Computer Networks 95 (2016) 35–50

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Routing optimization for IP networks with loop-free alternates

Matthias Hartmann[a], David Hock[a], Michael Menth[b,*]

[a] Institute of Computer Science, University of Würzburg, Würzburg, Germany
[b] Department of Computer Science, University of Tübingen, Tübingen, Germany

ARTICLE INFO

Article history:
Received 20 June 2014
Revised 1 October 2015
Accepted 4 November 2015
Available online 21 December 2015

Keywords:
IP fast reroute
Loop-free alternates
Routing optimization
Coverage
Maximum link utilization
Traffic engineering

ABSTRACT

Loop-free alternates (LFAs) have been developed for fast reroute (FRR) in intradomain IP networks. They are simple, standardized, and already offered by several vendors. However, LFAs have two major drawbacks. They often cannot provide failure protection against all single link or node failures in spite of physical connectedness, and some LFAs cause routing loops in scenarios with node or multiple failures.

LFAs may be applied for various reasons that we call applications in this work. We propose several definitions for LFA coverage that quantify the application-specific utility of LFAs available in the network. The availability of LFAs and whether they can cause routing loops heavily depend on the IP routing which is determined by the choice of administrative IP link costs. To maximize the benefit of LFA usage, we optimize the IP link costs using LFA coverage as objective function. We demonstrate the feasibility and effectiveness of that approach in several test networks, and show that the choice of the right optimization function is crucial to maximize LFA coverage. However, maximizing LFA coverage can lead to significant traffic imbalance and may result in high link loads. Therefore, we suggest Pareto-optimization and demonstrate that resulting link costs can lead to both high LFA coverage and low link loads.

1. Introduction

In IP networks, failures occur on a regular basis and often last only for a short time [1]. The distributed IP rerouting process is simple and robust [2], but it may be too slow for applications and services that require continuous network availability [3]. Recently, fast reroute (FRR) mechanisms have been proposed for IP networks [4]. With IP-FRR, a router can detour traffic around a failure location immediately after it has detected that the regular next-hop is no longer reachable. This reduces the time during which packets are lost from several seconds down to less than 50 ms. Then, regular IP rerouting is triggered. Therefore, the traffic affected by the failure is forwarded by IP-FRR mechanisms only until the rerouting process completes or the failure disappears.

The only IP-FRR mechanism that is already standardized by the IETF and implemented in new routers, e.g., current versions of Cisco IOS and Juniper OS, is the loop-free alternates (LFAs) concept [5]. An LFA is an alternate next-hop to which certain traffic can be sent without creating any loops so that this traffic reaches its destination over an alternative path. When the regular next-hop for a certain destination is no longer reachable by a router, it can deflect traffic to this destination over the LFA. LFAs do not require any signaling, they do not require changes to the basic IP routing protocol, and they do not require tunneling. These features facilitate incremental as well as partial

* Corresponding author. Tel.: +49 7071 2970505.
E-mail address: menth@uni-tuebingen.de (M. Menth).

http://dx.doi.org/10.1016/j.comnet.2015.11.009
1389-1286/

deployment, even in a multi-vendor network, and make LFAs a very attractive solution. However, LFAs have also two disadvantages. First, nodes may not have LFAs for all destinations [6–8] so that some traffic cannot be protected against single link or node failures although the network topology has alternate working paths. Second, some LFAs may cause extra-loops in case of node or multiple failures. An extra-loop is a forwarding loop caused by LFAs where packets loop between two or more nodes. This can even overload links and routers that are otherwise unaffected by the failure.

There are various incentives for the use of LFAs in IP networks. We call them applications and consider several of them. We argue that the utility of available LFAs depends on the application and measure the utility by application-specific LFA coverages. Some examples:

- LFA coverage can be measured by the fraction of destinations that each node can protect by LFAs, averaged over all nodes. This is an intuitive definition that nicely reflects the availability of LFAs in a network and was used for that purpose in most existing studies on LFAs. However, it does not relate to any specific application.
- One goal of IP-FRR is to reduce traffic loss between failure detection and the completion of the rerouting process. This is reflected by the fraction of the traffic that is lost due to missing LFAs, averaged over all considered failures. We use that as indirect measure for LFA coverage.
- Network providers can sell improved availability guarantees if traffic is protected by LFAs on its entire path so that only marginal traffic is lost in case of a failure. Thus, the LFA coverage may be quantified by the fraction of traffic for which the entire path can be protected by LFAs.
- If all flows carried over a link can be protected by LFAs, that link may fail without losing any traffic after LFA activation. As a consequence, IP rerouting may be delayed when such a link fails and graceful reconvergence techniques [9–12] can be utilized that prevent micro-loops. For short-lived link failures or maintenance operations, IP rerouting that can lead to routing instabilities and micro-loops, may be avoided even twice: once when the link goes down and once when it comes up again. For these applications, the LFA coverage may be expressed by the fraction of links for which all traffic carried under failure-free operation can be protected by LFAs.

We further diversify the definitions of LFA coverage with regard to the types of LFAs that may be used: all LFAs or only those that cannot create extra-loops. The relevance of avoiding temporary extra-loops is certainly application-specific.

The availability of LFAs and the LFA coverage obviously depend on the network topology and the routing. Thus, LFA coverage may be increased by changing the topology: additional (physical or virtual) links may be installed which provide LFAs that can be used during failures [13,14]. LFA coverage can also be increased by changing the routing by configuration of appropriate administrative link costs that determine the path layout in IP networks [15,16].

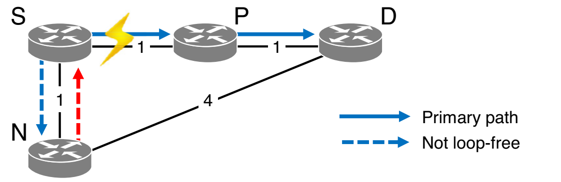


Fig. 1. Neighbor N cannot be used as LFA because it does not meet the loop-free condition.

In this work, we investigate the different definitions of LFA coverage in test networks with uniform link costs. We further apply these definitions as objective functions to optimize link costs in order to maximize LFA coverage. We show that this approach is feasible by achieving significant improvements in LFA coverage. However, tweaking link costs influences not only LFA coverage but also traffic distribution within the network. We show that maximizing LFA coverage can lead to significantly increased link loads both under failure-free conditions and after rerouting in failure cases so that traffic may be lost due to overload. That is not acceptable since these phases persist longer than the short rerouting interval for which LFAs reduce packet loss. Hence, maximization of LFA coverage can be counterproductive. To fix that problem, we propose Pareto-optimization to generate a set of link costs that are Pareto-optimal with regard to LFA coverage and maximum link loads. Some of these link costs lead to relatively high LFA coverage and relatively low maximum link loads so that a network administrator can choose appropriate ones to configure the network.

The remainder of this paper is structured as follows. Section 2 explains LFAs and Section 3 gives an overview of related work. Section 4 discusses various applications of LFAs that require different definitions of LFA coverage, and the potential of routing optimization is illustrated. Section 5 shows that there is a tradeoff between high LFA coverage and low link loads and suggests Pareto-optimization to find good compromises. Finally, Section 6 summarizes this work. A table with acronyms and notation is provided in Table 10 of the Appendix.

2. Loop-free alternates

LFAs provide fast protection for IP networks using link state routing protocols. They are intended to be used by a node immediately after it has detected a failure until the failure disappears or until IP rerouting has converged. In this section we review the definition of LFAs [5]. As general LFAs may cause extra-loops under some conditions, we define three sets of LFAs that avoid extra-loops to a different extent.

2.1. General or link-protecting LFAs

We consider a *source node* S and a *next-hop* P on a shortest path toward *destination* D, just like in Fig. 1, but with a link cost less than 3 for the link from N to D. In this scenario, another *neighbor node* N of S can be used by S as LFA to D for the potential failure of the link $S \rightarrow P$

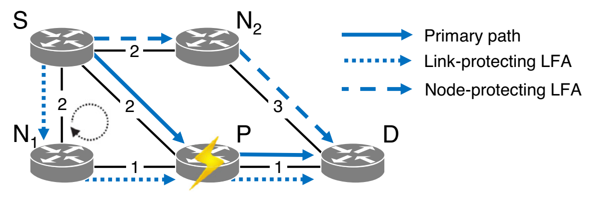


Fig. 2. Only the node-protecting LFA N_2 can be used to protect against the failure of node *P*.

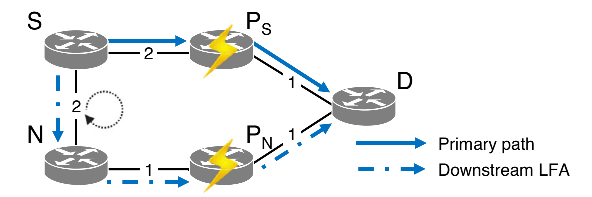


Fig. 3. Neighbor *N* is a downstream LFA of *S* but not vice-versa. The use of only downstream LFAs avoids loops in the presence of multiple failures.

when the shortest path from *N* to *D* does not contain *S*. To avoid loops, the following loop-free condition must be met:

$$\mathrm{dist}(N, D) < \mathrm{dist}(N, S) + \mathrm{dist}(S, D), \quad (1)$$

whereby dist(*A*, *B*) denotes the least cumulative cost on a path between *A* and *B*. If link $S \to P$ fails, *S* detours the traffic destined to *D* via LFA *N*, and from *N* the deviated packets take the shortest path toward *D*. Fig. 1 shows that such an LFA does not always exist. When link $S \to P$ fails, packets can only be rerouted to neighbor *N*. However, this creates a forwarding loop because the shortest path from *N* to *D* leads over *S*. Therefore, *N* cannot be used as LFA by *S* to protect against the failure of link $S \to P$. As node *S* does not have any other neighbor, this example shows that LFAs cannot protect all traffic against single link failures.

2.2. Node-protecting LFAs

In Fig. 2 both neighbors N_1 and N_2 of source *S* fulfill the loop-free condition with regard to destination *D* and can serve as LFAs to protect against the failure of the link $S \to P$. Now, we consider the failure of node *P*. If node *S* reroutes traffic to the alternate neighbor N_1, the next-hop is again *P* so that N_1 uses *S* as LFA, returns the traffic, and an extra-loop occurs. Therefore, N_1 cannot be used by *S* as LFA to protect against the failure of node *P*, but N_2 can be used for that purpose. A neighbor node *N* must meet the following node protection condition to protect destination *D* as LFA in case that node *P* fails:

$$\mathrm{dist}(N, D) < \mathrm{dist}(N, P) + \mathrm{dist}(P, D) \quad (2)$$

An LFA meeting only the loop-free condition is called link-protecting while an LFA also meeting the node protection condition is called node-protecting. Since the node protection condition implies the loop-free condition [17], every node-protecting LFA is also link-protecting, but not vice-versa.

