Link Weight Optimization for Enhancing IP Resilience using Multi-plane Routing

Yu Guo, Ning Wang, Michael Howarth
University of Surrey, United Kingdom, GU2 7XH
{Y.Guo, N.Wang, M.Howarth}@surrey.ac.uk

Kin-Hon Ho
The Hong Kong Polytechnic University
cskhho@comp.polyu.edu.hk

Abstract— **With the increasing importance of the Internet for delivering personal and business applications, the slow re-convergence after network failure of existing routing protocols becomes a significant problem. This is especially true for real time multimedia services where service disruption cannot be generally tolerated. In order to ensure fast network failure recovery, IP Fast Reroute (FRR) can be adopted to immediately reroute affected customer traffic from the default path onto a backup path when link failure occurs, thus avoiding slow Interior Gateway Protocol (IGP) re-convergence. We notice that IGP link weight setting plays an important role in influencing the protection coverage performance in intra-domain link failures. Therefore in this paper we present an IGP link weight optimization scheme for backup path provisioning, which works on top of a multi-plane enabled routing platform. The scheme aims to optimize the path diversity among multiple routing planes. Due to the large search space of possible intra-domain link weights, in this paper we adopted a global search method based on a Genetic Algorithm to optimize the IGP link weights. Evaluation results show that in most cases a set of optimal link weights can be found which ensures that there are no more critical shared links among all the diverse paths on each routing plane. As a result, backup paths can be always available in case of single link failures.**

I. INTRODUCTION

Current IP routing protocols are generally based on single-path routing, which does not take full advantage of the high path richness offered by the present Internet topology. Richness in path diversity can bring numerous benefits to network operations and management, for example improved resilience, or quality of service. Today, a typical network failure may trigger the underlying IGP (Interior Gateway Protocol) to re-converge, which may take several seconds to resume normal packet forwarding; in addition, the BGP (Border Gateway Protocol) re-convergence procedure across multiple autonomous domains may take even longer [1]. However, such convergence period is long enough for networks not to be able to support real-time multimedia services; these typically require a loss-of-connectivity below 50 milliseconds in order to avoid user-perceived disruption. In recent years, IP fast reroute (FRR) techniques have been designed for seamless recovery following network failures. The key idea is for routers adjacent to the failure to immediately divert traffic affected by the failure onto pre-computed backup paths while suppressing the routing convergence process [2].

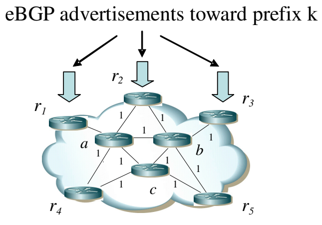
In our recently proposed multi-plane based FRR mechanism [3], controlled fast egress router switching can be employed for bypassing traffic around both intra- and inter-domain link failures by strategically deflecting the affected traffic to pre-determined alternate egress points. More specifically, in addition to the default path to the primary egress router which is used for traffic delivery in the failure-free state, additional backup egress routers can be pre-provisioned in alternate *routing planes*, so that the affected traffic can be immediately switched to the backup egress points in case of either intra- or inter-domain link failure on the default path. This FRR operation can be achieved by immediately remarking affected traffic from the default routing plane to one of the backup planes in which an alternate egress point towards the same destination is pre-determined. By employing this multi-plane based FRR technique, the overall failure recovery time can be significantly reduced. Nevertheless, the degree of path diversity largely determines the failure protection coverage in IP FRR techniques. In hop-by-hop IP routing, if the default next-hop overlaps all the backup next-hops for a specific destination prefix, affected traffic towards those destinations will not be able to successfully reroute around the failure. According to [3], careful selection of alternate egress points in backup routing planes for maximizing path diversity can significantly improve the overall failure protection coverage. However, where a next-hop is fully shared by all routing planes, direct packet deflection from the failed next-hop is not an option. Although the *crank-back* operation is introduced in [3], it needs some significant adaptation in the control/data plane of IP routers. Obviously, how to achieve full failure protection coverage without introducing additional complexity is an interesting research issue to be addressed.

