A BGP SOLVER FOR HOT-POTATO ROUTING SENSITIVITY ANALYSIS

B. Quoitin and S. Tandel
Computing Sciences and Engineering Department
University of Louvain-la-Neuve, Belgium
{quoitin,tandel}@info.ucl.ac.be

Abstract The interactions between the IGP and BGP routing protocols which are running inside an ISP's network are sometimes hard to understand. The problem becomes particularly complex when there are dozens of routers/links and several thousands of destination prefixes. In this paper, we present a publicly available routing solver to evaluate routing what-if scenarios. The solver is able to model the complete network of an ISP and given the external routes learned by this ISP, to compute the paths towards all the destination prefixes. We demonstrate the use of our routing solver, C-BGP, by showing the results of an analysis of the link/router failure sensitivity in a transit network. Based on the analysis' results, we can pinpoint links/routers whose failure has an important impact on the selection of BGP routes. The deployment of protection techniques that are used for optical links, SONET-SDH and MPLS should be considered for these links/routers.

Keywords: Interdomain routing, BGP, network design.

1. Introduction

With today's needs for better Internet services, ISP's network operators increasingly care about the resilience and performance of their networks. They seek to build networks that will accomodate varying traffic load and be robust to link and router failures. However, achieving these goals is not easy since managing large IP networks requires a good understanding of the interplay of the intradomain and interdomain routing protocols.

There are two routing protocols that interact in a domain and their paths selection methods differ. Basically, the intradomain routing protocol such as IS-IS [12] or OSPF [14] is used to compute the paths between any two routers within the domain. The objective of the intradomain routing is to find the paths that best fit a previously selected metric which can be, for instance, the delay along the path or the bandwidth. Many network operators use the CISCO default metric, which is one over the bandwidth [22].

On the other hand, the interdomain routing protocol, BGP [19, 21], is responsible for the selection of interdomain paths. That is, it selects the paths towards the networks that are outside the domain. The incentive behind the design of BGP was to provide reachability among domains and the ability for any domain to enforce its own routing policies, i.e. controlling what traffic enters and leaves the domain, and where. To the contrary of the intradomain routing protocol, BGP does not optimize a global metric [11] but relies on a *decision* *process* composed of a sequence of rules [19]. The routing decisions depend on local preferences, the length of the interdomain routes, the intradomain cost to the egress router and other tie-breaking criteria.

It is difficult for an operator to figure out the routing decisions performed by its routers in case of link/router failures or in case of configuration changes. Especially when the network is composed of dozens of routers and there are several thousands destination prefixes. For this reason, we propose an open-source routing solver that can be used by ISP network operators to study routing what-if scenarios based on routing information collected in their network. The solver takes as input the network topology, the IGP weight and the BGP routes learned by the ISP network. As output, the solver computes for each router the routes selected towards all the interdomain prefixes.

We use the solver to evaluate various routing scenarios, such as the sensitivity to link/router failures and IGP metric changes. In particular, we are able to pinpoint which links/routers cause the largest number of routing changes in case of failure and are thus eligible for the deployment of protection mechanisms [26].

The paper is structured as follows. First, we describe in Section 2 our routing model and how we implemented it. In Section 3 we describe a sample routing scenario: the analysis of the impact of IGP changes on BGP routing. Then we apply our scenario to an operational transit network and show our results in Section 4. Finally, we conclude in Section 6.

2. Routing Model

In order to accurately model the routing in an ISP's network, we need to accurately model the path selection performed by the intradomain and interdomain routing protocols. That is, we must compute for each router the next-hop that would have been selected to reach each destination prefix, we have designed and implemented an open-source routing solver, C-BGP [16]. This solver models the topology of the network, the intradomain routes, the BGP route filtering and the complete BGP decision process without reproducing the time-consuming packet exchanges that occur between simulated routers in packet-level simulators such as SSFNet [15] or J-Sim [24]. We are therefore able to model large ISP networks. We have also used C-BGP to perform very large scale simulations with up to 30.000 BGP routers.

A sketch of the solver's internals is provided in Fig. 1. The solver can be configured using a CISCO-like syntax and it can read standard input formats such as MRT dumps [13] and libpcap IS-IS trace [9]. The solver provides convenient ways to analyze the results of its computations. A first possibility is to have a look at any routers's routing tables. Another possibility is to trace the route followed by packets sent by one router to another.

2.1 Topology and IGP Models

We represent the network as a graph where nodes are routers and edges are layer-three links between routers. We do not model the network's topology at the facility level. Each edge is weighted by the IGP metric of the corresponding link. The network graph can be built in many different ways, such as manually building a representation of an existing network, extracting information from an IGP protocol trace captured in the network, or even building a synthetic network. Our tool can read IS-IS traces in libpcap format thanks to the LISIS parsing tool [1] .

