Konrad-Zuse-Zentrum
für Informationstechnik Berlin

Takustraße 7
D-14195 Berlin-Dahlem
Germany

CLAUS G. GRUBER
ARIE M.C.A. KOSTER
SEBASTIAN ORLOWSKI
ROLAND WESSÄLY
ADRIAN ZYMOLKA

A new model and a computational study for Demand-wise Shared Protection

ZIB-Report 05–55 (December 2005)

A new model and a computational study for Demand-wise Shared Protection

Claus G. Gruber[1] Arie M. C. A. Koster[2] Sebastian Orlowski[2]
Roland Wessäly[2,3] Adrian Zymolka[2]

Abstract

This report combines the contributions to INOC 2005 [20] and DRCN 2005 [9]. A new integer linear programming model for the end-to-end survivability concept demand-wise shared protection (DSP) is presented. DSP is based on the idea that backup capacity is dedicated to a particular demand, but shared within a demand. It combines advantages of dedicated and shared protection: It is more cost-efficient than dedicated protection and operationally easier than shared protection. In a previous model for DSP, the number of working and backup paths to be configured for a particular demand has been an input parameter; in the more general model for DSP investigated in this paper, this value is part of the decisions to take.

To use the new DSP model algorithmically, we suggest a branch-and-cut approach which employs a column generation procedure to deal with the exponential number of routing variables.

A computational study to compare the new resilience mechanism DSP with dedicated and shared path protection is performed. The results for five realistic network planning scenarios reveal that the best solutions for DSP are on average 15% percent better than the corresponding 1+1 dedicated path protection solutions, and only 15% percent worse than shared path protection.

Keywords: demand-wise shared protection, resilience, network design, integer linear programming

1 Introduction

It is of utmost importance for network operators to protect traffic against node and link failures. To this end, many survivability concepts have been proposed. Their applicability depends on the used technology (e.g., MPLS, ATM, SDH, WDM), and each of the concepts has its particular strengths and weaknesses with respect to investment cost, management effort, and recovery performance. Dedicated protection concepts (like 1+1 or $m:n$ dedicated

[1]Institute of Communication Networks, Munich University of Technology, 80290 Munich, Germany E-mail: claus.gruber@tum.de

[2]Zuse Institute Berlin (ZIB), Takustraße 7, D–14195 Berlin, Germany. Email: {koster,orlowski,wessaely,zymolka}@zib.de

[3]atesio GmbH, Sophie-Taeuber-Arp-Weg 27, D-12205 Berlin, Germany

path protection) are easy to implement in terms of network management and provide short recovery times, but the resulting networks are often rather expensive. In contrast, shared protection concepts (like $m : n$ shared path protection) make more efficient use of backup capacities and thus can potentially reduce network cost, at the expense of a more complex network management. In this article, we present a new model for the protection concept demand-wise shared protection (DSP) which combines the main advantages of dedicated and shared protection. It can be expected to be less expensive than dedicated protection and, at the same time, easier to realize than shared protection.

From a network operator's point of view, a survivable routing must fulfill two basic requirements: For each demand, a predetermined demand value has to be satisfied in the failure-free network state, and in any considered failure state, a specified fraction of the demand must survive. Further desirable features of a survivability concept are cost-efficiency, ease of network management, and short failure recovery times. Two approaches in this direction are p-cycles [6] which combine sharing of backup capacity with short recovery times, and a special case of DSP presented in [13] which combines shared backup capacity with ease of network management. Using DSP, backup capacity is *dedicated to* a particular demand, but *shared within* a demand. This concept is promising because

(i) capacity sharing can reduce total network cost compared to purely dedicated protection,

(ii) all paths are pre-established end-to-end, and

(iii) interdependencies between working and backup paths are limited to individual demands.

(a) 1+1 routing with 8 paths (b) DSP routing with 6 paths (c) DSP routing with 5 paths

Figure 1: Different survivable routings.

The concept of sharing backup resources within a demand is illustrated in Figure 1, where four demand units have to be routed between Frankfurt and Leipzig with full protection against single link failures. Figure 1(a) shows a 1+1 protection routing where no backup capacity is shared, leading to eight paths (e.g., lightpaths in WDM or VC-N-paths in SDH) which have to be configured. Figures 1(b) and 1(c) illustrate two DSP routings with six and five configured paths, respectively, where backup capacity is shared by the working paths of the demand. In both routings, any link failure is survived by at least four paths for this demand. The figures illustrate that sharing backup resources within a demand and allowing for more diversity in the routing may lead to fewer paths which have to be configured in total.

In [13], DSP is parameterized in order to balance the total number of paths to establish and the corresponding requirements on the routing (in terms of diversity). Varying the parameters yields different routings, as shown in Figure 1. However, the decision which parameters to apply has to be taken a priori, but it is not clear which of the resulting routings is preferable from a (total) cost perspective. In particular, a minimum number of paths need not be most cost-efficient. In Figure 1, eight paths are needed for 1+1 protection, compared to five or six paths using the shown DSP routings, while the least total number of hops (as a rough approximation of network cost) is needed in Figure 1(b) with six paths and ten hops in total.

The key observation is that the total number of needed (working and backup) paths is not determined by the two basic requirements for survivable routings stated above. So, rather than taking it as an input parameter, determining an optimal path number together with the routing is preferable. To realize this idea, we present a more general model for DSP in this paper. Besides the capacity restrictions, the only routing constraints in the new model are the two basic requirements formulated above (which do not prescribe the final path number), while the total network cost is to be minimized.