One distinct observation is that the failure protection coverage depends on the physical topology as well as the routing configuration under the normal network condition. In particular, the IGP link weight setting in multi-plane routing platforms plays an important role in influencing the protection coverage performance on intra-domain link failures. A detailed example will be illustrated in the next section. This observation motivates us to investigate how failure protection coverage can be enhanced or even guaranteed by tuning IGP link weights in the multi-plane routing environment. In this paper we propose a Genetic Algorithm (GA) based heuristic solution for computing optimized IGP link weight in order to achieve guaranteed failure protection coverage. Based on our simulation experiments on two real operational networks and one synthetically generated network topology, we show that the proposed scheme with integrated IGP link weight optimization and intelligent backup egress point selection is able to achieve 100% failure protection coverage.

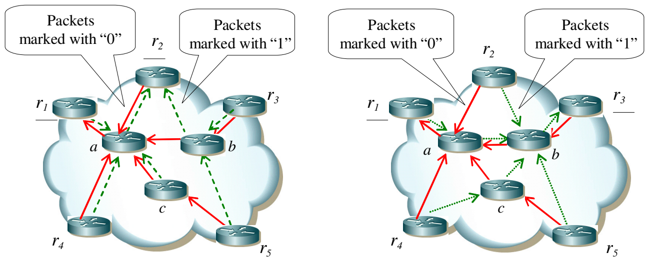
II. PROBLEM FORMULATION

A. *Multi-plane BGP FRR*

To achieve effective and fast failure recovery, alternative egress points used in back-up routing planes should be *pre-provisioned* carefully. The main idea is, for each remote destination prefix, to select both primary and backup egress points in the default and backup routing planes respectively in such a way as to maximize the degree of path diversity. As long as there are no shared links between the paths towards the primary and backup egress points, FRR can successfully reroute traffic to the backup egress point without traversing the failed link.



(a) Network topology



(b) Selected egress points: r_1, r_2 (c) Selected egress points: r_1, r_3

Figure 1. Example for multi-plane BGP reroute

Consider the example in Figure 1(a) where the IGP weight for each link is set to 1. We assume that two BGP routing planes are provisioned. Inter-AS traffic is injected into the network via border routers r_1 to r_5. Of these border routes, r_1, r_2 and r_3 have learnt a BGP route towards remote prefix *k*. We first assume r_1 and r_2 are selected as the primary egress points selected in plane 0 and 1 respectively. Planes 0 and 1 are considered as primary and backup routing planes respectively. In Figure 1(b), the shortest IGP paths from every router towards the selected egress points in planes 0 and 1 are indicated by solid line and dashed line respectively. If a link is included in the IGP paths over each of the planes towards a specific destination prefix, this link is regarded as a fully-shared link (or critical link) for that prefix. We can see in the example that links $r_4 \rightarrow a$ and $c \rightarrow a$ are fully shared by both two planes (i.e. the two links are traversed within both planes). This means that if either of these two links fails, local FRR to reroute traffic around the failed link is not possible because the backup path in plane 1 also include those two links. On the other hand, in Figure 1(c), if we select r_3 as the backup egress point instead of r_2, the backup IGP path indicated by dotted lines changes, and under this configuration, no links are shared between the two routing planes. In this case, no matter which of the two links fails, traffic can be successfully sent through the backup path to the destination prefix.

B. *IGP link weights tuning*

One distinct issue is how to maintain high protection coverage in the situation where egress points do not have high reachability towards remote destination prefixes. For instance, is it possible to still avoid fully shared links even if r_3 does not have a BGP route towards prefix *k*? We notice that IGP link weight setting can also influence the protection coverage on intra-domain link failures as an alternative approach to egress point selection. Considering the same topology as before, now shown in Figure 2 (a), r_1 and r_2 still remain as the primary egress points in planes 0 and 1, but the IGP link weights for certain links are configured differently from the previous example. The corresponding IGP shortest paths towards the same egress point r_2 are now different, shown in Figure 2(b) by dashed lines.

