Traffic-aware Inter-Domain Routing for Improved Internet Routing Stability

Peng Chen, Woon Hyung Cho, Zhenhai Duan, and Xin Yuan
Florida State University
{*pchen, cho, duan, xyuan*}*@cs.fsu.edu*

***Abstract*—This paper develops and studies a traffic-aware inter-domain routing (TIDR) protocol, which drastically improves the stability of the BGP-based inter-domain routing system. TIDR is designed based on two important Internet properties—the Internet access non-uniformity and the prevalence of transient failures. In TIDR, a network prefix is classified at an AS as either *significant* or *insignificant* from the viewpoint of a neighboring AS, depending on the amount of traffic exchanged between the prefix and the neighbor (including transit traffic). While BGP updates of significant prefixes are propagated with a higher priority, the propagation of updates of insignificant prefixes is aggressively slowed down. In particular, TIDR tries to localize the effect of *transient failures* on insignificant prefixes instead of propagating it onto the whole Internet. Importantly, TIDR will not create traffic black-holes due to the localization of transient failures. In this paper we present the design of TIDR and perform simulation experiments to study the performance of TIDR. Our simulation results show that TIDR can greatly improve the stability of BGP and also outperforms other existing schemes including Ghost Flushing and EPIC.**

I. Introduction

The Internet is composed of tens of thousands of network domains or Autonomous Systems (ASes), each of which is a logical collection of networks under the common administrative control [1]. ASes exchange the reachability of network prefixes via an inter-domain routing protocol. The inter-domain routing system underpins almost all the activities on the Internet, and plays a critical role in the user-perceived end-to-end network performance. When the inter-domain routing system performs poorly, we can at best achieve sub-optimal global Internet performance, regardless of how well we can tune other parts of the Internet, for example, the intra-domain routing systems [2].

The current Internet inter-domain routing system employs the Border Gateway Protocol (BGP) [3]. BGP is an adaptive routing protocol. When the current best route to a network prefix becomes unavailable due to a network failure event, BGP can converge onto a valid alternative route (if such route is available), though with a slow pace and high cost. For example, on average, it may take BGP a few minutes to converge following a single link or router failure [4]. In some extreme cases, up to 30 minutes convergence time had been reported lately [5]. During this lengthy convergence time, a large number of data packets can get lost or delayed, adversely affecting the performance of (real-time) applications such as VoIP, video streaming, and online gaming. Moreover, during the convergence of BGP, substantial update messages can be exchanged among BGP routers, which may cause an adverse impact on packet forwarding on the data plane [6] and may lead to cascaded network failures [7].

Implicit in the design of BGP is the assumption that network failure events are of the same importance to all users on the Internet—a network failure event can potentially be propagated on the global Internet using BGP. There are no explicit mechanisms to localize the effects of network failures. However, in reality, global Internet reachability does not imply the requirement of the global propagation of network failure events. In particular, an AS n may not be interested in a failure event if the communications between n and all the communicating ASes are not affected by the failure event. For example, an AS in the US may not be interested in the failure events in an Asian network, if the AS is not communicating with the network at all.

In principle, the design of BGP fails to recognize two important Internet properties concerning the use of the Internet and the nature of network failures. First, a user, or rather the AS she or he belongs to, normally only communicates with a small set of other network domains on the Internet at any given time [8], [9]. For example, in general, the top 10% of destination prefixes are responsible for more than 90% of traffic that an AS sends or receives. We refer to this property as the *Internet access non-uniformity*. From this AS' viewpoint, a failure event not affecting its communications with destination prefixes may not be relevant. Second, the majority of the network failures on the Internet are transient, which can recover within a short period of time [10], [11]. For example, a study on link failures on Sprint backbone showed that about 50% of failures recovered within 1 minutes, 80% within 10 minutes, and 90% within 20 minutes [10]. In addition, [11] showed that about 50% of BGP misconfiguration (in contrast to link or router failures) lasted less than 10 minutes. We refer to this property as the *prevalence of transient failures*. Within the short period of a transient failure, it is unlikely that an AS will dramatically change its Internet access pattern or behavior.