As these constraints are also satisfied by all solutions for any purely dedicated protection concept as well as for the DSP model with a fixed number of paths, the new model allows for further cost savings while keeping the operational advantages. On the other hand, capacities designed for DSP are also sufficient to accommodate a purely shared protection routing, whose optimal network cost therefore gives a lower bound for any DSP solution.

A computational study on five realistic network instances presented in Section 5 reveals that the best solutions for DSP are on average 15% percent better than the corresponding 1+1 dedicated path protection solutions, and the best solutions for SPP are not more than 15% percent better than the best DSP solutions. Furthermore, DSP is very beneficial for low protection levels and loses gain as the protection level advances towards 100%.

The article is organized as follows. In Section 2 we review the existing resilience mechanisms and introduce demand-wise shared protection. We compare their characteristics in Section 3. Section 4 presents a mixed-integer programming model and a branch-and-cut-and-price algorithm to compute a minimum cost hardware configuration together with a DSP routing. A computational comparison of the cost and bandwidth requirements of the different concepts is presented and discussed in Section 5. Our conclusions are summarized in Section 6.

2 Protection Mechanisms

In this section, the resilience concepts dedicated path protection, shared path protection, and demand-wise shared protection are specified. These concepts are then discussed with respect to the key properties capacity consumption, required signalization, and recovery time in Section 3.

In the following, a demand refers to a requirement of a number of connections to be established between two nodes in the network. These connections can be routed independently of each other.

2.1 Dedicated Path Protection

To survive a network element failure using 1+1 dedicated path protection [7], information is duplicated at the source of the demand and routed simultaneously along two disjoint paths towards the destination node. It depends on the planning requirements, whether these paths must be disjoint with respect to links, nodes, or other sets of network components.

Figure 3 depicts an example configuration of two demands protected by 1+1 path protection. In the example network, the demand from node A to node K is routed along the paths A-

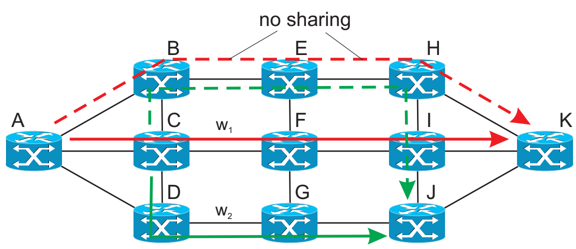


Figure 2: Example of a path configuration for 1+1 dedicated path protection, where traffic is simultaneously sent over working *and* backup path.

C-F-I-K as well as on the node-disjoint path A-B-E-H-K. In case of a single link or node failure, the sink of the demand still receives a copy of the data from one of the two paths.

Notice that by definition of dedicated path protection, the capacity occupied by a single demand cannot be used by another demand.

2.2 Shared Path Protection

For shared path protection [7], information is only sent over the working path in the failure-free state. For different network failure cases, there exist pre-calculated and pre-configured backup paths. Using appropriate signaling mechanisms, these backup paths are then established in reaction upon the failure by the source node of the demand. If working paths are not affected simultaneously by one considered failure pattern, they can share the resilience capacity with each other.

Figure 3 shows an example configuration for two demands protected by shared path protection. In the failure-free state, the demands A-K and C-J are routed on the (working) paths A-C-F-I-K and C-D-G-J only. No copying of data is required. A failure of a network element is detected by neighboring nodes via physical or protocol failure-detection mechanisms (e.g. Bidirectional Forwarding Detection [11]), notifying the source node(s) of the affected demand(s). After receipt of these failure signals, a source node switches the traffic from the failing path onto the pre-defined failure-free backup path (A-B-E-H-K and C-B-E-H-I-J).

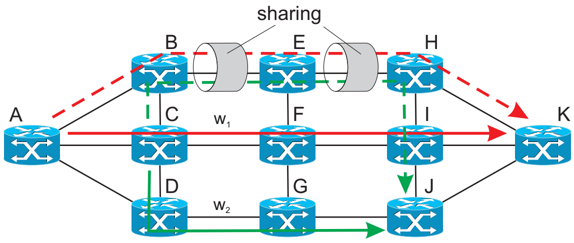


Figure 3: Example of a path configuration for shared path protection, where two demands (A-K and C-J) are protected.

In the example network, both working paths are link-disjoint; hence, if only single link failures are considered, the backup capacity on the subpath B-E-H can be shared between these two paths and the required overall capacity can be reduced.

2.3 Demand-wise Shared Protection

A survivable routing must fulfill two basic requirements: For each demand,

- a predetermined demand value has to be satisfied in the failure-free network state, and
- in any considered failure state, a specified fraction of the demand must survive.

The concept demand-wise shared protection consists of pre-establishing a set of paths for each demand such that those requirements hold. The number of paths is at least the required demand value to enable routing in the failure-free network state. Moreover, the routing is carried out such that in each failure state at least the specified portion of the paths survives. DSP does not dedicate paths to be exclusively for working or backup traffic.

DSP combines advantages of dedicated and shared path protection with each other. As with dedicated path protection, the capacity occupied by a single demand cannot be used by any other demand. Furthermore, as with shared path protection, backup paths are pre-established and only used in case of a network element failure. As main property of DSP, sharing of backup resources is restricted to paths of the same demand and, in consequence, a setup of backup paths (restoration) is not required in case of a failure.

An example of an admissible DSP configuration for the demand between node A and node K is depicted in Figure 4. The two working paths are A-C-F-I-K and A-D-G-J-K. Since they are node-disjoint, both paths can be protected by the backup path A-B-E-H-K.

