Relaxed Multiple Routing Configurations for IP Fast Reroute

Tarik Čičić*§, Audun Fosselie Hansen*‡, Amund Kvalbein*, Matthias Hartman†, Rüdiger Martin† and Michael Menth†
* Simula Research Laboratory, Martin Linges vei 17, N-1331 Fornebu, Norway
Email: {tarikc, audunh, amundk}@simula.no
† University of Würzburg, Institute of Computer Science, Am Hubland, D-97074 Würzburg, Germany
Email: {hartmann,martin,menth}@informatik.uni-wuerzburg.de
‡ Department of Informatics, University of Oslo, Gaustadelléen 23, N-0371 Oslo, Norway
§ Telenor Research and Innovation, Snarøyveien 30, N-1331 Fornebu, Norway

***Abstract*—Multi-topology routing is an increasingly popular IP network management concept that allows transport of different traffic types over disjoint network paths. The concept is of particular interest for implementation of IP fast reroute (IP FRR). First, it can support guaranteed, instantaneous recovery from any link or node failure. Second, different failures result in routing over different network topologies, which augments the parameter space for load distribution optimizations. Multiple Routing Configurations (MRC) is the state-of-the-art IP FRR scheme based on multi-topology routing today.**

In this paper we present a new, enhanced IP FRR scheme which we call "relaxed MRC" (rMRC). rMRC simplifies the topology construction and increases the routing flexibility in each topology. According to our experimental evaluation, rMRC has several benefits compared to MRC. The number of backup topologies required to provide protection against the same set of failures is reduced, hence reducing state in routers. In addition, the backup paths are shorter, and the link utilization is significantly better.

I. Introduction

When there is a connectivity failure or a topological change in a network, traditional intra-domain routing protocols like OSPF or IS-IS respond by triggering a network-wide re-convergence. Information about the failure is broadcast in the network, and all routers in the domain independently calculate a new valid routing table upon receiving the notification. This is a time-consuming process that typically involves a period of instability and invalid routing in the network [1], [2]. The time-scale of this re-convergence process has recently been improved to sub-second intervals [3]. However, this is still not acceptable for emerging time-critical Internet applications with stringent demands on network availability.

A number of mechanisms for faster failure handling have been proposed for both MPLS [4] and connectionless IP networks [5]–[9]. These mechanisms prepare alternative routes in advance, which are immediately ready for use by the node that detects the failure. Such mechanisms have two very attractive properties. First, they respond quickly to a failure and prevent packet loss by allowing packet forwarding to continue on alternate routes while the routing protocol converges on the new topology. Second, they allow routers to delay the sending of a failure notification for a period of time while relying on the available repair path. This way, short-lived failures can be handled without triggering a global re-convergence. A large percentage of experienced network failures are short-lived [10], and handling such failures locally can improve network stability.

Multi-topology (MT) routing is a powerful traffic engineering and network management concept based on introducing multiple logical topologies in the network. Each logical topology is used to route a special class of the network traffic, identifiable from the packet header. For example, multicast or high-priority DiffServ traffic could be routed separately from the remaining traffic. The IP community has recently shown a strong interest in this concept, and the standardization process is expected to be completed soon [11], [12].

Multi-topology routing is well suited for implementation of fast local recovery in connectionless IP networks [5]. Multiple Routing Configurations (MRC, [9]) represents the state-of-the-art fast reroute scheme based on MT routing. In MRC, traffic headed to a failed network component is tagged and forwarded by the detecting node on an alternative logical topology that does not use the failed component for routing. These "backup" topologies are created using a set of rules that, in biconnected networks, guarantee that there is such a logical topology for each failure and each network destination. Thereby, MRC guarantees recovery from any single link or node failure.

Multi-topology routing allows independent setting of link weights in the logical topologies. For a fault tolerance scheme based on MT routing like MRC, this means all links can have distinct weights before and during the IP FRR phase. A careful tuning of these weights can improve the load distribution in the network [13]. In MRC, link-failure protection requires every link to be excluded from routing in one backup topology. Such links are said to be "isolated", and their weight is set to infinity. A typical logical topology has many isolated links, which constrains the routing of the affected traffic.

In this paper we propose an improved fast reroute scheme based on multi-topology routing. We call the scheme "relaxed MRC" (rMRC). rMRC does not require that all links are isolated, and hence it is simpler and arguably easier to deploy

and manage. In addition, fewer isolated links in a topology result in less constrained routing. We analyze key performance metrics of the new scheme and show a notable improvement compared to the state-of-the-art.

This paper is organized as follows. In Sec. II we provide additional background and related work in network load optimizations and IP FRR. In Sec. III we present our relaxed recovery scheme. We provide a qualitative comparison with MRC in Sec. IV. Our evaluation method is described in Sec. V and evaluation results in Sec. VI. We conclude the article in Sec. VII.

II. Background

IP fast reroute should provide full protection against all single link and node failures in the network. The IETF IP FRR framework [5] distinguishes between different recovery schemes for use in IP networks. The simplest scheme is fast failure protection using Loop-Free Alternates (LFA, [6]). In case of failure, LFA redirects traffic to neighboring nodes which have a path to the destination that does not include the failed component. The simplest case is when there are one or more equal cost alternative paths from the detecting node to the destination (Equal-Cost Multi-Path forwarding, ECMP). ECMP can be used both for load balancing and failure recovery.