2.3. Downstream LFAs

We consider source *S* and destination *D* in Fig. 3. *N* provides a node-protecting LFA for *S* and vice-versa. If two nodes P_S and P_N fail simultaneously, *S* reroutes its traffic to *N*. Node *N* cannot forward the traffic, either, and reroutes it to *S* so that an extra-loop occurs. Such loops may happen during multiple failures and can be avoided if an LFA fulfills the downstream condition

$$\mathrm{dist}(N, D) < \mathrm{dist}(S, D). \quad (3)$$

An LFA fulfilling this condition is called downstream LFA. Allowing only downstream LFAs guarantees loop avoidance for all failure cases because packets always get closer to the destination. In Fig. 3, *N* is a downstream LFA for *S* but not vice-versa.

2.4. Use of LFAs

LFAs are pre-computed and installed in the forwarding information base (FIB) of a router. Normally, this is done for each destination so that we speak of per-prefix LFAs. If an LFA can protect all traffic for a specific next-hop, it may be used as a per-link LFA to simplify forwarding tables. However, per-link LFAs cannot protect as much traffic as per-prefix LFAs, they cannot protect against node failures, and cause forwarding loops in case of some node failures or multiple failures. Therefore, per-prefix LFAs are the preferred mechanism [18] and we study only per-prefix LFAs.

2.5. Loop avoidance classes

For our analysis, we define three different loop avoidance classes (LACs) of LFAs [17].

LP All link-protecting LFAs are used. They may cause extra-loops after node failures or multiple failures.

NP Only node-protecting LFAs are used to protect against failures except for the failure of the last link toward a destination, which may be protected by a link-protecting LFA. By definition, there is no node-protecting LFA for the last link. Due to the potential use of link-protecting LFAs, extra-loops may occur in case of multiple failures or when the destination node fails.

ND Only node-protecting downstream LFAs are used to protect against failures except for the failure of the last link toward a destination, which may be protected by a link-protecting downstream LFA. The selected LFAs do not cause any extra-loops.

The remainder of this work concentrates on the *LP*-LAC and the *ND*-LAC since they excel with the highest LFA coverage or avoidance of any loops, respectively.

3. Related work

Link cost optimization in IP networks has been studied for many years. Fortz and Thorup [19] used it to reduce the utilization of links under normal conditions using an additive objective function taking the load on all links of the network into account. Alternative metrics like the maximum load of a link relative to its capacity are used by other authors. Link costs may also be optimized to minimize relative link loads under normal conditions and in failure scenarios [2,20]. We also presented an algorithm for that task in [21]. We studied and compared various objective functions and introduced the idea of a primary and secondary optimization goal, e.g., maximum relative link load and path length [22]. Equal-cost paths may occur in IP networks which may be good for load balancing purposes, but bad for prediction of load distribution. Therefore, we proposed a method to optimize for unique shortest paths [23]. A large body of related work regarding link cost optimization can be found in these papers.

Multiple fast reroute mechanisms have been developed for IP networks [24,25]: multiple routing configurations [26], failure insensitive routing [27], not-via addresses [28], failure-carrying packets [29], and others. They can protect the network against all single failures as long as the network topology provides alternate paths. Therefore, routing optimization in this context usually aims at minimizing relative link loads during failure-free operation and sometimes also for likely failure scenarios. The authors of [30] minimize link utilization for failure-free conditions while taking care that link capacities suffice to accommodate the backup traffic in all single failures.

LFAs are simpler, easier to implement and deploy than the FRR methods mentioned before, and currently the only IP-FRR solution offered by vendors. However, LFAs often cannot protect all traffic against all single link failures and never against all single node failures even if the network topology provides alternative paths [6–8]. A recent IETF document [31] reports that in typical service provider access networks, all single link failures can be protected by general LFAs. It also analyzes the LFA coverage in several simulated backbone topologies. Another RFC [32] suggests to consider link bandwidths when selecting LFAs. Cisco's software Cariden MATE [33] illustrates and evaluates the LFA coverage. Retvari et al. [14,16] studied the availability of LFAs from a structural point of view, formulated topological prerequisites for high LFA coverage, and provided lower and upper bounds for LFA coverage for certain network structures. All these papers have in common that they consider only general LFAs which may cause loops in case of node failures or multiple failures. In previous work [17] we formulated the three loop avoidance classes, analyzed the LAC-specific LFA coverage, and showed that it heavily depends on link costs.

Trong Viet et al. [15] optimize link costs to maximize the average fraction of protected destinations per node and to minimize the maximum relative link load under failure-free conditions at the same time. In contrast to our work, they do not differentiate between different LFA types, they use only per-link LFAs, and they do not consider the relative link load in failure scenarios. Retvari et al. [14,16] propose a mixed integer program and a heuristic approach to improve the LFA coverage by link cost optimization. They show that the problem is NP-complete, and recently included the protection of node failures as well as lower and upper bounds on LFA coverage in their work [34].

As it may be impossible to achieve full LFA coverage, additions and modifications to LFAs have been proposed. In [17] we considered a combination of LFAs and not-via addresses. Juniper proposes in its LFA implementation guide [13] to increase LFA coverage by adding links or tunnels, e.g., MPLS label switched paths. Also Retvari et al. [14,16] showed that sometimes the addition of a few links significantly increases the availability of general LFAs and makes the network even fully protectable against single link failures. The authors of [35] propose E-LFAs to increase the LFA coverage, but they require protocol changes and they are more complex than normal LFAs, defeating their major advantage over other IP-FRR methods. Another modification of LFAs with the same pros and cons uses failure notifications [36]. Remote LFAs [37] have been recently proposed to extend the coverage of local LFAs. They are pre-installed tunnels and relay traffic to another node in the network from which the traffic can be forwarded to its destination. They are used in failure cases if local LFAs are not available. Like with not-via addresses, the drawback of remote LFAs is the tunneling overhead, but they do not require network-wide coordination. Csikor and Retvari showed that remote LFAs can greatly improve the LFA coverage in well-meshed networks, but they still had to add new IP links to achieve 100% LFA coverage [38].

4. Analysis and optimization of LFA coverage

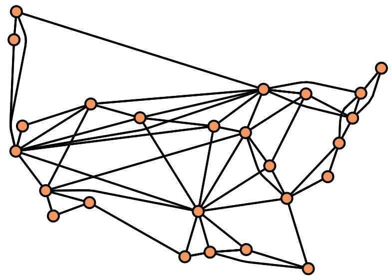
In this section, we first present the networks under study and briefly introduce our link cost optimization method. Then, we introduce various applications of IP-FRR and suggest performance metrics that capture the application-specific LFA coverage. To maximize the LFA coverage, we optimize link costs using various metrics as objectives functions and compare their benefit for specific LFA applications. Our study is LFA-type-aware in the sense that we consider separately general LFAs and LFAs that do not create extra-loops in case of node and multiple failures.

4.1. Networks under study

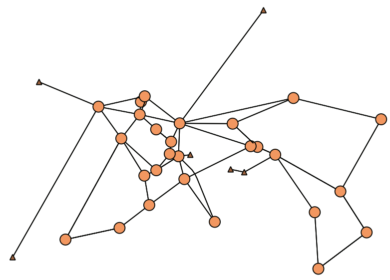
For the evaluation of our algorithms we use several widely used research topologies from the "Topology Zoo" [39,40] and the topology from the Nobel project [41]. They are illustrated in Figs. 4 and 5.

A topology is two-connected if any link or node can be removed without splitting the remaining network into several disconnected islands. As resilience mechanisms require such two-connected topologies to reroute traffic, we removed nodes from the original topologies to make them two-connected in order to simplify our analysis. The removed nodes are drawn as small triangles in Figs. 4(a)–(e), 4(f) and 5(a)–5(d).

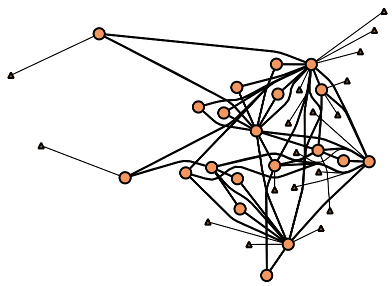
Table 1 provides the number of nodes $|\mathcal{V}|$ and links $|\mathcal{E}|$ in our investigated networks as well as the number of (access) nodes $|\mathcal{V}_A|$ removed from the original topologies to make them two-connected. All networks in our study have homogeneous link capacities except for the Rediris network for which real link capacities are provided in [40]. Table 1 also indicates the maximum and average node degree which is the number of neighbors of a node. We represent bandwidths and administrative link costs of all links



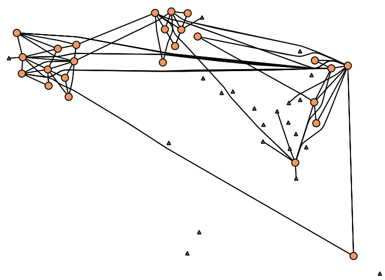
(a) AT&T Network:
25 core nodes, 112 uni-direct. links.



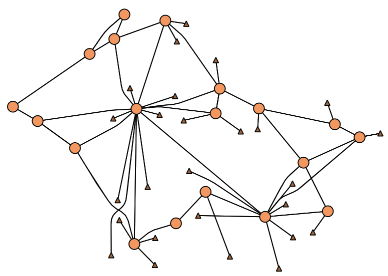
(b) BICS Network:
27 core nodes, 84 uni-direct. links.



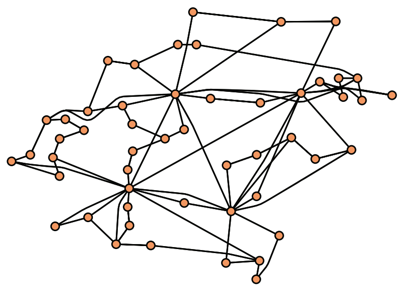
(c) China Telecom Network:
20 core nodes, 88 uni-direct. links.



(d) NTT Network:
25 core nodes, 112 uni-direct. links.



(e) CESNET Network:
19 core nodes, 60 uni-direct. links.

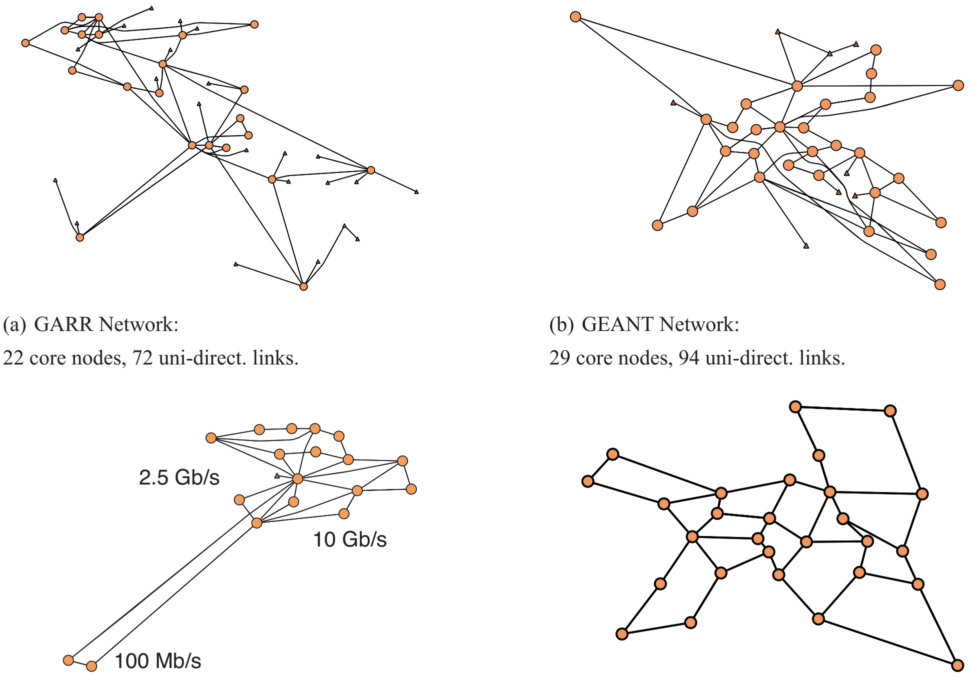


(f) DFN Network:
51 core nodes, 160 uni-direct. links.