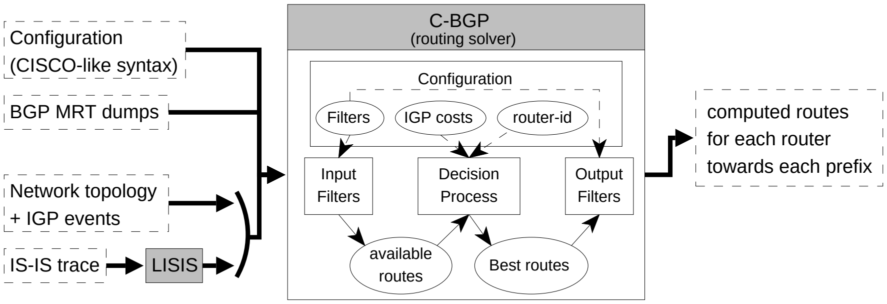

Figure 1. Overview of the routing solver.

The selection of paths by the intradomain routing protocol is modelled using the computation of the shortest paths based on the weight associated with each edge. In our model, we do not simulate the details of the intradomain routing protocol such as the propagation of the Link State Packets (LSPs). We compute the shortest paths in the solver using Dijkstra's SPF algorithm. These paths do not change until there is a weight change or a link/router failure. The model we have implemented currently supports a single area, the most common type of IS-IS deployment in large ISP networks.

In addition to the paths selected by the intradomain routes, the solver also supports the addition of static routes. These routes are typically used for peerings with neighbor domains or to direct traffic towards customers. The static routes do not participate in the intradomain routing protocol.

2.2 Models of the BGP routers

As explained earlier and shown in figure 1, our model contains the following information for each node that models a BGP router. First, an input and an output Adjacent Routing Information Bases (Adj-RIB-in/out) are used to store the BGP routes exchanged by the node with its neighbor BGP routers. The Adj-RIB-in contains the BGP routes that are available to this router. Second, a Local Routing Information Base (Loc-RIB) is used to store the best BGP routes selected by this node among the routes in the Adj-RIB-in. Finally, import and export filters are associated with each node.

The import and export filters that are applied to the BGP routes exchanged over the eBGP sessions. On commercial routers, those filters can be used to modify the BGP messages that are received or sent over the eBGP sessions. We studied BGP configurations from large ISP networks and found that they use complex BGP filters [18]. Besides the classical utilisation of the `local-pref` attribute for backup links and to prefer client routes over provider routes, those filters can contain complex operations on the BGP routes. Many ISPs rely on BGP communities for internal traffic engineering purposes or to allow their customers to influence the processing of their routes. Those various usages of the BGP communities [17] must be accurately modelled. Another frequently used attribute is the `MED`. It should be possible to consider or ignore this attribute in the received routes on a per eBGP session basis and it should also be possible to selectively use it on outgoing eBGP sessions. Another construct that we found frequently was the utilisation of AS-Path filters containing regular expressions. For this analysis, it is clear that an accurate model of an ISP network must be able to reproduce all the complexities found in those filters.

To be able to apply our solver to large ISP networks, we have developed a flexible conversion tool that is able to convert the actual BGP configuration of each router in the solver's configuration language. Our conversion tool is able to convert most of the BGP related commands supported by commercial routers. The current version supports both Cisco (IOS) and Juniper (JUNOS) router, but it was designed to easily support other languages. By using this conversion tool, a model of a large ISP network can be built. Furthermore, it is possible to update the model each time a BGP configuration changes and in some networks those configurations can change frequently.

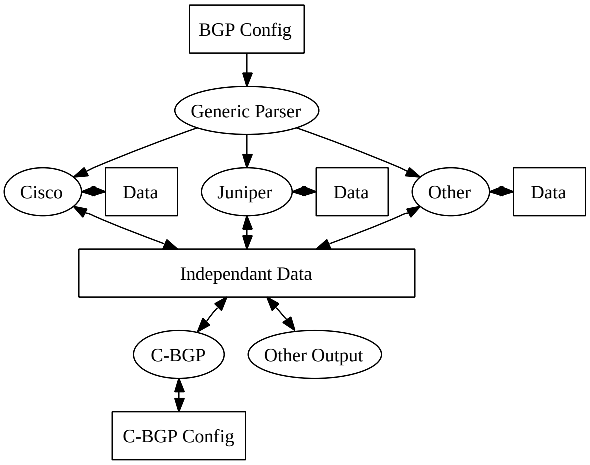

Figure 2. Overview of the converter.

Figure 2 represents the architecture of the converter. First a generic parser loads a specific parser in order to retrieve all the useful BGP commands. They are stored in an independent vendor data structure. Once the parsing has been done, the converter plugin converts this data in C-BGP configurations.

