IJREAT International Journal of Research in Engineering & Advanced Technology, Volume 2, Issue 2, Apr-May, 2014
**ISSN: 2320 – 8791 (Impact Factor: 1.479)**
**www.ijreat.org**

# Avoiding Link Failure Using Seamless BGP Reconfiguration

**N.Srinivasan[1], Mrs.N.Belina[2]**

[1]PG Student, Department of Computer Science and Engineering, Sriram Engineering College, Perumpattu-602 024, Tamil Nadu, India

[2]Assistant Professor, Department of Computer Science and Engineering, Sriram Engineering College, Perumpattu-602 024, Tamil Nadu, India

**Abstract**

External BGP (eBGP) controls routing information exchange between different ISPs, while internal BGP (iBGP) distributes inter domain routing information among routers in the same ISP. Both eBGP and iBGP configurations are critical for an ISP, as they typically enforce commercial agreements with other ISPs and traffic engineering policies. During the life of a network, iBGP and eBGP configurations evolve. The organization of iBGP sessions typically need to be periodically modified, e.g., when new iBGP routers are introduced while older ones are either decommissioned or moved to less traffic intensive areas. Also, iBGP configuration changes can be triggered by changes to the underlying Interior Gateway Protocol (IGP). IGP changes are often performed in ISPs e.g., to optimize the usage of network resources by fine-tuning of IGP weights. Unfortunately, IGP configuration adjustments can affect iBGP routing choices, possibly leading to routing and forwarding inconsistencies, as well as undesired side effects on internal and external traffic flows. IGP changes may thus require iBGP configuration changes. Similarly, eBGPconfiguration need to be changed. A typical use case is the provisioning of a new customer, which requires to establish new eBGP sessions on some border routers. As commercial relationships between ISPs change, operators also need to modify their eBGP routing policies. Prominent examples include the so-called "peering wars" that led to the depeering of large ISPs. As a result, routing policies are changed on a daily basis in some networks. The impact of changes to either iBGP or eBGP configuration is hard to predict. Indeed, local changes on one BGP router can affect routing information viewed by remote routers in a domino effect in which the organization of iBGP sessions and message timings may play a critical role. Unfortunately, network administrators lack methodologies and tools to perform reconfigurations with minimal impact on the traffic. Only a few best practices are available , but they typically focus on simple reconfiguration cases. Even worse, current best practices barely take into account the possibility of creating routing and forwarding anomalies during the reconfiguration process. In this paper, we address the problem of changing the BGP configuration of an ISP with no impact on data plane traffic. We consider both eBGP and iBGP configuration changes. Problem of finding an operational ordering of BGP reconfiguration steps which guarantees no packet loss. Unfortunately, finding such an operational ordering, when it exists, is computationally hard. To enable lossless reconfigurations, we propose a framework that extends current features of carrier-grade routers to run two BGP control planes in parallel. We present a prototype implementation and we show the effectiveness of our framework through a case study.

**Index Terms**—*Border Gateway Protocol (BGP), configuration, reconfiguration, migration, Virtual Routing and Forwarding (VRF), Ships in the Night.*

## 1. Introduction

The contribution of this paper is threefold. First, we show that long-lasting routing and forwarding anomalies can and do occur during BGP reconfigurations even when the initial and the final BGP configurations are anomaly-free. We simulated BGP reconfigurations in a Tier-1 network observing that a significant number of anomalies persists for large parts of the reconfiguration process. Such a study exposes the fragility of correct BGP configurations, as different kinds of anomalies can be triggered even by simple changes on a single BGP session. Second, we consider the problem of finding an ordering of configuration changes which guarantees an anomaly free reconfiguration process. We show that this problem is computationally intractable. Even worse, we present simple cases in which an anomaly-free reconfiguration ordering does not exist at all. Third, we propose a generic framework that enables lossless BGP reconfigurations. Our solution is based on enabling routers to support two independent and isolated control planes, by slightly extending current technology. Our proposal provably prevents both long-lasting and transient problems due to configuration changes. We describe a possible implementation of our framework, and we present a working prototype. Finally, we show the effectiveness of our approach through a use case and we study its scalability. Observe that, beyond addressing current needs of network operators, our framework can be leveraged to achieve additional agility and flexibility, possibly leading to to competitive advantages for ISPs. For example, the ability to frequently change eBGPconfiguration enables ISPs to adapt routing policies to observed traffic trends and turn off network devices during idle time (e.g., during the night). By rapidly and safely switching preference of routes received from their eBGPneighbors, ISPs can also reduce their transit costs, and take full advantage of services (e.g., Equinix

