Adaptive Post-failure Load Balancing in Fast Reroute Enabled IP Networks

Ning Wang, Abubaker Fagear
Centre for Communication Systems Research
University of Surrey
Guildford, United Kingdom
n.wang@surrey.ac.uk, abufageary@gmail.com

George Pavlou
Department of Electronic and Electrical Engineering
University College London
London, United Kingdom
g.pavlou@ee.ucl.ac.uk

***Abstract*— Fast reroute (FRR) techniques have been designed and standardised in recent years for supporting sub-50-millisecond failure recovery in operational ISP networks. On the other hand, if the provisioning of FRR protection paths does not take into account traffic engineering (TE) requirements, customer traffic may still get disrupted due to post-failure traffic congestion. Such a situation could be more severe in operational networks with highly dynamic traffic patterns. In this paper we propose a distributed technique that enables adaptive control of FRR protection paths against dynamic traffic conditions, resulting in self-optimisation in addition to the self-healing capability. Our approach is based on the Loop-free Alternates (LFA) mechanism that allows non-deterministic provisioning of protection paths. The idea is for repairing routers to periodically re-compute LFA alternative next-hops using a lightweight algorithm for achieving and maintaining optimised post-failure traffic distribution in dynamic network environments. Our experiments based on a real operational network topology and traffic traces across 24 hours have shown that such an approach is able to significantly enhance relevant network performance compared to both TE-agnostic and static TE-aware FRR solutions.**

I. INTRODUCTION

Emerging real-time multimedia applications and services pose stringent reliability, and subsequently, efficient fault recovery requirements on the underlying network platforms. In order to tackle the slow routing re-convergence problem upon link/node failures, various fast reroute (FRR) techniques have been proposed and standardised in recent years [1-4]. The basic operation of FRR techniques in IP networks can be described as follows. In addition to the default shortest IGP (e.g. OSPF) paths towards a destination prefix, each router also computes and maintains an alternative protection path for locally diverting traffic upon the failure of the default one. The enforcement of such protection paths is specific to different IP FRR techniques, for instance, deflection towards an alternative neighbouring router (next-hop) in Loop-Free Alternates (LFA [1]), or towards the next-next-hop using a tunnel in NotVia [2]. In all FRR techniques, the provisioning of protection paths follows a proactive approach in the sense that they are pre-computed and pre-configured *a priori* according to anticipated failure patterns. As a result, a repairing router is able to immediately divert affected traffic onto a pre-established protection path upon the detection of a failure. Such a make-before-break strategy is generally able to restrict the overall loss-of-connectively duration to sub-50 milliseconds, so that real-time applications do not suffer from any human-perceivable service disruption.

Nevertheless, it should be noted that the current FRR techniques do not take into account post-failure traffic optimisation requirements when computing protection paths. Although today's core networks are usually over-provisioned under the normal state, traffic congestion is quite common after failures due to the reduced network capacity. Therefore, customer flows may still get affected upon failures even with FRR protection, not directly by the actual loss-of-connectivity, but indirectly due to the post-failure traffic congestion along the activated protection path. In order to address this problem, traffic engineering (TE) –aware FRR techniques have been proposed in the literature [5][6]. The main idea is that the provisioning of FRR protection paths should also consider the anticipated traffic distribution upon a network failure. For instance, if multiple alternative FRR protection paths exist, the one that is expected to result in the best post-failure traffic conditions will be enforced [6]. This of course requires a fairly accurate estimation of the traffic matrix (TM), in a similar manner to current offline traffic engineering approaches.

Unfortunately, achieving fairly accurate long-term traffic forecasting is extremely difficult given the highly dynamic traffic patterns in today's operational networks. An interesting possibility to examine in the TE-aware FRR context is, despite the frequent traffic changes, as long as the traffic volume carried by individual links follows *correlated* changing patterns (e.g., increasing or decreasing proportionally in a "synchronised" manner), then static provisioning of protection paths might be adequate. This is because the overall *relative* traffic distribution across individual network links does not change significantly, even though their actual utilisations are highly dynamic. In order to test this assumption, we analysed the 7-day-long dynamics of traffic volumes in the GEANT network [7] based on the published dataset. The result is that the traffic changing patterns across individual links are largely *uncorrelated*. Figure 1 shows the overall traffic demand dynamics on three links connecting to the same point-of-presence (PoP) node in the GEANT network. As we can clearly see, the patterns of traffic dynamics among these links are generally uncorrelated. This observation implies that pure

978-1-4244-9220-6/11/$26.00 ©2011 IEEE

static FRR protection paths might be rigid in dealing with uncorrelated traffic patterns, potentially leading to suboptimal traffic distribution upon a failure. If we assume that each of the three links in the figure can be used by their common PoP node to enable a distinct FRR protection path towards a specific destination, it can be easily inferred that a static selection of one of them may not be adequate given the changing traffic distribution among them. In fact, the selection of the best candidate protection path should not only take into account the traffic conditions associated with *directly attached* links, but also the conditions further downstream from the neighbours towards the protected destination (see section III for more details).