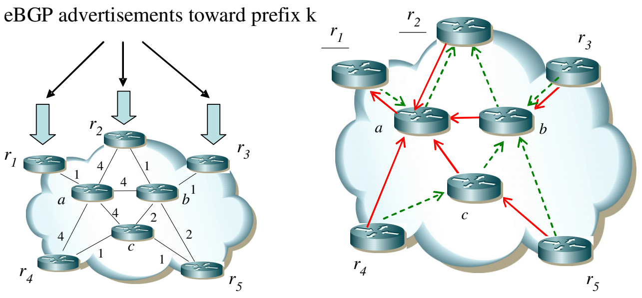


Figure 2. IGP link weight tuning for eliminating critical links

Here we can see that under the newly configured IGP link weights, the two critical links can be successfully eliminated without changing the original egress point selection (note in this example the degree of node r_3 is 1, therefore the shared link $r_3 \rightarrow b$ can not be avoided). This observation is also significant, as it provides opportunities to achieve fast failure recovery when border routers are not able to provide any external route towards the destination prefix. It also suggests that an integrated optimization of egress point selection with IGP link weight tuning in the management plane can be a promising approach to achieve a more efficient FRR.

C. *Network modeling and problem formulation*

As previously mentioned, the problem we consider in this paper is how to optimize IGP link weights for achieving maximum intra-domain path diversity for BGP fast reroute in the case of intra-domain link failures on top of the multi-plane

BGP platform. We model the network topology of one AS as a directed graph *(V, E)* with node set *V* and link set *E*. There is a set of border routers $I \subset V$ in each AS which can be regarded as ingress routers, through which traffic is injected into the AS. There is also a set of egress routers $J \subset V$ through which eBGP reachability advertisements on remote prefixes are received from neighboring ASes. We denote by *K* the set of prefix advertisements received across all the egress routers. For each prefix *k* ($k \in K$), let *Out(k)* denote the set of egress routers at which an advertisement for prefix *k* has been received. In the multi-plane BGP platform, we consider *M* logical planes to be pre-provisioned by the local AS, and a dedicated egress router is selected for each destination prefix *k* within each plane $0 \le m < (M-1)$. To enforce egress router selection, specific local preference (*local-pref*) values can be configured independently within each plane *m*, and the selected (primary) egress router will be assigned with the highest *local-pref* value. Our purpose is to find a set of IGP link weights $W = \{w_{u,v}\}: w_{u,v} > 0$, where each link $(u,v) \in E$ and $u, v \in V$. It is also worth mentioning that the intra-domain routing protocol running within the local AS is standard IGP such as OSPF or IS-IS, which is not multi-plane aware. In this case the IGP distance between each ingress/egress pair under the set of IGP link weights is the same across all routing planes.

The failure of a fully-shared link makes it impossible to perform fast reroute for traffic affected by the failure, and it is likely that IGP needs to re-converge before the traffic delivery service is restored. Therefore the number of fully-shared links should be minimized. A binary variable Q_k^l is defined to indicate whether intra-domain link *l* is a fully-shared link with regard to each aggregate flow injected from ingress router *i* and destined to prefix *k*. More specifically

$$Q_k^l = \begin{cases} 1 & \text{if } \sum_{m \in M} Y_k^{l,m} = M \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where

$$Y_k^{l,m} = \begin{cases} 1 & \text{if l constitutes the IGP path in plane m} \\ & \text{for traffic destined to prefix k} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

We also define another binary variable $X_k^{j,m}$ to indicate the actual egress point selection for prefix *k* in each plane *m*. Single Egress point Selection (SES) is adopted in our scheme, which means one single egress is selected for each prefix across all ingress routers within each plane. That is

$$X_k^{j,m} = \begin{cases} 1 & \text{if j is selected for prefix k as the egress router in plane m} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

In summary, the overall objective is to determine the value of a set of IGP link weights for each intra-domain link $l \in E$ and also the *local-pref* value setting for egress router selection in order to:

$$\text{Minimize} \sum_{k \in K} \sum_{l \in E} Q_k^l \quad (4)$$

Subject to the following constraints:

$$\text{If } X_k^{j,m} = 1 \text{, then } j \in Out(k) \quad \forall j \in J,\ 0 \le m < M \quad (5)$$

$$X_k^{j,m} \in \{0,1\},\ Y_k^{l,m} \in \{0,1\} \quad \forall j \in J,\ 0 \le m < M \quad (6)$$

Constraint (5) means the selected egress router *j* must be able to reach the destination prefix *k*. Constraint (6) makes sure that both variables *X* and *Y* are binary.