Direct [12]) aiming at more flexible establishment of upstream connectivity. The rest of the paper is organized as

follows. Section II provides some background. Section III states the BGP reconfiguration problem and highlights deficiencies of current best practices. Section IV presents examples in which anomaly free reconfiguration orderings do not exist. Section V and Section VI describe our framework and its evaluation. Section VII presents related work. Section VIII contains conclusion

## 2. BGP and Configuration Correctness

For each destination IP prefix, each BGP router selects its best route among the routes it has received from its neighbors. A route can be seen as a path on the BGP network graph associated with a set of attributes. Since BGP has an internal loop-detection mechanism [1], each route is a node simple path. Route attributes are used by the BGP decision process [1] to select the best route. The BGP decision process (summarized in Table I) applies steps sequentially until there is only one route left. Indeed, the iBGP topology regulates which routes are known by each router, the IGP topology affects route preference at different routers, and _ determines what routes are available towards each prefix. In the following, we refer to border routers receiving an eBGP route to a prefix as egresspoints for that prefix. Previous work has shown that both eBGP and iBGP configurations can result in incorrect routing and forwarding, as a consequence of conflicting routing policies. BGP correctness issues can be classified in signaling, forwarding, and dissemination anomalies. Signaling anomalies [4], [15], [16] or routing oscillations occur when BGP routers are unable to converge to a single stable routing state. Oscillations can delay BGP convergence for a possibly indefinite amount of time, wasting resources and negatively impacting traffic. In iBGP, routing oscillations are due to the interaction with the underlying IGP, and can be further classified into two categories: those induced by partial lack of visibility due to the route reflection topology and those induced by the peculiar semantics of the MED attribute. We disregard problems due to MED specific setting in this paper, 3 because of space limitations. We show in the following that BGP reconfigurations can be responsible for routing issues even when MED is ignored and BGP policies are very simple. Furthermore, our solution also prevents MED-induced issues during the reconfiguration (see Section V). Forwarding anomalies [4], [17] occur when routers make inconsistent forwarding choices. Besides inducing suboptimal forwarding and complicating network management and troubleshooting ,forwarding anomalies can also disrupt traffic by causing packet deflections and forwarding loops. Dissemination anomalies [5] or Loss of prefix Visibility(LoV) consist in improper route propagation that results in some routers having no route to a given prefix. When this happens, packets are either dropped because no route is known or forwarded according to a less-specific route. The former case creates a traffic blackhole, the latter results in inconsistencies between routing and forwarding plane. We say that a BGP configuration is anomaly-free if

no signaling, forwarding and dissemination anomalies occur for any destination prefix. In the worst case, each prefix is

learned from a different subset of border routers. In this case, we are consistent with previous work on configuration correctness [4], [5].

## 3. Seamless BGP Reconfigurations

In this section, we define the BGP seamless reconfiguration problem. By analyzing historical configuration changes deployed in a Tier-1 ISP, we show that the problem has practical relevance. Moreover, we show that incremental approaches and current best practices [9], [10] incur the risk of introducing reconfiguration-induced anomalies. Finally, by means of simulations, we quantify the disruptions generated by existing approaches in simple reconfiguration scenarios.