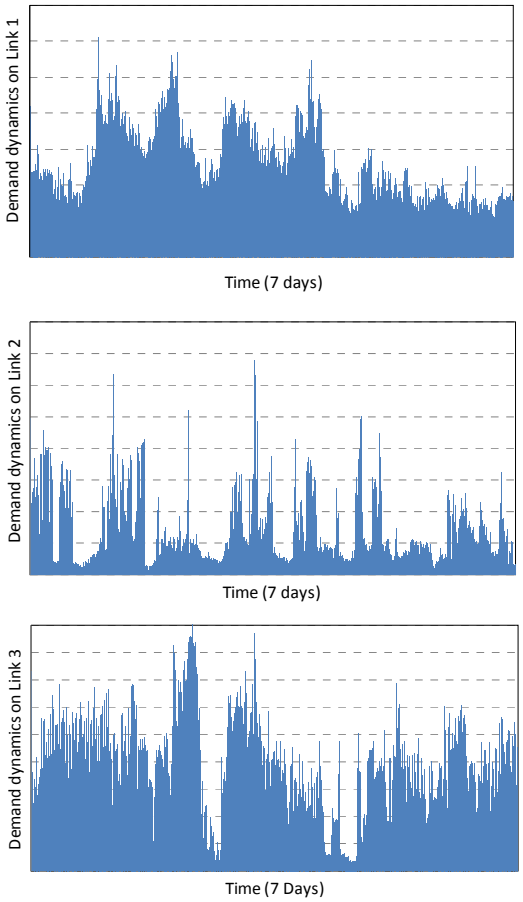


Figure 1. Traffic patterns of three outgoing links associated with one PoP in the GEANT Network

In this paper we investigate the feasibility of dynamically provisioning FRR protection paths in a self-managing manner in order to be adaptive to changing traffic patterns and achieve optimised post-failure network performance. Instead of configuring a single set of static FRR protection paths, we propose these paths to be periodically re-computed and re-configured according to the most recently captured traffic conditions. In general, there are two ways to realise this. In a *centralised approach*, a dedicated network management server periodically computes protection paths based on its up-to-date knowledge (e.g. through monitoring) about network conditions and subsequently re-configures them in all network routers. Alternatively, in this paper we introduce a *distributed approach* in which individual routers are responsible for computing by themselves protection paths (as a background process to the normal routing and forwarding operations), based on their own knowledge of up-to-date network conditions. The distributed approach is obviously more scalable and robust, achieving genuine self-healing and self-optimisation functions, but also much more technically challenging.

Regarding the specific FRR technique used, although in this paper we focus the adaptive re-configuration based on the Loop-free Alternates (LFA) mechanism [1], the proposed approach can be used with any other FRR mechanism that allows *non-deterministic* provisioning of protection paths. The classification of *deterministic* and *non-deterministic* FRR techniques will be presented in Section II. While the proposed adaptive FRR re-configuration can be regarded as a holistic paradigm for self-healing and self-optimisation functionality embedded in network routers, several key issues need to be carefully addressed:

- *Complexity of the algorithm for dynamically re-computing protection paths*: since protection paths are periodically computed in a distributed manner in individual routers, its time complexity (determining CPU processing overhead) should be sufficiently low in order not to disrupt normal foreground packet processing tasks by routers running the algorithm.
- *Router's knowledge about network conditions*: How will routers be able to collect necessary information about current network conditions in order to re-compute new protection paths given traffic pattern dynamicity?
- *Frequency of protection path re-configurations*: How often should the protection paths be re-computed and re-configured? It is not difficult to conceive the trade-off between the actual post-failure TE performance and the frequency of such re-configuration operations. Intuitively, the more frequently protection paths are re-configured, the better network performance can be achieved, especially in operational networks with high traffic dynamics. On the other hand, higher computing overhead will be also incurred for performing these tasks.

These key issues are specifically addressed when we present the proposed scheme in section III. According to our simulation experiments based on the operational GEANT network topology and traffic traces, significant performance improvement can be obtained in comparison to both standard TE-agnostic and static TE-aware FRR approaches. Detailed performance evaluations will be presented in Section IV.

II. FAST REROUTE TECHNIQUES

In this section we introduce the two most popular FRR techniques that are being standardised in the IETF, namely LFA [1] and NotVia [2], each representing *non-deterministic* and *deterministic* FRR mechanisms respectively.

According to LFA, when a *direct* neighbour of the repairing router has a native IGP path to the destination without traversing the protected network component, the repairing node can directly "deflect" the affected traffic to that neighbour for achieving FRR when the failure of the protected network component is detected. A *necessary condition* for a neighbour to become a feasible alternate next-hop candidate in