III. GENETIC ALGORITHM AND HEURISTIC ALGORITHM

In this paper we apply a Genetic Algorithm (GA) to solve the joint egress point selection and IGP link weight optimization problem for maximizing intra-domain path-diversity. Genetic Algorithm is a popular meta-heuristic method for solving optimization problems [7] [8]. We propose two GA strategies in this paper, namely the sequential approach and the integrated approach. The sequential approach solves the egress point selection and the IGP link weight optimization problems individually and sequentially, while the integrated approach solves the problems jointly.

In GA, each chromosome is a candidate solution to the optimization problem; these solutions evolve in each iteration toward the desired performance specified in the optimization objectives. Initially, the algorithm starts from a population of chromosomes which is randomly generated. Thereafter, the *fitness* value of each chromosome in the current population is evaluated according to a pre-defined cost function. From the current population, the algorithm randomly selects two parent chromosomes based on their fitness values. Chromosomes with higher fitness have a higher probability of being inherited by the next generation. Thereafter the selected parent chromosomes are recombined through *crossover* operations and randomly mutated to form child chromosomes, and then replace some parent chromosomes to generate a new population. Those iterations evolve to produce better solutions until either a predefined number of iterations has been reached, or the fitness values of individual chromosomes have converged.

A. *The Sequential GA Approach*

Our Sequential GA solves the joint egress point selection and IGP link weight tuning problems sequentially, in two stages. Intelligent egress router selection [3] is initially performed at the first stage to select the egress points for the default plane and all the backup planes. Then the outcome becomes an input to the IGP link weight optimization problem for the GA to determine an optimal link weight setting.

Table I. Illustration of an IGP link weight chromosome

IGP link weight	1	2	…	v-1	v
1	$w_{1,1}$	$w_{1,2}$	…	$w_{1,(v-1)}$	$w_{1,v}$
2	$w_{2,1}$	$w_{2,2}$	…	$w_{2,(v-1)}$	$w_{2,v}$
…	…	…	…	…	…
u-1	$w_{(u-1),1}$	$w_{(u-1),2}$	…	$w_{(u-1),(v-1)}$	$w_{(u-1),v}$
u	$w_{u,1}$	$w_{u,2}$	…	$w_{u,(v-1)}$	$w_{u,v}$

Our GA starts from a population of chromosomes with randomly generated IGP link weights $W = \{w_{u,v}\}: w_{u,v} > 0$ (as shown in Table I) where each link $(u,v) \in E$ and $u, v \in V$. The fitness of a chromosome takes into account the total number of fully shared links and is defined as

$$\text{Fitness}\, f = \frac{1}{\sum_{k \in K} \sum_{l \in E} Q_k^l + 1} \tag{7}$$

The higher the fitness of a chromosome, the fewer the number of critical links in the solution. Therefore, a high-fitness chromosome is desired. After calculating the fitness for each chromosome, two parent chromosomes are randomly selected, and a child chromosome is formed by crossover and mutation procedures on the parent chromosomes. This child chromosome then replaces the lowest-fitness chromosome in the current population. A new population is thus generated.

B. The Integrated GA Approach

Integrated GA aims to solve the joint optimization problem by simultaneously optimizing the egress router selection and the IGP link weight setting. In this scenario, each chromosome consists of two parts: one part is the egress router mapping, i.e. which egress router is selected in each routing plane (shown in Table II); the second part is the IGP link weights, which is the same as in the Sequential GA shown in Table I.

Table II. Illustration of the Egress router selection chromosome

Routing Plane	1	2	…	m-1	m
Egress Router	ER_1	ER_2	…	ER_{m-1}	ER_m

The initial generation of the chromosomes is formed of previous intelligent egress router selection [3] plus a set of randomly generated IGP link weights. The fitness value of each chromosome is calculated from a matching set of egress router selection and IGP link weights, and the same fitness function is used here as in the Sequential approach, shown in Eq. (7). At first, two parent chromosomes are randomly selected, where for each parent chromosome; a set of egress router selection and a set of link weights are included. Then both parts of the parent chromosomes are processed together under crossover and mutation procedures simultaneously. Thereafter the child chromosome is generated and replaces the corresponding part of the parent chromosome which has lower fitness value, and a new population is generated for the next iteration. Both approaches guarantee the produced solutions are no worse than the input solutions. The Integrated approach optimizes not only the IGP link weight but also the egress router selection at the same time, and hence its solutions are better than the Sequential one. The corresponding performance evaluation is presented in section IV.

