Improving Packet Delivery Performance of BGP During Large-Scale Failures

Amit Sahoo
Dept. of Computer Science
Univ. of Califonia, Davis
Davis, CA 95616
Email:asahoo@ucdavis.edu

Krishna Kant
Intel Corporation
Hillsboro,OR 97124
Email: krishna.kant@intel.com

Prasant Mohapatra
Dept. of Computer Science
Univ. of Califonia, Davis
Davis, CA 95616
Email:pmohapatra@ucdavis.edu

***Abstract*— The border gateway protocol (BGP) is known to take a long time to converge to a steady state following the failure of BGP routers or inter-router links. This has resulted in extensive analysis of BGP convergence delay and a number of schemes have been proposed to reduce this delay. But the convergence delay is a network centric metric and the end-user relevant effects of a failure are better characterized by packet losses and end-to-end delay. In this paper we study BGP convergence from the packet delivery perspective and show that a reduction in the convergence delay does not necessarily translate into an improvement in packet delivery. Our measurements provide insights into which BGP modifications are likely to decrease packet loss, and how any shortcomings can be rectified. We then modify couple of existing techniques, and are able to reduce the packet losses.**

I. Introduction

BGP (Border Gateway Protocol) [1] is the predominant inter-domain routing protocol used in the Internet. BGP belongs to the class of *path vector* routing protocols, wherein each node advertises the "best" route for each destination to all of its neighbors. If this primary path is withdrawn or replaced by the neighbor that advertised it, BGP selects the next best route to the destination, and this route is then advertised to the neighbors. However there is no guarantee that the backup route is still valid. In case the backup route has also failed, it will be removed from the forwarding table only after it is withdrawn by the neighbor which advertised it; and another backup route is chosen. This absence of information about the validity of a route can cause BGP to go through a number of backup routes before selecting a stable one. This cycle of withdrawals/advertisements can continue for a considerable amount of time and this delay is known as the *convergence delay* (or *recovery time*).

BGP convergence delay has been studied extensively in the literature, both via real measurements and modeling [2], [3], [4], [5]. Measurements by Labovitz et al. [2] showed that the convergence delay for isolated route withdrawals can be greater than 3 minutes in 30% of the cases and could be as high as 15 minutes. On top of that, packet loss rate can increase by 30x and packet delay by 4x during recovery. Our simulation studies [6] indicate that the recovery time increases with number of failures and can be significantly larger for large scale failures. A number of proposals have also been made to improve BGP convergence delay via a variety of techniques [7], [8], [9]. One topic that has been overlooked somewhat has been the effect of large-scale failures in BGP networks. We consider a failure to be "large-scale" if it directly affects a sizeable fraction of routers in the network. Besides significantly degrading the connectivity from and to the affected Autonomous Systems (ASes), large scale failures also have a big impact on the connectivity between the source-destination pairs that use the affected ASes for transit. We carried out a study [6] to characterize large scale failures and we showed that multiple simultaneous failures can cause the convergence delay to increase significantly. Hence we feel that it is important to study the effect of large-scale failures in terms of other metrics as well.

In this paper, we consider the question of whether convergence delay is the "right" metric to examine, or would other metrics be more appropriate with regard to the user experience during routing failures. We show that some of the techniques advanced in the past to reduce convergence delays do not exhibit a consistent behavior with respect to packet delivery metrics. We propose and measure a number of new metrics to identify the reasons behind this behavior. These results provide us with insights about the kind of BGP enhancements that are likely to improve the end user experience during failures and those that are likely to make it worse. We also present a couple of new methods – derived from known techniques – that are able to reduce the packet losses. To our knowledge, this is the first comprehensive study that investigates the correlation, if any, between convergence delays and packet delivery metrics, and analyzes how packet delivery is affected by various modifications to BGP.

The organization of the rest of the paper is as follows. The previous work in this area is summarized in Section II. We briefly talk about our experimental setup and methodology in Section III. Section IV discusses the link between packet delivery and convergence delay. In Section V we list proposals for reducing BGP convergence delay and analyze their impact on packet delivery. Section VI then presents modified versions of our previously proposed Speculative Invalidation scheme [9] and the Consistency Assertions scheme [8]. We end with the conclusions in Section VII.