Fig. 4. Network topologies under study—Part 1.

Table 1
Networks under study.

Network name and date		Size			Degree d		Geo
		$\|\mathcal{V}\|$	$\|\mathcal{E}\|$	$\|\mathcal{V}_A\|$	Avg.	Max.	location
Commercial network topologies from topology zoo [40]							
AT&T	2007–2008	25	112	0	4.48	10	US
BICS	2011/01	27	84	6	3.11	7	EU
China Telecom	2010/08	20	88	18	4.40	14	CH
NTT	2011/03	25	112	22	4.48	11	Global
Research and education network topologies from topology zoo [40]							
CESNET	2010/06	19	60	26	3.16	8	CZ
DFN	2011/01	51	160	0	3.14	12	DE
GARR	2010/12	22	72	22	3.27	8	IT
GEANT	2010/08	29	94	8	3.24	9	EU
RedIris	2011/03	18	60	1	3.33	10	ES
Topology from EU-Project NOBEL [41]							
Nobel-EU	2005/10	28	82	0	2.93	5	EU



(a) GARR Network:
22 core nodes, 72 uni-direct. links.

(b) GEANT Network:
29 core nodes, 94 uni-direct. links.

(c) RedIris Network:
18 core nodes, 60 uni-direct. links.

(d) Nobel-EU Network:
28 core nodes, 82 uni-direct. links

Fig. 5. Network topologies under study—Part 2.

by the vectors **c** and **k** so that the value for a specific link l is given by $\mathbf{c}(l)$ and $\mathbf{k}(l)$, respectively. The traffic aggregates (demands) between the pairs of different nodes constitute the traffic matrix $\mathcal{D}$. An aggregate $d \in \mathcal{D}$ has a source and destination node, and its rate is given by $r(d)$.

In our experiments, we assume that a failure affects links in both directions and we use single-shortest-path routing instead of equal-cost-multipath (ECMP). We consider the relative link load $\rho^{\max}$ in the failure-free case and after IP rerouting in all single bidirectional link failure cases as performance metric.[1] Therefore, we scale our artificially generated traffic matrices such that the relative link load $\rho^{\max}$ reaches 100% when uniform link costs $\mathbf{k_u}$ are used, i.e., all link costs are set to the same value.

4.2. Link cost optimization

Throughout our study, we use the “Threshold Accepting” heuristic presented in [21] to optimize link costs for a given objective function. While we have used the relative link load $\rho^{\max}$ as objective function in previous work, we study new objective functions in this paper to quantify the LFA coverage. To support these new objective functions, we extend the heuristic so that it first analyzes the availability of the different LFA types in every node of a network. Then, objective functions are calculated on this basis. We denote the new objective functions π_X^Y whereby Y indicates the specific variant and $X \in \{LP, ND\}$ indicates whether all LFAs (general LFAs, *LP*-LAC) or only those avoiding extra-loops (*ND*-LAC) are considered for protection. The objective functions are used for optimization of link costs and the corresponding optimized link costs are denoted by $\mathbf{k}_X^Y$.

In [22] we extended the general optimization algorithm of [21] so that it can optimize for a primary and secondary objective function. That means, if several link costs are found that are equally good with respect to the primary objective function, the ones are preferred which are better with respect to the secondary objective function. If not mentioned differently, we use the LFA coverage defined in the next sections as primary objective function and the maximum relative link load $\rho^{\max}$ as secondary objective function. In Section 5 we go into details of the optimization algorithm to extend it toward Pareto-optimization. The focus of this paper is not the optimization algorithm, but the different objective functions which could also be used with other optimization heuristics.

4.3. Use of LFAs to protect destinations

In all previous works, the fraction of protected destinations in a node, averaged over all nodes of a network (π^{dest}), has been used to quantify the LFA coverage. Moreover, only general LFAs have been taken into account (π_{LP}^{dest}) for which extra-loops can occur under some conditions. Therefore, link costs $\mathbf{k}_{LP}^{\text{dest}}$ optimized according to objective function π_{LP}^{dest} are denoted as conventionally optimized link costs. Both uniform link costs $\mathbf{k_u}$ and conventionally optimized link costs $\mathbf{k}_{LP}^{\text{dest}}$ constitute the baseline for our performance comparison.

[1] We will elaborate more on this metric in Section 5.1.

Table 2
π_{LP}^{dest}: percentage of destinations protected by general LFAs.

Network	$\mathbf{k}_\mathrm{u}$	$\mathbf{k}_{LP}^{\mathrm{dest}}$	$\mathbf{k}_{ND}^{\mathrm{dest}}$
AT&T	98.50	100.00	94.50
BICS	72.65	90.88	77.78
China Telecom	95.79	100.00	99.47
NTT	95.33	100.00	98.33
CESNET	87.43	98.25	83.33
DFN	72.08	93.10	76.86
GARR	74.89	98.27	87.01
GEANT	76.11	95.44	81.03
RedIris	88.24	98.69	85.62
Nobel-EU	61.24	89.55	77.25

Table 3
π_{ND}^{dest}: percentage of destinations protected only by LFAs that avoid extra-loops.

Network	$\mathbf{k}_\mathrm{u}$	$\mathbf{k}_{LP}^{\mathrm{dest}}$	$\mathbf{k}_{ND}^{\mathrm{dest}}$
AT&T	34.67	65.50	91.83
BICS	23.79	44.59	61.54
China Telecom	51.05	68.42	94.21
NTT	41.83	67.50	94.50
CESNET	13.74	36.84	66.67
DFN	27.06	42.90	57.49
GARR	36.80	49.57	71.43
GEANT	23.65	49.01	67.86
RedIris	29.08	46.41	75.82
Nobel-EU	29.23	43.25	51.85

4.3.1. *Percentage of protected destinations with* LP-*LAC*

Table 2 reveals that general LFAs protect between 61.2% and 98.5% of the destinations in the networks under study when uniform link costs $\mathbf{k}_\mathrm{u}$ are configured. Conventionally optimized link costs $\mathbf{k}_{LP}^{\mathrm{dest}}$ increase this range to values between 89.6% and 100%. These results confirm the findings from [14–16]: link cost optimization can tremendously increase the LFA coverage compared to uniform link costs in many networks and the achievable results depend on the network structure. In some networks (NTT, China Telecom, AT&T) even all destinations can be protected by general LFAs if conventionally optimized link costs $\mathbf{k}_{LP}^{\mathrm{dest}}$ are used.

The Nobel network deserves special attention as it yields the least LFA coverage. Its topology does not contain any triangles. As a consequence, all LFAs available with uniform link costs $\mathbf{k}_\mathrm{u}$ are node-protecting [14]. Therefore, the LFA coverage is 61.2%, no matter whether LFAs of the *LP*-LAC or only those of the *NP*-LAC are used for protection. The latter is not shown in the tables.

4.3.2. *Percentage of protected destinations with* ND-*LAC*

General LFAs may cause extra-loops in case of some node or multiple failures. Hence, they can worsen a failure situation instead of improving it. This is avoided if only LFAs of the *ND*-LAC are used for protection. We now investigate the fraction of destinations protected with LFAs of the *ND*-LAC (π_{ND}^{dest}); the results are compiled in Table 3. LFAs of the *ND*-LAC protect only between 13.7% and 51.1% of all destinations in networks with uniform link costs $\mathbf{k}_\mathrm{u}$ and between 36.8% and 68.4% in networks with conventionally optimized link costs $\mathbf{k}_{LP}^{\mathrm{dest}}$. This is due to the fact that the metric π_{ND}^{dest} reduces the set of eligible LFAs compared to π_{LP}^{dest} so that the LFA coverage is consistently lower or equal to the corresponding values in Table 2.

We now assume for the optimization that only LFAs of the *ND*-LAC Table 3 shows that they protect with optimized link costs $\mathbf{k}_{ND}^{\mathrm{dest}}$ between 51.9% and 94.5% of the destinations, which is a significant improvement compared to uniform link costs $\mathbf{k}_\mathrm{u}$ or conventionally optimized link costs $\mathbf{k}_{LP}^{\mathrm{dest}}$. Thus, even a large fraction of destinations can be protected by LFAs while avoiding extra-loops, but using the appropriate objective function for optimization is a prerequisite. Again, the achievable LFA coverage highly depends on the network structure.

For application in practice, it might be worthwhile to maximize the fraction of destinations protectable by LFAs that do not create extra-loops under any condition and extend that LFA coverage by general LFAs where LFAs of the *ND*-LAC are not available. Table 2 shows that between 77.3% and 99.5% of the destinations can be protected. The comparison of the values $\mathbf{k}_{ND}^{\mathrm{dest}}$ and $\mathbf{k}_{LP}^{\mathrm{dest}}$ makes a tradeoff evident: minimizing extra-loops reduces also the percentage of protected destinations; the extent of that reduction depends on the network structure.

4.4. *Use of LFAs to reduce traffic loss*

The major reason for using LFAs is the reduction of traffic loss from the detection of a failure until the completion of the IP rerouting process. To quantify the LFA coverage for this purpose, the fraction of protected destinations is not appropriate. We now consider the fraction of protected traffic to quantify the LFA coverage. However, this metric yields numbers close to 100% which are rather cumbersome to compare. Therefore, we take the fraction of unprotected traffic as metric instead, which can be interpreted as traffic loss in failure cases, and so we denote it as π^{loss}. We compute it as follows. For each link failure we calculate the fraction of traffic which is affected by that failure and not protected by an LFA, and average these values over all link failures. We take only single (bidirectional) link failures into account as we assume that their probability is two orders of magnitude larger than the one of node failures or multiple failures [42].

In contrast to the fraction of protected destinations π^{dest}, the unprotected traffic π^{loss} accounts for heterogeneous traffic matrices and for the amount of traffic forwarded by each node. Therefore, the calculation of the traffic loss π^{loss} requires the knowledge of the traffic matrix, which should be sufficiently stable to make the proposed metric meaningful. If the traffic matrix is not known for a network, we create a traffic matrix as described in Section 4.1.

4.4.1. *Percentage of lost traffic with* LP-*LAC*

Table 4 reports the traffic loss in failure cases when general LFAs are installed. The percentages vary between 0.02% and 3.75% for uniform link costs $\mathbf{k}_\mathrm{u}$. Conventionally optimized link costs $\mathbf{k}_{LP}^{\mathrm{dest}}$ reduce these values to a range between 0% and 1.31%. The improvement depends a lot on the network structure. The largest improvement is achieved in the GARR network where the unprotected traffic is reduced from 2.13% to 0.34%. When optimizing

Table 4
π_{LP}^{loss}: percentage of lost traffic when using general LFAs.