LFA is that this neighbour should not return the traffic back to the repairing router when the packets are being diverted towards the destination, i.e. the repairing node should not be on the shortest IGP paths from that LFA neighbour towards the protected destination. Let's take Figure 2 as an example. According to the IGP link weight setting, router *b* uses *c* as its default next-hop towards the destination *f* in the normal state. In case link *b*→*c* fails, the repairing router *b* may directly forward the customer traffic destined to *f* to one if its alternative neighbours such as *d* or *e*, without triggering the conventional IGP routing re-convergence procedure. According to the necessary condition indicated above, both *d* and *e* are feasible LFA candidates for such traffic diversion. Specifically, once customer traffic has been deflected onto *d*, *d* will use its own native IGP path *d*→*c*→*f* to send traffic towards the final destination which does not involve the failed link *b*→*c*. Similarly, router *e* is also able to successfully divert traffic towards *f* using native IGP path *e*→*f*. Based on this example, we can see that more than one feasible protection paths may exist, which offers the opportunity to strategically select one of them according to predicted traffic patterns. For instance if it can be anticipated that link *e*→*f* will become congested after the affected traffic is diverted onto it, then router *d* should be configured as the LFA neighbour in order to avoid using that link. We classify the FRR techniques in which *multiple* protection paths are possible for a specific network failure as *non-deterministic* ones. Another example of *non-deterministic* FRR approaches is the conventional IP tunnel-based technique specified in [3].

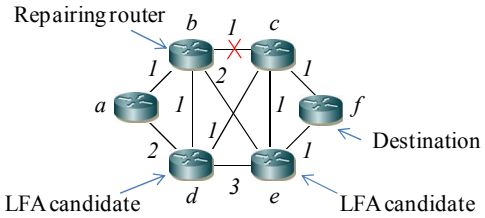


Figure 2. An example of LFA-based path protection

The NotVia [2] approach relies on special IP addresses assigned to each protected interface for enabling FRR. The semantics of a not-via address is that "a packet addressed to a not-via address must be delivered to the router advertising that address, not via the protected component with which that address is associated". When a failure occurs, the repairing router encapsulates the packet to a NotVia address of the protected interface. The actual diverting path towards the NotVia address is effectively the shortest path not including the failed network component. From the NotVia address, the routers along the repair path can know to which next-hop they must deliver the packet in order to avoid traversing the failed interface. Since the diverting path from the repairing router towards the NotVia address is *deterministic* (ignoring the ECMP effect), such an approach does not offer any opportunity for *selecting* an optimal protection path out of multiple choices.

III. ADAPTIVE TE AWARE LFA OPERATIONS

A. Overview

Let's start the illustration from the standard LFA operations. Given a repairing router *r* and its directly connected link *l* to be protected, the necessary condition for a neighbouring node of *r* (denoted by *t*) to become a feasible LFA towards destination *d* can be described as:

$$dist(t \to r) + dist(r \to d) > dist(t \to d) \qquad (1)$$

where *dist*(*i*→*j*) denotes the IGP distance from node *i* to node *j* (see Figure 3). In today's link state routing protocols such as OSPF, individual routers are able to compute by themselves feasible LFA candidates towards individual destinations, thanks to the topology information disseminated through link state advertisements (LSAs) across the network. As a result, the selected LFA candidate is installed locally by each repairing router as the backup next-hop towards each destination. Such configuration of LFA alternative next-hop towards each destination remains static during operation.

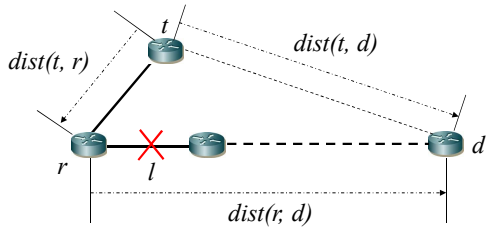


Figure 3. Necessary condition for LFA candidate selection

To enable individual routers to perform adaptive TE-aware LFA re-configuration against traffic dynamics, necessary information of the up-to-date traffic conditions needs to be disseminated across the network. As we mentioned previously, in order to achieve overall optimised post-failure network performance, the selection of LFA neighbours should make sure that the projected traffic distribution along the *end-to-end* (e2e) protection paths is optimised. This should not only include the traffic load condition associated with the local link connecting the LFA candidate, but also the remote load condition from that LFA candidate towards the final destination. Take Figure 3 as an example, when router *r* considers *t* to be the LFA candidate for protecting destination *d* against the failure of link *l*, the load on both the local link *r*→*t* and the (bottleneck) link load along the path from *t* towards *d* should be compared against their counterparts associated with *r*'s other LFA candidates towards *d* (not shown in the figure). While each router may have local monitoring on the load conditions associated with directly attached links, being able to know remote path/link conditions becomes an essential issue to be considered. Our proposed approach relies on the advanced IGP protocols that are able to periodically disseminate information of condition status per link across the network, with OSPF-TE [8] being an example (see Section III.C for more details).

Having obtained the up-to-date information about the traffic conditions on individual links, each potential repairing router runs a lightweight algorithm that computes the "remote"

shortest IGP paths from each of its feasible LFA neighbours to all protected destinations, along which the available/residual bandwidth of the bottleneck link is also obtained. The available bandwidth of the bottleneck link along this remote path is then considered jointly with the local bandwidth availability between the repairing router itself and its corresponding LFA neighbours in order to determine the best LFA candidate for each destination. Detailed specification on this algorithm and the structure of the supporting Traffic Engineering Information Base (TIB) will be presented in section III.B. In operational networks, such LFA computation process is performed periodically by individual routers upon receiving the newly disseminated OSPF-TE LSAs each time. Therefore, by configuring in OSPF-TE the time interval between adjacent LSA disseminations, the network administrator is effectively able to determine the frequency of the corresponding LFA re-computation and re-configurations accordingly.

