IP Routing

CS 552

Richard Martin

(with slides from S. Savage and S. Agarwal)

Outline

- Position paper
- Background on Internet Connectivity
  - Nor01 paper
- Background on BGP
- BGP convergence
- BGP and traffic
- Discussion

Position Paper

- Goals:
  - Practice writing to convince others
  - Research an interesting topic related to networking.
  - Generate reactions amongst fellow classmates/professors
- Size of the paper:
  - 5000-6000 words, 6-8 pages, 10 pt. Font
- Must have title and abstract
- Name on the paper is optional
- You will evaluate 2 papers
- You will revise your paper
- Hand in in PDF
- First draft due Oct 24th

Evaluation Criteria

- Is the position well defined?
  - Is it narrow enough to be manageable?
  - Are the communities of people involved with the position (and their positions) identified?
- Are the opposing positions articulated?
- Are rebuttals given to the opposing positions?
- What evidence is used to support the position?
  - Quantitative evidence based on experimentation?
  - General facts about the systems in question?
  - Anecdotes only?
- Is the paper logically organized?
- Most importantly, does you paper influence someone towards the position?

Position Topics I

- Peer to Peer technologies equals pirating.
  - (suggested by Thu Nguyen)
- SANs vs. LANs.
- Distributed hash tables (DHTs): What are they good for?
- Ipv4 is sufficient for the next 30 years.
- IP over direct links.

Position Topics 2

- Over-provisioning vs. QoS.
  - (Suggested by Badri Nath).
- Multicast vs. P2P for content distribution.
- Mobile IP is dead.
- Wireless Ad-hoc networks.
- Information will be free.
- Privacy will die soon (or is dead already)
- Bottom up standards are better.
- VANETs vs. Infostations vs. 3G/4G
- Others Possible (e.g. security)
  - Must convince the instructor position is worthy.

Academic Integrity

- DO: think about the position
  - Helps if you pick a position you care about (at least a little bit)
- DO: write your own text
- DO: Original research and properly cite sources at points embedded in the text.
- DON'T rip/off copy text
  - Longer quotes (100-200 words) ok, IF properly cited.
- Use papers and samples as models.

Review

- Basic routing protocols
  - Distance Vector (DV)
    - Exchange routing vector hop-by-hop
    - Pick routes based on neighbor's vectors
  - Link State (LS)
    - Nodes build complete graph and compute routes based on flooded connectivity information

Historical Context

- Original ARPA network had a dynamic DV scheme
  - replaced with static metric LS algorithm
- New networks came on the scene
  - NSFnet, CSnet, DDN, etc…
    - With their own routing protocols (RIP, Hello, ISIS)
    - And their own rules (e.g. NSF AUP)
- Problem:
  - how to deal with routing heterogeneity?

Inter-network issues

- Basic routing algorithms do not handle:
- Differences in routing metric
  - Hop count, delay, capacity?
- Routing Policies based on non-technical issues
  - E.g. Peering and transit agreements not always align with routing efficiency.

Internet Solution

- Autonomous System (AS)
  - Unit of abstraction in interdomain routing
  - A network with common administrative control
  - Presents a consistent external view of a fully connected network
  - Represented by a 16-bit number
    - Example: UUnet (701), Sprint (1239), Rutgers (46)
- Use an external gateway protocol between AS
  - Internet's is currently the Border Gateway Protocol, version 4 (BGP-4)
- Run local routing protocol within an AS, EGPs between the AS

BGP: Path Vector

- Link State
  - Too much state
  - Currently 11,000 ASs and > 100,000 networks
  - Relies on global metric & policy
- Distance vector?
  - May not converge; loops
  - Solution: path vector
  - Reachability protocol, no metrics
- Route advertisements carry list of ASs
  - E.g. router R can reach 128.95/16 through path: AS73, AS703, AS1

Summary

Link State

- Topology information is flooded within the routing domain
- Best end-to-end paths are computed locally at each router.
- **Best end-to-end paths determine next-hops.**
- Based on minimizing some notion of distance
- **Works only if policy is shared and uniform**
- Examples: OSPF, IS-IS

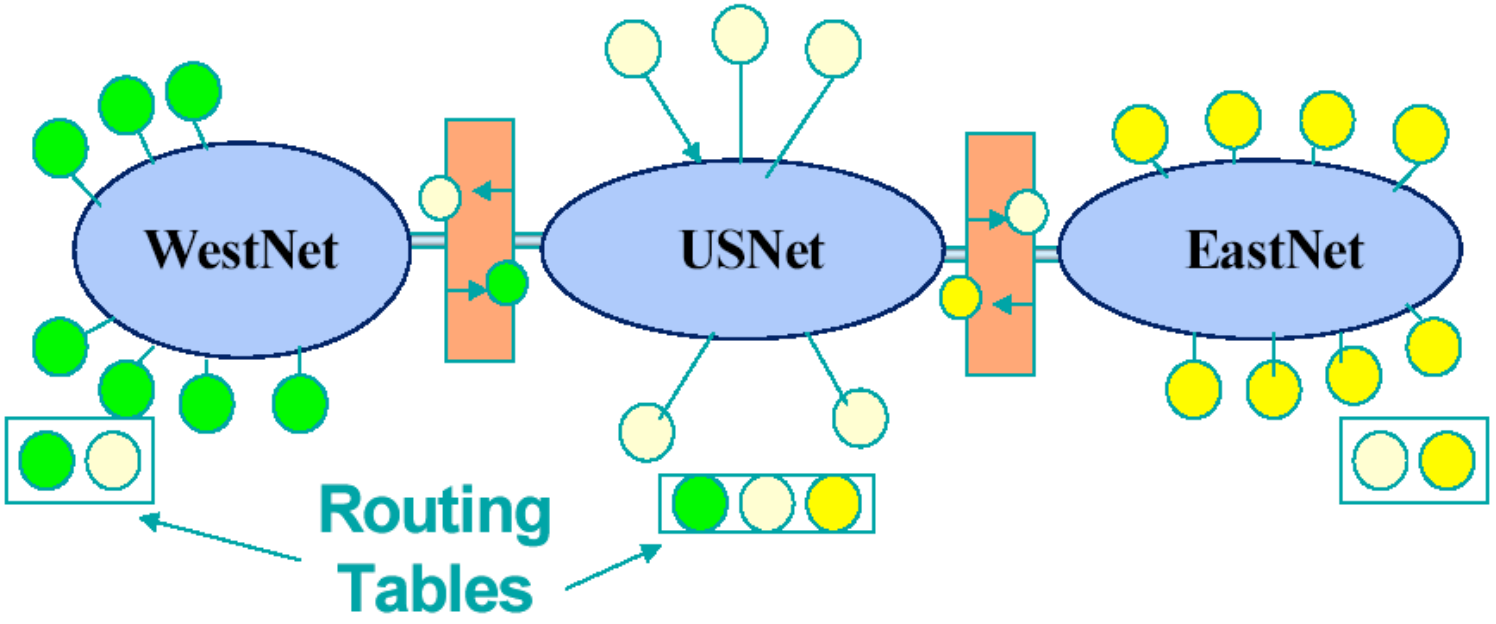
Vectoring

- Each router knows little about network topology
- Only best next-hops are chosen by each router for each destination network.
- **Best end-to-end paths result from composition of all next-hop choices**
- Does not require any notion of distance
- **Does not require uniform policies at all routers**
- Examples: RIP, BGP