### A. Problem Statement

We define a reconfiguration, or migration, as a sequence of configuration changes that turn an initial BGP configuration into a final one. We will use indexes to denote intermediate configurations. For example, Ct is the BGP configuration at time t. We define two special indexes i and f that refer to the initial and the final time in the reconfiguration, respectively. Throughout the paper, we assume Ciand Cfto be given as input and to be anomaly-free. Also, the IGP configuration and the eBGP routes to each destination are supposed not to change during the reconfiguration. As a consequence, the combination of egress points for any destination (i.e.,_) changes only as an effect of local eBGP configuration changes. We show that BGP reconfigurations are hard even when these assumptions hold. To improve network agility and quickly react to routing changes, we aim at enabling BGP reconfigurations of production networks at anytime, potentially even during peak hours. Performing reconfigurations during maintenance windows would be extremely slow, as maintenance windows are typically short and rare (e.g., few hours per month) because of stringent Service Level Agreements (SLA). To respect such SLAs, simply shutting down and restarting the network with the new configuration is also not viable. In addition, simultaneously overwriting configuration files on all the routers is unpractical, as it is likely to generate huge control plane churn, which, in turn, can overload routers. Moreover, the latter approach does not allow operators to keep the reconfiguration process under control, turning misconfigurations or human errors (e.g., typos) into a management nightmare .Hence, an incremental approach is needed. In this paper, we disregard migrations where router configurations are modified on a per-prefix basis. Indeed,

given the size of current BGPRIBs, per-prefix migrations incur severe penalties in the speed and the ease of management of the migration. Hence, we consider

reconfigurations in which the final configuration is installed at one router at each step. We define a migration as seamless if for any migration step j, with $i \leq j \leq f$

• Cjis anomaly-free;
• Cjis not subject to unintended traffic shifts.

An unintended traffic shift is a change in the best path selectedby a router to a given prefix in which the egress point is neither the initial nor the final one. We also talk about an unintended traffic shift when a router switches between the initial and the final egress points multiple times. By definition, unintended traffic shifts are peculiar to the reconfiguration problem, that is the reason why they have not been studied in prior work. We consider avoiding unintended traffic shifts as a primary requirement for seamless BGP migrations, since BGPnext-hop changes can disrupt traffic engineering policies (e.g., forcing traffic to exit from other continents), adversely impact costs (e.g., swelling traffic flows on transoceanic links), and significantly increase the likelihood of congesting some links(e.g., under-provisioned backup links). Personal communications with operators confirmed that avoiding unintended traffic shifts is among their most relevant concerns. During migrations that are not seamless, routing and
forwarding anomalies occur in intermediate configurations. These anomalies persist until another intermediate configuration(or the final one) is reached, which might require several migration steps. We refer to such persistent anomalies as migration anomalies. Migration anomalies can cause disruptive effects, among which forwarding deflections and loops, unintended traffic shifts, traffic blackholes, congestions, unnecessary iBGP churn, and unnecessary eBGP updates which increase the risk of route dampening [18]. On the contrary, we do not consider short-lived protocol-dependent issues, like those occurring transiently during protocol convergence, as they are unrelated to BGP reconfigurations. Nevertheless, our proposed solution also prevents this kind of issues to occur during the reconfiguration (see Section V).

## B. Frequency of BGP Reconfigurations

To illustrate the frequency of BGP configuration changes, we analyzed the BGP configurations of approximately 20% of the routers of a Tier-1 ISP, from April 2010 to July2011. The considered routers were new generation routers progressively added to the network during the considered

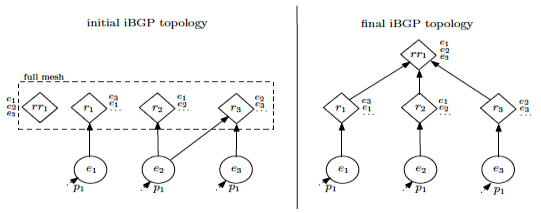



Fig. 1. An example in which the bottom-up strategy, suggested by the current best practices, creates routing oscillations during the reconfiguration.