B. Algorithm specification

As previously mentioned, in order to assist the computation of optimal LFA protection paths by individual routers, the value of link condition metrics needs to be periodically disseminated. We consider such type of information to be *available* (*or unused*) bandwidth as necessary input to the algorithm. Such information is in addition to the standard OSPF LSA properties such as interface connection and link weights.

Now we describe the network modelling for the problem of dynamic LFA selections. Let *G=(V, E)* represent a network topology with a set of routers *V* and a set of unidirectional links *E* with *e(i,j)* representing the link connected from neighbouring router *i* to *j*. Based on the configured IGP link weights, the shortest path from router *u* to *v* is denoted by $path(u \rightarrow v)$. During each interval for computing optimal LFA at individual routers, the following input is necessary:

***Static Input*:**

- Physical network topology *G=(V, E)*;
- Default next-hop towards each destination in the normal state;
- A set of feasible LFA next-hop candidates for each destination according to the necessary condition[1]; For each repairing router *r* and destination *d*, this set is denoted by $LNH^r(d)$;

***Dynamic (periodic) Input*:**

- Current available (unused) bandwidth on each link $(i, j) \in E$, denoted by *bw*(*i, j*), which is propagated periodically in OSPF-TE LSAs;
- Current traffic volume for repairing router *r* to forward to each destination *d*, denoted by $T^r(d)$, which can be periodically obtained by local measurements at individual repairing router *r* (see Section III.C). Effectively, $T^r(d)$ represents the anticipated destination-specific demand that needs to be rerouted by *r* in case the protected link fails.

It is worth mentioning that the set of feasible LFA candidates for each destination is regarded as static input solely based on the physical network topology, and hence it does not need to be periodically re-computed within each interval. The calculation of optimised TE-aware LFA at each repairing router *r* can be briefly described as follows. First, for each LFA-protected destination *d*, obtain the *local* available bandwidth from *r* itself to each feasible LFA next-hop for *d* (denoted by *l_bw*). Second, compute the (remote) available bandwidth of the bottleneck link along the IGP path from each feasible LFA next-hop towards the final destination *d* (denoted by *r_bw*) This can be done by a minor extension to the standard Dijkstra's shortest path tree algorithm in which the values of the minimum available bandwidth is recorded when computing hop by hop towards the destinations. The values of the local and remote available bandwidth is recorded by router *r* in its *traffic engineering information base* (*TIB*), and the *minimum* of the two values is actually the real bottleneck of the e2e protection path by using the corresponding LFA candidate. The selected LFA is finally configured in the actual backup forwarding table during the current interval according to the actual values of the e2e bandwidth availability. Specifically, the candidate associated with the highest e2e available bandwidth is finally selected. Such an operation is based on per-destination basis by each repairing router. It is important to note that the (projected) bandwidth availability on the selected e2e protection paths should be updated according to $T^r(d)$ after the determination of the LFA for each destination *d* by the repairing router *r*, as the associated traffic is expected to traverse the links along this protection path determined by the selection of that LFA upon the actual failure. This needs to be coordinated with the LFA selections for each protected link across all the affected destinations.

We take Figure 2 as an example again. Figure 4 shows an illustrative TIB maintained by router *b* for protecting destination *f* against the failure of link *b*→*c*. Out of the two LFA candidates *d* and *e*, *d* will be selected as the optimal LFA because the bottleneck associated with the e2e protection path (*b*→*d*→*c*→*f*) enabled by *d* has higher bandwidth availability than that by *e* (*b*→*e*→*f*), with the bottleneck of the latter being 35mbps at link *e*→*f* (*r_bw*). This is despite the fact that the local available bandwidth (*l_bw*) from the repairing router *b* itself towards *e* (62mbps) is higher than the other (46mbps). Since the failure of link *b*→*c* also affects the flows destined to *c* in addition to *f*, when the same repairing router *b* next considers the LFA candidate for destination *c*, it needs to take into account the fact that links *b*→*d* and *d*→*c* will have to carry diverted traffic towards *f*, as *d* has been previously selected as the LFA for *f*. In general, the pseudo code for selecting the optimal LFA candidate by a repairing router *r* for

[1] If a protected link has less than 2 feasible LFA candidates towards the destination, this link will not be considered in the algorithm. In this case either the link needs to be protected using the complementary NotVia FRR (as proposed in [4] for the scenario of non-existence of LFA), or the single feasible LFA becomes fixed during the operation.

each destination d (against the failure of the link connecting r and its default next-hop towards d) is presented in Figure 5.