In this paper we develop and study a novel traffic-aware inter-domain routing (TIDR) protocol, which improves the stability of BGP by capitalizing on the two aforementioned properties: *Internet access non-uniformity* and *prevalence of transient failures*. In TIDR, (destination) network prefixes are grouped into two classes for each AS n, based on the amount of traffic exchanged between the network prefixes and the AS (including transit traffic). TIDR improves the performance of

BGP by two means. First, BGP updates of significant and insignificant prefixes are processed and propagated differently in TIDR. While BGP updates of significant prefixes are propagated with a higher priority, the propagation of updates of insignificant prefixes is aggressively slowed down. Second, the effects of transient failure events on insignificant prefixes are localized instead of being propagated onto the whole Internet. In particular, when the current best route to a prefix is replaced by a less preferred valid alternative route due to a network failure, the BGP router will *not* propagate this alternative route to the neighbors to whom the prefix is insignificant, if the corresponding failure is transient. Importantly, TIDR will not create traffic black-holes due to the localization of transient failures. By combining these two mechanisms, TIDR achieves superior performance over BGP and other existing enhancements to BGP including Ghost Flushing and EPIC [12], [13] in terms of Internet routing stability.

The remainder of the paper is organized as follows. We provide background introduction of BGP and discuss related work in Section II. We motivate the design of TIDR in Section III. We present the design of TIDR in Section IV. Simulation studies are performed in Section V to contrast the performance of TIDR with BGP and other existing schemes. We conclude the paper and discuss future work in Section VI.

II. Background and Related Work

In this section we first briefly describe a few key aspects of BGP that are relevant to this paper (see [3] for a comprehensive description). Then we discuss the related work that aims to improve the performance of BGP.

A. Border Gateway Protocol

We model the AS graph of the Internet as an *undirected* graph $G = (V, E)$. Each node $v \in V$ corresponds to an AS, and each edge $e(u, v) \in E$ represents a BGP session between two neighboring ASes $u, v \in V$. Each node owns one or multiple network prefixes. Nodes exchange BGP route updates, which may be *announcements* or *withdrawals*, to learn of changes in reachability to destination network prefixes. A route withdrawal, containing a list of network prefixes, indicates that the sender of the withdrawal message can no longer reach the prefixes. In contrast, a route announcement indicates that the sender knows of a path to a network prefix. The route announcement contains a list of *route attributes* associated with the destination network prefix. One important route attribute is as_aspath, the path vector attribute that is the sequence of ASes that this route has been propagated over. We will use $r.as\_path$ to denote the as_path attribute of route r. Let $r.as\_path = \langle v_k v_{k-1} \dots v_1 v_0 \rangle$. The route was originated (first announced) by node v_0, which owns the destination network prefix. Before arriving at node v_k, the route was carried over nodes $v_1, v_2, \dots, v_{k-1}$ in that order. For convenience, we consider a specific destination network prefix d; all BGP updates are specific to the prefix.

After learning a set of candidate routes from neighbors, a node v selects a single *best* route to reach the destination, based on some local route selection policy. Node v then propagates the best route to its proper neighbors, after prepending its own AS number to the route. BGP is an incremental protocol. Updates are generated only in response to network events. In the absence of any events, no route updates are triggered or exchanged between neighbors. When the best route at node v is withdrawn due to some network failure event by the neighbor from where the route is learned, node v will choose an alternative best route among the candidate routes and propagate the new best route to the proper neighbors. However, the alternative route may be invalid in that it has been obsoleted by the same network failure event. If no alternative route is available at node v, node v will send a withdrawal message to the neighbors to which it has announced a best route to indicate that node v has no route to reach the destination. In the following we define a few terms that we use in this paper.