2.3 Computation of the Interdomain routes

Our model for the interdomain routing protocol relies on the computation of the paths that routers know once the BGP routing has converged [8]. For this purpose, we model the propagation of BGP messages and accurately reproduce the route selection performed by each router [19, 21]. Even if we model the propagation of BGP messages, the simulation we perform is static since we are not interested in the transient states of routing, but only in its outcome. This is reasonable approach since the large majority of Internet routes are stable in time [20, 25].

The model works as follows. Once the network topology is available and the intradomain routes have been computed, the solver begins the propagation of route advertisements. The solver starts with an arbitrary BGP router and advertises the routes known by this router. These routes either come from a manual configuration of the router or from a capture of the routes contained in the BGP RIB (Routing Information Base) of the router being modelled. The C-

BGP solver supports RIBs in MRT [13] format. For each route to be advertised, the solver builds UPDATE messages and sends them to the router's neighbors according to the output filters. For each BGP message to send, the solver looks up in the router's routing table to find the link along which the message must be forwarded to reach the next hop. The message is forwarded on a hop-by-hop basis until it reaches its final destination. The generated BGP messages are pushed on a single global linear first-in first-out queue that guarantees that the BGP messages are received in sequence (see Fig.3). In real routers, the BGP messages ordering is guaranteed by the TCP connections underlying the BGP sessions. The solver does this for all the BGP routers.

```
while (!msg_queue.empty()) {

  /* Get next message to process */
  (msg_type, dst, src, msg_content)= msg_queue.pop();

  /* Process message in destination router (dst) */
  if (msg_type == BGP_UPDATE) {

    route= msg_content;

    /* Does the destination router (dst) accept the route
       from the source router (src) */
    if (dst.in_filter(src, route) == ACCEPT)
      router.adj_rib_in[src].replace(route.prefix, route);
    else
      router.adj_rib_in[src].remove(route.prefix);

    /* Run the BGP decision process */
    dst.decision_process(route.prefix);

    /* Best route has changed, propagate to neighbors */
    if (dst.best_has_changed(route.prefix)) {

      foreach neighbor in (dst.neighbors) {

        /* Route can be redistributed to neighbor ? */
        if (router.out_filter(neighbor, route) == ACCEPT)
          msg_queue.push(BGP_UPDATE, dst, neigbor, route);

      }
    }
  }
}
```

Figure 3. Simplified algorithm for the BGP solver.

The solver continues the simulation by poping the first message from the queue, and waking up the router corresponding to the current hop of the message. If the BGP message is a WITHDRAW, the router removes from the corresponding Adj-RIB-in the route towards the withdrawn prefix, and runs the decision process. If the BGP message is an UPDATE, the router checks if the route it contains is accepted by its input filters. If so, the route is stored in the Adj-RIB-in and the router's decision process is run. The decision process retrieves from the Adj-RIB-ins all the feasible routes for the considered prefix, compares them and selects the best one. The selection process implemented in the C-BGP solver supports all the rules of the BGP decision process. The router then propagates its new best route to its neighbors by pushing new BGP messages on the global linear queue. The solver continues until the message queue is empty, which means that BGP has converged.

In addition, the interdomain model implemented in the C-BGP solver supports route-reflectors [1]. Route-reflectors are special BGP routers that are deployed to decrease the number of internal BGP (iBGP) sessions required inside a network. The solver is thus able to model ISP networks with an iBGP hierarchy.

3. Hot-Potato Routing Sensitivity

A sample application of our routing solver is the study of the sensitivity of a network to hot-potato routing. That is, we evaluate the impact of IGP changes on the selection of interdomain routes by BGP. This is important for an ISP for two main reasons. The first one is to predict which router will serve as an egress router to reach a given Internet prefix from a given ingress router. This choice is based on the routing information available through BGP, obtained from the various BGP peers of the domain, but it also depends on the internal paths from an ingress router to each of the peerings points (egress routers). Thus, this choice is sensitive to the internal paths computed by the intradomain routing protocol. The failure of a link, the failure of a router or a change in an IGP weight has an impact on the internal paths and the reachability. Therefore, it can have a dramatic impact on the BGP routing choices. As a consequence, such event can cause major traffic shifts or even cause Internet prefixes to become unreachable from certain ingresses.

The second reason is that changes in the intradomain routing of a domain may have an impact on the BGP routes that are announced outside the domain. An IGP event that seems unsignificant to the global Internet, as an IGP weight change, may be seen by all BGP routers in the Internet [5, 23]. In addition, some ISPs use the BGP Multi-Exit-Discriminator (MED) attribute when they have multiple peering links with the same neighbor domain [18]. The MED is used to inform the neighbor AS of the quality of each ingress router. The MED value can for instance contain the IGP cost of the path between the ingress and the egress routers. The neighbor AS will select the route with the lowest value of the MED. In this case, each time the IGP cost of the intradomain path changes, a BGP UPDATE message will be issued to update the MED value.