Use of LFA alone does not guarantee 100% failure recovery [14]. Therefore, an additional scheme such as Not-Via [7], FIR [8], or MRC [9] needs to be deployed to complement LFA.

An important challenge for fast reroute schemes is to minimize the adverse consequences a recovery operation will have on the backup paths and traffic distribution [15]. Network operators often carefully configure their networks to avoid overloaded links. The shifting of traffic to alternate links after a failure can lead to congestion and packet loss in parts of the network [16]. This can be the case both while the fast-reroute is active and, in case of permanent failure, after the re-convergence process. Appropriate link weight settings can mitigate the packet loss in all phases.

The first traffic engineering mechanisms for connectionless IP networks were based on finding a set of link weights that distributes the load on the available links in the network given some estimate of the traffic demands [17], [18]. Later, more robust methods have been developed that also take into account variations in the traffic demands [19] or link failures [20], [21]. In MT-based recovery schemes, load can be distributed during the recovery phase as well [13].

III. Relaxed MRC

MRC as presented in [9] and [13] creates a set of backup topologies so that each link and node in the network are isolated in one of them. Relaxed MRC, "rMRC", relaxes the requirement that each link must be isolated in a backup topology.

The reason MRC requires isolated links is in its method to solve what is called "the last hop problem" [16]. Normally, both link and node failures are protected by routing traffic around the next hop node. However, when the link used to reach the destination fails, only the next hop link should be avoided and not the entire node. MRC requires each link to be isolated in one of the backup topologies. In the last hop case, MRC uses the backup topology where the link is isolated and not the destination node.

The relaxed version (rMRC) does not explicitly isolate all links to solve the last hop problem, but instead uses an adjusted forwarding procedure (presented in Sec. III-C).

A. Definitions

In the network topology graph $G = (V, E)$ let $w(u, v)$ be the weight of the unidirectional link (edge) $e = uv$. Let $w_{\max}$ be the maximal normal link weight in the network, i.e. $1 \leq w(e) \leq w_{\max},\ \forall e \in E$. Let $w_{\mathrm{r}} = |E| w_{\max}$ be the *restricted link weight*. The purpose of restricted links is to influence where shortest paths are laid in backup topologies—any acyclic path consisting of edges $e : w(e) \leq w_{\mathrm{r}}$ in the network will be shorter than a single restricted link.

The rMRC network topology T_i comprises the graph G and a weight function $w_i : E \rightarrow \{1, 2, \ldots, w_{\max}, w_{\mathrm{r}}\}$. rMRC distinguishes between the default topology T_0 and backup topologies $T_i, i > 0$. In T_0 no links are restricted, i.e., $w_0(e) \leq w_{\max},\ \forall e \in E$.

We define an *isolated node* $v \in V$ in topology T_i as a node whose adjacent links all have weight of at least w_{r}.

Isolated nodes must be placed in backup topologies so that the following invariant holds:

Invariant 1: All nodes must be connected by a path of non-isolated nodes only.

This ensures that all nodes can reach each other in all backup topologies without transiting an isolated node.

In rMRC, links between two isolated nodes will be given the weight of infinity. This is because these nodes must never transit any traffic. Fig. 1a and b gives an example on a typical backup topology for MRC (a) and rMRC (b) where nodes 3, 4 and 5 are isolated. The example illustrates that rMRC (b) requires less isolated links (fat links) than MRC (b). This leaves more available links for routing during failures. Sec. III-C will explain why rMRC requires less isolated links.

B. Basic Backup Topology Construction

Backup topologies may be constructed using different methods. We present a simple heuristic algorithm that attempts to isolate approximately equally many nodes in the given number of backup topologies.

The algorithm initially creates backup topologies as clones of the default topology (G, w_0), without any isolated nodes. In this algorithm, node queue Q_{n} is created as an arbitrary sequence (line 5).

The algorithm tries to isolate nodes as they are pulled out of the node queue in round-robin fashion (line 8). Function `connected`(T_i, u) tests if node u can be isolated in topology T_i without violating invariant 1 (Sec. III-A). If node 1

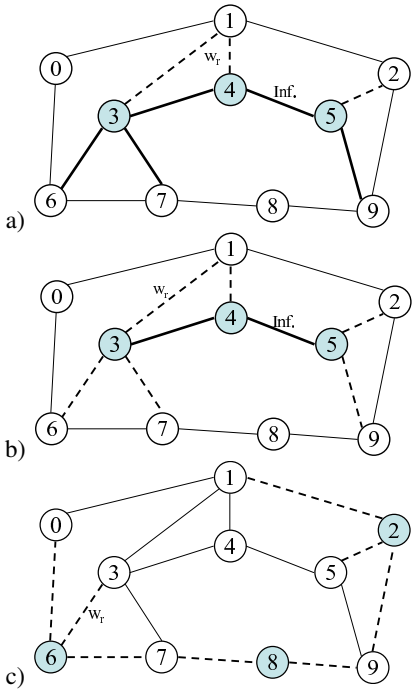

Fig. 1. Fault-tolerant multi-topology routing, sample backup configuration in MRC (a) and relaxed MRC (b, c). Nodes 3, 4 and 5 are isolated. Link 3-4, 3-6, 3-7, 4-5 and 5-9 are isolated and do not carry any traffic in MRC (a). In (b), only links 3-4 and 4-5 are isolated. The rMRC backup topology where node 6 is isolated is shown in (c).