Network	$\mathbf{k}_u$	$\mathbf{k}_{LP}^{dest}$	$\mathbf{k}_{LP}^{loss}$	$\mathbf{k}_{ND}^{loss}$
AT&T	0.02	0.00	0.00	0.01
BICS	1.69	0.55	0.33	1.01
China Telecom	0.12	0.00	0.00	0.00
NTT	0.32	0.00	0.00	0.01
CESNET	1.76	0.47	0.08	0.52
DFN	1.18	0.83	0.38	0.70
GARR	2.13	0.34	0.09	0.44
GEANT	1.66	0.42	0.13	0.66
RedIris	0.46	0.18	0.10	0.15
Nobel-EU	3.75	1.31	0.62	1.30

Table 5
π_{ND}^{loss}: percentage of lost traffic when using only LFAs that avoid extra-loops.

Network	$\mathbf{k}_u$	$\mathbf{k}_{LP}^{dest}$	$\mathbf{k}_{LP}^{loss}$	$\mathbf{k}_{ND}^{loss}$
AT&T	2.31	1.47	0.49	0.14
BICS	4.62	4.56	4.06	2.11
China Telecom	2.48	1.25	0.88	0.07
NTT	2.20	1.19	1.15	0.11
CESNET	6.43	5.57	4.60	1.69
DFN	2.96	2.68	2.55	1.74
GARR	4.25	3.90	3.78	1.23
GEANT	4.46	4.22	4.72	1.61
RedIris	4.85	4.04	3.23	0.75
Nobel-EU	5.11	4.58	4.86	2.71

the link costs to minimize the fraction of unprotected traffic ($\mathbf{k}_{LP}^{loss}$), the percentages of unprotected traffic lie in the range between 0% and 0.62% and are clearly lower than those for conventionally optimized link costs $\mathbf{k}_{LP}^{dest}$.

Thus, the new objective function π_{LP}^{loss} leads to superior optimized link costs because LFAs are preferably available in nodes that forward lots of traffic and for destinations to which lots of traffic is forwarded. This is different for the other link costs $\mathbf{k}_u$ and $\mathbf{k}_{LP}^{dest}$ which are either not optimized or optimized without the information of the traffic matrix. These results underline that the specific objective function used for optimization purposes matters a lot.

4.4.2. Percentage of lost traffic with ND-LAC

We now allow only LFAs of the *ND*-LAC to avoid potential extra-loops. According to Table 5 the percentage of unprotected traffic is in a range between 2.20% and 6.43% for uniform link costs $\mathbf{k}_u$, in a range between 1.19% and 5.57% for conventionally optimized link costs $\mathbf{k}_{LP}^{dest}$, and in a range between 0.49% and 4.86% for $\mathbf{k}_{LP}^{loss}$. These values are all rather high. Appropriate optimization takes into account that only LFAs of the *ND*-LAC; correspondingly optimized link costs $\mathbf{k}_{ND}^{loss}$ can reduce the percentage of unprotected traffic to a range between 0.07% and 2.71%, which is a significant improvement. Thus, conventionally optimized link costs $\mathbf{k}_{LP}^{dest}$ are not universal enough to sufficiently well approximate the quality of appropriately optimized link costs $\mathbf{k}_{ND}^{loss}$.

When using link costs $\mathbf{k}_{ND}^{loss}$, LFAs of the *ND*-LAC may be primarily used and complemented by general LFAs to minimize both the risk of extra-loops and traffic loss. Table 4 shows that this variant leaves about the same amount of traffic unprotected as conventionally optimized link costs $\mathbf{k}_{LP}^{dest}$; however, the risk of extra-loops is clearly reduced because mostly LFAs of the *ND*-LAC are taken.

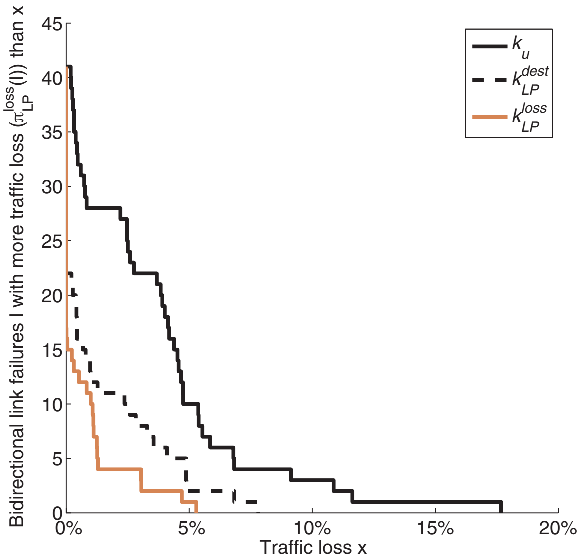


Fig. 6. Distribution of traffic loss for all single bidirectional link failures in the Nobel network when general LFAs are used.

4.4.3. Traffic loss distribution for general LFAs

The observed percentages of unprotected traffic are average values and seem small. Their real implication becomes clear in Fig. 6. It depicts the number of single link failures that cause more traffic loss than a certain percentage *x* due to missing LFAs; the evaluation is performed for the Nobel network with general LFAs.

With uniform link costs $\mathbf{k}_u$, 3.75% of the traffic is lost on average due to missing LFAs during single link failures. In 31 out of 41 link failure scenarios the traffic loss is lower than 5%, but 10 link failures cause more than 5% traffic loss. The failure of the link from Rome to Athens generates even 17.6% traffic loss which is quite a lot.

Conventionally optimized link costs $\mathbf{k}_{LP}^{dest}$ cause a smaller average traffic loss of 1.31%. In 19 out of 41 link failure scenarios even all traffic can be protected by LFAs. Only two link failures cause more than 5% traffic loss and the largest traffic loss is 8% for the failure of the link from Brussels to Frankfurt.

Link costs optimized to minimize the unprotected traffic $\mathbf{k}_{LP}^{loss}$ lead to even better results. In 25 out of 41 link failure scenarios all traffic can be protected by LFAs. Only the failure of the link from Athens to Belgrade exceeds the value of 5% and generates 5.5% traffic loss.

This evaluation exhibits that link costs optimized to reduce traffic loss are advantageous compared to uniform link costs $\mathbf{k}_u$ or conventionally optimized link costs $\mathbf{k}_{LP}^{dest}$. They increase the number of link failure scenarios in which all traffic can be protected and clearly decrease the number of link failure scenarios in which a large fraction of more than 5% of the overall traffic is lost. Thus, link cost optimization can have large effects for particular failure scenarios.

Table 6
$\pi_{LP}^{\mathbf{e2e}}$: percentage of e2e protected traffic using general LFAs.

Network	$\mathbf{k_u}$	$\mathbf{k}_{LP}^{\mathbf{dest}}$	$\mathbf{k}_{LP}^{\mathbf{e2e}}$	$\mathbf{k}_{ND}^{\mathbf{e2e}}$
AT&T	98.76	100.00	100.00	99.29
BICS	48.32	81.85	89.05	70.19
China Telecom	94.40	100.00	100.00	100.00
NTT	81.60	100.00	100.00	100.00
CESNET	46.21	85.29	97.49	83.53
DFN	30.83	44.88	73.82	56.03
GARR	30.24	87.27	96.78	80.27
GEANT	34.82	80.74	95.20	66.41
RedIris	86.41	94.53	97.07	92.33
Nobel-EU	0.00	47.64	79.22	52.14

Table 7
$\pi_{ND}^{\mathbf{e2e}}$: percentage of e2e protected traffic using only LFAs that avoid extra-loops.

Network	$\mathbf{k_u}$	$\mathbf{k}_{LP}^{\mathbf{dest}}$	$\mathbf{k}_{LP}^{\mathbf{e2e}}$	$\mathbf{k}_{ND}^{\mathbf{e2e}}$
AT&T	0.00	31.64	74.18	93.35
BICS	0.00	8.17	17.69	56.17
China Telecom	0.00	43.90	70.45	97.24
NTT	0.00	37.28	54.78	95.02
CESNET	0.00	7.17	22.31	69.67
DFN	0.00	4.07	7.71	28.82
GARR	0.00	3.62	17.07	75.18
GEANT	0.00	3.36	6.57	57.36
RedIris	0.00	12.63	29.39	83.86
Nobel-EU	0.00	0.74	7.87	37.39

4.5. Use of LFAs to increase the availability of entire paths

If all the links of a path from an ingress to an egress node are protected with LFAs, the overall availability of that path is tremendously improved: whatever link fails, the time to repair will be very short. On such paths, an ISP can provide a high-availability service to its customers. In the following, we evaluate the fraction of traffic whose entire paths can be protected by LFAs. We denote that metric as end-to-end (e2e) protected traffic $\pi^{\mathbf{e2e}}$ and link costs optimized according to that metric are denoted as $\mathbf{k}^{\mathbf{e2e}}$.

4.5.1. Percentage of end-to-end protected traffic with LP*-LAC*

Table 6 indicates the percentage of e2e protected traffic. With uniform link costs $\mathbf{k_u}$ between 30.2% and 98.8% of the traffic can be e2e protected by general LFAs. The Nobel network is an exception since not a single flow can be e2e protected. Due to the absence of triangles and the use of uniform link costs $\mathbf{k_u}$, all LFAs are node-protecting which cannot protect the last link of a path.

Conventionally optimized link costs $\mathbf{k}_{LP}^{\mathbf{dest}}$ increase the values $\pi_{LP}^{\mathbf{e2e}}$ to a range between 44.9% and 100%, and lead to 47.6% in the Nobel network. With link costs optimized for e2e protected traffic $\mathbf{k}_{LP}^{\mathbf{e2e}}$ between 73.8% and 100% of the traffic can be protected and even 79.2% in the Nobel network. Hence, the appropriately optimized link costs $\mathbf{k}_{LP}^{\mathbf{e2e}}$ significantly increase the percentage of e2e protected traffic $\pi_{LP}^{\mathbf{e2e}}$ compared to $\mathbf{k}_{LP}^{\mathbf{dest}}$.

4.5.2. Percentage of end-to-end protected traffic with ND*-LAC*

Table 7 extends this study toward the exclusive use of LFAs that avoid extra-loops in any failure scenario. With uniform link costs $\mathbf{k_u}$, not a single flow can be protected under these conditions in any network. We explain that phenomenon. The last hop toward a destination has distance 1 to this destination so that no other neighbor is closer to that destination. Therefore, it is impossible to find downstream LFAs to protect the failure of the last hop. As a result, not a single path can be e2e protected by LFAs of the *ND*-LAC with uniform link costs $\mathbf{k_u}$.

Conventionally optimized link costs $\mathbf{k}_{LP}^{\mathbf{dest}}$ e2e protect between 3.4% and 43.9% of the traffic and only 0.7% in the Nobel network. Link costs optimized for e2e protected traffic $\mathbf{k}_{LP}^{\mathbf{e2e}}$ with use of general LFAs e2e protect between 6.6% and 74.2% of the traffic, but appropriately optimized link costs $\mathbf{k}_{ND}^{\mathbf{e2e}}$ e2e protect between 28.8% and 97.2% of the traffic. Again, using the appropriate metric for routing optimization is crucial as approximations by similar metrics yield significantly worse results.

