of (unknown) variables. Obviously,

[4]As an example, an integer variable $i \in \mathbb{N}$ can be exchanged by a real variable $k = i + j$ $(j, k \in \mathbb{R})$. Obviously, any solution for the integer problem can be obtained by the real-value solution. Therefore, the relaxed LP problem provides a lower bound for the *Integer Linear Programming* (ILP) problem.

maximization problems can be formulated by minimizing the linear objective function $-\mathbf{c}^T\mathbf{x} - \mathbf{d}^T\mathbf{y}$. Similarly, an equality constraint $a_i x_i = b_i$ can be rewritten by two constraints $a_i x_i \leq b_i$ and $a_i x_i \geq b_i$.

There exist mathematical approaches to solve these kinds of formulations and to prove that a given solution is globally optimal. Thus, the basic idea of network optimization using linear programming is to formulate the optimization problem as linear program.

4.1.2.1 Solving Approaches

To understand linear programs in more detail we will first discuss the geometry of a linear program.

The example linear program

$$\text{maximize} \quad x_1 + x_2 \tag{4.2a}$$
$$\text{subject to:} \quad x_1 + 2x_2 \leq 7 \tag{4.2b}$$
$$2x_1 + x_2 \leq 6 \tag{4.2c}$$
$$x_1, x_2 \geq 0 \tag{4.2d}$$

can be represented geometrically as illustrated in Figure 4.2. The feasible region (polyhedra) is restricted by four equations resulting in the gray shaded area. The dotted lines show isolines, i.e. points on which the objective function $x_1 + x_2$ has the same value. As can easily be seen, the maximum possible value of $x_1 + x_2 = \frac{13}{3}$ is an edge point of the feasible region (indicated as circle in the figure).

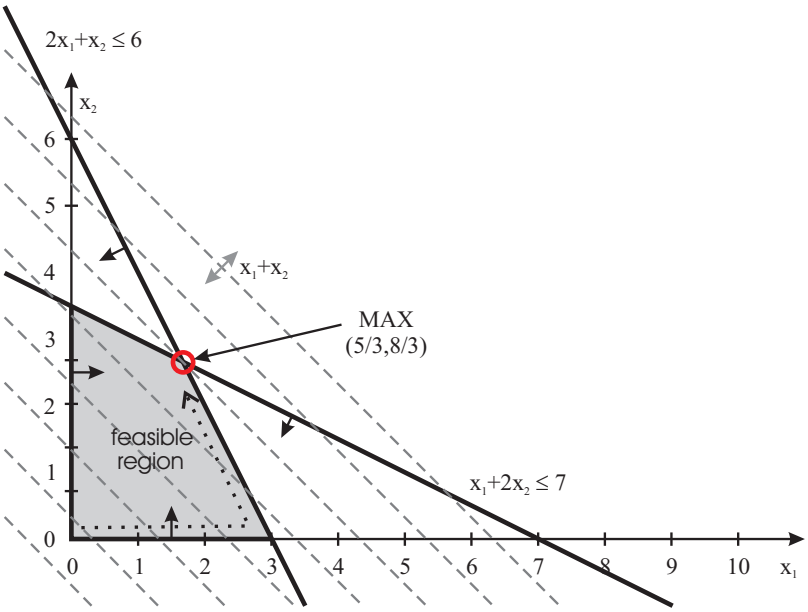


Figure 4.2: Geometric illustration of the LP equations.

Actually, this characteristic is common to all linear programs: The optimum value lies on an extreme point, a corner of the polyhedra.[5] Following this observation, an intuitive solving approach can be derived: One can move from one solution to another by following the border of the solution space (polyhedra) and finally end at the optimum extreme point. This method to solve an LP, which is called *Simplex Method*, was invented by G.B. Dantzig in 1947 who later wrote a comprehensive book on the subject [Dan63]. Since then, it has been the standard technique for solving LPs.

Using two slack variables x_3 and x_4 to allow equality of constraints the original problem can be rewritten as:

$$x_1 + x_2 = f(x) \tag{4.3a}$$
$$x_1 + 2x_2 + x_3 = 7 \tag{4.3b}$$
$$2x_1 + x_2 + x_4 = 6 \tag{4.3c}$$
$$x_1, x_2, x_3, x_4 \geq 0 \tag{4.3d}$$

The objective value of $f(x) = x_1 + x_2$ increases with x_1 and x_2. Thus, we can start with a feasible solution $x = (x_1, x_2, x_3, x_4) = (0, 0, 7, 6)$ and an objective value of $f(x) = 0$ and increase x_1 while leaving $x_2 = 0$ for the moment. Equation (4.3b) allows to increase x_1 to 7 (decreasing x_3 to 0) and maintains feasibility. From Equation (4.3c), we can increase x_1 to 3 (decreasing x_4 to 0) and maintain feasibility. Since the second condition of x_1 is stricter, we can move to the feasible solution $x = (3, 0, 4, 0)$ and an objective value of $f(x) = x_1 + x_2 = 3 + 0 = 3$.

In the geometric representation we moved from one extreme point (0,0) along the x-axis to the next extreme point (3,0). When again rewriting the equations so that the objective function $f(x)$ does no longer include variable x_1 we will gain the following equation system:

$$0.5x_2 - 0.5x_4 = f(x) - 3 \quad (Eq.\ (4.3a) - 0.5 \cdot Eq.\ (4.3c)) \tag{4.4a}$$
$$1.5x_2 + x_3 - 0.5x_4 = 4 \quad (Eq.\ (4.3b) - 0.5 \cdot Eq.\ (4.3c)) \tag{4.4b}$$
$$1x_1 + 0.5x_2 + 0.5x_4 = 3 \quad (0.5 \cdot Eq.\ (4.3c)) \tag{4.4c}$$
$$x_1, x_2, x_3, x_4 \geq 0 \tag{4.4d}$$

When increasing x_2 the term $f(x) = 0.5x_2 - 0.5x_4 + 3$ will increase. From Equation (4.4b), x_2 can be increased to $\frac{8}{3}$, decreasing x_3 to 0. From Equation (4.4c) x_2 can be increased to 6, decreasing x_1 to 0. Since the first condition is stricter, we can move to the new feasible solution $x = (\frac{5}{3}, \frac{8}{3}, 0, 0)$. Adding multiples of Equation (4.4b) to the other equations to

[5] The complete mathematical proof can be found in [BT97, pp. 75].

eliminate x_3 we get:

$$-(1/3)\ x_3 - (2/3)\ x_4 = f(x) - (13/3) \quad (Eq.\ (4.4a) - (1/3) \cdot Eq.\ (4.4b)) \tag{4.5a}$$
$$x_2 + (2/3)\ x_3 - (1/3)\ x_4 = (8/3) \quad ((2/3) \cdot Eq.\ (4.4b)) \tag{4.5b}$$
$$x_1 - (1/3)\ x_3 - (1/3)\ x_4 = (5/3) \quad (Eq.\ (4.4c) - (1/3) \cdot Eq.\ (4.4b)) \tag{4.5c}$$
$$x_1, x_2, x_3, x_4 \geq 0 \tag{4.5d}$$

Since the objective function $f(x)$ is independent of x_1 and x_2 and the coefficients of x_3 and x_4 are all negative, the optimal solution of the problem is found ($f(x) = \frac{13}{3} = 4\frac{1}{3}$). In the geometric representation we moved from one extreme point (3,0) to the next extreme point ($\frac{5}{3}, \frac{8}{3}$).

General Simplex Algorithm:

Writing the problem as a simplex tableau, the general simplex algorithm can be formulated as follows:

General form of the simplex tableau

$(a_{i,j})$	(b_i)
c^T	f

Simplex tableau of the example

x_1	x_2	x_3	x_4	
1	2	1	0	7
2	1	0	1	6
1	1	0	0	0

1. Choose a pivot column:	Choose a j such that $c_j > 0$. Make $x_j > 0$ in this pivot step.
2. Choose a pivot row:	Among the i's with $a_{i,j} > 0$, choose i to minimize $b_i/a_{i,j}$.
3. Pivot on element $a_{i,j}$:	Do row operations so that column j ends up with a 1 in the pivot row and 0 elsewhere.
4. Stop if all coefficients c_j are non-positive, otherwise repeat at 1.	

The complete example consequently reads as:[6]

x_1	x_2	x_3	x_4	
1	2	1	0	7
2	1	0	1	6
1	1	0	0	0

x_1	x_2	x_3	x_4	
0	1.5	1	-0.5	4
1	0.5	0	0.5	3
0	0.5	0	-0.5	-3

x_1	x_2	x_3	x_4	
0	1	2/3	-1/3	8/3
1	0	-1/3	-1/3	5/3
0	0	-1/3	-2/3	-13/3

Although the number of extreme points of the feasible set can increase exponentially with the number of variables and constraints, it has been observed in practice that the simplex method typically takes only $O(m)$ pivots to find an optimal solution [BT97, pp.124].

[6]Note, the tableau provides and easy way to determine the current objective: $f(x) = -f$.

In 1979 L.G. Khachian introduced the *Ellipsoid Method* with polynomial running time [Kha79] providing a fundamental for a new class of LP solvers, so called *Interior Point Methods*: Instead of passing from corner to corner of the polyhedra, they pass through the interior of the feasible region. Although much research is currently ongoing to improve interior-point methods, the simplex algorithm still enjoys an unsurpassed popularity, because of its better average performance.

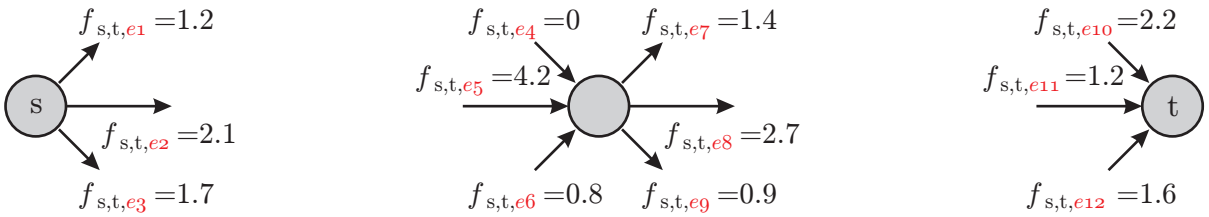
More details on solving approaches, finding initial solutions and the theory of linear and integer linear programming can be found in [BT97],[Dan63],[Wol98],[Chv83].

4.1.2.2 Basic Modeling Formulations

There exist two different formulations for the modeling of network optimization problems as LPs. In the following paragraphs, we will sketch both formulation approaches and discuss their strengths and weaknesses.

Flow-based Formulation:

The *flow-based formulation* approach is inspired by Kirchhoff's Current Law and is the 'classic' approach for network optimization problems. A flow variable $f_{s,t,e}$ models the traffic that is sourced by node s and targeted to node t on each edge e. A flow is inserted to the network at the source node and is removed at the destination node. Additionally, nodes in between have to transport the flow and are not allowed to add or remove traffic that belongs to that flow. In other words, the arithmetic sum of incoming traffic and outgoing traffic has to be equal at transient nodes.



(a) Source node of the demand.

(b) Intermediate node.

(c) Target node of the demand.

Figure 4.3: Node models of the flow approach.

The complete routing model using the flow-based formulation is given in Equations (4.6) to (4.11). Flow conservation is applied with Equations (4.7a) to (4.9), while capacity calculation and capacity restrictions are modeled with Equation (4.10) and (4.11), respectively.

Sets:

$\mathbb{N}$	Nodes of the physical network.
$\mathbb{E}$	Edges of the physical network ($\in \mathbb{N} \times \mathbb{N}$).
$\mathbb{D}$	Demand-relations between two physical nodes ($\in \mathbb{N} \times \mathbb{N}$).

Parameters:

$D_{s,t}$	$\in \mathbb{R}^+, (s,t) \in \mathbb{D}$	***D***emand between source node s and target node t.
Cost_e	$\in \mathbb{R}^+,\ e \in \mathbb{E}$	***Cost***s to use one capacity unit on edge e.
C_e	$\in \mathbb{R}^+,\ e \in \mathbb{E}$	Maximum usable ***c***apacity on edge e.

Variables:

$f_{s,t,e}$	$\in \mathbb{R}^+,\ (s,t) \in \mathbb{D}, e \in \mathbb{E}$	Amount of traffic (***f***low) for demand relation $s-t$ on edge e.
UCE_e	$\in \mathbb{R}^+,\ e \in \mathbb{E}$	***U***sed ***c***apacity on ***e***dge e for all demand relations.

Objective function:

$$\text{minimize} \sum_{e \in \mathbb{E}} (\text{Cost}_e \cdot UCE_e) \tag{4.6}$$

Constraints:

$\forall (s,t) \in \mathbb{D}, \forall n \in \mathbb{N}:$

$$\text{node } n \text{ is } s: \begin{cases} \sum_{e \in \text{incoming}(n)} f_{s,t,e} = 0 & (4.7a) \\ \sum_{e \in \text{outgoing}(n)} f_{s,t,e} = D_{s,t} & (4.7b) \end{cases}$$

$$\text{node } n \text{ is } t: \begin{cases} \sum_{e \in \text{incoming}(n)} f_{s,t,e} = D_{s,t} & (4.8a) \\ \sum_{e \in \text{outgoing}(n)} f_{s,t,e} = 0 & (4.8b) \end{cases}$$

$$\text{else:} \begin{cases} \sum_{e \in \text{incoming}(n)} f_{s,t,e} = \\ \sum_{e \in \text{outgoing}(n)} f_{s,t,e} & (4.9) \end{cases}$$

$$UCE_e = \sum_{(s,t)\in\mathbb{D}} f_{s,t,e} \qquad \forall e \in \mathbb{E} \tag{4.10}$$

$$UCE_e \leq \mathrm{C}_e \qquad \forall e \in \mathbb{E} \tag{4.11}$$

Path-based Formulation:

The *path-based formulation*[7] approach models the distribution of traffic on a set of pre-determined paths. Thus, instead of modeling flow conservation, paths are pre-computed and a variable $pc_{s,t,p}$ is associated to each path between two nodes s and t. Similar to the flow-based formulation an additional variable UCE_e is used to calculate the required capacity on an edge.

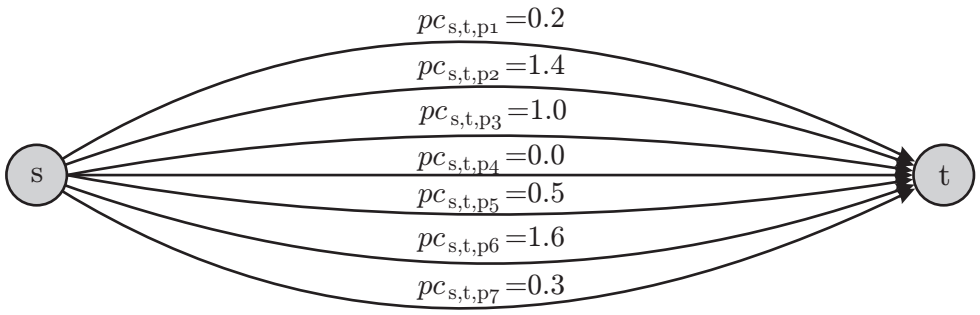


Figure 4.4: Illustration of the path approach. The demand is distributed on pre-determined paths.

The complete routing model using the path-based formulation is given in Equations (4.12) to (4.15).

Sets:

$\mathbb{N}$	Nodes of the physical network.
$\mathbb{E}$	Edges of the physical network ($\in \mathbb{N} \times \mathbb{N}$).
$\mathbb{D}$	Demand-relations between two physical nodes ($\in \mathbb{N} \times \mathbb{N}$).
$\mathbb{P}_{s,t}$	Paths between node s and node t.

Parameters:

$D_{s,t}$	$\in \mathbb{R}^+,\ (s,t) \in \mathbb{D}$	***D***emand between source node s and target node t.
Cost_e	$\in \mathbb{R}^+,\ e \in \mathbb{E}$	***Cost***s to use one capacity unit on edge e.
C_e	$\in \mathbb{R}^+,\ e \in \mathbb{E}$	Maximum usable ***c***apacity on edge e.

[7]This formulation approach is also called *link-based formulation* in the literature.

Variables:

$pc_{s,t,p}$	$\in \mathbb{R}^+,\ (s,t) \in \mathbb{D},$ $p \in \mathbb{P}_{s,t}$	Amount of traffic for demand relation $s-t$ on path p (***p***ath ***c***apacity).
UCE_e	$\in \mathbb{R}^+,\ e \in \mathbb{E}$	***U***sed ***c***apacity on ***e***dge e for all demand relations.

Objective function:

$$\text{minimize} \sum_{e \in \mathbb{E}} (\text{Cost}_e \cdot c_e) \tag{4.12}$$

Constraints:

$$\sum_{p \in \mathbb{P}_{s,t}} pc_{s,t,p} = D_{s,t} \qquad \forall (s,t) \in \mathbb{D} \tag{4.13}$$

$$UCE_e = \sum_{(s,t) \in \mathbb{D}} \sum_{\substack{p \in \mathbb{P}_{s,t} \\ e \in p}} pc_{s,t,e} \qquad \forall e \in \mathbb{E} \tag{4.14}$$

$$UCE_e \leq \mathrm{C}_e \qquad \forall e \in \mathbb{E} \tag{4.15}$$

Discussion of Flow- and Path-based Formulation:

Flow-based formulation models the traffic flow along links and through nodes of the network. Since the analogy to destination-based routing in communication networks is high, the modeling of routing problems is facilitated. However, one flow-conservation equation is required for each node and demand relation ($O(N \cdot D)$). Thus, a large number of interrelated equations have to be handled. Computation is complex for larger networks since the number of iterations for the simplex method is (usually) proportional to the number of generated constraints.

Path-based formulation, in contrast to that, requires the pre-calculation of paths. One variable per path and one equation per demand-relation is sufficient for the modeling of routing ($O(D)$). However, although the number of equations is smaller compared to *flow-based formulations*, an immense number of paths exist in larger networks. Thus, the number of variables is large when using *path-based formulations*. Nevertheless, although the optimality can no longer be guaranteed, good solutions can be calculated in reasonable time and with reasonable computer memory requirements, if only a sub-set of paths are taken into consideration.

In summary, both approaches have their strengths and weaknesses. Dependent on LP formulation and the exact problem, *path-* or *flow-based formulations* perform better in terms of running time or computer memory consumption. If optimality is not an issue, restricted path-based formulations may be advantageous. However, one has to be careful to calculate

and use the 'right' paths for the optimization. Considering the constant improvement of new approaches in LP solving (e.g. interior-point methods) in the last years, however, also flow-based formulations might be suitable for large networks in a few years time.

4.2 Resilient Network Optimization with Integer Linear Programming

In general, different resilience mechanisms require different Integer Linear Programming (ILP) formulations. However, the resilience classification framework of Chapter 3 illustrated that resilience mechanisms are miscellaneous combinations of characteristics and attributes. Thus, different resilience mechanisms may share more similarities than differences. Therefore, instead of providing complete ILP models for each resilience mechanism, we provide separate building blocks that we will combine later on. This approach significantly reduces the complexity, modeling, and programming effort when building integrated planning tools.

As seen in Section 4.1.2, there are two main choices for integer linear programming formulations: Flow Approach and Path Approach. Initially, it is not known which formulation has advantages concerning solvability and running time. Thus, in this section we present formulations for both approaches for an optimized resilient network design. Furthermore, we present a new approach that enhances the planning of resilient networks considerably by using a mathematical technique called *Column Generation*. For the ease of reading the same sets, variable- and parameter names are used in all linear programming formulations.

4.2.1 Common Models

4.2.1.1 Sets, Variables, and Parameters

Tables 4.1 and 4.2 summarize the most important sets, variables, and parameters. The variable names are chosen in a way to illustrate the meaning of the variable. In the following, all variables will be non-negative (≥ 0). The type of the variable, continuous (D), integer (I), or Boolean (B) is indicated as upper-script index. The complete summary of all used sets, variables, and parameters can be found in Appendix C.

Table 4.1: Common sets.

Symbol	Description
$\mathbb{N}$	Nodes of the physical network.
$\mathbb{E}$	Edges of the physical network ($\in \mathbb{N} \times \mathbb{N}$).
$\mathbb{S}$	Status of the network during different failure patterns. Including the failure-free state s_0.
$\mathbb{F}$	Failure patterns i.e. (failing edges or nodes).
$\mathbb{D}$	Demand-relations between two physical nodes ($\in \mathbb{N} \times \mathbb{N}$).

Table 4.2: Common variables or parameters.

Symbol	Type	Index	Description
D_d^D	real	$d \in \mathbb{D}$	The traffic value of a ***d***emand d.
WCE_e^D	real	$e \in \mathbb{E}$	The (maximum) required ***w***orking ***c***apacity on ***e***dge e.
UCE_e^D	real	$e \in \mathbb{E}$	The (maximum) ***u***sed ***c***apacity on ***e***dge e.
$UCES_{e,s}^D$	real	$e \in \mathbb{E}$, $s \in \mathbb{S}$	The ***u***sed ***c***apacity on ***e***dge e in network ***s***tate s.
$SRCE_e^D$	real	$e \in \mathbb{E}$	The ***s***hared ***r***esilience (backup) ***c***apacity on an ***e***dge e in network ***s***tate s.
$SRCES_{e,s}^D$	real	$e \in \mathbb{E}$, $s \in \mathbb{S}$	The required ***s***hared ***r***esilience ***c***apacity on ***e***dge e during a specific ***f***ailure pattern (network state s).
$DRCE_e^D$	real	$e \in \mathbb{E}$	The maximum ***d***edicated backup (***r***esilience) ***c***apacity on ***e***dge e.
$DRCES_{e,s}^D$	real	$e \in \mathbb{E}$, $s \in \mathbb{S}$	The used ***d***edicated ***r***esilience ***c***apacity on ***e***dge e during a specific failure pattern (network ***s***tate) s.
Max^D	real		A large positive number.

4.2.1.2 Objective Function

Section 2.1.1 discussed different kinds of objectives for resilient network planning. In the following, however, we will focus on the minimization of network design costs, i.e. the capital expenditures (CAPEX) to build the network.

$$\text{minimize CAPEX} \tag{4.16}$$

Minimization of Edge Capacity: In particular, we will focus on the minimization of required edge capacity. Equation (4.17) models the sum of used capacity on edges of the network.

$$CAPEX = \sum_{e \in \mathbb{E}} UCE_e^D \tag{4.17}$$

4.2.1.3 Hardware Configuration

Non-continuous capacity costs, e.g. due to modular hardware configurations, can easily be modeled using additional Boolean or integer variables. As an example, Equations (4.18) to (4.21) formulate the use of two possible line-card hardware designs (HWD). The integer variables $HWD10G_e^I$ and $HWD100G_e^I$ are used to count the required hardware designs on

an edge e. Design $HWD10G$ is for example able to transport 10Gbit/s while $HWD100G$ is able to transport 100Gbit/s.

$$10 \cdot HWD10G_e^I + 100 \cdot HWD100G_e^I \geq UCE_e^D \qquad \forall e \in \mathbb{E} \tag{4.18}$$

Equation (4.18) assures that an appropriate amount of designs are chosen for the required capacity (UCE_e^D in Gbit/s). Following this, dependent on the cost of the used equipment the network design cost can be modeled according to Equation (4.19).

$$CAPEX = \sum_{e \in \mathbb{E}} (COST10G \cdot HWD10G_e^I + COST100G \cdot HWD100G_e^I) \tag{4.19}$$

Additionally, maximum hardware configurations can be modeled using equations similar to (4.20) and (4.21). In this example, it is possible to install a maximum of five line-cards while only one line-card of type HWD100G is allowed per node.

$$\sum_{e \in \text{outgoing}(n)} HWD100G_e^I \leq 1 \qquad \forall n \in \mathbb{N} \tag{4.20}$$

$$\sum_{e \in \text{outgoing}(n)} (HWD10G_e^I + HWD100G_e^I) \leq 5 \qquad \forall e \in \mathbb{E} \tag{4.21}$$

4.2.2 Flow-based Formulations

Maximum flow problems have been formulated since the development of the Simplex algorithm (e.g. [Dan63], [Ber98, Gir94] and were applied to protection mechanisms in e.g. [SS99], [BM96]. However, to our knowledge the following formulations are the first that restrict multipath routing and provide formulations for SL2EPP, SLLPP, and SRP [**Gru05**].

4.2.2.1 Sets, Variables, and Parameters

Section 4.1.2 outlined the basic idea of formulating routing problems as Integer Linear Program using the flow approach. One variable per demand and edge, and one constraint per demand and transit node are required to provide flow conservation. Furthermore, traffic of one demand can be split and routed along different paths. Thus, multipath routing is already provided by the formulation. Traffic of one demand can be separated and combined at each intermediate node. However, the number of paths that can be used to route a demand is not restricted and an unlimited number of paths is possible. Thus, the effects of path-split restrictions cannot be investigated with this formulation. Additionally, paths of one demand cannot be separated from each other and the formulation of a working capacity reuse of failure-affected paths is complicated. We therefore change the classical flow-approach formulation and add an additional path index $i \in [1..\text{MaxWSplit}^I]$ to a flow variable. With this index, we can separate paths from each other and are able to limit the number of paths a demand is split into. Additionally, we add an index $j \in [1..\text{MaxRSplit}^I]$

to backup flow variables in order to model splits of working path into different backup paths during a failure. An illustration of these splits is depicted in Figure 4.5. Figure 4.5(a) shows an example of a demand that is split into three working paths using index $i \in \{1, 2, 3\}$. Figure 4.5(b) illustrates an example in which one working path ($i = 2$) is furthermore split into three resilience paths using index $j \in \{1, 2, 3\}$.

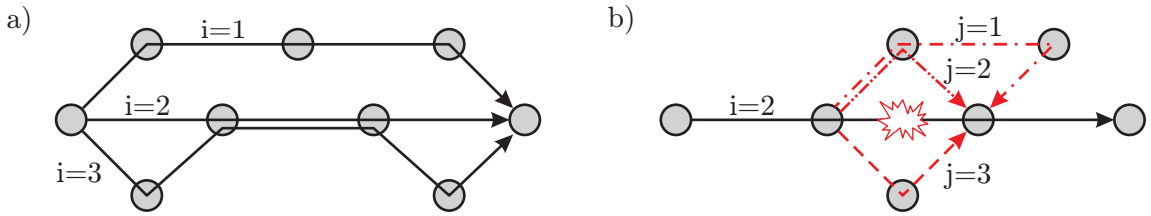


Figure 4.5: Example of a split of a demand into three working paths (a) and a split of one working path into three different resilience paths (b).

In the following we will use variable $WPE^D_{d,i,e}$ to represent the used bitrate of path i of a demand d on an edge e in a failure free network. We will furthermore define different network states $s \in \mathbb{S}$. Each failure pattern corresponds to exactly one network state s. Consequently, variable $RPWPES^D_{d,i,e,j,s}$ will represent the used bitrate of backup path j on edge e that protects working path i of demand d in network state s.

Table 4.3: Additional sets used in the flow approach formulation.

Symbol	Description
$\mathbb{I}_{\text{WSplit}}$	Multipath indices of a demand that is split into different working paths ($i \in [1..\text{MaxWSplit}^I$).
$\mathbb{I}_{\text{RSplit}}$	Multipath indices of a working path that is split into different backup paths ($j \in [1..\text{MaxRSplit}^I]$)).

Tables 4.3 and 4.4 summarize the additionally required sets and the most important variables and parameters for the flow-based formulation. A complete description of all sets, variables, and parameters can be found in Appendix C.

4.2.2.2 Constraint Building Blocks

Constraints of the ILP model using the flow-based formulation can be separated into building blocks. A combination of these building blocks to form complete resilient network optimization formulations will be presented in Section 4.2.2.3.

Failure Free Flow Conservation:
Traffic is generated at the source and terminated at the target node of a demand. Traffic should neither be lost nor added along the path. Therefore, the flow-based formulation

Table 4.4: Additional important variables and parameters used in the flow approach formulation.

Symbol	Type	Index	Description
$WPE^D_{d,i,e}$	real	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $e \in \mathbb{E}$	***W***orking traffic ***p***art i of demand d on physical ***e***dge e.
$WPE^B_{d,i,e}$	bool	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $e \in \mathbb{E}$	Flow indicator that will be one if $WPE^D_{d,i,e} > 0$ and zero otherwise.
$RPWPES^D_{d,i,e,j,s}$	real	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $e \in \mathbb{E}$, $j \in \mathbb{I}_{\text{RSplit}}$, $s \in \mathbb{S}$	Backup (***r***esilience) traffic ***p***art j protecting ***w***orking part i of demand d on physical ***e***dge e in network ***s***tate s.
$RPWPES^B_{d,i,e,j,s}$	bool	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $e \in \mathbb{E}$, $j \in \mathbb{I}_{\text{RSplit}}$, $s \in \mathbb{S}$	Flow indicator that will be one if $RPWPES^D_{d,i,e,j,f} > 0$.
$WPI^I_{d,i,e}$	int	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $e \in \mathbb{E}$	***I***ndex along the ***w***orking path ***p***art i of demand d. Increases by one on each edge.
$WPILS^I_{d,i,s}$	int	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $s \in \mathbb{S}$	***I***ndex of the first failing edge (***l***eft) along ***w***orking path ***p***art i of demand d in network ***s***tate s.
$Detour^B_{d,i,e,s}$	bool	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $e \in \mathbb{E}$, $s \in \mathbb{S}$	Indicator if the backup ***detour*** is in front of edge e along working path i of demand d for failure pattern f. Forced to be zero if the detour is in front.

models flow conservation of transit traffic at intermediate nodes.

$$IWPN^D_{d,i,n} = \sum_{e \in \text{incoming}(n)} WPE^D_{d,i,e} \qquad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall n \in \mathbb{N} \tag{4.22}$$

$$OWPN^D_{d,i,n} = \sum_{e \in \text{outgoing}(n)} WPE^D_{d,i,e} \qquad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall n \in \mathbb{N} \tag{4.23}$$

$$IWN^D_{d,n} = \sum_{i \in \mathbb{I}_{\text{WSplit}}} IWPN^D_{d,i,n} \qquad \forall d \in \mathbb{D}, \forall n \in \mathbb{N} \tag{4.24}$$

$$OWN^D_{d,n} = \sum_{i \in \mathbb{I}_{\text{WSplit}}} OWPN^D_{d,i,n} \qquad \forall d \in \mathbb{D}, \forall n \in \mathbb{N} \tag{4.25}$$

The incoming and outgoing traffic of working path i of demand d at a node n can be accumulated to $IWPN^D_{d,i,n}$ and $OWPN^D_{d,i,n}$, respectively (Equations (4.22) and (4.23)). Consequently, all incoming and outgoing working paths i of a demand d at a node n can be summed up to $IWN^D_{d,n}$ and $OWN^D_{d,n}$ (Equations (4.24) and (4.25)). Finally, the flow formulation can be modeled according to Equations (4.26a) to (4.28).

$\forall d \in \mathbb{D}, \forall n \in \mathbb{N}:$

$$n \text{ is source of } d\text{:} \begin{cases} IWN^D_{d,n} = 0 & (4.26a) \\ OWN^D_{d,n} = D^D_d & (4.26b) \end{cases}$$

$$n \text{ is target of } d\text{:} \begin{cases} IWN^D_{d,n} = D^D_d & (4.27a) \\ OWN^D_{d,n} = 0 & (4.27b) \end{cases}$$

$$\text{else: } \forall i \in \mathbb{I}_{\text{WSplit}} \begin{cases} IWPN^D_{d,i,n} = \\ OWPN^D_{d,i,n} & (4.28) \end{cases}$$

Equations (4.26a) and (4.26b) will be applicable if node n is the source of demand d. Traffic is generated at this node and thus the outgoing traffic has to be at least the demand value while the incoming traffic of the source node is zero. Similarly, the incoming traffic is at least the demand value at the destination node and no traffic is generated at this node for the demand (Equations (4.27a) and (4.27b)). Nodes in-between do not generate or terminate traffic for this demand. Thus, traffic has to be routed through a node and incoming traffic equals outgoing traffic (Equation (4.28)).

Note, that Equations (4.26a) to (4.27b) allow a multipath routing, i.e. a splitting of the demand into several working paths with different indices i. Equation (4.28) formulates flow conservation by applying one equation for each index i. However, as illustrated in Figure 4.6 these equations still allow that traffic is split into several paths with the same working path index i. To prevent this, the number of positive outgoing and incoming flow variables with the same number i have to be restricted.

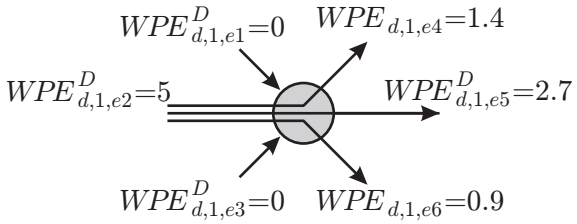


Figure 4.6: Undesired effect of splits of traffic of one working path (index i) to different outgoing edges at an intermediate node - not prevented by Equation (4.28) alone ($IWPN^D_{d,1,n} = OWPN^D_{d,1,n} = 5$).

In order to count outgoing and incoming flows of one traffic part i of a demand d a Boolean variable $WPE^B_{d,i,e}$ is required to indicate whether a corresponding flow variable $WPE^D_{d,i,e}$ is used (> 0). If an edge e transports traffic for working path i of demand d the flow variable $WPE^D_{d,i,e}$ will be greater than zero. Thus, the Boolean variable $WPE^B_{d,i,e}$ is forced to be one by Equation (4.29). Otherwise, if the edge is not used Equation (4.30) will require that the Boolean variable has to be zero. The possible values resulting from

the combination of both equations are shown in Table 4.5. Note that the parameter Max^D must be greater than any possible demand value in order to not prohibit any feasible solution.

$$WPE^B_{d,i,e} \cdot \text{Max}^D \geq WPE^D_{d,i,e} \qquad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall e \in \mathbb{E} \tag{4.29}$$

$$(WPE^B_{d,i,e} - 1) \cdot \text{Max}^D < WPE^D_{d,i,e} \qquad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall e \in \mathbb{E} \tag{4.30}$$

Table 4.5: Possible values of $WPE^B_{d,i,e}$ according to Equations (4.29) and (4.30).

Variable $WPE^D_{d,i,e}$	Variable $WPE^B_{d,i,e}$		
	Equation (4.29)	Equation (4.30)	both
$= 0$	0 or 1	0	0
> 0	1	0 or 1	1

The number of outgoing splits for part i of demand d on a physical node n can then be calculated with Equation (4.31) and forced to be at maximum one with Equation (4.32).

$$OWPN^I_{d,i,n} = \sum_{e \in \text{outgoing}(n)} WPE^B_{d,i,e} \qquad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall n \in \mathbb{N} \tag{4.31}$$

$$OWPN^I_{d,i,n} \leq 1 \qquad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall n \in \mathbb{N} \tag{4.32}$$

Since only one path variable for each index i is generated by the source of the demand, the incoming splits of one part are restricted implicitly and need not to be restricted explicitly.

Used Working Capacity on Edges:
The required capacity on an edge to transport traffic in a failure free network can easily be obtained by using Equation (4.33). Variable $WPE^D_{d,i,e}$ already models the required working capacity for part i of a demand d on an edge e. As working capacity cannot be shared, the total required working capacity of an edge e is the sum of all demand parts traversing the edge.

$$WCE^D_e = \sum_{d \in \mathbb{D}} \sum_{i \in \mathbb{I}_{\text{WSplit}}} WPE^D_{d,i,e} \qquad \forall e \in \mathbb{E} \tag{4.33}$$

Basic Resilience:
The detour of traffic around a failing element is common to all considered resilience mechanisms. The backup paths should obviously not traverse failing elements. Therefore, Equation (4.34) will force the flow variable of any backup traffic to zero if the edge or its adjacent nodes fail ($e \in \mathbb{F} \vee \text{source(e)} \in \mathbb{F} \vee \text{target(e)} \in \mathbb{F}$).

$$RPWPES^D_{d,i,e,j,s} = 0 \qquad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall e \in \mathbb{E}, \forall j \in \mathbb{I}_{\text{RSplit}}, \forall f \in \mathbb{F},$$
$$(e \in \mathbb{F}) \vee (\text{source(e)} \in \mathbb{F}) \vee (\text{target(e)} \in \mathbb{F}) \tag{4.34}$$

Similar to $IWPN^D_{d,i,n}$, $OWPN^D_{d,i,n}$, $IWN^D_{d,n}$, and $OWN^D_{d,n}$, the incoming and outgoing detoured traffic at a node can be modeled separately or as a sum of all parts j using Equations (4.35) to (4.38).

$$IRPWPNS^D_{d,i,n,j,s} = \sum_{e \in \text{incoming}(n)} RPWPES^D_{d,i,e,j,s}$$
$$\forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall n \in \mathbb{N}, \forall j \in \mathbb{I}_{\text{RSplit}}, \forall f \in \mathbb{F} \tag{4.35}$$

$$ORPWPNS^D_{d,i,n,j,s} = \sum_{e \in \text{outgoing}(n)} RPWPES^D_{d,i,e,j,s}$$
$$\forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall n \in \mathbb{N}, \forall j \in \mathbb{I}_{\text{RSplit}}, \forall f \in \mathbb{F} \tag{4.36}$$

$$IRWPNS^D_{d,i,n,s} = \sum_{j \in \mathbb{I}_{\text{RSplit}}} IRPWPNS^D_{d,i,n,j,s}$$
$$\forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall n \in \mathbb{N}, \forall f \in \mathbb{F} \tag{4.37}$$

$$ORWPNS^D_{d,i,n,s} = \sum_{j \in \mathbb{I}_{\text{RSplit}}} ORPWPNS^D_{d,i,n,j,s}$$
$$\forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall n \in \mathbb{N}, \forall f \in \mathbb{F} \tag{4.38}$$

To prevent the splitting of paths with the same index j at an intermediate node a Boolean variable $RPWPES^B_{d,i,e,j,s}$ is required. Using Equation (4.39) its value will be true if the backup flow variable $RPWPES^D_{d,i,e,j,s}$ is used (> 0). Table 4.5 shows the possible values of the Boolean variable $RPWPES^B_{d,i,e,j,s}$ dependent on the value of the flow variable $RPWPES^D_{d,i,e,j,s}$.

$$RPWPES^B_{d,i,e,j,s} \geq \frac{RPWPES^D_{d,i,e,j,s}}{\text{Max}^D}$$
$$\forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall n \in \mathbb{N}, \forall j \in \mathbb{I}_{\text{RSplit}}, \forall f \in \mathbb{F} \tag{4.39}$$

Table 4.6: Possible values of $RPWPES^B_{d,i,e,j,s}$ according to Equation (4.39).