Algorithm 1: Basic rMRC backup topology generator.

Input: Desired number of backup topologies n, graph G
Output: Backup topologies $T_1, \ldots, T_n$, if successful

1 **for** $i \in \{1 \ldots n\}$ **do**
2 $T_i \leftarrow (G, w_0)$ `// Backup topology` i
3 $S_i \leftarrow \emptyset$ `// Isolated nodes in` T_i
4 **end**
5 $Q_\mathrm{n} \leftarrow V(G)$ `// Node queue`
6 $i \leftarrow 1$
7 **while** $Q_\mathrm{n} \neq \emptyset$ **do**
8 $u \leftarrow$ `first` (Q_n)
9 $j \leftarrow i$
10 **repeat**
11 **if** `connected`(T_j, u) **then**
12 **forall** $(u, v) \in E(G)$ **do**
13 **if** $w_j(u, v) = w_\mathrm{r}$ **then**
14 $w_j(u, v) \leftarrow \infty$
15 **else**
16 $w_j(u, v) \leftarrow w_\mathrm{r}$
17 $S_j \leftarrow S_j \cup \{u\}$
18 **else**
19 $j \leftarrow (j \bmod n) + 1$
20 **until** $u \in S_j$ **or** $i = j$
  `// If` $i = j$`, all backup topologies tried`
21 **if** $u \notin S_i$ **then**
22 Abort execution
23 $i \leftarrow (i \bmod n) + 1$
24 **end**

was the next node to be tested in Fig. 1b, the test would return false. This is because node 4 would then not have a path of non-isolated nodes to all other nodes. If node 0 was the next node, the test would return true. In that case, the link weights are altered for the adjacent links of u (lines 12-16). If a neighbor of u was already isolated, the link between them will get weight ∞ (line 14). Else, the link will get the weight w_r (line 16). If `connected`(T_i, u) returns **false**, all other backup topologies are tried in sequence (line 19).

In some cases the desired number of backup topologies is too low for the input graph G, and the algorithm will have to abort and exit without success (line 22).

C. Forwarding

In multi-topology routing, all packets have to be identified with the topology they are routed in. The topology ID has to be encoded in the packet header. All nodes have to maintain routing information for all topologies to be able to forward data in any of them. This basic forwarding is done in steps 1 and 2 in the procedure in Fig. 2.

Failure-detecting nodes have a special role. They have to change the topology the packet is routed in from the default (normal) topology to the appropriate backup topology. Topology change can occur only once; if the packet is already tagged by a backup topology, step 3 drops the packet. If the failure is detected toward an intermediate node (not last hop) in the forwarding path, the appropriate backup topology is the one that has the failed node isolated. Then, regardless of whether there is a link or node failure that has been detected the packet will be rerouted around the failure to the destination.

If the failure is detected on the last hop in the forwarding path, the same last hop will always be returned in step 4, and step 5 will be evaluated to "Yes" for rMRC (left). For MRC (right), this step will be evaluated to "Yes" only when the link between the nodes is isolated in the same topology as the detecting node. We illustrate how the rMRC last hop handling works using Fig. 1. Assume node 6 detects a failure toward the last hop node 3. rMRC topology where node 3 is isolated is shown in Fig. 1b. Here, path 6-3 has still the lowest cost but must not be selected since link 6-3 may have failed. Instead, rMRC uses the topology where the detecting node 6 is isolated (Fig. 1c). In this backup topology, any neighbor of node 6 may be used to reach the destination. It is however favorable to pre-calculate which neighbor is closest to the destination and store this information in a data structure we call *last-hop recovery table* (used in step 6 in Fig. 2). In our example in Fig. 1c, node 7 is closest to the destination and the path will be 6-7-3. Since node 6 itself is isolated in this topology, packets will not loop back to the failed link 6-3.

IV. Qualitative Comparison

The core benefit of rMRC compared to MRC is that rMRC does not require all links to be isolated. MRC requires isolated links for solving the last hop problem. A node in rMRC does this by forcing packets to be sent to another neighbor than the normal next hop. Therefore, topologies with isolated nodes

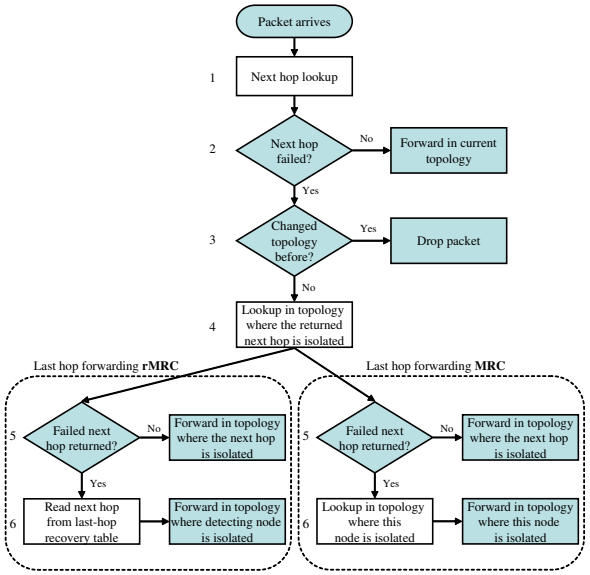

Fig. 2. rMRC and MRC forwarding. It illustrates the difference in how they solve last hop failures (steps 5-6).

only suffice. As a result, more links can be used for forwarding and the flexibility on how to route traffic is improved.