The flow chart of the proposed algorithm is shown in figure 3.

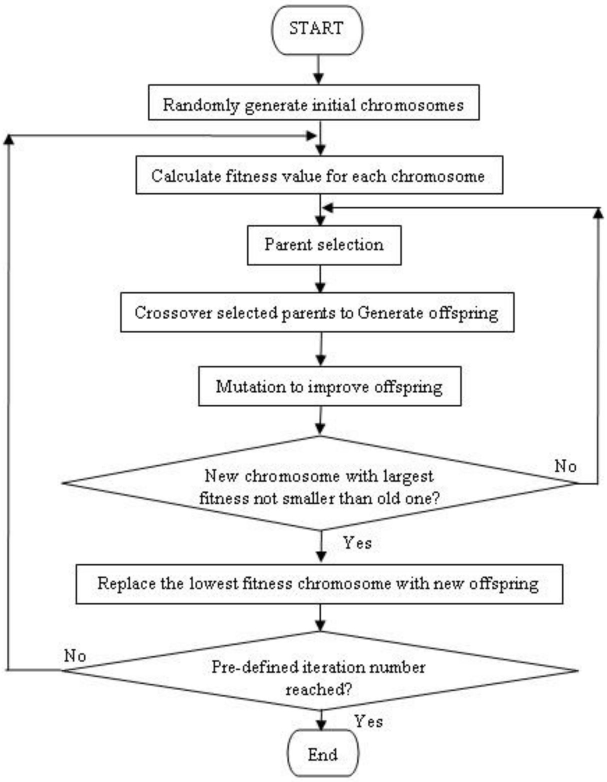


Figure 3. Flow chart for the proposed algorithm

Step 1. Based on [3], create a mapping table which maps intelligently selected egress routers for each routing plane *m* which can deliver the customer traffic towards each destination prefix *k* ($k \in K$) as shown in Table II, this being the first part of our chromosomes.

Step 2. Generate a set of randomly assigned link weights for the considered topology; this is the second part of our chromosomes. We generate *S* sets of link weights for all the intra-domain links $W = \left\langle w_1, w_2, \ldots w_{|E|} \right\rangle$ where |*E*| is the total number of intra-domain links in the network topology. The link weight value is generated between a range of 1 and *MaxLinkWeight*, and we define *MaxLinkWeight* to be 128 in this paper. Here the value of *S* should be exactly the same with *K* to complete the whole chromosome set. Therefore *K*=*S* initial chromosomes are produced.

Step 3. Decode each chromosome to calculate its fitness value. The fitness value is calculated by taking into account the egress router selection in each chromosome and the matching part of IGP link weights in the same chromosome. As we discussed before, fewer fully shared links among default plane and backup planes means better path diversity and therefore is desirable, so in terms of Fitness we are expecting largest value to be equal to 1 which means $\sum_{k\in K}\sum_{l\in E} Q_k^l$ =0 and the degree of path diversity is improved.

Step 4. From the whole set of chromosomes we need to select two parent chromosomes for reproduction. As we know, chromosomes that have higher fitness values have a higher probability of being inherited in the next generation. Therefore, we first sort all the chromosomes in the descending order according to their fitness values. We then partition them by half into two disjoined sets: UC-upper class (top *K/2* chromosomes with higher fitness values) and LC-lower class (bottom *K/2* chromosomes with lower fitness values). We select one parent chromosome C_U^i randomly from UC and another parent C_L^i randomly from LC in generation *i* for creating the child C^{i+1} in generation *(i+1)* through crossover and mutation procedures in the next step.

Step 5. We use a crossover probability threshold $K_C \in [0,0.5)$ to decide the genes of which parent to be inherited into the child chromosome in the crossover procedure. We also introduce a mutation probability threshold $K_M \in [0,0.1)$ to randomly replace some old genes with new ones. If a random number is larger than K_C, the gene of parent from upper class is inherited into the child; otherwise the gene of parent from lower class is inherited into the child. However if this random number is even lower than K_M, the old genes will be replaced by randomly generated new genes. It is worth mentioning that when we do the crossover and mutation, they work on both parts of the chromosomes together.