timeframe. Among those routers, some have been replaced after their introduction by other routers of a different brand this happened 17 times. Among the configuration changes, sessions additions and removals were the most common .Sessions additions happened 5, 828 times, encompassing 976eBGP sessions and 4, 852 iBGP sessions. Session removals were less frequent but still not rare, as they happened 236times for eBGP sessions and 1, 440 times for iBGP sessions .At each router, eBGP sessions were typically added in groups, while iBGP sessions were mostly added in pairs of redundant sessions with two route-reflectors. By only looking at route map names, we also registered 41 changes of inbound eBGPpolicy and 77 modifications of outbound eBGPpolicy. Finally, we collected less frequent miscellaneous changes, encompassing the promotion of a router to the role of routereflector(11 times), AS number modification on an eBGPpeer (8 times), and address family enabling (3 times) and disabling (5 times) on eBGP sessions. These numbers testify that reconfigurations of already established BGP sessions are also performed by operators, even if less frequently than the addition or the removal of BGP sessions.

## C. Current Best Practices Provide No Guarantees

Currently, network operators can only rely on a few rules of thumb that only apply to simple topological changes, like the replacement of a fully-meshed iBGP topology with a two layer route reflection hierarchy [9], [10]. In the following, we show that current best practices provide no guarantees on the absence of migration anomalies. To be as general as possible, we consider as current best practice an extension of the procedures proposed in [9], [10] devised after discussions with operators. Such an extension consists in reconfiguring routers, one at the time, on a per-layer basis, in a bottom-up fashion (i.e., starting from the bottom layer up to the top one). Each router r is reconfigured by activating all the sessions r has in the final configuration before shutting down all the sessions r maintains exclusively in the initial configuration. An example of migration oscillation created by the bottomup approach is reported in Fig. 1. The graphical convention we adopt in the figure is the same we use for iBGP topologies throughout the paper. Circles represent iBGP routers having no clients, while diamonds represent route-reflectors. Sessions between clients and route-reflectors are drawn as lines terminating with an arrow on the side of the route-reflector However,

despite our assumption of anomaly-free initial and final configurations, we proved in [19] that finding an operational ordering that guarantees no migration anomalies is NP-hard in both the iBGP and the eBGP cases. Indeed, we showed a polynomial-time reduction from 3-SAT problem, based on mapping Boolean assignments of a 3-SAT instance to reconfiguration orderings. Even worse, in this section we present examples in which every operational ordering leads to migration anomalies. We first tackle iBGP topology

changes, then we address the problem of changing eBGP policies.

## A. iBGP Topology Changes

The problem of changing the iBGP topology can be formalized as follows. We refer to an operational ordering that guarantees a seamless migration as seamless ordering. Session Ordering Computation Problem (SOCP): given the initial and final iBGP topologies Bi and Bf , compute a seamless ordering in which to add sessions in Bf \Bi and to remove sessions in Bi \ Bf .To be as general as possible, we allow multiple sessions involving the same router to be simultaneously added or removed at each migration step. This closely reflects the degree of freedom that operators have. Indeed, multiple sessions involvingthe same router r can be simultaneously reconfigured by changing the configuration of r. On the contrary, admitting simultaneous changes on arbitrary sessions is less realistic, since perfect synchronism between routers must be assumed for both configuration commits and processing of BGP updates at multiple devices. Moreover, allowing simultaneous operations on different routers overcomplicates controlling there configuration, e.g., if a commit fails. Observe that SOCP does not take into account possible changes in the interdomain routing. Indeed, given an initia lconfiguration Ci= (Bi, I,_), _ is assumed not to change throughout the migration process. In the following, we show that even if eBGP is stable, there are cases in which a seamless ordering does not exist. Even worse, there are casesin which
i) every reconfiguration ordering is not oscillation free;
ii) every reconfiguration ordering is not LoV-free;
iii) every reconfiguration ordering is not deflection-free;
iv) every reconfiguration ordering is subject to unintended traffic shifts.