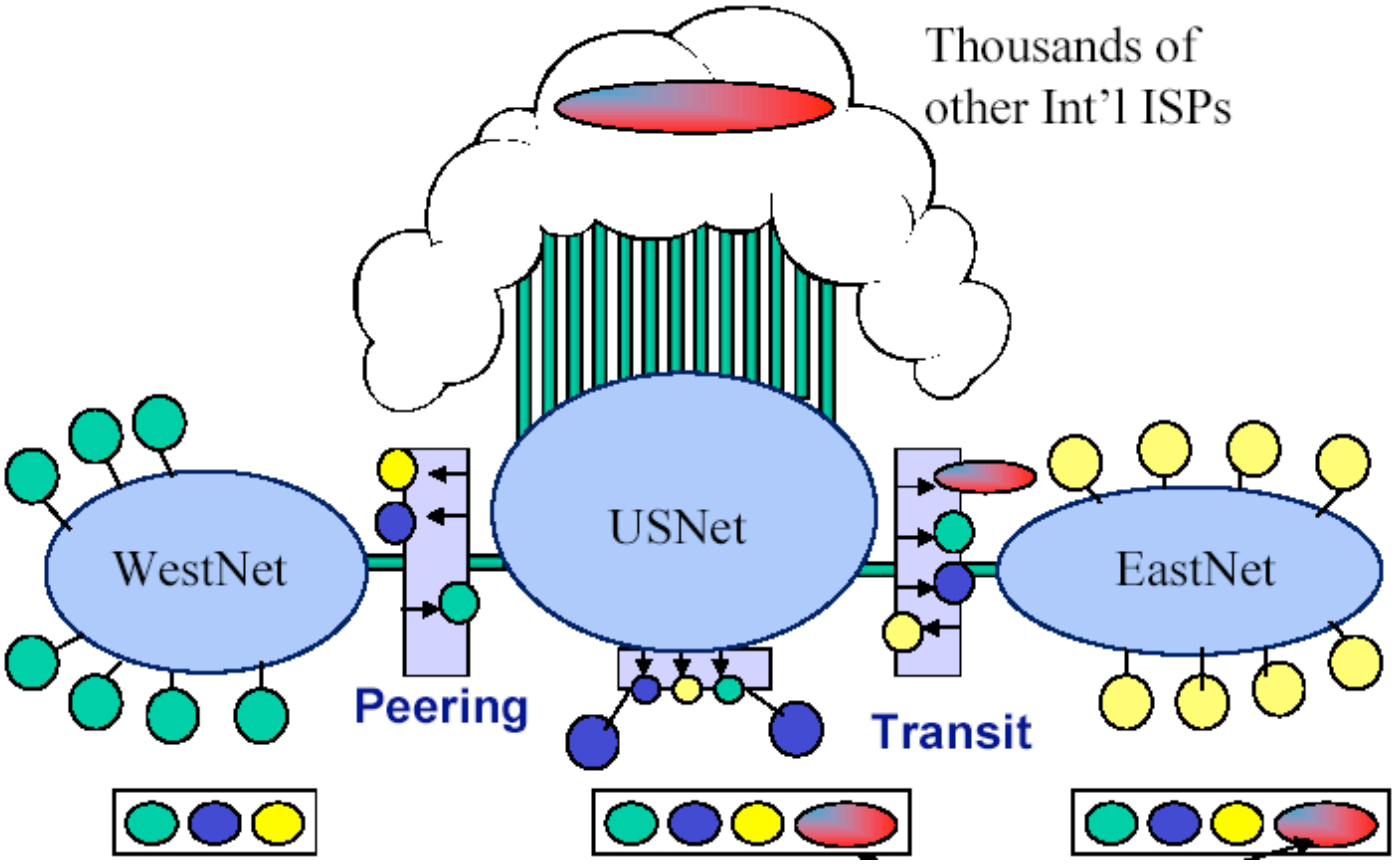
Peering and Transit

- Peering
  - Two ISPs provide connectivity to each others customers (traditionally for free)
  - Non-transitive relationship
- Transit
  - One ISP provides connectivity to every place it knows about (usually for money)

Peering



Transit


Thousands of
other Int'l ISPs
WestNet
USNet
EastNet
Peering
Transit

Exchanges and Point of Presence

- Exchange idea:
  - Amortize cost of links between ISPs
- ISP's buy link to 1 location
  - Exchange data/routing at that location
- 1 Big link at exchange point cheaper than N smaller links

Peering and Transit

- Peering and Transit are points on a continuum
  - Some places sell “partial transit”
  - Other places sell “usage-based” peering

Issues are:

- Which routes do you give away and which do you sell?
- To whom? Under what conditions?

Interconnect Economics

From: Market Structure in the Network Age

by Hal Varian

http://www.sims.berkeley.edu/~hal/Papers/doc/doc.html

Metcalf's Law

How much is a network worth?

- Approximation: 1 unit for each person a person can communicate with
  - The more people I can talk to, the more I value the network.
- N people in the network $\Rightarrow$
  - network is worth N^2 "units"
- Network value scales as N^2, (not N) is called Metcalf's law

Implications for Peering

- Simple model of network value implies peering should often happen
- What is the increase in value to each party's network if they peer?
  - Want to compute change in value, ΔV
  - Take larger network value and subtract old
- $\Delta V_1 = N_1(N_1+N_2) - (N_1)^2 = N_1 N_2$
- $\Delta V_2 = N_2(N_1+N_2) - (N_2)^2 = N_1 N_2$

Symmetric increase in value

- Simple model shows net increase in value for both parties
- Both network's values increase is equal!
  - Smaller network: a few people get a lot of value
  - Larger network: a lot get a small value.
- Helps explain "symmetric" nature of most peering relationships, even between networks of different sizes

Takeovers

- Instead of peering, what if the larger network acquires the smaller one?
  - suppose it pays the value for the network too
- $\Delta V = (N_1+N_2)^2 - (N_1)^2 - (N_2)^2 = 2(N_1 N_2)$
  - Captures twice as much value by acquisition as peering
- An incentive to not peer
  - E.g. to force a sale or merger, allowing larger network to capture a greater value than by peering

Reasons not to Peer

- Asymmetric Traffic
  - More traffic goes one way than the other
  - Peer who carries more traffic feels cheated
  - Hassle
- Top tier (big) ISPs have no interest in helping lower tier ISPs compete
  - The “Big Boys” all peer with each other at no/little cost
- Harder to deal with problems without strong financial incentive

A lower tier strategy

- Buy transit from big provider
- Peer at public exchange points to reduce transit cost
- Establish private point-to-point peering with key ISPs
- When you're big enough, negotiate peering with transit provider