$LNH^b(f)$	Available l_bw (local)	Available r_bw (remote)	e2e bw *(bottleneck)*
d*	46mbps	57mbps	46mbps
e	62mbps	35mbps	35mbps

Figure 4. Router b's TIB entries for LFA selection towards f

Step 1. Obtain available bandwidth on the local link towards each LFA candidate t for destination d:

$$\text{for each } t \in LNH^r(d): \; l\_bw(t) = bw(r,t)$$

Step 2. For each $t \in LNH^r(d)$, compute the available bandwidth on the bottleneck link along the path from t to d:

$$r\_bw(t) = \min(bw(i,j)), (i,j) \in path(t \to d)$$

Step 3. Determine the actual end-to-end bandwidth bottleneck for each $t \in LNH^r(d)$:

$$e\_bw(t) = \min(l\_bw(t), r\_bw(t))$$

Step 4. Select the LFA candidate t^* that is associated with the maximum end-to-end bandwidth bottleneck. That is:

$$t^* \leftarrow t \text{ with } \max(e\_bw(t)), \quad \forall t \in LNH^r(d)$$

Step 5. Update in the local *TIB* the (projected) available bandwidth on the links along the protection path associated with t^* (i.e. $r \to t^* \to d$, deduce by $T^r(d)$)

Figure. 5 LFA selection for destination d at repairing router r

C. *Major requirements and issues*

In this section we discuss in detail specific requirements and issues related to practical deployment in real operational network environments.

- *Computing Complexity*

We first analyse the time complexity of periodically computing TE-aware LFA backup paths at individual routers. From the algorithm specification presented in Section III.B, we can see that each repairing router is only responsible for computing optimised LFA candidates against the potential failure of its *directly attached* links. For each local link to be protected, every feasible LFA candidate (according to the necessary condition in (1)) needs to be examined towards the destination, including the bandwidth availability of both the local link (l_bw) and remote path from the LFA neighbour towards the final destination (r_bw). From the algorithm description in Figure 5, we can see that the computing time is mainly spent on *Step 2* for checking the bandwidth availability of the bottleneck link on the remote paths (r_bw). As we have mentioned, this procedure needs to record the current minimum bandwidth availability (bottleneck) when computing hop-by-hop the shortest path tree towards individual destinations based on the Dijkstra's algorithm, but this does not introduce any additional time complexity. Given that the time complexity of the optimised Dijkstra's algorithm is $O(|V|\log|V|)$, then the time complexity of computing the LFAs at each repairing router r is $O(D(r)|V|\log|V|)$, where $D(r)$ is the degree (i.e. total number of neighbours) of r. It should be noted that an efficient computing strategy for each repairing router r is to compute all at once the shortest path trees from each of r's neighbours towards all destinations, regardless whether they are feasible LFA candidates for individual destinations or not. The actual "filtering" operation is performed when determining the optimal LFA on per (local) link - destination bases whose complexity is $O(D(r)|V|)$, and this is not the major factor of the overall complexity. As we can see, the computation of optimised LFAs is very lightweight, and such an algorithm can be certainly activated at a timescale of several minutes, for example 10 or more, during operation time.

- *Gathering necessary traffic information*

As mentioned previously, in order to periodically disseminate dynamic bandwidth conditions across the network, TE-aware IGP routing protocols such as OSPF-TE is necessary. According to [8], a set of link-based sub-TLV (Type/Length/Value) metrics in OSPF-TE link state advertisements is defined for propagating bandwidth-related information across individual routers. Although such sub-TLVs were originally defined for enabling the establishment of TE-aware label switched paths (LSPs) in MPLS environments, they can also be used for other purposes in pure IP routing, for instance to use the *unreserved bandwidth* metric to denote available bandwidth [10]. As indicated in [8], each OSPF-TE speaker may actively monitor the "traffic engineering" network topology with bandwidth awareness, and adaptively react to the changing network condition by re-computing optimal routes. Effectively, this may not only refer to the computation of default traffic delivery paths in the normal state, but also the re-configuration of protection paths against potential network failures, as is the case of computing optimal LFAs for IP fast reroute purposes addressed in this paper.

According to section III.B, each repairing router r also needs to obtain the up-to-date information of the traffic volume towards protected destinations, i.e. $T^r(d)$. Such information can be gathered through network measurement tools such as NetFlow integrated in individual routers, based on which the information on the overall traffic volume on per-destination basis can be periodically derived. Such information is locally "exported" and used for computing optimised LFA by individual repairing routers.

- *Re-configuration frequency*

Last but not least, how often the LFA-based protection paths should be re-computed and re-configured is a key issue that needs to be carefully considered during network operation. Intuitively, the frequency of protection paths re-configurations depends on how dynamic the network conditions are. Too frequent computation of protection paths in the operational networks with less dynamic traffic conditions may often lead to *unchanged* configuration results, in which case the CPU time of routers are wasted. On the other hand, less frequent computation of protection paths may have insensitive reactions to traffic dynamics, resulting in suboptimal post-failure performance. In order to determine the best trade-off

between complexity and performance, it is essential for network operators to accurately capture the traffic patterns in their networks. Our analysis on the traffic dynamics in the GEANT network indicates that re-configuration interval at tens of minutes may result in optimised network performance. Detailed experimental results are presented in Section IV.

We recommend that the periodical re-computation and re-configuration of LFA protection paths be triggered upon the receipt of new OSPF-TE LSAs which is also synchronised with the interval of internal traffic measurement exports for deriving $T^r(d)$. Therefore, through the network administrator's configuration of the time interval for broadcasting LSAs in OSPF-TE and local traffic measurements, the frequency of protection path re-configurations can be determined. A special case is that the actual network failure occurs *during* the re-computation process of LFA configurations for the next interval. In this case, the current LFA configuration needs to be immediately activated.