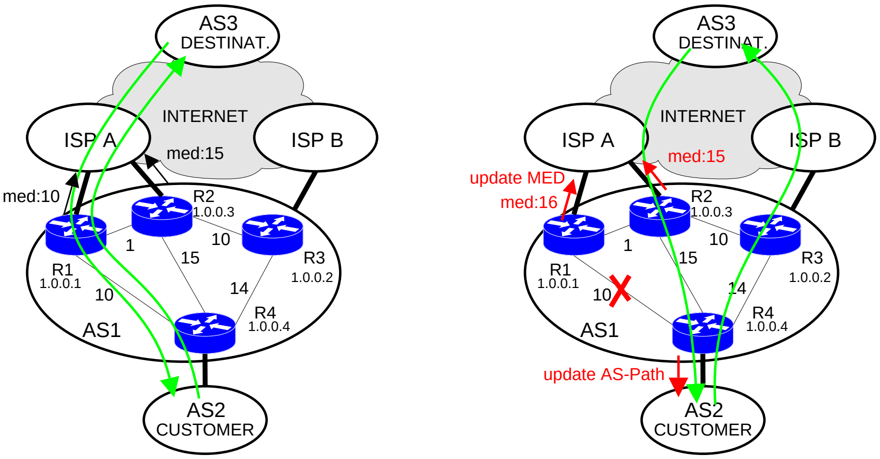

Figure 4. Example network: impact of a link failure.

We show an example domain in Fig. 4. This domain is composed of 4 routers, R1, R2, R3 and R4. The domain has two providers, ISP A which is connected to routers R1 and R2 and ISP B which is connected through R3. In addition, the domain has a single customer, AS2 which is connected to R4. We consider the traffic flows between the domain's customer and a remote destination network, AS3. When all links are up and running, (left side of Fig. 4), AS1 learns equal quality BGP routes towards AS3 from its ISPs. This is common in the Internet today [2]. The traffic sent by the customer to AS3 is received by R4 which forwards it to the closest egress router, R1, where it is handed to ISP A.

The traffic in the reverse direction enters the domain at R1. AS1 uses the MED attribute to indicate to ISP A the best ingress router to reach AS2's prefix. For this purpose, AS1 uses as MED value the IGP cost between the ingress routers and R4. ISP A will prefer the route with the lower value of the MED, which is through R1 in this case.

Now, if the link between R1 and R4 fails, the network will converge to the state shown in the right part of Fig. 4. R4 will select R3 as egress router to reach the prefix of AS3 since the closest egress router according to the IGP metric is R3. An update message is thus sent to AS2 in order to update the AS-Path. On the other hand, R1 and R2 advertise new BGP messages to ISP A in order to update the MED values. The best ingress router to reach AS2's prefix is now R2. The traffic coming from AS3 to AS2 now enters through R2.

3.1 Routing changes

Our methodology for studying the impact of IGP changes on the path selection is as follows. First, we build a representation of the network inside the routing solver, as explained in Sec. 2.1. We let the solver compute the routes in each router, then, we store a snapshot of the selected paths. This snapshot corresponds to the state of routing when everything is up and running.

We then apply our changes to the IGP: link failures, router failures or metric changes. We let the routing solver recompute the paths. Now, we compare the newly obtained routes to the snapshot. For each router and for each destination prefix, we classify the routing change as shown in Table 1.

Prefix up/down	The reachability of the prefix has changed. The prefix has either become up or down.
Peer change	The next-hop AS has changed.
Egress change	The egress router has changed, but not the nex-hop AS.
Intra Cost change	The cost of the intradomain path towards the egress router has changed.
Intra Path change	The intradomain path towards the egress router has changed, but the IGP cost remains the same.
No change	The route has not changed.

Table 1. Classification of routing changes.

The rationale behind the classification presented in Table 1 is to provide a ranking of the importance of routing changes. First, if the reachability of some prefixes is impacted by an IGP change (class *Prefix up/down*), that means that there is something wrong in the network design: a single failure (link or router) may cause important outages, with lots of destinations unreachable. Second, if the *next-hop AS has changed*, that means that BGP messages may have to be announced outside the ISP's network, with a new AS-path. Moreover, if the next-hop AS has changed, the cost charged by the new next-hop AS for the traffic may be higher than the previous one, causing additional financial costs. If there is an *Intra Cost change*, and if the routers advertise MED, BGP messages will be sent to inform of the new MED values. Finally, any *Intra Path change* may cause important traffic shifts in the domain and possibly congestion problems.

The computational complexity of the analysis is directly proportional to the number of prefixes in the routing tables. A full BGP routing table can contain more than 180.000 prefixes. However, when considering the routes announced by all the neighbors of one domain, it appears that a lot of prefixes are learned from the same neighbors, with the same BGP attributes (local-preference, as-path, MED, next-hop) [10]. The outcome of the decision process will be the same for these prefixes. Therefore, we group together the prefixes announced with the same attributes by the same neighbor routers in order to decrease the analysis time.