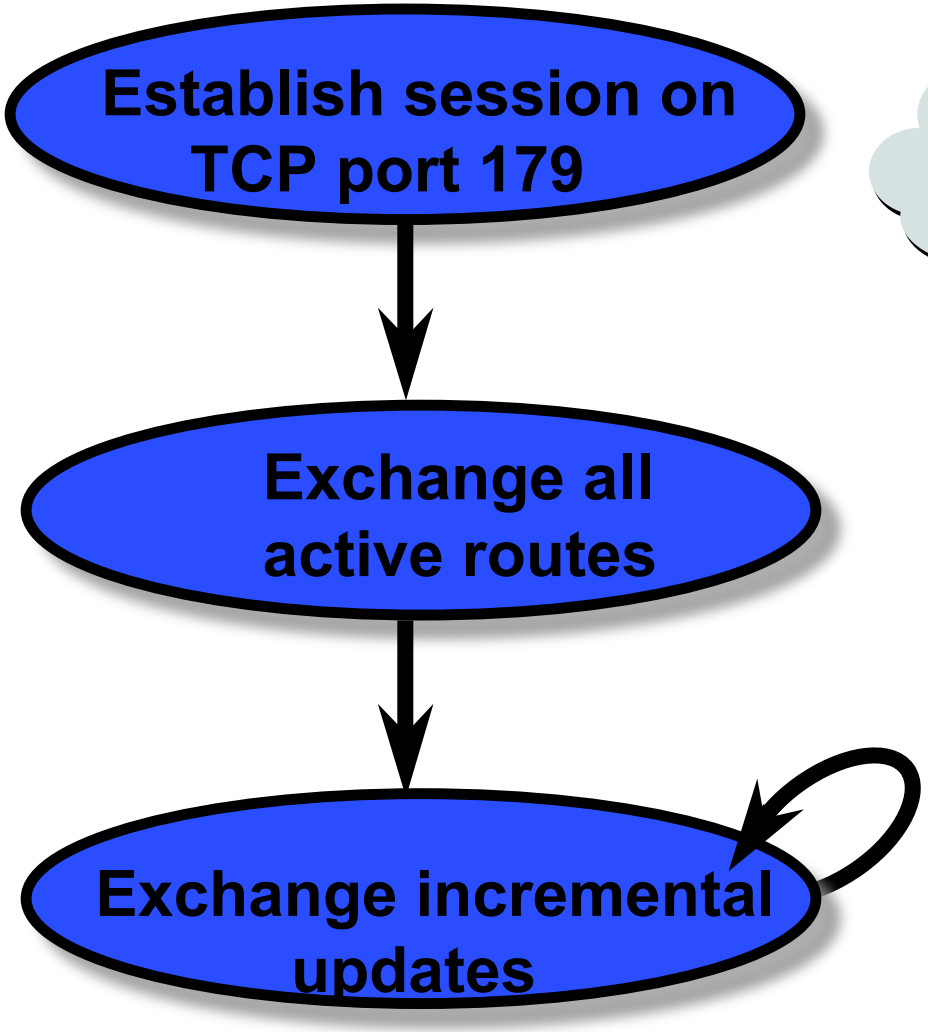
BGP and Traffic

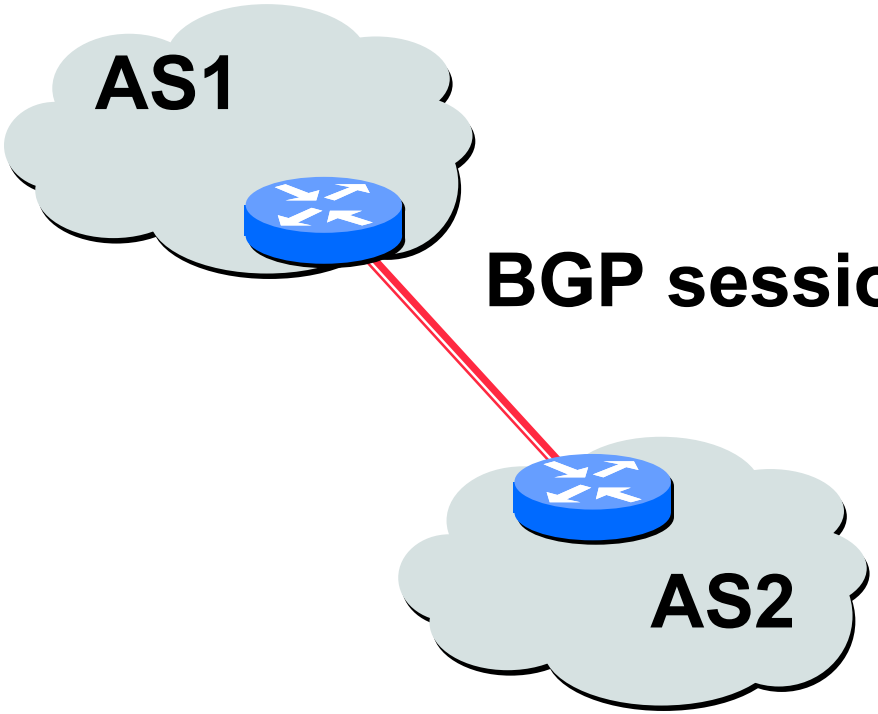
- Network engineering
  - Estimate traffic matrix
  - Tune network for performance
- Stability assumptions for estimation, tuning
- Reality:
  - Inter-domain connectivity grown rapidly
  - Large # of BGP entries, changes
  - Can result in unstable Traffic Matrix
  - Can be bad for performance

Important BGP attributes

- LocalPREF
  - Local preference policy to choose “most” preferred route
- Multi-exit Discriminator
  - Which peering point to choose?
- Import Rules
  - What route advertisements do I accept?
- Export Rules
  - Which routes do I forward to whom?

BGP Operations (Simplified)





While connection is ALIVE exchange route UPDATE messages

Four Types of BGP Messages

- **Open** : Establish a peering session.
- **Keep Alive** : Handshake at regular intervals.
- **Notification** : Shuts down a peering session.
- **Update** : Announcing new routes or withdrawing previously announced routes.