IV. PERFORMANCE ANALYSIS

A. Simulation setup

We use the GEANT network topology and the actual 24-hour traffic traces for evaluating the performance of the proposed scheme [9]. The GEANT topology consists of 23 point-of-presence (PoP) nodes and 74 unidirectional links with bandwidth capacity up to 10Gbps. The traffic traces data are compiled based on 15-minute interval monitoring, which gives altogether 96 distinct traffic traces across every 24 hours. The maximum link utilisation (MLU) dynamics in the normal situation during this period is depicted in Figure 6 (starting from 12:00 noon), and indicates that there is no traffic congestion under the failure-free condition throughout the period.

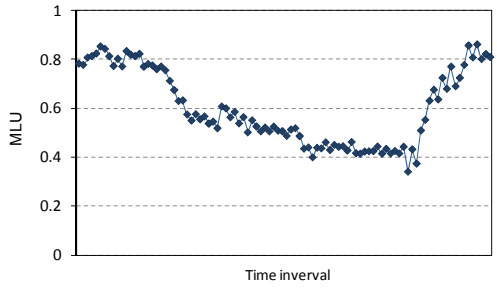


Figure 6. 24-hour *MLU* performance in the normal condition

B. Evaluation metrics

In order to evaluate the proposed TE-aware LFA re-configuration technique in a comprehensive fashion, we consider the following performance metrics, mainly focusing on (post-failure) traffic optimality and re-configuration frequency against the actual traffic dynamics. It is worth mentioning that, since LFA by itself is not able to provide 100% protection coverage for all link-destination pairs, we only consider the situation where there exists at least one feasible LFA candidate. Specific performance metrics include:

- *24-hour post-failure link utilisation variation (pLUV)*: this metric indicates the projected maximum, minimum and mean post-failure utilisation of each link across the 24-hour duration.
- *Post-failure maximum link utilisation (pMLU)*: This metric indicates the projected post-failure utilisation of the most utilised link across the entire network at each LFA re-configuration interval.
- *LFA switching dynamics*: This metric indicates the total number of times LFA candidates are switched on per <repairing router, protected destination> pair during the 24-hour period.

C. Performance evaluation

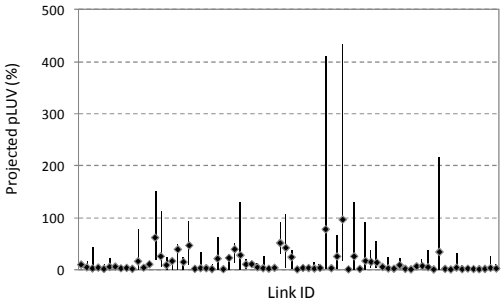
For the *pLUV* and *pMLU* performance metrics, we compare the following approaches:

(1) *TE-agnostic LFA (LFA-TA)*: a feasible LFA candidate is selected without taking into account post-failure traffic performance. We consider the worse case scenario in order to show how bad the performance could be if the least desired candidate is selected.

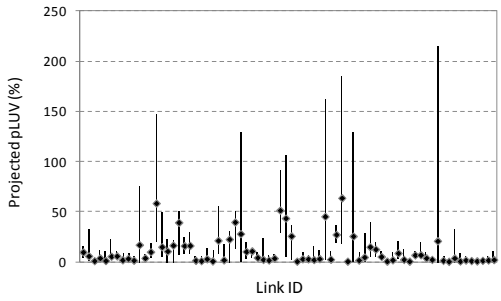
(2) *Static TE-aware LFA (LFA-ST)*: the selection of the LFA candidates takes into account one single "averaged" traffic matrix across the 24-hour duration. Such an approach aims at a static protection configuration that is *oblivious* to traffic dynamics.

(3) *Adaptive TE-aware LFA with x-minute re-configuration interval (LFA-ATx)*: LFA candidates are periodically re-computed and re-configured at a time interval of x minutes. In our experiments, we focus on the scenarios where x = 15, 30 and 60 (minutes).

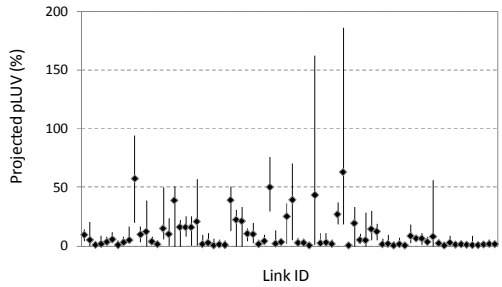
We first investigate the *pLUV* performance in Figure 7. From Figure 7(a) we can see that *LFA-TA* results in both the *highest* and the *most significantly varied* utilisations (between the min and the max values). For instance, the projected mean utilisation of the most loaded link already reaches 96.2%, with the worst-case being as high as 433%. The static TE-aware approach is able to significantly improve the situation as indicated in Figure 7(b). When adaptive LFA re-configurations are applied, the *pLUV* performance can be further enhanced, depending on the frequency of re-configurations. As we expected, the *LFA-AT15* case achieves the best performance, in terms of both the absolute utilisation values (max, min and mean) and the degree of utilisation variation on per-link basis. The worst case of projected 185.1% utilisation of the bottleneck link is due to the fact that a single unique protection path for a failed link contains a low-capacity link of 155Mbps which unavoidably leads to post failure congestion (also indicated in the follow-up *pMLU* analysis). With the decrease of re-configuration frequencies, the corresponding *pLUV* performance also noticeably deteriorates, due to the more insensitive reaction to traffic dynamics. Effectively the performance of *LFA-AT30* and *LFA-AT60* are very similar according to Figure 7 (d)(e).