It is simple to extend those examples to cases in which noreconfiguration ordering is free from different kinds of anomalies, e.g., some orderings creates migration oscillationswhile others forwarding loops. In the following, we show two examples in which migration oscillations and migration loopscannot be avoided, respectively. Similar examples for the other kinds of migration anomalies can be found in [19].

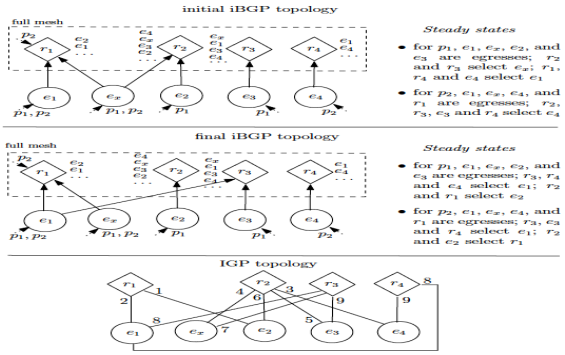



Fig. 4. TWICE-BAD gadget, an iBGP topology change case in which an oscillation-free reconfiguration ordering does not exist.

Fig. 4 depicts an example in which every reconfiguration ordering creates a permanent oscillation in an intermediate configuration.

Observe that both the initial and the final configurations are oscillation-free. Indeed, it is easy to check that the configurations are guaranteed to converge to the stable states reported in Fig. 4. In Bi, all routers but s send traffic to e0, since r1 does not receive the route announced by e1, and r2 prefers routes from e0 over those from e1. Similarly, in Bf , all routers but s select the route received from e1, since r2 does not receive the route announced by e0, and r1 prefers routes from e1 over those

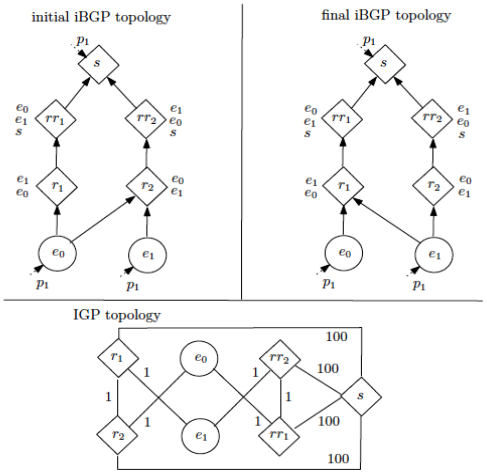



Fig. 5. PYLON gadget, an iBGP topology change case in which a loop-free reconfiguration ordering does not exist.

from e0. However, one of the following cases apply to the intermediate configuration in every reconfiguration ordering. • remove(e0, r2) before add(e1, r1): r1 and r2 are forced to select (r1 e0) and (r2 e1) respectively, hence a loop occur between r1 and r2 (see the IGP topology). • add(e1, r1) before remove(e0, r2) : because of path preferences, r1 and r2 will select (r1 e1) and (r2 e0) respectively. As a consequence, rr1 and rr2 will select (rr1 r1 e1) and (rr2 r2 e0) respectively, giving rise to a loop between rr1 and rr2 (see the IGP topology). In both cases, a migration loop occur.