announcement
=
prefix + attributes values

BGP Attributes

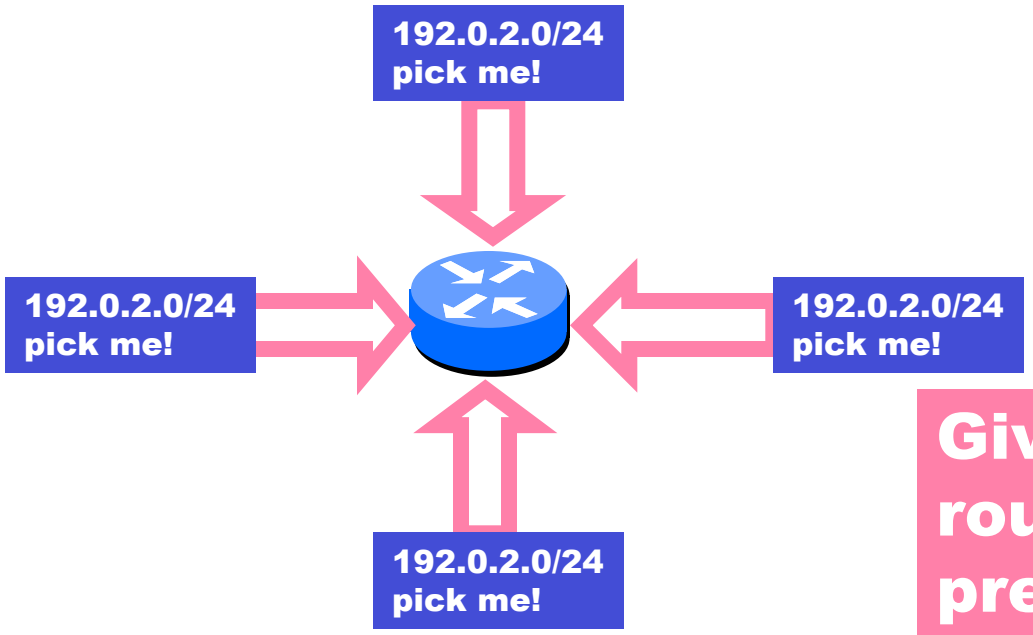
Value	Code	Reference
1	ORIGIN	[RFC1771]
2	AS_PATH	[RFC1771]
3	NEXT_HOP	[RFC1771]
4	MULTI_EXIT_DISC	[RFC1771]
5	LOCAL_PREF	[RFC1771]
6	ATOMIC_AGGREGATE	[RFC1771]
7	AGGREGATOR	[RFC1771]
8	COMMUNITY	[RFC1997]
9	ORIGINATOR_ID	[RFC2796]
10	CLUSTER_LIST	[RFC2796]
11	DPA	[Chen]
12	ADVERTISER	[RFC1863]
13	RCID_PATH / CLUSTER_ID	[RFC1863]
14	MP_REACH_NLRI	[RFC2283]
15	MP_UNREACH_NLRI	[RFC2283]
16	EXTENDED COMMUNITIES	[Rosen]
...		
255	reserved for development	

From IANA: http://www.iana.org/assignments/bgp-parameters

Most important attributes

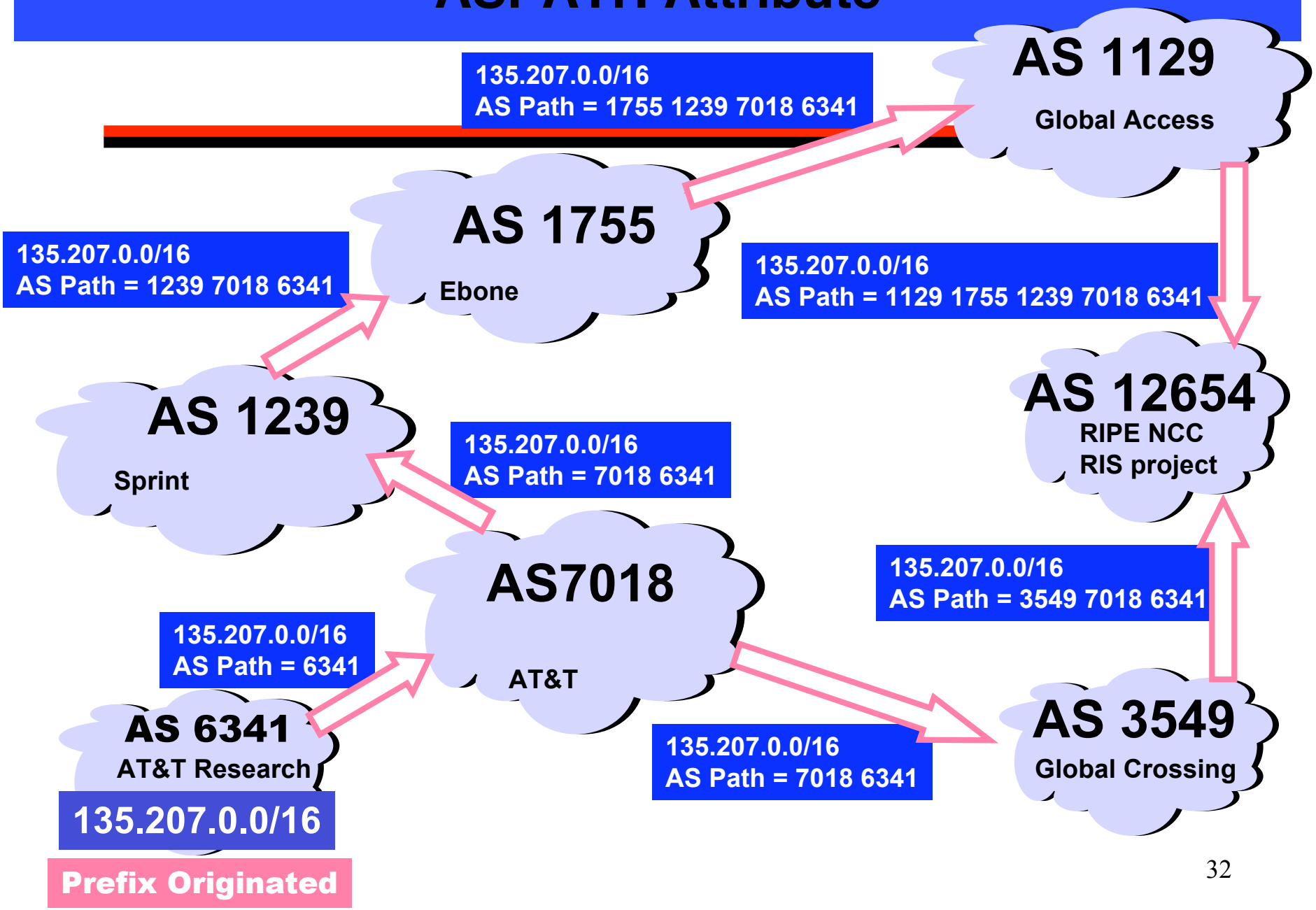
Not all attributes need to be present in every announcement

Attributes are Used to Select Best Routes



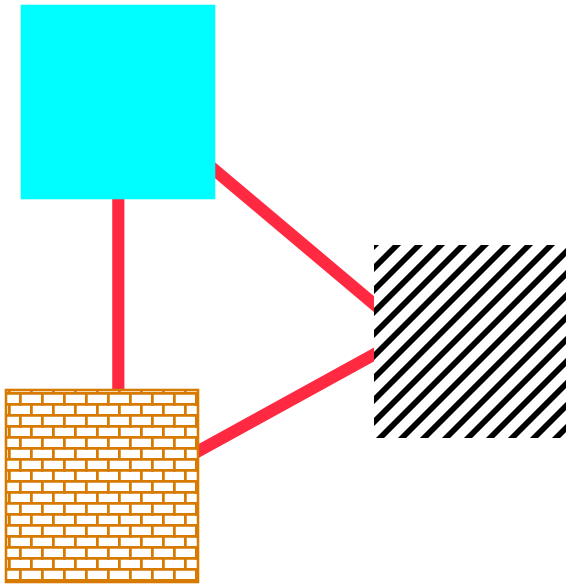
Given multiple routes to the same prefix, a BGP speaker must pick at most one best route (Note: it could reject them all!)