Variable $RPWPES^D_{d,i,e,j,s}$	Variable $RPWPES^B_{d,i,e,j,s}$
$= 0$	0 or 1
> 0	1

Following this, the number of outgoing backup paths j at a node n during a failure f can be counted with Equation (4.40) and limited using Equation (4.41).

$$ORPWPNS^{I}_{d,i,n,j,s} = \sum_{e \in \text{outgoing}(n)} RPWPES^{B}_{d,i,e,j,s}$$
$$\forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall n \in \mathbb{N}, \forall j \in \mathbb{I}_{\text{RSplit}}, \forall f \in \mathbb{F} \quad (4.40)$$
$$ORPWPNS^{I}_{d,i,n,j,s} \leq 1$$
$$\forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall n \in \mathbb{N}, \forall j \in \mathbb{I}_{\text{RSplit}}, \forall f \in \mathbb{F} \quad (4.41)$$

Enumeration of Working Paths:
The location of an edge relative to failure(s) along the working path is important for calculating the required resilience capacity. Depending on the failure location, reserved working capacity will be able to be reused for resilience purposes if this is supported by the technology (e.g. MPLS). Figure 4.7 shows an example of a local-to-egress backup path, in which no additional capacity is required on edge (E-F) for the backup path. The working traffic is detoured in front of the failure and the capacity reserved for the working path can be reused to transport the detoured traffic. However, in front of the failure the capacity on edges, which is reserved for the working path, cannot be reused and additional capacity is required on edge B-C for the backup path. Note, that the overall required capacity might be reduced further if another resilience mechanism would be used that is able to detour the working traffic already at node C (e.g. combined with a signaling of a failure message to C).

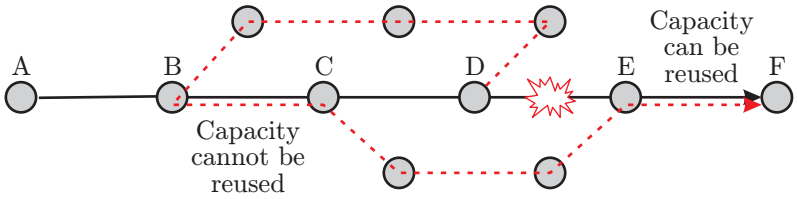


Figure 4.7: Example of a local-to-egress backup path. Working capacity can be reused on edge E-F for the backup path.

Since capacity can only be reused on edges that are located behind the detour along the working path, the edges used by the path have to be enumerated to distinguish their location relative to the failure(s). Therefore, variable $WPI^{I}_{d,i,e}$ represents a number that models the index for each working path i of a demand d on an edge e. Equations (4.42) and (4.43) ensure that this number increases by one along the working path ($WPE^{B}_{d,i,e} = 1$). If an edge is not used to transport traffic for the demand ($WPE^{B}_{d,i,e} = 0$) no stringent

restrictions will be applied to the index variables by these equations.

$$WPI^I_{d,i,e} \geq WPI^I_{d,i,e_2} + 1 - (1 - WPE^B_{d,i,e}) \cdot \text{Max}^D \qquad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \\ \forall e \in \mathbb{E}, \forall e_2 \in \mathbb{E}, e \neq e_2, \; e_2 \text{ targets the source of } e. \tag{4.42}$$

$$WPI^I_{d,i,e} \leq WPI^I_{d,i,e_2} + 1 + (1 - WPE^B_{d,i,e}) \cdot \text{Max}^D \qquad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \\ \forall e \in \mathbb{E}, \forall e_2 \in \mathbb{E}, e \neq e_2, \; e_2 \text{ targets the source of } e. \tag{4.43}$$

Calculation of the Lowest Failure Index:
Additionally, the index of the first failing element along the path (Figure 4.8) has to be determined for every failure pattern f. Let variable $WPILS^I_{d,i,s}$ model this index number and let $\mathbb{E}_f$ be the set of all edges that are either directly affected or are in front of the failing edge or node ($e \in f \vee target(e) \in f$).

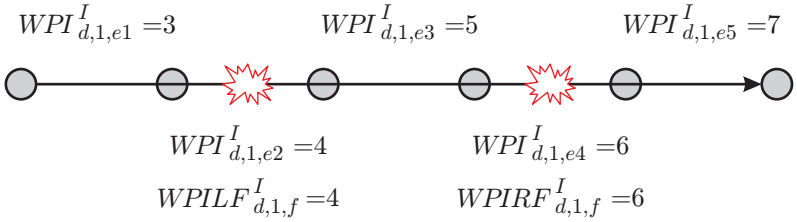


Figure 4.8: Example of an enumeration of the working path and the corresponding failure index of one failure pattern.

Equation (4.44) provides an upper bound for variable $WPILS^I_{d,i,s}$: If the failure affected edge $e \in \mathbb{E}_f$ is used by a working path i of demand d term $(1 - WPE^B_{d,i,e})$ will be zero. Consequently, the value of $WPILS^I_{d,i,s}$ has to be smaller or equal to the index of any failure-affected edge along the path. Additionally, a lower bound to $WPILS^I_{d,i,s}$ will be set by Equation (4.48) in combination with Equations (4.45) to (4.47): If at least one edge on the path failed $WPILS2^B_{d,i,f}$ would have to be one (Equation (4.45)). Furthermore, following Equation (4.47), exactly one of the variables $WPILS3^B_{d,i,e,f}$ along the working paths (Equation (4.46)) has to be one. $WPILS^I_{d,i,s}$ consequently has to be greater or equal to the index number of exactly one failing edge along the working path (Equation (4.48)). Thus, $WPILS^I_{d,i,s}$ has to be equal to the smallest failure affected edge along the path to meet all constraints of the five equations.

$$WPILS^I_{d,i,s} \leq WPI^I_{d,i,e} + (1 - WPE^B_{d,i,e}) \cdot \text{Max}^D \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall f \in \mathbb{F}, \forall e \in \mathbb{E}_f \tag{4.44}$$

$$WPILS2^B_{d,i,f} \geq WPE^B_{d,i,e} \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall f \in \mathbb{F}, \forall e \in \mathbb{E}_f \tag{4.45}$$

$$WPILS3^B_{d,i,e,f} \leq WPE^B_{d,i,e} \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall f \in \mathbb{F}, \forall e \in \mathbb{E}_f \tag{4.46}$$

$$\sum_{e \in \mathbb{E}_f} WPILS3^B_{d,i,e,f} = WPILS2^B_{d,i,f} \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall f \in \mathbb{F} \tag{4.47}$$

$$WPILS^I_{d,i,s} \geq WPI^I_{d,i,e} - (1 - WPILS3^B_{d,i,e,f}) \cdot \text{Max}^D - (1 - WPE^B_{d,i,e}) \cdot \text{Max}^D \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall f \in \mathbb{F}, \forall e \in \mathbb{E}_f \tag{4.48}$$

However, if a working path i of demand d is not affected by failure pattern f the term $(1 - WPE^B_{d,i,e})$, $e \in \mathbb{E}_f$, will be 1. Thus, Equation (4.44) provides no stringent upper bound for $WPILS^I_{d,i,s}$. Also, $WPILS2^B_{d,i,f}$ can be zero according to Equation (4.45) since all variables $WPE^B_{d,i,e}$, $e \in \mathbb{E}_f$ are zero. Thus, no variable $WPILS3^B_{d,i,e,f}$ has to be one (Equation (4.47)) and Equation (4.48) provides no stringent lower bound to variable $WPILS^I_{d,i,s}$.

Calculation of the Highest Failure Index:
The index of the last failure can be calculated similarly to the calculation of the index of the first failure along the path using Equations (4.49) to (4.52). Equation (4.49) provides a lower bound for $WPIRS^I_{d,i,s}$ restricting it to be greater or equal to the indices of all failing elements along the path ($WPE^B_{d,i,e} = 1$). If the path fails in failure pattern f, $WPIRS2^B_{d,i,e,f}$ will be 1 (Equation (4.50)) and exactly one of the variables $WPIRS3^B_{d,i,e,f}$ of edges along the path (Equation (4.51)) has to be 1 (Equation (4.52)). Following this, the value of $WPIRS^I_{d,i,s}$ is limited from above to be smaller than exactly one failure affected edge along the path. In order to meet all constraints of the five equations, $WPIRS^I_{d,i,s}$ has to be equal to the index of the last failure affected edge along the path.

$$WPIRS^I_{d,i,s} \geq WPI^I_{d,i,e} - (1 - WPE^B_{d,i,e}) \cdot \text{Max}^D \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall f \in \mathbb{F}, \forall e \in \mathbb{E}_f \tag{4.49}$$

$$WPIRS2^B_{d,i,f} \geq WPE^B_{d,i,e} \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall f \in \mathbb{F}, \forall e \in \mathbb{E}_f \tag{4.50}$$

$$WPIRS3^B_{d,i,e,f} \leq WPE^B_{d,i,e} \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall f \in \mathbb{F}, \forall e \in \mathbb{E}_f \tag{4.51}$$

$$\sum_{e \in \mathbb{E}_f} WPIRS3^B_{d,i,e,f} = WPIRS2^B_{d,i,f} \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall f \in \mathbb{F} \tag{4.52}$$

$$WPIRS^I_{d,i,s} \leq WPI^I_{d,i,e} + (1 - WPIRS3^B_{d,i,e,f}) \cdot \text{Max}^D + (1 - WPE^B_{d,i,e}) \cdot \text{Max}^D \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall f \in \mathbb{F}, \forall e \in \mathbb{E}_f \tag{4.53}$$

If a working path is not affected by a failure, no stringent lower and upper bounds will be imposed to variable $WPIRS^I_{d,i,s}$ by Equations (4.49) to (4.53) as described in the previous section.

Used Resilience Capacity On Edges:
As seen in building block *Enumeration of Working Paths*, working capacity might be used for resilience purposes if a detour was performed in front of a considered edge along the working path. The required resilience capacity on an edge e for a working path i in failure pattern f ($RCEWPS^D_{d,i,e,s}$) can thus be calculated as the required detour capacities ($RPWPES^D_{d,i,e,j,s}$) diminished by reusable working capacity (Equation (4.54)).

$$RCEWPS^D_{d,i,e,s} = \begin{cases} \sum_{\mathbb{I}_{\text{RSplit}}} RPWPES^D_{d,i,e,j,s} - WPE^D_{d,i,e} & \text{if } WPI^I_{d,i,e} > WPILS^I_{d,i,s} \\ \sum_{\mathbb{I}_{\text{RSplit}}} RPWPES^D_{d,i,e,j,s} & \text{if } WPI^I_{d,i,e} \leq WPILS^I_{d,i,s} \end{cases}$$
$$d \in \mathbb{D}, \mathbb{I}_{\text{WSplit}}, \forall e \in \mathbb{E}, f \in \mathbb{F} \quad (4.54)$$

In this notation, Equation (4.54) is non-continuous and has to be made continuous using a combination of several equations. Additionally, capacity can also be reused from other working paths of this demand or even of other demands if they are also detoured in front of the failure.

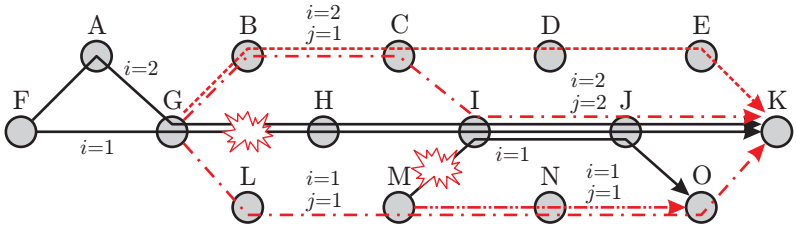


Figure 4.9: Example of working capacity reuse of different working paths for resilience purposes.

Figure 4.9 shows an example of possibilities of working capacity reuse with local-to-egress path protection. Demand *M-O* is routed along one working path $i = 1$ using edges *M-I-J-O* while demand *F-K* is split into two working paths $i = 1$: *F-G-H-I-J-K* and $i = 2$: *F-A-G-H-I-J-K*. In case of failure pattern f, consisting of the edges *G-H* and *M-I*, the dotted resilience paths are used. Working path $i = 1$ of demand *F-K* is detoured along path *G-L-M-N-O-K* while working path $i = 2$ is split into two resilience paths $j = 1$: *G-B-C-D-E-K* and $j = 2$: *G-B-C-I-J-K*. Additionally, working path *M-I-J-O* is detoured along resilience path *M-N-O*. Since failing edge *G-H* is in front of the edges along both working paths of demand *F-K* the working capacity of both working paths can be reused for resilience on edge *I-J* and *J-K*. Additionally, the working capacity of demand *M-O* can also be reused on edge *I-J* since the failure *M-I* is in front of edge *I-J* along the working path. Thus, calculating the required capacity on an edge is even more complicated than shown in Equation (4.54).

We can identify three types of working capacities that can be reused for resilience on an edge.

1. Working capacity of the protected working path, if the detour is in front of the considered edge along the path.

2. Working capacity of another protected working path of the demand, if the path is affected and the detour is in front of the considered edge along the path.

3. Working capacity of another demand if the working path is affected and the detour is in front of the considered edge along the path.

The Boolean variable $Detour^B_{d,i,e,s}$ is used to indicate whether a detour is in front or behind an edge. This variable will be restricted to be zero if the edge e is in front of the detour along the working path i of demand d of failure f. When using local protection mechanisms the detour starts in front of the first failure. Thus, Equation (4.55) will force the variable to be zero if it is located in front of the first failure along the path ($WPI^I_{d,i,e} < WPILS^I_{d,i,s}$). Table 4.7 summarizes the possible values of this variable dependent on $WPI^I_{d,i,e}$ and $WPILS^I_{d,i,s}$.

For local resilience:

$$WPI^I_{d,i,e} - WPILS^I_{d,i,s} \geq (Detour^B_{d,i,e,s} - 1) \cdot \mathrm{Max}^D \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\mathrm{WSplit}}, \forall e \in \mathbb{E}, \forall f \in \mathbb{F} \tag{4.55}$$

Table 4.7: Possible values of $Detour^B_{d,i,e,s}$ according to Equation (4.55).

Variable $WPI^I_{d,i,e}$ and $WPILS^I_{d,i,s}$	Variable $Detour^B_{d,i,e,s}$
$WPI^I_{d,i,e} \geq WPILS^I_{d,i,s}$	0 or 1
$WPI^I_{d,i,e} < WPILS^I_{d,i,s}$	0

The detour point of end-to-end path protection mechanisms is the source of the demand itself. Thus, every edge is located behind the failure and no restrictions have to be applied to $Detour^B_{d,i,e,s}$. In regional resilience the detour is located at least $\mathrm{MinFront}^I$ hops in front of the first failure. Since further variables are required to calculate the detour position for regional resilience Equation (4.55) will be modified later in this chapter.

The potential capacity reduction can be restricted according to Equations (4.56) to (4.58) with this Boolean variable. Let $UWCPES^D_{d,i,e,s}$ denote the reusable working capacity of demand d and path i on edge e during failure pattern f. Its value will be zero if the path is not affected by the failure (Equation (4.58)), zero if the detour is located behind the considered edge e along the working path (Equation (4.56)), and maximal the

value that is reserved for transporting the working traffic on this edge ($WPE^D_{d,i,e}$) otherwise (Equation (4.57)).

$$UWCPES^D_{d,i,e,s} \leq Detour^B_{d,i,e,s} \cdot \mathrm{Max}^D \qquad \forall d \in \mathbb{D}, i \in \mathbb{I}_{\mathrm{WSplit}}, \forall e \in \mathbb{E}, f \in \mathbb{F} \tag{4.56}$$

$$UWCPES^D_{d,i,e,s} \leq WPE^B_{d,i,e2} \cdot \mathrm{Max}^D \qquad \forall d \in \mathbb{D}, i \in \mathbb{I}_{\mathrm{WSplit}}, \forall e \in \mathbb{E}, f \in \mathbb{F}, e2 \in \mathbb{E}_f \tag{4.57}$$

$$UWCPES^D_{d,i,e,s} \leq WPE^D_{d,i,e} \qquad \forall d \in \mathbb{D}, i \in \mathbb{I}_{\mathrm{WSplit}}, \forall e \in \mathbb{E}, f \in \mathbb{F} \tag{4.58}$$

The sum of reusable working capacity on an edge e during failure pattern f ($UWCES^D_{e,s}$) can then be calculated according to Equation (4.59)

$$UWCES^D_{e,s} = \sum_{d \in \mathbb{D}} \sum_{i \in \mathbb{I}_{\mathrm{WSplit}}} UWCPES^D_{d,i,e,s} \qquad \forall e \in \mathbb{E} \forall f \in \mathbb{F} \tag{4.59}$$

However, if working capacity should never be reused Equation (4.60) would have to be added in place of Equations (4.56) to (4.58).

$$UWCPES^D_{d,i,e,s} \leq 0 \qquad \forall e \in \mathbb{E}, f \in \mathbb{F} \tag{4.60}$$

The remaining capacity that has to be reserved on an edge e for a working path i and failure pattern f can then be calculated using Equations (4.61) and (4.62). The remaining resilience capacity on an edge e for working path i and failure pattern f ($RCEWPS^D_{d,i,e,s}$) is diminished by a fraction ($UUWCPES^D_{d,i,e,s}$) of the possible reusable working capacity ($UWCES^D_{e,s}$).

$$RCEWPS^D_{d,i,e,s} = \sum_{j \in \mathbb{I}_{\mathrm{RSplit}}} RPWPES^D_{d,i,e,j,s} - UUWCPES^D_{d,i,e,s} \qquad \forall d \in \mathbb{D}, i \in \mathbb{I}_{\mathrm{WSplit}}, \forall e \in \mathbb{E}, f \in \mathbb{F} \tag{4.61}$$

$$UWCES^D_{e,s} \geq \sum_{d \in \mathbb{D}} \sum_{i \in \mathbb{I}_{\mathrm{WSplit}}} UUWCPES^D_{d,i,e,s} \qquad \forall e \in \mathbb{E}, \forall f \in F \tag{4.62}$$

Consequently, the required resilience capacity on an edge e for a whole demand d in case of failure pattern f is the sum of the backup paths of all working paths and can be calculated according to Equation (4.63). To provide enough capacity the maximum amount for all individual failure patterns has to be provided on an edge (Equation (4.64)).

$$RCEWS^D_{d,e,s} = \sum_{i \in \mathbb{I}_{\mathrm{WSplit}}} RCEWPS^D_{d,i,e,s} \qquad \forall d \in \mathbb{D}, \forall e \in \mathbb{E}, \forall f \in \mathbb{F} \tag{4.63}$$

$$RCEW^D_{d,e} \geq RCEWS^D_{d,e,s} \qquad \forall d \in \mathbb{D}, \forall e \in \mathbb{E}, \forall f \in \mathbb{F} \tag{4.64}$$

However, as seen in Chapter 3 resilience capacity can be reserved and used individually for each demand (dedicated) or shared between working paths of different demands that are not affected simultaneously by a failure pattern (shared). Thus, the real required

resilience capacity on an edge e (RCE_e^D) is a combination of dedicated ($DRCE_e^D$) and shared ($SRCE_e^D$) capacity (Equation (4.65)). The required dedicated capacity on an edge e can be calculated according to Equation (4.66). Similarly, if capacity can be shared between different demands, the required shared capacity on an edge e for one or for all failure patterns will be calculated using Equations (4.67) and (4.68).

$$RCE_e^D = SRCE_e^D + DRCE_e^D \qquad \forall e \in \mathbb{E} \tag{4.65}$$

$$DRCE_e^D = \sum_{d \in \mathbb{D}, d \text{ is dedicated}} RCEW_{d,e}^D \qquad \forall e \in \mathbb{E} \tag{4.66}$$

$$SRCES_{e,s}^D \geq \sum_{d \in \mathbb{D}, d \text{ is shared}} RCEWS_{d,e,s}^D \qquad \forall e \in \mathbb{E}, \forall f \in \mathbb{F} \tag{4.67}$$

$$SRCE_e^D \geq SRCES_{e,s}^D \qquad \forall e \in \mathbb{E}, \forall f \in \mathbb{F} \tag{4.68}$$

Used Capacity on Edges:
Consequently, the used capacity on an edge can be calculated as a sum of working and backup capacity.

$$UCE_e^D = RCE_e^D + WCE_e^D \qquad \forall e \in \mathbb{E} \tag{4.69}$$

Resilience Flow Conservation - Part A:
Flow conservation is required for detoured backup flows ($RPWPES_{d,i,e,j,s}^D$) additionally to the defined basic resilience constraints. If a failure occurs, traffic will be shifted from an affected working path to backup paths, transported along them and removed at the end of the paths again. Depending on the protection mechanism and the number of element failures, the location of the end-nodes of the backup paths can easily be obtained. Equations (4.70a) to (4.72) show equations for these mechanism and Table 4.8 summarizes the detour points.

$\forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall n \in \mathbb{N}, \forall f \in \mathbb{F}, \forall e \in \mathbb{E}_f, n \notin \mathbb{F}$:

n is detour point A:
$$\begin{cases} IRWPNS_{d,i,n,s}^D = 0 & \text{(4.70a)} \\ ORWPNS_{d,i,n,s}^D \geq WPE_{d,i,e}^D & \text{(4.70b)} \end{cases}$$

n is detour point B:
$$\begin{cases} IRWPNS_{d,i,n,s}^D \geq WPE_{d,i,e}^D & \text{(4.71a)} \\ ORWPNS_{d,i,n,s}^D = 0 & \text{(4.71b)} \end{cases}$$

else: $\forall j \in \mathbb{I}_{\text{RSplit}}$
$$\begin{cases} IRPWPNS_{d,i,n,j,s}^D = \\ ORPWPNS_{d,i,n,j,s}^D & \text{(4.72)} \end{cases}$$

Table 4.8: Detour points A and B for Equations (4.70a) to (4.72).

Resilience Mechanism:	Detour point A	Detour point B
End-2-end path protection	n is $source(d)$	n is $target(d)$
Local-link protection with single link failure	n is $source(e)$	n is $target(e)$
Local-to-egress protection with single element failure	n is $source(e)$	n is $target(d)$

Resilience Flow Conservation - Part B:
The locations of the detours are dependent on the working paths and the order of failures therein, when considering multiple failures or regional protection. Thus, additional constraints are required to identify these detour points. As already presented in Section 3.5.1 all considered path-based resilience mechanisms can be built using regional resilience. For simplicity, we therefore present equations for regional resilience in the following only.

In regional resilience, the backup paths start between MaxFront^I and MinFront^I hops in front of the first failure and end between MinBack^I and MaxBack^I hops behind the last failure. Additionally, the backup path has to be routed disjoint to the working path in the forbidden regions between MinFront^I and MinBack^I around a failure (Figure 4.10(a)). However, if the distance of the source to the first failure is shorter than MaxFront^I the backup path will have to start at the source node. Similarly, if the distance of the last failure to the target node of the demand is shorter than MaxBack^I, the backup path will end at the target node (Figure 4.10(b)).

Illustrated in Figure 4.11, the exact location of the backup path end-points in the allowed region along the working path is unimportant from a capacity point of view (node B or node C and node F or G). Therefore, in the following we assume that backup paths start at the minimal possible node and end at the maximum possible node along the path. If required, resilience path parts routed in parallel to working paths will be able to be reduced in a post-processing step.

First the maximum and minimum used path index ($MinWPI^I_{d,i}$ and $MaxWPI^I_{d,i}$) have to be determined using Equations (4.73) to (4.76) and (4.77) to (4.80).

$$MinWPI^I_{d,i} \leq WPI^I_{d,i,e} + (1 - WPE^B_{d,i,e}) \cdot \text{Max}^D \qquad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall e \in \mathbb{E} \tag{4.73}$$

$$MinWPI2^B_{d,i,e} \leq WPE^B_{d,i,e} \qquad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall e \in \mathbb{E} \tag{4.74}$$

$$\sum_{e \in \mathbb{E}} MinWPI2^B_{d,i,e} = 1 \qquad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}} \tag{4.75}$$

$$MinWPI^I_{d,i} \geq WPI^I_{d,i,e} - (1 - MinWPI2^B_{d,i,e}) \cdot \text{Max}^D - (1 - WPE^B_{d,i,e}) \qquad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}} \tag{4.76}$$

The minimum index is limited from above according to Equation (4.73) and is at least equal to any enumeration index along the working path ($WPE^B_{d,i,e} = 1$). Furthermore, exactly one variable $MinWPI2^B_{d,i,e}$ along the working path (Equation (4.74)) is forced to

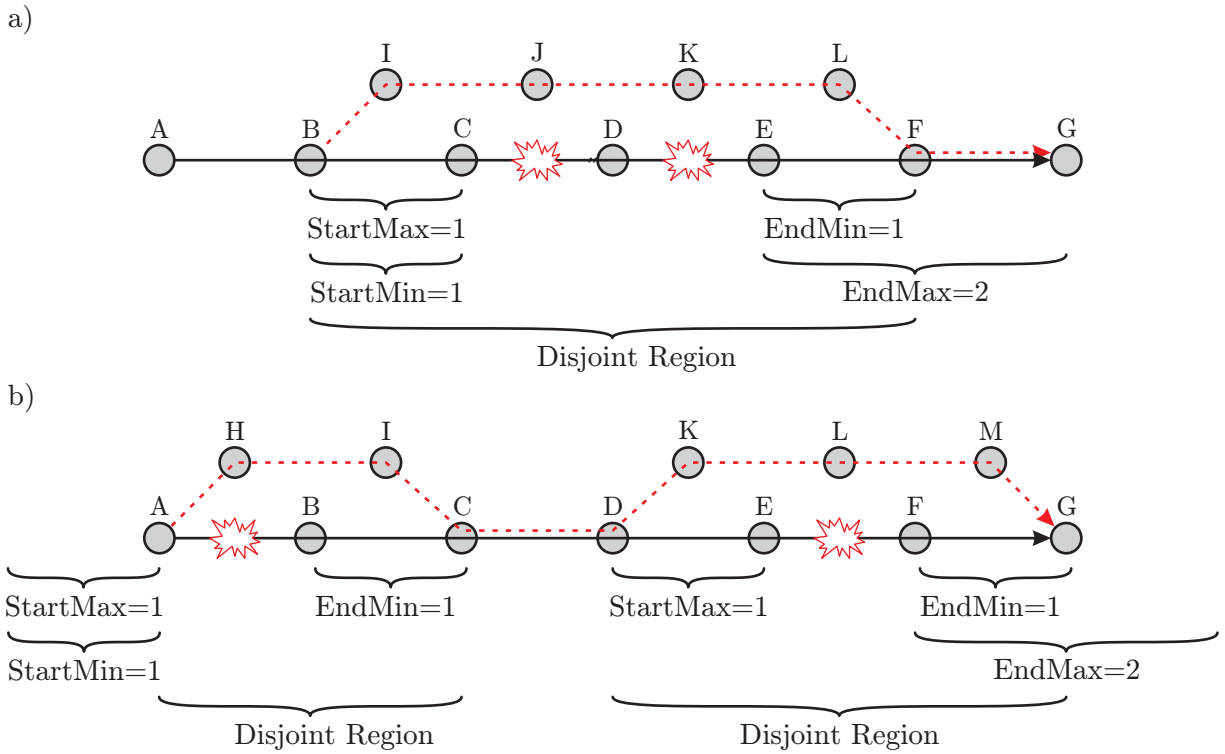


Figure 4.10: Example of regional backup paths. If the distances to the end-points of the demands are smaller than the parameters MaxFront^I or MaxBack^I, the end-points will be used as start or end of the regional backup path.

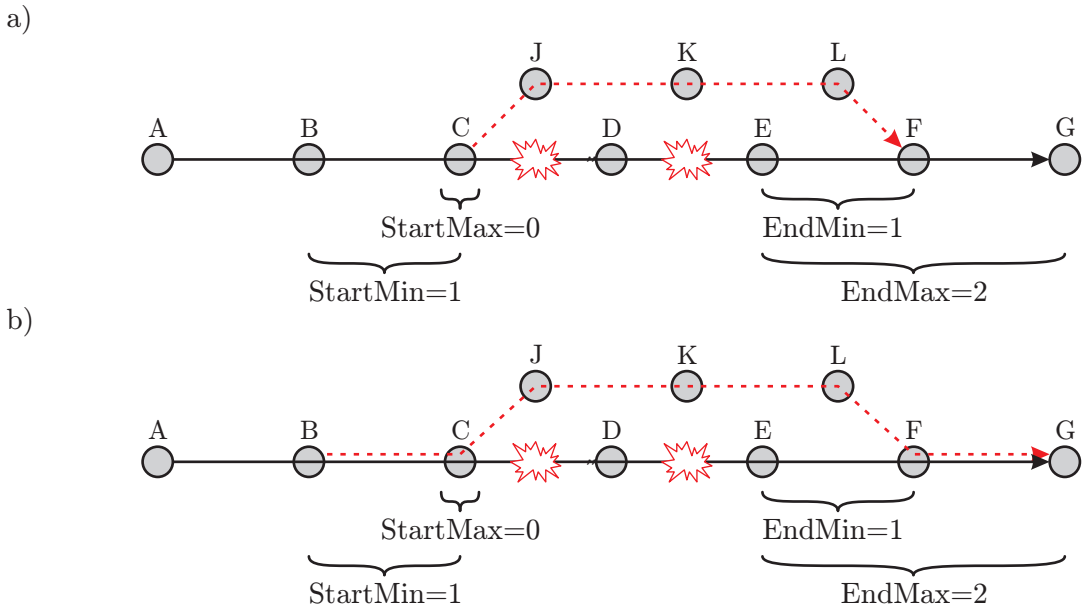


Figure 4.11: Example of regional backup paths. The exact location of the detour end-points inside the allowed region is unimportant from a capacity point of view.

be one by Equation (4.75). Thus, using Equation (4.76) the minimum index is bound from below to be greater than exactly one working index along the path. Again, as a combination of all four equations, the minimum index is equal to the index of the first edge along the path. Note, that if an edge is not used by the working path ($WPE^B_{d,i,e} = 0$), no further bounds will be imposed by the four equations.

Similarly to the calculation of the minimum index, Equations (4.77) to (4.80) set the value of variable $MaxWPI^I_{d,i}$ equal to the index of the last edge along the working path.

$$MaxWPI^I_{d,i} \geq WPI^I_{d,i,e} - (1 - WPE^B_{d,i,e}) \cdot \text{Max}^D \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall e \in \mathbb{E} \tag{4.77}$$

$$MaxWPI2^B_{d,i,e} \leq WPE^B_{d,i,e} \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall e \in \mathbb{E} \tag{4.78}$$

$$\sum_{e \in \mathbb{E}} MaxWPI2^B_{d,i,e} = 1 \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}} \tag{4.79}$$

$$MaxWPI^I_{d,i} \leq WPI^I_{d,i,e} + (1 - MaxWPI2^B_{d,i,e}) \cdot \text{Max}^D + (1 - WPE^B_{d,i,e}) \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}} \tag{4.80}$$

Let $SMin^I_{d,i,s}$, and $EMax^I_{d,i,s}$ denote bounds of the indices of the start and end location of the detour dependent on failure f. Using the already presented index variable for the first failure along a working path $WPILS^I_{d,i,s}$ we can limit the variables according to Equations (4.81) to (4.86). Equation (4.81) limits the start node of the detour from below by the distance to the first failure (parameter MaxFront^I). However, if this distance is smaller than the distance to the source node of the demand, Equation (4.82) will add a more stringent lower bound to the location of the detour.

$\forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall f \in \mathbb{F}$:

$$SMin^I_{d,i,s} \geq WPILS^I_{d,i,s} - \text{MaxFront}^I \tag{4.81}$$

$$SMin^I_{d,i,s} \geq MinWPI^I_{d,i} \tag{4.82}$$

$$SMin^I_{d,i,s} \leq WPILS^I_{d,i,s} - \text{MinFront}^I + SMax^I_{d,i,s} \cdot \text{Max}^D \tag{4.83}$$

$$SMin^I_{d,i,s} \leq MinWPI^I_{d,i} + (1 - Start^B_{d,i,s}) \cdot \text{Max}^D \tag{4.84}$$

$$WPILS^I_{d,i,s} - \text{MinFront}^I \geq MinWPI^I_{d,i} - (1 - Start^B_{d,i,s}) \cdot \text{Max}^D \tag{4.85}$$

$$WPILS^I_{d,i,s} - \text{MinFront}^I \leq MinWPI^I_{d,i} + Start^B_{d,i,s} \cdot \text{Max}^D \tag{4.86}$$

In contrast to that, upper bounds are applied using Equations (4.83) and (4.84). Depending on the value of the Boolean variable $Start^B_{d,i,s}$ this upper bound is either defined by Equation (4.83) to be MinFront^I hops in front of the first failure ($Start^B_{d,i,s} = 0$) or defined by Equation (4.84) to be the demand source node itself ($Start^B_{d,i,s} = 1$). Which of these two equations is activated depends on the distance of the first failure to the demand source node. If the index of the first failure reduced by parameter MinFront^I is smaller than the index of the source node ($WPILS^I_{d,i,s} - \text{MinFront}^I < MinWPI^I_{d,i}$), Equation (4.85) will be able to be fulfilled if $Start^B_{d,i,s}$ is zero. Thus, Equation (4.83) is activated. Otherwise

$Start^B_{d,i,s}$ is forced to be one according to Equation (4.86) and the upper bound of $SMax^I_{d,i,s}$ is applied by Equation (4.84).

For the location of the end of the detour, similar equations are required. Equations (4.87) to (4.88) thus limit variable $EMax^I_{d,i,s}$ from below. Dependent on the distance of the failure to the destination node of the demand either Equation (4.87) or (4.88) provide a more stringent upper bound to $EMax^I_{d,i,s}$.

$\forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall f \in \mathbb{F}$:

$$EMax^I_{d,i,s} \leq WPIRS^I_{d,i,s} + \text{MaxBack}^I \tag{4.87}$$
$$EMax^I_{d,i,s} \leq MaxWPI^I_{d,i} \tag{4.88}$$
$$EMax^I_{d,i,s} \geq WPILS^I_{d,i,s} + \text{MinBack}^I - End^B_{d,i,s} \cdot \text{Max}^D \tag{4.89}$$
$$EMax^I_{d,i,s} \geq MaxWPI^I_{d,i} - (1 - End^B_{d,i,s}) \cdot \text{Max}^D \tag{4.90}$$
$$WPIRS^I_{d,i,s} + \text{MaxBack}^I \geq MaxWPI^I_{d,i} - End^B_{d,i,s} \cdot \text{Max}^D \tag{4.91}$$
$$WPIRS^I_{d,i,s} + \text{MaxBack}^I \leq MaxWPI^I_{d,i} + (1 - End^B_{d,i,s}) \cdot \text{Max}^D \tag{4.92}$$

Finally, dependent on Boolean variable $End^B_{d,i,s}$ Equation (4.89) ($End^B_{d,i,s} = 0$) or Equation (4.90) ($End^B_{d,i,s} = 1$) provides a lower bound to $End^I_{d,i,f}$. If the index of the last failure added by MinBack^I hops is smaller than the index of the destination node of the demand ($WPIRS^I_{d,i,s} + \text{MinBack}^I < MaxWPI^I_{d,i}$), Equation (4.91) will be able to be fulfilled if $End^B_{d,i,s}$ is zero. Thus, in this case Equation (4.89) is activated. Otherwise, if the minimal end-point of the detour is greater than the index of the last edge along the path ($WPIRS^I_{d,i,s} + \text{MinBack}^I > MaxWPI^I_{d,i}$), $End^B_{d,i,s}$ will be forced to be one by Equation (4.92) and the end of the detour is limited to the end of the working path by Equation (4.90).