B. eBGP Policy Changes

Similarly to the iBGP topology change problem, the eBGP policy change problem is stated as follows. Policy Ordering Application Problem (POAP): given the initial and the final routing policies, compute an ordering in which to apply the new policies on routers while guaranteeing a seamless migration. Basically, POAP boils down to studying how intermediate policies affect the set of routes injected in iBGP. Indeed, both the IGP and the iBGP topologies are assumed not to change during the reconfiguration, that is Ci= (B, I,_i) and Cf= (B, I,_f), with possibly _i 6= _f . In intermediate configurations, function _ can also be different from both _i and _f . Hence, our formulation of the problem encompasses both cases in which

i) the set of egress points for a given prefix changes only in the intermediate configurations; and
ii) the set of egress points for a given prefix changes also between the initial and the final configurations.
Assuming again that eBGP is stable throughout the migration, the _ function in intermediate configurations depends

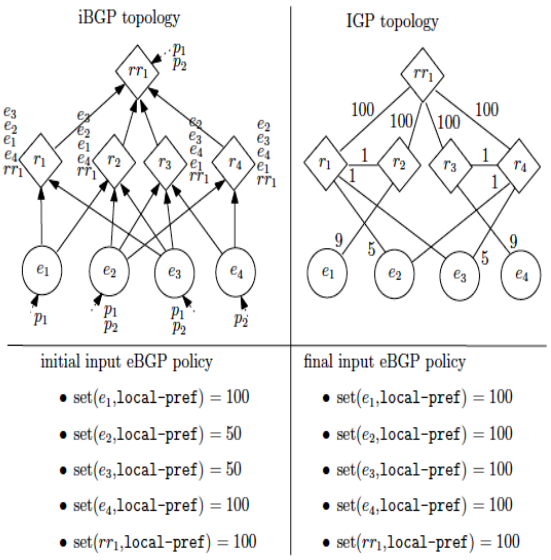



Fig. 6. CAROUSEL gadget, an eBGP policy reconfiguration case in which forwarding loops occur in every reconfiguration ordering.

only on the reconfiguration ordering. Again, migration anomalies cannot be avoided in some cases, even if both the initial and the final configurations are anomaly-free. Fig. 6 shows an example in which migration loops cannot be avoided. Consider prefix p1. In the initial configuration, e2 and e3 do not select eBGP routes, because of the local-preference settings, and e1 and rr1 are the only two egress points for p1. Hence, r1, r2, e2, and e3 select the route from e1 because of egress point preferences, while r3, r4, and e4 select the route from rr1 because it is the only route they receive. The IGP topology ensures that no deflection occurs. In the final configuration, all ei with i = 1, 2, 3 and rr1 are egress points for p1. Also, r1 and r2 select e3, and r3, r4, and e4 select e2, because of egress point preferences. Since r1 and r2 (r3 and r4, respectively) agree on the egress point to use, no deflection occurs.

Similar arguments apply to p2. However, if e2 is reconfigured before e3, then r2 starts receiving and selecting the route from e2, because of egress point preferences. On the contrary, r1 keep selecting the route from e1, as it does not receive the route from e2

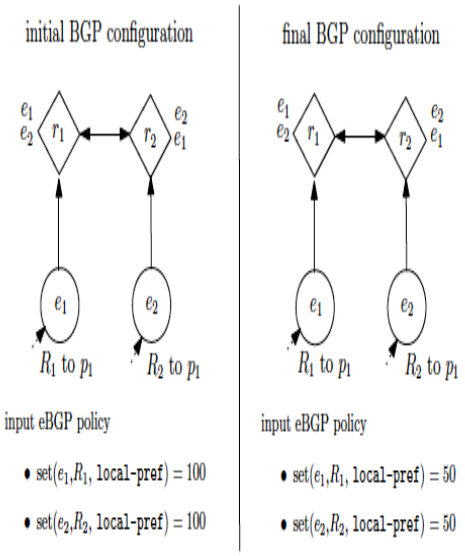



Fig. 7. FRAGILE gadget, an eBGP policy reconfiguration case in which unintended traffic shifts occur in every reconfiguration ordering.

p1is load-balanced among e1 and e2, since r1 and e1 use R1, while r2 and e2 use route R2. However, if e1 is migrated first, then all iBGP routers start preferring R2 because the route is temporarily assigned a higher local-preference with respect to R1. Hence, r1 and e1 are subject to an unnecessary traffic shift that holds until routing policy is changed on e2. A symmetrical traffic shift occurs if e2 is migrated before e1

.