In a first version of DSP (see [12, 13]), the number of (working and backup) paths which has to be established in total is pre-determined based on connectivity arguments, and given as an input value. Two cases have been considered: (i) exploration of the maximum node-connectivity between every pair of nodes (DSP-MAX), and (ii) exploration of node-connectivity two between every pair of nodes (DSP-TWO).

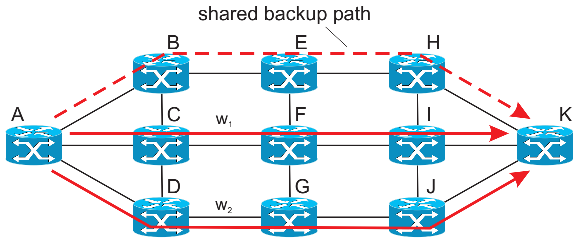


Figure 4: Example of a path configuration for demand-wise shared protection, where backup capacity can be shared between working paths belonging to the same demand.

The above introduced general version of DSP can be realized by explicitly formulating the two basic requirements as linear inequalities, see Section 4. This way, both the total number of paths to be established and their routing remain flexible and is part of the optimization.

3 Comparison

In this section, we discuss and compare qualitative and quantitative properties of the different mechanisms. Table 1 summarizes the reaction times, bandwidth requirements and management complexity of the three protection mechanisms.

	1+1 dedicated path protection	shared path protection	demand-wise shared protection
sharing	no sharing	between any disjoint working paths	between paths within a demand
number of backup paths	one per working path	flexible	flexible
required signalization	none	backward (and forward)	backward
reaction time	very fast	medium	fast
management complexity	small	large	medium

Table 1: Comparison of resilience characteristics.

3.1 Required Capacity

Using 1+1 dedicated path protection, backup capacity is not shared. Each signal is duplicated and therefore the used capacity is more than doubled if the working path is always routed on a shortest-hop path. Notice, however, that it depends on the particular (discrete) capacity/cost structure how much extra investment is necessary. With shared path protection, the capacity of resilience paths can be shared if the working paths are not affected by the same failure pattern. A reduction of up to 70% used capacity can be achieved compared to dedicated path protection [8]. The concept of DSP enables the sharing of resilience capacity. However, the sharing possibilities are reduced compared to shared path protection since capacity can only be shared between working paths of one demand. Thus, the capacity requirements of DSP are bounded from above by 1+1 dedicated path protection and bounded from below by shared path protection.

3.2 Reaction Time

1+1 dedicated path protection requires no signalization to the source or the sink node. From a technology point of view it is possible for the sink node to receive both copies simultaneously. Thus, in case of a network element failure along one path a very fast and almost hitless non-stop operation can be performed.

DSP requires a failure detection and backward signalization before the traffic can be detoured to the surviving paths at the source node of the demand. Hence, only operation at the end nodes is necessary, which makes recovery fast.

In contrast to DSP, resilience capacity can be shared between any disjoint working paths in shared path protection. If the target node is not included in the traffic data (e.g., as it is the case for MPLS), a reconfiguration of switches (e.g., the adjustment of a mirror in DWDM) is required to setup paths for detoured traffic. This adjustment has to be done before the backup path can be used. Thus, the reaction time of shared path protection (or then restoration) requires additional time for the switching adjustment along the backup path.

3.3 Mechanism Complexity

1+1 dedicated path protection requires one resilience path per working path. From a configuration and management point of view this structure is very simple. The configuration, choice, and management of shared resources however is more complicated: the sharing of resources prevents the pre-establishing of backup connections, and thus these connections have to be setup on request. In DSP, all paths are pre-established but their usage depends on the failure state. Due to the guarantee that sufficiently many paths survive in any failure state, the traffic to be protected has to be reassigned to these surviving paths. Hence, no path setup, but only a traffic reorganization in the source node has to be carried out.

4 Mathematical Model for DSP

To quantify the benefits of DSP, we formulate the corresponding network design problem as integer linear program. The model for DSP, as well as the other concepts, is composed of two major blocks: one for the routing which requires link capacities, and one for the hardware configuration (for both nodes and links) which provides these capacities. The goal is to find a cost-minimal network design including

- a topology,

- a hardware configuration, and
- a (dedicated, shared, DSP) protection routing.

For the hardware configuration, the generic integer linear programming (ILP) model presented in [14] is used for the selection of a topology, switches, cards, ports, and link capacities. This hardware model is used for all studied survivability concepts; only the routing model changes. Since the focus of this paper is on the routing model for DSP, we omit the details of the hardware model here and refer to [14] for further information.

In this section, the routing model and the algorithm used for computing DSP solutions are described in more detail. For the other survivability concepts we refer to [18] for dedicated and shared path protection, and to [12] for DSP-MAX and DSP-TWO.

4.1 Routing Model for DSP

The network is modeled as an undirected graph $G = (V, E)$. For the routing part, let $y_e \in \mathbb{Z}_+$ denote the capacity of link $e \in E$, which is to be determined along with the topology, hardware, and routing decisions; these capacities are derived from a given solution of the hardware ILP.

Let $\mathcal{S}$ be the set of *operating states* for which the network is to be designed. In addition to the failure-free state, this set comprises a subset $\mathcal{S}^*$ of *failure states*, each of which is characterized by its failing nodes or links. The set $V^s \subseteq V$ consists of all non-failing nodes in operating state $s \in \mathcal{S}$.