Nodes able to be a start of a detour can be identified with Equations (4.93) to (4.95). Only if the index of an edge along the working path is exactly the lower bound, Boolean variable $startA^B_{d,i,e,f}$ will be allowed to be 1 according to Equations (4.93) and (4.94). Consequently, only if this variable is one and the edge is part of the working path variable $startB^B_{d,i,e,f}$ will be allowed to be 1 in Equation (4.95). Following Equations (4.96) and (4.97) the Boolean variable $StartN^B_{d,i,n,s}$ will have to be true if at least for one outgoing edge the constraints are met. As a combination of these five equations, nodes with an attached variable of $StartN^B_{d,i,n,s}$ equal to one are identified as start nodes of a detour for

a failure pattern f.

$$SMin^I_{d,i,s} - WPI^I_{d,i,e} \leq (1 - startA^B_{d,i,e,f}) \cdot \text{Max}^D \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall e \in \mathbb{E}, \forall f \in \mathbb{F} \tag{4.93}$$

$$WPI^I_{d,i,e} - SMin^I_{d,i,s} \leq (1 - startA^B_{d,i,e,f}) \cdot \text{Max}^D \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall e \in \mathbb{E}, \forall f \in \mathbb{F} \tag{4.94}$$

$$2 \cdot startB^B_{d,i,e,f} \leq startA^B_{d,i,e,f} + WPE^B_{d,i,e} \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall e \in \mathbb{E}, \forall f \in \mathbb{F} \tag{4.95}$$

$$StartN^B_{d,i,n,s} \leq \sum_{e \in \text{outgoing}(n)} startB^B_{d,i,e,f} \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall n \in \mathbb{N}, \forall f \in \mathbb{F} \tag{4.96}$$

$$StartN^B_{d,i,n,s} \cdot \text{Max}^D \geq \sum_{e \in \text{outgoing}(n)} startB^B_{d,i,e,f} \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall n \in \mathbb{N}, \forall f \in \mathbb{F} \tag{4.97}$$

Similarly, the end nodes of a regional backup path detour can be identified using upper bound $EMax^I_{d,i,s}$ according to Equations (4.98) to (4.101). Nodes with a positive attached variable of $EndN^B_{d,i,n,s}$ equal to one are identified as end nodes of a detour for a failure pattern f.

$$EMax^I_{d,i,s} - WPI^I_{d,i,e} \leq (1 - endA^B_{d,i,e,f}) \cdot \text{Max}^D \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall e \in \mathbb{E}, \forall f \in \mathbb{F} \tag{4.98}$$

$$WPI^I_{d,i,e} - EMax^I_{d,i,s} \leq (1 - endA^B_{d,i,e,f}) \cdot \text{Max}^D \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall e \in \mathbb{E}, \forall f \in \mathbb{F} \tag{4.99}$$

$$2 \cdot endB^B_{d,i,e,f} \leq endA^B_{d,i,e,f} + WPE^B_{d,i,e} \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall e \in \mathbb{E}, \forall f \in \mathbb{F} \tag{4.100}$$

$$EndN^B_{d,i,n,s} \leq \sum_{e \in \text{outgoing}(n)} endB^B_{d,i,e,f} \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall n \in \mathbb{N}, \forall f \in \mathbb{F} \tag{4.101}$$

$$EndN^B_{d,i,n,s} \cdot \text{Max}^D \geq \sum_{e \in \text{outgoing}(n)} endB^B_{d,i,e,f} \quad \forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall n \in \mathbb{N}, \forall f \in \mathbb{F} \tag{4.102}$$

Finally, Equations (4.103) to (4.108) formulate the resilience flow conservation. If the node n is the start of the detour, variable $StartN^B_{d,i,n,s}$ will be one. Thus, at least the amount of the working path flow variable $WPE^D_{d,i,e}$ has to be detoured at the node according to Equation (4.103) and no backup traffic is allowed to terminate at the node according to Equation (4.105). Similarly, for the node n at the end of the detour the

variable $EndN^B_{d,i,n,s}$ is one. Thus the incoming traffic equals the affected traffic according to Equation (4.104) and no outgoing traffic for the resilience path is allowed at the node according to Equation (4.106).

$\forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall n \in \mathbb{N}, \forall f \in \mathbb{F}, \forall e \in \mathbb{E}_f, n \notin \mathbb{F}$:

$$ORWPNS^D_{d,i,n,s} \geq WPE^D_{d,i,e} - (1 - StartN^B_{d,i,n,s}) \cdot \text{Max}^D \tag{4.103}$$

$$IRWPNS^D_{d,i,n,s} \geq WPE^D_{d,i,e} - (1 - EndN^B_{d,i,n,s}) \cdot \text{Max}^D \tag{4.104}$$

$\forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall n \in \mathbb{N}, \forall f \in \mathbb{F}, n \notin \mathbb{F}$:

$$IRWPNS^D_{d,i,n,s} \leq (1 - StartN^B_{d,i,n,s}) \cdot \text{Max}^D \tag{4.105}$$

$$ORWPNS^D_{d,i,n,s} \leq (1 - EndN^B_{d,i,n,s}) \cdot \text{Max}^D \tag{4.106}$$

If the node n is neither the start-point nor the end-point of the detour, both variables $StartN^B_{d,i,n,s}$ and $EndN^B_{d,i,n,s}$ will be zero and no stringent restrictions are applied by Equations (4.103) to (4.106). However, strict flow conservation is required for these nodes using Equations (4.107) and (4.108).

$\forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall n \in \mathbb{N}, \forall j \in \mathbb{I}_{\text{RSplit}}, \forall f \in \mathbb{F}, n \notin \mathbb{F}$:

$$ORPWPNS^D_{d,i,n,j,s} \geq IRPWPNS^D_{d,i,n,j,s} - StartN^B_{d,i,n,s} \cdot \text{Max}^D - EndN^B_{d,i,n,s} \tag{4.107}$$

$$ORPWPNS^D_{d,i,n,j,s} \leq IRPWPNS^D_{d,i,n,j,s} + StartN^B_{d,i,n,s} \cdot \text{Max}^D + EndN^B_{d,i,n,s} \tag{4.108}$$

Additionally, in order to model the disjoint regions, backup paths are not allowed to traverse edges between MinFront^I and MinBack^I around a failure. Following Equations (4.109) and (4.110), both variables $disjA^B_{d,i,e,f}$ and $disjB^B_{d,i,e,f}$ will have to be 1 if the index of a considered edge is in between a disjoint region around a failure pattern. Therefore, the backup path is not allowed to traverse these edges using Equation (4.111).

$$WPI^I_{d,i,e} \geq WPI^I_{d,i,e2} - \text{MinFront}^I - disjA^B_{d,i,e,f} \cdot \text{Max}^D$$
$$\forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall e \in \mathbb{E}, \forall f \in \mathbb{F}, \forall e2 \in \mathbb{E}_f \tag{4.109}$$

$$WPI^I_{d,i,e} \geq WPI^I_{d,i,e2} + \text{MinBack}^I - disjB^B_{d,i,e,f} \cdot \text{Max}^D$$
$$\forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall e \in \mathbb{E}, \forall f \in \mathbb{F}, \forall e2 \in \mathbb{E}_f \tag{4.110}$$

$$RCEWPS^D_{d,i,e,s} \leq (2 - disjA^B_{d,i,e,f} - disjB^B_{d,i,e,f}) \cdot \text{Max}^D$$
$$\forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall e \in \mathbb{E}, \forall f \in \mathbb{F} \tag{4.111}$$

Finally, for regional protection with a positive MinFront^I value the detour is not immediately in front of the failure and Boolean variable $Detour^B_{d,i,e,s}$ has to be changed according to the detour location ($SMin^I_{d,i,s}$). Thus, Equation (4.55) has to be exchanged with Equation (4.112) for regional resilience.

$$WPI^I_{d,i,e} - SMin^I_{d,i,s} \geq (Detour^B_{d,i,e,s} - 1) \cdot \text{Max}^D$$
$$\forall d \in \mathbb{D}, \forall i \in \mathbb{I}_{\text{WSplit}}, \forall e \in \mathbb{E}, \forall f \in \mathbb{F} \tag{4.112}$$

4.2.2.3 Building Block Combinations

We can combine the individual building blocks to form complete resilience mechanism models. Table 4.9 shows the combination for the four considered resilience mechanisms for a protection of single link failures.

Table 4.9: Example combinations of building blocks for single link failures.

Resilience Mechanism	SE2EPP	SL2EPP	SRPP	SLLPP
Failure Free Flow Conservation	x	x	x	x
Used Working Capacity on Edges	x	x	x	x
Basic Resilience	x	x	x	x
Enumeration of Working Paths			x	
Calculation of the Lowest Failure Index			x	
Calculation of the Highest Failure Index			x	
Used Resilience Capacity on Edges	x	x	x	x
Resilience Flow Conservation Part A	x	x	x	x
Resilience Flow Conservation Part B			x	
Used Capacity on Edges	x	x	x	x
Minimization of Edge Capacity	x	x	x	x

4.2.3 Path-based Formulations

As seen in Section 4.1.2 instead of modeling the traversal of flows through a node it is also possible to model the distribution of traffic onto pre-defined structures. Therefore, possible paths have to be calculated and one variable is attached to each path. Other path-based formulations can be found e.g. in [RSM03, SP04, Gro04].

4.2.3.1 Sets, Variables, and Parameters

Again, we model the network as a directed graph $G = (\mathbb{N}, \mathbb{E})$ where $\mathbb{N}$ represents the possible set of nodes and $\mathbb{E}$ the set of possible edges that can be used to form the optimal resilient routing constellation. We represent each duct of the network as a pair of counter-directional edges. Furthermore, a demand $d \in \mathbb{D}$ represents a demand value that has to be routed between two nodes. The different failure states are denoted as $s \in \mathbb{S}$ whereas s_0 denotes the failure-free state. As presented in Section 4.1.2 the path approach models traffic flow along pre-determined paths. The set of all possible working paths for a given demand d in a network state s is denoted as $\mathbb{P}_{d,s}$ and the set of resilience paths protecting a path p of demand d in the network state s as $\mathbb{P}'_{d,s,p}$.

Using these sets, specific resilience paths can be used for different resilience mechanisms. Thus, implementation issues and understanding of the approaches is simplified: One formulation approach can be used for all path-based protection mechanisms.

Table 4.10: Additional sets used in the path approach formulation.

Symbol	Description
$\mathbb{P}_d$	Possible working path of a demand d in failure-free state s_0.
$\mathbb{P}_{d,s}$	Possible working path of a demand d during failure state s.
$\mathbb{P}'_{d,s,p}$	Possible resilience path of a demand d and path p during failure state s.

Table 4.11: Additional important variables and parameters used in the path approach formulation.

Symbol	Type	Index	Description
$RCDS^D_{d,s}$	real	$d \in \mathbb{D}$, $s \in \mathbb{S}$	The used ***r***esilience ***c***apacity for ***d***emand d that is required in network ***s***tate s.
$RCPS^D_{d,s,p,p'}$	real	$d \in \mathbb{D}$, $s \in \mathbb{S}$, $p \in \mathbb{P}_d$, $p' \in \mathbb{P}_{d,s,p}$	The required ***r***esilience (backup) ***c***apacity on ***p***ath p' that protects path p of demand d in network ***s***tate s.
$RPU^B_{d,s,p,p'}$	bool	$d \in \mathbb{D}$, $s \in \mathbb{S}$, $p \in \mathbb{P}_d$, $p' \in \mathbb{P}_{d,s,p}$	The indicator if a ***r***esilience (backup) ***p***ath p' of working path p of demand d is ***u***sed (capacity > 0) during network ***s***tate s.
$RPUC^I_{d,s,p}$	int	$d \in \mathbb{D}$, $s \in \mathbb{S}$, $p \in \mathbb{P}_d$	***C***ounter how many ***r***esilience ***p***aths are ***u***sed for demand d and working path p.
$WCDS^D_{d,s}$	real	$d \in \mathbb{D}$, $s \in \mathbb{S}$	The required ***w***orking ***c***apacity for ***d***emand d that is required in network ***s***tate s.
$WCP^D_{d,p}$	real	$d \in \mathbb{D}$, $p \in \mathbb{P}_d$	The required ***w***orking ***c***apacity on ***p***ath p for demand d.
$WPU^B_{d,p}$	bool	$d \in \mathbb{D}$, $p \in \mathbb{P}_d$	The indicator if a ***w***orking ***p***ath p of demand d is ***u***sed (capacity > 0).
$WPUC^I_d$	int	$d \in \mathbb{D}$	***C***ounter how many ***w***orking ***p***aths are ***u***sed for demand d.

Failure Free Routing:
In failure-free state s_0 traffic for a demand relation d is routed along a working path p. The amount of capacity that is required on the path is expressed by $WCP^D_{d,p}$. The sum of

all working paths have to be at least equal to the demand value D_d^D.

$$WCDS_{d,s}^D = \sum_{p \in \mathbb{P}d,s} WCP_{d,p}^D \qquad \forall d \in \mathbb{D}, \forall s \in \mathbb{S} \tag{4.113a}$$

$$WCDS_{d,s_0}^D \geq D_d^D \qquad \forall d \in \mathbb{D} \tag{4.113b}$$

4.2.3.2 Constraint Building Blocks

Failure Affected Routing:
Similarly, traffic on a working path p will be detoured on resilience path p', if the working path is affected. Variable $RCPS_{d,s,p,p'}^D$ denotes the amount of capacity that is required on resilience path p'. In any network state s the demand value D_d^D should be routed. Parameter $k_{d,s}^D \in [0..1]$ additionally models a possible reduction of survivable traffic in a specific failure case.

$$RCDS_{d,s}^D = \sum_{\substack{p \in \mathbb{P}d,s_0 \\ p \cap s \neq \{\}}} \sum_{p' \in \mathbb{P}d,s,p} RCPS_{d,s,p,p'}^D \qquad \forall d \in \mathbb{D}, \forall s \in S \tag{4.114a}$$

$$WCDS_{d,s}^D + RCDS_{d,s}^D \geq k_{d,s}^D \cdot D_d^D \qquad \forall d \in \mathbb{D}, \forall s \in \mathbb{S} \tag{4.114b}$$

Used Working Capacity on Edges:
Consequently, the required working capacity on an edge e in a specific network state s ($WCES_{e,s}^D$) and the maximum required working capacity on an edge (WCE_e^D) can be modeled according to Equations (4.115a) and (4.115b).

$$WCES_{e,s}^D = \sum_{d \in \mathbb{D}} \sum_{\substack{p \in \mathbb{P}_{d,s} \\ e \in p}} WCP_{d,p}^D \qquad \forall e \in \mathbb{E}, \forall s \in \mathbb{S} \tag{4.115a}$$

$$WCE_e^D \geq WCES_{e,s}^D \qquad \forall e \in \mathbb{E}, \forall s \in \mathbb{S} \tag{4.115b}$$

Used Resilience Capacity on Edges:
The required shared capacity on an edge (RCE_e^D) can be calculated using Equations (4.116a) and (4.116b).

$$SRCES_{e,s}^D = \sum_{d \in \mathbb{D}} \sum_{\substack{p \in \mathbb{P}_{d,s_0} \\ s \cap p \neq \{\}}} \sum_{\substack{p' \in \mathbb{P}_{d,s,p} \\ e \in p'}} RCPS_{d,s,p,p'}^D \qquad \forall e \in \mathbb{E}, \forall s \in \mathbb{S} \tag{4.116a}$$

$$RCE_e^D \geq SRCES_{e,s}^D \qquad \forall e \in \mathbb{E}, \forall s \in \mathbb{S} \tag{4.116b}$$

Used Capacity on Edges:
Since working capacity can be reused in case of failures, the used capacity on edges is not a sum of WCE_e^D and RCE_e^D. Instead, Equations (4.117a) and (4.117b) models the required

capacity on edges.

$$\sum_{d\in\mathbb{D}}\Big(\sum_{\substack{p\in\mathbb{P}_{d,s}\\ e\in p}} WCP^D_{d,p} + \sum_{\substack{p\in\mathbb{P}_{d,s_0}\\ p\cap s\neq\{\}}}\sum_{\substack{p'\in\mathbb{P}_{d,s,p}\\ e\in p'}} RCPS^D_{d,s,p,p'}\Big) = UCES^D_{e,s} \qquad \forall e\in\mathbb{E}, \forall s\in\mathbb{S} \quad (4.117a)$$

$$UCE^D_e \geq UCES^D_{e,s} \qquad \forall e\in\mathbb{E}, \forall s\in\mathbb{S} \quad (4.117b)$$

Restriction of Capacity:
If the available capacity on an edge is restricted Equation (4.118a) will have to be added to the system.

$$-UCE^D_e \geq -C_e \qquad \forall e\in\mathbb{E} \quad (4.118a)$$

Restriction of Working Path Splits:
The number of splits in which a path can be split into can be restricted using Equations (4.119a) to (4.119c). A Boolean variable is forced to be to one if a working path is used ($WCP^D_{d,p} > 0$). Finally, the integer variable $WPUC^I_d$ that counts the number of used paths is limited from above to be max. $MaxWSplit^I_d$.

$$WPU^B_{d,p}\cdot \mathrm{Max}^D \geq WCP^D_{d,p} \qquad \forall d\in\mathbb{D}, \forall p\in\mathbb{P}_{d,s_0} \quad (4.119a)$$

$$WPUC^I_d = \sum_{p\in\mathbb{P}_{d,s_0}} WPU^B_{d,p} \qquad \forall d\in\mathbb{D} \quad (4.119b)$$

$$MaxWSplit^I_d \geq WPUC^I_d \qquad \forall d\in\mathbb{D} \quad (4.119c)$$

Restriction of Resilience Path Splits:
Similarly, the number of splits in which a working path can be split to can be restricted using Equations (4.120a) to (4.120c).

$$RPU^B_{d,s,p,p'}\cdot \mathrm{Max}^D \geq RCPS^D_{d,s,p,p'} \qquad \forall d\in\mathbb{D}, \forall p\in\mathbb{P}_{d,s}, \forall p'\in\mathbb{P}_{d,p,s}, \forall s\in\mathbb{S} \quad (4.120a)$$

$$RPUC^I_{d,s,p} = \sum_{p'\in\mathbb{P}_{d,p,s}} RPU^B_{d,s,p,p'} \qquad \forall d\in\mathbb{D}, \forall p\in\mathbb{P}_{d,s}, \forall s\in\mathbb{S} \quad (4.120b)$$

$$MaxRSplit^I_d \geq RPUC^I_{d,s,p} \qquad \forall d\in\mathbb{D}, \forall p\in\mathbb{P}_{d,s}, \forall s\in\mathbb{S} \quad (4.120c)$$

Failure Free Traffic Distribution:
In order to restrict the traffic distribution onto multiple paths to equal portions Equations (4.121a) to (4.121b) are required. The amount of traffic of two working paths is compared with each other. If both working paths are used variables $WPU^B_{d,p}$ are one. Thus, the amount of both working paths has to be equal. If only one working path is used,

no restrictions will be imposed by the equations.

$$WCP_{d,p_1} - WCP_{d,p_2} - D_d^D \cdot (2 - WPU_{d,p_1}^B - WPU_{d,p_2}^B) \geq 0 \quad \forall d \in \mathbb{D}, \forall p_1, p_2 \in \mathbb{P}_{d,s_0}, p1 \neq p2 \tag{4.121a}$$

$$D_d^D \cdot (WPU_{d,p_1}^B + WPU_{d,p_2}^B - 2) - WCP_{d,p_1} + WCP_{d,p_2} \geq 0 \quad \forall d \in \mathbb{D}, \forall p_1, p_2 \in \mathbb{P}_{d,s_0}, p_1 \neq p_2 \tag{4.121b}$$

Failure Affected Traffic Distribution:
Similar to the failure-free case, the amount of traffic on used resilience paths can be forced to be equal using Equations (4.122a) and (4.122b).

$$RCPS_{d,s,p,p_1'}^D - RCPS_{d,s,p,p_2'}^D - k_{d,s}^D \cdot D_d^D \cdot (2 - RPU_{d,s,p,p_1'}^B - RPU_{d,s,p,p_2'}^B) \geq 0 \quad \forall d \in \mathbb{D}, \forall p \in \mathbb{P}_{d,s_0}, \forall p_1', p_2' \in \mathbb{P}_{d,p,s}, p_1' \neq p_2' \tag{4.122a}$$

$$k_{d,s}^D \cdot D_d^D \cdot (RPU_{d,s,p,p_1'}^B + RPU_{d,s,p,p_2'}^B - 2) - RCPS_{d,s,p,p_1'}^D + RCPS_{d,s,p,p_2'}^D \geq 0 \quad \forall d \in \mathbb{D}, \forall p \in \mathbb{P}_{d,s_0}, \forall p_1', p_2' \in \mathbb{P}_{d,p,s}, p_1' \neq p_2' \tag{4.122b}$$

4.2.4 Column Generation

In order to calculate an optimal network design all paths have to be calculated and included (as variables) when using the path approach. Concerning millions of possible paths between two nodes, however, the calculation and storage of these paths is often not possible because of memory restrictions. The flow approach in contrast to that considers all paths implicitly. However, flow conservation equations have to be solved for each demand at each node. The number of equations and the problem complexity increases with the network size and the number of demand relations. Thus, solving a flow approach LP is impracticable for networks with more than 20 nodes today.

However, when inspecting optimal network designs, the number of chosen paths is relatively small. Thus, if one could predict - right from the beginning - which paths are likely to be chosen, the path approach could be used to calculate optimal network designs even for large networks. The result would still be optimal since paths that are not chosen for the optimal design (used capacity is zero) can be left out from the initial equation system.

The idea of *Column Generation* is to combine the knowledge about the initial optimization problem with its mathematical equation structure and to obtain candidate paths during the solving process. Variables (columns in the ILP formulation) are added gradually during the optimization process. This idea of *Column Generation* or *delayed column generation* appeared implicitly in the *Dantzig-Wolfe Decomposition* [DW60], which in turn was inspired by work of Ford and Fulkerson [FF58] and was since then used in Operation

Research.[8] However, only recently *Column Generation* found its application to network design problems (e.g. [TS04, **KOW+05**, **GKZ+05**, **GKO+05**]).

In the following we will explain the principle of *Column Generation* in more detail and will present formulas to efficiently plan resilient networks in the remainder of this chapter.

4.2.4.1 Duality and Pricing

Consider the initial problem of Equations (4.3):

$$\begin{aligned} \text{Primal: maximize } & x_1 + x_2 \\ \text{subject to: } & x_1 + 2x_2 \leq 7 && \text{(4.123a)} \\ & 2x_1 + x_2 \leq 6 && \text{(4.123b)} \\ & x_1, x_2 \geq 0 \end{aligned}$$

Equations (4.123a) and (4.123b) provide a constraint to the solution. If these two equations are multiplied with a weight factor $\lambda_1 > 0$ and $\lambda_2 > 0$ and are added together, the following equation will result:

$$\lambda_1 \cdot (x_1 + 2x_2) + \lambda_2 \cdot (2x_1 + x_2) \leq 7\lambda_1 + 6\lambda_2$$

Regrouped to variables x_1 and x_2 a relationship to the objective function appears.

$$(\lambda_1 + 2\lambda_2) \cdot x_1 + (2\lambda_1 + \lambda_2) \cdot x_2 \leq 7\lambda_1 + 6\lambda_2$$

The coefficients of the two variables x_i have to be at least 1 to form an upper bound to the objective.

$$\begin{aligned} (\lambda_1 + 2\lambda_2) &\geq 1 \\ (2\lambda_1 + \lambda_2) &\geq 1 \end{aligned}$$

Finally, if these constraints are met, an upper bound to the optimal value will be given by $7\lambda_1 + 6\lambda_2$. The complete so called *dual formulation* of the example LP is as follows:

$$\begin{aligned} \text{Dual: minimize } & 7\lambda_1 + 6\lambda_2 \\ \text{subject to: } & \lambda_1 + 2\lambda_2 \geq 1 \\ & 2\lambda_1 + \lambda_2 \geq 1 \\ & \lambda_1, \lambda_2 \geq 0 \end{aligned}$$

In general, any LP formulation can be transformed to its dual and vice versa. Notably, any optimal solution of a primal problem is an optimal solution for the dual formulation and the dual provides an upper bound to the primal problem. The complete proof of the strong and weak duality theorem is straightforward and can be found e.g. in [BT97]. Obviously, this transformation provides possibilities in speeding up the solution time, if the dual of the

Table 4.12: Transformation rules between primal and dual formulation

Primal		Dual	
minimize	$\mathbf{c}'\mathbf{x}$	maximize	$\mathbf{p}'\mathbf{b}$
subject to:	$\mathbf{a}_i'\mathbf{x} \geq b_i$	subject to:	$p_i \geq 0$
	$\mathbf{a_i}'\mathbf{x} \leq b_i$		$p_i \leq 0$
	$\mathbf{a_i}'\mathbf{x} = b_i$		p_i free
	$x_j \geq 0$		$\mathbf{p}'\mathbf{A}_j \leq c_j$
	$x_j \leq 0$		$\mathbf{p}'\mathbf{A}_j \geq c_j$
	x_j free		$\mathbf{p}'\mathbf{A}_j = c_j$

problem can be solved faster than the primal. Table 4.12 summarizes the transformation rules between a primal (left) and a dual (right) formulation of a linear program. Note, that for each constraint in the primal we introduce a variable in the dual and for each variable in the primal we add a constraint in the dual. This relationship between constraints and variables between the primal and the dual system provides the basis for the idea behind *Column Generation*: Adding variables only if they are required.

Figure 4.12 illustrates the basic principle of *Column Generation*. At the beginning of the process, a sub-set of variables is determined with which the primal optimization is started. With this subset of variables, the result is far from being optimal. However, since the number of variables is limited, the amount of required memory and the required solving time can be restricted. The idea of *Column Generation* is now to find missing variables that could improve the solution. For this, the so-called pricing problem has to be solved. Constraints in the dual system that are left out (due to missing primal variables) have to be checked whether they violate the solution. If this is not the case, the constraint will be insignificant and the primal variable will not be used for the optimal design. However, if a constraint violates the solution, the corresponding primal variable could improve the optimization and has to be added to the primal system. Thus, variables are added gradually to the primal system until no improving variable can be found anymore. Since no dual constraint violates the solution and the primal solution is optimal, the global optimal solution is found - even if not all variables are implicitly included in the formulations. In theory, the *Column Generation* process can end-up in adding all possible variables. Thus, *Column Generation* can be much slower compared to an optimization that starts with all variables right from the beginning. However, for most real-life problems, a small number of iterations is sufficient and *Column Generation* provides a fast mechanisms to find optimal solutions with limited memory requirements.

[8]Main applications have been the optimization of *crew-scheduling* problems (e.g. [Dan63]) and *cutting stock* problems (e.g. [GG61, GG63]).

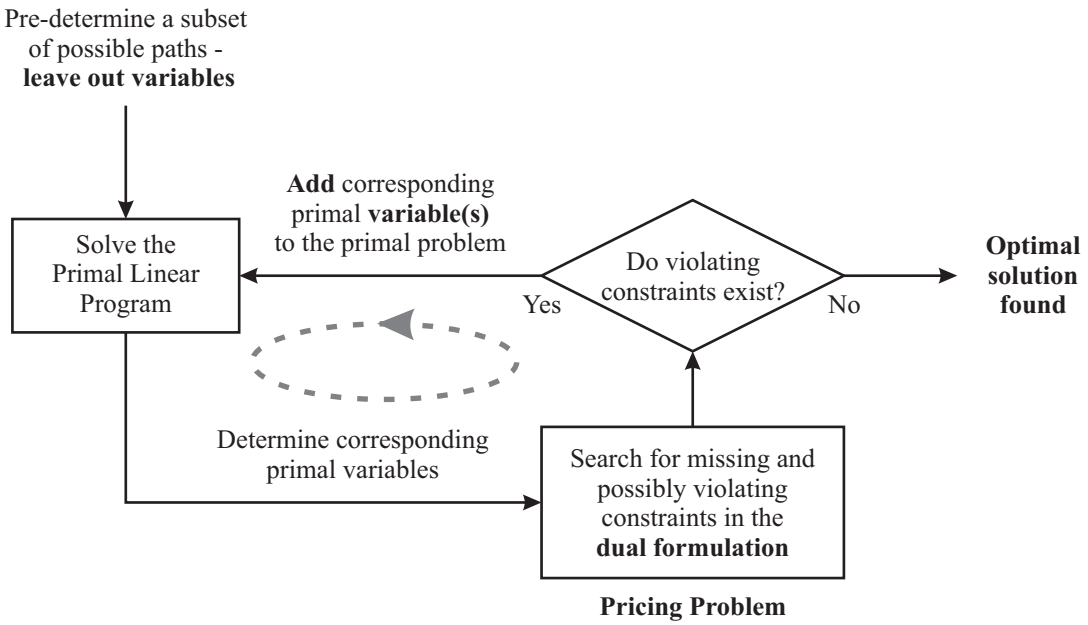


Figure 4.12: Principle of *Column Generation*.

4.2.4.2 Column Generation for Protection

In the following, we present novel formulations for the dual system of the resilient network-planning problem and discuss the resulting pricing problems. Equations (4.124) summarize the path approach formulas from Section 4.2.3. A dual variable has to be assigned to each primal constraint (indicated in brackets).

$$\text{minimize} \sum_{e \in \mathbb{E}} UCE_e^D \tag{4.124a}$$

$$[\pi_{d,s}] \qquad \sum_{p \in \mathbb{P}_{d,s}} WCP_{d,p}^D + \sum_{\substack{p \in \mathbb{P}_{d,s_0} \\ s \cap p \neq \{\}}} \sum_{p' \in \mathbb{P}'_{d,s,p}} RCPS_{d,s,p,p'}^D \geq k_{d,s}^D \cdot D_d^D \qquad \forall d \in \mathbb{D}, \forall s \in \mathbb{S} \tag{4.124b}$$

$$\sum_{d \in \mathbb{D}} \Big(\sum_{\substack{p \in \mathbb{P}_{d,s} \\ e \in p}} WCP_{d,p}^D + \sum_{\substack{p \in \mathbb{P}_{d,s_0} \\ s \cap p \neq \{\}}} \sum_{\substack{p' \in \mathbb{P}'_{d,s,p} \\ e \in p'}} RCPS_{d,s,p,p'}^D \Big) = UCES_{e,s}^D \qquad \forall e \in \mathbb{E}, \forall s \in \mathbb{S} \tag{4.124c}$$

$$[\sigma_{e,s}] \qquad UCE_e^D \geq UCES_{e,s}^D \qquad \forall e \in \mathbb{E}, \forall s \in \mathbb{S} \tag{4.124d}$$

$$[\tau_e] \qquad -UCE_e^D \geq -C_e^D \qquad \forall e \in \mathbb{E} \tag{4.124e}$$

$$WCP_{d,p}^D \geq 0 \qquad \forall d \in \mathbb{D}, \forall p \in \mathbb{P}_{d,s_0} \tag{4.124f}$$

$$RCPS_{d,s,p,p'}^D \geq 0 \qquad \forall d \in \mathbb{D}, \forall s \in \mathbb{S}, \forall p \in \mathbb{P}_{d,s_0}, \forall p' \in \mathbb{P}_{d,p,s} \tag{4.124g}$$

$$UCE_e^D \geq 0 \qquad \forall e \in \mathbb{E} \tag{4.124h}$$

The schematic coefficient matrix notation shows the structure of the path approach in matrix-vector notation and helps to find the coefficients for the dual formulation.

Table 4.13: Schematic coefficient matrix: $\oplus = +1$, $\ominus = -1$

	$WCP^D_{d,p}$	$RCPS^D_{d,s,p,p'}$	UCE^D_e
$\pi_{s,d}$	$\oplus\ \oplus\ \oplus$ $\ \ \oplus\ \oplus$ $\oplus\ \ \oplus$	$\oplus\ \oplus\ \oplus\ \oplus\ \oplus$ $\oplus\ \oplus\ \oplus$	
$\sigma_{e,s}$	$\ \ \ominus\ \ \ominus$ $\ominus\ \ominus\ \ominus\ \ominus$ $\ominus\ \ominus\ \ominus\ \ominus$	$\ominus\ \ominus\ \ominus$ $\ominus\ \ominus\ \ominus$ $\ominus\ \ominus$	$\oplus$ $\oplus$ $\oplus$
τ_e			$\oplus$ $\oplus$ $\oplus$

A working path variable $WCP^D_{d,p}$ appears once for each demand and failure state if it is not affected by the failure ($p \cap s = \{\}$). Furthermore, the path will be subtracted from the used capacity if p uses edge e ($e \in p$) and p does not contain any errors ($p \cap s = \{\}$). The working path does not appear directly when dealing with physical capacity constraints. Additionally, the variable is restricted to be positive. Thus, the dual constraint corresponding to variable $WCP^D_{d,p}$ is as follows:

$$\sum_{\substack{s \in \mathbb{S} \\ p \cap s = \{\}}} \pi_{d,s} - \sum_{\substack{s \in \mathbb{S} \\ p \cap s = \{\}}} \sum_{e \in p} \sigma_{e,s} \leq 0, \quad \forall d \in \mathbb{D}, \forall p \in \mathbb{P}_d$$

.

A resilience path variable $RCPS^D_{d,s,p,p'}$ appears only once for a demand given a failure state and working path. It will be subtracted from the used capacity if e is used by p' and is not part of the failure. Again, the variable is restricted to be positive. Thus, the dual constraint for the resilience variable is as follows:

$$\pi_{d,s} - \sum_{\substack{e \in p' \\ e \notin s}} \sigma_{e,s} \leq 0, \quad \forall d \in \mathbb{D}, \forall s \in \mathbb{S}, s \neq s_0, \forall p \in \mathbb{P}_d, \forall p' \in \mathbb{P}'_{d,s,p}$$

The capacity variable UCE^D_e is used in all network states $s \in \mathbb{S}$ and in the physical capacity constraint. Following this, the dual constraint for the capacity variable is as follows:

$$\sum_{s \in \mathbb{S}} \sigma_{e,s} - \tau_e \leq 1, \quad \forall e \in \mathbb{E}$$

Finally, the objective function can be derived from the right side of the primal equation system.

$$\text{maximize} \left(\sum_{d \in \mathbb{D}} \sum_{s \in \mathbb{S}} k^D_{d,s} \cdot D^D_d \pi_{d,s} - \sum_{e \in \mathbb{E}} (C^D_e \cdot \tau_e) \right)$$

The complete dual LP consequently reads as

$$\text{maximize } (\sum_{d \in \mathbb{D}} \sum_{s \in \mathbb{S}} (k^D_{d,s} \cdot D^D_d \cdot \pi_{d,s}) - \sum_{e \in \mathbb{E}} (C^D_e \cdot \tau_e)) \tag{4.125a}$$

$$[WCP^D_{d,p}] \qquad \sum_{\substack{s \in \mathbb{S} \\ p \cap s = \{\}}} \pi_{d,s} - \sum_{\substack{s \in \mathbb{S} \\ p \cap s = \{\}}} \sum_{e \in p} \sigma_{e,s} \leq 0 \qquad \forall d \in \mathbb{D}, \forall p \in \mathbb{P}_d \tag{4.125b}$$

$$[RCPS^D_{d,s,p,p'}] \qquad \pi_{d,s} - \sum_{\substack{e \in p' \\ e \notin s}} \sigma_{e,s} \leq 0 \qquad \forall d \in \mathbb{D}, \forall s \in \mathbb{S}, s \neq s_0, \forall p \in \mathbb{P}_d, \forall p' \in \mathbb{P}'_{d,s,p} \tag{4.125c}$$

$$[UCE^D_e] \qquad \sum_{s \in \mathbb{S}} \sigma_{e,s} - \tau_e \leq 1 \qquad \forall e \in \mathbb{E} \tag{4.125d}$$

$$\pi_{d,s} \geq 0 \qquad \forall d \in \mathbb{D}, \forall s \in \mathbb{S} \tag{4.125e}$$

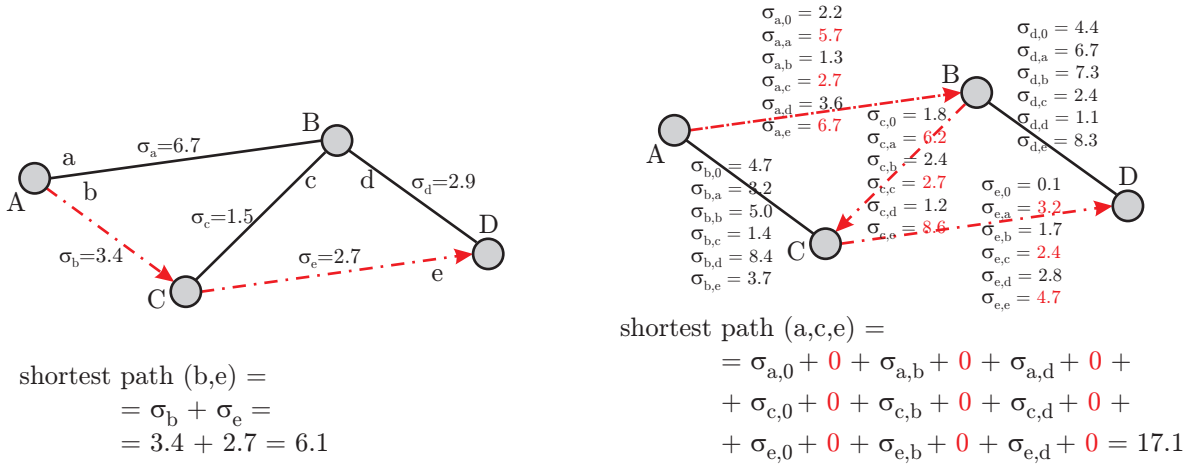
$$\sigma_{e,s} \geq 0 \qquad \forall e \in \mathbb{E}, \forall s \in \mathbb{S} \tag{4.125f}$$

$$\tau_e \geq 0 \qquad \forall e \in \mathbb{E} \tag{4.125g}$$

Solving the Pricing Problem:
The purpose to formulate the dual system is to find a criterion showing which variable of the primal system can improve the solution. Primal variables (paths) that are not included in the formulations correspond to left-out dual equations. However, primal variables will be able to improve the solution only, if the dual equations are violated by the current solution. Thus, we have to check equations (4.125b), (4.125c) and (4.125d) whether left-out equations exist that violate the given solution. Values for the dual variables $\pi_{d,s}$, $\sigma_{e,s}$ and τ_e can be obtained by the MIP solver after each optimization run.

When inspecting the dual formulation we observe that Equation (4.125d) can never be violated by any solution. This is because we added the capacity variables for all edges right from the beginning. Thus, no primal variable and no dual-constraint is missing. In contrast to that, we added only a sub-set of working and resilience path variables to the primal system at the beginning. Thus, some equations of (4.125b) and (4.125c) were missing in the dual.