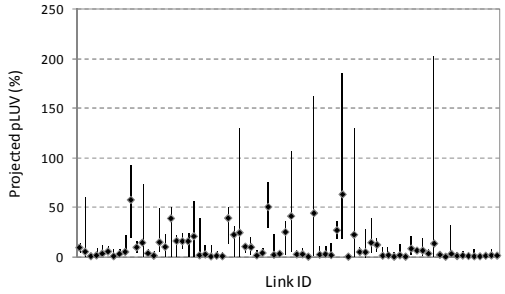
(a) *pLUV* performance in LFA-TA



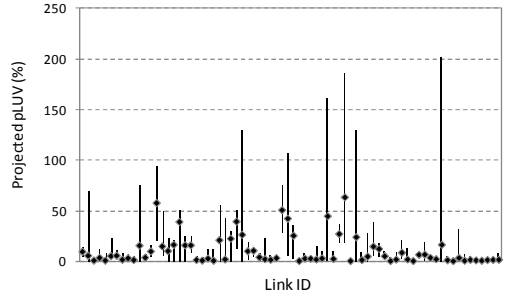
(b) *pLUV* performance in LFA-ST



(c) *pLUV* performance in LFA-AT15

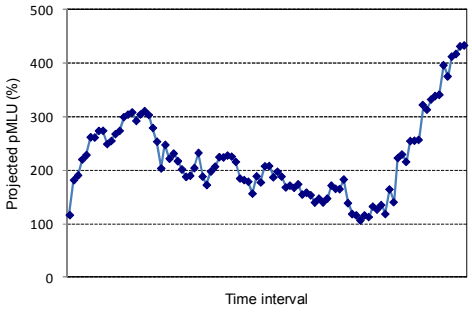


(d) *pLUV* performance in LFA-AT30

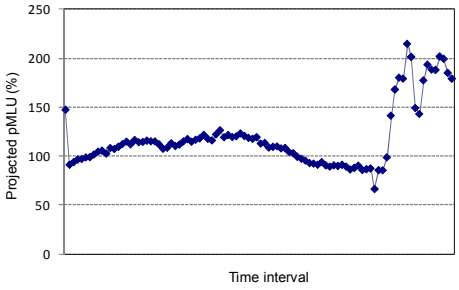


(e) *pLUV* performance in LFA-AT60

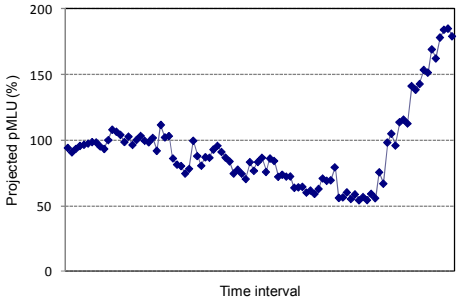
Figure 7. Projected *pLUV* performance comparison



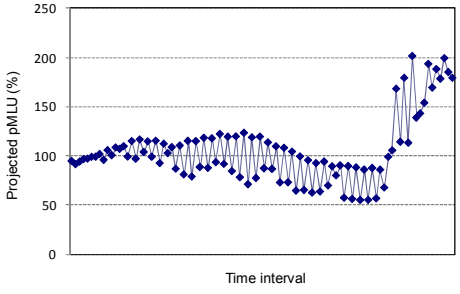
(a) *pMLU* performance in LFA-TA



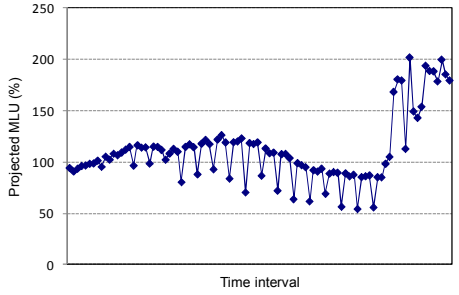
(b) *pMLU* performance in LFA-ST



(c) *pMLU* performance in LFA-AT15



(d) *pMLU* performance in LFA-AT30



(e) *pMLU* performance in LFA-AT60

Figure 8. Projected *pMLU* performance comparison

	LFA-TA	*LFA-ST*	*LFA-AT15*	*LFA-AT30*	*LFA-AT60*
Congestion-free percentage (%)	0.0	30.2	75.0	59.4	49.0

Table 1. Congestion-free percentage comparison

Now let's investigate the projected *pMLU* performance. It is widely accepted that maximum link utilisation (*MLU*) metric is one of the most important ones for evaluating the performance of Internet traffic engineering. Given that we are addressing the traffic distributions upon the post-failure activation of LFA protection paths, the projected *pMLU* is the actual metric to be examined. Figure 8 shows the relevant 24-hour *pMLU* dynamics under different configuration scenarios. From all the performance figures we can see that the projected *pMLU* exceeds significantly 100% towards the end of the period, and this is because the overall incoming traffic volume increases significantly during the busy time, and the failure of a link inevitably forces the affected traffic to follow a unique feasible LFA protection path containing low capacity links..