ASPATH Attribute

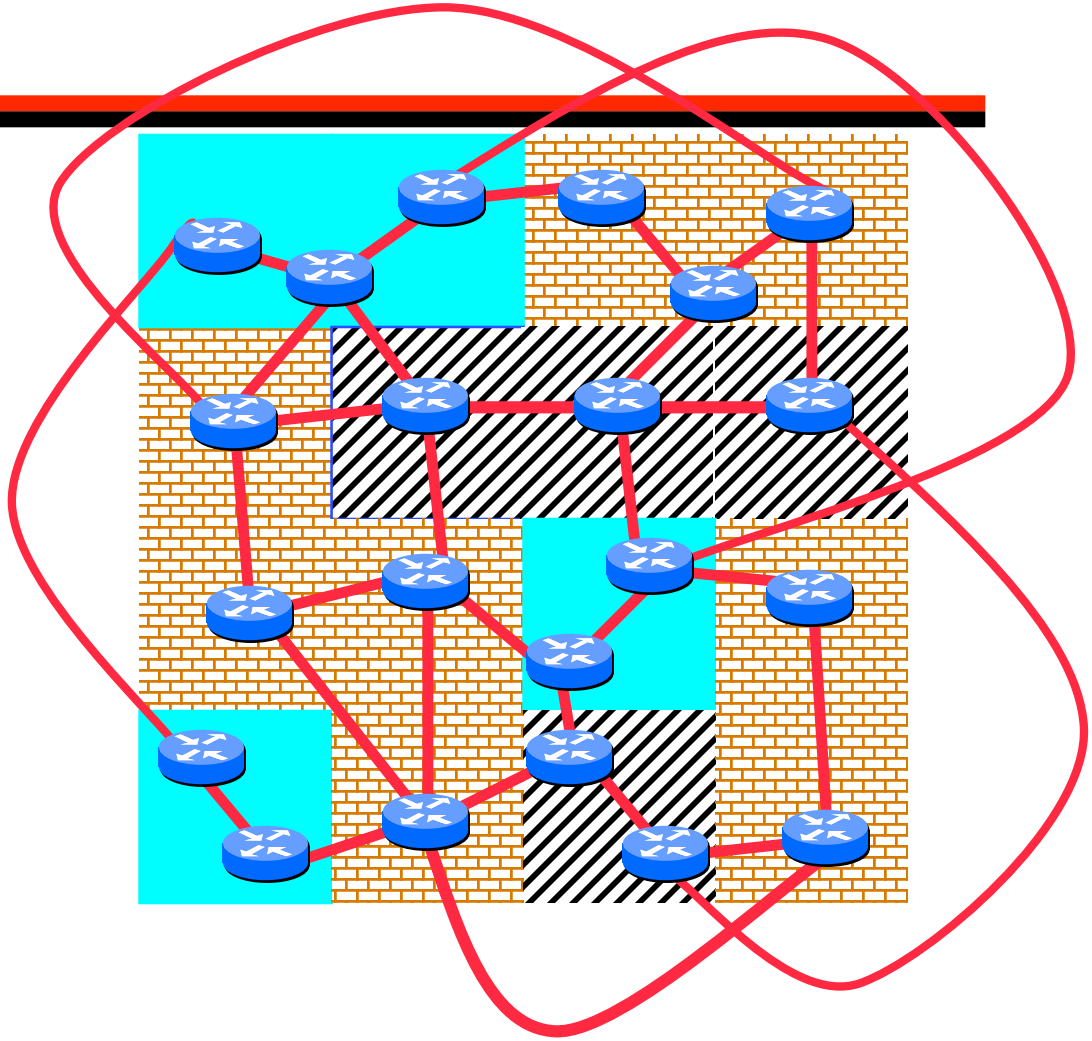


AS Graphs Do Not Show Topology!

BGP was designed to throw away information!

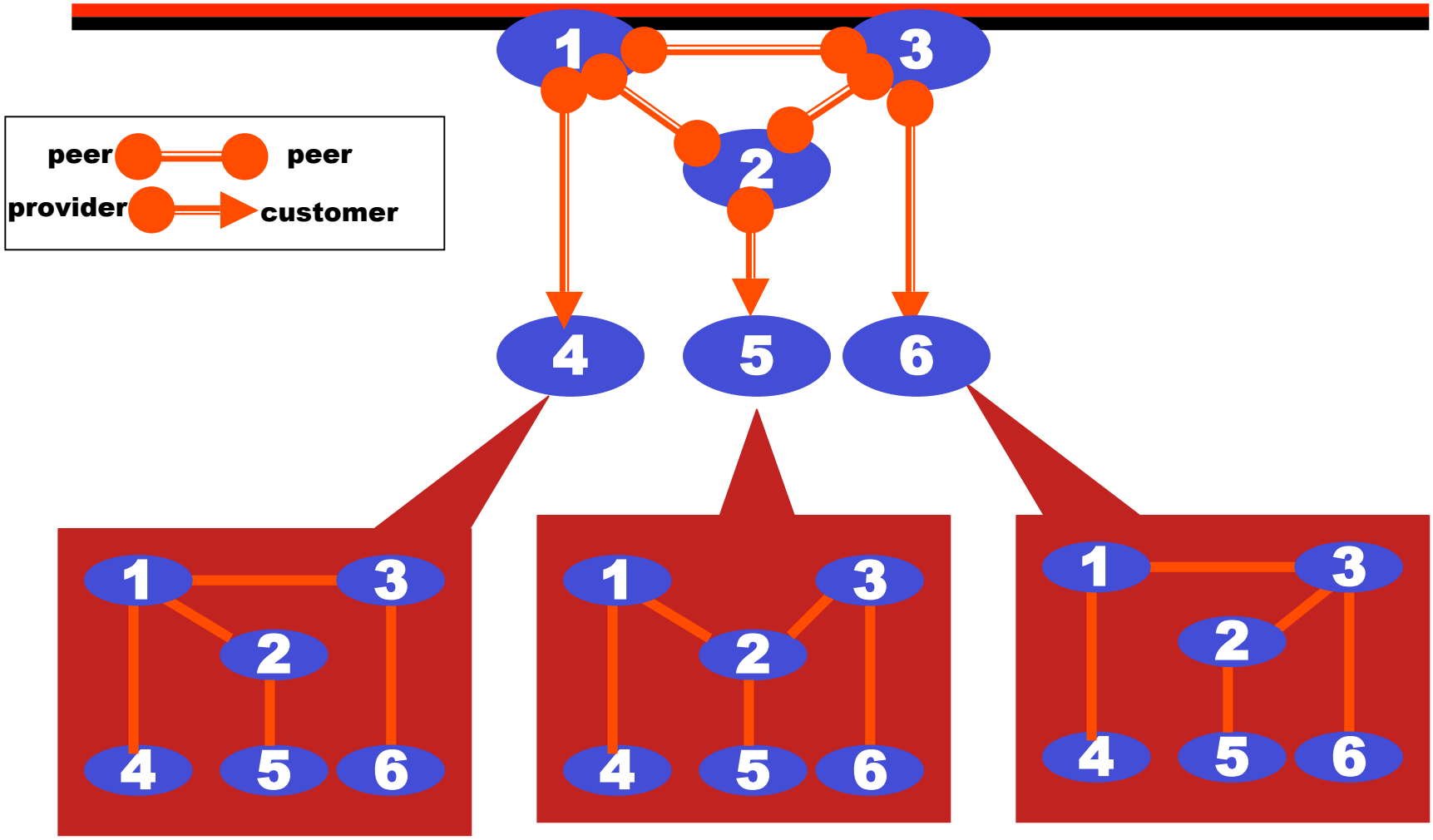


The AS graph may look like this.