A. Forwarding Procedure

MRC solves the last hop problem by special assignment of isolated links and nodes and an additional lookup in the forwarding procedure (step 6 in Fig. 2). For example, in Fig. 1a, node 6 will use the topology where node 3 and link 6-3 are isolated. In this topology, the path to node 3 will be 6-0-1-3.

The third route lookup in MRC is substituted by a table lookup in rMRC, which is a negligible difference performance-wise.

B. Algorithm Complexity

The complexity of the presented rMRC algorithm is, similarly to MRC, determined by the loops and the complexity of the connectivity testing. An algorithm that tests whether a network is connected is bound to worst case $\mathcal{O}(|V| + |E|)$. The number of runs of the inner loop in Alg. 1 is bound by the maximum node degree Δ. In worst case, we must run through all n configurations to find a configuration where a node can be isolated. The worst case running time for the complete algorithm is then bound by $\mathcal{O}(n\Delta|V||E|)$.

While the worst-case running time of the relaxed algorithm is unchanged, the algorithm itself appears somewhat easier to understand and implement.

C. Effect of Isolated Links Elimination

Our key expectation is that the different ways to isolate a node will result in rMRC yielding shorter backup path lengths than MRC. When a node detecting a failure routes a packet according to the backup topology where the failed component is isolated, the packet follows that topology all the way to the egress node of the network. A basic property of both methods is that a topology isolates more than one node. However, no traffic is routed through an isolated node, only from or to an isolated node as ingress or egress of the network, respectively. So, in the case where the egress node for the traffic is isolated in the same topology as the component that has failed, traffic will be routed over the links that are attached to the isolated egress. In rMRC, such egress has more attached non-isolated links and thus paths to choose from, compared to MRC.

An example can be seen in Fig. 1. If node 3 is the egress node, backup traffic can only be routed over link 1-3 when using MRC (a). For rMRC, links 1-3, 6-3 and 7-3 can be used to reach node 3 (Fig. 1b). Having more options on how to reach this isolated egress will give a higher probability of finding a shorter backup path.

A related hypothesis is that the lower backup path lengths of the relaxed method lead also to an improved load distribution in the network. If for example a path of length n between a source and a destination is reduced to $n - 1$, there are one fewer router and link to carry the traffic between these nodes, decreasing the total network load and possibly increasing the total network utilization.

V. Evaluation Method

There are several performance metrics commonly used in evaluation of IP FRR schemes, including state requirements, backup path lengths, and load distribution.

Fault-tolerant multi-topology routing requires the routers to store additional information about the backup topologies. The amount of state required in the routers is related to the number of such backup topologies. An excessive amount of this state may affect router operation, in which case generating few topologies will be the goal. We evaluate how many backup topologies are necessary for MRC and rMRC to guarantee fault tolerance.

Backup path length will affect the total network load and the end-to-end delay. The reason to evaluate the backup path length in addition to the network load is also that the path length evaluation is independent of the traffic matrix, which makes the results more robust.

When the failure occurs, IP FRR will immediately start forwarding data traffic over backup paths. Since the backup paths already carry their normal traffic, there is real danger of congestion even in an otherwise well-provisioned network. We evaluate how well fault-tolerant multi-topology routing methods can optimize their load distribution and avoid congestion in the case of failure.

Evaluation of, e.g., state requirements of a fast reroute scheme requires experimenting with a large number of diverse network topologies, while load distribution optimizations are computationally expensive. We have therefore used two evaluation methods, one for the state requirements and backup path lengths, and one for the load distribution evaluation.

A. State Requirements and Backup Path Lengths

We have used synthetic network topologies based on the Waxman model [22], created using the Brite generator [23], as well as some publicly available real topologies. Families of 100 networks of size 16–64 nodes and two or three times as many links are tested. All link weights in the topologies are configured to the unit weight so that the path length calculations equal the hop count.

Algorithms for MRC (as in [9]) and rMRC (Alg. 1, Sec. III) are used to create backup topologies with the minimum number of topologies. For example, for a given topology Alg. 1 is run with $n = 2, 3, \ldots$, until the first successful execution. The results of these runs are presented in the state requirements analysis.

Based on the created topologies, we measure the backup path lengths (hop count) achieved by our schemes after a node failure. The backup path lengths are calculated for each source-destination pair in the network and for each node failure on the path between them.

B. Network Load Distribution

When multi-topology fault-tolerant routing is active in the network, the load distribution depends on three factors:

1) The link weight assignment used in the default (normal) topology,
2) The structure of the backup topologies, (i.e., which links and nodes are isolated in each of them),
3) The link weight assignments used in the normal links ($w(e) \leq w_{\max}$) of the backup topologies.