4. Case Study on the Géant network

In this section we describe the results of our evaluation of hot-potato routing sensitivity on the Géant [3] network (AS20965). Géant is the pan-European research network and it is operated by Dante. It carries research traffic from the european National Research and Education Networks (NRENs) connecting universities and research institutions. Géant has Points of Presence (POPs) in all the european countries [2] . The graph that models Géant is composed of 23 nodes and 37 links. In addition, there are 48 peering links.

Using the methodology described in Section 3.1, we simulated all the single-link/single-router failures in Géant and observed the impact on the BGP routes selected by each Géant router. We obtained the Géant topology from an IS-IS trace collected on one router, in libpcap format. We also obtained a dump of the routes advertised through the iBGP by all the Géant routers , in MRT format. The IS-IS trace and the BGP dump are dated from the 1st of July, 2004. We grouped the 140.334 prefixes in the BGP routing table in 403 clusters of prefixes learned from the same routers with the same attributes.

4.1 Routing changes

The results of the single-link failure analysis are shown in Fig. 5. On the x-axis, we show all the internal links of Géant. Upon request of Géant, we do not reveal the names of the routers. Instead, we label each router with a unique number. Each link is named from the two routers it connects. On the y-axis, we show the number of routing changes cumulated on all the Géant routers. The routing changes are classified according to Table 1. The links on the x-axis are ordered according to the number of routing changes caused by their failure. We observe that the failure of nearly 50% of the Géant links cause more than 100.000 routing changes. These links should probably be protected by the addition of parallel links, SONET-SDH protection or the use of MPLS protection tunnels [26].

In particular, the R1-R3 link causes more than 450.000 routing changes. This is due to its particular position in the Géant topology. Actually, the R3 router has a peering with an important peer AS as well as with many Géant POPs. Many POPs use R1 as a transit router to reach the R3 router. Therefore,

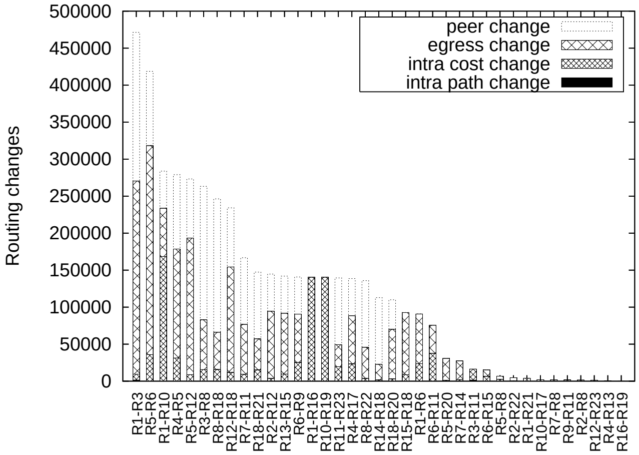

Figure 5. Most sensitive single link failure.

removing the R1-R3 link causes all the above POPs to use another route to reach the prefixes received by R3. The other links that cause large numbers of routing changes also connect to important border routers. We can also observe that that there is nearly no routing changes in the *Intra Path change* class. This is due to the absence of multiple equal cost paths between ingresses and egresses in Géant.

Fig. 6 shows the results of the single-router failure analysis. On the x-axis, we show all the routers of Géant. On the y-axis, we show the number of routing changes cumulated on all the Géant routers. The routers on the x-axis are ordered according to the number of routing changes caused by their failure. We observe that the impact of nearly half the Géant routers cause routing changes. The failure of a single router corresponds to the failure of all the links that are connected to this router. The consequence is that the routers whose failure cause the largest number of routing changes are the routers that connect to the most critical links identified in Fig. 5. Routers R5, R3, R18, R11 and R15 have peerings with important peers of Géant. Routers R1 and R6 are used by a large number of Géant POPs as transit routers to reach important destinations. We can also observe that a few routing changes concern losts of reachability. This occurs when routers that provide access to single-homed customers of Géant fail. The impact seems low in term of the number of prefixes but this means that the concerned customers of Géant have lost their connectivity to the research Internet unless they have another access through a commercial provider for instance.

4.2 Impact on traffic

In addition to studying the impact of failures on routing changes, we also evaluated the importance of traffic shifts caused by link failures. Indeed, when routes change some traffic flows may be forwarded along different intradomain paths. This will occur for traffic flows that are forwarded based on routes that have changed. Traffic shifts will modify the distribution of the traffic inside the

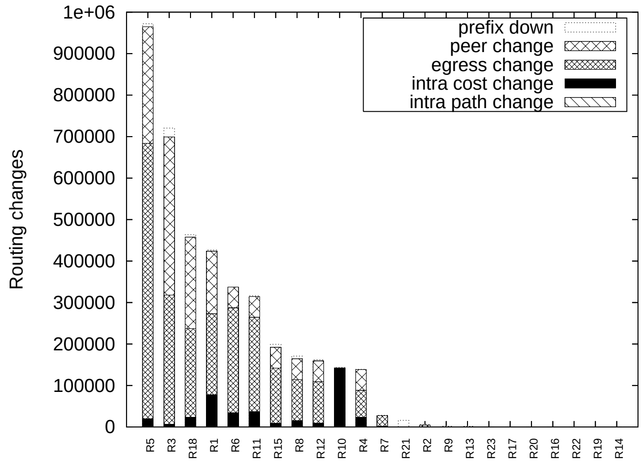

Figure 6. Most sensitive single router failure.

network and change the load of some links. As a consequence, some links can even become congested.