Let $\mathcal{D}$ be a set of point-to-point communication demands. For every demand $uv \in \mathcal{D}$ and every operating state $s \in \mathcal{S}$ in which neither u nor v fails (i. e., $u, v \in V^s$), a *demand value* $d^s_{uv} \in \mathbb{Z}_+$ is specified. In the failure-free state, this value denotes the demand to be routed; in failure states, it is the demand value which must survive in this situation. For ease of notation, we set $d^s_{uv} := 0$ if $u \notin V^s$ or $v \notin V^s$. A demand $uv \in \mathcal{D}$ may be routed on one or more paths from the set $\mathcal{P}_{uv}$, which comprises all simple uv-paths (i.e., without loops). In failure state $s \in \mathcal{S}^*$, the subset $\mathcal{P}^s_{uv} \subseteq \mathcal{P}_{uv}$ denotes all surviving paths. Using non-negative integer flow variables $f_{uv}(P) \in \mathbb{Z}_+$ for all demands $uv \in \mathcal{D}$ and all paths $P \in \mathcal{P}_{uv}$, the routing model reads as follows:

$$\sum_{P \in \mathcal{P}^s_{uv}} f_{uv}(P) \geq d^s_{uv} \qquad uv \in \mathcal{D},\ s \in \mathcal{S} \tag{1}$$

$$y_e - \sum_{uv \in \mathcal{D}} \sum_{\substack{P \in \mathcal{P}_{uv}: \\ e \in P}} f_{uv}(P) \geq 0 \qquad e \in E \tag{2}$$

$$y_e,\ f_{uv}(P) \in \mathbb{Z}_+ \tag{3}$$

The demand constraints (1) and capacity constraints (2) formulate a multicommodity flow problem where it is allowed to route more flow than strictly necessary. Constraints (1) ensure for each demand that at least the specified number of paths survives in any operating state. Notice that extensions of this model, such as hop limits, are straightforward by restricting the set of admissible paths appropriately.

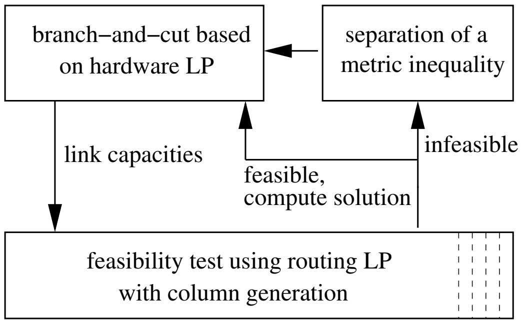


Figure 5: Algorithmic approach.

4.2 A Branch-and-Cut Algorithm with Column Generation

Our proposed solution approach, which has also successfully been applied to problems with other routing formulations [15, 19], is similar to Benders decomposition [2]. The central procedure is a branch-and-cut algorithm (see [17] for a detailed description) based on an LP relaxation which contains only hardware variables and constraints, but no routing information. To strengthen the LP relaxation, cutting planes are separated during the whole branch-and-bound process, such as band inequalities [4], GUB cover inequalities [21], and generalizations [15, 19] of metric inequalities [10].

Each time an integer hardware configuration is identified during the branch-and-bound process, it is tested for feasibility with respect to the (missing) routing constraints. If a feasible routing exists within the corresponding capacities, such a routing is generated. Otherwise, a violated metric inequality is derived from the dual objective function of the routing LP, which is then added to the relaxation in order to cut off the infeasible hardware configuration. This process is illustrated in Figure 5.

To test whether a feasible routing exists within given link capacities $\bar{y}$ derived from the hardware part of the model, the routing LP has to be solved with $\bar{y}$ as right-hand side in the capacity constraints (2). As the routing LP has an exponential number of path variables, column generation has to be employed in order to know whether the LP has a solution or not. Optimality of a given solution can be guaranteed if feasibility of $\bar{y}$ with respect to the routing constraints can be tested exactly. For instance, this is the case if the pricing problem is polynomially solvable. In the remainder of this section, we show that for DSP with single node or link failures and fractional routing path variables, the pricing problem can be solved by shortest path computations and thus in polynomial time. With integer path variables, this approach still leads to a valid lower bound and hence to a quality guarantee for solutions.

Testing a link capacity vector $y := (y_e)_{e \in E}$ for feasibility with respect to the routing constraints can be formulated as the following optimization problem (with the dual variables in brackets) by introducing a capacity excess variable α:

$$
\begin{aligned}
&(\alpha^* :=)\ \min\ \alpha \\
&\sum_{P \in \mathcal{P}^s_{uv}} f_{uv}(P) \geq d^s_{uv} && uv \in \mathcal{D},\ s \in \mathcal{S} && [\pi^s_{uv}] \\
\alpha - &\sum_{uv \in \mathcal{D}} \sum_{\substack{P \in \mathcal{P}_{uv}: \\ e \in P}} f_{uv}(P) \geq -y_e && e \in E && [\mu_e] \\
&\alpha,\ f_{uv}(P) \in \mathbb{R}_+
\end{aligned}
$$

The capacity vector y admits a feasible routing if and only if $\alpha^* = 0$. Suppose that the LP above is initialized with a restricted set of paths which admit a feasible solution for sufficiently large α (for instance, two node-disjoint paths for each demand). Let α^*_r denote the optimal value of such a restricted LP and notice that $\alpha^* \leq \alpha^*_r$.