We can also see that all TE-aware approaches achieve dramatic performance improvements in comparison to the TE-agnostic solution (*LFA-TA*) that suffers from persistent post-failure congestion across the entire period (Figure 8(a)). On the other hand, adaptive re-configuration of LFA protection paths may further improve the *pMLU* performance against the static *LFA-ST* approach (Figure 8(b)), even though the latter aims at an oblivious FRR protection configuration for coping with traffic dynamics. By comparing the *pMLU* performance across individual adaptive approaches with different re-configuration frequencies, we can see that *LFA-AT15* achieves significantly higher congestion-free percentage than *LFA-AT30* and *LFA-AT60* across the 96 intervals (see Table 1). The congestion-free percentage is defined as the ratio between the number of intervals where the projected *pMLU* does not exceed 100% and the total number of intervals. It is worth mentioning that since the traffic monitoring operations are performed at every 15-minute interval [9], it is unknown whether the *pMLU* performance can be further improved with a higher frequency of LFA re-configurations.

Finally, we examine the frequency of LFA candidate re-configurations during the period. Here we only investigate the *LFA-AT15* scenario since it can be regarded as the "worst" case as far as computation overhead is concerned. Figure 9 plots the times of LFA candidate switching on per <repairing router, protected destination> pair basis across the 96 time intervals. Those pairs that have less than two LFA candidates are not included in the figure, as adaptive LFA switching operations cannot be applied to them. This leaves altogether 142 pairs having *non-deterministic* LFA protections in the GEANT network topology. As we can see from the figure, for the pair that has the most frequent LFA switching, the total number of times is 35 during the 96 intervals. That means on average there is an LFA switching for this pair every 2.7 time intervals, which corresponds to around 40 minutes. The majority of the pairs have less than 10 switches during the period, which means their LFA candidates are very infrequently switched. It is intuitive that by increasing the time interval, the corresponding LFA switching will be less frequent, but most possibly at the expense of less optimal performance. Hence it is obvious that an optimised trade-off should be sought between traffic optimisation and complexity.

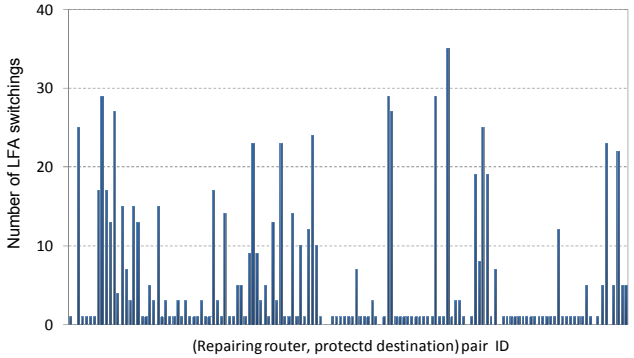


Figure 9. Number of LFA switching times for each <repairing router, protected destination> pair

V. CONCLUSIONS

In this paper we introduced a lightweight distributed approach for supporting adaptive re-configurations of LFA-based protection paths against traffic dynamics. The aim is to achieve a comprehensive fault and performance management solution for protecting real-time traffic from failures and potential post-failure congestion in self-managed networks. By enabling individual routers to periodically re-provision LFA protection paths as a background process according to the disseminated up-to-date traffic condition information, significant performance enhancement can be achieved in comparison to both TE-agnostic LFA and static TE-aware LFA approaches.

ACKNOWLEDGEMENT

The research leading to these results has been performed within the UniverSelf project (www.UniverSelf-project.eu) and received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 257513

REFERENCES

[1] A. Atlas and A. Zinin, "Basic Specification for IP Fast Reroute: Loop-Free Alternates", IETF RFC 5286, September 2008

[2] M. Shand et al, "IP Fast Reroute Using Not-via Addresses", IETF draft, draft-ietf-rtgwg-ipfrr-notvia-addresses-06.txt, October 2010

[3] S. Bryant et al., "IP Fast Reroute Using Tunnels," draft-bryant-ipfrr-tunnels-03, November 2007

[4] M. Shand and S. Bryant, "IP Fast Reroute Framework", IETF RFC 5714, January, 2010

[5] A. Kvalbein et al, "Post-failure routing performance with multiple routing configurations", Proc. IEEE INFOCOM 2007

[6] K. H. Ho et al, "Optimizing Post-Failure Network Performance for IP Fast Reroute Using Tunnels", Proc. ACM/ICST QShine 2008

[7] The GEANT network, http://www.geant.net

[8] D. Katz et al, "Traffic Engineering Extensions to OSPF Version 2", IETF RFC 3630, September 2003

[9] S. Uhlig et al, "Providing Intra-domain Traffic Matrices to the Research Community", ACM CCR, Vol. 36, Issue 1, 2006

[10] M. Zhang et al, "GreenTE: Power-Aware Traffic Engineering", Proc. IEEE ICNP 2010