II. Related Work

There have been a few studies that have looked at different metrics for BGP convergence. Hao and others [10] suggested that *data plane convergence* time is a better indicator of the impact on end users than BGP protocol convergence time. They defined *data plane convergence* to be the state in which the next hops to all destinations at all nodes have stabilized. They also proposed the *average downtime* metric, which is the duration for which a node loses connectivity to the destination. Zhang and others [11] showed that decreasing the BGP convergence delay might actually increase the packet losses in some situations.

Pei and others [12] studied the packet delivery characteristics after link failures for a number of protocols, including BGP, in different types of networks. They measured a number of different metrics such as packet losses, TTL expirations, throughput and packet delay. They also observed that the packet losses were not directly proportional to the convergence delays. Our paper investigates this issue in greater detail in the context of BGP, and studies the factors that affect the packet delivery characteristics.

III. Evaluation Methodology

We used a number of synthesized topologies for our studies. A modified version of BRITE [13] was used for topology generation and BGP simulations were carried out using SSFNet [14].

A. Topology Generation

We used 120 AS topologies for our experiments. We had only one BGP router in each AS, which enabled us to identify invalid routes after a failure and helped us analyze the performance of different BGP modifications. We used topologies with a "realistic" degree distribution, derived from the actual degree distribution for inter-AS links [15]. For our 120 AS network we used the degree distribution in the range 1-40. This gave us a degree distribution which decays as a power law with an exponent of about -1.85. The average degree was about 3.67. We randomly placed all the routers on a 1000x1000 grid and the routers were linked together using a pseudo-preferential connectivity scheme in the sense that one of the ends of an edge was selected randomly but the other end was selected according to the degree of the node. For all links, we used a one way delay of 25 ms (transmission, propagation and reception delays).

Although large-scale failures could be scattered throughout the network, many failure scenarios (e.g. those caused by natural and man-made disasters) are expected to be geographically concentrated. We therefore simulated failures in a single contiguous area of the grid (around the center of the grid to avoid edge effects). We assumed that all the routers in the failed area become inoperative at the same time. We experimented with failures of different sizes. In later sections the size of the failure is represented by the fraction(in percent) of ASes/routers that are failed. In order to measure packet loss and end-to-end packet delay, we put a traffic source and sink in each AS. Each source sent UDP packets at a constant rate to all the sinks in the network. The packet loss and delay metrics are explained in greater detail in Section IV.

B. BGP Simulation

We used the SSFNet simulator for our experiments because it has been used extensively in the research community for large-scale BGP simulations. In the simulations, the *path length* (i.e., number of hops along the route) was the only criterion used for selecting the routes and there were no policy based restrictions on route advertisements. The MRAI parameter [1] limits the rate at which updates are sent between two BGP peers. In our experiments, the MRAI timer was set to the default value of 30 seconds and the MRAI timer was applied on a per-peer basis, as is commonly done in the Internet. All the timers were jittered as specified in RFC 4271 [1] resulting in a reduction of up to 25%. We did simulate processing delays for BGP updates, but the delays were much lower than the MRAI and hence can be expected to have little effect on the recovery time.

IV. BGP Performance Metrics

For a path vector routing protocol such as BGP, the obvious performance metric relates to how quickly the routes in the network converge to a stable state, after the failure/repair of links and/or nodes. While this convergence time is an important metric, it does not relate directly to the user experience. From a user's perspective, the important metrics relate to increased packet loss and packet delays during the convergence. The precise impact of loss and delay depends on the transport layer and the application. However classification of traffic based on the transport protocol or application is almost non-existent in the Internet today. As the same routing infrastructure and algorithms are likely to be used for all types of traffic, metrics specific to traffic types are unlikely to be very useful. In the next two subsections we specify the metrics that we use for packet loss and end-to-end packet delay, and look at how they correlate with the convergence delays.

A. Packet Loss Metrics

Until now we have spoken of "packet loss" metrics informally. In particular we use the following two metrics:

- L_{total}: Fraction of packets lost over the entire BGP network, during the simulation period.
- ℓ_{rate} : Packet loss rate.

The packet losses are counted at the traffic sinks. Some source-destination pairs get permanently disconnected as after a failure . We do not consider those packet losses because those are not related to BGP. We are only interested in the packet losses between those source-destination pairs that are temporarily disconnected because of the BGP convergence process.

