Today

• Inferring BGP relationships
  – Opportunities for machine learning?
  – A candidate for the course project

• Known problems of BGP
  – Scaling issues caused by multi-homing
  – Instability
  – Delayed convergence
    • Slow failover
  – Prefix hijacking
On Inferring Autonomous System Relationships in the Internet

Lixin Gao

Abstract

The Internet consists of rapidly increasing number of hosts interconnected by constantly evolving networks of links and routers. Interdomain routing in the Internet is coordinated by the Border Gateway Protocol (BGP). BGP allows each autonomous system (AS) to choose its own administrative policy in selecting routes and propagating reachability information to others. These routing policies are constrained by the contractual commercial agreements between administrative domains. For example, an AS sets as Internet Service Providers (ISPs), companies and universities. Since two ISPs might merge into one and each administrative domain can possess several ASes, an administrative domain can operate one or several ASes. Routing within an AS is controlled by intradomain routing protocols such as static routing, OSPF, IS-IS, and RIP. A pair of ASes interconnect via dedicated links and/or public network access points, and routing between ASes is determined by the interdomain routing protocol such as Border Gateway Protocol (BGP). One key distinct feature of the interdomain routing protocol is that it allows each AS to
Background

• Route import/export follow business relationship between ASes

• Routing advertisements include AS path information

• → Use this information to infer AS-level topology
An annotated AS Graph

Figure 2: An Annotated AS Graph

- Sibling: provide transit services for each other.
  - May belong to the same company
What type of paths are legitimate?
The valley-free property

- Valley-free: After traversing a provider-to-customer or peer-to-peer edge, the AS path can not traverse a customer-to-provider or peer-to-peer edge
The basic algorithm

- Step 1: compute degree of each AS
  - The node with the largest number of neighbors is the top-level provider on an AS path
• Step 2: Assign pairwise AS relationships
  – AS pairs before the largest degree nodes are customer-providers
  – AS pairs after the largest degree node are provider-customers
Phase 2: Parse AS path to initialize consecutive AS pair’s transit relationship
1. For each as path \((u_1, u_2, \ldots, u_n)\) in \(RT\),
2. find the smallest \(j\) such that \(\text{degree}[u_j] = \max_{1 \leq i \leq n} \text{degree}[u_i]\)
3. for \(i = 1, \ldots, j - 1\),
4. transit\([u_i, u_{i+1}] = 1\)
5. for \(i = j, \ldots, n - 1\),
6. transit\([u_{i+1}, u_i] = 1\)

Phase 3: Assign relationships to AS pairs
1. For each AS path \((u_1, u_2, \ldots, u_n)\),
2. for \(i = 1, \ldots, n - 1\),
3. if transit\([u_i, u_{i+1}] = 1\) and transit\([u_{i+1}, u_i] = 1\)
4. edge\([u_i, u_{i+1}] = \text{ sibling-to-sibling}\)
5. else if transit\([u_{i+1}, u_i] = 1\)
6. edge\([u_i, u_{i+1}] = \text{ provider-to-customer}\)
7. else if transit\([u_i, u_{i+1}] = 1\)
8. edge\([u_i, u_{i+1}] = \text{ customer-to-provider}\)
Sources of errors

• Misconfigurations
  – An AS may erroneously provide transit service between providers

• Partial view
  – An AS’s degree may be larger
Taken into configuration errors

- Counts the number of times an AS provides transit service for another AS

- Assign relationships if an AS is found only to provide transit service for another AS, or the number of times an AS providing transit service for another AS exceeds a threshold
Refined Algorithm:
Input: BGP routing tables
Output: Annotated AS graph $G$

Phase 1: Same as Phase 1 in the basic algorithm

Phase 2: Count the number of routing table entries that infer an AS pair having a transit relationship
1. For each AS path $(u_1, u_2, \ldots, u_n)$,
2. find the smallest $j$ such that $\text{degree}[u_j] = \max_{1 \leq i \leq n} \text{degree}[u_i]$
3. for $i = 1, \ldots, j - 1$,
4. $\text{transit}[u_i, u_{i+1}] = \text{transit}[u_i, u_{i+1}] + 1$
5. for $i = j, \ldots, n - 1$,
6. $\text{transit}[u_{i+1}, u_i] = \text{transit}[u_{i+1}, u_i] + 1$