The sum of all edges $\sigma_{e,s}$ of a path has to be bigger than $\pi_{d,s}$ to satisfy the equality. In other words, if a path exists that is smaller than $\pi_{d,s}$ the equality will be violated and the path potentially improves the solution. To test if any left-out constraint is violated, we could interpret the dual variable values $\sigma_{d,s}$ as link weights as illustrated in Figure 4.13(a).



shortest path (b,e) =
$= \sigma_b + \sigma_e =$
$= 3.4 + 2.7 = 6.1$

(a) Pricing problem for a resilience path.

shortest path (a,c,e) =
$= \sigma_{a,0} + 0 + \sigma_{a,b} + 0 + \sigma_{a,d} + 0 +$
$+ \sigma_{c,0} + 0 + \sigma_{c,b} + 0 + \sigma_{c,d} + 0 +$
$+ \sigma_{e,0} + 0 + \sigma_{e,b} + 0 + \sigma_{e,d} + 0 = 17.1$

(b) Pricing problem for a working path.

Figure 4.13: Illustration of the pricing problem for shared protection.

A shortest path algorithm[9] can then be applied on the resulting graph in failure state s with link weights $\sigma_{d,s}$. If the shortest path is longer than $\pi_{d,s}$, the constraints will not violate the solution and no path will have to be added to the primal system. However, if this is not the case, all paths that are shorter than $\pi_{d,s}$ will potentially improve the solution. For simplicity, however, we will add only the shortest path since it has the highest potential to improve the solution. If other paths are required, they will be added during the following iterations.

Similarly, only a sub-set of working paths was added to the initial system. However, Equation (4.125b) is more complex. Although it can be rewritten as

$$\sum_{\substack{s \in \mathbb{S} \\ p \cap s = \{\}}} (\pi_{d,s} - \sum_{e \in p} \sigma_{e,s}) \leq 0 \qquad \forall d \in \mathbb{D}, \forall p \in \mathbb{P}_d \tag{4.126}$$

it still does not have a structure suitable for standard graph algorithms.[10]

The pricing problem can itself be modeled as linear program. The idea of the approach is to reformulate constraint $p \cap s = \{\}$ of Equation (4.126) by using reduced cost vectors $\pi'_{d,s} \in \mathbb{R}_0^+$ and $\sigma'_{d,s,e} \in \mathbb{R}_0^+$ that are equal to $\pi_{d,s}$ and $\sigma_{d,s,e}$ in all positions except those at which the found path p is affected by failure s. There, the values have to be zero so that the overall sum of σ and π is not changed. With this, the maximization objective can be

[9] Finding shortest paths can be performed using Dijkstra's algorithm [Dij59] in polynomial time [Joh77] or even better by using appropriate data structures [AMO93].

[10] The optimization problem that we denoted *Variable Cost Shortest Path Problem* seems to be NP complete.

written according to Equation (4.127).

$$\text{maximize} \sum_{s \in \mathbb{S}} (\pi'_{d,s} - \sum_{e \in \mathbb{E}} \sigma'_{d,s,e}) \tag{4.127}$$

Path p is modeled using flow approach formulations and a Boolean flow variable f for each demand d on a physical edge e.

$\forall d \in \mathbb{D}, \forall n \in \mathbb{N}:$

$$n \text{ is source node of } d\text{:} \begin{cases} \sum_{e \in \text{incoming}(n)} f_{d,e} = 0 & (4.128a) \\ \sum_{e \in \text{outgoing}(n)} f_{d,e} = 1 & (4.128b) \end{cases}$$

$$n \text{ is target node of } d\text{:} \begin{cases} \sum_{e \in \text{incoming}(n)} f_{d,e} = 1 & (4.129a) \\ \sum_{e \in \text{outgoing}(n)} f_{d,e} = 0 & (4.129b) \end{cases}$$

$$\text{else:} \begin{cases} \sum_{e \in \text{incoming}(n)} f_{d,e} = \\ \sum_{e \in \text{outgoing}(n)} f_{d,e} & (4.130) \end{cases}$$

Following this, we introduce a Boolean variable X_s that will be one if the formed path is affected by a failure (Equation (4.131)) and zero otherwise (Equations (4.132) and (4.133)).

$$\sum_{\substack{e \in \mathbb{E} \\ e \in s}} f_{d,e} \leq X_s \cdot \text{Max}^D \qquad \forall d \in \mathbb{D}, \forall s \in \mathbb{S}, s \neq s_0 \tag{4.131}$$

$$\sum_{\substack{e \in \mathbb{E} \\ e \in s}} f_{d,e} \geq X_s \qquad \forall d \in \mathbb{D}, \forall s \in \mathbb{S}, s \neq s_0 \tag{4.132}$$

$$X_{d,s} = 0 \qquad \forall d \in \mathbb{D}, s = s_0 \tag{4.133}$$

The reduced cost vector $\pi'_{d,s} \in \mathbb{R}_0^+$ will be furthermore set to be zero if the path lies on a failure using Equation (4.134). Since variable $\pi'_{d,s}$ is included in the maximization objective, a restriction from above with Equation (4.135) to be $\pi_{d,s}$ otherwise is sufficient.

$$\pi'_{d,s} \leq (1 - X_{d,s}) \cdot \text{Max}^D \qquad \forall d \in \mathbb{D}, \forall s \in \mathbb{S} \tag{4.134}$$

$$\pi'_{d,s} \leq \pi_{d,s} \cdot \text{Max}^D \qquad \forall d \in \mathbb{D}, \forall s \in \mathbb{S} \tag{4.135}$$

Similarly, the reduced edge-cost vector $\sigma'_{d,s,e}$ will be forced to be zero using Equations (4.136) and (4.137), if the path does not use edge e ($f_{d,e} = 0$) or is affected by failure

s ($X_{d,s} = 1$). Furthermore, if the path is not affected by failure s ($X_{d,s} = 0$) and edge e is part of the path ($f_{d,e} = 1$), Equations (4.138) and (4.139) force $\sigma'_{d,s,e}$ to be $\sigma_{d,e}$.

$$\sigma'_{d,s,e} \leq f_{d,e} \cdot \mathrm{Max}^D \qquad \forall d \in \mathbb{D}, \forall s \in \mathbb{S}, \forall e \in \mathbb{E} \qquad (4.136)$$

$$\sigma'_{d,s,e} \leq (1 - X_{d,s}) \cdot \mathrm{Max}^D \qquad \forall d \in \mathbb{D}, \forall s \in \mathbb{S}, \forall e \in \mathbb{E} \qquad (4.137)$$

$$\sigma'_{d,s,e} \geq \sigma_{d,e} - (1 - f_{d,e}) \cdot \mathrm{Max}^D - X_{d,s} \cdot \mathrm{Max}^D \qquad \forall d \in \mathbb{D}, \forall s \in \mathbb{S}, \forall e \in \mathbb{E} \qquad (4.138)$$

$$\sigma'_{d,s,e} \leq \sigma_{d,e} \cdot \mathrm{Max}^D \qquad \forall d \in \mathbb{D}, \forall s \in \mathbb{S}, \forall e \in \mathbb{E} \qquad (4.139)$$

Thus, by performing an ILP optimization as sub-problem, paths can be found that potentially improve the solution. If the optimal objective of the sub-problem is negative, the primary problem will be optimal.

4.2.4.3 Column Generation for Global Restoration

Close inspection of the primal and dual system reveals that the differences of Equations (4.125b) and (4.125c) are caused by the fact, that working paths are valid for multiple network states. As long working paths are not affected by a failure, they are not rerouted. Relaxing this characteristic transforms the optimization problem from *protection* to *global restoration*. All working paths, even if not affected by the failure, can be rerouted. Thus, working and backup paths need not to be differentiated any longer. Variable $CP^D_{d,s,p}$ models the required amount of capacity for a demand d, along a path p in failure state s. The primal system consequently changes to

$$\text{minimize} \sum_{e \in \mathbb{E}} UCE^D_e \qquad (4.140a)$$

$$[\pi_{d,s}] \quad \sum_{p \in \mathbb{P}_{d,s}} CP^D_{d,s,p} \geq k^D_{d,s} \cdot D^D_d \qquad \forall d \in \mathbb{D}, \forall s \in \mathbb{S} \qquad (4.140b)$$

$$UCES^D_{e,s} = \sum_{d \in \mathbb{D}} \sum_{\substack{p \in \mathbb{P}_{d,s} \\ e \in p}} CP_{s,d,p} \qquad \forall e \in \mathbb{E}, \forall s \in \mathbb{S}, \qquad (4.140c)$$

$$[\sigma_{e,s}] \quad UCE^D_e \geq UCES^D_{e,s} \qquad \forall e \in \mathbb{E}, \forall s \in \mathbb{S} \qquad (4.140d)$$

$$[\tau_e] \quad UCE^D_e \leq C^D_e \qquad \forall e \in \mathbb{E} \qquad (4.140e)$$

$$CP^D_{d,s,p} \geq 0 \qquad \forall d \in \mathbb{D}, \forall s \in \mathbb{S}, \forall p \in \mathbb{P}_{d,s} \qquad (4.140f)$$

$$UCE^D_e \geq 0 \qquad \forall e \in \mathbb{E} \qquad (4.140g)$$

The schematic coefficient matrix of the restoration formulation illustrated in Table 4.14 shows both the differences to the initial LP as well as the insights into the structure of the coefficient matrix needed for the formulation of the dual LP.

Due to the possible restoration of paths in any network state, the variable $CP_{d,s,p}$ appears only in exactly one row in Equation (4.140b) and Equation (4.140d) in any network status s. The capacity constraint and the cost function remain unchanged.

Table 4.14: Schematic coefficient matrix for the relaxed primal LP

	$CP^D_{d,s,p}$										UCE^D_e		
$\pi_{s,d}$	$\oplus$				$\oplus$			$\oplus$					
		$\oplus$				$\oplus$	$\oplus$						
			$\oplus$	$\oplus$						$\oplus$			
$\sigma_{e,s}$	$\ominus$		$\ominus$			$\ominus$				$\ominus$	$\oplus$		
		$\ominus$	$\ominus$		$\ominus$		$\ominus$	$\ominus$		$\ominus$	$\oplus$	$\oplus$	
	$\ominus$			$\ominus$	$\ominus$		$\ominus$			$\ominus$			$\oplus$
τ_e											$\oplus$		
												$\oplus$	
													$\oplus$

The complete dual LP consequently reads as

$$\text{maximize } (\sum_{d\in\mathbb{D}}\sum_{s\in\mathbb{S}}(k^D_{d,s}\cdot D^D_d\cdot\pi_{d,s})+\sum_{e\in\mathbb{E}}(C^D_e\cdot\tau_e)). \tag{4.141}$$

$$[CP_{d,s,p}] \quad \pi_{d,s}-\sum_{e\in p}\sigma_{e,s}\leq 0, \quad \forall d\in\mathbb{D},\forall s\in\mathbb{S} \tag{4.142}$$

$$[UCE^D_e] \quad \sum_{s\in\mathbb{S}}\sigma_{e,s}-\tau_e\leq 1, \quad \forall e\in\mathbb{E} \tag{4.143}$$

$$\pi_{d,s}\geq 0 \qquad \forall d\in\mathbb{D},\forall s\in\mathbb{S} \tag{4.144}$$

$$\sigma_{e,s}\geq 0 \qquad \forall e\in\mathbb{E},\forall s\in\mathbb{S} \tag{4.145}$$

$$\tau_e\geq 0 \qquad \forall e\in\mathbb{E} \tag{4.146}$$

Solving the Pricing Problem:
All edge-capacity variables were added to the primal system from the beginning. Thus, Equation (4.143) will not be violated if any solution exist to the problem. Missing paths in the primal system that potentially improve the optimization will be obtained by checking if Equation (4.142) is violated by any missing dual constraint. Similarly to protection, a shortest path algorithm with $\sigma_{e,s}$ as link weights is sufficient to check the equation for violation and to create paths that can be added to the primal system. When adding the shortest path only, at most one path per failure pattern is added to the primal system in each pricing iteration.

4.3 Chapter Summary

In this chapter, we discussed network-optimization approaches for the optimization of resilient networks. While algorithmic and meta-heuristic approaches can achieve good solu-

tions in reasonable time, no information about the quality of the solution can be obtained. However, optimization approaches that are based on Integer Linear Programming are able to calculate mathematical optimal solutions. Lower bounds to the (unknown) best solution are provided during the optimization process. After discussing advantages and the general concept of linear programming, we presented complete formulations for resilient network planning for four path-based resilience mechanisms using the flow and as well as the path approach. We furthermore separated the models into building blocks in order to facilitate the integration of the models in planning tools. Additionally, we discussed the mathematical theory of duality in linear programming and presented a new approach that enhances the planning of resilient networks considerably by using the mathematical technique *Column Generation*. Hence, the proposed methods allow the optimal network planning for protection and restoration for four path-based resilience mechanisms.

Chapter 5

Evaluation of Resilience Mechanisms

In order to provide guidelines for the design of resilient networks, this chapter provides an evaluation of selected resilience mechanisms. It is organized as follows: Section 5.1 discusses the evaluation environment in more detail and lists evaluation criteria, considered resilience mechanisms, and the basic structure of the developed resilient network optimization program. Consequently, we analyze capacity optimization results in Section 5.2. In particular, we discuss effects of topology and path restrictions on the required capacity and compare the capacity requirements of the investigated resilience mechanisms. Following this, we analyze the effect of multipath routing on capacity requirements and discuss the necessity of multipath routing. In addition, we compare running times and memory consumptions of the optimization approaches that have been presented in Chapter 4. Section 5.3 analyzes the recovery time of OSPF rerouting, MPLS path protection, and restoration and presents novel recovery time formulas and simulation results. Finally, we discuss the complexity of path-based resilience mechanisms in Section 5.4.

5.1 Evaluation Environment

5.1.1 Evaluation Criteria

Capacity Requirements:
The resilience classification of Chapter 3 revealed that some resilience mechanisms might require less capacity to protect a network against single link failures compared to other mechanisms.[1] In order to confirm the theoretical deliberations and to quantify the achievable capacity gains, we will thus compare the minimum required capacity requirements of different resilience mechanisms with each other. For this, we will optimize example networks using the linear programming formulations of Chapter 4.

[1] As an example, a resilient network design using SE2EPP might use less capacity than a design with SLLPP (see Section 3.5.3).

Recovery Time:
The recovery time discussion of Chapter 2 showed that some applications require very fast recovery times in sub-second range. Thus, in order to find resilience mechanisms that are suited for these applications analysis of recovery times has to be performed. Therefore, we will evaluate the recovery time from a theoretical point of view and will support the findings by analyzing simulation results.

Complexity:
Finally, the complexity of a resilience mechanism reflects the operation costs and threats caused by misconfiguration. Thus, as third important evaluation criteria, we will analyze the complexity to handle working and backup paths.

5.1.2 Considered Resilience Mechanisms

The resilience classification framework of Chapter 3 showed that the backup structures are one of the key characteristics of resilience mechanisms. The structure and the location of the reacting entities relatively to the occurred failure influence the recovery time and the possibilities to share capacity. Protection mechanisms, i.e. mechanisms that use pre-calculated, pre-configured, and pre-established backup resources, are furthermore able to react very fast on considered failure patterns. Restoration approaches, in contrast to that, have slower recovery times but are able to react dynamically to the occurrence of a failure. An overall lower bound on capacity requirements for any resilience mechanism will be achieved if all paths can be rerouted in case of a failure (*Global Restoration*). Since any backup structure can be combined with a construction of corresponding backup paths. We will focus on path-based protection mechanisms and global restoration in the following. In particular, we will investigate the following five resilience mechanisms:

- *Shared End-to-end Path Protection* (SE2EPP),
- *Shared Local-to-egress Path Protection* (SL2EPP),
- *Shared Local Link Path Protection* (SLLPP),
- *Shared Regional Path Protection* (SRPP).
- *Global Restoration* (GR)

5.1.3 Resilient Network Optimization Program

During this thesis a resilient network optimization and configuration program *Resilient Network* as well as a generic multi-layer graph library *GRAPH* were developed. In the following sub-section, we will briefly describe the main components of the software that will be used to perform the evaluation.

5.1.3.1 GRAPH Library

Chapter 4 demonstrated that telecommunication structures and planning problems can easily be modeled using graph structures. In fact, graph-theory provides a large fund of theory and algorithms that are suited to model and optimize telecommunication networks. However, due to the fast evolution of communication technology and the interaction between different technology layers, a detailed modeling of characteristics and a coordinated optimization between layers is required. Although there exist a number of specialized commercial and open-source planning and configuration tools that were developed for example by

- network providers (e.g. Deutsche Telekom),
- telecommunication consulting companies (e.g. Detecon),
- software developing companies (e.g. Wandl, VPISystems, OPNET, ESG),
- vendors (e.g. Cisco, Siemens), or
- research institutions and small spin-offs,

there exists no generic planning framework that can easily be adapted to different technologies and application fields (e.g. mobile and fixed networks). Therefore, a generic multi-layer graph library was developed. *GRAPH* provides well-tested software components that can be combined to facilitate and accelerate the modeling, simulation, and optimization of multi-layer telecommunication network structures.

The main building blocks of the software-library, which became part of a company [**GF06**] by the developers Claus G. Gruber and Jochen Frings in 2005, are depicted in Figure 5.1.

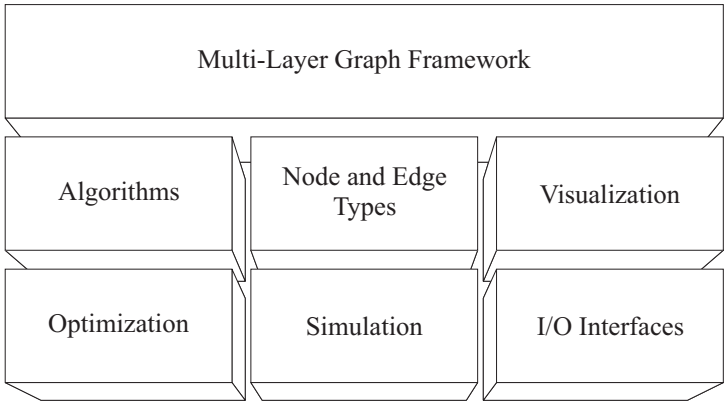


Figure 5.1: Building blocks of the *GRAPH* library.

Multi-Layer Graph Framework:
The multi-layer graph framework provides a generic structure for the modeling of networks in multiple layers. Right from the beginning, the structure was developed to enable the

application in different technologies and application areas. Therefore, *GRAPH* provides well-tested generic multilayer graph-, node-, and edge-structures that facilitate the modeling of technology characteristics and the development of special-purpose applications. It has been shown that several thousand graph-instances can be handled in parallel. Additionally, due the full object oriented approach, a multi-user developer environment is provided. The reuse of thoroughly tested algorithms furthermore allows a faster development of new applications.

Special Container Types:
Next to generic node- and edge-types, special objects such as OSPF routers are part of the *GRAPH* library. The provision of additional container classes (e.g. Lists, Maps, Queues, Paths, and Cycles) facilitates the modeling of network structures even more.

Algorithms:
GRAPH provides a number of algorithms that can be applied to any graph structure. Currently supported algorithms range from shortest path based algorithms (e.g. Dijktra, Modified Dijkstra, BFS, k-shortest-path algorithm) to disjoint path based algorithms (e.g. k-disjoint-shortest path algorithm) and to path-, tree- and cylce-search algorithms (e.g. all paths, all cycles, minimum spanning tree). Furthermore, reliability related algorithms such as the calculation of terminal pair availability in mesh networks (based on *Ordered Binary Decision Diagrams* (OBDD)) and minimum cut algorithms are provided.

Simulation:
In addition, an event-based simulation suite is included in the library. Parallel, quasi real-time or event-based simulations are supported by using Posix threads. A network protocol simulation and analysis can thus easily be implemented in applications that are based on *GRAPH*.

Optimization:
Furthermore, *GRAPH* provides a generic meta-heuristic framework as well as random number generators to facilitate the development of heuristic approaches. Currently supported heuristics are Simulated Annealing, Genetic Algorithm, and Tabu Search. In addition, special container classes facilitate the interaction to the ILP optimization suite *CPLEX* [Ilo06].

I/O Interfaces:
To load, save, and modify complex data structures, the library provides several I/O interfaces such as the graph modeling language (GML) [Him97] and the generic Markup Language XML [BPSM+06].

Graphical User Interface:
Finally, a graphical user interface (GUI) is provided. Similar to the multi-layer graph

framework, the GUI has been developed to facilitate its extension for the development of special purpose applications.

5.1.3.2 Optimization Program *Resilient Network*

Figure 5.2 depicts the main building blocks of the resilient network optimization program *Resilient Network* that is based on the generic multi-layer graph library *GRAPH*. In order to decouple the complex optimization problem, four related graph structures are used in the program. Figure 5.3 illustrates the graph layers and their relations: Demand traffic can be routed along different working paths which are themselves protected by resilience paths. However, both working and resilience paths are routed on a physical topology.

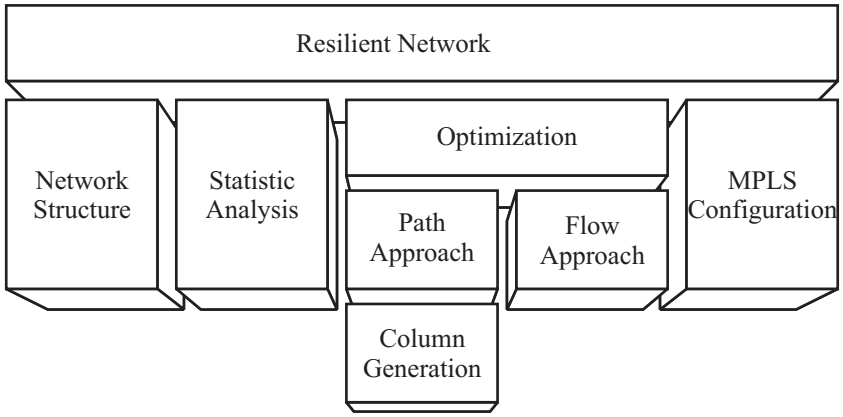


Figure 5.2: Building blocks of the optimization program *Resilient Network*.

Optimization:
The ILP equations of Chapter 4 were integrated in *Resilient Network* and modeled using the C++ interface *Concert* of the MIP optimization program *CPLEX* [Ilo06]. Due to the grouping of equations into building blocks and the object-oriented approach of the multi-layer graph structure, the implementation of the different ILPs has been facilitated considerably: Individual building blocks could be implemented into different independent classes. Thus, a combination of class instances by the main control is sufficient to formulate the individual resilience mechanisms.

Automatic Configuration:
Furthermore, an MPLS configuration module was developed. The module offers an interface between a solution of working and backup paths and router configuration. MPLS switches of a network can thus be configured automatically by the program using SNMP, telnet, or SSH-based approaches. The configuration module uses an internal generic router structure. With this, it is possible to extend the program in order to support different router types. A reference implementation for the character-based interface of Cisco IOS version 12.0 [Cis06] was created.

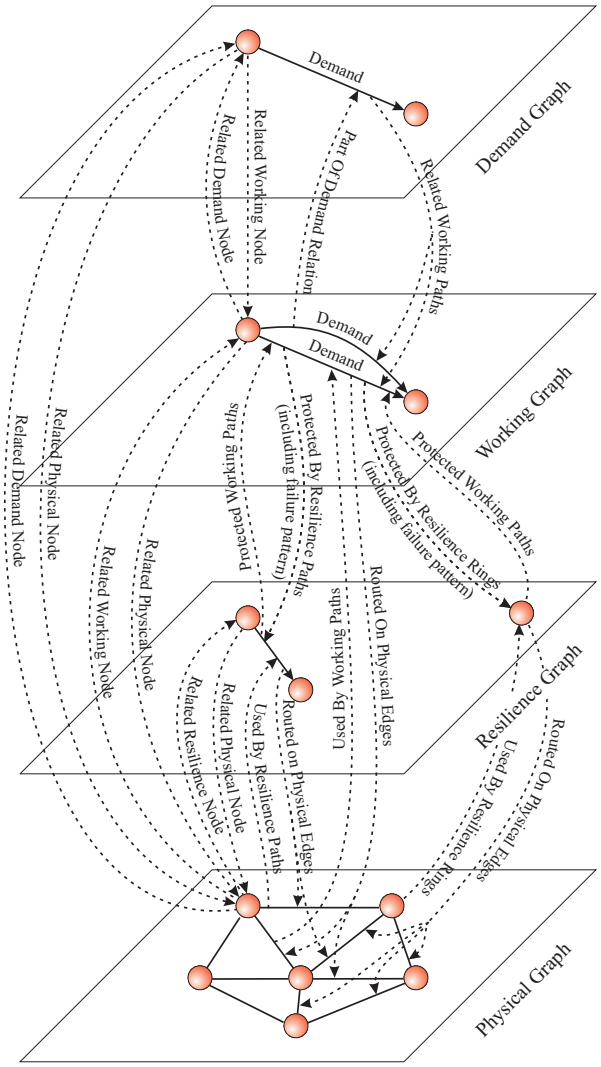


Figure 5.3: Four graph layers of the optimization program *Resilient Network*.

5.2 Capacity Optimization Results

In this section we present resilient network optimization results that were obtained by the developed network optimization program *Resilient Network* using the ILP solver ILOG CPLEX in version 9.03 [Ilo06].

5.2.1 Capacity Requirements of Resilience Mechanisms

The comparison of resilience mechanisms of Chapter 3 already provided a theoretical classification of the investigated resilience mechanisms with respect to the required capacity. In the following, we will investigate the absolute differences between the approaches based on numerical results. Table 5.1 shows the required capacity for a protection against single link

failures for three example networks[2]. Two versions of shared regional protection were used: SRPP (2-0/0-2) and SRPP (1-0/0-1). The numbers in brackets indicate the hop distance of the detour points before and after the failure (StartMin-EndMin/StartMax-EndMax) as illustrated in Figure 4.10 of Chapter 4.

Table 5.1: Required capacity for different resilience mechanisms.

Reference Network	SLLPP	SL2EPP	SRPP (1-0/0-1)	SRPP (2-0/0-2)	SE2EPP	GR
Germany	9897.32	9689.61	9658.18	9658.18	9658.18	9658.18
Europe	6220.5	5656.5	5436.11	5342.55	5322.22	5315
USA	9602.75	8613.46	8258.33	8092.29	8084.33	7758

In all example networks, the biggest amount of capacity will be required for a protection with SLLPP. Less capacity will be required for a protection with SL2EPP. Even further reductions will be achieved when using regional protection mechanisms SRPP (1-0/0-1) and SRPP (2-0/0-2). Finally, SE2EPP will require the smallest amount of capacity of the considered protection mechanisms. The overall lowest amount of capacity will be required for *Global Restoration*.

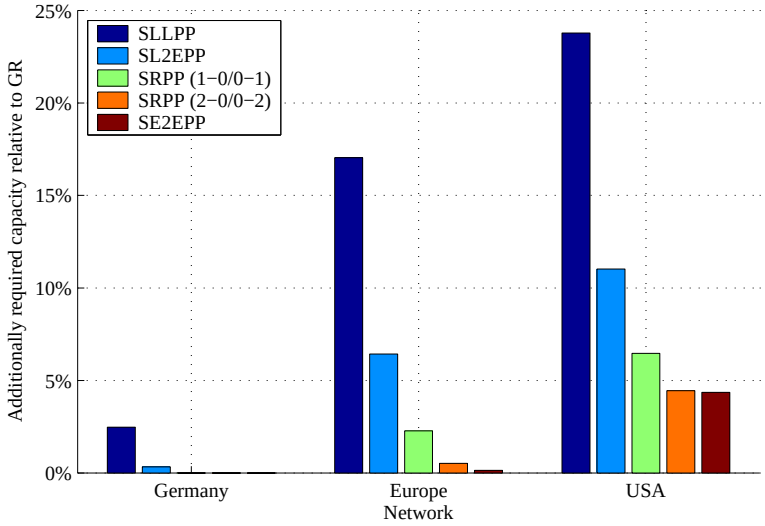


Figure 5.4: Additionally required capacity relative to *Global Restoration* for different resilience mechanisms.

Figure 5.4 depicts the differences of the capacity requirements relatively to global restoration. While the differences between the resilience mechanisms are quite small for network Germany (2.5%), around 17.0% and 23.8% additional capacity will be required

[2]Details of the example networks (NOBEL) can be found at http://sndlib.zib.de [SND06]. Due to memory restrictions, the number of possible paths was limited.

when protecting the networks Europe and USA with SLLPP. Still, around 6.4% and 11.0% more capacity will be required when using SL2EPP compared to GR. However, while shared regional protection and end-to-end protection are quite efficient in the networks Germany and Europe, 4% additional capacity will be required in the sparsely meshed network USA.

The numerical values confirmed the theoretical deliberations of Chapter 3. However, absolute differences of capacity requirements between the considered resilience mechanisms are dependent on network topology and demand values.

5.2.2 Capacity Requirements Dependent on Nodal Degree

The resilient network planning cycle of Chapter 2 highlighted the dependency of the chosen network topology from routing and equipment placement. An addition of links will increase the number of routes from which optimal working and backup path constellation can be chosen. Thus, in mesh networks, backup routes can be aligned more efficiently in order to share capacity.

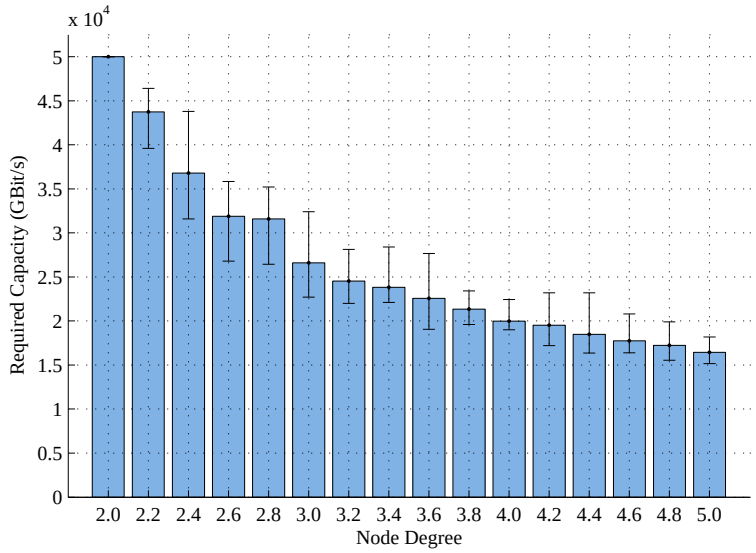


Figure 5.5: Minimal required capacity for networks with ten nodes and different node degrees that are protected against single link failures using SE2EPP. The number of candidate paths was limited for node degrees ≥ 3.6.

Figure 5.5 illustrates the minimally required amount of capacity for case-study networks with ten nodes and different node degrees using resilience mechanism SE2EPP to survive single link failures. A demand value of 100Gbit/s was routed between each node-pair while an unlimited number of path splits was allowed. Figure 5.5 depicts minimum, average, and maximum capacity requirements that were obtained by optimizing ten different network topologies per node degree.

The required capacity reduces when using networks with higher node degrees. The addition of links will increase the number of paths and will thus increase the solution space

for the optimization. However, a considerable capacity reduction of almost 50% compared to a ring network (degree 2.0) could already be achieved with a node degree of around 3.0.

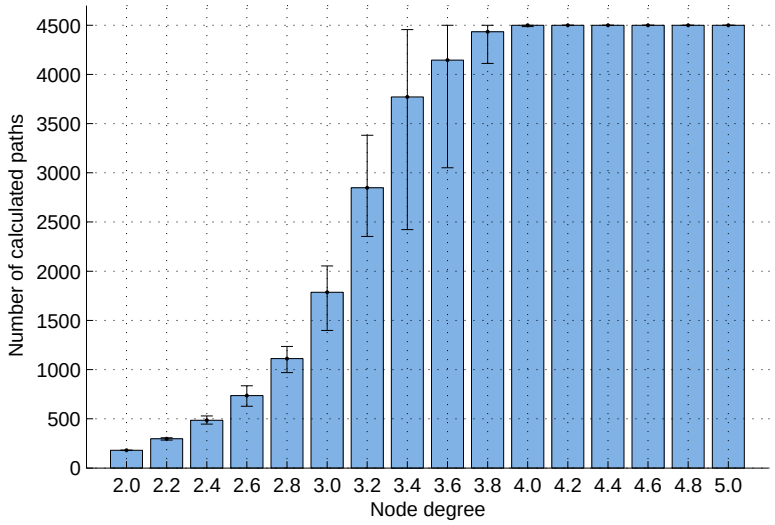


Figure 5.6: Number of calculated working paths (minimum/average/maximum) for the ten-node example networks using the path approach. The number is restricted for networks with node degrees ≥ 3.6.

Figure 5.6 and 5.7 depict the number of considered working and backup paths that were used in the study. The average number of working and backup paths already reached 3771.8 and 745651.8 for the 10 example networks with a node degree of 3.4. Since computer memory (RAM) was restricted to 8 GByte, the number of considered paths had to be limited for networks with a node degree greater or equal to 3.6. For these networks 50 shortest (hop-count) working paths per demand and at most 50 shortest resilience paths per working path and failure were calculated.[3] However, when considering the large amount of available paths it is not surprising that good solutions could already be achieved with node degrees around 3.0.

[3]Please note, the number of hops per working path decreases with increasing node degree. Thus, the number of failures that affect a working path and with it the number of resilience paths reduces slightly with increasing node degree.

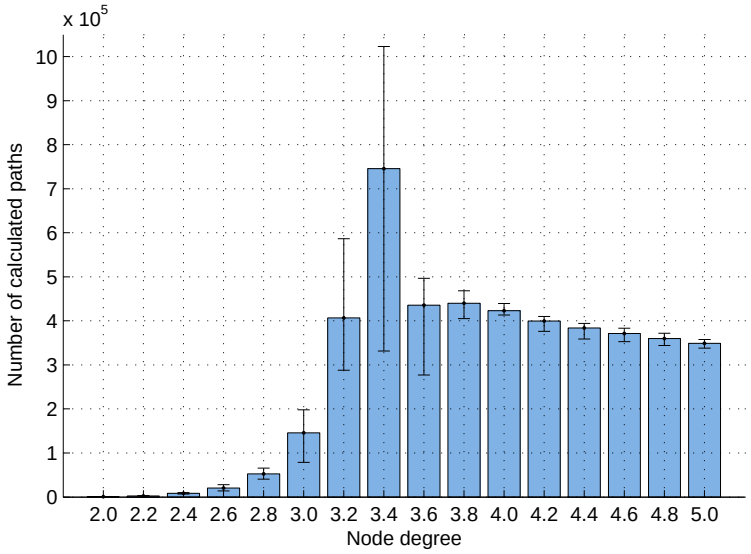


Figure 5.7: Number of calculated backup paths (minimum/average/maximum) for the ten-node example networks using the path approach. The number is restricted for networks with node degrees ≥ 3.6.

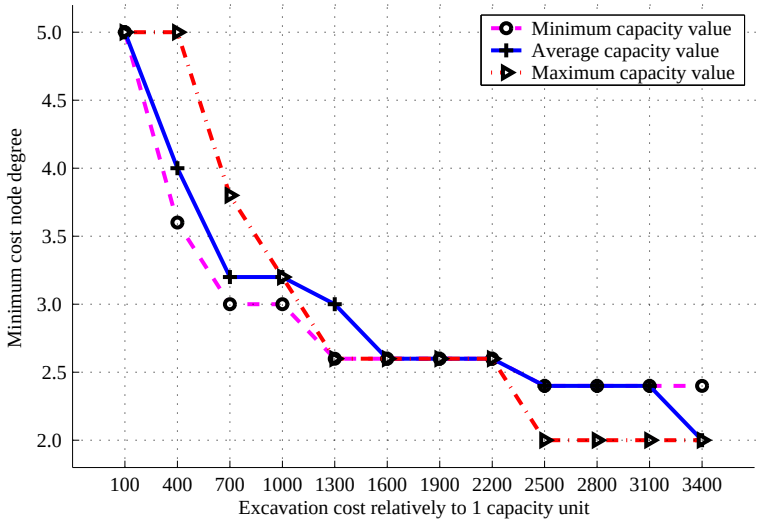


Figure 5.8: Network degree with the minimum total cost of capacity and excavation. The excavation costs are relative to one capacity unit.

When we take excavation costs into account, networks with a large number of edges will become quite expensive. Although capacity reductions can still be achieved by adding edges, excavation costs for providing higher node degrees are immense (Figure 5.8). The excavation costs are varied in relation to the cost of one capacity unit. The curves for the minimum, average and maximum capacity requirements are shifted in x-axis. For reasonable excavation costs networks with a node degree of about 2.4 to 3.0 are preferred.

5.2.3 Length of Optimal Working and Backup Paths

Chapter 4 outlined the possibility to use a sub-set of paths in order to speed-up optimization time or to allow a computation of larger networks using the path approach. However, the question remains, which paths should be selected for the optimization process. Thus, in the following, we will analyze the length-distribution of optimized solutions and will inspect the length of the chosen paths.

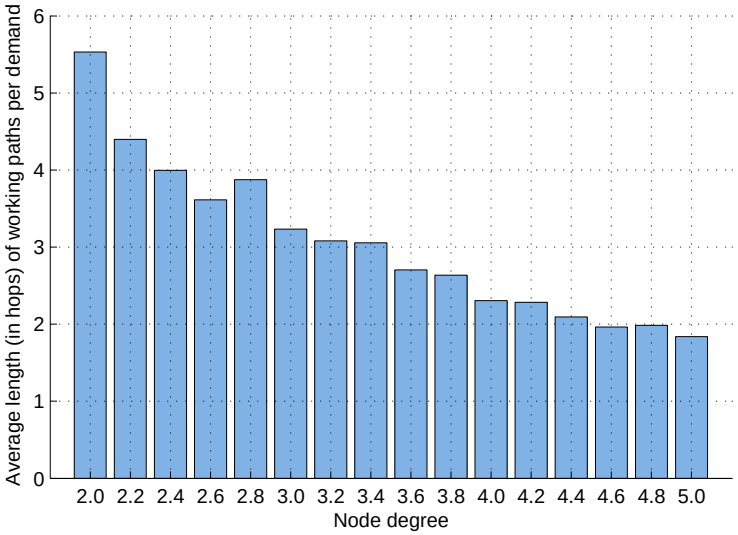


Figure 5.9: Average length (in hops) of demand paths for the path approach.