It is certainly possible to define metrics other than those that we listed, but these should capture much of what we intuitively think of as packet losses. Furthermore these metrics can be used to calculate other more complex metrics, if the

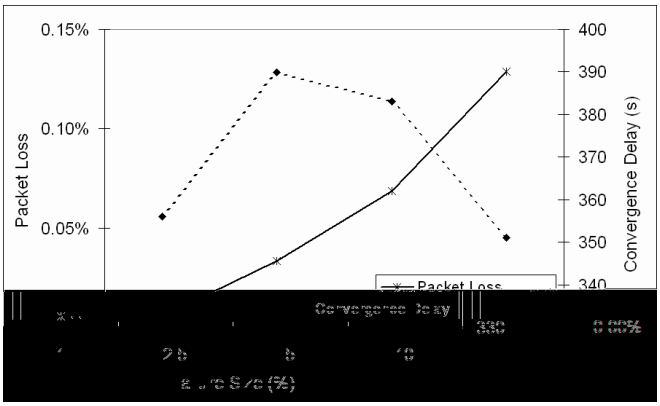

Fig. 1. Packet Loss and Convergence Delay

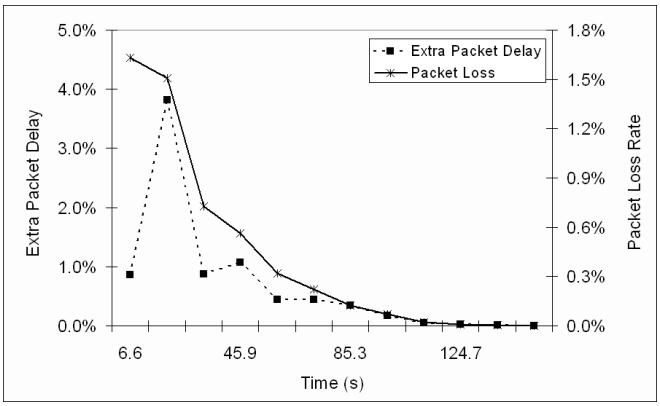

Fig. 2. Packet Delay and Packet Loss for 5% failure

need arises. The metric L_{total} is clearly the simplest and it gives us a measure of the cumulative impact of a failure. But it gives us no indication about the duration or the burstiness of the packet loss process. The second metric (ℓ_{rate}) captures the temporal variation in the packet losses. ℓ_{rate} introduces a new parameter, the observation subinterval τ to reasonably estimate the instantaneous loss rate. A smaller value of τ will give more accurate results if we increase the traffic accordingly. However that increases the simulation overhead. For our experiments, we used a τ of 13.125 seconds (half of the average jitterred MRAI).

We have plotted the total packet loss (L_{total}) and the convergence delay (T_{conv}) as a function of the failure magnitude (in terms of fraction of routers failed) in Fig. 1. We define T_{conv} as the duration needed for the routes at all BGP routers to stabilize. We can see that there is little correlation between the curves. L_{total} keeps on increasing with the size of the failure whereas T_{conv} peaks and then goes down. As it typically takes BGP longer to remove the routes to the failed destinations than to find the best valid route to an active destination, T_{conv} is effectively equal to the time needed to purge the routes to the failed destinations from the network. The packet loss on the other hand is dependent on the convergence of routes to active destinations, and hence it is not surprising that L_{total} and T_{conv} are not strongly correlated. Unfortunately most proposals designed to improve the BGP convergence process, use an improvement in the total convergence delay as proof of success. It makes more sense for us to look at the *forwarding path* convergence delay ($T_{nextHop}$), which is the time needed for the *next hops* to all active destinations to stabilize. Indeed we find that if we measure $T_{nextHop}$ independently at each AS and compute the average, then this average value is well correlated with L_{total} (Coefficient of Correlation = 0.95). Thus it might seem that average $T_{nextHop}$ can be used to estimate the effect that a BGP modification has on packet losses. However we found that the change (in comparison to normal BGP) in L_{total} and the change in average $T_{nextHop}$ was not proportional, when we use a BGP modification like Ghost Flushing. Therefore we cannot use convergence delays as predictors of packet delivery performance.