Step 6. Replace one parent chromosome in current population by the improved child chromosome. In our replacement scheme, the chromosome with lower fitness is always replaced. This is the end of a successful iteration as long as the largest fitness value in the new population is not lower than the largest fitness in the old population.

Step 7. Repeat step 3-6 until the maximum number of iteration is reached.

IV. PERFORMANCE EVALUATION

A. Experimental set up

We evaluate the performance of our algorithms on two operational network topologies: the Abilene network (AS11537) [10], which consists of 11 Point-of-Presence (PoP) nodes and 28 unidirectional links, and the GÉANT network (AS20965) [11] which consists of 23 PoP nodes and 74 unidirectional links. In addition to operational networks, our evaulation is also conducted on synthetic network topology topologies generated by BRITE [12]. The synthetic topologies contain 50 nodes with border routers being randomly selected, and we consider 5, 10, 15 and 20 egress routers in these topologies. According to [13], a small fraction of IP routing prefixes account for a large fraction of the Internet traffic. Therefore, we consider 100 popular routing prefixes in our experiments. As these routing prefixes are usually popular destinations, we assume that each egress router can reach all of them. For each experiment, 10 independent trials are performed and the average result is taken.

B. Evaluation results

For each experiment, the fitness value of the best solution produced by the Sequential GA approach and the Integrated GA approach are obtained for comparison. In addition, the values calculated under the actual and the initial randomly generated IGP link weights of each network topology are also presented for comparison (they are represented in following figures by “Average Value under Real Link Weight” and “Average Original Value” respectively).

1). ABILENE topology

Since the Abilene topology has 5 egress routers, up to 5 routing planes can be used. We examine up to 4 planes because any additional routing plane will not increase the overall intra-domain path diversity any further. We can see from Figure 4 that, as the number of planes used increases, the fitness value (i.e. the achievable path diversity) improves. This is expected since additional routing plane may offer more diverse intra-domain paths towards a destination prefix.

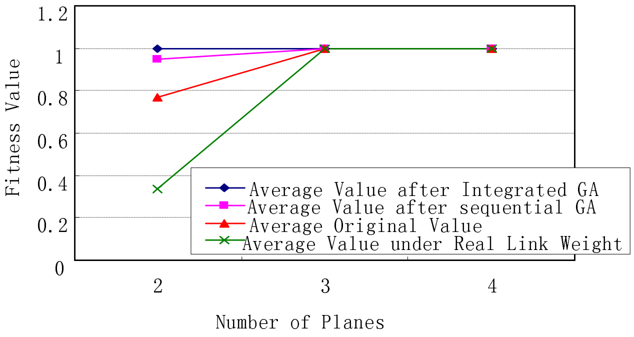


Figure 4. Average result of Genetic Algorithm for ABILENE topology

Under the real Abilene link weights, optimal path diversity (i.e. when the fitness value is equal to 1) can be achieved when 3 or 4 planes are used. This means that there is no fully shared links between primary plane and backup planes, so a GA for improving path diversity is not needed. However, when 2 planes are used, neither the real link weights nor the initially random generated link weights can achieve optimal path diversity. Therefore we can apply the GA to improve the result. We can see that the integrated approach works better than the sequential one as it can achieve optimal path diversity whereas the sequential approach is still 5% away from the optimum. Nevertheless, the results showed that our GA approaches can significantly improve the overall path diversity. Through our Integrated GA approach, optimal path diversity can be achieved with a minimal number of routing planes, thereby reducing the required overheads on implementation.

2). GEANT topology

We examine up to 5 planes in the GEANT topology (there are 7 egress routers in the topology). In comparison to the results in the Abilene network, Figure 5 shows that it is more difficult to achieve optimal path diversity in the GEANT due to network connectivity. In both ABILENE and GEANT, their real link weights perform worse than random link weights, which is because the real link weights are optimized for other aspects of Traffic Engineering, not for FRR. Under the real GEANT link weights, the performance is poor even when the number of planes is high. The performance is only 14% of the optimum even when 5 planes are used. In comparison, the approach under the set of randomly generated chromosomes works better, but still at best only 58% of optimum. Therefore we can apply the Genetic algorithm to improve the situation. With the increasing numbers of planes to be used, our GA approach finds better solutions. As with the Abilene results, the integrated approach works better than the sequential approach irrespective of the number of network planes evaluated.