Definition 1 (Valid route): A route $r.as\_path = \langle v_k v_{k-1} \dots v_1 v_0 \rangle$ is a valid route at a node v, iff the AS path $\langle v_k v_{k-1} \dots v_1 v_0 \rangle$ can be used to carry traffic from v to d.

Definition 2 (Fail-down failure event): Following a fail-down network failure event, the Internet AS graph becomes disconnected. In particular, from a node v's perspective, there is no valid route to reach destination d.

Definition 3 (Fail-over failure event): Following a fail-over network failure event, the Internet AS graph is still connected. In particular, from a node v's perspective, there is at least a valid alternative route to reach destination d.

In order to reduce the churn rate of BGP updates, a *minimum route advertisement interval* (MRAI) timer is applied to *announcement* updates to space out the messages sent to a neighbor for a given network prefix. After a node v sends an announcement to a neighbor, it has to wait an MRAI interval before sending a new announcement again. The current suggested value for MRAI is 30 seconds. The MRAI timer does not apply to withdrawal messages.

B. Related Work

Based on the same Internet access non-uniformity property, Rekhter and Chinoy proposed a Partial Reachability Injection (PRI) scheme [14], where only partial inter-domain reachability information is injected to the intra-domain routing system. However, PRI was concerned with the problem to balance the requirements on memory and processing power of intra-domain routers and the encapsulation overhead of inter-domain traffic. TIDR handles a different problem—the stability of inter-domain routing.

Ghost Flushing [12] improves the convergence of BGP by expediting the removal of outdated "ghost" information in the Internet. However, Ghost Flushing achieves the improved BGP convergence with a relatively high cost; it may double the number of update messages sent on the Internet compared with BGP. Moreover, outdated ghost information can still be chosen and propagated in Ghost Flushing.

EPIC [13] and Root Cause Notification (RCN) [15] both improves performance of BGP by carrying the root-cause

information (RCI) in the BGP updates when a network failure event occurs. Using RCI, BGP routers can eliminate all the obsoleted alternative routes and ensure that only valid alternative routes are chosen and propagated.

In [16], the authors proposed a novel differentiated BGP update processing (DUP) scheme to improve the performance of BGP. In DUP, a BGP router v sends an update message to a neighbor with a higher priority if v is on the best route from the neighbor to the destination. Otherwise, the update message is sent with a lower priority. TIDR is in line with DUP in that it also differentiates the processing of BGP updates. However, TIDR relies on the *significance* of prefixes to differentiate the processing of BGP updates. Moreover, the impacts of transient failures are also localized in TIDR.

III. Motivation and Intuition

The design of traffic-aware inter-domain routing (TIDR) is based on two important Internet properties—the Internet access non-uniformity and the prevalence of transient network failures. Intuitively (and simplified), if a network failure event is transient and an AS is not communicating with a network prefix whose reachability is affected, the failure event needs not to be propagated to the AS. In this way, TIDR can localize the effect of transient failures and can greatly improve the stability of the inter-domain routing. In this section we motivate the design of TIDR and illustrate the intuition behind the scheme. We present the detailed design of TIDR in the next section.

A. Internet Access Non-Uniformity

It has long been observed that the traffic on the Internet is distributed non-uniformly among ASes [17], [14], [9], [8], and this observation has been consistent over time. For example, in early 1970s, Kleinrock and Naylor had observed that the traffic on ARPANET was highly concentrated; the top 12.6% of site pairs were responsible for 90% of traffic observed on APRPNET [17]. Based on the measurement on the NSFNET backbone in late 1980s, Rekhter and Chinoy showed that the top 10% network prefixes are responsible for at least 85% of transit traffic. Similar trends were also observed in more recent work by Fang and Peterson [9] (Year 1999 measurement) and Rexford *et al.* [8] (Year 2002 measurement). These studies provided us with the insights into the non-uniform distribution nature of the Internet traffic among ASes or network prefixes. (Briscoe, Odlyzko, and Tilly provided a theoretical model helping to explain this broadly observed phenomenon regarding the non-uniform value of networks in [18].) Our recent study on the data traffic collected at a border router on the campus network of the Florida State University (FSU) over a 16 days period shows that the Internet access non-uniformity also holds from the viewpoint of edge networks (not shown here due to page limit).

