Brief Announcement

Bringing Order To BGP: Decreasing Time and Message Complexity

Anat Bremler-Barr
Interdisciplinary Center
Herzliya, Israel
bremler@idc.ac.il

Nir Chen
Interdisciplinary Center
Herzliya, Israel
chen.nir@idc.ac.il

Jussi Kangasharju
Darmstadt University of
Technology, Germany
jussi@tk.informatik.tu-darmstadt.de

Osnat (Ossi) Mokryn
Interdisciplinary Center
Herzliya, Israel
ossi@idc.ac.il

Yuval Shavitt
Tel-Aviv University
Israel
shavitt@eng.tau.ac.il

Categories and Subject Descriptors: C.2.2: Network Protocols

General Terms: Algorithms

Keywords: Routing, BGP, convergence, path exploration

1. INTRODUCTION

The Border Gateway Protocol (BGP), the defacto routing protocol of the internet, is known to generate excessive traffic following changes in the underlying backbone topology. Specifically, BGP was shown to converge very slowly and have a worst case of $O(N!)$ message complexity after a **fail down** (detachment) of a network in [1, 2]. The authors show that BGP may explore all possible paths of increasing length, in a different version of the *counting to infinity* problem.

In this paper we investigate the behavior of BGP following an *up* (reattachment of a network) event. We find that *race conditions* may lead to a path exploration, that may not have a significant effect on the convergence time, but may increase significantly the message complexity, and hence the amount of BGP traffic. Race conditions in path exploration are an inherent phenomenon of distributed networks, where the variable link delays may cause the router to receive and send less preferred updates before receiving the more preferred update messages

Increasing the amount of BGP traffic has several implications: burdening the backbone routers; increasing the convergence time; it may falsely trigger the route flap damping mechanism; it makes it harder for service providers and researchers to understand the root cause of a given change.

To solve this problem, we here suggest a minor modification to the waiting rule of BGP that pseudo-orders the network and prevents race conditions from happening. Thus, the latency is reduced by half and the message complexity from $O(DE)$ to $O(E)$ in *up* events (where D is the Diameter of the internet and E is the number of connections between ASes). From an empirical study on raw BGP update dumps and simulation, we estimate that up to 25% of the sent messages in an *up* event, can be eliminated by our modification.

PODC'07, August 12–15, 2007, Portland, Oregon, USA.
ACM 978-1-59593-616-5/07/0008.

2. BGP OVERVIEW AND BACKGROUND

BGP is a distance and path vector routing protocol. Each router sends its preferred route to a destination (*prefix*) to its neighboring peers, along with a set of attributes. The *ASpath* attribute indicates the preferred path to that destination. Using the *ASpath* information routes with cycles are avoided. A router records for each destination the last announcement received from *each* of its peers, chooses the best one according to its set of criteria, and announces it to all of its neighbors. In the absence of a policy dictated preferences, the route with the shortest *ASpath* attribute is chosen for each destination. BGP updates are sent in the form of *announcements*, if a preferred route to a destination has changed or a *withdrawal* message to a no longer available destination [1]. To suppress the amount of messages following a topological change, the IETF is recommending the use of a fixed timer, termed *minRouteAdver*. The timer forces a fixed delay of 30 seconds between consecutive announcements to the same destination at each router.

3. PSEUDO-ORDERING BGP

The *minRouteAdver* rule has an important effect on reducing the message complexity. The message complexity without *minRouteAdver* can be exponential in n not only after *fail events* as was previously considered, but also during *up* events [3]. However, even with the use of the *minRouteAdver* timer rule, the asynchronous nature of the network may lead to race conditions inherent to distributed networks, that can cause BGP to explore less preferred paths first. Thus,

[1]A new announcement is also an implicit withdrawal to a previous route to that destination

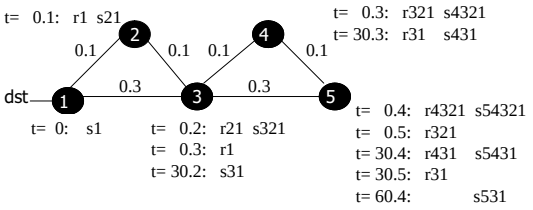

Figure 1: An example that illustrates the race conditions in BGP path exploration. The information near each router indicates the (t) time, the (r) - received message and the (s) - sent message.

the message complexity in BGP with *minRouteAver* rule is $O(DE)$ [2] as opposed to $O(E)$ [3] in BGP in synchronous network.

As an example, we consider the network in Figure 1, and describe the sequence of events following the reattachment of destination dst, where each link delay (in seconds) is marked at the links side. AS 3, for example, receives the update message from AS 2 and only then from AS 1. Hence, assuming AS 3 prefers the shortest path route, it changes its preferred path twice, each time announcing it to its neighbors: the first announcement carries ASpath 321 and the second ASpath 31. In a similar way it can be seen that AS 5 might send 3 messages, due to only one *up* event.