Table 6 demonstrates that complementing the coverage of LFAs of the *ND*-LAC with LFAs of the *LP*-LAC for $\mathbf{k}_{ND}^{\mathbf{e2e}}$ significantly increases the fractions of e2e protected traffic to a range between 52.1% and 100%. The percentage of e2e protected traffic is then similar to the one of conventionally optimized link costs $\mathbf{k}_{LP}^{\mathbf{dest}}$ but most potential extra-loops are avoided. However, $\mathbf{k}_{ND}^{\mathbf{e2e}}$ leads to clearly less e2e protected traffic than $\mathbf{k}_{LP}^{\mathbf{e2e}}$ when general LFAs.

4.6. Use of LFAs to preferably protect traffic with high-availability requirements

Only some networks allow to avoid traffic loss completely or to e2e protect all traffic with LFAs. Therefore, it seems reasonable to preferably protect traffic with high-availability requirements in those networks. This approach can be considered as a form of differentiated resilience [43–46]. For that purpose, we propose an extension of objective functions for link cost optimization and demonstrate its effectiveness in a challenging experiment.

4.6.1. Extension of objective functions for routing optimization with preferred protection of high-priority traffic

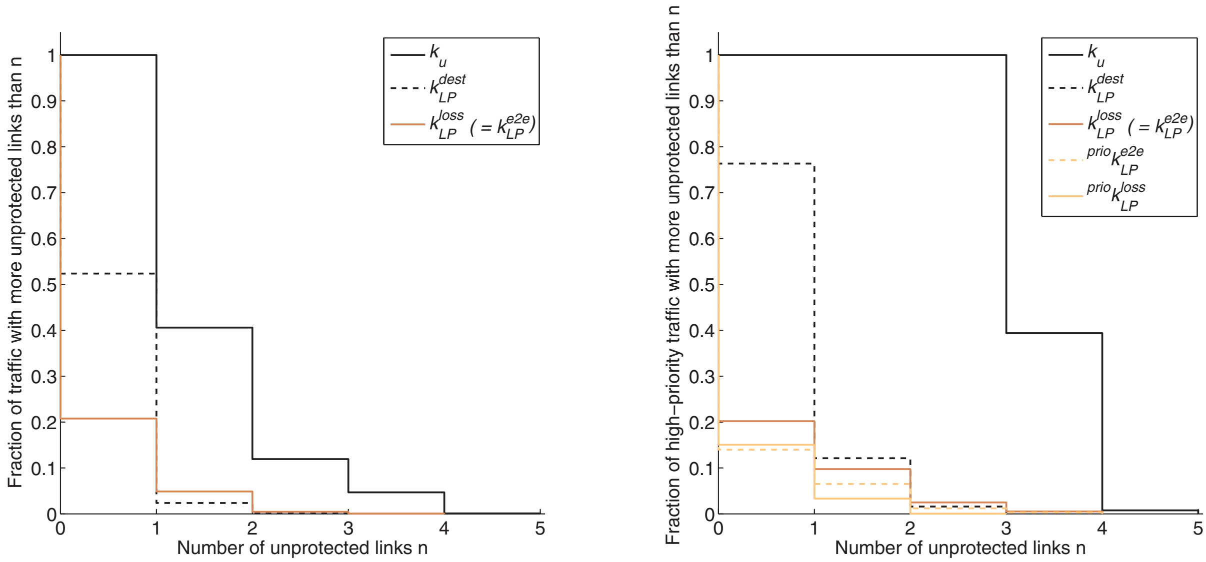
We assume that traffic of some ingress–egress pairs $d \in \mathcal{D}_h$ has high-availability requirements and that all other traffic has low-availability requirements. We call these traffic classes high- and low-priority traffic. Our goal is to preferably protect high-priority traffic. To prioritize high-priority traffic for the purpose of optimization, we modify the original traffic matrix by adding a priority offset $r_{\mathsf{offset}}^{\mathsf{prio}}$ to the rates of high-priority demands:

$$r_{\mathsf{modified}}(d) = \begin{cases} r(d) & d \in \mathcal{D} \setminus \mathcal{D}_h \\ r(d) + r_{\mathsf{offset}}^{\mathsf{prio}}(d) & d \in \mathcal{D}_h \end{cases}. \quad (4)$$

We use the overall traffic rate in the network $D_\Sigma = \sum_{d \in \mathcal{D}} r(d)$ to define the priority offset as

$$r_{\mathsf{offset}}^{\mathsf{prio}}(d) = D_\Sigma \cdot \frac{r(d)}{\min_{\{\delta \in \mathcal{D}_h\}} (r(\delta))}, d \in \mathcal{D}_h. \quad (5)$$

This definition makes the demand-specific priority offsets $r_{\mathsf{offset}}^{\mathsf{prio}}(d)$ proportional to the original traffic rates $r(d)$ and ensures that the priority offset for the high-priority demand with the smallest rate equals the overall traffic rate D_Σ. Thereby, $r_{\mathsf{modified}}(d)$ of the smallest high-priority traffic aggregate is larger than the sum of modified rates of



(a) With regard to the overall traffic.

(b) With regard to high-priority traffic.

Fig. 7. Distribution of the number of unprotected links.

all other low-priority traffic aggregates. As a consequence, the smallest high-priority traffic aggregate will be more respected in optimizations than any other low-priority traffic aggregate.

Variations for the modification of the traffic matrix, e.g., scalar multiplications or zeroing the rates of low-priority aggregates, are possible. Extensions based on modifications of the traffic matrix can be successfully applied only to traffic-aware objective functions such as π^{loss} proposed in Section 4.4 or π^{e2e} proposed in Section 4.5. It cannot be applied to the conventional objective function π^{dest} as this is not aware of any traffic demands.

4.6.2. Evaluation

In Fig. 7(a) we report the fraction of traffic in the Nobel network for which more than n links cannot be protected by general LFAs. These values depend on the link costs used in the network. With uniform link costs $\mathbf{k}_{\text{u}}$ any traffic is affected by the failure of at least one link. This is in accordance with the results presented in Table 6. About 12% of the traffic cannot be protected against the failure of three or four links on its path. Such traffic is quite vulnerable. Conventionally optimized link costs $\mathbf{k}_{LP}^{\text{dest}}$ clearly reduce the fraction of traffic affected by various numbers of link failures. Optimized link costs $\mathbf{k}_{LP}^{\text{e2e}}$ minimize the fraction of traffic affected by one or more link failures. In this particular experiment, these link costs also minimize the traffic loss so that $\mathbf{k}_{LP}^{\text{e2e}}$ equals $\mathbf{k}_{LP}^{\text{loss}}$. Note that these link costs lead to a larger fraction of traffic that is affected by at least two link failures compared to conventionally optimized link costs $\mathbf{k}_{LP}^{\text{dest}}$.

With uniform link costs $\mathbf{k}_{\text{u}}$, 12% of the traffic cannot be protected on three or four links. Since this traffic seems a challenge for being protected by LFAs, we define it as high-priority traffic in our experiment. We preferably protect that traffic with general LFAs using for optimization the metrics π_{LP}^{loss} and π_{LP}^{e2e} combined with the presented above. They yield the optimized link costs ${}^{\text{prio}}\mathbf{k}_{LP}^{\text{loss}}$ and ${}^{\text{prio}}\mathbf{k}_{LP}^{\text{e2e}}$.

Fig. 7(b) displays the percentage of high-priority traffic for which more than n links cannot be protected by LFAs. With uniform link costs $\mathbf{k}_{\text{u}}$, three or more links cannot be protected for 100% of that traffic which is in line with the design of the experiment. Conventionally optimized link costs $\mathbf{k}_{LP}^{\text{dest}}$ clearly reduce the number of links on which the high-priority traffic cannot be protected, but 77% of the high-priority traffic still misses LFA protection on one or more links. Link costs optimized to minimize traffic loss $\mathbf{k}_{LP}^{\text{loss}}$ (as well as $\mathbf{k}_{LP}^{\text{e2e}}$) reduce that percentage to 20%. Our proposed extensions decrease that value further down to about 15% and they also clearly reduce the fraction of high-priority traffic that cannot be protected on more than one or two links. This experiment substantiates that it is possible to improve the protection of a preferred subset of high-priority traffic.

4.7. Use of LFAs to fully protect link failures

As outlined in Section 1, advanced applications of IP-FRR delay the normal rerouting process. This can be done with losing hardly any traffic only if all traffic affected by a link failure is protected by LFAs. Therefore, the advanced applications can be performed only for links for which all carried traffic is protected by LFAs. We define a link as fully protected if all traffic affected by its bidirectional failure can be fully protected by LFAs. Furthermore, we define π^{link} as the fraction of fully protected links which should be maximized. The link costs optimized by this metric are denoted as $\mathbf{k}^{\text{link}}$.

Table 8
π_{LP}^{link}: percentage of links fully protected by general LFAs.

Network	$\mathbf{k_u}$	$\mathbf{k}_{LP}^{dest}$	$\mathbf{k}_{LP}^{link}$	$\mathbf{k}_{ND}^{link}$
AT&T	96.43	100.00	100.00	92.86
BICS	42.86	73.81	78.57	54.76
China Telecom	90.91	100.00	100.00	77.27
NTT	82.14	100.00	100.00	87.50
CESNET	66.67	83.33	86.67	56.67
DFN	35.00	53.75	58.75	40.00
GARR	41.67	77.78	86.11	61.11
GEANT	48.94	78.72	85.11	61.70
RedIris	66.67	86.67	86.67	66.67
Nobel-EU	0.00	46.34	70.73	41.46

Table 9
π_{ND}^{link}: percentage of links fully protected only by LFAs that avoid extra-loops.

Network	$\mathbf{k_u}$	$\mathbf{k}_{LP}^{dest}$	$\mathbf{k}_{LP}^{link}$	$\mathbf{k}_{ND}^{link}$
AT&T	0.00	58.93	21.43	73.21
BICS	0.00	19.05	9.52	47.62
China Telecom	0.00	52.27	29.55	68.18
NTT	0.00	58.93	21.43	71.43
CESNET	0.00	30.00	0.00	53.33
DFN	0.00	15.00	0.00	20.00
GARR	0.00	16.67	2.78	52.78
GEANT	0.00	23.40	2.13	51.06
RedIris	0.00	20.00	10.00	60.00
Nobel-EU	0.00	2.44	4.88	41.46

4.7.1. *Percentage of fully protected links with* LP-*LAC*

Table 8 gives the fraction of links fully protected with general LFAs. The percentages for uniform link costs $\mathbf{k_u}$ vary between 35.0% and 96.4%. The Nobel network is again an exception as not a single link can be fully protected with LFAs under uniform link costs $\mathbf{k_u}$. With uniform link costs $\mathbf{k_u}$, any link is a last link toward its destination, and cannot be protected since all available LFAs are node-protecting in the Nobel network due to the absence of triangles. Conventionally optimized link costs $\mathbf{k}_{LP}^{dest}$ clearly increase the fraction of fully protected links to a range between 46.3% and 100%. Using π_{LP}^{link} as objective function for optimization, the fractions of fully protected links can be even further increased in most cases; we observe the major effect of appropriate optimization in the Nobel network with 70.7% fully protected links compared to 46.3% for conventionally optimized link costs $\mathbf{k}_{LP}^{dest}$. Our results manifest that LFAs can be applied in some networks to fully protect all links. This allows for delayed IP rerouting and enables advanced applications. However, in most networks, only a subset of all links are fully protected so that IP rerouting can be delayed only for those links whose traffic is fully protected by LFAs. As a consequence, advanced applications may be performed only on a link-specific basis. That introduces complexity and lowers the benefit which rather questions the use of LFAs to enable advanced applications in such networks.