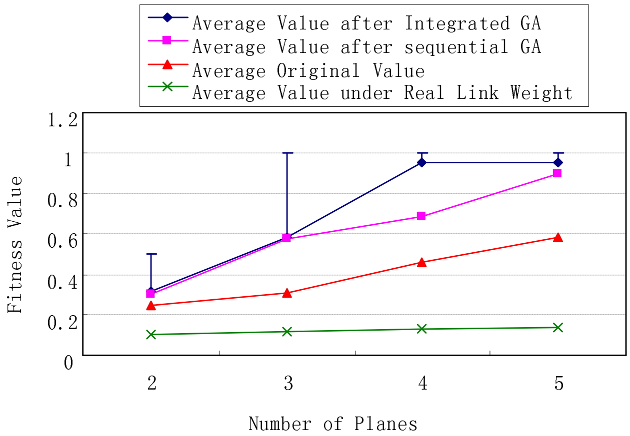
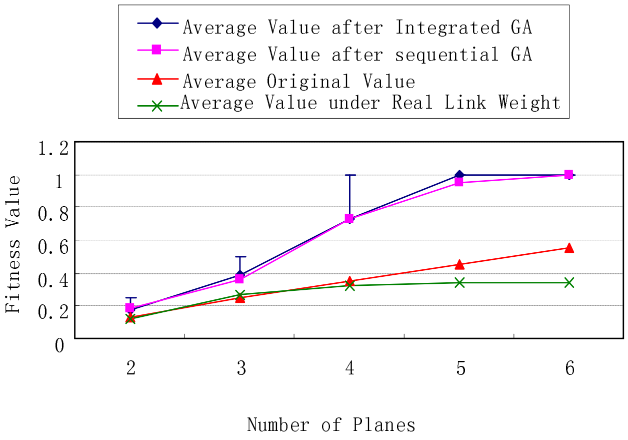


Figure 5. Average result of Genetic Algorithm for GEANT topology

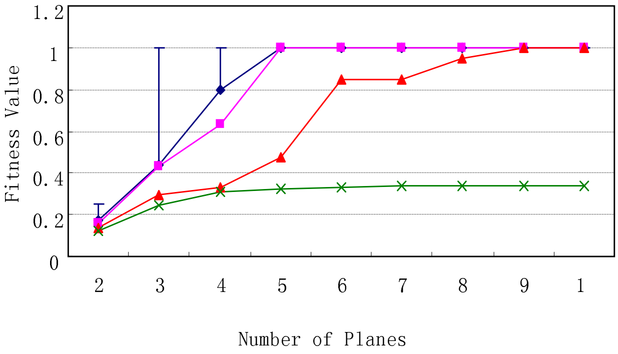
3). RANDOM topology

For the random BRITE topology, which has 50 PoP nodes, we consider two scenarios with 10 and 20 egress routers. For 10 egress routers the results are similar when using more than 6 routing planes, and for 20 egress routers, the results are similar when using more than 10 routing planes. Therefore we show in figure 6(a) and (b) the results for up to 6 planes and 10 planes when using 10 and 20 egress routers respectively.

We notice that that real link weights (assuming real IGP link weights are all equal to 1) performed badly in obtaining optimal results no matter how many planes we used, being at best 34% of optimal path diversity. The 100 sets of random generated chromosomes works better than real link weights but still we can see that when there are 10 egress router, the result is at best 55% of the optimal result. Therefore we apply Genetic algorithm to improve the result. As the number of planes increases, it becomes quicker to find the optimal results in both the Sequential approach and the Integrated approach. The Sequential approach and Integrated approach are quite similar in this case, however the Integrated approach still works better.