Phase 3: Assign relationships to AS pairs
1. For each AS path $(u_1, u_2, \ldots, u_n)$,
2. for $i = 1, \ldots, n - 1$,
3. if $(\text{transit}[u_{i+1}, u_i] > L \text{ and } \text{transit}[u_i, u_{i+1}] > L)$
   \quad or $(\text{transit}[u_i, u_{i+1}] \leq L \text{ and } \text{transit}[u_i, u_{i+1}] > 0$
   \quad and $\text{transit}[u_i, u_{i+1}] \leq L \text{ and } \text{transit}[u_i, u_{i+1}] > 0)$
4. $\text{edge}[u_i, u_{i+1}] = \text{sibling-to-sibling}$
5. else if $\text{transit}[u_{i+1}, u_i] > L \text{ or } \text{transit}[u_i, u_{i+1}] = 0$
6. $\text{edge}[u_i, u_{i+1}] = \text{provider-to-customer}$
7. else if $\text{transit}[u_i, u_{i+1}] > L \text{ or } \text{transit}[u_{i+1}, u_i] = 0$
8. $\text{edge}[u_i, u_{i+1}] = \text{customer-to-provider}$
Inferring peering relationships

- First identify all AS pairs that cannot be peers

- If the degrees between the two ASes do not differ too much, assign them to be peers
Final Algorithm:
Input: BGP routing tables
Output: Annotated AS graph $G$

Phase 1: Use either Basic or Refined algorithm to coarsely classify AS pairs into provider-customer or sibling relationships

Phase 2: Identify AS pairs that can not have a peering relationship
1. For each AS path $(u_1, u_2, \ldots, u_n)$,
2. find the AS $u_j$ such that $\text{degree}[u_j] = \max_{1 \leq i \leq n} \text{degree}[u_i]$
3. for $i = 1, \ldots, j - 2$,
   4. notpeering$[u_i, u_{i+1}] = 1$
5. for $i = j + 1, \ldots, n - 1$,
   6. notpeering$[u_i, u_{i+1}] = 1$
7. if edge$[u_{j-1}, u_j] \neq $sibling-to-sibling and edge$[u_j, u_{j+1}] \neq $ sibling-to-sibling
   8. if degree$[u_{j-1}] > $degree$[u_{j+1}]$
   9. notpeering$[u_j, u_{j+1}] = 1$
   10. else
      11. notpeering$[u_{j-1}, u_j] = 1$

Phase 3: Assign peering relationships to AS pairs
1. For each AS path $(u_1, u_2, \ldots, u_n)$,
2. for $j=1, \ldots, n-1$,
3. if notpeering$[u_j, u_{j+1}] \neq 1$ and notpeering$[u_{j+1}, u_j] \neq 1$ and $\text{degree}[u_j]/\text{degree}[u_{j+1}] < R$ and $\text{degree}[u_j]/\text{degree}[u_{j+1}] > 1/R$
4. edge$[u_j, u_{j+1}] = $ peer-to-peer
Evaluation

• Verified using AT&T internal information
Comments

• Missing edges may affect the results

• Heuristics may or may not hold

• Interesting to conduct such a study again and compare results
Today

• Inferring BGP relationships
  – Opportunities for machine learning?
  – A candidate for the course project

• Known problems of BGP
  – Scaling issues caused by multi-homing
  – Instability
  – Delayed convergence
    • Slow failover
  – Prefix hijacking
Multi-homing

• Connect to multiple providers
  – Goal: Higher availability, more capacity

• Problems:
  – Provider-based addressing breaks
  – Everyone needs their own address space
Multi-homing increases routing table size

<table>
<thead>
<tr>
<th></th>
<th>ISP1</th>
<th>ISP2</th>
</tr>
</thead>
<tbody>
<tr>
<td>204.0.0.0/8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>204.1.0.0/16</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

You can reach 128.0.0.0/8 And 204.1.0.0/16 via ISP1

ISP1

128.0.0.0/8

204.1.0.0/16

Mutil-home.com

ISP2

204.0.0.0/8

ISP3

128.0.0.0/8

204.1.0.0/16

ISP1
Global routing tables continue to grow

Source: http://bgp.potaroo.net/as6447/
Internet Routing Instability
Craig Labovitz, Student Member, IEEE, G. Robert Malan, Student Member, IEEE, and Farnam Jahanian, Member, IEEE