4.7.2. *Percentage of fully protected links with* ND-*LAC*

When delaying normal rerouting in IP networks, it may be crucial to avoid extra-loops caused by fast rerouting mechanisms as they persist until rerouting has completed. Therefore, avoidance of extra-loops seems important in this context. Table 9 compiles the percentages of links that can be fully protected by LFAs of the ND-LAC. With uniform link costs $\mathbf{k_u}$ not a single link can be fully protected. Due to the uniform link costs $\mathbf{k_u}$, any link is a last link on the path to its destination, and appropriate downstream LFAs cannot exist to protect such links. With conventionally optimized link costs $\mathbf{k}_{LP}^{dest}$ at least a small percentage of links—at most 59.0%, mostly clearly less—can be fully protected without causing extra-loops. Link costs maximizing the fraction of links fully protected by general LFAs $\mathbf{k}_{LP}^{link}$ fully protect an even lower fraction of links—at most 29.6%. This looks surprising, but they were not optimized for the use of ND-LAC LFAs. Link costs optimized to maximize the percentage of links fully protected by ND-LAC LFAs $\mathbf{k}_{ND}^{link}$ yield clearly better results than $\mathbf{k}_{LP}^{dest}$ and $\mathbf{k}_{LP}^{link}$ in the range between 41.5% and 73.2%. The DFN network is an exception with only 20% fully protected links. Although appropriate routing optimization can significantly increase the fraction of fully protected links, still a good subset of links cannot be fully protected. This observation holds for any investigated network.

5. Keeping link loads under control

In the previous section we have optimized link costs to maximize the LFA coverage using different metrics, e.g. π^{dest}, π^{loss}, π^{e2e}, or π^{link}. In addition to the reported results we have observed that optimized link costs sometimes lead to high relative link loads although relative link loads were minimized by the optimizer's secondary objective function. This problem has not been pointed out in literature before.

In the following, we first define a link load metric that is suitable in the context of resilient networks using LFAs. We propose an extension to our link cost optimization algorithm to find Pareto-optimal link costs. Performance results suggest that optimality in LFA coverage and in link load seem to be contradicting goals. Nevertheless, we demonstrate that it is possible to choose Pareto-optimal link costs leading to good LFA coverage and to moderate relative link loads.

5.1. *Definition of relative link load*

The load of a link l can be determined by the sum of the rates $r(d)$ of all demands d that are forwarded over link l. In particular, we consider in this work the load $\rho(l)$ of a link relative to its capacity $\mathbf{c}(l)$.

We observe three different link load stages in IP networks with regard to failure scenarios:

1. Link load under failure-free operation
2. Link load with traffic rerouted by LFAs before rerouting
3. Link load after IP routing has reconverged to failure state

Thereby we neglect stages during the rerouting process that may temporarily increase some link load values. The definition of the relative link load should cover all relevant stages that persist for sufficiently long time.

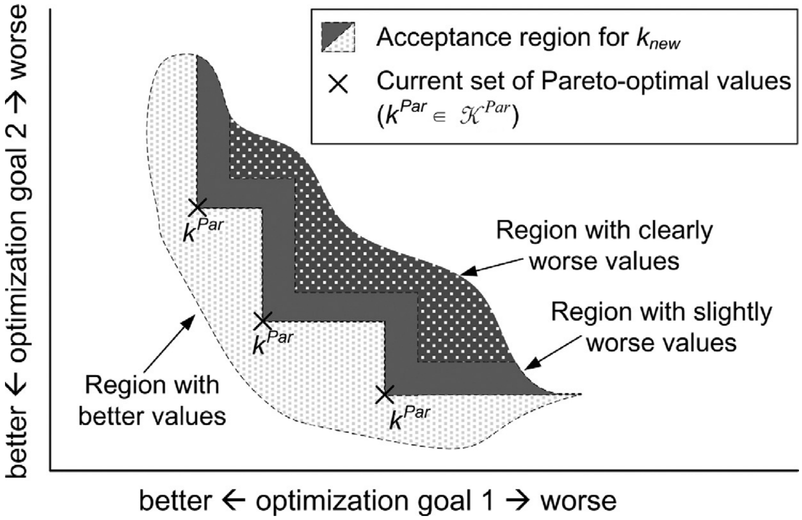


Fig. 8. Threshold Accepting algorithm with two objective functions: acceptance region for a new link cost vector k_{new}.

We assume now that LFAs are used to reduce the lost traffic until rerouting has completed. Here, link load stage (1) and (3) are persistent so that their maximum values should be respected for evaluations. In contrast, stage (2) is negligible as it lasts only in the order of a second. Furthermore, we consider node failures and multiple link failures clearly less likely than single (bidirectional) link failures. As a result, we define the maximum relative load $\rho^{\max}(l)$ of a link l as the maximum load experienced under failure-free conditions and after rerouting in single bidirectional link failure cases. The performance metric of interest is the maximum link load in the network:

$$\rho^{\max} = \max_{l \in \mathcal{E}} (\rho^{\max}(l)). \tag{6}$$

Note that this metric is independent of the kind of LFAs that are used for protection because the definition of $\rho^{\max}(l)$ does not include the fast reroute stage. As we do not assume the persistent use of LFAs, we can afford temporary extra-loops through LFAs for rare failure events. Therefore, we choose the use of general LFAs for our experiments.

Traffic loss happens if the load on a link is larger than 100% and reduces the traffic rate seen by a next hop. However, we do not work with original traffic matrices but with traffic matrices that are scaled such that the maximum link load $\rho^{\max}$ is 100% for uniform link costs $\mathbf{k}_{\text{u}}$. As the scaling of our traffic matrices is artificial anyway, we do not take into account that traffic is lost on links with more than 100% load, which may happen for other than uniform link costs $\mathbf{k}_{\text{u}}$. This approach is justified as we are only interested in the ability of different link costs to equally distribute the load from a relative traffic matrix through the network.

5.2. Pareto-optimization of link costs

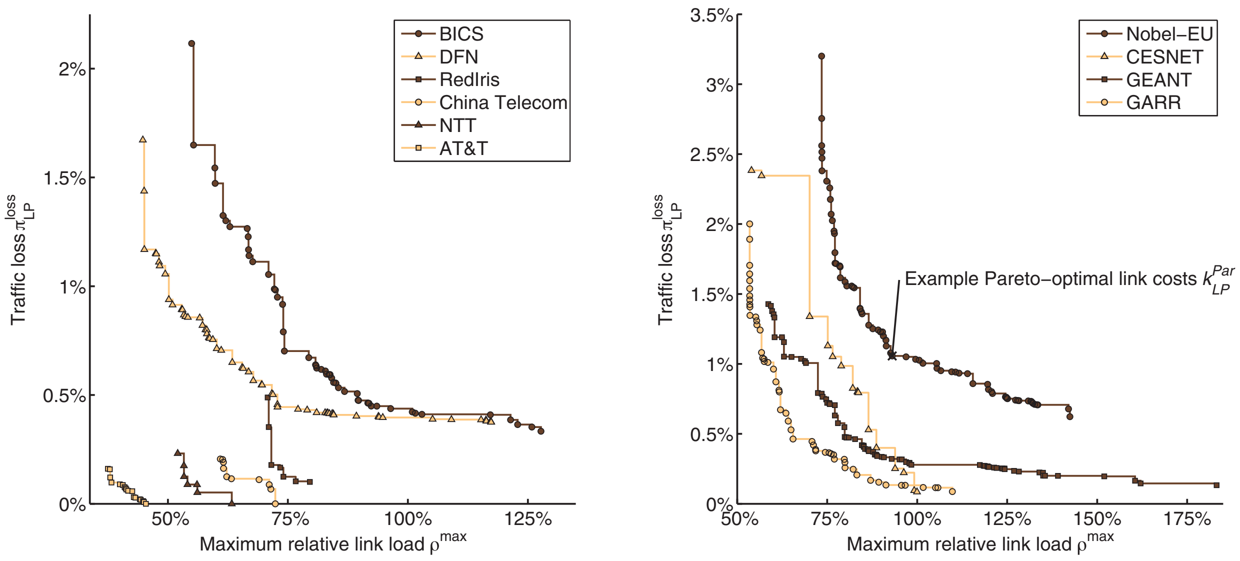
An element of a set is Pareto-optimal with regard to several metrics if no other element of this set is better with regard to all considered metrics. We are interested in finding a set of optimized link costs that are Pareto-optimal with regard to the fraction of lost traffic π_{LP}^{loss} due to missing LFAs and relative link load $\rho^{\max}$. To achieve this, we briefly review the principle of our optimization heuristic [21] and extend it for Pareto-optimization.

5.2.1. Link cost optimization using Threshold Accepting

Threshold Accepting randomly steps through the solution space of all link costs and searches for the link cost vector that the objective function f. The algorithm works with a current link cost vector $\mathbf{k}$ and records the best link cost vector $\mathbf{k}_{\text{best}}$ ever found. It explores the solution space by randomly choosing a new link cost vector $\mathbf{k}_{\text{new}}$ from a defined “neighborhood” of $\mathbf{k}$. To be able to escape from a local minimum, $\mathbf{k}_{\text{new}}$ is not only accepted as next current link cost vector if it is better than the current $\mathbf{k}$, but also if it is not worse than a threshold θ, i.e., if $f(\mathbf{k}_{\text{new}}) < f(\mathbf{k}) + \theta$. The exploration of the search space continues until no more improvements can be found for a specified number of iteration steps.

5.2.2. Extension of Threshold Accepting for Pareto-optimization

We modify the sketched Threshold Accepting algorithm to find Pareto-optimal results. We now have multiple objective functions f_i and a set of Pareto-optimal link costs $\mathcal{K}^{\text{Par}}$ instead of a single best result $\mathbf{k}_{\text{best}}$. We only need to define the acceptance regions for the extension of Threshold Accepting. A new link cost vector $\mathbf{k}_{\text{new}}$ is accepted if there is no other Pareto-optimal link cost vector $\mathbf{k}^{\text{Par}} \in \mathcal{K}^{\text{Par}}$ which is more than θ_i better than $\mathbf{k}_{\text{new}}$ in all objective functions f_i. This principle is depicted in Fig. 8 for two objective functions. There is a region with better link costs, a region with acceptable link costs that are not Pareto-optimal, and a region with unacceptable link costs. After a new Pareto-optimal link cost vector has been found, link cost vectors that are no longer Pareto-optimal need to be removed from the set of Pareto-optimal link costs $\mathcal{K}^{\text{Par}}$.