## 5. A General Solution for BGP Reconfigurations

Section IV shows that seamless BGP reconfigurations cannot be always achieved by just adding and removing sessions. Intuitively, the problem is that local changes can unpredictably impact routing decisions at remote iBGP routers. We argue that additional configuration tools are needed to. To avoid data plane anomalies, our solution specify what control plane must be used network-wide for packet forwarding. We refer to this approach as BGP Ships-In-The-Night (SITN).

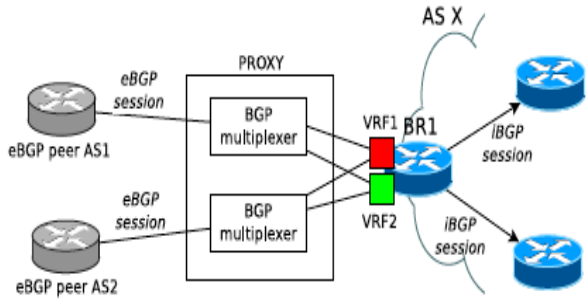
### A. Requirements and Challenges for Two Control Planes

The main advantage of BGP SITN is that it allows us to reconfigure a single router without affecting routing decisions of other routers. Indeed, running the initial and the final configurations in separate control planes enables each router to compute both the initial and the final BGP routing tables (RIBs). Then, a

router reconfiguration just mandates the router to forward traffic according to the final RIB instead of the initial one. Unfortunately, current routers cannot natively support multiple BGP routing processes on the same set of eBGP routes This prevents independent propagation of external routes to all the VRFs, since only the best routes can be leaked from one VRF to another. A workaround to propagate all the external routes to all the VRFs is to configure multiple parallel eBGP peerings. However, this solution is unpractical as it unnecessarily duplicates eBGP peerings and requires coordinated configuration changes on both sides of those peerings. Forwarding inconsistencies must also be avoided. If two routers disagree about which VRF a packet should be assigned to, the network could experience forwarding deflections, loops and congestion, hence packet loss [2].

### B. Proposed Solution

The BGP SITN approach requires three key components: a dispatching mechanism to propagate all the external routes to multiple namespaces, a front-end interface which propagates iBGP updates from one "active" namespace to the eBGP




8 Architecture of our solution.

neighbor, and a tagging mechanism, either implicit or explicit. While we can leverage multiple tagging mechanisms (MPLS and VRF-lite, for instance), we currently lack support for the other two key components. To this end, we propose to interpose a proxy component between each border router and its eBGP peers, as depicted in Fig. 8. The architecture of the proxy is similar to the one of BGP-Mux [24] in that the proxy maintains an eBGPpeering with external neighbors and one iBGP client session per VRF configured on the border router. However, we extend the architecture proposed in [24] to support the concept of "active" namespace and the selective propagation of iBGP updates to the eBGPneighbor. Indeed, the proxy distinguishes one active VRF from several passive VRFs. All VRFs receive external routes from eBGP peers, but only information in the active VRF is considered when sending eBGP updates to external neighbors. While the proxy can be implemented as a standalone device, we envision its functionality to be built directly inside border router to facilitate reconfigurations. Since the proxy maintains eBGP peerings on behalf of a border router, it needs to be configured. The proxy configuration is simple as consists in the following information.

- the address of each eBGP peer;
- for each VRF, the name of the VRF and the address of the interface on the border router which is assigned to that VRF; and
- the name of the active VRF.

Finally, to implement the tagging mechanism, the proxy exploits the third-party BGP next-hop feature that implicitly maps packets from external neighbors to the active VRF. More precisely, whenever the active VRF is changed, the proxy advertises to its eBGP peers a change of the BGP next-hop, forcing them to send data packets to the interface bound to the new active VRF. For this reason, the proxy does not need any packet forwarding ability.