Figures 5.9 to 5.10 show the average length and the length-distribution of chosen working paths for the optimization of the ten-node networks for node degrees 2.0 to 3.4. When more paths are available, the chosen working paths tend to be very short (between 1 and 2 hops). This is due to the fact, that capacity on working paths cannot be shared and longer paths (in terms of hops) directly increase the overall used capacity. Thus, for an optimal solution shorter working paths are preferred.[4]

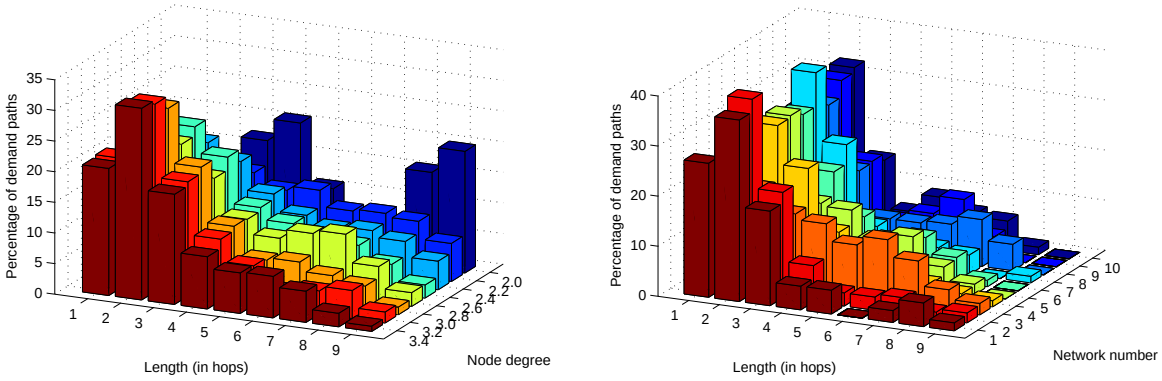
In contrast to working paths however, capacity on backup paths can be shared. Thus, capacity that is already required to protect other working paths can be reused without adding costs to the objective function. Thus, a mixture of path lengths is predominant in Figures 5.11 and 5.12 for backup paths.[5]

5.2.4 Requirement for Multipath Routing

In the following, we will analyze the effect of multipath routing on capacity requirements and discuss the necessity of multipath. Figures 5.13 and 5.14 illustrate the number of

[4]Note, only a small number of short paths are available in sparsely meshed networks. Thus, there is no tendency towards short paths for the networks with small node degrees in Figure 5.10(a).

[5]Note, due to the path restriction for networks with node degrees greater or equal 3.6, long paths are not included in the optimization. Thus, the average path length decreases in Figure 5.11.



(a) For networks with varying node degrees.

(b) For ten different networks with node degree 3.4.

Figure 5.10: Probability distribution function of demand path lengths (in hops) for the path approach.

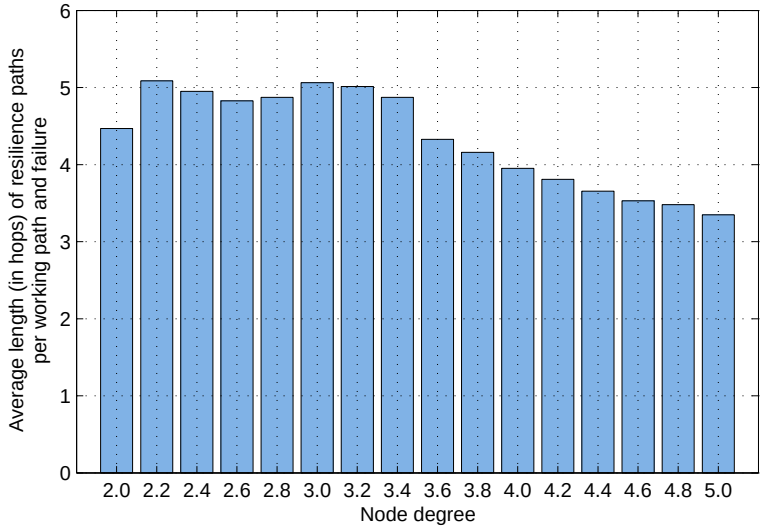
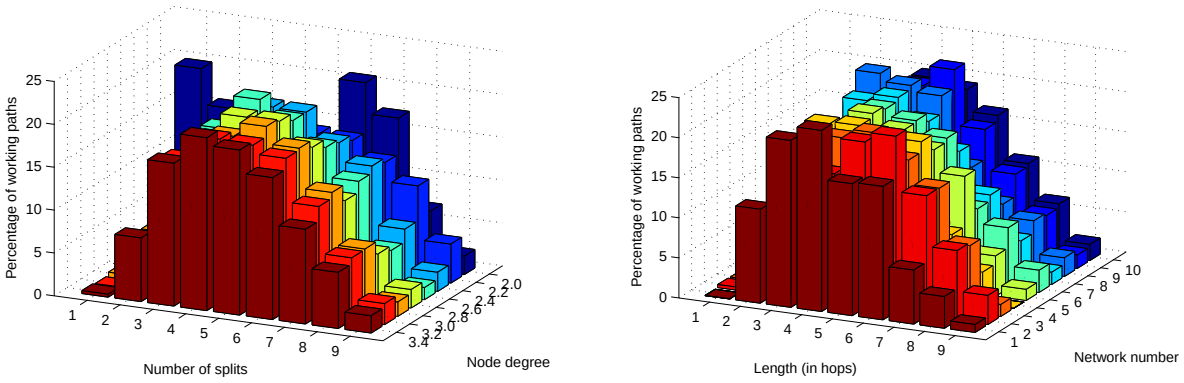


Figure 5.11: Average length (in hops) of resilience paths for the path approach.

working paths per demand for the capacity minimization of the ten-node example networks using SE2EPP for a protection against single link failures. Although the number of splits was not restricted by the optimization program, the average number of working paths per demand is only around 1.7 for all example networks with a node degree greater or equal to 2.6 and even less for networks with smaller node degrees.[6]

The probability distribution function that is shown in Figure 5.14 additionally reveals that around 60% of all demands are not split while around 30% of all demands are split

[6]Note a split of a demand is also possible in ring networks. Thus, the average number of splits can be greater than 1 as depicted in Figure 5.13



(a) For networks with varying node degrees.

(b) For ten different networks with node degree 3.4.

Figure 5.12: Probability distribution function of resilience path lengths (in hops) for the path approach.

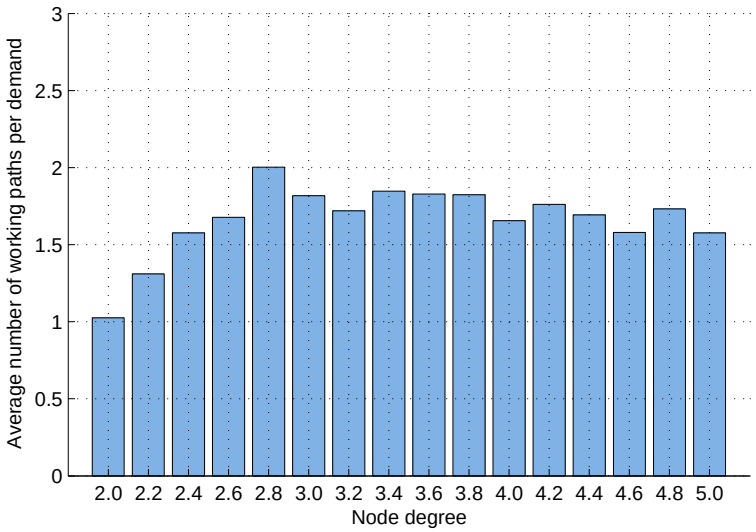


Figure 5.13: Average number of demand path splits for the path approach.

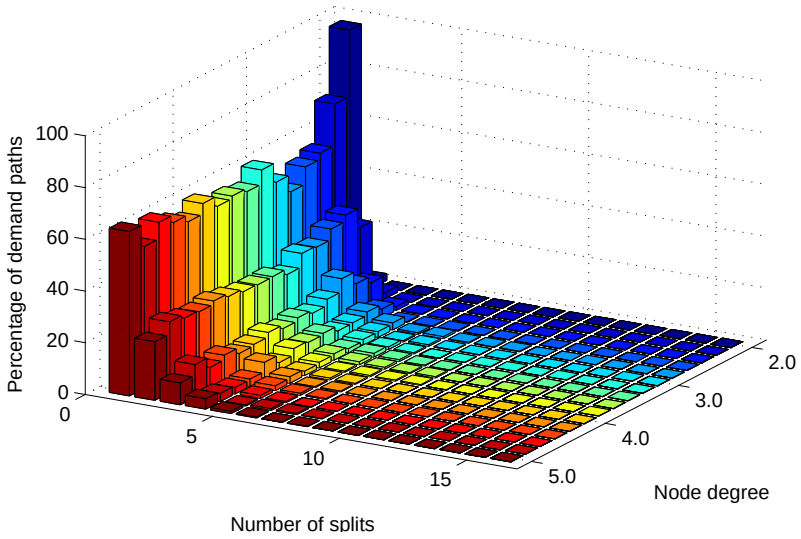


Figure 5.14: Probability distribution function of the number of demand path splits.

only once in the optimal networks designs (with an exception of ring networks). Very rarely a demand is split into several working paths (15 was the highest observed value for all example networks).

Similarly, Figures 5.15 and 5.16 depict the number of backup paths per working path and failure. In contrast to demand splits, the average number of working path splits increases with the node degree. This is because backup capacity can be shared more efficiently, if equal amounts of capacities are shared. Thus, working paths are split into small parts with similar capacity. However, the average number of splits stays between 1.0 for rings and increases to only 2.5 for the example networks with a node degree of 5.0 in the example. The probability distribution function of the number of working path splits additionally shows an increasing distribution to using more paths with increased node degree. However, even for the networks with node degree of 5.0 only 25% of all paths were split more than three times in the optimal network designs.

The above numbers indicate that only a limited number of path splits is required to achieve a cost-optimal network design. In order to further investigate the effects of multipath routing on the required capacity, in the following, we will restrict the number of splits already during the optimization process. Table 5.2 shows the minimal required capacity for network Germany dependent on multipath restrictions. The demands are protected with SE2EPP against single link failures.

The additionally required capacity relative to the minimum required capacity without multipath restriction is depicted in Figure 5.17. With single path routing, 14.4% more capacity is required compared to the optimal solution with unlimited multipath routing. However, the additional required capacity reduces further to 5.9% when allowing one split (two working and two resilience paths per failure). Only 1.5% additional capacity is required when allowing three splits of demands and working paths in case of a failure. Thus,

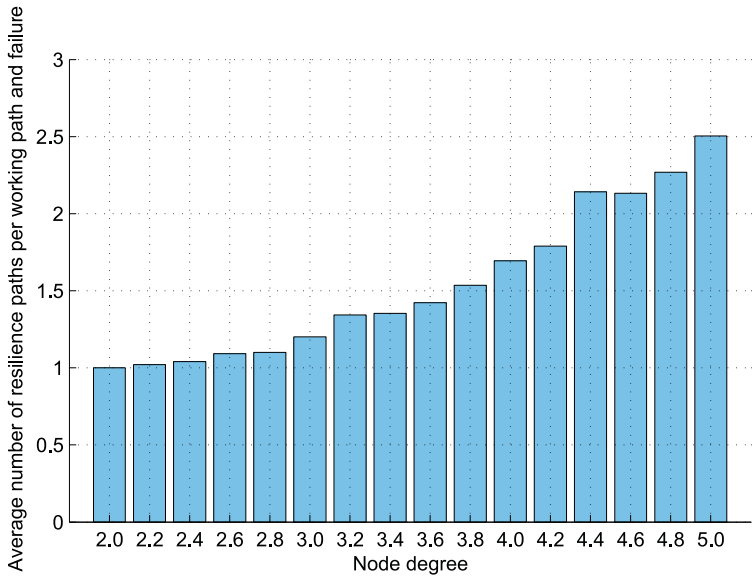


Figure 5.15: Average number of working path splits for each failure pattern using the path approach.

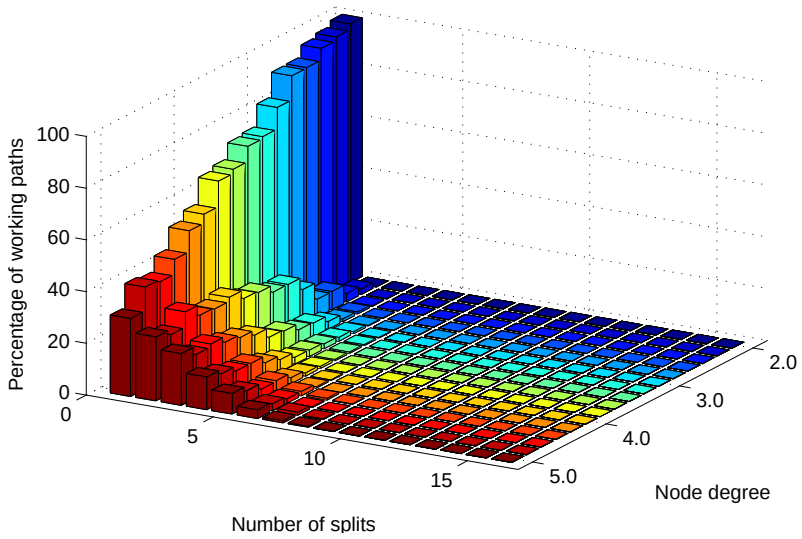


Figure 5.16: Probability distribution function of the number of working path splits.

Table 5.2: Required capacity for SE2EPP (single link protection) for the German network dependent on multipath restrictions.

Number of working paths per demand	Number of backup paths per working path and failure			
	1	max. 2	max. 3	unlimited
1	11047.2	10841.5	10383.29	9697.35
max. 2	10890.2	10229.7	9989.22	9658.18
max. 3	10223.4	10010.5	9802.37	9658.18
unlimited	n.a.	n.a.	n.a.	9658.18

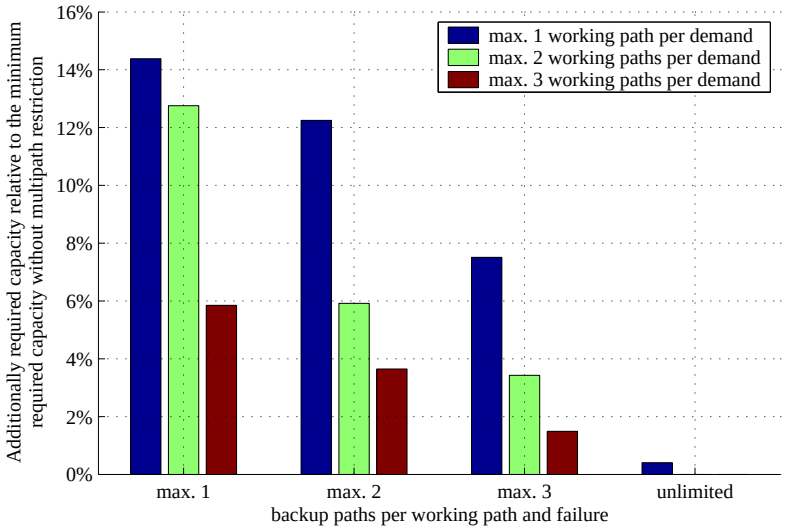


Figure 5.17: Additionally required capacity relative to the minimum required capacity without multipath restriction.

multipath restriction has certainly an influence on the required capacity. However, the difference to the optimal value without path restrictions is small in the example network. Thus, the possibility to use a splitting of paths helps to reduce the overall required capacity. Especially, the splitting of working paths in order to have similar batches of detour capacity is beneficial. However, as indicated by the simulation results, only little benefit can be gained from a fine granular splitting when routes as well as distribution ratios are optimized.

5.2.5 Comparison of Optimization Approaches

The mathematical technique *Column Generation* was discussed in Chapter 4. Paths that potentially improve the solution are added iteratively during the solution process. In the following, we compare the classical path approach with *Column Generation*.

Figure 5.18 and 5.19 illustrate the maximum required memory and the computation times for the calculation of an optimal solution for *Global Restoration* using different test

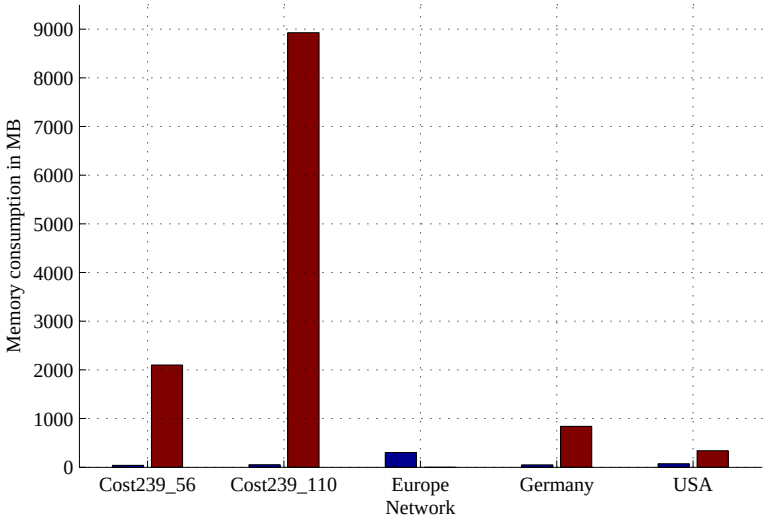


Figure 5.18: Memory usage for an optimal calculation of global restoration.

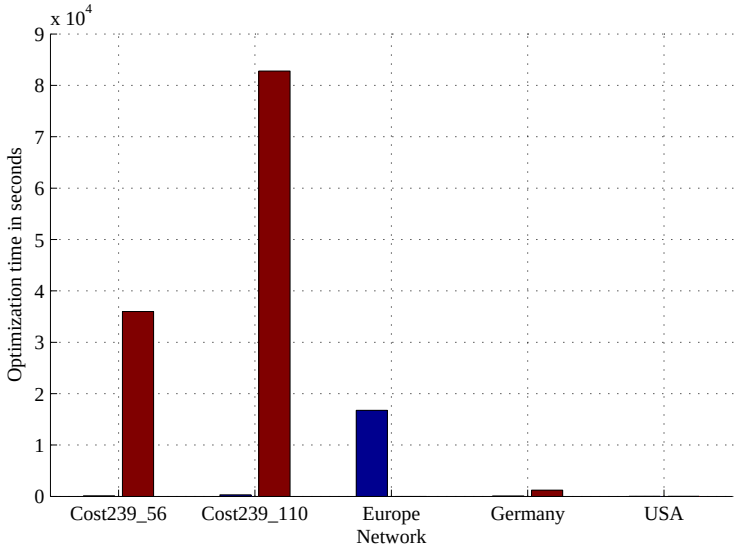


Figure 5.19: Time for an optimal calculation of *Global Restoration*.

networks. As can clearly be seen, the optimization with *Column Generation* was able to reduce the required amount of memory by at least one order of magnitude and even more for larger networks. Similarly, immense reductions in calculation times are possible with *Column Generation*. A reduction in calculation time from almost a day (23 hours) to 5 minutes could be achieved for the network Cost239. Even more, an optimal result could be obtained for the European network in around 4.5 hours whereas it was not possible to calculate the solution with the path approach. Thus, these examples reveal the benefits of *Column Generation*: It enables the calculation of larger networks and reduces memory consumption and required calculation time significantly.

5.2.6 Summary

In this section, we performed resilient network optimizations of different case-study networks. The comparison of capacity requirements of the five path-based resilience mechanisms confirmed the classification of Chapter 3. However, the differences between the resilience mechanisms seem to be small in highly meshed networks. In order to quantify the differences in specific networks, optimizations approaches of Chapter 4 can be applied.

Capacity Requirements:
$$\text{GR} \leq \text{SE2EPP} \leq \text{SRPP(2-0/0-2)} \leq \text{SRPP(1-0/0-1)} \leq \text{SLLPP}$$

As already mentioned in Chapter 4 the currently best lower bound of the overall required capacity used for protection purposes in relation to the required working capacity is dependent on the node degree d of the network: $\frac{1}{d-1}$. Highly meshed networks might therefore reduce capacity requirements of resilience mechanisms significantly. The case-study optimization revealed capacity reductions of about 100% with a node degree of 3.0 compared to a pure ring network (degree 2.0). However, when excavation costs are taken into account networks with high node degrees will become very expensive.

Furthermore, an analysis of path-lengths for optimal resilient network constellations indicated that short working paths can reduce the overall capacity requirements. In contrast to that however, a mixture of all path length were selected for backup purposes. If capacity can be shared efficiently, no extra cost will be applied for longer backup paths. Therefore, short working paths and a mixture of short and long paths for backup purposes should be used in optimization algorithms or pre-calculated for path approach optimizations with a limited number of candidate paths.

The investigation of multipath splits furthermore revealed that multipath routing can reduce the capacity requirements. Especially the split of working paths in order to have similar batches of detour capacity is beneficial. However, only small capacity reductions can be gained by more fine granular splitting.

Finally, we performed network optimization calculations and compared the memory consumption and running times of path approach optimizations with that of *Column Generation*. In the example networks, memory requirements could be reduced by at least one order of magnitude. Furthermore, calculation times could be reduced significantly using the proposed optimization approach. With this technique, even large telecommunication networks can be planned in an optimal fashion.

5.3 Recovery Time Analysis

Another important characteristic of a resilience mechanism is its recovery time, i.e. the perceived overall outage time caused by a failure. As discussed in Section 2.3, recovery times should be short in order to limit the amount of lost data and to reduce the amount of control plane activity. Thus, understanding and accelerating the recovery time of a resilience mechanism is important both for network providers and equipment vendors [AJY00, BJ01, SG01].

Although OSPF and MPLS are deployed since several years, little is known about their recovery time behavior. Especially the recovery time of OSPF is said to be long lasting and in the range of tens of seconds [DR00b, SCK+03, GRWC03, ICB+04, Cho05]. Actually, in the late 1980s, the time in which OSPF was invented - there were no strict Quality of Service requirements and achievable bitrates of network links were rather small. Thus, at that time, the demand for a fast recovery time was limited.

Since network providers have the tendency not to publish information about their networks, only a few real-time measurements or simulations of OSPF recovery times are available [AJY00, SG01, BJ01, GRWC03, FFEB05]. Thus, today, OSPF is often regarded as a slow resilience mechanism.

To ultimately dispel doubts about fast recovering resilience mechanisms we will therefore analyze the rerouting behavior of OSPF in detail. After introducing a refined recovery time model in Section 5.3.1, Section 5.3.2 will present a theoretical analysis of OSPF, summarize proposals for a new version of OSPF to enhance its recovery time, and discuss recovery time simulation results for the enhanced OSPF protocol. Finally, we will present theoretical analysis of different MPLS protection and restoration mechanisms in Section 5.3.3.

5.3.1 Recovery Time Model

Several generic recovery time models exist in the literature that separate the recovery time into different time segments (e.g. RFC 3469 [VSFH03] and ITU-T I.630 [IT99]). Figure 5.20 depicts the recovery model of RFC 3469. There, recovery time is defined as "*the time required for a recovery path to be activated (and traffic flowing after) a fault. Recovery Time is the sum of the Fault Detection Time, Hold-off Time, Notification Time, Recovery Operation Time, and the Traffic Restoration Time*".

The recovery model assumes a linear dependency of time segments. However, while this model is applicable for resilience mechanisms in which one instance is reacting, resilience mechanisms that use distributed reactions of backup routes cannot be modeled adequately. In OSPF for example, all routers are informed about a failure and react in parallel to each other. Thus, multiple reactions are performed. Therefore, for the analysis of OSPF rerouting we will use a refined recovery model that is depicted in Figure 5.21. The new recovery time model defines seven time sequences that are not linear dependent on each other, i.e. the overall recovery time is not equal to the sum of the individual time segments. Table 5.3 summarized the proposed time segments that we will discuss in the following:

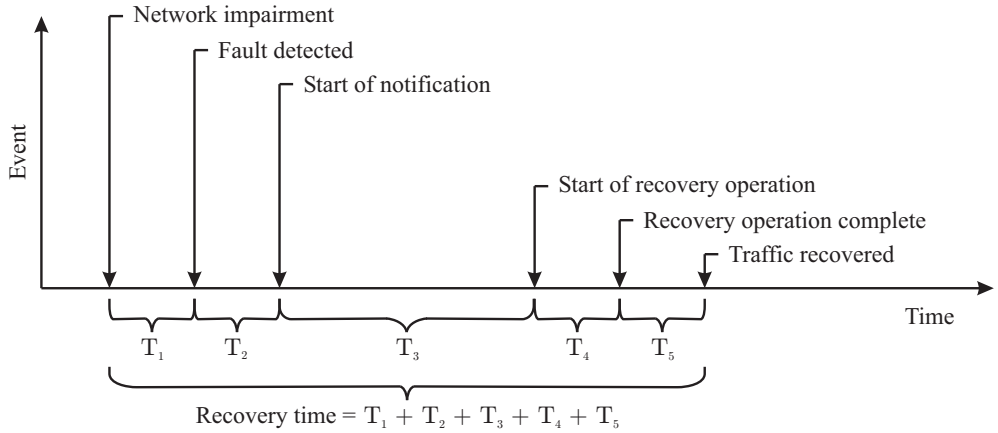


Figure 5.20: Recovery time model based on RFC 3460 [VSFH03]. T_1 = Fault detection time, T_2 = Fault hold-off time, T_3 = Fault notification time, T_4 = Recovery operation time, T_5 = Traffic recovery time.

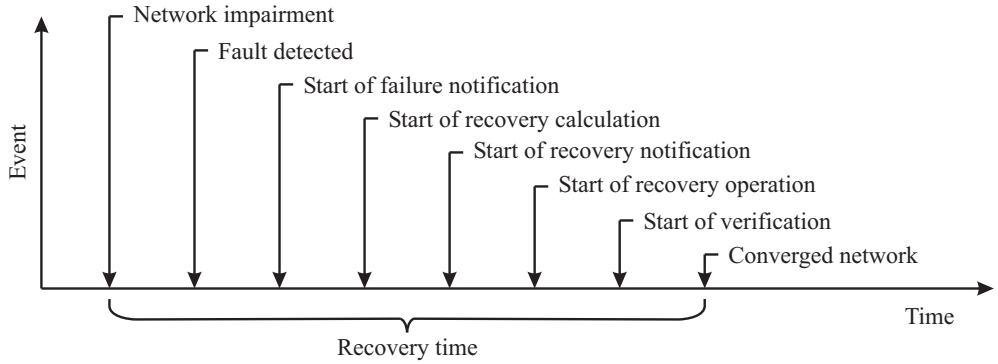


Figure 5.21: Proposed new recovery time model.

Before resilience mechanisms can be initiated, a network element failure has to be detected. The first time segment, time T_{Detect}, therefore models the time between the occurrence of a failure and its detection and localization by an appropriate failure detection mechanism. Following this, recovery mechanisms have to be initiated. However, it is sometimes reasonable to delay the reaction of a mechanism and to wait for an appropriate reaction in other layers. Thus, only if failures are still persistent after hold-off time $T_{HoldOff}$, a reaction has to be performed and appropriate decision entities have to be informed accordingly. The required time to inform these elements and the time to decide about and calculate the appropriate recovery measure at the notified entities are represented by times $T_{NotifyCalc}$ and $T_{Calculation}$, respectively. Following this, reacting entities have to be notified about the recovery measure (time $T_{NotifyReact}$) and appropriate recovery operations have to be performed (time $T_{Reaction}$). Finally, time T_{VSD} is required to verify the recovery operation, synchronization of signals and/or detectors and the additional delay caused by longer backup paths (compared to the failure-free path).

Table 5.3: Recovery time segments.

T_{Detect}	Failure detection and localization time	Time required to detect and locate a network element failure.
$T_{HoldOff}$	Hold-off time	Configurable time between the detection of a failure and the start of the failure notification. This time may be zero.
$T_{NotifyCalc}$	Notification time 1	Time to inform the calculating entities about the fault.
$T_{Calculation}$	Calculation time	Time to calculate or decide appropriate recovery measure.
$T_{NotifyReact}$	Notification time 2	Time to inform the reacting entities about the recovery measure.
$T_{Reaction}$	Recovery operation time	Time to perform the appropriate recovery operations. This may include message exchanges between the reacting entities.
T_{VSP}	Additional delay time	Additional time due to verification, synchronization and additional detour propagation times on the backup path.

Recovery Time Definition:
For the recovery time analysis of a distributed protocol, two times are of importance: The recovery time of the routes (T_{RTR}) and the convergence time of the protocol (T_{CTP}), i.e. control-plane activities to perform the resilience operation. We will define the two times as follows:

Definition of the recovery time of the routes:

The time between a failure of a network element and the last change of a network's router forwarding information base (FIB) caused by the occurrence of the network element failure. This time includes the detection of the failure, propagation of the failure information in the network, recalculation of routes, and the configuration of the forwarding information bases of all routers in the network.

Definition of the convergence time of the protocol:

The time between a failure of a network element and the last processing of a topology update message that was caused by the occurrence of a network element failure. This time includes the time T_{RTR} and the processing of topology update messages that do not have any effect on a forwarding information base.[7]

[7]RFC 3469 [VSFH03] defines the *Network Route Convergence Time* as the time taken for the network routing protocols to converge and for the network to reach a stable state.

5.3.2 Recovery Time of OSPF

5.3.2.1 Theoretical Analysis

In the following, we will analyze the recovery time of OSPF by modeling all components of the recovery time segments and clearly define their interdependence. We furthermore present equation systems for each time segment and state standardized and typical timer values that are used in today's available software by Cisco Systems Inc. [Cis06] and Juniper Networks Inc. [Jun06].

Fault Detection Time:
As described in Section 2.2.1 network element failures are detected by exchanging *Hello Packets* between routers. A router sends *Hello packets* periodically with time T_{Hello} on all outgoing interfaces while a link between two routers is assumed to be failed, if no *Hello Packet* was received during the router dead interval (T_{RD}). Routers on both sides of the failure thus independently detect a bidirectional link failure, e.g. caused by a fiber cut. The required timers for failure detection as well as typically used timer values are summarized in Table 5.4.

Table 5.4: Hello Protocol timers.

Timer	Name	Typical Value(s)	Short Description
T_{Hello}	Hello Interval	Configurable with a granularity of 1 second. Typical default value: 10 seconds [Cis06, Jun06]	Interval between the transmissions of Hello Packets on an outgoing interface. This value must be equal in the whole network.
T_{RD}	Router Dead Interval	Configurable [SG01, Cis06]. Typical value $4 \cdot T_{Hello}$ [Hui00, Moy00]	If no Hello was received during T_{RD} the adjacent link is assumed to be failed.

Assuming default settings of $T_{RD} = 4 \cdot T_{Hello}$, the detection time of a link failure is thus between three and four times the Hello Interval (Equation (5.1)).

$$3 \cdot T_{Hello} \leq T_{Detect} \leq 4 \cdot T_{Hello} \tag{5.1}$$

Notification Time:
After failure detection at a router, a Link State Advertisement (LSA) is broadcast towards all routers of the OSPF area. The failure notification time of a router, i.e. the time until a router is aware that a remote failure has occurred, can thus be modeled as the minimum sum of internal processing times of LSAs at n intermediate routers ($= n \cdot T_{LSAProcessing}$) and the total transmission time of the message ($= T_{Trans}$), i.e. the length of the connecting wire between the failure detecting router and the receiving router times the speed of the

packet on the links.

$$T_{NotifyCalc} = n \cdot T_{LSAProcessing} + T_{Trans} \tag{5.2}$$

To further analyze the processing times in a router, additional information is required about the internal LSA packet processing. Figure 5.22 therefore depicts an overview of an OSPF routing instance model. Table 5.5 summarizes the most important internal timer values. Due to flooding of LSAs on more than one outgoing interface, it is possible that

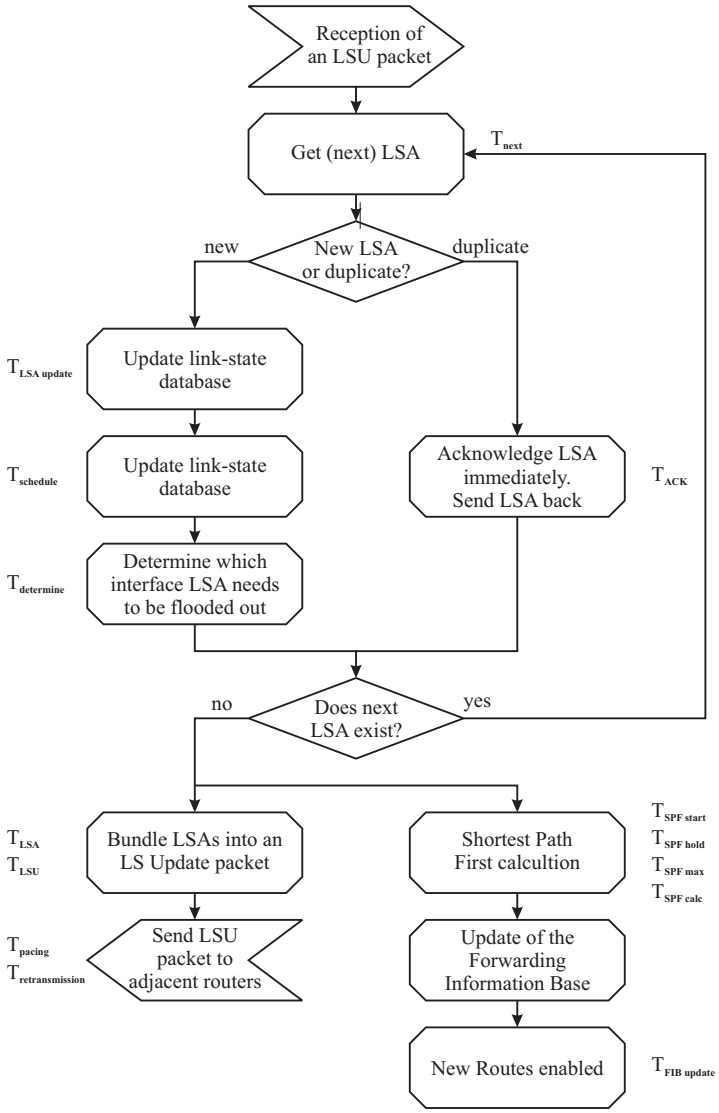


Figure 5.22: Overview of an OSPF routing instance model based on [SG01].

several copies of the same information are processed in the network at the same time. Thus, after acknowledgment of an LSU packet each router has to check whether the LSAs were already received or if they contain new information. When a duplicated LSA is received,

Table 5.5: Link State Advertisement timer overview.

Timer	Name	Typical Value(s)	Short Description
$T_{LSA\ New}$	LSA new time		Time to determine if the LSA is a new or duplicate one.
$T_{LSA\ Update}$	LSA update interval		Time to update the LS database.
$T_{Determine}$	Determine outgoing interfaces		Time to determine on which interfaces the LSA has to be sent out.
T_{Bundle}	LSU generation time		Time to generate an LSU (i.e. time to bundle LSAs together).
T_{Pacing}	Pacing Timer	33ms [Cis06], 1 second [Net03]	Time between two successive LSU packets are send down an interface.
T_{Next}	Get (next) LSA time		Time to get (next) LSA out of an LSU package.
$T_{LSA\ Age}$	Age of the LSA		Age of the LSA in seconds [Moy98].
$T_{Max\ Age}$	Maximum age of the LSA	According to [Moy98] the maximum age is one hour.	If the MaxAge expires, the LSA information is deleted from the LS database.
$T_{LSA\ MinUpdate}$	Minimum LSA update interval.	5 seconds [Cis06, Net03]	Minimum interval in which an LSA can be updated.
T_{Ack}	Acknowledge time		Time to generate and send an Acknowledge.
$T_{Retransmission}$	Retransmission Timer	66ms [Cis06], 5 seconds [Net03]	Time before retransmitting an LSU package, if not acknowledged.
$T_{Min\ LSA\ Arrival}$	Minimum LSA arrival time	1 second [Moy98, Cis06]	Maximum rate at which a router will accept updates of any given LSA via flooding.
$T_{LSA\ Refresh}$	Refresh timer	30 minutes [Moy98, Cis06]	LSAs which are still valid are retransmitted after the Refresh timer expires to prevent false deletion.

the LSA is discarded. However, when new information is available, the link state database has to be updated and the LSA has to be send to all neighboring routers except that from which the LSA packet was received from. Thus, to process a new LSA, the time T_{LSANew} is needed to check, whether the LSA is new or is a duplicate of an already received LSA and time $T_{LSUpdate}$ is needed to update the link state database. Finally, before sending the LSA to other routers, the new LSAs have to be created and bundled together T_{Bundle}. Additionally, to reduce the CPU load at neighbor routers, T_{Pacing} was introduced to allow a minimum time between the sending of two successive LSUs down an interface. Altogether, with a parallelization of LSA acknowledgment and LSA processing, the processing time at an intermediate router can be approximated according to Equation (5.3).

$$T_{LSAProcessing} \approx T_{LSANew} + T_{LSUpdate} + T_{Bundle} + T_{Pacing} \tag{5.3}$$

Finally, with a transmission velocity of approximately $\frac{2}{3}$ the speed of light (approximately 5μs per kilometer), the transmission time between two routers can be calculated according to Equation (5.4). Thus, a distance of 4500 km (approx. distance between New York and San Diego) can be traversed in about 22.5 ms.

$$T_{Trans} = 5\ \frac{\mu s}{km}\ \cdot\ \text{minimum wire-distance} \tag{5.4}$$

SPF scheduling and SPF calculation time:
After the reception of a new LSA, the Shortest Path First calculation will be scheduled. To reduce the load on router processors, however, the start of the SPF calculation is dependent on the time distance between the receptions of two consecutive LSAs and the timers $T_{SPFHold}$ and $T_{SPFDelay}$.