Reality may be closer to this…

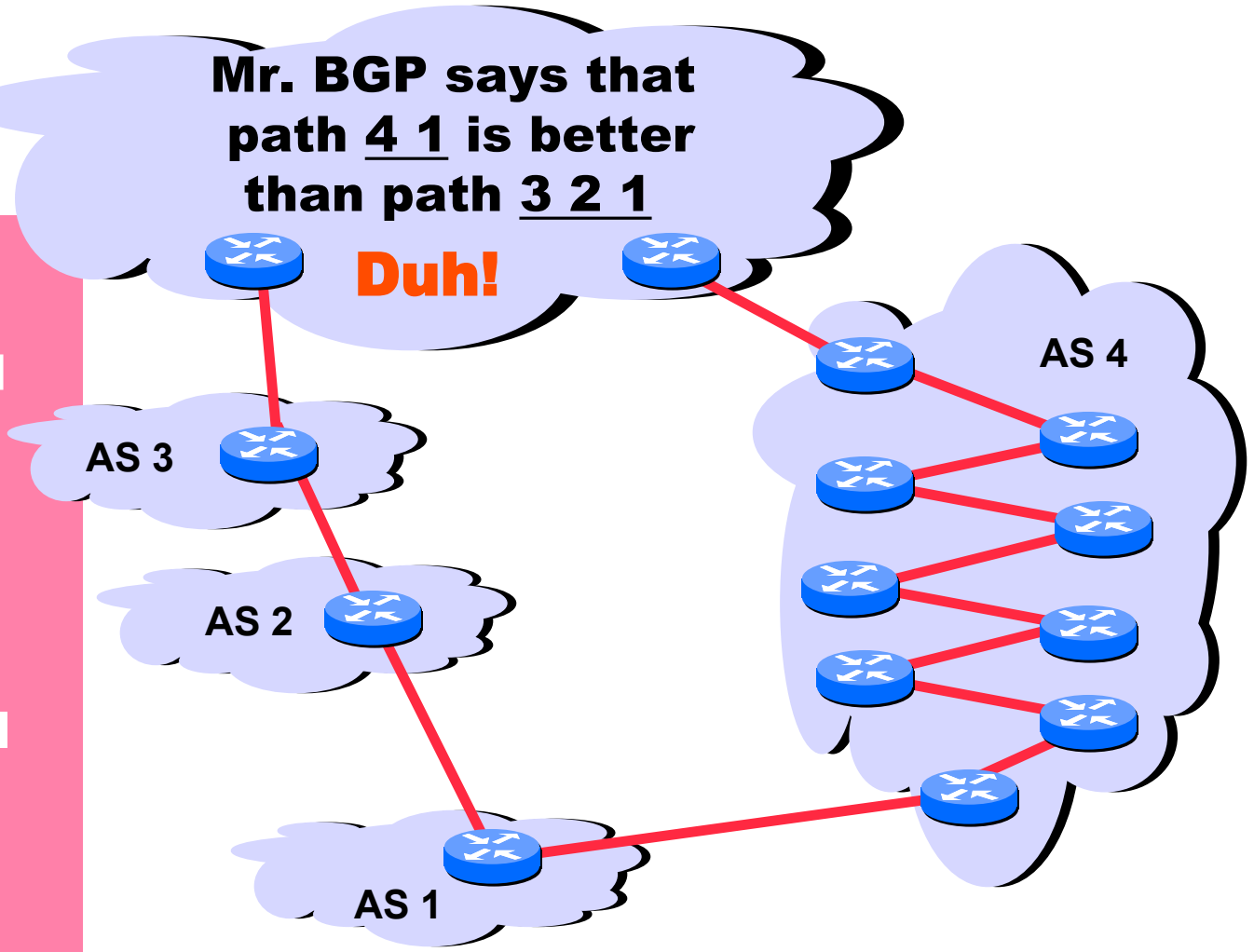
AS Graphs Depend on Point of View



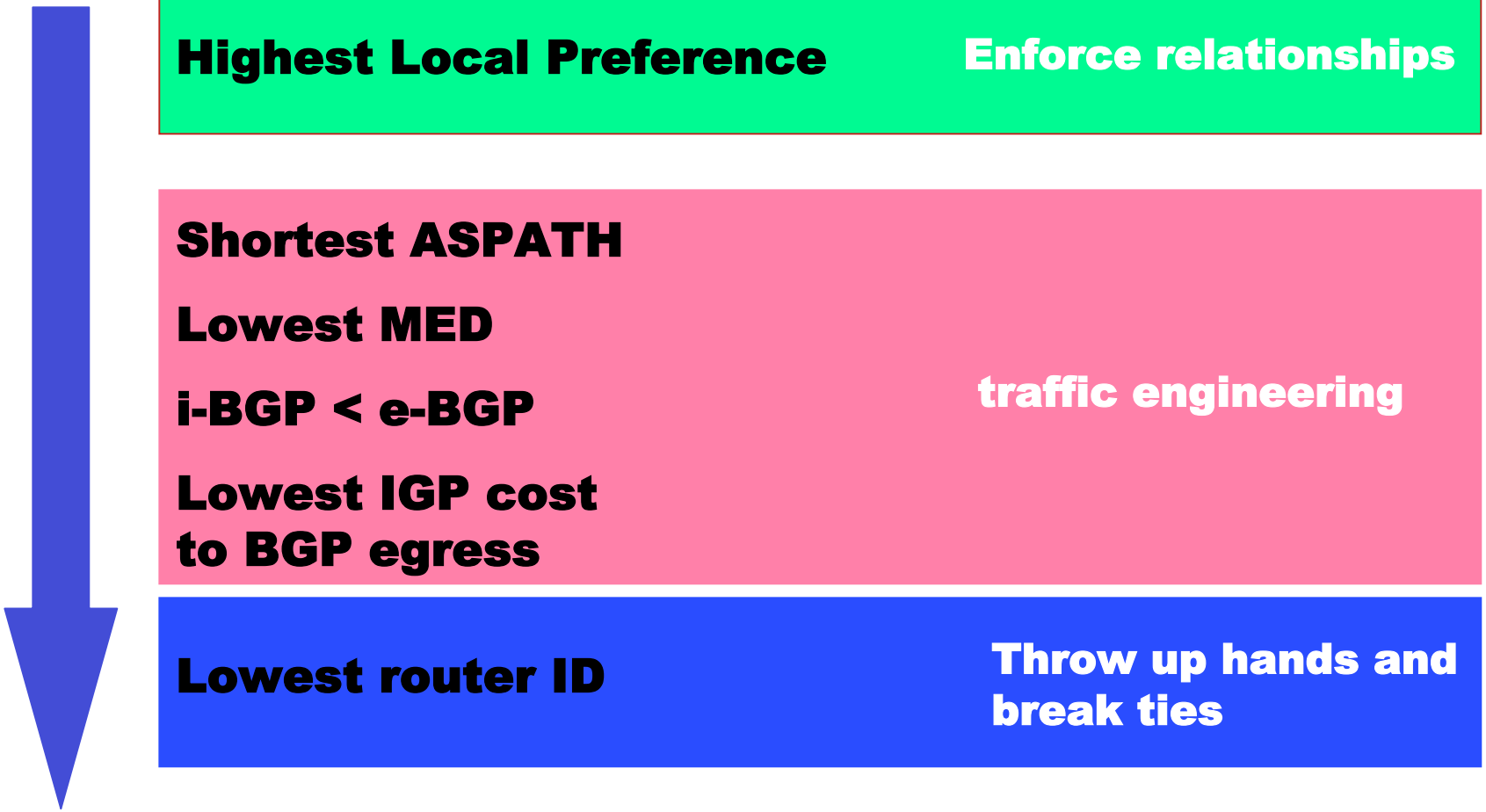
This explains why there is no UUNET (701) Sprint (1239) link on previous slide!