The link weights in the default topology (1) are important since all non-affected traffic is distributed according to them, while backup topologies are used only for the traffic affected by the failure. The backup topology structure (2) dictates which links are used in the recovery paths for each failure. The backup topology link weight assignments (3) determine which among the available backup paths are actually used.

The load distribution in the network (1) and (3) can be optimized using IP link weight optimization techniques. There are different approaches regarding the question whether IP link weights should be optimized primarily for the load distribution in the failure-free case or for the fast reroute phase (in which case some of the failure-free performance may need to be offered). This mainly depends on the network operators' management policies. Fault-tolerant multi-topology routing allows link weight settings in the backup topologies independent from the default topology. This allows us to optimize the failure free phase *and* improve the fast reroute load balancing.

We use ECMP forwarding to further improve the load distribution. Since this implies existence of this mechanism in the routers, we also use ECMP for fast reroute in cases an alternate equal-cost path is available after failure.

1) Considered Network Topologies: For the computationally demanding load distribution optimizations, we used several realistic network topologies, such as Cost239, Geant, LabNet, and Nobel. Among these, Geant represents an existing network while the remaining three are popular research topologies and show how the future networks should look like to properly support resilience mechanisms and fault management. This is reflected among other things in the network connectivity, Geant being relatively sparse compared to the others (Fig. 3).

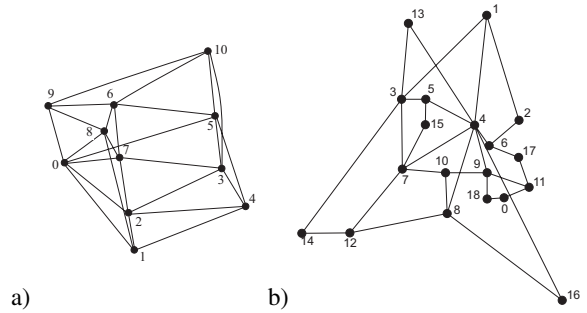

Fig. 3. Cost239 topology (a) and Geant topology (b) used in the evaluation.

2) Optimization Framework: Network operators often plan and configure their network based on an estimate of the traffic demands from each ingress node to each egress node. Clearly, the knowledge of such a demand matrix provides the opportunity to construct the backup topologies in a way that gives better load balancing and avoids congestion after a failure.

In this paper we optimize the load distribution for MRC and rMRC using the same three-step procedure:

1) The link weights in the normal topology are optimized for the given demand matrix while only taking the failure free situation into account.
2) For "intelligent" backup topology construction, the load distribution in the failure free case is used to weight the impact of each node failure on the link load in the network. The aim is to isolate nodes that carry a large amount of transit traffic in different backup topologies. Thus, if such a node fails, their traffic is deviated over different topologies with separate link weights, leading to a larger optimization potential. To that purpose, [13] describes a heuristic that sums up the total transit traffic through each node and isolates fewer heavy-traffic nodes, or more light-traffic nodes, per backup topology.
3) When the backup topologies are constructed, the normal link weights ($w(e) \leq w_{\max}$) of the backup topologies are optimized to get a good load distribution after any link or node failure.

For a clear and meaningful comparison, we take identical backup topologies for MRC and rMRC, except for the isolated links of MRC whose weights are relaxed to w_{r} in rMRC as described in Sec. III. For rMRC, only links between isolated nodes are still isolated.

3) Traffic Matrix: To evaluate the load distribution in the network, we require the knowledge of the traffic matrix. The structure of the matrix directly influences the optimal link weight setting. Thus, it is necessary to know the traffic de-

mands between all origin and destination pairs in the network. Even for real networks, this data is generally unavailable due to its confidentiality and difficulties in collecting it. We chose to synthesize the origin-destination (OD) flow data by drawing flow size values from a probability distribution and matching these values with the OD pairs according to the heuristic described in [24]. In short, we sort the OD pairs according to their node degree and the likelihood of one of them being used as the backup node in the case of a single link failure. Then, we match the sorted OD pair list with the sorted list of flow intensities generated using the gravity model, which is suited for this purpose [25]. The generated OD matrix is scaled so that the most loaded link in the failure-free case has utilization of 100 %.

4) Optimization Method: The traffic distribution in a network can be measured in terms of maximum link utilization and minimized by appropriate link weight settings. To optimize the link weights we use a self-developed software based on a simulated annealing-like principle [26]. In this paragraph, we formalize our optimization objectives.

Link weights determine which path are used for routing in IP networks. The traffic load described in the traffic matrix is routed along these paths. We represent the link weights for topology T_i by a vector w_i with one entry for each link (edge) $e \in E$. Given the link weight vector w_0 for the default topology T_0, we evaluate the link utilization $\rho(e, w_0)$ on all links $e \in E$ in the network during the failure-free case. This yields our objective function for optimization step (1) from above:

$$\text{minimize } \rho^{E}_{\max}(w_0) = \max_{e \in E} \left(\rho(e, w_0)\right) \qquad (1)$$

The algorithm implemented by our software heuristically searches the vector space of possible link weight vectors w_0 as described in [26].