Importantly, by correlating the collected data traffic and BGP updates over the same period of time, we observed that none of the BGP updates was related to the top 10% network prefixes in terms of the amount of traffic. That is, the majority of BGP updates sent to FSU do not have a direct effect on the delivery of the majority of traffic sent and received by FSU. Indeed, it has been observed that a large portion of BGP updates were caused by a small percentage of highly active network prefixes [19], and the reachability to popular destinations was very stable [8]. These observations are also intuitively reasonable in that Internet users are less likely to communicate with unstable destinations whose reachability constantly changes.

B. Prevalence of Transient Failures

To ensure the global reachability, a *long-term* network event such as a planned policy change or a change in AS relationship should be advertised to all the ASes on the Internet, regardless of whether or not the ASes are communicating with a prefix whose reachability is affected by the event. Otherwise, when the ASes try to communicate with the prefix, they may not have a route to reach the prefix. On the other hand, an AS may not need to be informed of a *transient failure* if the AS is not communicating with any prefixes whose reachability is affected by the event. The tuition is that, if an AS is not communicating with a prefix whose reachability is affected by a transient failure, it is unlikely that the AS will communicate with the prefix before the transient failure recovers. (TIDR can avoid traffic black-holes even if the AS changes its access pattern before the failure recovers.)

It has been observed that the majority of network failures on the Internet are transient and last for a short period of time. For example, a study on link failures in the Sprint backbone network shows that about 50% of failures recovered within 1 minutes, 80% within 10 minutes, and 90% within 20 minutes [10]. In addition, [11] shows that, about 50% of BGP misconfiguration (in contrast to link or router failures) lasted for less than 10 minutes.

Capitalizing on the aforementioned Internet access non-uniformity and the prevalence of transient failures, in the next section, we present a traffic-aware inter-domain routing (TIDR) scheme that can greatly improve the stability of the inter-domain routing. Importantly, it achieves the performance improvement while ensuring the reachability to all network prefixes, if the network is connected.

IV. Traffic-Aware Inter-Domain Routing

In designing TIDR, we wish to achieve a number of objectives. First, TIDR should improve the stability of the current BGP-based inter-domain routing system. Second, it should not create any routing black-holes after the routing system stabilizes, if a valid alternative route is available. TIDR achieves the first design objective by capitalizing on the Internet access non-uniformity and the prevalence of transient failures. Consider an arbitrary (provider) AS v and a neighboring AS n. AS v classifies all network prefixes into two classes: a "significant" prefix class, and an "insignificant" prefix class, with respect to AS n. The classes can be defined based on different criteria, which we will elaborate toward the end of this section.

A. The TIDR Algorithm

In TIDR, the processing and propagation of network prefixes in the two classes are handled differently. While updates related to significant prefixes are propagated with a high priority, updates related to insignificant prefixes are propagated with a lower priority. More importantly, when the current best route to an insignificant prefix becomes unavailable and is replaced by a less preferred route at node v, node v does not need to propagate the less preferred route to the neighbor, if the corresponding network failure event is transient. In this way, TIDR localizes the effect of transient failures and can dramatically reduce the churn rate of BGP updates on the Internet. In contrast, any changes in the reachability to a significant prefix are always propagated to the neighbor (subject to an MRAI timer) to ensure that the neighbor has the most up-to-date reachability information to the significant prefixes.