For the purpose of evaluating the traffic shifts caused by link failures, we obtained a prefix-prefix traffic matrix from Géant. A router-router traffic matrix is not sufficient since we need to re-route the traffic flows based on the BGP routing tables. The traffic matrix was obtained by collecting Netflow statistics on all the external interfaces of Géant. We used a one-day traffic matrix and we accurately reproduced how the traffic flows are forwarded inside Géant. By traffic flow, we mean the amount of IP traffic sent from a source prefix to a destination prefix, without regards to the protocols that are used. In order to reproduce the forwarding of one flow, we use the routing tables computed by C-BGP after each link failure. Then, we proceed on a hop-by-hop basis. We know the ingress router that received the flow and we perform a lookup in its routing table in order to find the next-hop router. We continue with the next-hop routers along the intradomain path until we reach the egress router. We add the volume of the flow to all the links we traversed. We do this for all the flows and obtain the volume of traffic carried by each individual link.

We show in Fig. 7 and Fig. 8 the traffic volume carried by each link after the failure of links R1-R3 and R5-R6 respectively. In these figures, we distinguish the links directions since each direction may carry a different volume of traffic. On the x-axis, we show all the directed links ordered based on the volume of traffic they carry before the failure. The y-axis shows the amount of traffic carried by the corresponding link. In each figure, we show two curves. The first one, labelled “default”, represents the links load when all the links are up and running. The second one represents the links load after the link failure.

Fig. 7 shows the links load after the failure of R1-R3. We observe that the traffic originally carried by the R1-R3 link now passes through other links. We also observe that the load of the previously most loaded link, which is R5-R6, has nearly doubled. This nevertheless does not cause congestion since the Géant links have a high capacity compared to the traffic volume that they

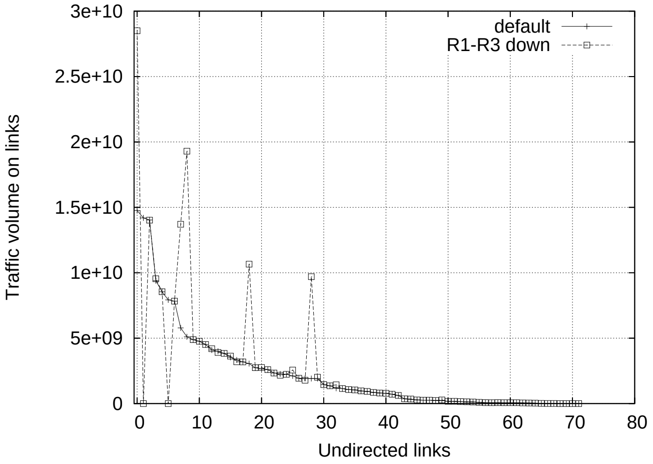

Figure 7. Impact of the failure of R1-R3 on the links load.

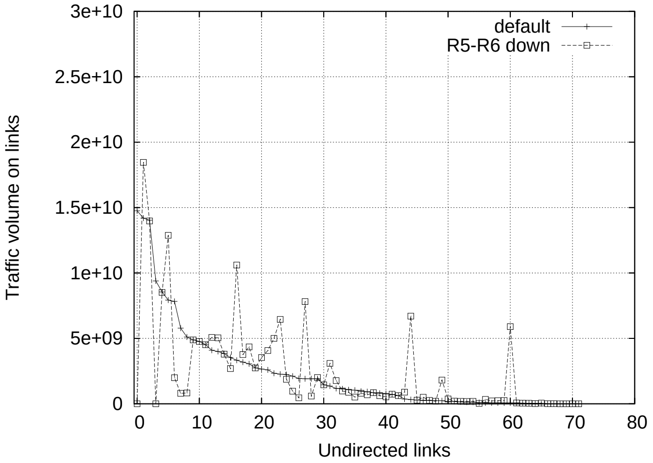

Figure 8. Impact of the failure of R5-R6 on the links load.

actually carry. However, this demonstrates that link failures do cause important traffic shifts.