If $\alpha^*_r = 0$, a feasible routing for the link capacities y has been found, and we are done. Otherwise, $\alpha^*_r > 0$ means that no such feasible routing exists on the restricted path set. It remains to test whether a routing would exist after adding additional path variables to the LP. An improving path variable is indicated by the violation of the corresponding dual constraint with respect to the current dual solution. The dual of the LP above, with the associated primal variables in brackets, is:

$$
\max \sum_{uv \in \mathcal{D}} \sum_{s \in \mathcal{S}} d^s_{uv} \pi^s_{uv} - \sum_{e \in E} y_e \mu_e \tag{4}
$$

$$
\sum_{\substack{s \in \mathcal{S}: \\ P \in \mathcal{P}^s_{uv}}} \pi^s_{uv} - \sum_{e \in P} \mu_e \leq 0 \qquad uv \in \mathcal{D},\ P \in \mathcal{P}_{uv} \qquad [f_{uv}(P)] \tag{5}
$$

$$
\sum_{e \in E} \mu_e \leq 1 \qquad [\alpha] \tag{6}
$$

$$
\mu_e,\ \pi^s_{uv} \in \mathbb{R}_+ \tag{7}
$$

Constraint (6) is always satisfied by the optimal dual solution because α is contained in the restricted primal LP. Therefore, it suffices to test whether all dual constraints (5) are satisfied in order to state optimality of α^*_r with respect to *all* path variables. If $\mathcal{S}^*$ consists of all single link and node failures, this can be done as follows. First, notice that a routing path P fails if and only if it contains the failing link or node. Second, a simple path contains a failing node as inner node if and only if it also contains exactly two incident links of the failing node. Third, w.l.o.g., $\pi^s_{uv} = 0$ can be assumed if $u \notin V^s$ or $v \notin V^s$. Using these observations, the dual constraint (5) of a path P of demand uv can be rewritten as

$$\begin{aligned}
0 \le \sum_{e \in P} \mu_e - \sum_{\substack{s \in \mathcal{S}: \\ P \in \mathcal{P}^s_{uv}}} \pi^s_{uv} &= \sum_{e \in P} \mu_e - \Big(\sum_{s \in \mathcal{S}} \pi^s_{uv} - \sum_{\substack{s \in \mathcal{S}^*: \\ P \notin \mathcal{P}^s_{uv}}} \pi^s_{uv} \Big) \\
&= \sum_{e \in P} \mu_e + \sum_{\substack{s \in \mathcal{S}^*: \\ P \notin \mathcal{P}^s_{uv}}} \pi^s_{uv} - \sum_{s \in \mathcal{S}} \pi^s_{uv} \\
&= \sum_{e = xy \in P} \big(\mu_e + \pi^e_{uv} + \frac{1}{2}(\pi^x_{uv} + \pi^y_{uv}) \big) - \sum_{s \in \mathcal{S}} \pi^s_{uv}
\end{aligned}$$

The last sum only depends on the demand uv, not on the path P. Hence, by defining

$$\gamma_e := \mu_e + \pi^e_{uv} + \frac{1}{2}(\pi^x_{uv} + \pi^y_{uv}) \qquad (e = xy) \qquad \text{and} \qquad l_{uv} := \sum_{s \in \mathcal{S}} \pi^s_{uv},$$

constraint (5) can be rewritten as

$$\gamma(P) \; := \; \sum_{e \in P} \gamma_e \; \ge \; l_{uv}.$$

Since $\gamma_e \ge 0$ for all $e \in E$, violation of this inequality can be tested by searching a shortest uv-path P^* with respect to the link weights γ_e with Dijkstra's algorithm [5]. If $\gamma(P^*) < l_{uv}$, then P^* violates its dual constraint and can be added to the primal LP. Otherwise, if the shortest uv-path P^* satisfies its dual constraint $\gamma(P^*) \ge l_{uv}$, all other uv-paths satisfy their dual constraints (5) as well.

Summarizing, the decision whether a feasible (fractional) DSP-routing exists can be done in polynomial time by solving $|\mathcal{D}|$ shortest path problems. Although the separation of metric inequalities is based on a fractional routing, the described approach still leads to a valid lower bound on the whole planning problem with an integer routing. In fact, a capacity vector y is only cut off by a separated inequality if no fractional routing exists within the given capacities, which means that no integer routing exists either. If, on the other hand, a fractional routing but no integer one can be found, y is declared to be infeasible without being cut off. In this way, an optimal solution may not be found, but the overall lower bound is still valid and leads to a quality guarantee for any feasible solution.

For generating an integer routing out of a fractional one, a combination of rounding and rerouting techniques is used. In a first step, the routing formulation is solved as an integer program whose set of variables is restricted to those paths which have a non-zero flow in the fractional routing. The objective is to minimize the total flow in the network. Even if no integer routing exists within the given capacities using the restricted path set, the routing is made integer by rounding up some flow values. In any case, the resulting integer routing is post-processed by a min-cost-flow heuristic within the given capacities. It tries to shift flow away from links with a small flow-to-capacity ratio, in order to free these links. This post-processing aims to set the capacity on some links to zero and thus to a smaller total network cost. On our test instances, the running time of the whole heuristic is usually less than half a second per iteration.

5 Case Study

In order to compare the survivability concepts rather than the codes used to generate feasible solutions, it is desirable to have provably optimal solutions for the various resilience mechanisms at hand. Integer linear programming is a very powerful approach in order to obtain high-quality solutions for network design problems. Realistic network scenarios, however, are typically too complex to be solved to proven optimality within reasonable time bounds. Nevertheless, it is still possible to draw valuable information from the results if upper and lower bounds on the optimal network cost are available which are not too far apart.

In this section, we compare the new DSP variant from Section 4 to

- the first DSP version [12, 13], where the number of paths to be used is set to the maximum node-connectivity (DSP-MAX) or to two (DSP-TWO).
- 1+1/1:1 protection as an upper bound for DSP,
- shared path protection (SPP) as a lower bound for DSP.