Given the backup topologies $T_i (i = 1, \ldots, n)$ with their link weights w_i and the link weight vector w_0 for default topology T_0, we now can evaluate the link utilization $\rho^{w_0}(e, s, w)$ for link $e \in E$ in failure scenario $s \in S$, where $w = (w_1, \ldots, w_n)$ are the link weights vectors for the backup topologies. The set S hereby denotes the set of protected network element failure scenarios, e.g., all single link and node failures, and does not contain the failure-free case. Note that during failure scenario s the nodes adjacent to the failure send traffic over appropriate backup topologies according to MRC or rMRC. Thus, $\rho^{w_0}(e, s, w)$ is composed of the link utilization in the individual topologies T_i where the routing follows w_i. This yields our objective function for optimization step (3) from above:

$$\text{minimize } \rho^{w_0,E,S}_{\max}(w) = \max_{e \in E, s \in S} \left(\rho^{w_0}(e, s, w)\right) \qquad (2)$$

subject to the condition that special link weights may not be changed. The software again searches the space of possible link weight vectors for backup topologies T_i where w_0 for the default topology remains fixed.

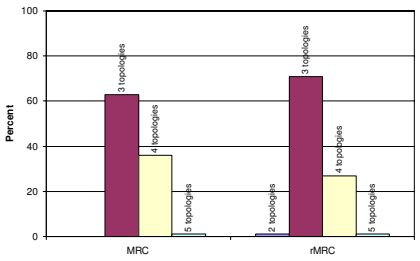

Fig. 4. Number of backup topologies for MRC and rMRC calculated for 100 random Waxman topologies with 32 nodes and 64 links

TABLE I
NUMBER OF BACKUP TOPOLOGIES FOR SOME REAL TOPOLOGIES

Network	Nodes	Links	MRC	rMRC
Abilene	11	14	5	4
German Tel.	10	29	3	3
DFN	13	64	2	2
Geant	19	30	5	4
Cost239	11	26	3	2

VI. EVALUATION RESULTS

A. Scalability—State Requirements

Relaxed backup topologies defined and described in Sec. III do not isolate all links. Therefore, there is more flexibility in rMRC than in MRC to decrease the number of backup topologies. Figure 1 can be used as an illustration for this difference. Assume that the process of isolating nodes (and links for MRC) should continue from the topologies presented for MRC (Fig. 1a) and rMRC (Fig. 1b). For MRC, nodes 1, 2 and 7 are not candidates to be isolated, because isolating any of them would disconnect one or more of nodes 4, 5 and 3 from the rest of the topology. For rMRC, it is only node 1 that is excluded from the list of candidates, since its isolation would lead to disconnection of node 4.

Figure 4 and Tab. I show the number of backup topologies generated with the MRC and rMRC algorithms as presented in Alg. 1. We observe that the increased flexibility with rMRC can decrease the number of topologies needed.

B. Path Lengths

Since routing in a backup topology is restricted, fault-tolerant multi-topology routing will potentially give backup paths that are longer than the optimal paths in the re-converged network.

Figure 5 shows the distribution of path lengths for normal failure-free routing, IP re-convergence, MRC and rMRC in networks with 32 nodes and 64 links (different network sizes show the same tendency). We see that the performance of less constrained rMRC is slightly better than the performance of MRC and close to the optimal full IP re-convergence. It is important to remember that IP FRR gives that performance immediately after the failure is detected, while the optimal scheme does not yield this until the re-convergence is completed.

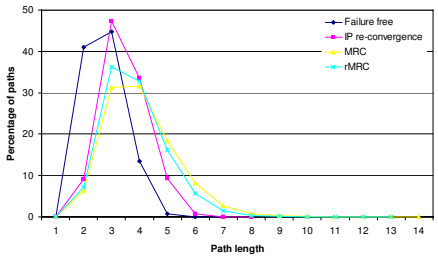

Fig. 5. Path length distribution.

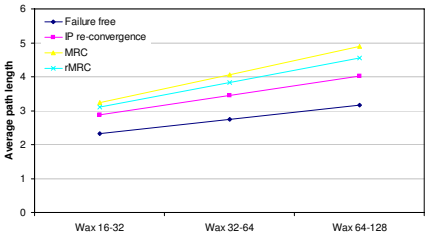

Fig. 6. Mean path length as function of the network size.

Mean path lengths for different network sizes are shown in Fig. 6. As the size of the networks increases the path lengths also increase. Still, rMRC shows a better performance compared to MRC. In Fig. 7, we show how the number of backup topologies influences the backup path lengths for MRC and rMRC in topologies with 32 nodes and 64 links. Increasing the number of backup topologies to a few more than the minimum achievable seems to improve the performance. However, the improvement seems to diminish if the number of backup topologies reaches a certain level.

C. Load Distribution

We present the load distribution for the tested networks in form of the complementary cumulative distribution function (CCDF). We show results for the failure-free load balancing, the reconverged network after a link failure (but without a new link-optimization process), then for rMRC and finally for the MRC fast reroute. If, for example, a CCDF line matches values $x = 0.5$ and $y = 0.68$, this means that 68% of the links have load utilization of 50% or more. The results are scaled so that the link with the highest load in the failure-free case has unit utilization 1.00. For all distributions except the failure-free case, the depicted values represent the maximal load, i.e., the load a particular link has experienced after the worst-case link failure. Note that in these simulations, we never drop traffic due to congestion. Instead, we let the utilization of some links exceed 100%. Hence, all load values should be considered relative.