In Fig. 8, we show the links load following the failure of link R5-R6. We observe that the traffic shifts are less localized than with the failure of R1-R3. More links have their load changed. This may seem surprising since Fig. 5 shows that the number of routing changes due to the failure of R5-R6 is lower than the number of routing changes due to the failure of R1-R3. As explained in Section 3, this is due to the complex interaction of IGP, BGP and the traffic. The failure of R1-R3 and the failure of R5-R6 have not the same impact on the intradomain routes computed by the IGP. These routes are used by ingress routers to reach egress routers. Then, the ingress-egress routes are used by the BGP routes to cross Géant. Depending on which routes are used to forward important amounts of traffic, the impact will differ.

Finally, Fig. 9 shows a summary of the impact of all the link failures on the traffic. The x-axis shows all the link failures ordered based on the load of the most loaded link. The y-axis provides the following statistics: the median, 5th- and 95th-percentile as well as the mean load and the load of the most loaded link. First, we observe that in Géant, the failure of some links cause a large increase of the maximum link load. There is even a link failure that causes the maximum link load to nearly double. Second, we observe that in Géant, the shape of the links load distribution is not much impacted. Indeed, the median and the 5th and 95th percentile does not move much.

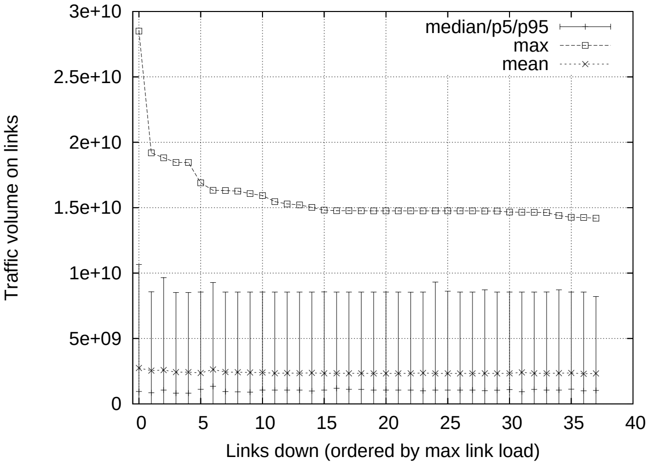

Figure 9. Most sensitive single link failure.

5. Related work

There are some prior works in the literature that are related to our project. First, our routing solver shares some similarities with AT&T's NetScope [6]. The aim of NetScope was to provide the networking industry with a software system to support traffic measurement and network modeling. NetScope was able to model the intradomain routing and study the implications of local traffic changes, configuration and routing. However, NetScope does not model the interdomain routing protocol, BGP. In addition, NetScope is not publicly available.

The BGP emulator designed and implemented by Nick Feamster at the MIT [4] is also similar to our routing solver. It computes the outcome of the BGP route selection process for each router in a single AS, based on a single snapshot of the network state. The BGP emulator relies on an algorithm that does not simulate the complex details of BGP message passing. It is however not publicly available and we do not know any utilisation of this tool for evaluating the impact of link/router failure.

Hot-potato routing has recently be studied in [23]. The paper describes the interaction between the IGP and BGP and proposes an analytical model of hot-potato routing. In addition, this paper proposes metrics to evaluate network sensitivity to hot-potato disruptions. These metrics can be used as tools to assist in the design of networks that are less sensitive to hot-potato disruptions. The model is able to compute the impact of hot-potato disruptions on the traffic matrix. However, the analytical model that is proposed does not take into account the iBGP hierarchy which is now frequent in many large ISP networks. The model does not take into account the BGP policies that are enforced by the ISPs. Finally, there is no publicly available implementation of the analytical model described in [23].

6. Conclusion

In this paper, we have described a pragmatic approach to study the interaction between an intradomain and an interdomain protocols in an ISP's network. We have implemented an open-source solver that is now available to network operators and to the research community. The solver is able to take input from a real ISP's network by relying on widespread file formats such as MRT dumps and libpcap IS-IS traces.

We also described a what-if scenario that can be evaluated by our solver. We showed how to evaluate the impact of link/router failures on the selection of BGP routes. This is an important problem faced by network operators and a hot research topic. Our tool makes possible the identification of the links/routers that are important for the routing and should be protected. We also studied the impact of failures on the traffic matrix and we showed that both the routing and the traffic points of view are important. We already performed additional analysis on Géant and Internet2 and we are currently collaborating with a large commercial transit network to apply our model to its network.

Other utilisations of the routing solver can also be envisaged. For instance, we could study the impact of a peering failure. An ISP could also use the tool to compare the addition of new peering links before signing a contract with a new provider or with a peer on an Interconnection Point. The solver could also be used to study the impact of route-reflectors on routing. In particular, it could help finding the best location to deploy route-reflectors. Finally, the impact of intradomain traffic engineering techniques such as [7] on BGP could also be studied.