Abstract—This paper examines the network interdomain routing information exchanged between backbone service providers at the major U.S. public Internet exchange points. Internet routing instability, or the rapid fluctuation of network reachability information, is an important problem currently facing the Internet engineering community. High levels of network instability can lead to packet loss, increased network latency and time to convergence. At the extreme, high levels of routing instability have led to the loss of internal connectivity in wide-area, national networks. In this paper, we describe several unexpected trends in routing instability, and examine a number of anomalies and pathologies observed in the exchange of inter-domain routing information. The analysis in this paper is based on data collected from BGP routing messages generated by border routers at five of the Internet core’s public exchange points during a nine month period. We show that the volume of these routing updates is several orders of magnitude more than expected and that the majority of this routing information is redundant, or pathological. Furthermore, our analysis reveals several unexpected trends and ill-behaved systematic properties in Internet routing. We finally posit a number of explanations for these anomalies and evaluate their potential impact on the Internet infrastructure.

flaps have led to the transient loss of connectivity for large portions of the Internet. Overall, instability has three primary effects: increased packet loss, delays in the time for network convergence, and additional resource overhead (memory, CPU, etc.) within the Internet infrastructure.

The Internet is comprised of a large number of interconnected regional and national backbones. The large public exchange points are often considered the “core” of the Internet, where backbone service providers peer, or exchange traffic and routing information with one another. Backbone service providers participating in the Internet core must maintain a complete map, or default-free routing table, of all globally visible network-layer addresses reachable throughout the Internet.

The Internet is divided into a large number of different regions of administrative control commonly called autonomous systems. These autonomous systems (AS’s) usually have distinct routing policies and connect to one or more remote AS’s at private or public exchange points. AS’s are traditionally

• Goals: how often BGP sends updates to change routes

• Methodology:
  – Analyzing BGP logs for a long time
Background

• Another ACM SIGCOMM test of time award paper
• A first large scale study of BGP traffic
• Motivated much improvement to BGP
Terms

• WADiff: withdrawal → announcement
• AADiff: announcement → announcement
• WADup: same route withdrawal → announcement
• AADup: same route announcement → announcement
• WWDup: same route withdrawal → withdrawal
Observed pathologies

• Repeated WWDup, WADup, AADup
• Why are they pathologies?
<table>
<thead>
<tr>
<th>Network</th>
<th>Announce</th>
<th>Withdraw</th>
<th>Unique</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provider A</td>
<td>1127</td>
<td>23276</td>
<td>4344</td>
</tr>
<tr>
<td>Provider B</td>
<td>0</td>
<td>36776</td>
<td>8424</td>
</tr>
<tr>
<td>Provider C</td>
<td>32</td>
<td>10</td>
<td>12</td>
</tr>
<tr>
<td>Provider D</td>
<td>63</td>
<td>171</td>
<td>28</td>
</tr>
<tr>
<td>Provider E</td>
<td>1350</td>
<td>1351</td>
<td>8</td>
</tr>
<tr>
<td>Provider F</td>
<td>11</td>
<td>86417</td>
<td>12435</td>
</tr>
<tr>
<td>Provider G</td>
<td>2</td>
<td>61780</td>
<td>10659</td>
</tr>
<tr>
<td>Provider H</td>
<td>21197</td>
<td>77931</td>
<td>14030</td>
</tr>
<tr>
<td>Provider I</td>
<td>259</td>
<td>2479023</td>
<td>14112</td>
</tr>
<tr>
<td>Provider J</td>
<td>2335</td>
<td>1363</td>
<td>853</td>
</tr>
</tbody>
</table>

- Majority of BGP updates are WWDup
- WWDup belong to ASes that never announce them
Why?