For this purpose, node v maintains two types of timers for each neighbor (the timers are prefix-specific). The first one is an MRAI timer, and the second one is a TIDR timer. The TIDR timer applies only to insignificant prefixes, while MRAI may apply to both significant and insignificant prefixes. Now we describe the TIDR timer in detail. A TIDR timer is normally associated with a much larger expiration period. Ideally, a TIDR timer should be large enough that, the majority of transient failures should recover before the timer expires. Based on the previous studies [10], [11], we set a TIDR timer to 10 minutes in our simulation studies. A BGP router applies the TIDR timer to an insignificant prefix under two conditions: it is the first node observing the failure event (adjacent to the failure), or it is the first node to have a valid alternative route to the prefix (i.e., it receives a withdrawal message and it has an alternative route). When the current best route to an *insignificant* prefix is replaced by a less preferred alternative route at node v under the above two conditions, the alternative route is not propagated to the neighbor immediately. Instead, the alternative route has to be held by the amount of time specified by the TIDR timer. The hope is that, the failure event causing this reachability change will recover before the TIDR timer expires. In this way, the neighbor needs not to be informed of the failure event, and the stability of the Internet inter-domain routing system can be greatly improved.

One challenge in this approach is that, although a neighbor n is unlikely to communicate with an insignificant prefix p, we may create potential packet forwarding black-holes if the neighbor n changes its access pattern to indeed communicate with p and the alternative route chosen by node v is invalid. To address this issue, TIDR utilizes a mechanism similar to Root Cause Information (RCI) [13], [15]. In RCI, the root cause information of a network failure is carried in the BGP updates. When node v receives a (withdrawal or announcement) update message, it can flush out all the local *invalid* alternative routes (learned from other neighbors), before choosing the next best route. In this way, the alternative route chosen (if any) can be guaranteed to be valid. If node v is the first node observing the failure (that is, node v is neighboring to the failure), the procedure of selecting and processing the alternative route is similar and we omit it here.

Algorithm 1 Traffic-Aware Inter-Domain Routing: at node v

```
cbr: Current best route to prefix p
Node v receives an update to withdraw cbr
Mark all invalid alternative based on RCI
Choose next valid best route r
if (r is empty) then
  // No valid alternative route
  Send withdrawal to proper neighbors
else
  // r is a valid alternative route
  if (p is significant for a neighbor n) then
    Send r to neighbor n subject to MRAI timer
  else
    // p is insignificant
    if (Node v is first node to have alternative route) then
      Hold r till TIDR timer expires
      // TIDR timer canceled when cbr becomes available
      // Send r when TIDR timer expires
    else
      Hold r till MRAI timer expires
    end if
  end if
end if
```

So far we have focused on the handling of BGP announcement updates. If node v does not have any (valid) alternative routes, a withdrawal message will be immediately sent to the neighbor to avoid traffic black-holing, regardless of the prefix being significant or not.

Algorithm 1 summarizes the basic protocol of TIDR.

B. Significant vs. Insignificant Prefixes

In this section we discuss how node v learns if a prefix is significant or not with respect to a neighbor. There are a few different approaches to achieve the goal with different trade-offs between the protocol complexity and the granularity of control. Ideally, each AS (or its provider) should measure its traffic access pattern and determines the significant prefixes, e.g., the prefixes responsible for 90% of traffic. Note that such traffic measurements are often available now for traffic engineering or payment purpose. Then the AS can inform the rest of the Internet the set of significant prefixes by extending the BGP protocol. However, this approach can be costly and the Internet may not be able to learn the most up-to-date set of significant prefixes from the AS. The access pattern of the AS may change over time, and the list of significant prefixes may change accordingly. However, it takes time to propagate the up-to-date significant prefix list to the rest of the Internet.