## 6. Evaluation

In order to show the feasibility and effectiveness of our solution, we implemented a prototype that can perform seamless reconfigurations. We use our prototype to perform a use case, and we evaluate the scalability of our solution. Finally, we qualitatively compare our approach with alternative proposals.

### A. Implementation

The system is based on an extended version of the provisioning system presented in [2] to which we

added support for VRFs and route-maps. At each migration step, our system reconfigures one border router by interacting with the corresponding proxy and switching the active VRF on it. We implemented the proxy as a standalone script of about 400 lines in Perl. Observe that the proxy can be interposed between a border router and an eBGPneighbor without tearing down the BGP peering by taking advantage of the BGP graceful shutdown mechanism [25]. Our prototype proxy has some known limitations: first, it requires the ability to define logical interfaces on the border router; second, it requires the proxy, the external neighbour and the

border router to share the same layer 2 infrastructure. However, these limitations could be easily avoided if the proxy were directly integrated in the router operating system. Given the simple architecture of the proxy, we believe such an integration to be possible on commercial routers.

### B. Scalability

We now estimate the overhead of our approach in terms of additional router memory and CPU processing power needed to maintain two control planes. Regarding memory, we focus on the FIB size as RIBs can be easily scaled by adding low cost RAM VRFs might be a significant performance improvement(e.g., it would compress repeated BGP attributes across VRFs), we find that routers currently store a separate copy of the RIB and the FIB for each VRF. The results above suggest that our solution can be deployed in today's networks. In particular, we stress that operators providing MPLS VPN services already have most of the machinery in place to implement BGP SITN. Others should weigh the augmented network agility against the cost of introducing new technologies to configure two control planes. We believe that the long term gain in network agility can motivate operators to bear the initial deployment cost.

## 7. Conclusions

Network operators regularly change router configurations. BGP reconfigurations do not make an exception, as confirmed by our analysis of a Tier-1 ISP's historical configuration data. Since today's SLAs are stringent, reconfigurations must be performed with minimal impact on data plane traffic .In this paper, we show that routing and forwarding anomalies, possibly resulting in high packet loss ratios, can occur during BGP reconfigurations, even when MED is not used and simple policies are deployed. Unfortunately, current best practices do incur long-lasting anomalies even during common BGP reconfigurations, as we show by simulating a full-mesh to route reflection reconfiguration on a Tier-1 ISP. Hence, we study the problem of finding an operational ordering so that all intermediate configurations are anomaly free. Unfortunately, the problem of deciding whether such an ordering exists is computationally intractable. Also, we show several cases where such an ordering simply does not exist. Finally, we propose a solution that enables provably lossless BGP reconfigurations by leveraging existing technology to run multiple isolated control planes in parallel. We describe an implementation of this framework, evaluate its scalability, and illustrate its effectiveness through a case-study. Our findings show that achieving lossless BGP reconfigurations is a hard problem in the general case. However, there might exist specific reconfigurations that can be performed safely, i.e., without relying on multiple control planes. Understanding what kinds of reconfigurations can be carried out under what assumptions remains an interesting open problem.

## References

[1] Y. Rekhter, T. Li, and S. Hares, "A Border Gateway Protocol 4 (BGP-4)," RFC 4271, 2006.

[2] L. Vanbever, S. Vissicchio, C. Pelsser, P. Francois, and O. Bonaventure,"Seamless Network-Wide IGP Migrations," in Proc. SIGCOMM, 2011.

[3] B. Fortz and M. Thorup, "Optimizing OSPF/IS-IS Weights in a Changing World," IEEE Jour. on Sel. Areas in Comm., vol. 20, no. 4, pp. 756 –767, May 2002.

[4] T. Griffin and G. Wilfong, "On the correctness of ibgp configuration," inProc. SIGCOMM, 2002.

[5] S. Vissicchio, L. Cittadini, L. Vanbever, and O. Bonaventure, "i-BGP Deceptions: More Sessions, Fewer Routes," in Proc. INFOCOM, 2012.