Figure 5.23 depicts a (simplified) trace of three possible reactions inside an OSPF router caused by the reception of two different new LSAs (e.g. one LSA for each direction of a failing bidirectional link). LSAs that are received during the SPF Delay timer will not change the scheduled SPF time. Thus, SPF calculation followed by a possible new FIB update need to be performed only once. Therefore, the following time is added to the convergence time: $T_{SPFDelay} + T_{SPFCalc}$. However, if the second LSA is received after the SPF Delay time has exceeded ($T_1 + T_{SPFDelay} < T_2$), a new SPF Delay interval will be triggered. Additionally, as depicted in the middle part of Figure 5.23, the SPF Hold timer can lengthen the interval between the SPF calculations. In this case, the following time is added to the convergence time:

$$T_{SPFDelay} + MAX(T_{SPFHold}, T_{SPF_Delay}) + T_{SPFCalc}$$

Configuring the Forwarding Information Base:
After new routes have been calculated, the forwarding information base has to be configured. Dependent on the implementation, a download of the new routes to the Forwarding Information Base (FIB) takes different time. Shaikh [SG01] reports FIB update times

Table 5.6: Shortest Path First timer overview.

Timer	Name	Typical Value(s)	Short Description
$T_{SPFSchedule}$	Schedule SPF calculation		Time to schedule the SPF calculation.
$T_{SPFDelay}$	SPF start time	Configurable, all set to 0 [Cis06]	Timer to delay SPF calculations after the receiption of a new LSA.
$T_{SPFHold}$	SPF hold time	hold timer. Configurable, set to 3s [Net03]	Minimum time between two SPF calculations.
$T_{SPFCalc}$	SPF calculation time	Dependent on the number of nodes n. $O(n^2)$ or $O(n \log n)$ [SG01], $0.00000247n^2$ + 0.000987 for specific Cisco routers [GRWC03], 1-40ms [SG01], 600ms for 300 routers [MI03], for two sample networks assumed to be 100ms and 70ms	Time needed to calculate the shortest path tree.

between 100 ms and 300 ms for the two investigated Cisco Routers. However, own measurements of Juniper routers showed average FIB update times of 500ms. Table 5.7 gives a detailed list of timers to update the FIB.

Overall Convergence Time:
Thus, the overall convergence time of a router x ($T_{RTR}(x)$) with a distance of n hops to the failure can be calculated according to Equation (5.5). The last new received LSA traversed n intermediate routers.

$$\begin{aligned} &3 \cdot T_{Hello} + T_{Trans} + (n+1) \cdot (T_{LSANew} + T_{LSAUpdate} + T_{Bundle} + T_{Pacing}) + \\ &+T_{SPFDelay} + T_{SPFCalc} + T_{RInstallDelay} + T_{FIB} \\ &\leq T_{RTR}(x) \ \leq \\ &4 \cdot T_{Hello} + T_{Trans} + (n+1) \cdot (T_{LSANew} + T_{LSAupdate} + T_{Bundle} + T_{Pacing}) + \\ &+MAX(T_{SPFHold}, T_{SPFDelay}) + T_{SPFCalc} + T_{RInstallDelay} + T_{FIB} \end{aligned} \tag{5.5}$$

The minimum convergence time of a router x is thus a sum of the minimum time to detect a failure ($3 \cdot T_{Hello}$), the time to create and send an LSA at a failure detecting router ($T_{Bundle} + T_{Pacing}$), the transmission time of the LSA towards router x (T_{Trans}), the LSA

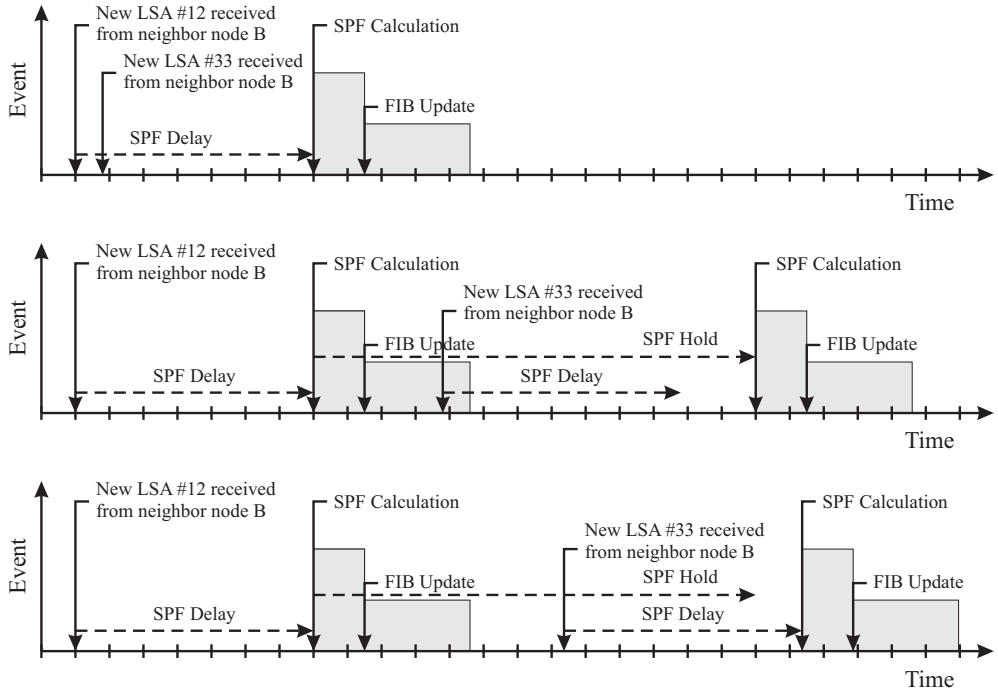


Figure 5.23: Example trace of typical reactions inside an OSPF router.

Table 5.7: Configuration of the Forwarding Information Base timer overview.

Timer	Name	Typical Value(s)	Short Description
$T_{RInstallDelay}$	Route install delay	observed to be 0.2 seconds [GRWC03]	Delay between successfull calculation of routes and FIB updating
$T_{FIBUpdate}$	FIB update time	for some Cisco routers 100-300ms [SG01]	Time to update the Forwarding Information Base

processing times in the n intermediate routers ($n \cdot (T_{LSANew} + T_{LSAUpdate} + T_{Bundle} + T_{Pacing})$), the reception of the LSA at router x ($T_{LSANew} + T_{LSAUpdate}$), a calculation time T_{SPF} with a minimum delay of $T_{SPFDelay}$, and the configuration time of the forwarding information base ($T_{RInstallDelay} + T_{FIB}$).
The maximum convergence time of a router x is a sum of the maximum time to detect a failure ($4 \cdot T_{Hello}$), the time to create and send an LSA at a failure detecting router ($T_{Bundle} + T_{Pacing}$), the transmission time of the LSA towards router x (T_{Trans}) the LSA processing times in the n intermediate routers ($n \cdot (T_{LSANew} + T_{LSAUpdate} + T_{Bundle} + T_{Pacing})$), the reception of the LSA at router x ($T_{LSANew} + T_{LSAUpdate}$), a calculation time T_{SPF} with a delay of $T_{SPFDelay}$ or $T_{SPFHold}$, and the configuration time of the forwarding information base ($T_{RInstallDelay} + T_{FIB}$).

Because of the distributed approach of the OSPF protocol, the routers perform their OSPF Link State updates and calculations in parallel. Thus, the stable state of all forwarding information bases (T_{RTR}) will be reached with the convergence of the last router (Equation (5.6)).[8]

$$T_{RTR} \geq T_{RTR}(x) \qquad \forall x \in \text{Routers} \tag{5.6}$$

Similarly, the convergence time of the protocol, i.e. the processing of the last (duplicate) LSA, is dependent on the number of routers in the OSPF area and their topology. Assuming the worst case topology (a ring of m routers), the time T_{CTP} can be calculated according to Equation (5.7).

$$T_{CTP} \leq \begin{cases} 4 \cdot T_{Hello} + T_{Trans} + m \cdot T_{LSANew} + (m-1) \cdot (T_{LSAupdate} + T_{Bundle} + T_{Pacing}) \\ T_{RTR} \end{cases} \tag{5.7}$$

No verification and synchronization procedure is defined in OSPF. Thus, the recovery time of OSPF (T_{RT}) is equivalent to the convergence time of the routes T_{RTR}. Dependent on the delay difference between the original failure-free route and the new route an additional small (positive or negative) propagation time T_{VSP} is added to the route convergence time (Equation (5.8)).

$$T_{RT} = T_{RTR} + T_{VSP} \tag{5.8}$$

Recovery Time Discussion:
Some components of the convergence time are very large compared to others. Transmission and processing times of LSAs for example are small compared to the large default values of SPF Delay and Hold timers. Thus, times T_{RT}, T_{CTP} and T_{RTR} are dominated today by the Router Dead Interval, the SPF Delay and SPF Hold Timers. With default values T_{RT} is around 40 to 50 seconds.

During the convergence time and the distribution of topology change information, the routers do not have the same view of the topology. Thus, false routes, routes towards failed elements as well as routing loops can occur until all routers have been converged.

5.3.2.2 Enhancement Proposals - Reducing the Recovery Time

Some years have passed since the OSPF protocol was standardized. Network characteristics and requirements have changed. In particular, processor speeds and bitrates have grown rapidly. Thus, in the last few years there were several proposals to extend and change the OSPF protocol in order to increase the performance and accelerate its convergence and recovery time. In the following, we will shortly present and summarize the most important proposals.

[8]It must be noted, that some older OSPF implementations, e.g. as reported in [SG01], wait for the calculation of the shortest path before forwarding the LSAs. However, since no additional information is required the forwarding of LSAs can be performed in parallel to the calculation. This is the default behavior in today's implementations.

Reducing the Failure Detection Time:
The recovery time of any resilience mechanism can be accelerated by speeding up the failure detection time. As mentioned in the previous section the minimum OSPF Hello Interval is one second (according to the OSPF standard [Moy98]). Lower layer mechanisms e.g. in SDH and SONET are able to detect a failure in less than 10 milliseconds (Loss Of Signal, Loss of Frame, Alarm Indication Signal) [VPD04]. By using a multi-layer signalization, the OSPF rerouting process could be started immediately. However, failures of the IP layer cannot be detected by lower-layer mechanisms. Link bandwidths as well as the processor speeds have increased. Thus, it is possible to send probing packets more frequently. In 2000, Alaettinoglu et al. [AJY00] proposed to reduce the Hello interval and to reduce (or even set to zero) the SPF Delay and SPF Hold timers. Simulation results of the IGP protocol IS-IS and OSPF in [AJY00] and [GRWC03] revealed no routing instabilities while reducing the Hello Interval. Similar simulations by [BJ01] using sub-second Hello Timers in OSPF networks reported a considerable improvement of convergence times without significantly adding to the processor load. However, the latter reported an increase of route flaps while reducing the Hello interval below 275ms due to missed Hello packets. Recently, major router vendors including Cisco and Juniper included the possibility to send Hello packets more frequently (sub-second range). A corresponding protocol called bidirectional forwarding detection is currently discussed for standardization in the IETF [KW06].

Differentiate Good from Bad News:
A fast reaction upon failures is required for the transport of real-time data. However, often before a network element fails a frequent change between a functional and a failed status can be observed. With sub-second detection of faults and the absence of dampening timers, e.g. SPF Delay, the network would react rapidly on these link transients and the routes would be changed frequently. To overcome this problem that occurs when having sub-second convergence Alaettinoglu et al. [AJY00] proposes to use adaptive dampening methods to treat 'bad news' different from 'good news'. I.e. the network reacts fast on a failure but react slowly on the situation when a link comes up again.

New Algorithm for the SPF Calculation:
The shortest path calculations today use the Dijkstra or Bellmann-Ford algorithm. These algorithms recalculate routes to all destination of the OSPF area. However, considering single failures, only few routes are affected. Thus the CPU intensive SPF calculation can be replaced by new algorithms that re-compute affected routes only. With run times in the order of O(log n) compared to O(n log n) of the Dijkstra algorithm, larger OSPF areas can be accomplished with reduced SPF calculation times [AJY00, FFEB05].

Prioritize LSA Propagation to SPF Calculation:
Some older router implementations perform an SPF calculation before sending LSA messages. However, no additional information is generated by the SPF calculation that should be included in the LSA. However, there is no advantage and no technical reason why the

flooding of the LSA should wait for the completion of the SPF calculation. If the forwarding cannot be done in parallel to the SPF calculation as proposed in Figure 5.22 the propagation of topology change information should be prioritized local SPF calculations to accelerate the overall convergence time.

Explicit Marking and Prioritizing Hello and LSA Packets:
To prevent the delay or loss of OSPF messages, e.g. Hellos, LSAs or Acknowledgments, due to congestion of links or high CPU loads on the routers, [AJY00] and [Cho05] propose to prioritize OSPF messages to data. Former versions of [Cho05] additionally proposes to treat any packet received over a link as surrogate for a Hello packet in further OSPF versions.

5.3.2.3 Simulation of the OSPF Convergence Behavior

In order to confirm the convergence time of OSPF we implemented the above router instance model in the network simulator NS-2 [NS206]. Table 5.8 presents selected simulation results for simulations with NS-2 version 2.7 on three different example networks [**EGI+03**].

Table 5.8: Minimum, mean, and maximum recovery times measured in simulations using: $T_{FIB} = 0.300$s; $T_{Pacing} = 0.033$s; T_{Hello} was varied randomly by $\pm 10\%$. 10 measurements per value.

#	Network	Link failure between	$T_{Hello}/T_{RD}/T_{SPFDelay}/T_{SPFHold}$	Min/Mean/Max T_{RTR} in seconds	Min/Mean/Max T_{CTP} in seconds
1	NSF Net, 14 nodes, 42 edges	Seattle - Palo Alto	10/40/5/10	36.98/42.29/51.76	36.90/42.30/51.78
2	NSF Net, 14 nodes, 42 edges	Seattle - Palo Alto	1/4/5/10	8.44/8.64/8.86	8.45/8.65/8.87
3	UUNet, 42 nodes, 79 edges	Chicago - Detroit	1/4/0/0	3.62/4.03/4.39	3.64/4.11/4.50
4	UUNet, 42 nodes, 79 edges	Seattle - San Francisco	1/4/0/0	3.34/3.98/4.45	3.41/4.06/4.54
5	KING Net, 20 nodes, 51 edges	Buffalo - Houston	1/4/0/0	3.82/4.14/4.70	3.82/4.19/4.72
6	KING Net, 20 nodes, 51 edges	Phoenix - San Francisco	1/4/0/0	3.07/3.95/4.39	3.44/4.06/4.45

The first simulation result illustrates the recovery times for default values which are in the theoretical investigated range.[9] As discussed in the previous section the recovery

[9]Note, due to the random variation of T_{Hello} of $\pm 10\%$ to avoid synchronization effects, the maximum detection time T'_{RD} changed to 110% T_{RD}.

times are dominated by failure detection times and are around 40 seconds. When reducing the failure detection timers to the allowed minimum values of $T_{Hello} = 1$ and $T_{RD} = 4$, the recovery times were reduced to around 9 seconds. Since the time difference between the detection at both ends of the failure plus the propagation of this information is less than 5 seconds, the second LSA reaches the nodes during the SPF calculation delay time ($T_{SPFDelay}$). Thus, $T_{SPFHold}$ was not used. Furthermore, results 3 to 6 show simulation results with deactivated SPF throttle timers ($T_{SPFDelay} = 0$, $T_{SPFHold} = 0$) which is proposes by Cisco Systems Inc. Overall recovery times of around 4 to 5 seconds can be reached.

When further reducing the detection time to sub-second ranges, sub-second recovery times can be reached. However, the exact values are very much dependent on router implementation (parallelization of tasks and FIB configuration time T_{FIB}). Using the proposed T_{Hello} values of about 275ms [BJ01] recovery times in the order of one second are achievable. Table 5.9 lists recovery time simulation results for idealized lower layer detection and calculation times at different routers in a nationwide U.S. example network as a reaction to a link failure.

Table 5.9: Route recovery time at different routers after link failure 'Chicago - Detroit' with idealized timers: $T_{Detect} = 5$ms; $T_{SPF} = 0$ms; $T_{SPFDelay} = 0$; $T_{SPFHold} = 0$; $T_{FIB} = 300$ms; $T_{LSAnew} + T_{Processing} = 600\mu$s; $T_{Pacing} = 0$ms.

Router	Convergence Time of the Routers in seconds
Chicago	0.355
San Francisco	0.387
Miami	0.392
Boston	0.347
Overall mean value	**0.374**

5.3.3 Recovery Time of MPLS

Only few real-time measurements are available for the recovery time of MPLS [CMU03]. A white paper, published by the router vendor Cisco Systems Inc., gives also only vague information about the recovery time of MPLS [Cis03]: "*MPLS Fast Reroute feature provide a mechanism for rapidly repairing (under 50ms; actual fallover time may be greater or less than 50ms, depending on the hardware platform, the number of TE Tunnels and/or Network prefixes) an LPS by routing along a detected failure in order to minimize the length of service interruption experienced while the head-end attempts to establish a replacement LSP.*" Similarly, RFC 3469 [VSFH03] that defines the above mentioned recovery model of

MPLS does not give any insights or formulas to calculate or estimate MPLS recovery times other than that of [Cis03][10].

5.3.3.1 Theoretical Analysis

As seen in Chapter 3 a large number of path-based resilience mechanisms exist. However, especially the location of reacting entities and the location of the path computational elements play an important role when considering recovery time.

Although a centralized computation approach is possible with MPLS, we will focus on a distributed approach in which the recovery path computational element (PCE) coincides with the start-location of the detour as proposed in [DR00a]. Because of that, the recovery time of the MPLS resilience mechanisms can be calculated according to Equation (5.9).

$$T_{RT} = T_{Detect} + T_{HoldOff} + T_{Notify} + T_{Calculation} + T_{Establishment} + T_{VSP} \tag{5.9}$$

The MPLS recovery time analysis is based on [Aut02]. The formulas are extended to model the differences in propagation delay between the working path and the backup path. Additionally, new models are added for local-to-egress protection. Table 5.10 summarizes the used variables in the MPLS recovery time analysis.

Fault Detection Time and Hold-Off Time:
Similarly to OSPF, there exist two approaches to detect a link failure: Heartbeat detection and other/lower layer mechanisms. In heartbeat detection, packets are sent in constant time intervals between two MPLS switches. Although this mechanism is similar to the OSPF *Hello protocol* the intervals are considerably smaller in MPLS and in the range of some ms. Additionally, failure detection mechanisms of other technology layers can be used, e.g. loss of frame of SDH/SONET mechanisms or BFD messages. In addition, a configurable hold-off timer can be used to delay a reaction to allow resilience mechanisms of other layers to perform the recovery.

Notification Time:
After a failure f is detected the calculating and reacting entities of all affected MPLS paths must be notified. Figure 5.24 depicts the locations of these entities for the different resilience mechanisms.

Assuming a sequential processing of I affected flows, i.e. I working paths traversed the failed link and the router upstream of the failure processes all affected MPLS iteratively, at most $(I-1)$ paths are processed before the processing of path p is initiated. After this processing time of $I \cdot P$ a message is sent back to the detouring node and traverses $n_{p,f}$ intermediate routers. In local protection and local-to-egress protection, the detecting node (adjacent to the failure) is responsible to detour the traffic. Thus, no further signaling

[10]RFC 3469: "*Fastest MPLS recovery is assumed to be achieved with protection switching and may be viewed as the MPLS LSR switch completion time that is comparable to, or equivalent to, the 50ms switch-over completion time of the SONET layer.*"

Table 5.10: Variables used in the MPLS recovery time analysis.

w_p	Number of routers of working path p.
$dw_{p,f}$	Number of detoured routers from working path p during failure f.
I	Number of paths on a link.
g_f	Number of affected paths on a detouring node for failure f.
$n_{p,f}$	Number of upstream routers, i.e. number of routers between the failed network element and the detour node of path p and failure f.
$b_{p,f}$	Number of links on the backup route from the detour router to the merging router of path p.
P	Message processing time per router. A sequential processing is assumed, i.e. a processing of x flows needs x·P time. P is assumed to be 10 ms as reported in [RM99]
D	Propagation delay of a link (5 μs / km)
$L(l)$	Length of link l in km
S	Time to change/alter the switching table. Is assumed to be 10 ms as reported in [Ram99]
C	Time to calculate a new constrained based backup route. This time depends on the network size and the number and characteristics of path constrains. It is possible that a single calculation yields backup paths for several affected MPLS flows (e.g. Dijkstra, k shortest path). However, for simplicity, we assume a calculation for each individual MPLS flow that takes 2 ms

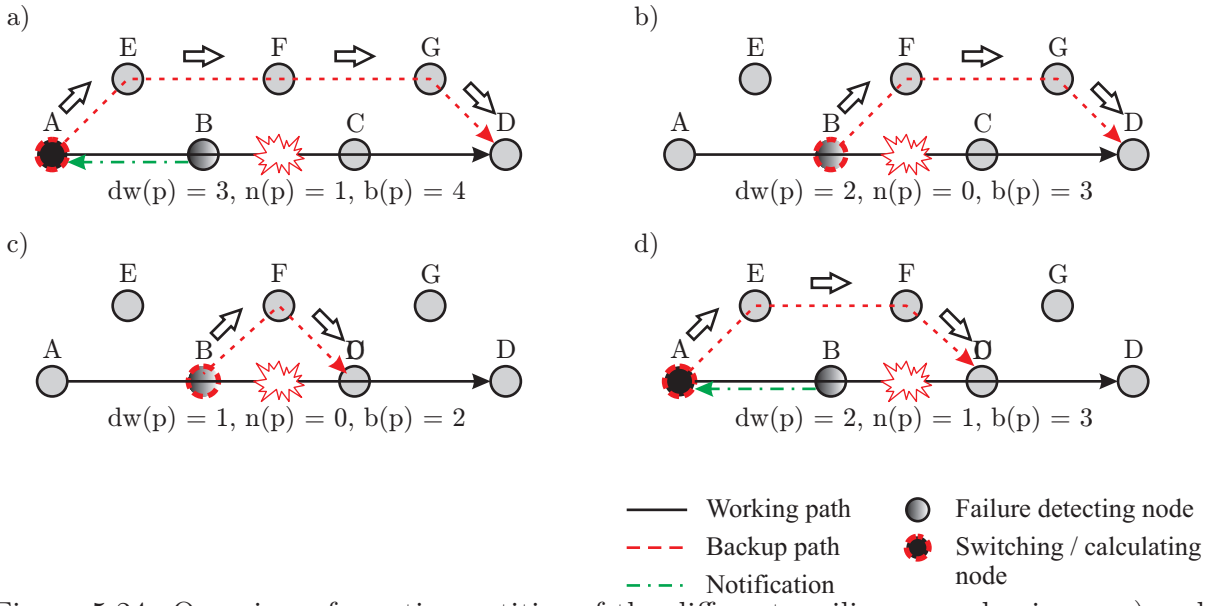


Figure 5.24: Overview of reacting entities of the different resilience mechanisms. a) end-to-end path protection/restoration, b) local-to-egress protection/restoration, c) local link protection/restoration, d) regional protection/restoration, e) global restoration.

has to be started ($n_{p,f} = 0$). In end-to-end protection and regional protection, however, a message has to be sent upstream to the detouring node and an edge-length dependent propagation delay $l_e \cdot D$ as well as a processing delay P in all $n_{p,f}$ intermediate nodes is required. The required notification time can be calculated according to Equations (5.10).

$$T_{Notify}(p) = \begin{cases} I \cdot P & \text{for local link or local-to-egress protection/restoration} \\ I \cdot P + \sum_{e \in n_{p,f}} (P + l_e \cdot D) & \text{for end-to-end or regional protection/restoration.} \end{cases} \tag{5.10}$$

Calculation Time:
If the backup path is not pre-calculated, a node has to calculate backup paths for all g affected paths for which the node is responsible. Assuming a sequential calculation, time $g \cdot C$ is required to calculate backup paths for all g affected paths.

$$T_{Calculation}(p) = \begin{cases} 0 & \text{for pre-calculated or pre-established protection.} \\ g_f \cdot C & \text{for restoration.} \end{cases} \tag{5.11}$$

Establishment Time:
If the backup path is pre-established the detour node has to alter the switching table only (time S). However, if the backup path is not pre-established, resources have to be reserved and the nodes along the backup paths have to be configured. The establishment time of a backup path is dependent on the configuration protocol. The Resource Reservation Protocol (RSVP) [BZB+97] that is used in MPLS uses a two-way configuration procedure and a reservation/configuration message traverses the backup path twice (back and forth). Thus, twice the processing times P, configuration times S and edge-propagation delays $l_e \cdot D$ are required.

$$T_{Establishment}(p) = \begin{cases} S & \text{for pre-established protection.} \\ 2 \cdot \sum_{e \in b(p)} (D + l_e \cdot P) + S & \text{else.} \end{cases} \tag{5.12}$$

Verification, Synchronization and Propagation Delay:
Finally, an additional delay perceived by the traffic sink is caused by the path propagation time difference of the backup path b and the detoured working path part dw and can be modeled according to equation (5.13).

$$T_{VSP}(p) = \sum_{e \in b(p)} (l_e \cdot P) - \sum_{e \in dw(p)} (l_e \cdot P) \tag{5.13}$$

Overall Convergence Time:
Assuming a detection time T_{Detect} in the order of 10 ms [KW06], no hold-off time $T_{Hold-Off} = 0$, propagation delays D of 5μs per kilometer ($\frac{2}{3}$ speed of light), processing delays P of some ten μs and switching and calculation times for each individual flow in the order of some ten ms and some ms, we can categorize the recovery time for path-based resilience mechanisms according to table 5.11.

Table 5.11: Recovery time categorization.

	Pre-established protection	Pre-calculated protection	Restoration
Local backup paths	20 to 50ms	some hundreds of ms	hundreds of ms to seconds
Local-to-egress backup paths	20 to 50ms	some hundreds of ms	hundreds of ms to seconds
Regional backup paths	20ms to hundred ms	some hundreds of ms	hundreds of ms to seconds
End-to-end backup paths	hundred ms	some hundreds of ms	hundreds of ms to seconds

When inspecting the equations, the contributing factor of recovery time can be identified as signaling and path set-up. Thus, in general, sub-second recovery times can only be reached by protection mechanisms while recovery times below one hundred ms are only possible with local protection mechanisms or protection mechanisms with reduced upstream signaling scope.

5.3.4 Summary

In this section, we analyzed the recovery time of OSPF and MPLS path based resilience mechanisms. A theoretical analysis of OSPF revealed that recovery times in the order of one second could be reached with OSPF. However, even sub-second convergence times are possible by increasing the processing power of line card processors.

The theoretical analysis of path-based resilience mechanisms confirmed the classification of Chapter 3:

Recovery Time:
SLLPP $\leq$ SRPP(1-0/0-1) $\leq$ SRPP(2-0/0-2) $\leq$ SE2EPP $\leq$ GR

We revealed that sub-second recovery times can only be reached by protection mechanisms. Recovery times below one hundred ms are only possible with local protection mechanisms or protection mechanisms with reduced upstream signaling scope.

5.4 Configuration Complexity

Chapter 2 showed that a large OPEX cost-factor is generated by network operation. Especially the set-up, teardown, and reconfiguration of services are complex and can cause manifold misconfiguration if done by hand. While the number of possible paths is large, only a sub-set of paths is chosen by network optimization.

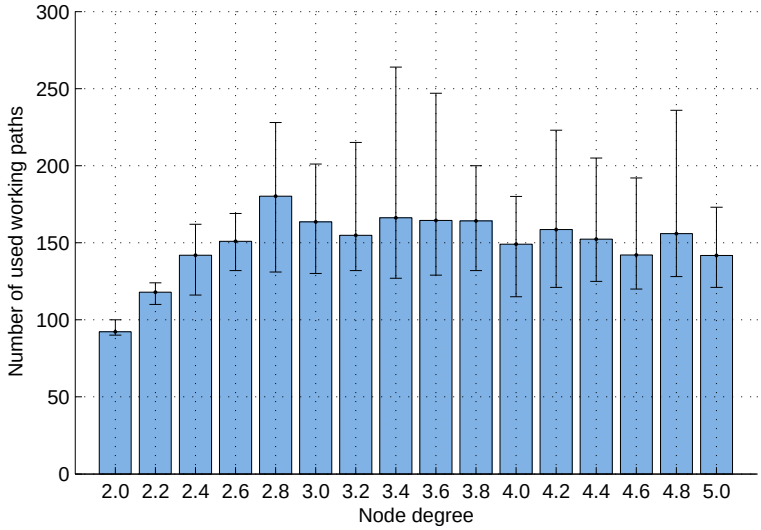


Figure 5.25: Number of used working paths for the 10-node example networks (min/average/max).

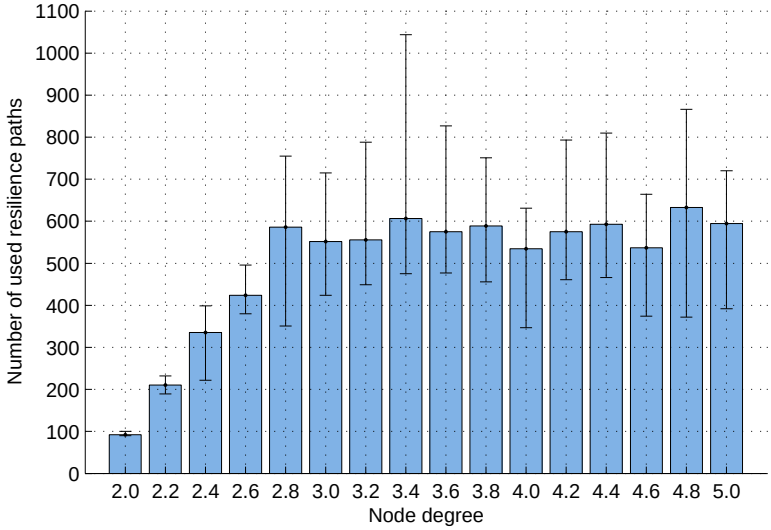


Figure 5.26: Number of used backup paths for the 10-node example networks (min/average/max).

Figures 5.25 and 5.26 depict the number of used working and backup paths for the protection of the 10-node example networks (SE2EPP against single link failures). In average, 149.8 working path were chosen for the optimal network design (varying from 90 to 264) which is only slightly higher than the minimum possible number of $10 \cdot 9 = 90$ paths. The number is relatively constant for all example networks with a node degree equal or higher than 2.4 (average 153.5) and is even smaller for more sparsely meshed networks.

Similarly to working paths, the number of backup paths[11] remains relatively constant for all example networks. In average 577.5 backup paths were chosen for networks with a node degree higher than 2.8. Thus, in average around 727 paths have to be configured for the example networks.[12]

Although, the number stays reasonable, a configuration of these paths is quite complex if done by hand. However, path configurations can easily be automated. As already mentioned in Section 5.1.3 the network optimization program *Resilient Network* was enhanced with an MPLS configuration module. A reference implementation of Cisco IOS version 12.0 and tests on Cisco 7200 routers confirmed that a tool-based automatic configuration of routers is possible with relative simple mechanisms.

[11]This number can be reduced by combining backup paths that can be used to protect several failure patterns or working paths. However, for simplicity of router configuration backup paths were kept separate.

[12]Note, the number of working and backup paths was not limited during the optimization process. If required by network operators, the numbers can be limited by adding simple additional equations to the ILP that restrict $\sum WPU^B_{d,p}$ or $\sum RPU^B_{d,s,p,p'}$.

Chapter 6

Summary and Outlook

6.1 Summary

Highly available communication networks have become one of the cornerstones of our society. Because of their importance, a large number of mechanisms were developed in order to reduce network outage time caused by network equipment failures. The choice of the resilience mechanism is an important issue when designing telecommunication networks and has a substantial influence on capital and operational expenditures. Furthermore, in order to obtain cost-efficient networks, network optimization procedures have to be performed that take routing, dimensioning, and failure-free as well as failure-affected network states jointly into account.

This thesis contributed to these tasks in three areas: resilience mechanism classification, resilient network optimization and resilience mechanism evaluation.

Resilience Classification Framework:
By analyzing existing resilience mechanisms, it became clear that a number of issues impede the choice towards a suitable resilience mechanism today: The parallel development of resilience mechanisms created a plethora of mechanisms and terminologies that are described by different standardization bodies and companies. In addition, different emphasis was given to individual characteristics of the mechanisms. Guidelines, how a network with a given resilience mechanism should be designed, are often not provided.

This thesis therefore presented a resilience classification framework with that resilience mechanisms can be described systematically. We showed that every resilience mechanism is a combination of individual characteristics and can thus be decomposed into building blocks. We proposed eight building blocks with which resilience mechanisms can be described precisely. Since individual influences of characteristics can be taken into account separately and systematically, the framework facilitates an analysis of resilience mechanisms considerably. In addition, by comparing example classifications, we showed that many used resilience mechanisms differ in very few characteristics only. Thus, a theoretical comparison of different resilience mechanisms considering capacity requirements and

recovery time can be performed easily with the framework. Furthermore, new resilience mechanisms can be discovered by combination of building blocks. As an example, we described a novel resilience mechanism that dynamically reacts to traffic load changes and network equipment failures. Each network node adapts routes as well as traffic distribution locally and autonomously.

Resilient Network Optimization:
In this thesis, we analyzed the classical network design and planning process in detail and summarized the requirements of resilient network planning. We showed that cost-optimal network designs can only be achieved by a joint consideration of failure-free and failure-affected network states. Furthermore, we highlighted that a detailed analysis and understanding of resilience mechanism characteristics and their reactions in case of failures is required in order to select the best resilience mechanism for the intended purpose of the network. While a theoretical analysis helps to classify resilience mechanisms, network optimization and analysis can quantify the differences of the resilience mechanisms for the given network. We furthermore discussed network optimization approaches that exist in the general literature and showed that most of these approaches do not provide information about the quality of the obtained solution. Therefore, this thesis presented optimization approached based on linear programming that are either able to obtain the cost-optimal solution or at least provide information about the optimality gap, i.e. the difference of the found solution to the unknown optimal solution. In particular, we provided complete formulations for two linear programming approaches for path-based protection and restoration mechanisms: Flow- and path-based formulations. While flow based-formulations are inferior in complexity and required calculation time today, new linear programming approaches based on interior point methods will help to close the gap between the two approaches. The presented formulations using the path-approach however, can directly be applied to the design of resilient networks. We furthermore, applied a mathematical technique called *Column Generation* with which the optimization of resilient networks can be accelerated considerably and large telecommunication networks can be planned in an optimal fashion.

Evaluation of Multipath Resilience Mechanisms:
In order to provide more insights in resilience mechanisms and cost-optimal topology and path-selection this thesis furthermore evaluated five popular path-based protection and restoration mechanisms. Next to a comparison of the mechanisms based on the resilience classification framework, we analyze case study optimization results to deduct quantitative capacity requirements. We showed the influence of node degree on capacity requirements and compared the capacity requirements of the investigated resilience mechanisms with each other. In addition, we analyzed the selected paths of the optimal solutions in order to provide guidelines for the development of faster algorithms and heuristics for the planning of resilience networks. We showed that short paths are preferable for failure free routes while a mix of all path-lengths should be considered for backup paths.

In addition, a special focus of this thesis was on multipath capable path-based resilience mechanisms. Therefore, we discussed issues of multipath routing and evaluated the influence of multipath routing on capacity requirements. It is shown that multipath is especially beneficial for working paths to create capacity batches that can be shared more efficiently. However, the analysis of case-study results revealed that little benefits can be achieved with multiple splits. A split of traffic in two or three different parts that are forwarded along different routes is sufficient to provide cost-optimal solutions.

Furthermore, this thesis analyzed the recovery times of OSPF and path-based MPLS resilience mechanisms. We presented formulas with which the recovery time of OSPF rerouting as well as path-based MPLS protection and restoration mechanisms can be calculated. Recovery time simulations supported the theoretical deliberations.

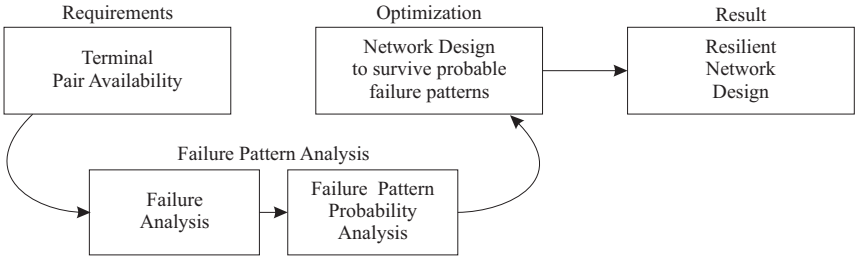
6.2 Outlook

The contributions of this thesis can serve as a basis for future research in the area of resilient network planning that is conceivable in several directions. Let us present a selection of future research topics.