Alternatively, each AS v can measure the traffic between itself and its immediate neighbors (regardless of the final destination of the traffic), and choose the top network prefixes responsible for the majority (e.g., 90%) of the traffic. In this

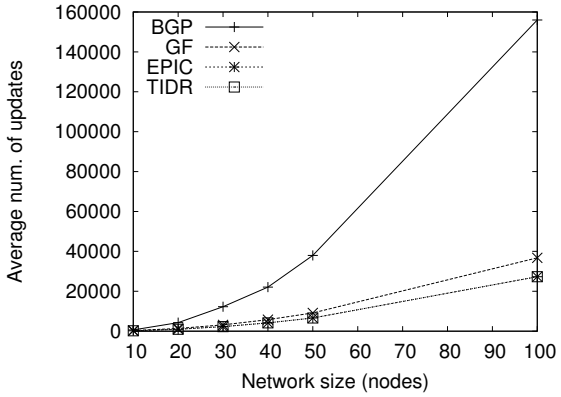

Fig. 1. Clique fail-down.

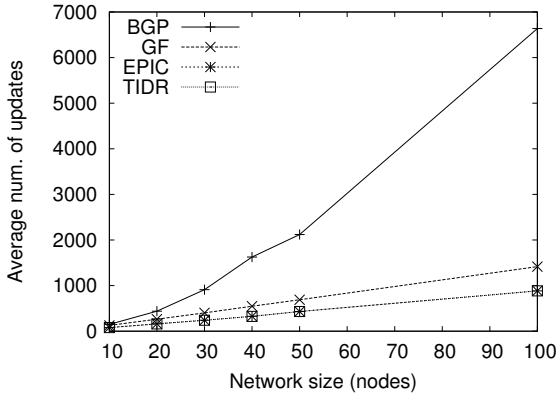

Fig. 2. Waxman fail-down.

way, no information regarding the significant prefixes needs to be propagated on the Internet. Instead, they are inferred locally. However, a drawback of this approach is that, ASes with large volume of traffic may shadow the ASes with smaller amount of traffic—smaller ASes may never get the up-to-date routing information.

A better approach is to combine the aforementioned two methods. In this approach, each AS n informs its neighbor v a traffic threshold beyond which a prefix is considered to be significant, instead of a list of significant prefixes. Node v merges the thresholds learned from its neighbors, and recursively informs the proper neighbors the new threshold. Node v can merge the thresholds in a number of different ways, for example, selecting the smallest threshold (for the same network prefix). The advantage of the method is that, the change in threshold is much less frequent than the list of the significant prefixes; therefore, the communication complexity can be greatly reduced. Second, smaller ASes will not be shadowed by larger ASes. By specifying a proper threshold, smaller ASes can also receive the most up-to-date reachability information of their significant prefixes. We envision this method will first be deployed, given its low complexity and expressiveness in specifying significant prefixes.

V. Performance Evaluation

In this section we perform simulation studies to illustrate the performance of TIDR, and contrast it with BGP, Ghost Flushing (GF) [12], and EPIC [13]. We implement TIDR in the simBGP simulator [20], which has implemented BGP, GF, and EPIC.

A. Simulation Set-Up

In the simulation studies, we used two different topology families—Clique (i.e., complete graph) and Waxman random topologies. The Waxman topologies were generated using the Brite topology generator [21] with both α and β set to 0.5. The propagation delay on each link is chosen randomly between 0.01 and 0.1 seconds. The processing delay on each node is chosen randomly between 0.001 and 0.01 seconds. For TIDR, we set MRAI timer to be 30 seconds and TIDR timer 10 minutes.

We simulate both link fail-down and fail-over events. To create a fail-down event, we attach a dummy node to a randomly chosen node in the network topology. We fail this link during the simulation. To create a fail-over event, we attach a dummy node to *two* randomly chosen nodes in the topology. We randomly fail one of the two links between the dummy node and the topology. To simplify the simulation set-up, only the dummy node announces a network prefix (all other nodes do not announce prefixes). In addition, the announced prefix is assigned with 20% probability to be significant for the rest of the nodes in the topology, and 80% probability to be insignificant. Each link failure event is chosen as a long-term event with 20% probability (with a recovery time longer than 10 minutes), and a short-term one with 80% probability (with a recovery time less than 10 minutes). We repeat the simulation 30 times, each with different attach points and random seeds.