(a) Pareto-optimal link costs part 1

(b) Pareto-optimal link costs part 2

Fig. 9. Percentage of traffic loss $\pi_{LP}^{\mathbf{loss}}$ and maximum link load ρ^{max} for Pareto-optimal link costs.

5.3. Evaluation

We perform the above described Pareto-optimization for all test networks to minimize traffic loss π_{LP}^{loss} and the maximum relative link load ρ^{max}. Fig. 9(a) and (b) reveals the outcome. The results of the different networks are partitioned into the two figures in a way that optimizes readability. For this reason also the scaling of the x-axis differs in both figures.

Each point in the figures corresponds to a Pareto-optimal link cost vector and its position in the graph reveals the percentage of traffic without LFA protection π_{LP}^{loss} as well as the relative link load ρ^{max}. The Pareto-optimal link costs of a single network are linked by lines and identified by the same markers. Values for uniform link costs $\mathbf{k}_u$ are not presented in the figures for the sake of readability. They all lead to 100% maximum link load and their traffic loss is given in Table 4.

We first consider the AT&T, the NTT, and the China Telekom networks in Fig. 9(a). All Pareto-optimal link costs of these networks extend only over a relatively small region. However, the link costs creating the least traffic loss π_{LP}^{loss} lead to about 15% more relative link load ρ^{max} than the link costs minimizing that metric. This is already a significant difference so that care must be taken in choosing an appropriate link cost vector for configuration.

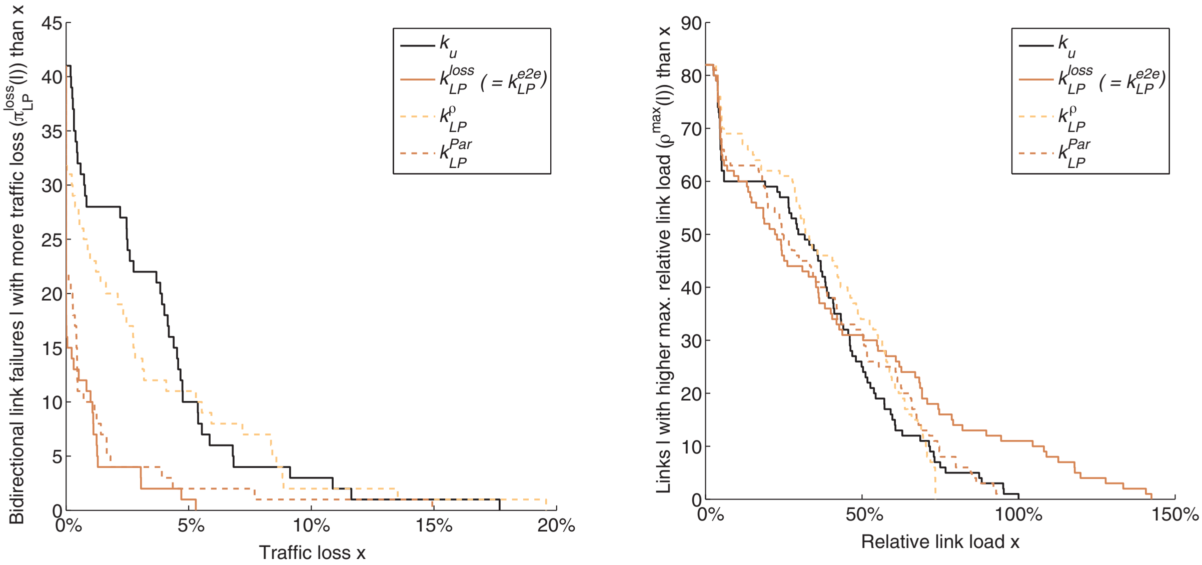
For all other networks in both Fig. 9(a) and (b), the traffic loss due to missing LFAs and the relative link load of the Pareto-optimal link cost vectors differ a lot. Optimized link costs with low traffic loss due to missing LFAs often lead to relative link loads above 100%, which is worse than with uniform link costs $\mathbf{k}_u$. In general, low values for relative link load apparently lead to large values of traffic loss and vice-versa. Thus, the two considered performance metrics seem to be contradicting optimization goals.

Nevertheless, the evaluations in Fig. 9(a) and (b) also show for all investigated networks that some of the Pareto-optimal link costs perform relatively well with regard to traffic loss and maximum relative link loads. A network administrator can choose one of these link costs for configuration by trading traffic loss π_{LP}^{loss} off for maximum relative link load ρ^{max}. As a consequence, the network will face only little traffic loss due to missing LFAs and face limited link loads even after rerouting in case of single link failures.

5.4. Quality of selected Pareto-optimal link costs

For further analysis, we select the Pareto-optimal link costs $\mathbf{k}_{LP}^{Par}$ for the Nobel network that are marked in Fig. 9(b). We compare them in detail with uniform link costs $\mathbf{k}_u$, link costs optimized to minimize the maximum link load $\mathbf{k}_{LP}^{\rho}$, and link costs optimized to minimize the percentage of unprotected traffic $\mathbf{k}_{LP}^{loss}$.

Fig. 10(a) displays the percentage of unprotected traffic in the Nobel network. It is similar to Fig. 6 but includes different optimized link costs. With uniform link costs $\mathbf{k}_u$, at least some traffic cannot be protected in any single bidirectional link failure scenario and in 10 failure scenarios more than 5% traffic will be lost. With link costs optimized to minimize the maximum link load $\mathbf{k}^{\rho}$, some traffic remains unprotected in 32 out of 41 bidirectional single link failure scenarios and more than 5% of the traffic cannot be protected in 11 link failure scenarios. In contrast, with the selected Pareto-optimal link costs $\mathbf{k}_{LP}^{Par}$, some traffic remains unprotected in 22 out of 41 bidirectional single link failure scenarios and more than 5% of the traffic cannot be protected in only two link failure scenarios. This is a significant improvement. Link costs optimized to minimize the percentage of unprotected traffic $\mathbf{k}_{LP}^{loss}$ lead to traffic loss in only 16 of 41 bidirectional link failure sce-



(a) Distribution of traffic loss for all single bidirectional link failures.

(b) Distribution of maximum relative load for all unidirectional links.

Fig. 10. Comparison of the Pareto-optimal link costs marked in Fig. 9(b) with other link costs in the Nobel network when general LFAs are used.

narios and the failure of only one link leads to more than 5% traffic loss. Thus, the optimized link costs $\mathbf{k}_{LP}^{\text{loss}}$ outperform all other presented link costs with regard to traffic loss, but the chosen Pareto-optimal link costs are not much worse.

Fig. 10(b) provides the number of links $l \in \mathcal{E}$ for which the relative link load $\rho^{\max}(l)$ exceeds a certain link load value x. The link costs optimized to minimize the percentage of unprotected traffic $\mathbf{k}_{LP}^{\text{loss}}$ lead to maximum link loads larger than 75% on 18 out of 82 unidirectional links. This seems unacceptable compared with the performance of the other link costs. Even uniform link costs $\mathbf{k}_{\text{u}}$ have only 6 links with maximum link loads larger than 75% and a significantly lower maximum load for most links. This is improved by the selected Pareto-optimal link costs $\mathbf{k}_{LP}^{\text{Par}}$ and of course by the link costs $\mathbf{k}^{\rho}$ minimizing the maximum relative link load. In particular, the maximum link load for $\mathbf{k}_{LP}^{\text{Par}}$ is lower than the one for uniform link costs $\mathbf{k}_{\text{u}}$. This deeper analysis qualifies the selected Pareto-optimal link costs as a good tradeoff between low link loads and.

6. Conclusion

Loop-free alternates (LFAs) constitute a simple fast reroute mechanism for IP networks (IP-FRR) and it is the only IP-FRR mechanism that is already standardized. However, LFAs usually cannot protect all traffic in a network even against single link failures and some LFAs may create extra-loops in case of node and multiple failures. LFAs may be applied to reduce lost traffic between the detection of a failure and the completion of IP rerouting, to improve the availability for some traffic aggregates, or to protect all traffic on a link to delay IP routing if that link fails. In this work, we looked at LFA coverage in 10 test networks from an application point of view. Therefore, metrics of interests are traffic loss due to missing LFAs, percentage of end-to-end protected traffic, and percentage of fully protected links. Moreover, we differentiated between general LFAs and those that avoid extra-loops under any condition. In contrast, previous work studied LFA coverage only as percentage of protected destinations and potential extra-loops were not considered.

We showed that administrative IP link costs can be set such that LFA coverage can be significantly increased. The achievable LFA coverage heavily depends on the network structure. In a few networks all traffic can be protected by LFAs after routing optimization, but only if extra-loops are acceptable in case of unlikely failures. When allowing only LFAs that avoid extra-loops, LFA coverage is reduced, and 100% LFA coverage cannot be achieved in any network. In such a case, the choice of the right objective function for routing optimization has a large influence on the resulting LFA coverage. As a result, “conventionally optimized link costs” that maximize the percentage of protected destinations often do not produce good results. It is also crucial to respect for the optimization whether all LFAs may be used for or only those that do not cause extra-loops. As some traffic aggregates may be more important than others with regard to fast protection, we developed a method that preferentially protects such traffic and demonstrated its viability by a challenging experiment.

We observed that optimizing link costs to improve only LFA coverage can lead to a huge imbalance of traffic in the network so that traffic may be lost due to overload. This is counterproductive as minimizing traffic loss is a major motivation for the use of LFAs. To solve that problem, we proposed Pareto-optimization yielding a set of link costs that are Pareto-optimal with regard to traffic loss due to missing LFAs and maximum relative link load. Some link costs among them perform well with regard to both metrics.

Acknowledgments

The authors acknowledge the funding by the Deutsche Forschungsgemeinschaft (DFG) under grants TR257/23-3 (Matthias Hartmann and David Hock) and ME2727/1-1 (Michael Menth). The authors alone are responsible for the content of the paper.

Appendix

Table 10
Summary of abbreviations and symbols.

Name	Description
$\pi^{\mathbf{dest}}$	Percentage of protected destinations
$\pi^{\mathbf{loss}}$	Percentage of lost traffic due to missing LFAs
$\pi^{\mathbf{e2e}}$	Percentage of end-to-end protected traffic
$\pi^{\mathbf{link}}$	Percentage of fully protected links
ρ^{max}	Maximum relative link load
$\mathbf{k_u}$	Uniform link costs
$\mathbf{k^{dest}}$	Link costs optimized for $\pi^{\mathbf{dest}}$
$\mathbf{k^{loss}}$	Link costs optimized for $\pi^{\mathbf{loss}}$
$\mathbf{k^{e2e}}$	Link costs optimized for $\pi^{\mathbf{e2e}}$
$\mathbf{k^{link}}$	Link costs optimized for $\pi^{\mathbf{link}}$
$\mathbf{k}^{\rho}$	Link costs optimized for ρ^{max}
$\mathbf{k^{Par}}$	Pareto-optimal link costs
LAC	Loop avoidance class
LP-LAC	LAC that uses all (general) LFAs
NP-LAC	LAC that uses only LFAs which avoid extra-loops in case of node failures
ND-LAC	LAC that uses only LFAs which avoid extra-loops in case of node failures and multiple failures

References

[1] D. Turner, K. Levchenko, A.C. Snoeren, S. Savage, California fault lines: understanding the causes and impact of network failures, in: ACM SIGCOMM, 2010. New Delhi, India.