To eliminate these race conditions, we suggest a pseudo-ordering rule similar in nature to some weak version of a synchronizer [4], hand tailored to the BGP routing problem. The basic idea of the algorithm is that each router waits enough time before it announces its preferred route to assure that it receives the message with the shortest route. The BGP protocol already enforces a delay between the propagation of announcements due to the *minRouteAdver* fixed timer rule. Hence, our modification does not add to the overall time, but divides the delay in a more beneficial way. We assume here that routers select the shortest path route among the ones announced by their peers, hence we assume a localized shortest path selection [2]. We further validate our assumption in [3] using empirical results and show that this is a direct consequence of the common practice policy rules.

Specifically, we change the *minRouteAdver* rule to the following pseudo-ordering rule:

```
A router announces its preferred
  new ASpath of length l to its peering,
iff at least Delta seconds
  have passed from the time
  its preferred ASpath has changed.
```

We present two schemes, based on the chosen delay:

1. *Basic version (Delta= $D \cdot h$)* - The delay Delta is set to $D \cdot h$, where h is the maximal link delay between two BGP speaking routers. We prove in [3] that by following this waiting rule, BGP message complexity reduces to $O(E)$ and the convergence delay is $\sum_{i=1}^{i=D}(D \cdot h + h) = D^2h + Dh$.

[2]BGP is *not* a shortest path protocol and its routes are chosen according to policies. However, most routers usually select the shortest route among those announced by their neighbors (where some of the neighbors do not announce routes due to policy decisions [5])

	Time	Message
BGP no MinRouteAdver	Dh	$2^{n/2}$
Synchronous BGP	Dh	E
BGP with MinRouteAdver	$30D$	DE
Pseudo Ordering	$D^2h + Dh$	E
Adaptive Pseduo Ordering	$\frac{D^2h+3Dh}{2}$	E

Table 1: The Convergence Complexity (Time and Message) of *up* event where h is the bound on one hop delay, and D is the Internet Diameter, n the number of ASes and E the number of the links between ASes in the internet

2. *Adaptive version (Delta=$l \cdot h$)* - In this case the router takes into account the length of the ASpath sent in the message, l, and thus the delay Delta is set to: $l \cdot h$. We prove in [3] that the message complexity is reduced to $O(E)$ and show that the time complexity decreases by half to $\sum_{i=1}^{i=D}(i \cdot h + h) = \frac{D^2h+3Dh}{2}$.

Table 1 summarize the worst case analysis of the current status of BGP during *up* events and the results of the analysis of our algorithm.

In the paper [3] we show that our modification does not harm the convergence time. We prove by using knowledge on the Internet topological characteristic ($D \sim 12$ [6], $h = 1$ [7]) that the worst case complexity is significantly less than that of BGP, and the average case complexity is similar to the first scheme and halved by the second.

4. EXPERIMENTAL RESULTS

In the paper [3] we analyze the average case using a simulation on SSFNet [8], and show that 22% of the messages in the average case of an *up* event and around 8.5% of all the BGP messages can be eliminated using our modification. We also verify our results by investigating BGP behavior during *up* events using BGP update dumps. We find that routers in edge autonomous systems (ASes) perform more path explorations due to race conditions, and that the closer a network is to the core the less likely it is for its routers to engage in an extensive path exploration. Using the BGP dumps we estimate that 25% of the messages in the average case of an *up* event can be eliminated.

5. REFERENCES

[1] C. Labovitz, R. Wattenhofer, S. Venkatachary, and A. Ahuja, "The Impact of Internet Policy and Topology on Delayed Routing Convergence," in *Proc. INFOCOM*, April 2001.

[2] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanianitz, "Delayed Internet Routing Convergence," in *Sigcomm*, September 2000.

[3] A. Bremler-Barr, N. Chen, J. Kangasharju, O. Mokryn, and Y. Shavitt, "Bringing Order to BGP: Decreasing Time and Message Complexity," Technical report,2007, http://www1.idc.ac.il/faculty/bremler.

[4] B. Awerbuch, "Complexity of Network Synchronization," *Journal of the Association for Computing Machinery*, vol. 32, no. 4, pp. 804–823, 1985.

[5] L. Gao, "On Inferring Automonous System Relationships in the Internet," *IEEE Transactions on Networking*, 2002.

[6] A. Bremler-Barr, Y. Afek, and S. Schwartz, "Improved BGP Convergence via Ghostflushing," in *INFOCOM*, 2003.

[7] A. Feldmann, H. Kong, O. Maennel, and A. Tudor, "Measuring BGP Pass-Through Times," in *Passive and Active Measurement Workshop (PAM)*, 2004.

[8] "Ssfnet," http://www.ssfnet.org/.