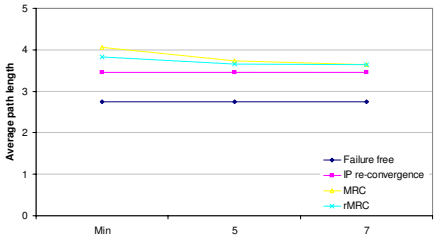

Fig. 7. Mean path length as function of the number of topologies. All networks have 32 nodes and 64 links. "Min" means the minimal number of backup topologies achieved by our algorithm for the given input topology; typically 3 or 4.

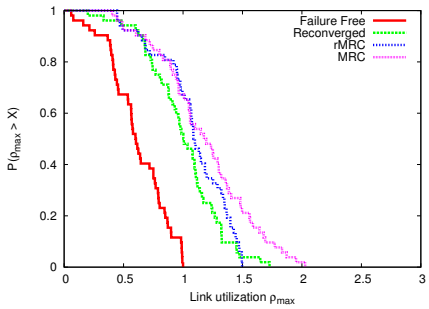

Fig. 8. Load distribution on Cost239 links.

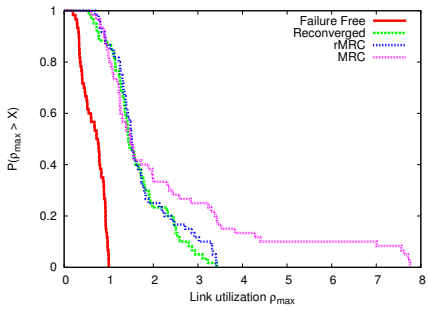

Fig. 9. Load distribution on Geant links.

For Cost239 (Fig. 8), with the link weights optimized for the failure-free case, the maximum link utilization for re-converged routing is 1.73. Optimized rMRC has the maximum link utilization of 1.50, and MRC of 2.03. Again, it is important to remember that IP FRR outperforms the re-converged routing immediately after the failure is detected—it does not need to wait for the routing process to converge.

The results indicate a significantly lower fast reroute load if rMRC is deployed rather than MRC. If we divide all links by traffic load into two equally large groups, the difference is particularly big (up to 35 %) for the high-load half, while MRC and rMRC behave similarly for the low-load half.

It is interesting that this significant difference is observed despite that in some 60 % of the cases nodes select an ECMP alternate for the affected traffic, in which case fault-tolerant multi-topology routing is not used at all.

The difference is dramatic for the Geant network, where the relative maximum link utilization for re-converged routing and the optimized rMRC is almost the same and lies around 3.42, while the optimized MRC performs poorly with a ratio of 7.76 (CCDF in Fig. 9).

Analysis of these results shows that the sparse connectivity of Geant effectively hinders the optimization process. For example, if the Geant point of presence in Austria should fail (represented by node 9 in Fig. 3b), a string of East-European countries (nodes 18, 0, 11, 17) is left without an important point of attachment. All traffic from these nodes passes link 6-17 that quickly becomes fully utilized and stops the optimization process. Furthermore, the congestion on 6-17 happens much sooner in MRC than rMRC. This is because rMRC only isolates link 9-10 among links adjacent to node 9, while in MRC, also 4-9 and 9-18 are isolated. When a neighbor that has node 9 as the last hop discovers the failure, it assumes a link failure and reroutes the traffic. In MRC all this traffic is routed toward the only non-isolated link 11-9, while in rMRC it can be rerouted also over links 4-9 and 18-9.

VII. Conclusion

In this paper we have proposed relaxed Multiple Routing Configurations (rMRC) for IP fast reroute. It is a simplification and enhancement of conventional MRC in the sense that the requirements for the backup topologies are relaxed. We explained the basic operation, the backup topology creation, and the link weight optimization that are applicable to MRC and rMRC. Using these algorithms, we compared the performance of the new rMRC to the one of MRC, normal IP re-convergence, and normal IP routing.

The results showed that the relaxed requirements of the rMRC have several benefits. The presented algorithm can guarantee link and node fault tolerance with fewer backup topologies than MRC. Furthermore, rMRC increases the connectivity of the backup topologies, so that the length of the backup paths is shortened and the link utilization in failure cases is lower due to improved load distribution. Therefore, we believe that rMRC is the best multi-topology routing based approach for IP fast reroute.

Our tests indicate that sparser networks may not always be able to improve the load distribution using link weight optimizations and current backup topology algorithms. As future work, we will explore more advanced mechanisms for backup topology creation, which we believe will together with link weight optimizations improve the load distribution whenever possible. Furthermore, the relaxed fault-tolerant multi-topology routing eases the formal reasoning and could result in better understanding and algorithms for, e.g., multi-fault tolerance.

References

[1] D. Watson, F. Jahanian, and C. Labovitz, "Experiences with monitoring OSPF on a regional service provider network," in *ICDCS '03: Proceedings of the 23rd International Conference on Distributed Computing Systems*. IEEE Computer Society, 2003, pp. 204–213.