(a) 10 egress routers



(b) 20 egress routers

Figure 6. Average result of Genetic Algorithm for RANDOM topology

The results showed here are average values, and if we look at best case result of Integrated Approach, as shown in Figure 6, optimal path diversity can be achieved within 10,000 iterations if we use more than 3 planes when there are 10 egress routers and more than 2 planes when there are 20 egress routers. This is also true for the Sequential approach. The improvement in terms of fitness when using both approaches can be up to 66% on average compared to actual link weight when there are 10 or 20 egress routers. When more than 2 routing planes are used, optimal results can be guaranteed if we do not constrain the number of iterations. However when we use only 2 routing planes, an optimal result can not always be obtained; the best case is after 1,687,230 iterations when there are 20 egress routers in the topology and the Integrated approach is used, when an optimal result can be reached. If we use the Sequential approach, even after 4,000,000 iterations, there are still 2 shared links in the network. When there are only 10 egress routers in the topology, by running the Sequential approach for 615434 iterations or Integrated GA for 19414 iterations, total number of shared links can be reduced from 7 to 2, which is still quite a significant improvement in path diversity.

V. CONCLUSION

The current BGP does not take full advantage of rich path diversity offered by the present Internet topology since it employs single-path routing. Our previous work attempted to extend BGP with multi-plane functionality in order to achieve fast rerouting (FRR) under intra- or inter-domain link failure. By employing multi-plane BGP, traffic destined to a large number of routing prefixes can be protected by FRR. Yet, full failure protection (i.e. protection for all routing prefixes) may not be achieved under certain network configurations. In this paper, we have proposed an optimization problem of IGP link weight setting together with egress point selection in order to achieve full failure protection under intra-domain link failures. We presented an IGP link weight optimization scheme on top of the multi-plane enabled BGP platform which aims to optimize the path diversity among multiple routing planes. We proposed two Genetic Algorithms, namely Sequential and Integrated approach, to intelligently manipulate IGP link weights in order to solve the problem. We conducted our experiments on real and synthetic networks. The results showed that in most cases a set of link weights that result in no critical shared link among all the diverse paths on each routing plane can be found. As a result, optimal path diversity can be achieved. We also found that the Integrated GA approach works better than the sequential approach.

ACKNOWLEDGEMENT

This work is partially supported by the EU FP7 COMET (Content Mediation Architecture for Content-aware Networks) Project.

REFERENCES

[1] O. Bonaventure, C. Filsfiles et al, “Achieving Sub-50 Milliseconds Recovery Upon BGP Peering Link Failures”, IEEE/ACM Trans. on Networking, Vol. 15, No. 5, October 2007

[2] S. Nelakuditi, S. Lee et al, “Fast Local Rerouting for Handling Transient Link Failures”, IEEE/ACM Trans. On Networking, Vol. 15, No. 2, 2007

[3] N. Wang, Y. Guo, K.-H. Ho, M. Howarth and G. Pavlou, “Fast Network Failure Recovery Using Multiple BGP Routing planes”, IEEE Global Communications Conference (GLOBECOM), 2009

[4] M. Shand, S. Bryant et al, “IP Fast Reroute with Notvia Addresses”, IETF Internet draft, Febuary 2008, work in progress

[5] A. Kvalbein, A. Hansen et al, “Fast IP Network Recovery using Multiple Routing Configurations”, Proc. IEEE INFOCOM 2006

[6] J. Wang, S. Nelakuditi, “IP Fast Reroute with Failure Inferencing”, Proc. SIGOCOMM workshop on Internet Network Management, 2007

[7] M. Ericsson, M.G.C. Resende, P.M. Pardalos, “A Genetic Algorithm for the Weight Setting Problem in OSPF Routing”, Journal of Combinational Optimization, 6, 299-333, 2002

[8] Ning Wang, George Pavlou, “Traffic Engineered Multicast Content Delivery without MPLS Overlay”, IEEE Transactions on Multimedia, Vol. 9, No. 3, April 2007

[9] Anton Riedl, “A Hybrid Genetic Algorithm for Routing Optimization in IP Networks Utilizing Bandwidth and Delay Metrics”, Proc. 2nd IEEE International Workshop on IP Operations and Management (IPOM), 2002

[10] The Abilene Network: http://www.stanford.edu/services/internet2/abilene.html

[11] The GÉANT Network: http://www.geant.net.

[12] The BRITE topology generator, http://www.cs.bu.edu/brite/

[13] N. Spring et al, “Measuring ISP topologies with Rocketfuel”, IEEE/ACM Trans. On Networking, Vol. 12, No. 1, 2004.