• Many reasons. Not fully understood
  – Stateless BGP, does not remember what have sent to peers
  – Send withdrawals to all peers
  – Physical link errors
  – Unjittered timers causing synchronization
  – IGP, BGP interactions
  – Conflicting routing policies
Data analysis techniques

• Time series analysis
• Frequency analysis
  – Fast Fourier transform
  – Maximum entropy spectral estimation
• Different estimation methods, but both find significant frequencies at seven days, and 24 hours
Main results

• Much more updates than expected
  – 99% is pathological. Impressive!
  – A taxonomy to analyze pathologies

• Root causes unknown
  – Worth a new study

• Motivated much follow-up work
Delayed Internet Routing Convergence

Craig Labovitz, Member, IEEE, Abha Ahuja, Member, IEEE, Abhijit Bose, and Farnam Jahanian

Abstract—This paper examines the latency in Internet path failure, failover, and repair due to the convergence properties of interdomain routing. Unlike circuit-switched paths which exhibit failover on the order of milliseconds, our experimental measurements show that interdomain routers in the packet-switched Internet may take tens of minutes to reach a consistent view of the network topology after a fault. These delays stem from temporary routing table fluctuations formed during the operation of the Border Gateway Protocol (BGP) path selection process on Internet backbone routers. During these periods of delayed convergence, we show that end-to-end Internet paths will experience intermittent loss of connectivity, as well as increased packet loss and latency. We present a two-year study of Internet routing convergence through the experimental instrumentation of key portions of the Internet infrastructure, including both passive data collection and fault-injection machines at major Internet exchange points. For example, transient disruptions in backbone networks that previously impacted a handful of scientists may now cause enormous financial loss and disrupt hundreds of thousands of end users.

Since its commercial inception in 1995, the Internet has lagged behind the public switched telephone network (PSTN) in availability, reliability, and quality of service (QoS). Factors contributing to these differences between the commercial Internet infrastructure and the PSTN have been discussed in various literature [18], [26]. Although recent advances in the IETF’s Differentiated Services working group promise to improve the performance of application-level services within some networks, across the wide-area Internet these QoS
Delayed Internet Convergence

- Methodologies
Experiments setup

- Actively inject BGP faults
  - How is fault injected?

- Passively listen at peering sessions, and use NTP synchronized machines to calculate the convergence time

- Actively send probe packets to observe end-to-end packet loss and latency

- Much BGP work later uses similar measurement techniques
Types of routing events

• Tup: A previously unavailable route is announced as available.

• Tdown: A previously available route is withdrawn.

• Tshort: An active route with a long ASPath is implicitly replaced with a new route possessing a shorter ASPath.

• Tlong: An active route with a short ASPath is implicitly replaced with a new route possessing a longer ASPath.
Results show delayed convergence

• Bad news travels slow.
Slow routing convergence results in poor end-to-end performance

(a) Loss

(b) Latency
What causes the delayed routing convergence?

- A simple BGP convergence model reveals that in the worse case, all possible paths are explored before a prefix is withdrawn.
- No minimum advertisement timer: synchronized network, global message queue
Failover

- BGP is designed for scaling more than fast failover
  - Many mechanisms favor this balance
  - Route flap damping, for example.
    - If excess routing changes (“flapping”), ignore for some time.
    - Has unexpected effects on convergence times.
  - Route advertisement/withdrawal timers in the 30 second range
    - Effect: tens of seconds to many minutes to recover from “simple” failures.
  - 15-30 minute outages not uncommon.
BGP prefix hijacking attacks

• Security: There is almost none!
  – Some providers filter what announcements their customers can make. Not all do.
How can it happen?

- ISP
  - 152.3.0.0/16
  - 152.3.137.212

- Duke
  - 152.3.0.0/16

- UNC
  - 152.3.0.0/16
AS 7007 Incident

- Was a major disruption on April 25, 1997
- AS 7007 leaked part of its routing table to the entire Internet
- Routes were deaggregated /24

https://en.wikipedia.org/wiki/AS_7007_incident
• Pakistan aimed to censor YouTube

• AS 17557 (Pakistan Telecom) advertised a more specific YouTube prefix 208.65.153.0/24 to its neighbor AS 3491 (PCCW)
  – YouTube 208.65.153.0/22

• PCCW advertised to the rest of the Internet

• YouTube advertised more specific prefixes to regain control
Hard to Trace Spam

• If a source IP sends spam, it may get on a blacklist

• How to evade blacklisting?
  – Hijack a prefix (typically an unassigned one)
  – Send spam from that prefix
  – Withdraw
Summary

• Infer AS relationships

• Some known BGP problems
  – Scaling
  – Pathological announcements
  – Delayed convergence
  – Prefix hijacking

• Re-run the experiments for your course projects!
  – Observe the IPv6 Internet