The approach for the concepts DSP-MAX and DSP-TWO is similar to the approach described in Section 4. The minimum cost network design with 1+1 protection can be formulated similarly to the DSP variants and is not explained further.

The integer linear programming formulation for SPP is more complex in the sense that the flow variables $f_{uv}(P)$ have to be indexed for each operating state, resulting in an order of magnitude more flow variables [16]. Moreover, our postprocessing techniques to generate integer routings cannot be applied in this context. As a consequence, in contrast to all other concepts, solutions of SPP may contain fractional flows. Thus, the difference between the values for DSP and SPP is not only due to the more flexible routing mechanism of SPP, but also to the relaxation to fractional flows. Moreover, in case of SPP we could not always finish the column generation completely (i.e., solve linear programming relaxations for the constraints at hand to optimality) which implies that the lower bound presented is only guaranteed under the precondition of a restricted path-set $\mathcal{P}_{uv}$ for all demands.

All computations follow the algorithmic approach presented in Section 4.2. As underlying LP solver, CPLEX 9.1 [1] has been used. A time bound of three hours of CPU time is used for all computations.

5.1 Instances compared

A total of 5 network instances has been used in this case study. Table 2 shows some characteristics for the topology, demands, and operating states. Besides the failure-free state, we consider all single link and node failures as operating states (i.e., $|\mathcal{S}| = |V|+|E|+1$). The networks NSFNET, GERMANY and EUROPE have been used before in the study of the first DSP version and originate from the MultiTeraNet project [3]. The GERMANY-EXT network bases on the GERMANY network, but contains five more links as to increase the connectivity and by this extend the solution space for the various concepts (in particular for DSP). The P23 network is a modified version of a network of one of our industrial partners. All instances are available from the authors upon e-mail request.

instance	$\lvert V\rvert$	$\lvert E\rvert$	$\lvert\mathcal{D}\rvert$	$\sum d^s_{uv}$	$\lvert\mathcal{S}\rvert$
NSFNET	14	21	91	2710	36
GERMANY	17	26	58	686	44
GERMANY-EXT	17	31	58	686	49
EUROPE	28	41	378	1008	70
P23	11	34	24	4621	46

Table 2: Instance characteristics

The network designs for NSFNET, GERMANY, GERMANY-EXT and EUROPE have been computed with an optical equipment cost model, see [12] for details. For P23, a SDH equipment cost structure is used.

For every network instance, we consider three scenarios: 50%, 75%, and 100% protection of all traffic requirements. Fractional survivability requirements are rounded up as to guarantee that at least the percentage of the traffic survives. Combined with the 5 different networks, the 3 protection scenarios result in a total of 15 instances used in the case study.

5.2 Results

The results of our computational comparison are presented in Figures 6–8 for 50%, 75%, and 100% traffic protection, respectively. Each of these figures shows lower and upper bounds on the network cost for each of the networks and each of the concepts. The values are normalized according to the upper bound for 1+1 protection. In this way, the savings by DSP solutions to 1+1 solutions can easily be detected.

For NSFNET the gap between lower and upper bounds is roughly $2-3\%$, which allows more concise statements than for the other networks. P23 turns out to be the most difficult one of our test set.

Besides the cost figures, we also can compare the number of paths actually routed in the various designs. For 1+1 protection, DSP-MAX, and DSP-TWO, these numbers are fixed beforehand, but for DSP the number of paths is part of the optimization. Figure 9 shows this number of paths for the P23 network, the different concepts, and the various protection levels.

5.3 Discussion

From the figures, we can observe that DSP is very beneficial for low protection levels and loses gain as the protection level advances towards 100%. For 50% and 75% protection, the best solution by DSP is in many cases below the lower bound for 1+1 protection, which demonstrates that DSP is indeed a more cost-effective survivability concept than 1+1 protection regardless of optimality of the solutions. On average over all versions (50%, 75%, 100%), the best solutions for DSP are 15% percent better than the corresponding 1+1 dedicated path protection solutions.

Compared to SPP, the DSP solutions for NSFNET with 50% protection do not differ much