The main reason for a disconnect between $T_{nextHop}$ and packet losses is that BGP continues to forward packets regardless of whether the route has converged. As long as the router knows some next hop to reach the destination, packets will be forwarded. Although the path taken by a packet during routing convergence may be sub-optimal, the packets will definitely reach the destination if all the nodes on the traversed path have a stable and valid route to the destination. Of course routes do change a lot during convergence, and as a result a packet might reach the destination even though the source did not have a valid route to begin with. The opposite can also happen. Thus, it is impossible to estimate the packet loss from the convergence delays.

We also measured the packet loss rate (ℓ_{rate}) and we show the temporal variation for a 5% failure in Fig. 2. As can be expected, the loss rate starts off high and then gradually goes down to 0.

B. Packet Delay Metrics

In the absence of congestion, end-to-end packet delay during convergence is dependent on the length of the traversed path. However, the length of the path may actually be worsened by BGP modifications which prefer longer paths over shorter ones that are suspected to be invalid. Thus, we could have an algorithm that decreases packet loss but increases the packet delay. Packet delays can also be more difficult to analyze than packet losses because they cannot be accumulated. Therefore we have to look at the variation in the packet delay with time, and that makes it a bit unwieldy for comparisons. Hence we decided to concentrate on packet loss for comparing BGP modifications.

We did measure the packet delays however, and we show the variation in the packet delay for a 5% failure in Fig. 2. In the figure we have plotted the relative increase in the packet delay over the steady state value. This metric is measured at the traffic sink using observation subinterval τ only the delays for the currently connected source are considered. We see that the "extra packet delay" starts off low, and oscillates a couple of times before going to 0. We believe that the delays increases when routes to hitherto inaccessible destinations are received.

As the first routes that are discovered for a destination are likely to be sub-optimal or even invalid, the traversed path will also be longer than optimal, and hence packet delay goes up; eventually going down as better routes are learnt. This happens periodically because new routes are learnt after each MRAI.

V. Impact of BGP variants on Packet Delivery Behavior

In this section we analyze three proposals designed to improve BGP convergence: Ghost Flushing, Consistency Assertions and our Speculative Invalidation scheme. We selected the first two because these two BGP modifications have been cited the most in the literature. We first provide a brief overview of the proposals.

A. Convergence Delay Improvement Schemes

Ghost Flushing [7] proposes to improve BGP convergence by removing invalid routes (ghosts) quickly from the network. In normal BGP, a route advertisement might be delayed because only one route (for a particular destination) can be sent to a neighbor in one Minimum Route Advertisement Interval (MRAI). Note that the new route advertisement not only advertises a new route but also withdraws the older, possibly invalid, route. Therefore a delay in sending out a new route could cause the neighbor to possibly use an invalid route for a longer period of time. To make matters worse, the neighbor could also forward the invalid route to other nodes. Ghost Flushing solves this problem by sending out an explicit withdrawal without waiting for the MRAI timer to expire, if the new route is worse than the older route.

Consistency Assertions [8] tries to identify and remove invalid routes from the routing tables. The basic idea is that if a path advertised by one neighboring AS (A) contains another neighboring AS (B), then the paths (to the corresponding destination) advertised by both the neighbors must be consistent. If they are not, then the directly learnt route (from B) is preferred over the indirectly learnt route (from A), and the route from A is marked as "infeasible". Similarly, if the route from A contains B, but B has not advertised a route to the corresponding destination, the route is considered infeasible.

Our Speculative Invalidation [9] scheme attempts to improve BGP convergence delay by identifying ASes that are likely to have suffered complete or partial failure. All routes that contain these ASes are considered infeasible. We maintain a *failCount* for each AS, and this value is incremented if a route containing that AS is withdrawn or replaced. We consider the *failCount* to be a measure of the probability that an AS has suffered some kind of failure, and the ASes with the largest *failCount*s are considered "suspect". The details of this scheme can be found in our previous paper [9]. For the experimental results shown here, we used a time slot of 5.25 seconds (1/5th of the average jittered MRAI), a history of 5, and the scheme was executed at ASes/routers with a degree greater than 4.