For each simulation run, we ensure that the routing system is stable before the failure event occurs. We summarize the total number of BGP updates (including both withdrawals and announcements) sent after the failure event during the simulation. We then compute the average number of BGP updates over the 30 simulation runs.

B. Simulation Results

Figures 1 and 2 show the average number of BGP updates in fail-down events of Clique and Waxman topologies, respectively. Note first that, in a fail-down event, TIDR behaves identically with EPIC (TIDR timer is activated only if there is a valid alternative route). In addition, they outperform both BGP and GF. Figure 3 shows the average number of BGP updates in fail-over events of Clique network topologies. In this case, TIDR outperforms BGP, GF, and EPIC. In particular, compared with BGP, the average number of route updates is reduced by 71% to 92% for the Clique topologies. More importantly, as the network size increases, the relative performance improvement of TIDR over BGP (and GF/EPIC) becomes more significant. Note also that the performance of

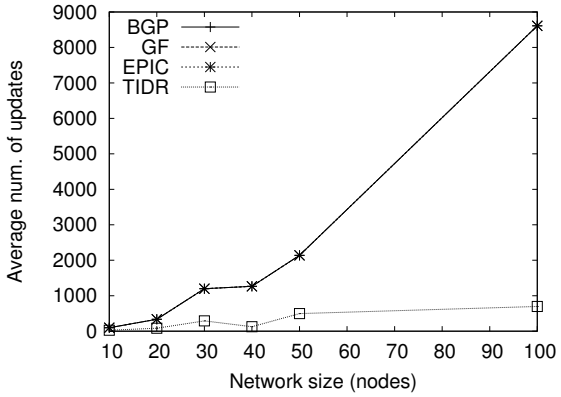

Fig. 3. Clique fail-over.

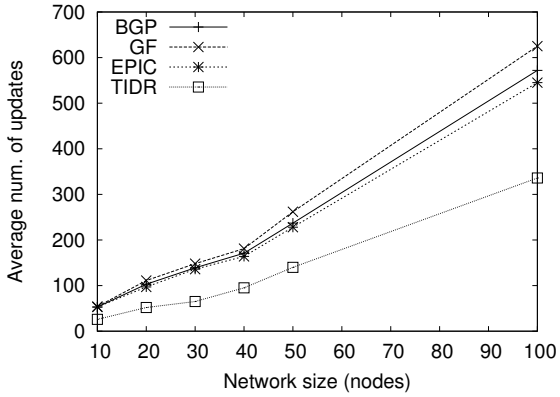

Fig. 4. Waxman fail-over.

BGP, GF and EPIC is identical in terms of the number of updates generated. Note that, in a *clique* network topology, a node will choose the *valid* alternative route to the dummy node, no matter which of the BGP, GF, and EPIC is used.

Figure 4 shows the average number of BGP updates in fail-over events of Waxman random topologies. As we can see from the figure, TIDR outperforms BGP, GF, and EPIC. In particular, the average number of route updates is reduced by 41% to 57%. Note that, EPIC only slightly improves BGP in this regard, and GF generates more updates than BGP. This is not surprising as GF may double the number of update messages sent on the Internet compared with BGP.

In summary, TIDR provides the same performance as EPIC for fail-down network events. It outperforms BGP and GF. For fail-over events, TIDR outperforms BGP, GF, and EPIC. Given the prevalence of multi-homing on the Internet, it is likely that many network failure events on the Internet will be fail-over events, which signifies the importance of TIDR in improving the Internet routing stability.

VI. Conclusion and Future Work

In this paper we developed and studied a traffic-aware inter-domain routing (TIDR) protocol, which improves the stability of the BGP-based Internet routing system. The design of TIDR capitalized on two important Internet properties—the Internet access non-uniformity and the prevalence of transient failures. In this paper we presented the design of TIDR and performed simulation studies. Our simulation studies showed that TIDR greatly improves the stability of BGP and outperforms other existing schemes including Ghost Flushing and EPIC. In this paper we only considered assigning prefix significance based on traffic volume. However, we believe the general idea to allow networks to specify the significance of prefixes is powerful and may enable new services. For example, a network may request premium routing service to a set of network prefixes. We plan to further explore this idea in our future work.