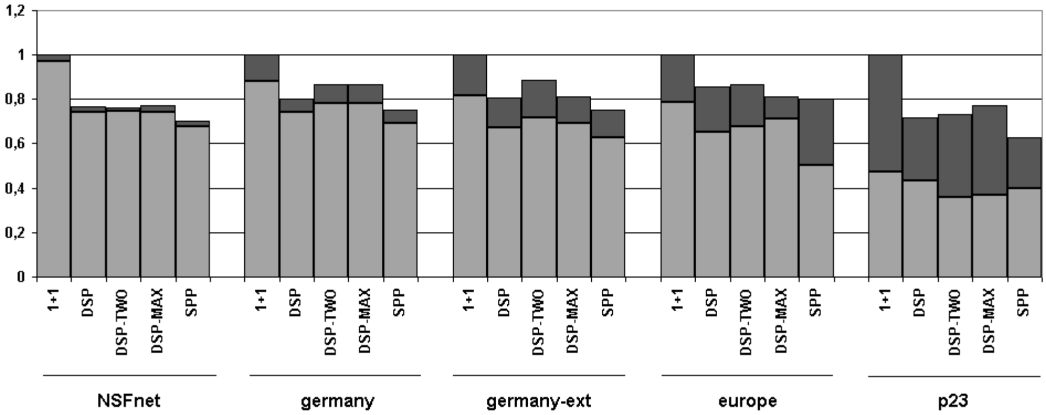


Figure 6: Relative network cost values and lower bounds for **50% protection**

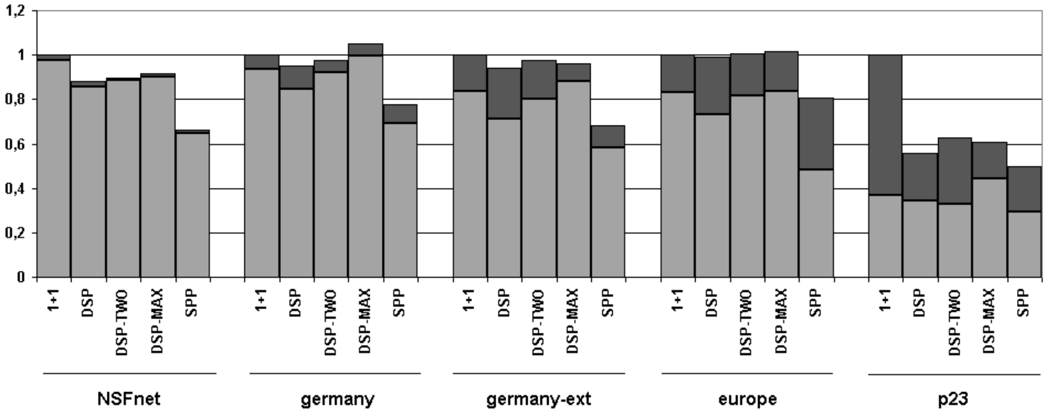


Figure 7: Relative network cost values and lower bounds for **75% protection**

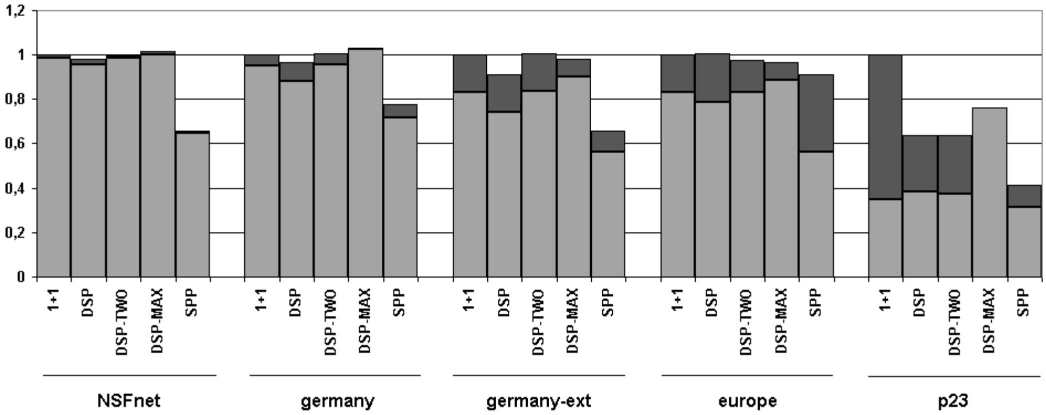
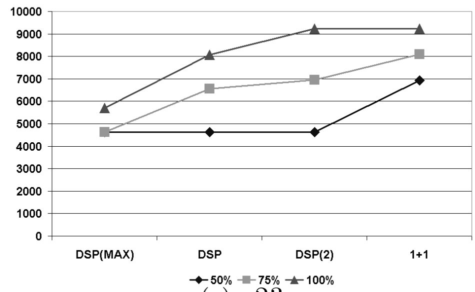
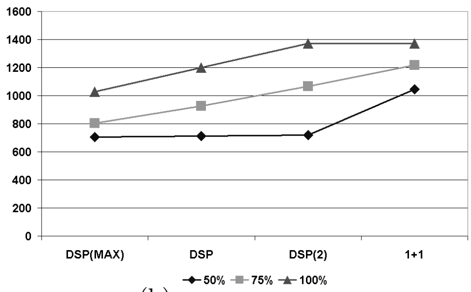


Figure 8: Relative network cost values and lower bounds for **100% protection**



(a) P23



(b) GERMANY-EXT

Figure 9: Number of paths in designs for various concepts

from the SPP values, indicating that in this case DSP is close to what can be achieved by shared protection mechanisms. The increasing gap between lower and upper bounds (both for DSP and SPP) for the other instances does not allow for more general statements in this direction. This is also the case for higher protection levels (e.g., the upper bound for DSP and the lower bound for SPP grow apart). On average over all versions (50%, 75%, 100%), the best solutions for SPP are an additional 15% percent better than the best DSP solutions.

The flexibility of DSP compared to DSP-MAX and DSP-TWO is reflected in both the cost values and the number of paths routed. In most cases, the DSP solution has a value somewhat below the solution values of both DSP-MAX and DSP-TWO. In fact, for 75% and 100% protection, the lower bound of DSP-MAX is in several cases higher than the upper bound of DSP indicating that DSP outperforms DSP-MAX on network cost. For 50% protection, this effect cannot be observed: The number of paths used for a demand in DSP is in between DSP-MAX and DSP-TWO. For 50% protection, these numbers are often equal, and hence the flexibility in a DSP design compared to DSP-MAX is very limited (cf. Figure 9).

Also, the lower bound is typically lower than those of DSP-MAX and DSP-TWO. Although the existence of (significantly) better solutions could be one reason, the higher complexity of solving DSP is to some extent responsible for this: the number of branch-and-bound nodes processed within the three hours of computation time is significantly lower for DSP than for DSP-MAX and DSP-TWO. The higher complexity of DSP is also the reason that in a single case the upper bound of DSP is above DSP-MAX, DSP-TWO as well as 1+1 protection.