B. Impact on Packet Delivery Behavior

We first look at how the schemes affect the convergence delays. In Fig. 3 we plot the improvement over normal BGP for T_{conv} and average $T_{nextHop}$. GF, CA and SI refer to Ghost Flushing, Consistency Assertions and Speculative Invalidation respectively; and we will be using these notations in the rest of the paper. All the schemes reduce T_{conv} significantly, but as discussed earlier, T_{conv} has very little to do with the packet delivery behavior. The average $T_{nextHop}$ value should be more closely related to the packet losses, and here the schemes differ quite a bit. CA reduces $T_{nextHop}$ moderately, but both GF and SI increase $T_{nextHop}$ slightly, although the shapes of the curves are different.

Now we look at the packet delivery performance. We have plotted the improvement in L_{total} over normal BGP in Fig. 4. We see that CA decreases the packet loss moderately whereas GF and SI do not have much of an effect. For GF, average $T_{nextHop}$ is still well correlated with the packet loss (Coefficient of Correlation = 0.93). However we can see in Figs. 3 and 4 that the magnitude of the change in the metrics, as compared to normal BGP, is not proportional. Furthermore, $T_{nextHop}$ values for GF and SI are noticeably different, but the packet losses are effectively the same. Thus, although we can get some indication of packet loss behavior from the $T_{nextHop}$ values, such as, CA should perform better than SI; it is not possible to make accurate predictions.

C. What characteristics affect packet delivery?

We now attempt to identify the reasons behind the packet loss behavior that we have observed in Fig. 4. In order to do that we measure some parameters during the simulation. The first parameter that we measure is something that we call "lost connectivity" ($t_{LostConn}$). At the end of the simulation, the "lost connectivity" for a "connected destination" at a router is the cumulative duration for which the router did not have a valid route to that destination. A destination is said to be "connected" if the router has a valid path to it after convergence is complete. $t_{LostConn}$ has three components: the duration for which there is no route to the destination ($t_{NoRoute}$), the duration for which all neighbors advertise an invalid route to the destination ($t_{AllInval}$), and the duration for which BGP chooses an invalid route even though valid routes are available ($t_{BgpInval}$). $t_{NoRoute}$ is different from the other two components because packet loss is 100% when there is no route to the destination. During the $t_{AllInval}$ and $t_{BgpInval}$ periods, some packets might still reach the destination even though the route at the source is invalid.

We add up $t_{NoRoute}$, $t_{AllInval}$ and $t_{BgpInval}$ for all the connected destinations at a router and then average the sum over all the routers in the network. The averaged cumulative values are denoted by $T_{NoRoute}$, $T_{AllInval}$ and $T_{BgpInval}$ respectively. The averaged cumulative $t_{LostConn}$, denoted by $T_{LostConn}$, is the sum of $T_{NoRoute}$, $T_{AllInval}$ and $T_{BgpInval}$. We measured these three metrics for all the different algorithms to get a better understanding of their performance. For normal BGP, $T_{AllInval}$ was the biggest component of