[2] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian, "Delayed Internet Routing Convergence," *IEEE/ACM Transactions on Networking*, vol. 9, no. 3, pp. 293–306, June 2001.

[3] P. Francois, C. Filsfils, J. Evans, and O. Bonaventure, "Achieving sub-second IGP convergence in large IP networks," *ACM SIGCOMM Computer Communication Review*, vol. 35, no. 2, pp. 35 – 44, July 2005.

[4] V. Sharma and F. Hellstrand, "Framework for multi-protocol label switching (MPLS)-based recovery," in *IETF*, RFC 3469, Feb. 2003.

[5] M. Shand and S. Bryant, "IP Fast Reroute Framework," IETF Internet Draft (work in progress), June 2007, draft-ietf-rtgwg-ipfrr-framework-07.txt.

[6] A. Atlas and A. Zinin, "Basic specification for IP fast reroute: Loop-free alternates," Internet Draft, Mar. 2007, draft-ietf-rtgwg-ipfrr-spec-base-06.txt.

[7] S. Bryant, M. Shand, and S. Previdi, "IP fast reroute using not-via addresses," Internet Draft (work in progress), July 2007, draft-bryant-shand-IPFRR-notvia-addresses-01.txt.

[8] S. Nelakuditi, S. Lee, Y. Yu, Z.-L. Zhang, , and C.-N. Chuah, "Fast local rerouting for handling transient link failures," *IEEE/ACM Transactions on Networking*, vol. 15, no. 2, pp. 359–372, Apr. 2007.

[9] A. Kvalbein, A. F. Hansen, T. Čičić, S. Gjessing, and O. Lysne, "Fast IP network recovery using multiple routing configurations," in *Proceedings of IEEE INFOCOM*, Apr. 2006.

[10] A. Markopoulou, G. Iannaccone, S. Bhattacharyya, C.-N. Chuah, and C. Diot, "Characterization of failures in an IP backbone network," in *Proceedings INFOCOM*, Mar. 2004.

[11] P. Psenak, S. Mirtorabi, A. Roy, L. Nguen, and P. Pillay-Esnault, "MT-OSPF: Multi topology (MT) routing in OSPF," IETF, RFC4915, June 2007.

[12] T. Przygienda, N. Shen, and N. Sheth, "M-ISIS: Multi topology (MT) routing in IS-IS," Internet Draft (work in progress), Oct. 2005, draft-ietf-isis-wg-multi-topology-11.txt.

[13] A. Kvalbein, T. Čičić, and S. Gjessing, "Post-failure routing performance with multiple routing configurations," in *Proceedings of IEEE INFOCOM*, Apr. 2007.

[14] M. Gjoka, V. Ram, and X. Yang, "Evaluation of IP fast reroute proposals," in *Proceedings COMSWARE*, Jan. 2007, pp. 1–8.

[15] S. Rai, B. Mukherjee, and O. Deshpande, "IP resilience within an autonomous system: Current approaches, challenges, and future directions," *IEEE Communications Magazine*, vol. 43, no. 10, pp. 142–149, Oct. 2005.

[16] S. Iyer, S. Bhattacharyya, N. Taft, and C. Diot, "An approach to alleviate link overload as observed on an IP backbone," in *Proceedings INFOCOM*, Mar. 2003.

[17] B. Fortz and M. Thorup, "Internet traffic engineering by optimizing OSPF weights." in *Proceedings INFOCOM*, 2000, pp. 519–528.

[18] A. Sridharan, R. Guirin, and C. Diot, "Achieving near-optimal traffic engineering solutions for current OSPF/IS-IS networks," *IEEE/ACM Transactions on Networking*, vol. 13, no. 2, pp. 234–247, April 2005.

[19] B. Fortz and M. Thorup, "Optimizing OSPF/IS-IS weights in a changing world," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 4, pp. 756 – 767, May 2002.

[20] ——, "Robust optimization of OSPF/IS-IS weights," in *INOC*, oct 2003, pp. 225–230.

[21] A. Sridharan and R. Guerin, "Making IGP routing robust to link failures," in *Networking*, Waterloo, Canada, 2005.

[22] B. M. Waxman, "Routing of multipoint connections," *IEEE Journal on Selected Areas in Communications*, vol. 6, no. 9, pp. 1617–1622, Dec. 1988.

[23] A. Medina, A. Lakhina, I. Matta, and J. Byers, "BRITE: An approach to universal topology generation," in *Proceedings of IEEE MASCOTS*, Aug. 2001, pp. 346–353.

[24] A. Nucci, A. Sridharan, and N. Taft, "The problem of synthetically generating IP traffic matrices: Initial recommendations," *SIGCOMM Comput. Commun. Rev.*, vol. 35, no. 3, pp. 19–32, July 2005.

[25] M. Roughan, "Simplifying the synthesis of internet traffic matrices," *SIGCOMM Comput. Commun. Rev.*, vol. 35, no. 5, pp. 93–96, Oct. 2005.

[26] M. Menth, M. Hartmann, and R. Martin, "Robust IP link costs for multilayer resilience," in *In Proceedings 6th IFIP-TC6 Networking Conference*, May 2007.