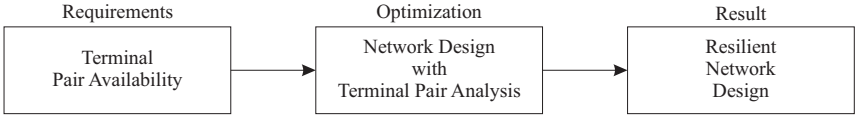
Current trends in the design of transport networks indicate an evolution towards networks with a limited number of technology layers. Especially, the combination of electrical switching (e.g. MPLS or Carrier Grade Ethernet) with bitrate efficient optical transport (e.g. Wavelength Division Multiplex) is considered to cut down on network costs considerably. Furthermore, the integration of devices and the use of joint automated control and management functionality enable the deployment of multi-layer resilience mechanisms. So far, the analysis of resilience mechanisms and the resilience network optimization in this thesis were restricted to one technology layer. However, an interdigitation of resilience mechanism of multiple layers might be beneficial from an availability and cost point of view. Therefore, new combined resilience mechanisms have to be developed and strategies where and which protection mechanism should be deployed have to be analyzed. Furthermore, in order to find cost-optimal multi-layer networks new optimization methods have to be developed [**SPG$^+$06**].

The development of new services and the introduction of high-speed access technologies will influence the traffic patterns and traffic dynamics in access but also in transport networks. To dynamically adapt to traffic changes, automatic control and management functionality are conceivable in future networks: Instead of finding an optimal resilience network design, path computation elements can be deployed in a distributed manner to adapt routes and traffic distributions to changing traffic requirements. In addition, a combination of proactive or reactive protection or restoration mechanisms can be used that either prepare the network for the next probable failures or react fast upon the occurrence of a failure. Therefore, new resilience mechanisms, algorithms or concepts have to be developed in order to provide, fast, distributed, dynamic, and cost efficient resilient networks.

Furthermore, the optimization approach of this thesis uses a two-step design that is depicted in Figure 6.1(a). In a first step availability analysis are performed in order to obtain probable failure patterns. Consequently, the network is optimized in order to provide resilience mechanisms for the specific failure patterns. However, approaches that take terminal-pair availability requirements directly into account might further reduce costs if not all elements of a path have to be protected (Figure 6.1(b)). However, due to the non-linearity of terminal-pair calculations (e.g. shown in [YLK02, KLY99]) an optimization of resilient networks based on terminal-pair availability is rather complex. New calculation and optimization approaches have thus to be developed in order to allow a cost-optimal resilient network planning.



(a) Two-step optimization approach.



(b) Optimization approach based on terminal-pair availability.

Figure 6.1: Comparison of optimization approaches.

Finally, when considering multiple paths or shared protection mechanisms, it becomes apparent that next to terminal-pair availability another measure of resilience exist: Terminal Pair Available QoS [**AG06**]. Figure 6.2 depicts an example demand that is routed along multiple paths that have different capacity (C) and availability (A) values.

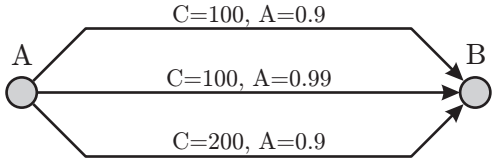


Figure 6.2: Example demand that is routed via multiple paths

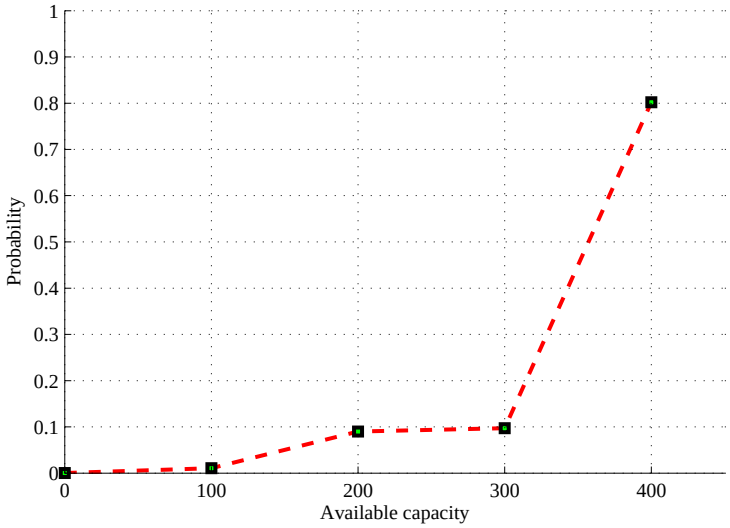


Figure 6.3: Terminal pair available capacity for the example demand.

Obviously, the available end-to-end capacity is dependent on the individual path availabilities. Capacity or in general QoS characteristics are thus dependent on availability as depicted in Figure 6.3. Similarly, when using shared protection mechanisms, the shared capacity on a backup path can be used by another working path. Thus, dependent on the failure probabilities of the working paths, backup capacity is available or not. Thus, from a network operator's point of view another type of service could be offered by network operators for which network optimization approaches are required.

Appendix A

Resilience Terminology

Availability: *Availability is the probability that an item will be able to perform its designed functions at the stated performance level, within the stated conditions, and in the stated environment when called upon to do so.* [Kal02]. *When we assume constant component failure- and repair-rates the availability can be approximated using Mean Time Between Failure (MTBF) and Mean Time To Repair (MTTR) values:*[1]

$$A \approx \frac{MTBF}{MTBF + MTTR} \tag{A.1}$$

Backup Resources: *A resource, e.g. a path, that is used in fault condition to restore traffic of a working path. The recovery path can either be an equivalent recovery path and ensure no reduction in quality of service, or be a limited recovery path and thereby not guarantee the same quality of service (or some other criteria of performance) as the working path. Synonyms for a backup resource are: recovery resource, alternative resource, and protection resource.* [IT03c, VSFH03]

Bidirectional: *Same values and characteristics apply for both directions. E.g. a bidirectional failure: A failure occurs simultaneously for both oppositional directions.*

Dedicated Resources: *Reserved recovery resources that may be used to protect one working resource and cannot be shared.*

Demand: *The aggregation of flows between each pair of nodes on the transport network.* [Gro04]

Extra (Preemtible) Traffic: *Traffic that is purposely placed on a backup resource in the knowledge that, on failure, this (extra) traffic will be disconnected to make way for the backup traffic from the failed working connection.* [IT03c]

[1] Details of the approximation can be found in [Ise99, Annex B]

Failure: *Termination of the capability to transfer user or OAM information due to an outage.* [IT03c]

Fault Tolerant: *Extent to which a functional unit will continue to operate at a defined performance level even though one or more of its components have failed.* [ETS05]

Global Restoration: *A resilience mechanism in which new routes for all working paths are calculated, configured, and established dynamically after the detection of a fault.*

Guaranteed Restoration: *A restoration mechanism in which suited backup resources are guaranteed for the considered failure or the considered failure patterns. However, the backup resources are calculated, configured, and activated dynamically after the detection of a fault.*

Mean Time Between Failure: *Mean Time Between Failure (MTBF) is the average time a device will function before failing.*

Mean Time To Repair: *Mean Time To Repair (MTTR) is the average time that it takes to repair a failure.*

Multipath: *Multiple resources that carry the traffic of a demand or working path based on a certain load splitting rule.*

Outage: *An event in which a service becomes unavailable due to a failure of some type (typically temporary).* [ACCC03]

Path Group: *A logical bundling of multiple working paths of one or several demands, each of which is routed identically.* [VSFH03]

Pre-configured: *A recovery resource that is prepared for establishment but needs to be activated. Variants include the case where an optical path or trail is configured, but no switches are set.*

Pre-established: *A recovery resource that is established prior to any failure on the working path.* [VSFH03]

Pre-reserved: *A recovery resource with reserved required resources on all hops along its route. The resources held by a set of recovery paths may be shared.* [VSFH03]

Protection: *A resilience mechanism that uses suited pre-planned, pre-configured, and pre-established backup resources.*

Reliability: *The probability of performing a specified function without failure under given conditions for a specified period of time.*

Rerouting: *Restoration in IP networks.*[2]

Resilience: *The capacity of a system exposed to threats to adapt by resisting or changing in order to reach and maintain an acceptable level of functioning and structure.*

Restorability: *The percentage of demands, demand capacity, or paths that can be restored in case of a failure.*

Restoration: *A resilience mechanism in which backup resources for failure affected working paths are calculated, configured, and established dynamically after the detection of a fault.*

Robustness: *The condition of a product or process design that remains relatively stable, with a minimum of variation, even though factors that influence operations or usage, such as environment, are constantly changing.*

Revertive Mechanism: *A resilience mechanism that is able to revert, i.e. switch back to the working path, after the successful reparation of the failure.*

Shared Resources: *Reserved recovery resources that will be available to protect different working resources if the protected resources are not simultaneously subject to a failure.*

Shared Risk Group (SRG): *SRG is a group of links or nodes that can fail simultaneously due to a single failure incident.* [IT03c]

Spare Capacity: *The required capacity along backup routes.* [Gro04].

Survivability: *The ability to continue to provide service in the event of a failure.* [Gro04]

Survivable Network: *A network that is capable of restoring traffic in the event of a failure.* [IT98]

Unidirectional: *In one direction only, e.g. an unidirectional path has no associated reverse path.*

[2]Compare the definition of rerouting in RFC 3469 [VSFH03]: "*A recovery mechanism in which the recovery path or path segments are created dynamically after the detection of a fault on the working path. In other words, a recovery mechanism in which the recovery path is not pre-established.*"

Working Resources: *A resource, e.g. a path, that is used in fault-free condition. Synonyms for a working resource are primary resource and active resource.* [IT03c, VSFH03]

Appendix B

(Meta-) Heuristics

Meta-Heuristics are general strategies that can be used as guidance during the search for feasible solutions and are often very good approaches to find good solutions in limited time. In the following, we will sketch two most widely used probabilistic meta-heuristics.

B.1 Simulated Annealing

Simulated Annealing was introduced by S. Kirkpatrick et al. in 1983 [KGV83]. Its name is derived from the analogy between the way liquids freeze and crystallize or in which metals cool and anneal. At high temperature, the molecules of a liquid are moving with respect to each other. When cooled down this momentum is lost. Amazingly, however, when a liquid is cooled down slowly the molecules are able to reach minimum energy states and pure crystalline structures can be constructed with this annealing process. In particular, Equation B.1 depicts the so-called *Boltzmann probability distribution*. The expression E denotes an energy state, T the temperature and the quantity k the Boltzman constant. Thus, at any temperature the energy level and the structure are able to change. Nevertheless, the lower the temperature, the more unlikely is the change.

$$\text{Probability}(\delta E) = e^{-\frac{\delta E}{kT}} \tag{B.1}$$

There often exist a large number of possible solutions for an optimization problem. Small changes of a solution may result in better or worse results. However, accepting only changes leading to better solutions may result in a local minimum as depicted in Figure B.1. Changes in the opposite direction should be possible during the optimization process to be able to reach the globally best solution. Thus, the idea of Simulated Annealing is to start with an initial solution and allow small changes of this solution. At the beginning (high temperature) the probability to accept a slightly worse solution (up-hill climbing) is high. However, this probability reduces with time (temperature decrease) to result in a stable minimum at the end.

Several cooling procedures are proposed in the literature including constant cooling ($T(x+1) = a \cdot T(x), a < 1$) and cooling by Lundy and Mees ($T(x+1) = T(x)/(1+$

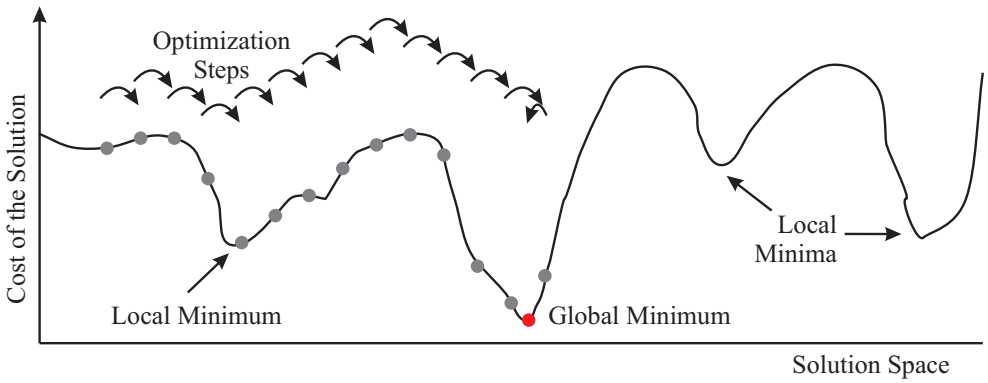


Figure B.1: Local and global minima in the solution space.

$\beta T(x)), \beta \to 0)$.

Possible pseudo-code of Simulated Annealing:

```
Choose initial solution s_c;
Best solution s_b = s_c;
Choose starting temperature: T;
do {
   do {
      Choose random neighbor s in the neighborhood of s_c
      d = f(s) - f(s_c);
      if(d < 0) {
         s_c = s;
         s_b = s;
      }
      else {
         x = random[0..1];
         if(x < exp(-d/(k*T))) then s_c = s;
      }
   } while (coolingCondition != true)
   cool temperature T;
} while (StoppingCondition != true)
Print best solution s_b
```

After the definition of the initial solution and a start temperature, a neighbor solution of the current solution is created. If the optimization criteria improved ($d < 0$), the changed solution is accepted, otherwise the solution is accepted with a certain probability only. Thus, at the beginning almost any changed solution is accepted due to the high temperature (uphill climb). After e.g. a certain amount of iterations, the cooling condition is reached and the temperature is decreased. Finally, the optimization procedure stops, if the stopping condition is reached. Simulated Annealing is a very simple approach that often generates very good results in a reasonable amount of time. However, starting temperature

and cooling conditions have to be chosen carefully. For this, much experience and several tests are required to find good values for a specific problem instance.

B.2 Genetic Algorithm

Genetic Algorithms emulate evolution processes of biological organisms. According to the principles of natural selection and survival of the fittest, the organisms evolve from generation to generation. The idea to apply these natural processes to optimization problems was first introduced by Holland in 1975 [Hol75]. A solution of an optimization problem is considered an individual with specific characteristics. In biology, these characteristics are coded as DNA sequence. According to this sequence, every individual has specific survivability or fitness characteristics that can be considered as cost value. Weak individuals become extinct whereas individuals with higher fitness characteristics reproduce with other individuals to form a new generation. Thus, good gene sequences are likely to survive and the proportion of good characteristics in child populations increases.

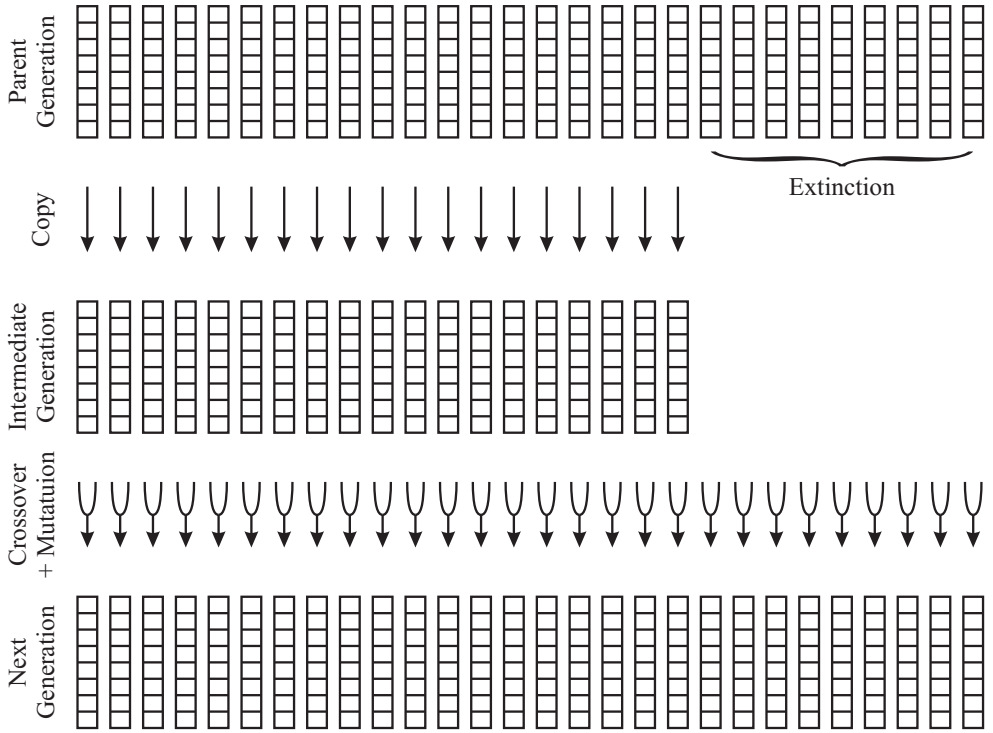


Figure B.2: Selection process of Genetic Algorithms.

Many different individuals form a generation as illustrated in Figure B.2. As in biology, individuals with a good fitness rate (strong individuals) are more likely to find reproduction partners. Thus, in a first step individuals are selected according to their fitness value. E.g. the probability that an individual with a high fitness rate is selected is proportional to its fitness value. Following this selection process, DNA combinations of two individuals form

a new individual of a next generation. Additionally, mutation processes of DNA sequences are possible and certain DNA values can be changed (Figure B.3). Finally, after some generations, a good solution (individual) is found.

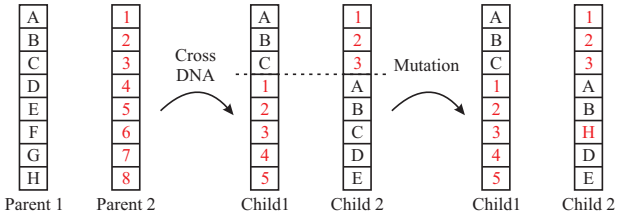


Figure B.3: Example of DNA combination and mutation in Genetic Algorithms.

Possible pseudo-code of Genetic Algorithm:

```
Initialize parent-population with N individuals;
Choose one individual s of the parent-population as best solution s_b;
do {
   Calculate the fitness value f(s) of each individual s;
   for each individual s of the parent-population {
      d = f(s) - f(s_b);
      if(d < 0) s_b = s;
   }
   for each individual s of the parent-population {
      p = f(s) / f(s_b);
      x = random[0..1];
      if(x < p) add s to the selected-population;
   }
   for N chosen pairs of individuals t_1 and t_2 of the selected-population {
      child individual c = CrossDNA(t_1,t_2);
      c = Mutate(c);
      add individual c to the child population;
   }
} while (StoppingCondition != true)
Print best solution s_b
```

Certainly, Genetic Algorithm is a little more complex than Simulated Annealing. However, GA approaches are widely deployed in e.g. economic modeling and market trading due to its efficiency and faster running times. Dependent on the optimization problem either genetic algorithm or simulated annealing approaches perform better. However, the coding of properties in DNA strings, selection, crossover and mutation properties have to be chosen carefully. Similarly to Simulated Annealing, much experience and several tests are required to find good values for a specific problem instance.

Appendix C

Used Sets, Variables, and Parameters

C.1 Sets

$\mathbb{N}$	Nodes of the physical network.
$\mathbb{E}$	Edges of the physical network ($\in \mathbb{N} \times \mathbb{N}$).
$\mathbb{S}$	Status of the network during different failure patterns. Including the failure-free state s_0.
$\mathbb{F}$	Failure patterns i.e. (failing edges or nodes).
$\mathbb{D}$	Demand-relations between two physical nodes ($\in \mathbb{N} \times \mathbb{N}$).
$\mathbb{D}_D$	Demand-relations between two physical nodes ($\in \mathbb{N} \times \mathbb{N}$) that use dedicated resilience. I.e. capacity on backup paths is only be shared between the same working path.
$\mathbb{D}_{Dsp}$	Demand-relations between two physical nodes ($\in \mathbb{N} \times \mathbb{N}$) that use dedicated resilience but can share capacity between multiple working paths of the same demand.
$\mathbb{D}_S$	Demand-relations between two physical nodes ($\in \mathbb{N} \times \mathbb{N}$). that use shared resilience. I.e. capacity on backup paths can be shared between disjoint working paths of demands.
$\mathbb{I}_{\text{WSplit}}$	Multipath indices of a demand that is split into different working paths ($i \in [1..\text{MaxWSplit}^I$).
$\mathbb{I}_{\text{RSplit}}$	Multipath indices of a working path that is split into different backup paths ($j \in [1..\text{MaxRSplit}^I]$)).
$\mathbb{P}_{d,s}$	Possible working path of a demand d during failure state s.
$\mathbb{P}'_{d,s,p}$	Possible resilience path of a demand d and path p during failure state s.

C.2 Variables and Parameters

C_e^D	real	$e \in \mathbb{E}$	The maximum available ***c***apacity on an edge e.
D_d^D	real	$d \in \mathbb{D}$	The traffic value of a ***d***emand d.
$Detour_{d,i,e,s}^B$	bool	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $e \in \mathbb{E}$, $s \in \mathbb{S}$	Indicator if the backup ***detour*** is in front of edge e along working path i of demand d for failure pattern f. Forced to be zero if the detour is in front.
$DRCE_e^D$	real	$e \in \mathbb{E}$	The maximum ***d***edicated backup (***r***esilience) ***c***apacity on ***e***dge e.
$DRCES_{e,s}^D$	real	$e \in \mathbb{E}$, $s \in \mathbb{S}$	The used ***d***edicated ***r***esilience ***c***apacity on ***e***dge e during a specific failure pattern (network ***s***tate) s.
$EMax_{d,i,s}^I$	int	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $s \in \mathbb{S}$	Working path index on which to return to the working path for regional resilience mechanisms.
$EndN_{d,i,n,s}^B$	bool	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $n \in \mathbb{N}$, $s \in \mathbb{S}$	Indicator if the ***n***ode can be ***end*** of a detour.
$IWN_{d,n}^D$	real	$d \in \mathbb{D}$, $n \in \mathbb{N}$	***I***ncoming traffic of ***w***orking paths of demand d on a physical ***n***ode n.
$IWPN_{d,i,n}^I$	int	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $n \in \mathbb{N}$	Number of ***i***ncoming ***w***orking traffic ***p***arts i of demand d on physical ***n***ode n.
$IWPN_{d,i,n}^D$	real	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $n \in \mathbb{N}$	***I***ncoming ***w***orking traffic ***p***art i of demand d on physical ***n***ode n.
$IRPWPNS_{d,i,n,j,s}^D$	real	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $n \in \mathbb{N}$, $j \in \mathbb{I}_{\text{RSplit}}$, $s \in \mathbb{S}$	Number of ***i***ncoming backup (***r***esilience) traffic ***p***arts i of demand d on ***n***ode n in network ***s***tate s.
$IRWPNS_{d,i,n,s}^D$	real	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $n \in \mathbb{N}$, $s \in \mathbb{S}$	***I***ncoming backup (***r***esilience) capacity for ***w***orking traffic ***p***art i on ***n***ode n in network ***s***tate s.
$k_{d,s}^D$	real	$d \in \mathbb{D}$, $s \in \mathbb{S}$	Survivability value. Defines what fraction of the demand survives in network state s.
Max^D	real		A large positive number.
MaxFront^I	int		Parameter for regional resilience mechanisms.
MaxBack^I	int		Parameter for regional resilience mechanisms.
$MaxRSplit_d^I$	int	$d \in \mathbb{D}$	***Max***imum allowed number of backup (***r***esilience) path ***split***s.

$MaxWSplit^I_d$	int	$d \in \mathbb{D}$	***Max***imum allowed number of ***w***orking demand ***split***s.
$MaxWPI^I_{d,i}$	int	$d \in \mathbb{D}$	***Max***imum used ***w***orking ***p***ath index of demand d and part i.
MinBack^I	int		Parameter for regional resilience mechanisms.
MinFront^I	int		Parameter for regional resilience mechanisms.
$WPI^I_{d,i,e}$	int	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$	***Min***imum used ***w***orking ***p***ath index of demand d and part i.
$OWN^D_{d,n}$	real	$d \in \mathbb{D}$, $n \in \mathbb{N}$	***O***utgoing ***w***orking traffic of demand d on a physical ***n***ode n.
$OWPN^D_{d,i,n}$	real	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $n \in \mathbb{N}$	***O***utgoing ***w***orking traffic ***p***art i of demand d on physical ***n***ode n.
$OWPN^I_{d,i,n}$	int	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $n \in \mathbb{N}$	Number of ***o***utgoing ***w***orking traffic ***p***arts i of demand d on physical ***n***ode n.
$ORWPNS^D_{d,i,n,s}$	real	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $n \in \mathbb{N}$, $s \in \mathbb{S}$	The ***o***utgoing ***r***esilience (backup) capacity for a ***w***orking path ***p***art i at a ***n***ode n during network ***s***tate s.
$ORWPNS^D_{d,i,n,s}$	real	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $n \in \mathbb{N}$, $s \in \mathbb{S}$	The ***o***utgoing ***r***esilience (backup) capacity for a ***w***orking path ***p***art i at a ***n***ode n during network ***s***tate s.
$RCDS^D_{d,s}$	real	$d \in \mathbb{D}$, $s \in \mathbb{S}$	The used ***r***esilience ***c***apacity for ***d***emand d that is required in network ***s***tate s.
RCE^D_e	real	$e \in \mathbb{E}$	The (maximum) required ***r***esilience (backup) ***c***apacity on ***e***dge e.
$RCEW^D_{d,e}$	real	$d \in \mathbb{D}$, $e \in \mathbb{E}$	The (maximum) required ***r***esilience (backup) ***c***apacity of demand d on ***e***dge e.
$RCEWS^D_{d,e,s}$	real	$d \in \mathbb{D}$, $e \in \mathbb{E}$, $s \in \mathbb{S}$	The (maximum) required ***r***esilience (backup) ***c***apacity of demand d on ***e***dge e in network ***s***tate s.
$RCEWPS^D_{d,i,e,s}$	real	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $e \in \mathbb{E}$, $s \in \mathbb{S}$	The ***r***esilience (backup) capacity of ***w***orking traffic ***p***art i on ***e***dge e in network ***s***tate s.
$RCPS^D_{d,s,p,p'}$	real	$d \in \mathbb{D}$, $s \in \mathbb{S}$, $p \in \mathbb{P}_d$, $p' \in \mathbb{P}_{d,s,p}$	The required ***r***esilience (backup) ***c***apacity on ***p***ath p' that protects path p of demand d in network ***s***tate s.

$RPU^B_{d,s,p,p'}$	bool	$d \in \mathbb{D}$, $s \in \mathbb{S}$, $p \in \mathbb{P}_d$, $p' \in \mathbb{P}_{d,s,p}$	The indicator if a ***r***esilience (backup) ***p***ath p' of working path p of demand d is ***u***sed (capacity > 0) during network ***s***tate s.
$RPUC^I_{d,s,p}$	int	$d \in \mathbb{D}$, $s \in \mathbb{S}$, $p \in \mathbb{P}_d$	***C***ounter how many ***r***esilience ***p***aths are ***u***sed for demand d and working path p.
$RPWPES^B_{d,i,e,j,s}$	bool	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $e \in \mathbb{E}$, $j \in \mathbb{I}_{\text{RSplit}}$, $s \in \mathbb{S}$	Flow indicator that will be one if $RPWPES^D_{d,i,e,j,f} > 0$.
$RPWPES^D_{d,i,e,j,s}$	real	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $e \in \mathbb{E}$, $j \in \mathbb{I}_{\text{RSplit}}$, $s \in \mathbb{S}$	Backup (***r***esilience) traffic ***p***art j protecting ***w***orking part i of demand d on physical ***e***dge e in network ***s***tate s.
$SMin^I_{d,i,s}$	int	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $s \in \mathbb{S}$	Working path index on which the detour to backup paths can occur.
$StartN^B_{d,i,n,s}$	bool	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $n \in \mathbb{N}$, $s \in \mathbb{S}$	Indicator if the node can be start of a detour.
$SRCE^D_e$	real	$e \in \mathbb{E}$	The ***s***hared ***r***esilience (backup) ***c***apacity on an ***e***dge e in network ***s***tate s.
$SRCES^D_{e,s}$	real	$e \in \mathbb{E}$, $s \in \mathbb{S}$	The required ***s***hared ***r***esilience ***c***apacity on ***e***dge e during a specific ***f***ailure pattern (network state s).
UCE^D_e	real	$e \in \mathbb{E}$	The (maximum) ***u***sed ***c***apacity on ***e***dge e.
$UCES^D_{e,s}$	real	$e \in \mathbb{E}$, $s \in \mathbb{S}$	The ***u***sed ***c***apacity on ***e***dge e in network ***s***tate s.
$UWCES^D_{e,s}$	real	$e \in \mathbb{E}$, $s \in \mathbb{S}$	The ***u***sed ***w***orking ***c***apacity of demand d on edge e in network state s.
$UWCPES^D_{d,i,e,s}$	real	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $e \in \mathbb{E}$, $s \in \mathbb{S}$	The ***u***sed ***w***orking ***c***apacity on path ***p***art i on edge e in network state s.
$WCDS^D_{d,s}$	real	$d \in \mathbb{D}$, $s \in \mathbb{S}$	The required ***w***orking ***c***apacity for ***d***emand d that is required in network ***s***tate s.
WCE^D_e	real	$e \in \mathbb{E}$	The (maximum) required ***w***orking ***c***apacity on ***e***dge e.
$WCES^D_{e,s}$	real	$e \in \mathbb{E}$, $s \in \mathbb{S}$	The required ***w***orking ***c***apacity on ***e***dge e in case of ***f***ailure pattern s.
$WCP^D_{d,p}$	real	$d \in \mathbb{D}$, $p \in \mathbb{P}_d$	The required ***w***orking ***c***apacity on ***p***ath p for demand d.
$WPU^B_{d,p}$	bool	$d \in \mathbb{D}$, $p \in \mathbb{P}_d$	The indicator if a ***w***orking ***p***ath p of demand d is ***u***sed (capacity > 0).
$WPUC^I_d$	int	$d \in \mathbb{D}$	***C***ounter how many ***w***orking ***p***aths are ***u***sed for demand d.
$WPE^B_{d,i,e}$	bool	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $e \in \mathbb{E}$	Flow indicator that will be one if $WPE^D_{d,i,e} > 0$ and zero otherwise.

$WPE^D_{d,i,e}$	real	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $e \in \mathbb{E}$	***W***orking traffic ***p***art i of demand d on physical ***e***dge e.
$WPI^I_{d,i,e}$	int	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $e \in \mathbb{E}$	***I***ndex along the ***w***orking path ***p***art i of demand d. Increases by one on each edge.
$WPILS^I_{d,i,s}$	int	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $s \in \mathbb{S}$	***I***ndex of the first failing edge (***l***eft) along ***w***orking path ***p***art i of demand d in network ***s***tate s.
$WPIRS^I_{d,i,s}$	int	$d \in \mathbb{D}$, $i \in \mathbb{I}_{\text{WSplit}}$, $s \in \mathbb{S}$	***I***ndex of the last failing edge (***r***ight) along ***w***orking path ***p***art i of demand d in network ***s***tate s.
$WSplit^I_d$	int	$d \in \mathbb{D}$	The used number of ***w***orking demand ***splits*** of demand d.

List of Figures

List of Tables

Abbreviations

ANSI	American National Standards Institute
APS	Automatic Protection Switching
ATIS	Alliance for Telecommunications Industry Solutions
ATM	Asynchronous Transfer Mode
BER	Bit Error Rate
BFD	Bidirectional Forwarding Detection
BFS	Breadth First Search
BGP	Border Gateway Protocol
BLSR	Bidirectional Line Switched Ring
BMBF	Bundesministerium für Bildung und Forschung
CAPEX	Capital Expenditures
CGE	Carrier-Grade Ethernet
CTP	Convergence Time of a Protocol
DSPP	Demandwise Shared Path Protection
DTU	Denmarks Tekniske Universitet
DWDM	Dense WDM
ECMP	Equal Cost Multi Path
ETSI	European Telecommunications Standards Institute
FCC	Federal Communications Commission
FEC	Forwarding Equivalence Class
FIB	Forwarding Information Base
FIT	Failures In Time
GA	Genetic Algorithm
GML	Graph Modeling Language
GR	Global Restoration
GUI	Graphical User Interface
HWD	Hardware Design
IEEE	Institute of Electrical and Electronics Engineers, Inc.
IETF	The Internet Engineering Task Force

IGP	Interior Gateway Protocol
ILP	Integer Linear Program
ITU-T	International Telecommunication Union, Telecommunication Standardization Sector
KING	BMBF and Siemens project entitled "Key Components of the Mobile Internet of Next Generation"
LDP	Label Distribution Protocol
LER	Label Edge Router
LP	Linear Program
LSA	Link State Advertisement
LSR	Label Switched Router
LSU	Link State Update
MEF	Metro Ethernet Forum
MILP	Mixed Integer Linear Program
MPLS	Multi Protocol Label Switching
MTBF	Mean Time Between Failures
MTTR	Mean Time To Repair
NOC	Network Operations Center
OAM	Operation, Administration, and Maintenance
OPEX	Operational Expenditures
OSPF	Open Shortest Path First
PBB	Provider Backbone Bridging
PBB-TE	Provider Backbone Bridging - Traffic Engineering
PBT	Provider Backbone Transport
PCE	Path Computational Element
PSTN	Public Switched Telephone Network
QoS	Quality of Service
RCF	Resilience Classification Framework
RCT	Route Convergence Time
RFC	Request For Comments
RSVP	Resource Reservation Protocol
RWA	Routing and Wavelength Assignment
SBPP	Shared Backup Path Protection
SDH	Synchronous Digital Hierarchy
SE2EPP	Shared End-to-end Path Protection
SL2EPP	Shared Local-to-egress Path Protection
SLLPP	Shared Local Link Path Protection
SNMP	Simple Network Management Protocol

SONET	Synchronous Optical Network
SPF	Shortest Path First
SRG	Shared Risk Group
SRLG	Shared Risk Link Group
SRPP	Shared Regional Path Protection
SSH	Secure Socket Shell
T-MPLS	Transport Multi Protocol Label Switching
TCP	Transmission Control Protocol
TPA	Terminal Pair Availability
TPAB	Terminal Pair Available Bandwidth
TPAQoS	Terminal Pair Available QoS
TUM	Technische Universität München
VDSL	Very High Speed Digital Subscriber Line
VLAN	Virtual Local Access Network
VLAN-XC	VLAN Cross-Connect
WDM	Wavelength Division Multiplex
XML	Extensible Markup Language
ZIB	Zuse Institut Berlin

Bibliography

[ABG+01] D. Awduche, L. Berger, D. Gan, T. Li, V. Srinivasan, and G. Swallow. RSVP-TE: Extensions to RSVP for LSP Tunnels. Request For Comments - RFC 3209, Internet Engineering Task Force - IETF, http://www.ietf.org, Dec 2001.

[ACCC03] P.J. Aduskevicz, R. Callon, and W. Hall (Co-Chairs). Final Report. Report, Network Reliability and Interoperability Council IV - Focus Group 2 - Network Reliability, Nov 2003.

[ADF+01] L. Andersson, P. Doolan, N. Feldman, A. Fredette, and B. Thomas. LDP Specification. Request For Comments - RFC 3036, Internet Engineering Task Force - IETF, http://www.ietf.org, Jan 2001.

[**AG06**] A. Autenrieth and C.G. Gruber. Verwendung von Ende-zu-Ende-Verfügbarkeitsrechnung beim Verbindungsaufbau. Offenlegungsschrift DE 10 2004 036 260 A1 2006.03.23, Bundesrepublik Deutschland, Deutsches Patent und Markenamt, Mar 2006.

[AJY00] C. Alaettinoglu, V. Jacobson, and H. Yu. Toward Millisecond IGP Convergence. In *20th North American Network Operators' Group (NANOG) meeting*, Washington D.C., U.S.A., Oct 2000. `http://www.nanog.org/mtg-0010/igp.html`.

[All00] Alliance for Telecom Industry Solutions. ATIS Telecom Glossary 2000. http://www.atis.org/tg2k/, 2000.

[AMA+99] D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. McManus. Requirements for Traffic Engineering Over MPLS. Request For Comments - RFC 2702, Internet Engineering Task Force - IETF, http://www.ietf.org, Sep 1999.

[AMO93] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin. *Network Flows - Theory, Algorithms and Applications*. Prentice-Hall Inc, Upper Saddle River, New Jersey 07458, USA, Apr 1993.

[ANS95] ANSI. Synchronous Optical Network (SONET) - Automatic Protection Switching. *ANSI T1.105.01-1995*, 1995.

[ANS01] ANSI. Technical Report on Enhanced Network Survivability Performance. *ANSI T1.TR.68-2001*, Feb 2001.

[Aut02] A. Autenrieth. *Differentiated Resilience in IP-Based Multilayer Transport Networks*. Technische Universität München, Germany, Sept 2002.

[BA02] E. Blanton and M. Allman. On making TCP more robust to packet reordering. *ACM SIGCOMM Computer Communication Review*, 32(1):20–30, Jan 2002.

[Ber98] D.P. Bertsekas. *Network Optimization: Continous and Discrete Models*. Athena Scientific, P.O.Box 805, Nashua, NH 03061-0805, U.S.A., May 1998.

[**BGS05**] S. Butenweg, C.G. Gruber, and T. Schwabe. Verfahren für ein Inter-Domain Mehrwege-Routing. Offenlegungsschrift DE 103 35 335 A1 2005.03.10, Bundesrepublik Deutschland, Deutsches Patent und Markenamt, Mar 2005.

[Bha99] R. Bhandari. *Surivivable Networks - Algorithms for Diverse Routing*. Kluwer Academic Publishers, 101 Philip Drive, Norwell, MA 02061, USA, 1999.

[BJ01] A. Basu and J.G.Riecke. Stability Issues in OSPF Routing. In *Proc. of ACM SIGCOMM*, pages 225–236, San Diego, U.S.A., Aug 2001.