Among the three DSP variants, DSP-MAX turns out to be the best computable, i.e., the gap between lower and upper bounds is typically the smallest after three hours of computation time. The requirement in DSP-MAX to use many disjoint paths seems to be responsible for this: the number of disjoint path combinations is typically small if the connectivity between two nodes is high in comparison to the number of nodes in the network. This is in particular the case in P23 with 100% protection: DSP-MAX could be solved to optimality in only two seconds. A high spreading of the traffic routing hardly allows to leave links unused. In connection with a coarse granularity of installable capacities, their effective utilization is hampered by the limited routing flexibility. This explains why DSP-TWO and DSP find significantly better solutions than the optimal DSP-MAX network, although the number of established paths is about 65% higher (cf. Figure 8 and Figure 9).

6 Conclusion

Demand-wise Shared Protection is a promising approach to protect a network against single element failures. It combines the good characteristics of shared path protection and dedicated path protection: good reaction times, good bandwidth requirements and a simple network management.

A quantification by integer linear programming showed how expensive solutions for DSP are compared to other protection mechanisms: DSP is an excellent alternative to 1+1 dedicated path protection, in particular if not 100% of all demands needs to be protected. Average cost savings in the order of around 21% have been identified for 50% protection, while still around 15% over all scenarios.

Solving the integer programming models for DSP turns out to be computationally more complex. In order to design proven minimum cost networks for DSP in reasonable short computation times, further improvements of the mathematical optimization approach are therefore needed.

References

[1] CPLEX Version 9.1. http://www.ilog.com.

[2] J. Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4(1):238–252, 1962.

[3] A. Betker, C. Gerlach, R. Hülsermann, M. Jäger, M. Barry, S. Bodamer, J. Späth, C. Gauger, and M. Köhn. Reference transport network scenarios. Technical report, BMBF MultiTeraNet, July 2003. http://www.ikr.uni-stuttgart.de/IKRSimLib/Referenz_Netze_v14_full.pdf.

[4] G. Dahl and M. Stoer. A polyhedral approach to multicommodity survivable network design. *Numerische Mathematik*, 68(1):149–167, 1994.

[5] E.W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.

[6] W. Grover and D. Stamatelakis. Cycle oriented distributed preconfiguration: Ring-like speed with mesh-like capacity for self-planning network restoration. In *Proceedings of the IEEE International Conference on Communications 1998 (ICC'98),* Atlanta, USA, pages 537–543, June 1998.

[7] W. D. Grover. Mesh-based Survivable Networks, Options and Strategies for Optical, MPLS, SONET and ATM Networking. In *Prentice Hall PTR, Upper Saddle River, New Jersey, U.S.A.*, 2003.

[8] C. G. Gruber and al. A Comparison of MPLS with KING Hammock Routing. http://www.lkn.ei.tum.de, Report of the research project KING, December 2003.

[9] C. G. Gruber, A. M. C. A. Koster, S. Orlowski, R. Wessäly, and A. Zymolka. A computational study for Demand-wise Shared Protection. In *Proceedings of DRCN 2005*, pages 421–428, Naples, Italy, 2005. Design of Reliable Communication Networks.

[10] M. Iri. On an extension of the maximum-flow minimum-cut theorem to multicommodity flows. *Journal of the Operations Research Society of Japan*, 13(3):129–135, 1971.

[11] D. Katz and D. Ward. Bidirectional Forwarding Direction. draft-ietf-bfd-base-02.txt, IETF Network Working Group, March 2005.

[12] A. M. C. A. Koster, A. Zymolka, M. Jäger, and R. Hülsermann. Demand-wise shared protection for meshed optical networks. *Journal of Network and Systems Management*, 13(1):35–55, 2005.

[13] A. M. C. A. Koster, A. Zymolka, M. Jäger, R. Hülsermann, and C. Gerlach. Demand-wise Shared Protection for Meshed Optical Networks. In *Proceedings of DRCN 2003, the Fourth International Workshop on the Design of Reliable Communication Networks, Banff, Canada*, pages 85–92, October 2003.

[14] A. Kröller and R. Wessäly. Integrated optimization of hardware configuration and capacity dimensioning in SDH and opaque WDM networks. In *Proceedings of the First International Network Optimization Conference (INOC 2003), Paris*, pages 349–354, October 2003.

[15] S. Orlowski. Local and global restoration of node and link failures in telecommunication networks. Diploma thesis, Technische Universität Berlin, February 2003. Available from `http://www.zib.de/orlowski`.

[16] S. Orlowski and R. Wessäly. Comparing restoration concepts using optimal network configurations with integrated hardware and routing decisions. *Journal of Network and System Management*, 13(1):99–118, March 2005.

[17] M. W. Padberg and G. Rinaldi. A branch and cut algorithm for the resolution of large–scale symmetric traveling salesman problems. *SIAM Review*, 33:60–100, 1991.

[18] M. Pióro and D. Medhi. *Routing, Flow, and Capacity Design in Communication and Computer Networks.* Morgan Kaufmann Publishers, 2004.

[19] R. Wessäly. *Dimensioning Survivable Capacitated NETworks.* PhD thesis, Technische Universität Berlin, April 2000.

[20] R. Wessäly, S. Orlowski, A. Zymolka, A. M. C. A. Koster, and C. G. Gruber. Demand-wise Shared Protection revisited: A new model for survivable network design. In *Proceedings of the second International Network Optimization Conference, Lisbon, Portugal*, pages 100–105, March 2005.

[21] L.A. Wolsey. Valid inequalities for 0/1 knapsacks and MIPs with Generalized Upper Bound constraints. *Discrete Applied Mathematics*, 29:251–261, 1990.