7. Acknowledgments

This work was supported by: the Walloon Government within the WIST TOTEM project (`http://totem.info.ucl.ac.be`), the E-NEXT European Network of Excellence and a grant from France Telecom R&D. We are grateful to Tim Griffin and Richard Gass from Intel Research and to Nicolas Simar from Dante for providing us with the Géant data. We also thank Olivier Bonaventure, Cristel Pelsser and Steve Uhlig for their insightful comments on the project.

Notes

1. The LISIS tool was written by Olivier Bonaventure (`http://totem.info.ucl.ac.be`).

2. An overview map of the G´eant network is publicly available from `http://www.geant.net/upload/pdf/Topology_Oct_2004.pdf`.

References

[1] T. Bates, R. Chandra, and E. Chen. BGP Route Reflection - An Alternative to Full Me sh IBGP. Internet Engineering Task Force, RFC2796, April 2000.

[2] O. Bonaventure, P. Trimintzios, G. Pavlou, B. Quoitin (Eds.), A. Azcorra, M. Bagnulo, P. Flegkas, A. Garcia-Martinez, P. Georgatsos, L. Georgiadis, C. Jacquenet, L. Swinnen, S. Tandel, and S. Uhlig. *Internet Traffic Engineering*, pages 118–179. September 2003. Chapter of COST263 final report, LNCS 2856, Springer-Verlag.

[3] Dante. The GEANT Network. `http://www.geant.net`.

[4] N. Feamster, J. Winick, and J. Rexford. A model of BGP routing for network engineering. In *Proc. of ACM SIGMETRICS*, June 2004.

[5] A. Feldmann, O. Maennel, M. Mao, A. Berger, and B. Maggs. Locating internet routing instabilities. In *ACM SIGCOMM2004*, August 2004.

[6] Anja Feldmann, Albert Greenberg, Carsten Lund, Nick R eingold, and Jennifer Rexford. Netscope: Traffic engineering for ip networks. *IEEE Network Magazine*, March 2000.

[7] B. Fortz, J. Rexford, and M. Thorup. Traffic engineering with traditional IP routing protocols. *IEEE Communications Magazine*, October 2002.

[8] T. Griffin and G. Wilfong. An analysis of BGP convergence properties. In *Proc. of ACM SIGCOMM*, September 1999.

[9] The Tcpdump Group. libpcap: packet capture library. `http://www.tcpdump.org`.

[10] B. Halabi. *Internet Routing Architectures (2^{nd} edition)*. Cisco Press, 2000.

[11] B. Huffaker, M. Fomenkov, D. Plummer, D. Moore, and K. Claffy. Distance Metrics in the Internet. In *Proc. of IEEE International Telecommunications Symposium (ITS)*, September 2002.

[12] J. Moy. *OSPF : anatomy of an Internet routing protocol*. Addison-Wesley, 1998.

[13] Merit Network. MRT: multi-threaded routing toolkit. `http://www.mrtd.net`.

[14] D. Oran. OSI IS-IS intra-domain routing protocol. Request for Comments 1142, Internet Engineering Task Force, February 1990.

[15] B. J. Premore. SSF Implementations of BGP-4. available from `http://www.cs.dartmouth.edu/~beej/bgp/`, 2001.

[16] B. Quoitin. C-BGP, an efficient BGP simulator. `http://cbgp.info.ucl.ac.be/`, September 2003.

[17] B. Quoitin and O. Bonaventure. A survey of the utilization of the BGP community attribute. Internet draft, draft-quoitin-bgp-comm-survey-00.txt, work in progress, March 2002.

[18] B. Quoitin, S. Uhlig, C. Pelsser, L. Swinnen, and O. Bonaventure. Interdomain traffic engineering with BGP. *IEEE Communications Magazine*, May 2003.

[19] Y. Rekhter, T. Li, and S. Hares. A Border Gateway Protocol 4 (BGP-4). Internet draft, draft-ietf-idr-bgp4-26.txt, work in progress, October 2004.

[20] J. Rexford, J. Wang, Z. Xiao, and Y. Zhang. BGP routing stability of popular destinations. In *Proc. Internet Measurement Workshop*, November 2002.

[21] J. Stewart. *BGP4 : interdomain routing in the Internet*. Addison Wesley, 1999.

[22] CISCO Systems. Technical support & information. `http://www.cisco.com`.

[23] R. Teixeira, T. Griffin, G. Voelker, and A. Shaikh. Network sensitivity to hot potato disruptions. In *Proc. of ACM SIGCOMM*, August 2004.

[24] H. Tyan. *Design, realization and evaluation of a component-based compositional software architecture for network simulation*. PhD thesis, Ohio State University, 2002.

[25] S. Uhlig, V. Magnin, O. Bonaventure, C. Rapier, and L. Deri. Implications of the topological properties of internet traffic on traffic engineering. In *ACM Symposium on Applied Computing*, March 2004.

[26] J.-P. Vasseur, M. Pickavet, and P. Demeester. *Network Recovery: Protection and Restoration of Optical, SONET-SDH, and MPLS*. Morgan Kaufmann, 2004.