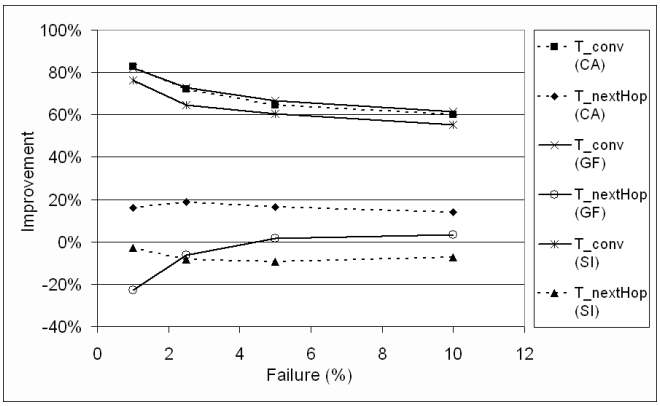

Fig. 3. Improvement in Convergence Delay

Fig. 4. Improvement in Packet Loss

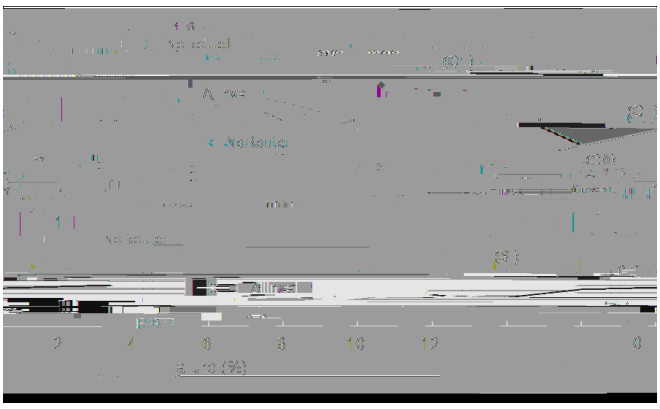

Fig. 5. Variation in $T_{NoRoute}$ and $T_{AllInval}$

Fig. 6. Packet Loss for Modified Schemes

$T_{LostConn}$, accounting for about two-thirds. $T_{BgpInval}$ made up about 20%, and $T_{NoRoute}$ accounted for the rest (around 15%).

We show the change in $T_{NoRoute}$ and $T_{AllInval}$ (in comparison to normal BGP) for the BGP modifications in Fig. 5. We see that all the schemes reduce the $T_{AllInval}$ delay, with CA performing the best and SI the worst. This reduction can be attributed to the fact that all the three schemes attempt to remove invalid routes quickly. Although we have not shown it here, GF and CA reduce $T_{BgpInval}$ moderately but SI does not. We also see that all the schemes increase $T_{NoRoute}$, with GF faring the worst. GF increases $T_{NoRoute}$ because in some cases a router sends out withdrawals immediately but sends the new routes later, and this might result in the neighbor having no route to the destination. CA and SI also increase $T_{NoRoute}$ because they mark some routes as infeasible, which can lead to a situation in which all the routes for a destination at a router are marked as infeasible. Obviously an increase in $T_{NoRoute}$ increases the packet loss, while a reduction in $T_{AllInval}$ does the opposite. These two phenomena seem to cancel each other out when we use GF or SI, resulting in little change in the packet loss. CA is the best at reducing $T_{AllInval}$ and $T_{BgpInval}$, and is in the middle of the pack as far as $T_{NoRoute}$ is concerned, resulting in a net improvement in packet loss. But once again, although we can make general observations like, CA should perform better then GF; it is difficult to make accurate predictions about the packet loss from these results. What these results do tell us are the strengths and weaknesses of the different algorithms. We use these observations to improve the schemes.

VI. Modified Schemes to Improve Packet Delivery

We saw that all the schemes suffer from an increase in $T_{NoRoute}$. It is difficult to fix this problem for GF, because it is a direct consequence of the early withdrawals that are sent out. One option could be to distinguish "early" withdrawals from normal ones, so that a router could use (but not advertise) the withdrawn route until a new route is received. However this approach is a bit complicated. On the other hand it is much simpler to fix this problem in CA. CA is successful at reducing $T_{AllInval}$, but at the same time $T_{NoRoute}$ goes up sharply and it might be because CA is too aggressive in marking routes as infeasible. We attempt to rectify this increase in $T_{NoRoute}$ by using the "best route" even if all the routes are infeasible. However we do not advertise these routes to neighbors outside the AS, because doing so would defeat the purpose of marking them as infeasible in the first place. Thus the convergence characteristics of the modified scheme should be similar to the original CA scheme.

We take a different approach for modifying SI, because SI

does not reduce $T_{AllInval}$ and $T_{BgpInval}$ significantly. As a result of that, $T_{AllInval}$ and $T_{BgpInval}$ account for nearly half of the packet losses when we use SI. This is in contrast to CA or GF, where $T_{NoRoute}$ is responsible for an overwhelming majority of the losses. The fact that $T_{AllInval}$ and $T_{BgpInval}$ are not reduced, means that there is scope to significantly improve the scheme for identifying invalid routes. An obvious cause of problems with the current scheme are possible errors in identifying the failed ASes. In order to avoid this issue, we use a different approach while keeping the core idea the same. Instead of an all or nothing scheme in which we mark a route as either *valid* or *invalid*, we measure the likelihood that a route is infeasible. For each AS we assign a *grade* (in the range 0-10), proportional to the *failCount* for that AS, and we reduce the *degree of preference* of each route by the sum of the *grades* of all ASes in that route. Thus stable routes are likely to have a higher *degree of preference* and are hence more likely to be used and advertised, leading to better packet delivery characteristics. We tested out the modified schemes and show the packet loss performance for them in Fig. 6. As we can see, the modified schemes are successful in reducing the packet losses. As expected, $T_{NoRoute}$ is also reduced for the modified schemes. However the modified SI scheme does not reduce the convergence delays significantly as it does not invalidate any routes.