[BM96] D. Banerjee and B. Mukherjee. A Practical Approach for Routing and Wavelength Assignment in Large Wavelength-Routed Optical Networks. *IEEE Journal on Selected Areas in Communications (JSAC)*, 14(5):903–908, Jun 1996.

[BP01] A.B. Brown and D.A. Patterson. To Err is Human. In *Proc. of the First Workshop on Evaluating and Architecting System Dependability (EASY 2001)*, Goteborg, Sweden, Jul 2001. `http://roc.cs.berkeley.edu/papers/easy01.pdf`.

[BPSM+06] T. Bray, J. Paoli, C.M. Sperberg-McQueen, e.Maler, and Francois Yergeau. Extensible Markup Language (XML) 1.0 (Fourth Edition). Technical report, World Wide Web Consortium, http://www.w3.org, Aug 2006.

[BS02] S. Butenweg and T. Schwabe. Resilience Requirements of Users, Applications and Transport Control. Report wp12-Report-ResReq-vr1-3, Research project KING - Key Components of Mobile Internet of Next Generation, Jun 2002.

[BS04] Byte and Switch. SANs on MANs, Storage Rides the Metro. `http://www.byteandswitch.com/document.asp?doc_id=46661&print=true`, Jan 2004.

[BT97] B. Bertsimas and J.N. Tsitsiklis. *Introduction to Linear Optimization*. Athena Scientific, PO 805, Nashua, NH 03061-0805, USA, 1997.

[BZB+97] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin. Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification. Request For Comments - RFC 2205, Internet Engineering Task Force - IETF, http://www.ietf.org, Sep 1997.

[Cab00] CableLabs. VoIP Availability and Reliability Model for the PacketCable Architecture. Technical Report PKT-TR-VoIPAR-V01-001128, CableLabs, 2000. `http://www.cablelabs.com/specifications/archives/pkt-tr-voipar-v01-0011%28.pdf`.

[CFM99] R. Coltun, D. Ferguson, and J. Moy. OSPF for IPv6. Request For Comments - RFC 2740, Internet Engineering Task Force - IETF, http://www.ietf.org, Dec, 1999.

[**CG06**] J. Charzinski and C.G. Gruber. Abschätzung des Bandbreitenbedarfs in einem Kommunikationsnetz mit Zugangskontolle. Deutsche Patentschrift DE 10 2004 055 722 B3 2006.06.08, Deutsches Patentamt, Jun 2006.

[CGLS01] M. Clouqueur, W.D. Grover, D. Leung, and O. Shai. Mining the Rings: Strategies for Ring-to-Mesh Evolution. In *Proc. of the 3rd International Workshop on the Design of Reliable Communication Networks (DRCN 2001)*, pages 113–120, Budapest, Hungary, Oct. 7-10 2001.

[**CGWW06**] J. Charzinski, C.G. Gruber, U. Walter, and M. Winter. Automatische Nachführung von Netzparametern bei Veränderung der Verkehrslast. Deutsche Patentschrift DE 10 2004 045 980 B3 2006.05.18, Deutsches Patentamt, May 2006.

[Cho05] G. Choudhury. Prioritized Treatment of Specific OSPF Version 2 Packets and Congestion Avoidance. Request For Comments - RFC 4222, Internet Engineering Task Force - IETF, http://www.ietf.org, Oct 2005.

[Chv83] V. Chvátal. *Linear Programming*. W.H. Freeman and Company Ltd, Sep 1983.

[Cis01] Cisco. AutoBandwidth Allocator for MPLS Trafc Engineering - A Unique New Feature of Cisco IOS Software. White paper, Cisco Systems, Inc., San Jose, CA, U.S.A., 2001.

[Cis03] Cisco. Advanced topics in MPLS-TE deployment. White paper, Cisco Systems, Inc., San Jose, CA, U.S.A., Dec, 2003.

[Cis06] Cisco Systems Inc. Company Homepage. http://www.cisco.com, 2006.

[CMU03] E. Calle, J.L. Marzo, and A. Urra. Protection performance components in MPLS networks. In *Proc. of International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS 2003)*, Montreal, Canada, Jul 2003.

[CNRS98] E. Crawley, R. Nair, B. Rajagopalan, and H. Sandick. A Framework for QoS-based Routing in the Internet. Request For Comments - RFC 2386, Internet Engineering Task Force - IETF, http://www.ietf.org, Aug 1998.

[Com93] Federal Communications Commission. Section 63.100 of U.S. Code of Federal Regulations rules (47 CFR 63.100). Report, Federal Communications Commission, 1993. `http://www.fcc.gov/oet/info/rules`.

[Cra93] D. Crawford. Fiber Optics Cable Dig-ups, Causes and Cures. In *Network Reliability: A Report to the Nation - Compendium of Technical Papers*, National Engineering Consortium, Chicago, Jun 1993.

[Dan63] G.B. Dantzig. *Linear Programming and Extensions*. Princeton University Press, Jun 1963.

[Det06] Detecon International GmbH. NetWorks. http://www.networks.detecon.com, 2006.

[DG01] J. Doucette and W. Grover. Comparison of Mesh Protection and Restoration Schemes and the Dependency on Graph Connectivity. In *Proc. of the 3rd International Workshop on the Design of Reliable Communication Networks (DRCN 2001)*, pages 221–228, Budapest, Hungary, Oct 2001.

[Dij59] E.W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Matematik*, 1:269–271, 1959.

[DR00a] B. Davie and Y. Rekhter. *MPLS - Technology and Applications*. Morgan Kaufmann Publishers, 340 Pine Street, Sixth Floor, San Francisco, CA 94104-3205, USA, Mar 2000.

[DR00b] B. Davie and Y. Rekhter. *MPLS Technology and Applications*. Morgan Kaufmann Publishers, 340 Pine Street, Sixth Floor, San Francisco, CA 94104-3205, USA, Jan 2000.

[DSS03] P. Datta, M. Sridharan, and A.K. Somani. A Simulated Annealing Approach for Topology Planning and Evolution of Mesh-Restorable Optical Networks. In *Proc. of the 7th IFIP Working Conference on Optical Networks Design and Modelling*, Budapest, Hungary, Feb 2003.

[Dun94] J. Duncanson. Inverse Multiplexing. *IEEE Communications Magazine*, 32(4):34–41, Apr 1994.

[DW60] G.B. Dantzig and P. Wolfe. Decomposition Principle for Linear Programs. *Operations Research*, 8(1):101–111, Jan 1960.

[**EGI+03**] T. Engel, C.G. Gruber, A. Iselt, A. Kirstädter, G. Schollmeier, T. Schwabe, and C. Winkler. Report Basic Resilience Mechanisms. Technical Report wp12-dc02-vr1-1-wap, Research project KING - Key Components of Mobile Internet of Next Generation, Jan 2003.

[EMDP06] Ed. E. Mannie and Ed. D. Papadimitriou. Recovery (Protection and Restoration) Terminology for Generalized Multi-Protocol Label Switching (GMPLS). Request For Comments - RFC 4427, Internet Engineering Task Force - IETF, http://www.ietf.org, Mar 2006.

[ETS98] ETSI. Network Aspects (NA); Availability performance of path elements of international digital paths. ETSI Technical Report ETSI EN 300 416-v1.2.1-1998-08, European Communication Standards Institute, Aug 1998.

[ETS03] ETSI. User's Quality of Service Criteria for Internet Access in Europe. ETSI Technical Report ETSI TR 102 276 v1.1.1, European Communication Standards Institute, Oct 2003.

[ETS05] ETSI. Access and Terminals (AT); IP Capable Services for Multimedia Broadband Cable Networks; Availability and Reliability. ETSI Technical Report ETSI TR 101 971 v1.1.1, European Communication Standards Institute, May 2005.

[FAS+06a] D. Fedyk, D. Allan, G. Sunderwood, H. Shah, N. Bitar, A. Takacs, and D. Caviglia. General Discussion of Provider Backbone Transport in 802.1ah Networks. IEEE Document ah-bottorff-pbt-v1-0506, IEEE 802 LAN/MAN Standards Committee, www.ieee802.org/1/files/public/docs2006/ah-bottorff-pbt-v1-0506.pdf, May 2006.

[FAS+06b] D. Fedyk, D. Allan, G. Sunderwood, H. Shah, N. Bitar, A. Takacs, and D. Caviglia. GMPLS control of Ethernet. Internet-Draft draft-fedyk-gmpls-ethernet-pbt-01.txt, Internet Engineering Task Force - IETF, http://www.ietf.org, Oct 2006.

[FCC06] FCC. Federal Communication Commission. http://www.fcc.gov, 2006.

[FCGC01] A. Fei, J. Cui, M. Gerla, and D. Cavendish. A Dual-Tree Scheme for Fault-Tolerant Multicast. In *Proc. of the IEEE International Conference on Communications (ICC 2001), Helsinki, Finland*, Jun 2001.

[FF58] L.R. Ford and D.R. Fulkerson. A Suggested Computation for Maximal Multi-Commodity Network Flows. *Management Science*, 5(1):97–101, Oct 1958.

[FFEB05] P. Francois, C. Filsfils, J. Evans, and O. Bonaventure. Achieving Sub-Second IGP Convergence in Large IP Networks. In *Proc. of ACM SIGCOMM*, pages 35–44, Jul 2005. `http://www.nanog.org/mtg-0010/igp.html`.

[FT02] B. Fortz and M. Thorup. Optimizing OSPF/IS-IS Weights in a Changing World. *IEEE Journal on Selected Areas in Communications*, 20(4), May 2002.

[FVA06] A. Farrel, J.-P. Vasseur, and J. Ash. A Path Computation Element (PCE)-Based Architecture. Request For Comments - RFC 4655, Internet Engineering Task Force - IETF, http://www.ietf.org, Aug 2006.

[GCM+03] A. Groebbens, D. Colle, S. De Maesschalck, I. Lievens, M. Pickavet, P. Demeester, L. Tran, K. Steenhaut, and A. Nowe. Efficient Protection in MPλS Networks Using Backup Trees. *Photonic Network Communications*, 6(3):191–206, Nov 2003.

[**GF06**] C.G. Gruber and J. Frings. Layonics - Company homepage. http://www.layonics.com, 2006.

[GG61] P.C. Gilmore and R.E. Gomory. A Linear Appraoch to the Cutting Stock Problem. *Operations Research*, 9(6):849–859, Nov 1961.

[GG63] P.C. Gilmore and R.E. Gomory. A Linear Appraoch to the Cutting Stock Problem - Part II. *Operations Research*, 11(6):863–888, Nov 1963.

[Gir94] A. Girard. *Routing and Dimensioning in Circuit-Switched Networks*. Addison Wesley, Jun 1994.

[GJT04] A. Gunnar, M. Johansson, and T. Telkamp. Traffic matrix estimation on a large IP backbone: a comparison on real data. In *Proc. of the 4th ACM SIGCOMM conference on Internet measurement*, pages 149–160, Taormina, Sicily, Italy, 2004.

[**GKO+05**] C.G. Gruber, A.M. Koster, S. Orlowski, R. Wessäly, and A. Zymolka. A New Model and a Computational Study for Demand-wise Shared Protection. ZIB-Report 05-55, Zuse Insitute Berlin, Optimization Online, Dec 2005.

[**GKZ+05**] C.G. Gruber, A.M. Koster, A. Zymolka, R. Wessäly, and S. Orlowski. A Computational Study for Demand-wise Shared Protection. In *Proc. of the 5th International Workshop on the Design of Reliable Communication Networks (DRCN 2005)*, Ischia, Italy, Oct 2005.

[**GLS05**] C.G. Gruber, B. Lichtinger, and T. Schwabe. Self Regulating Traffic Distribution - A Distributed Traffic Engineering and Resilience Concept. In *Workshop on High Performance Switching and Routing (HPSR 2005)*, Kowloon, Hong Kong, P.R. China, May 2005.

[Gro04] W.D Grover. *Mesh-Based Survivable Networks: Options and Strategies for Optical, MPLS, SONET and ATM Networking*. Prentice Hall PTR; 1st edition, Upper Saddle River, New Jersey 07458, 2004.

[**Gru03a**] C.G. Gruber. A Comparison of MPLS with KING Hammock Routing. Technical report, Institute Of Communication Networks, Munich University of Technology, Dec 2003.

[**Gru03b**] C.G. Gruber. Resilient Networks with Non-Simple p-Cycles. In *Proc. of the International Conference on Telecommunications (ICT 2003)*, Papeete, Tahiti, French Polynesia, Feb 2003.

[**Gru04**] C.G. Gruber. Bandwidth Requirements of MPLS Protection Mechanisms. In *Proc. of the 7th INFORMS Telecommunications Conference (INFORMS 2004)*, Boca Raton, Florida, U.S.A., Mar 2004.

[**Gru05**] C.G. Gruber. A Comparison on Bandwidth Requirements of Path Protection Mechanisms. In *4th International Conference on Networking (ICN 2005)*, St. Gilles Les Bains, Reunion Island, France, Apr 2005.

[GRWC03] M. Goyal, K.K. Ramakrishnan, and F. Wu-Chi. Achieving Faster Failure Detection in OSPF Networks. In *Proc. of the IEEE International Conference on Communications (ICC 2003), Anchorage, U.S.A.*, May 2003.

[**GS02**] C.G. Gruber and D.A. Schupke. Capacity-efficient Planning of Resilient Networks with p-Cycles. In *Proc. of the 10th International Telecommunication Network Strategy and Planning Symposium (Networks 2002)*, Munich, Germany, Jun 2002.

[**GS04**] C.G. Gruber and T. Schwabe. A Scalable Resilient Inter-Domain Multi Path Routing Concept. In *Proc. of the 7th INFORMS Telecommunications Conference (INFORMS 2004)*, Boca Raton, Florida, U.S.A., Mar 2004.

[**GSB06**] C.G. Gruber, T. Schwabe, and S. Butenweg. Verfahren und Netzknoten für eine sebst-regelnde, autonome und dezentrale Verkehrsverteilung in einem Mehrwege Netz. Europäische Patentschrift EP 1 623 541 B1, Europäisches Patentamt, Sep 2006.

[Har01a] W.C. Hardy. *QoS - Measurement and Evaluation of Telecommunications Quality of Service*. John Wiley and Sons, Ltd, 605 Third Avenue, New York, NY 10158-0012, USA, 2001.

[Har01b] S. Harris. The Tao of IETF - A Novice's Guide to the Internet Engineering Task Force. Request For Comments - RFC 3160, Internet Engineering Task Force - IETF, http://www.ietf.org, Aug 2001.

[HBB+04] R. Hülsermann, S. Bodamer, M. Barry, A. Betker, C. Gauger, M. Jäger, M. Köhn, and J. Späth. A Set of Typical Transport Network Scenarios for Network Modelling. In *Proc. of the 5th ITG-Fachtagung Photonische Netze*, pages 65–72, Leipzig, Germany, 2004.

[Him97] M. Himsolt. GML: A portable Graph File Format. Technical report, Universtät Passau, 94030 Passau, Germany, Dec 1997.

[Hol75] J.H. Holland. Adaption in Natural and Artificial Systems. *MIT Press*, 1975.

[Hou01] G. Houston. Analyzing the Internet's BGP Routing Table. *The Internet Protocol Journal*, 4(1), Mar 2001.

[Hui00] C. Huitema. *Routing on the Internet, 2nd Edition*. Prentice Hall PTR, Upper Saddle River, New Jersey 07458, Jan 2000.

[ICB+04] G. Iannaccone, C.-N. Chuah, S. Bhattacharyya, C. Chuah, and C. Diot. Feasibility of IP Restoration in a Tier-1 Backbone. In *Proc. of IEEE Infocom 2004, Hong Kong*, Mar 2004.

[Ilo06] Ilog Inc. Company Homepage. http://www.ilog.com, 2006.

[Inc06] Wikimedia Foundation Inc. Wikipedia - free online dictionary. http://www.wikipedia.org, 2006.

[Ise98] A. Iselt. Redundancy Domains - A Novel Approach for Surivivable Communication Networks. In P.J. Kühn and Roya Ulrich, editors, *Broadband Communications*, pages 249–260. Chapman and Hall- Thompson Science, 2-6 Boundary Row, London SE1 8HN, UK, 1998.

[Ise99] A. Iselt. *Ausfallsicherheit und unterbrechungsfreies Ersatzschalten in Kommunikationsnetzen mit Redundanzdomänen*. Herbert Utz Verlag, München, 1999.

[IT94] ITU-T. Terms and definitions related to quality of service and network performance including dependability. ITU-T Recommendation E.800, ITU-T, Aug 1994.

[IT97] ITU-T. Interworking of SDH network protection architectures. ITU-T Recommendation G.842, ITU-T, Apr 1997.

[IT98] ITU-T. Types and characteristics of SDH network protection architectures. ITU-T Recommendation G.841, ITU-T, Oct 1998.

[IT99] ITU-T. ATM Protection Switching. ITU-T Recommendation I.630, ITU-T, Feb 1999.

[IT00] ITU-T. Architecture of transport networks based on the synchronous digital hierarchy (SDH). ITU-T Recommendation G.803, ITU-T, Mar 2000.

[IT01] ITU-T. Communications quality of service: Framework and definitions. ITU-T Recommendation G.1000, ITU-T, Nov 2001.

[IT02] ITU-T. Internet Protocol Data Communication Service - IP Packet Transfer and Availability Performance Parameters. ITU-T Recommendation Y.1540, ITU-T, Dec 2002.

[IT03a] ITU-T. Availability Performance Parameters and Objectives for End-to-End International Constant Bit-rate Digital Paths. ITU-T Recommendation G.827, ITU-T, Sep 2003.

[IT03b] ITU-T. Generic Protection Switching - Linear Trail and Subnetwork Protection. ITU-T Recommendation G.808.1, ITU-T, Dec 2003.

[IT03c] ITU-T. Protection Switching for MPLS networks. ITU-T Recommendation Y.1720, ITU-T, Sep 2003.

[IT04] ITU-T. Performance and Availability Parameters. ITU-T Recommendation Y.1561, ITU-T, May 2004.

[IT06a] ITU-T. Characteristics of Transport MPLS Equipment Functional Blocks. ITU-T Recommendation G.8121, ITU-T, Mar 2006.

[IT06b] ITU-T. Network performance objectives for IP-based services. ITU-T Recommendation (pre-published) Y.1541, ITU-T, Feb 2006.

[Joh77] D.B. Johnson. Efficient Algorithms for Shortest Paths in Sparse Networks. *Journal of the ACM*, 24(1):1–13, 1977.

[Jun06] Juniper Networks Inc. Company Homepage. http://www.juniper.net, 2006.

[Kal02] S. Kalra. Availability of IP/MPLS networks / Carrier Class Availability for IP Networks. In *29th North American Network Operators' Group (NANOG) meeting*, Eugene, Oregon, U.S.A., Oct 2002. `http://www.nanog.org/mtg-0210/ppt/sanjay.ppt`.

[Kat00] D. Katz. IS-IS and OSPF - A Comparative Anatomy. In *19th North American Network Operators' Group (NANOG) meeting*, Washington D.C., U.S.A., Jun 2000. `http://www.nanog.org/mtg-0006/katz.html`.

[**KGRB06**] A. Kirstädter, C.G. Gruber, J. Riedl, and T. Bauschert. Carrier-Grade Ethernet for Packet Core Networks. In *Proc. of the 2006 International Conference Asia Pacific Optical Communications (APOC) - invited paper*, Gwangju, Korea, Sep 2006.

[KGV83] S. Kirkpatrick, C.D. Jr. Gerlatt, and M.P Vecchi. Optimization by Simulated Annealing. *Science*, 220:670–680, 1983.

[Kha79] L.G. Khachian. A Polynomial Algorithm in Linear Programming. *Soviet Mathematics Doklady*, 20:191–194, 1979.

[KIWP06] A. Kirstädter, A. Iselt, C. Winkler, and S. Pasqualini. A Quantitative Study on the Influence of ASON/GMPLS on OPEX. *International Journal of Electronics and Communication (AEÜ)*, 60(1):509–520, Jan 2006.

[KK03] D. Katz and K. Kompella. Traffic Engineering (TE) Extensions to OSPF Version 2. Request For Comments - RFC 3630, Internet Engineering Task Force - IETF, http://www.ietf.org, Sep 2003.

[KLY99] S.Y. Kuo, S.-K. Lu, and F.M. Yeh. Determing terminal-pair reliability based on Edge Expansion Diagrams using OBDD. *IEEE Transactions on Reliability*, T-R 48(3):234–246, Mar 1999.

[KNE97] Y. Katsube, K. Nagami, and H. Esaki. Toshiba's Router Architecture Extensions for ATM : Overview. Request For Comments - RFC 2098, Internet Engineering Task Force - IETF, http://www.ietf.org, Feb 1997.

[**KOW+05**] A.M. Koster, S. Orlowski, R. Wessly, A. Zymolka, and C.G. Gruber. Demand-wise Shared Protection Revisited: A new model for survivable network design. In *Proc. of the International Network Optimization Conference (INOC 2005)*, Lisbon, Portugal, Mar 2005.

[KR05] K. Kompella and Y. Rekhter. OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS). Request For Comments - RFC 4203, Internet Engineering Task Force - IETF, http://www.ietf.org, Oct 2005.

[Kuh97] D.R. Kuhn. Sources of Failure in the Public Switched Telephone Network. *IEEE Computer Magazine*, 30(4):31–36, 1997.

[KW06] D. Katz and D. Ward. Bidirectional Forwarding Detection. Internet-Draft draft-ietf-bfd-base-05.txt, Internet Engineering Task Force - IETF, http://www.ietf.org, Jun 2006.

[LAJ98] C. Labovitz, A. Ahuja, and F. Jahanian. Experimental Study of Internet Stability and Wide-Area Backbone Failures. Technical Report CSE-TR-382-98, University of Michigan, 1998. `http://citeseer.ist.psu.edu/labovitz98experimental.html`.

[LG02] M. Laor and L. Gendel. The effect of packet reordering in a backbone link on application throughput. *IEEE Network Magazine*, 16(5):28–36, Sep 2002.

[LI05] M. Lackovic and R. Inkret. Overview of Resilience Schemes in Photonic Transmission Network. In *Proc. of the 7th International Conference on Transparent Optical Networks (ICTON 2005)*, Barcelona, Spain, Jul 2005.

[Mas06] C. Mas. Expenditures Study for Network Operators. In *Proc. of 8th International Conference on Transparent Optical Networks (ICTON 2006)*, Nottingham, United Kingdom, Jun 18-22 2006.

[Mer06] Merit/MichNet Network. Homepage. `http://www.merit.edu`, 2006.

[Met04] Metro Ethernet Forum. Requirements and Framework for Ethernet Service Protection in Metro Ethernet Networks. Technical Specification MEF 2, Metro Ethernet Forum, Feb 2004.

[MI03] R. Muralidhar and K. Ishiguro. Scalable control plane architecture for network equipment. Intel Developer Forum, Sep 2003.

[ML06] M. Medard and S.S. Lumetta. *Wiley Encyclopedia of Engineering - Chapter 1: Network Reliability and Fault Tolerance*. John Wiley and Sons, Ltd, 605 Third Avenue, New York, NY 10158-0012, USA, Online: http://citeseer.ist.psu.edu/646287.html, 2006.

[MMH06] R. Martin, M. Menth, and M. Hemmkeppler. Accuracy and Dynamics of Hash-Based Load Balancing Algorithms for Multipath Internet Routing. In *Proc. of 3rd International Conference on Broadband Networks (BROADNETS 2006)*, San Jose, U.S.A., Oct. 1-5 2006.

[Moy94] J. Moy. Multicast Extensions to OSPF. Request For Comments - RFC 1584, Internet Engineering Task Force - IETF, http://www.ietf.org, Mar 1994.

[Moy98] J Moy. OSPF Version 2. Request For Comments - RFC 2328, Internet Engineering Task Force - IETF, http://www.ietf.org, 1998.

[Moy00] J. Moy. *OSPF - Anatonomy of an Internet Routing Protocol*. Addison Wesley, 6th Printing, 75 Arlington Street, Suite 300, Boston MA 02116, U.S.A., Nov 2000.

[NEH+96a] P. Newman, W. Edwards, R. Hinden, E. Hoffman, F. Ching Liaw, T. Lyon, and G. Minshall. Ipsilon Flow Management Protocol Specification for IPv4. Request For Comments - RFC 1953, Internet Engineering Task Force - IETF, http://www.ietf.org, May 1996.

[NEH+96b] P. Newman, W. Edwards, R. Hinden, E. Hoffman, F. Ching Liaw, T. Lyon, and G. Minshall. Ipsilon Flow Management Protocol Specification for IPv4. Request For Comments - RFC 1954, Internet Engineering Task Force - IETF, http://www.ietf.org, May 1996.

[NEH+96c] P. Newman, W. Edwards, R. Hinden, E. Hoffman, F. Ching Liaw, T. Lyon, and G. Minshall. Ipsilon Flow Management Protocol Specification for IPv4. Request For Comments - RFC 1987, Internet Engineering Task Force - IETF, http://www.ietf.org, Aug 1996.

[Net03] Extreme Networks. Summit 200 series switch installation and user guide. http://www.extremenetworks.com, Dec 2003.

[Net05] Network Reliability Steering Committee (NRSC). Annual Report 2005. http://www.atis.org/ATIS/NRSC/view.htm, Oct 2005.

[NLM96] P. Newman, T.L. Lyon, and G. Minshall. Flow Labelled IP: A Connectionless Approach to ATM. In *Proc. of IEEE INFOCOM 1996*, pages 1251–1260, San Francisco, CA, USA, Mar 24-28 1996.

[NS206] NS2. The Network Simulator - Version 2.27, Information Science Institute, University of Southern Califonia, U.S.A. http://www.isi.edu/nsnam/ns/, 2006.

[OS03] E. Osborne and A. Simha. *Traffic Engineering with MPLS*. Cisco Press - Networking Technology Series, July 2003.

[PA00] V. Paxson and M. Allman. Computing TCP's Retransmission Timer. Request For Comments - RFC 2988, Internet Engineering Task Force - IETF, http://www.ietf.org, Nov 2000.

[Pat02] D. Patterson. A Simple Way to Estimate the Cost of Downtime. In *Proc. of the 16th Systems Administration Conference (LISA 2002), Philadelphia, USA*, November 2002.

[PSA05] P. Pan, G. Swallow, and A. Atlas. Fast Reroute Extensions to RSVP-TE for LSP Tunnels. Request For Comments - RFC 4090, Internet Engineering Task Force - IETF, http://www.ietf.org, May 2005.

[RDK+97] Y. Rekhter, B. Davie, D. Katz, E. Rosen, and G. Swallow. Cisco Systems' Tag Switching Architecture Overview. Request For Comments - RFC 2105, Internet Engineering Task Force - IETF, http://www.ietf.org, Feb 1997.

[RH04] Ed R. Hinden. Virtual Router Redundancy Protocol (VRRP). Request For Comments - RFC 3768, Internet Engineering Task Force - IETF, http://www.ietf.org, Apr 2004.

[Rie04] A. Riedl. *Routing Optimization and Capacity Assignment in Multi-Service IP Networks*. Technische Universitt Mnchen, 2004.

[RM99] S. Ramamurthy and B. Mukherjee. Survivable WDM networks - Part II - Restoration. In *Proc. of the IEEE International Conference on Communications (ICC 1999)*, Vancouver, Canada, Jun 1999.

[Rob99] T.G. Robertazzi. *Planning Telecommunication Networks*. IEEE Press, 445 Hoes Lane, PO 1331, Piscataway, NJ 08855-1331, USA, Jan 1999.

[RSM03] S. Ramamurthy, L. Sahasrabuddhe, and B. Mukherjee. Survivable WDM Mesh Networks. *IEEE Journal of Lightwave Technology*, 21(4):870–883, Apr 2003.

[RVC01] E. Rosen, A. Viswanathan, and R Callon. Multiprotocol Label Switching Architecture. Request For Comments - RFC 3031, Internet Engineering Task Force - IETF, http://www.ietf.org, 2001.

[SAF01] D.A. Schupke, A. Autenrieth, and T. Fischer. Survivability of Multiple Fiber Duct Failures. In *Proc. of the 3rd International Workshop on the Design of Reliable Communication Networks (DRCN 2001)*, pages 213–219, Budapest, Hungary, Oct.-10 2001.

[SBLL06] N. Sprecher, D. Berechya, F. Lingyuan, and J. Liu. GMPLS Control of Ethernet VLAN Cross Connect Switches. Internet-Draft draft-sprecher-gels-ethernet-vlan-xc-01.txt, Internet Engineering Task Force - IETF, http://www.ietf.org, Mar 2006.

[Sch01] J. Schallenburg. Is 50ms Restoration Necessary. In *Presentation to the IEEE Bandwidth Management Workshop IX*, Montebello, Jun 2001.

[Sch05] D.A. Schupke. *Cycle-Based Protection for Optical Transport Networks*. Herbert Utz Verlag, 2005.

[SCK+03] G. Schollmeier, J. Charzinski, A. Kirstädter, C. Reichert, K. Schrodi, Y. Glickmann, and C. Winkler. Improving the Resilience in IP Networks. In *IEEE Workshop on High Performance Switching and Routing (HPSR 2003)*, Torino, Italy, Jun 2003.

[SEKE06] A. Schmid-Egger, A. Kirstädter, and J. Eberspächer. *Trends in Telecommunication Networking - Managing Development and Application of Digital Technologies*. Springer Verlag, Berlin Heidelberg New York, 2006.

[SG01] A. Shaikh and A. Greenberg. Experience in Black-Box OSPF Measurement. In *Proc. of the 1st ACM SIGCOMM Workshop on Internet Measurement*, pages 113–125, San Diego, U.S.A., 2001.

[**SG05**] T. Schwabe and C.G. Gruber. Traffic Variations Caused by Inter-Domain Rerouting. In *Proc. of the 5th International Workshop on the Design of Reliable Communication Networks (DRCN 2005)*, Ischia, Italy, Oct 2005.

[**SGA02**] D.A. Schupke, C.G. Gruber, and A. Autenrieth. Optimal Configuration of p-Cycles in WDM Networks. In *Proc. of the IEEE International Conference on Communications (ICC 2002)*, New York City, NY, U.S.A., Apr 2002.

[**SGGS01**] D.A. Schupke, W.D. Grover, C.G. Gruber, and D. Stamatelakis. p-Cycles: Network Protection with Ring-speed and Mesh-efficiency. In *1st COST270 Workshop on Reliability of Optical Networks, Systems, and Components*, Dubendorf, Switzerland, Dec 2001.

[**SGSP06**] M. Scheffel, C.G. Gruber, T. Schwabe, and R. Prinz. Optimal Multi-Topology Routing for IP Resilience. *AEÜ Journal of Electronics and Communications*, 60(1), Jan 2006.

[SL04] H. Smit and T. Li. Intermediate System to Intermediate System (IS-IS) Extensions for Traffic Engineering (TE). Request For Comments - RFC 3784, Internet Engineering Task Force - IETF, http://www.ietf.org, Jun 2004.

[SM97] G. C. Sacket and C. Y. Metz. *ATM and Multiprotocol Networking*. McGraw-Hill Inc.; 1st edition, New York, USA, Jan 1997.

[Smi06] P. Smith. Internet Routing Table Analysis Update. In *South Asian Network Operators Group Meeting (SANOG) VII*, Mumbai, India, Jan 2006.

[SND06] SNDLIB. Simple Network Description Library, 2006.

[Sos94] J. Sosnosky. Service Applications for SONET DCS Distributed Restoration. *IEEE Journal on Selected Areas in Communications*, 12(1):59–86, Jan 1994.

[SP04] D. Schupke and R. Prinz. Capacity Efficiency and Restorability of Path Protection and Rerouting in WDM Networks Subject to Dual Failures. *Photonic Network Communications*, 8(2):191–207, Aug 2004.

[**SPG$^+$06**] M. Scheffel, R. Prinz, C.G. Gruber, A. Autenrieth, and D. Schupke. Optimal Routing and Grooming for Multilayer Networks with Transponders and Muxponders. In *Proc. of 49th annual IEEE Global Telecommunications Conference (GLOBECOM 2006)*, San Francisco, California, U.S.A., Nov 2006.

[SS99] B. Sanso and P. Soriano. *Telecommunications Network Planning*. Kluwer Academic Publishers, 101 Philip Drive, Norwell, MA 02061, USA, 1999.

[Ste96] S.G. Steinberg. Netheads vs. Bellheads. *Wired Magazine*, 4(10), Oct 1996.

[TCC$^+$05] J. Tapolcai, P. Cholda, T. Cinkler, K. Wajda, A. Jajszczyk, A. Autenrieth, S. Bodamer, D. Colle, G. Ferraris, H. Lonsethagen, I.-E. Svinnset, and D. Verchere. Quality of Resilience (QoR): NOBEL Approach to the Multi-Service Resilience Characterization. In *Proc. of 2nd International Conference on Broadband Networks (BROADNETS 2005)*, pages 328– 1337, Boston, U.S.A., Oct 2005.

[TFP+03] M. Tacca, A. Fumagalli, A. Paradisi, F. Unghvary, K. Gadhiraju, S. Lakshmanan, S.M. Rossi, A. de Campos Sachs, and D.S. Shah. Differentiated Reliability in Optical Networks: Theoretical and Practical Results. *Journal of Lightwave Technology*, 21(11):2576– 2586, Dec 2003.

[TS04] T. Thomadsen and T. Stidsen. Optimal design of hierarchical ring networks. In *INFORMS Telecom*, Richard Petersens Plads, Building 321, DK-2800 Kgs. Lyngby, Mar 2004. Informatics and Mathematical Modelling, Technical University of Denmark, DTU.

[VCD+05] S. Verbrugge, D. Colle, P. Demeester, R. Hülsermann, and M. Jäger. General Availability Model for Multilayer Transport Networks. In *Proc. of the 5th International Workshop on the Design of Reliable Communication Networks (DRCN 2005)*, Naples, Italy, Oct 2005.

[VCP+06] S. Verbrugge, D. Colle, M. Pickavet, P. Demeester, S. Pasqualini, A. Iselt, A. Kistädter, R. Hülsermann, FJ. Westphal, and M. Jäger. Methodology and input availability parameters to calculate OpEx as well as CapEx costs for realistic network scenarios. *Journal of Optical Networking*, 5(6):509–520, Jun 2006.

[VPD04] J.-P. Vasseur, M. Pickavet, and P. Demeester. *Network Recovery: Protection and Restoration of Optical SONET-SDH, and MPLS*. Morgan Kaufmann Publishers, 340 Pine Street, Sixth Floor, San Francisco, CA 94104-3205, USA, 2004.

[VPI06] VPISystems Inc. One Plan. http://www.vpisystems.com, 2006.

[VSFH03] Ed. V. Sharma and Ed. F. Hellstrand. Framework for Multi-Protocol Label Switching (MPLS)-based Recovery. Request For Comments - RFC 3469, Internet Engineering Task Force - IETF, http://www.ietf.org, Feb 2003.

[Wan] Wandl Inc. Network Planning and Analysis Tool (NPAT). http://www.wandl.com.

[WKG+99] D. Williams, S. Kamat, R. Guerin, A. Orda, and T. Przygienda. QoS Routing Mechanisms and OSPF Extensions. Request For Comments - RFC 2676, Internet Engineering Task Force - IETF, http://www.ietf.org, Aug 1999.

[Wol98] L.A. Wolsey. *Integer Programming*. John Wiley and Sons, Inc, USA, July 1998.

[**WOZ+05**] R. Wessäly, S. Orlowski, A. Zymolka, A.M. Koster, and C.G. Gruber. Modeling and Solving Demand-wise Shared Protection (DSP). In *6. ITG-Fachtagung Photonische Netze*, Leipzig, Germany, May 2005.

[WVFB96] R. Woundy, A. Viswanathan, N. Feldman, and R. Boivie. ARIS: Aggregate Route-Based IP Switching. Internet-Draft draft-woundy-aris-ipswitching-00.txt, Internet Engineering Task Force - IETF, http://www.ietf.org, Nov 1996.

[YLK02] F.M. Yeh, H.-Y. Lin, and S.-Y. Kuo. Analyzing Network Reliability with Imperfect Nodes using OBDD. In *Proc. of the IEEE Pacific Rim international symposium on dependable computing (PRDC)*, pages 89–96, Tsukaba Science City, Japan, Dec 2002.

[ZD02] H. Zhang and A. Durresi. Differentiated Multi-Layer Survivability in IP/WDM Networks. In *Proc. of the 8th IEEE-IFIP Network Operations and Management Symposium (NOMS 2002)*, Florence, Italy, Apr 2002.

[ZDL06] R. Zhao, Q. Dai, and R. Lehnert. Planning of Hybrid Fibre-VDSL Access Networks Using Particle Swarm Optimization. In *Proc. of the World Telecommunication Congress 2006*, Budapest, Hungary, May 2006.

[ZGL05] R. Zhao, S. Götze, and R. Lehnert. A Visual Planning Tool for Hybrid Fibre-VDSL Access Networks with Heuristic Algorithms. In *Proc. of the 5th International Workshop on the Design of Reliable Communication Networks (DRCN 2005)*, Naples, Italy, Oct 2005.

For § 6 Promotionsordnung, Technische Universität München: Thesis prepublications of the author are [**SGGS01**, **SGA02**, **GS02**, **Gru03a**, **Gru03b**, **EGI+03**, **GS04**, **Gru04**, **KOW+05**, **Gru05**, **WOZ+05**, **GLS05**, **GKZ+05**, **SG05**, **GKO+05**] and other publications of the author are [**BGS05**, **CGWW06**, **CG06**, **AG06**, **GSB06**, **SPG+06**, **KGRB06**, **SGSP06**].