References

[1] S. Halabi and D. McPherson, *Internet Routing Architectures*, 2nd ed. Cisco Press, 2000.

[2] P. Francois and O. Bonaventure, "Avoiding transient loops during the convergence of link-state routing protocols," *to appear in IEEE/ACM Transactions on Networking*, 2007.

[3] J. Stewart, *BGP4: Inter-Domain Routing In the Internet*. Addison-Wesley, 1999.

[4] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian, "Delayed internet routing convergence," in *SIGCOMM*, 2000, pp. 175–187.

[5] Z. Mao, R. Bush, T. Griffin, and M. Roughan, "BGP beacons," in *Proc. of Internet Measurement Workshop*, Oct. 2003.

[6] S. Agarwal, C. Chuah, S. Bhattacharyya, and C. Diot, "Impact of BGP dynamics on intra-domain traffic," in *Proc. ACM SIGMETRICS*, Jun. 2004.

[7] D. Chang, R. Govindan, and J. Heidemann, "An empirical study of router response to large BGP routing table load," in *Proc. of Internet Measurement Workshop*, Nov. 2002.

[8] J. Rexford, J. Wang, Z. Xiao, and Y. Zhang, "BGP routing stability of popular destinations," in *Proc. of Internet Measurement Workshop*, Nov. 2002.

[9] W. Fang and L. Peterson, "Inter-AS traffic patterns and their implications," in *Proc. IEEE GLOBECOM*, Dec. 1999.

[10] G. Iannaccone, C. Chuah, R. Mortier, S. Bhattacharyya, and C. Diot, "Analysis of link failures in an IP backbone," in *Proc. of ACM SIGCOMM Internet Measurement Workshop*, Nov. 2002.

[11] R. Mahajan, D. Wetherall, and T. Anderson, "Understanding BGP misconfiguration," in *Proc. ACM SIGCOMM*, Pittsburgh, PA, Aug. 2002.

[12] Y. Afek, A. Bremler-Barr, and S. Schwarz, "Improved BGP convergence via ghost flushing," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 10, Dec. 2004.

[13] J. Chandrashekar, Z. Duan, Z.-L. Zhang, and J. Krasky, "Limiting path exploration in BGP," in *Proc. IEEE INFOCOM*, Miami, FL, Mar. 2005.

[14] Y. Rekhter and B. Chinoy, "Injecting inter-autonomous system routes into intra-autonomous system routing: a performance analysis," *ACM SIGCOMM Computer Communication Review*, vol. 22, no. 1, Jan. 1992.

[15] D. Pei, M. Azuma, D. Massey, and L. Zhang, "BGP-RCN: Improving BGP convergence through root cause notification," *Elsevier Computer Networks*, Jun. 2005.

[16] W. Sun, Z. M. Mao, and K. Shin, "Differentiated BGP update processing for improved routing convergence," in *Proceedings of IEEE International Conference on Network Protocols (ICNP)*, Nov. 2006.

[17] L. Kleinrock and W. Naylor, "On measured behavior of the ARPANet-work," in *AFIPS Conference Proceedings, 1974 National Computer Conference*, May 1974.

[18] B. Briscoe, A. Odlyzko, and B. Tilly, "Metcalfe's law is wrong," *IEEE Spectrum*, Jul. 2006.

[19] R. V. Oliveira, R. Izhak-Ratzin, B. Zhang, and L. Zhang, "Measurement of highly active prefixes in BGP," in *Proc. IEEE GLOBECOM*, Nov.-Dec. 2005.

[20] simBGP, "simbgp: a simple bgp simulator," http://www.bgpvista.com/simbgp.php.

[21] BRITE, "Boston university Representative Internet Topology gEnerator," http://www.cs.bu.edu/brite/.