VII. Conclusion

In this paper, we studied the correlation between packet loss and BGP convergence delays, and analyzed the performance of three different BGP modifications. We did not find any correlation between packet loss and the total convergence delay. The *forwarding convergence* delay ($T_{nextHop}$) was found to be well correlated with the packet loss; but when BGP modifications were used, the changes in $T_{nextHop}$ and the packet loss were not proportional. Therefore convergence delay metrics cannot be used as predictors or packet delivery performance. Among the BGP modifications, Consistency Assertions (CA) was able to reduce the packet losses moderately while Ghost Flushing (GF) and Speculative Invalidation (SI) were not.

We measured three new metrics, $T_{NoRoute}$, $T_{AllInval}$ and $T_{BgpInval}$ in order to explain the observed performance, and from these measurements we identified approaches to reduce the packet loss. We implemented these changes for CA and SI, and were able to improve the packet delivery behavior.

References

[1] Y. Rekhter, T. Li, and S. Hares, "Border Gateway Protocol 4," RFC 4271, Jan. 2006.

[2] C. Labovitz et al., "Delayed internet routing convergence," in *Proc. ACM SIGCOMM 2000*, Stockholm, Sweden, Aug. 28–Sep. 1, 2000, pp. 175–187.

[3] D. Pei, B. Zhang, et al., "An analysis of convergence delay in path vector routing protocols," *Computer Networks*, vol. 30, no. 3, Feb. 2006, pp. 398–421.

[4] T.G. Griffin and B.J. Premore, "An experimental analysis of BGP convergence time," in *Proc. ICNP 2001*, Riverside, CA, Nov. 11–14, 2001, pp. 53–61.

[5] D. Obradovic, "Real-time Model and Convergence Time of BGP," in *Proc. IEEE INFOCOM 2002*, vol. 2, New York, NY, Jun. 23–27, 2002, pp. 893–901.

[6] A. Sahoo, K. Kant, and P. Mohapatra, "Characterization of BGP recovery under Large-scale Failures," in *Proc. ICC 2006*, Istanbul, Turkey, June 11–15, 2006.

[7] A. Bremler-Barr, Y. Afek, and S. Schwarz, "Improved BGP convergence via ghost flushing," in *Proc. IEEE INFOCOM 2003*, vol. 2, San Francisco, CA, Mar. 30–Apr. 3, 2003, pp. 927–937.

[8] D. Pei, X. Zhao, et al., "Improving BGP convergence through consistency assertions," in *Proc. IEEE INFOCOM 2002*, vol. 2, New York, NY, June 23–27, 2002, pp. 902–911.

[9] A. Sahoo, K. Kant, and P. Mohapatra, "Speculative Route Invalidation to Improve BGP Convergence Delay under Large-Scale Failures," in *Proc. ICCCN 2006*, Arlington, VA, Oct. 9–11, 2006.

[10] F. Hao, S. Kamat, and P. V. Koppol, "On metrics for evaluating BGP routing convergence," Bell Laboratories Tech. Rep., 2003.

[11] B. Zhang, D. Massey, and L. Zhang, "Destination Reachability and BGP Convergence Time," in *Proc. GLOBECOM 2004*, vol. 3, Dallas, TX, Nov. 29–Dec. 3, 2004, 1383–1389.

[12] D. Pei, et al., "A study of packet delivery performance during routing convergence," in *Proc. DSN 2003*, San Francisco, CA, June 22–25, 2003, pp. 183–192.

[13] A. Medina, A. Lakhina, et al., "Brite: Universal topology generation from a user's perspective," in *Proc. MASCOTS 2001*, Cincinnati, OH, August 15–18, 2001, pp. 346-353.

[14] "SSFNet: Scalable Simulation Framework". [Online]. Available: http://www.ssfnet.org/

[15] B. Zhang, R. Liu, et al., "Measuring the Internet's vital statistics: Collecting the Internet AS-level topology," *ACM SIGCOMM Computer Communication Review*, vol. 35, issue 1, pp. 53–61, Jan. 2005.