[2] B. Fortz, M. Thorup, Robust optimization of OSPF/IS-IS weights, in: International Network Optimization Conference (INOC), 2003, pp. 225–230. Paris, France.

[3] A. Basu, J.G. Riecke, Stability issues in OSPF routing, in: ACM SIGCOMM, 2001, pp. 225–236. San Diego, CA, USA.

[4] M. Shand, S. Bryant, RFC5714: IP Fast Reroute Framework, 2010.

[5] A. Atlas, A. Zinin, RFC5286: Basic Specification for IP Fast Reroute: Loop-Free Alternates, 2008.

[6] P. Francois, O. Bonaventure, An evaluation of IP-based fast reroute techniques, in: ACM Conference on Emerging Networking EXperiments and Technologies (CoNEXT), 2005, pp. 244–245. Toulouse, France.

[7] A.F. Hansen, T. Cicic, S. Gjessing, Alternative schemes for proactive IP recovery, in: Second Conference on Next Generation Internet Design and Engineering (NGI), 2006. Valencia, Spain.

[8] M. Gjoka, V. Ram, X. Yang, Evaluation of IP fast reroute proposals, in: IEEE International Conference on Communication System Software and Middleware (COMSWARE), 2007. Bangalore, India.

[9] P. Francois, O. Bonaventure, Avoiding transient loops during the convergence of link-state routing protocols, IEEE/ACM Trans. Netw. 15 (6) (2007) 1280–1292.

[10] M. Shand, S. Bryant, S. Previdi, C. Filsfils, P. Francois, O. Bonaventure, RFC6976: Framework for Loop-Free Convergence Using the Ordered Forwarding Information Base (oFIB) Approach. http://www.rfc-editor.org/rfc/rfc6976.txt, 2013.

[11] P. Francois, M. Shand, O. Bonaventure, Disruption-free topology reconfiguration in OSPF networks, in: IEEE Infocom, 2007.

[12] F. Clad, P. Merindol, J.-J. Pansiot, P. Francois, O. Bonaventure, Graceful convergence in link-state IP networks: a lightweight algorithm ensuring minimal operational impact, IEEE/ACM Trans. Netw. 22 (1) (2014) 300–312.

[13] Juniper Networks, Understanding and Deploying Loop-free Alternate Feature. http://kb.juniper.net/library/CUSTOMERSERVICE/GLOBAL_JTAC/technotes/8010056-001-EN.pdf, 2009.

[14] G. Retvari, J. Tapolcai, G. Enyedi, A. Csaszar, IP fast ReRoute: loop free alternates revisited, in: IEEE Infocom, 2011. Shanghai, China.

[15] H. Trong Viet, P. Francois, Y. Deville, O. Bonaventure, Implementation of a traffic engineering technique that preserves IP fast reroute in COMET, in: Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications (ALGOTEL), 2009. Carry Le Rouet, France.

[16] G. Retvari, L. Csikor, J. Tapolcai, G. Enyedi, A. Csaszar, Optimizing IGP link costs for improving IP-level resilience, in: International Workshop on the Design of Reliable Communication Networks (DRCN), 2011. Krakow, Poland.

[17] R. Martin, M. Menth, M. Hartmann, T. Cicic, A. Kvalbein, Loop-free alternates and not-via addresses: a proper combination for IP fast reroute? Comput. Netw. 54 (8) (2010) 1300–1315.

[18] Cisco Systems, Cisco ASR 9000 Series Aggregation Services Router MPLS Configuration Guide, Release 4.0. http://www9.cisco.com/en/US/docs/routers/asr9000/software/asr9k_r4.0/mpls/configuration/guide/gc40asr9kbook.pdf, 2010.

[19] B. Fortz, M. Thorup, Internet traffic engineering by optimizing OSPF weights, in: IEEE Infocom, 2000, pp. 519–528. Tel-Aviv, Israel.

[20] A. Sridharan, R. Guerin, Making IGP routing robust to link failures, in: IFIP-TC6 Networking Conference (Networking), 2005. Ontario, Canada.

[21] M. Menth, M. Hartmann, R. Martin, Robust IP link costs for multilayer resilience, in: IFIP-TC6 Networking Conference (Networking), 2007. Atlanta, GA, USA.

[22] M. Hartmann, D. Hock, C. Schwartz, M. Menth, Objective functions for optimization for resilient and non-resilient IP routing, in: Seventh International Workshop on Design of Reliable Communication Networks (DRCN), 2009. Washington, D.C., USA.

[23] D. Hock, M. Hartmann, M. Menth, C. Schwartz, Optimizing unique shortest paths for resilient routing and fast reroute in IP-based networks, in: IEEE/IFIP Network Operations and Management Symposium (NOMS), 2010. Osaka, Japan.

[24] S. Rai, B. Mukherjee, O. Deshpande, IP resilience within an autonomous system: current approaches, challenges, and future directions, IEEE Commun. Mag. 43 (10) (2005) 142–149.

[25] A. Raj, O. Ibe, A survey of IP and multiprotocol label switching fast reroute schemes, Comput. Netw. 51 (8) (2007) 1882–1907.

[26] A. Kvalbein, A.F. Hansen, T. Cicic, S. Gjessing, O. Lysne, Multiple Routing Configurations for Fast IP Network Recovery, IEEE/ACM Transactions on Networking 17 (2) (2009) 473–486.

[27] S. Nelakuditi, S. Lee, Y. Yu, Z.-L. Zhang, C.-N. Chuah, Fast Local Rerouting for Handling Transient Link Failures, IEEE/ACM Transactions on Networking 15 (2) (2007) 359–372.

[28] S. Bryant, S. Previdi, M. Shand, RFC6981: A Framework for IP and MPLS Fast Reroute Using Not-Via Addresses. http://www.rfc-editor.org/rfc/rfc6981.txt, 2013.

[29] K. Lakshminarayanan, M. Caesar, M. Rangan, T. Anderson, S. Shenker, I. Stoica, Achieving convergence-free routing using failure-carrying packets, in: ACM SIGCOMM, 2007. Kyoto, Japan.

[30] P.-K. Tseng, W.-H. Chung, Joint coverage and link utilization for fast IP local protection, Comput. Netw. 56 (15) (2012) 3385–3400.

[31] K. Kompella, J. Drake, S. Amante, W. Henderickx, L. Yong, RFC6571: Loop-free Alternate (LFA) Applicability in Service Provider (SP) Networks, 2012.

[32] S. Litkowski, B. Decraene, C. Filsfils, K. Raza, M. Horneffer, P. Sarkar, Operational Management of Loop Free Alternates. http://tools.ietf.org/html/draft-ietf-rtgwg-lfa-manageability, 2015.

[33] IP-FRR LFA coverage evaluation via Cariden MATE, in: Cisco Live!, 2011. London, UK.

[34] L. Csikor, J. Tapolcai, G. Retvari, Optimizing IGP link costs for improving IP-level resilience with loop-free alternates, Comput. Commun. 36 (6) (2013) 645–655.

[35] S.S. Lor, M. Rio, Enhancing repair coverage of loop-free alternates, in: London Communications Symposium, 2010. London, UK.

[36] S. Cevher, T. Chen, I. Hokelek, J. Kang, V. Kaul, Y. Lin, M. Pang, M. Rodoper, S. Samtani, C. Shah, J. Bowcock, G. Rucker, J. Simbol, A. Staikos, An integrated soft handoff approach to IP fast reroute in wireless mobile networks, in: International Conference on COMmunication Systems and NETworks (COMSNETS), 2010. Bangalore, India.

[37] S. Bryant, C. Filsfils, S. Previdi, M. Shand, N. So, RFC7490: Remote Loop-Free Alternate (LFA) Fast Reroute (FRR). http://www.rfc-editor.org/rfc/rfc7490.txt, 2015.
[38] L. Csikor, G. Retvari, IP fast reroute with remote loop-free alternates: the unit link cost case, in: IEEE International Workshop on Reliable Networks Design and Modeling (RNDM), 2012.
[39] S. Knight, H.X. Nguyen, N. Falkner, R. Bowden, M. Roughan, The internet topology zoo, IEEE J. Selected Areas Commun. 29 (9) (2011) 1765–1775.
[40] S. Knight, H. Nguyen, N. Falkner, R. Bowden, M. Roughan, The Internet Topology Zoo, http://www.topology-zoo.org/, 2012.
[41] SNDlib 1.0—Survivable Network Design Data Library. http://sndlib.zib.de, 2005.
[42] J.-P. Vasseur, M. Pickavet, P. Demeester, Network Recovery, first ed., Morgan Kaufmann/Elsevier, 2004.
[43] P. Cholda, A. Mykkeltveit, B.E. Helvik, O.J. Wittner, A. Jajszczyk, A survey of resilience differentiation frameworks in communication networks, IEEE Commun. Surv. Tutorials 9 (4) (2007) 32–55.
[44] A. Autenrieth, A. Kirstädter, Engineering end-to-end IP resilience using resilience-differentiated QoS, IEEE Commun. Mag. 40 (1) (2002) 50–57.
[45] M. Brunner, G. Nunzi, T. Dietz, I. Kazuhiko, Customer-oriented GMPLS service management and resilience differentiation, IEEE Trans. Netw. Serv. Manage. 1 (2) (2004) 92–102.
[46] C.S. Ou, S. Rai, B. Mukherjee, Extension of segment protection for bandwidth efficiency and differentiated quality of protection in optical/MPLS networks, Opt. Switching Networking 1 (1) (2005) 19–33.

Matthias Hartmann is currently working at ENisco GmbH & Co. KG. Before he studied computer science and mathematics at the University of Würzburg/Germany, the University of Texas at Austin/USA, and at the Simula Research Laboratory/Oslo, Norway, and he was working as a researcher at the Chair of Communication Networks at the Institute of Computer Science in Würzburg/Germany from where he obtained his PhD in 2015.

David Hock is a senior consultant for research and development at Infosim GmbH & Co. KG and is coordinating the research activities in the area of software-defined networking. Before he studied computer science and mathematics at the University of Würzburg/Germany and at the BTH in Karlskrona/Sweden, and he was working as a research assistant at the Chair of Communication Networks at the Institute of Computer Science in Würzburg/Germany from where he obtained his PhD in 2014. His current main research interests are in the integration of software-defined networking and network management.

Michael Menth is a full professor at the Department of Computer Science at the University of Tuebingen/Germany and chairholder of Communication Networks. He received a diploma and a PhD degree in 1998 and 2004 from the University of Würzburg/Germany. Prior he studied Computer Science at the University of Texas at Austin and worked at the University of Ulm/Germany. His special interests are performance analysis and optimization of communication networks, resource and congestion management, resilience issues, network management, software-defined networking, industy 4.0, smart grids, and Internet of things. He holds numerous patents and received various scientific awards for innovative work.