Shorter Doesn't Always Mean Shorter

In fairness: could you do this "right" and still scale? Exporting internal state would dramatically increase global instability and amount of routing state



Route Selection Summary

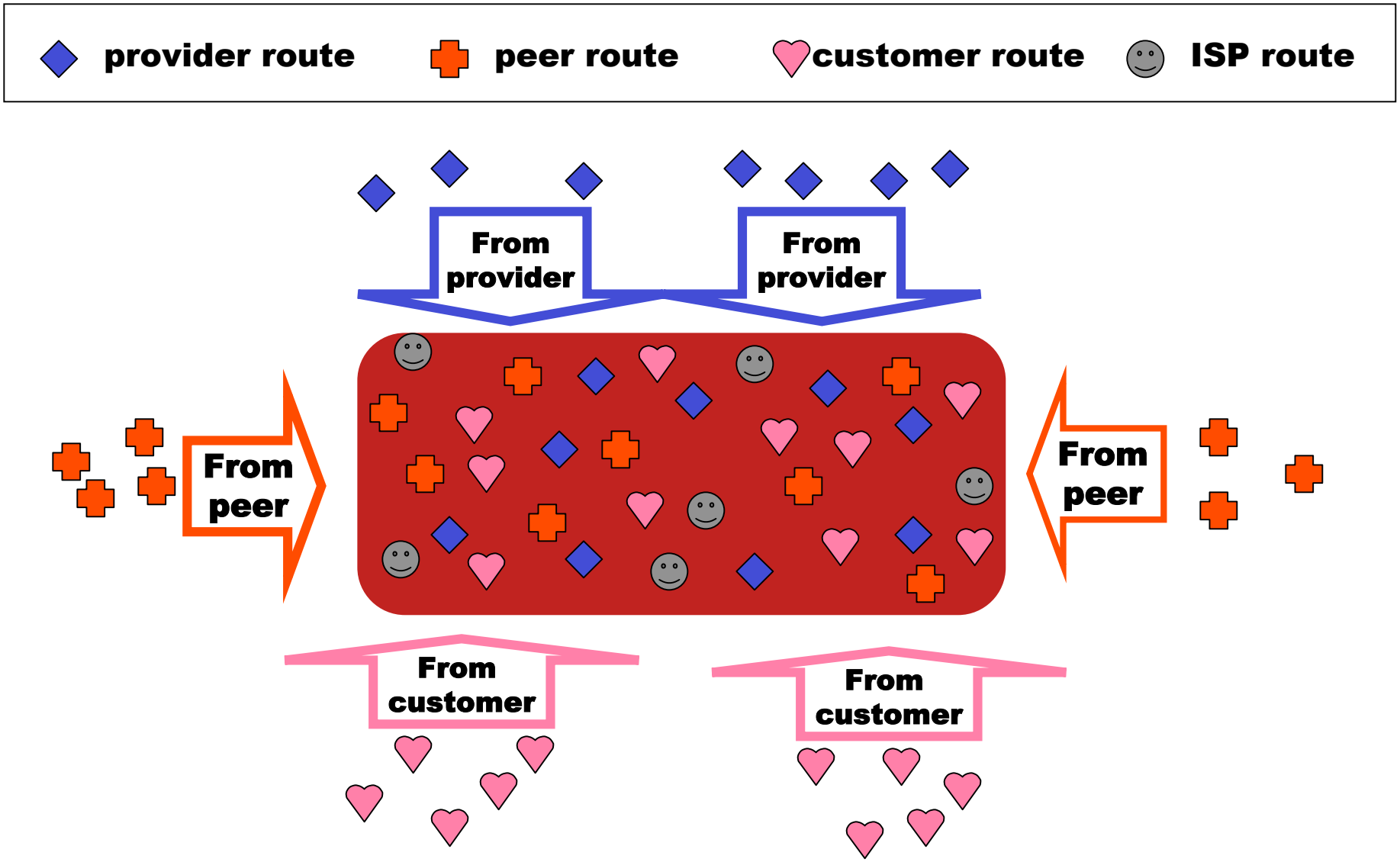


Implementing Customer/Provider and Peer/Peer relationships

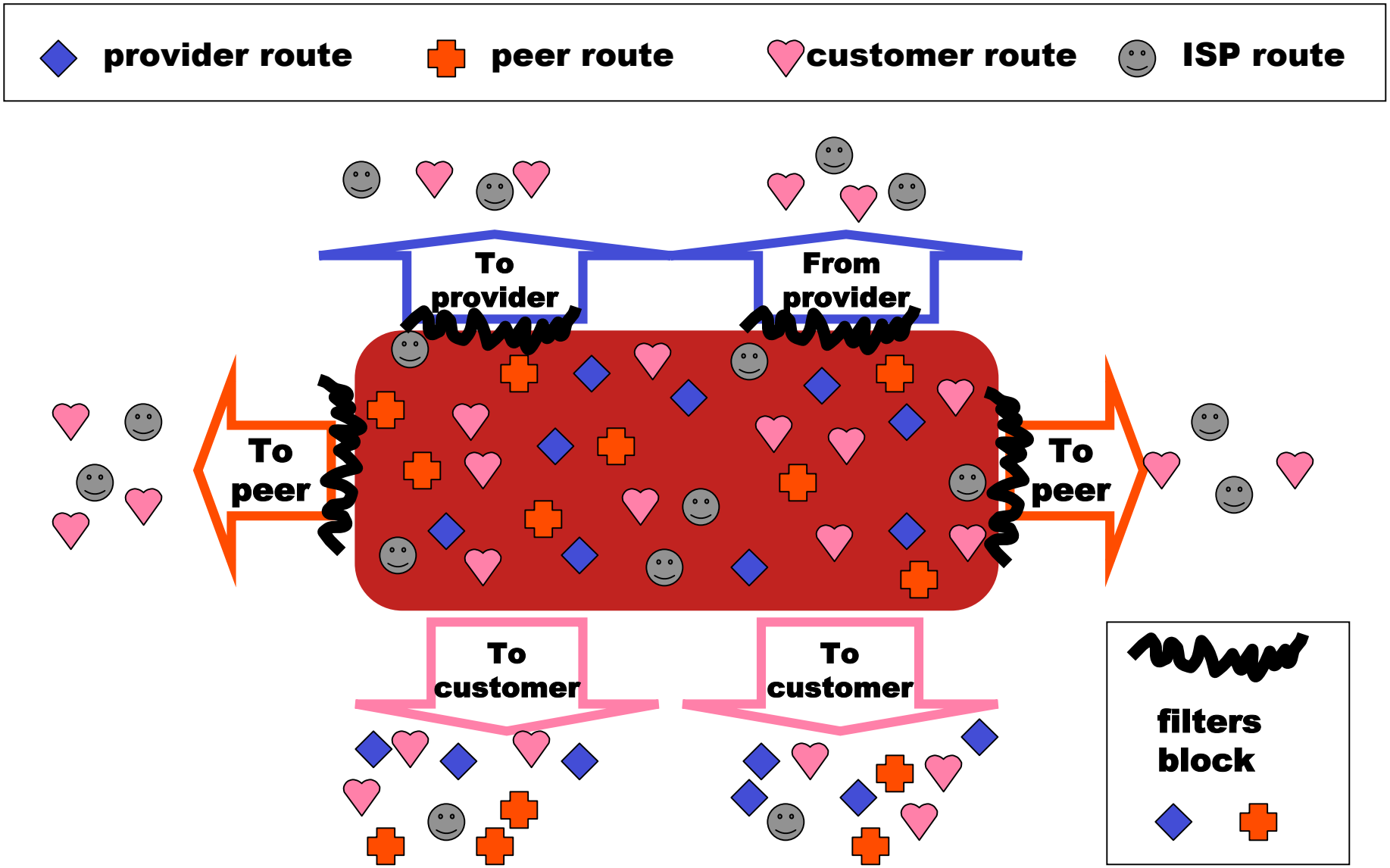
Two parts:

- Enforce transit relationships
  - Outbound route filtering
- Enforce order of route preference
  - provider < peer < customer

Import Routes



Export Routes

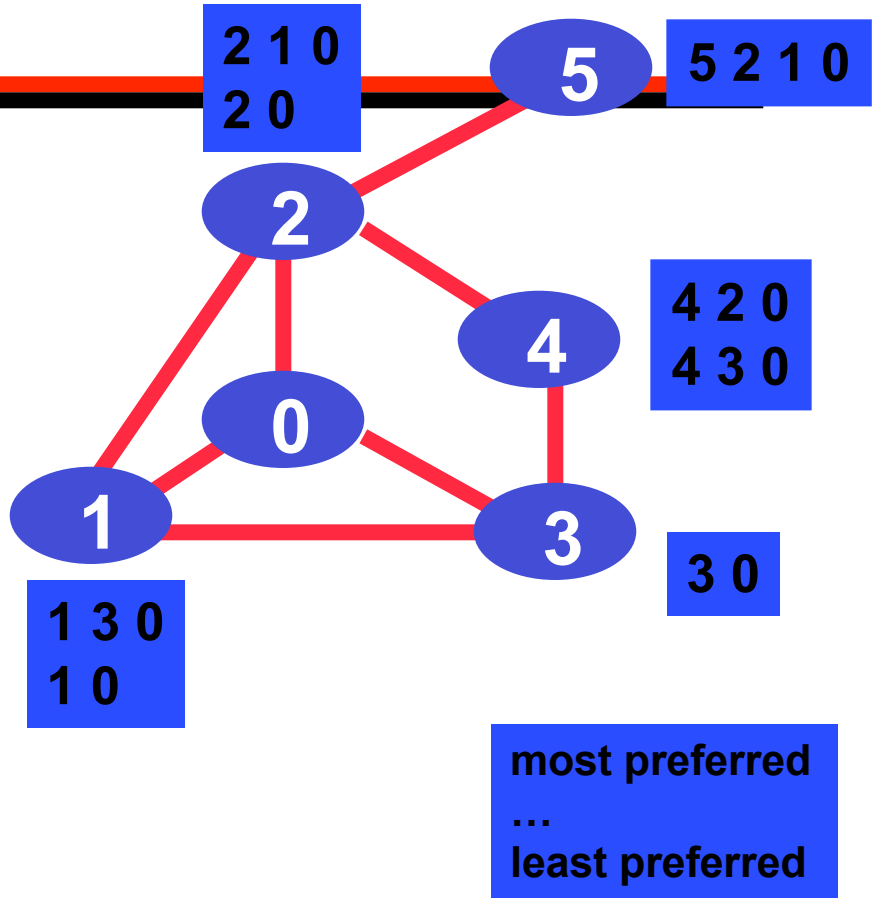


Bad News

- **BGP is not guaranteed to converge on a stable routing. Policy interactions could lead to "livelock" protocol oscillations.**
  See "Persistent Route Oscillations in Inter-domain Routing" by K. Varadhan, R. Govindan, and D. Estrin. ISI report, 1996
- **Corollary: BGP is not guaranteed to recover from network failures.**

An instance of the *Stable Paths Problem* (SPP)

- A graph of nodes and edges,
- Node 0, called *the origin*,
- For each non-zero node, a set or permitted paths to the origin. This set always contains the "null path".
- A ranking of permitted paths at each node. Null path is always least preferred. (Not shown in diagram)



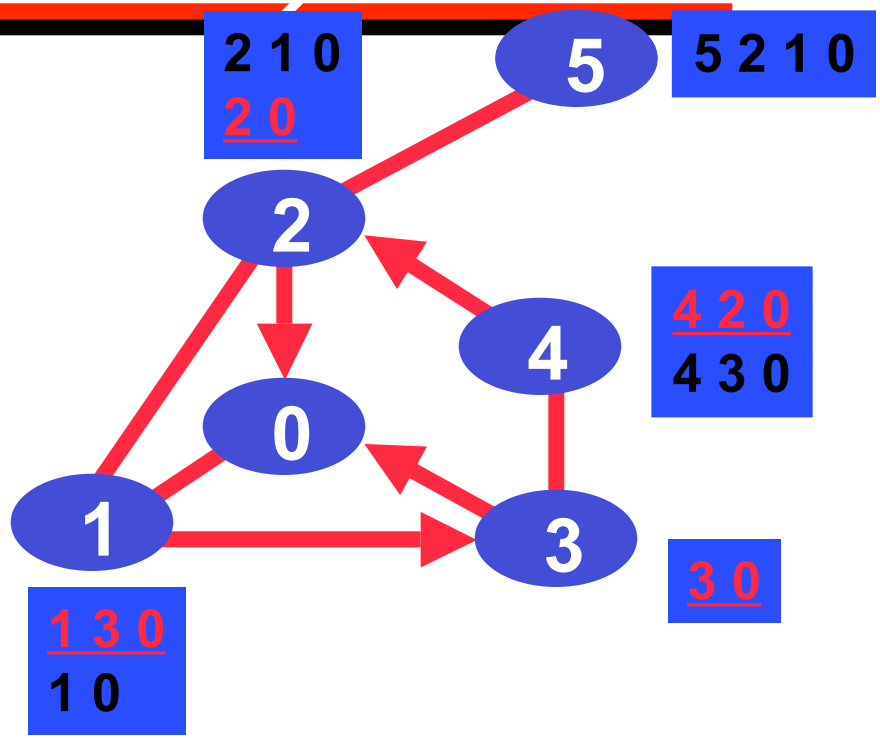
When modeling BGP : nodes represent BGP speaking routers, and 0 represents a node originating some address block

Yes, the translation gets messy!

A Solution to a Stable Paths Problem

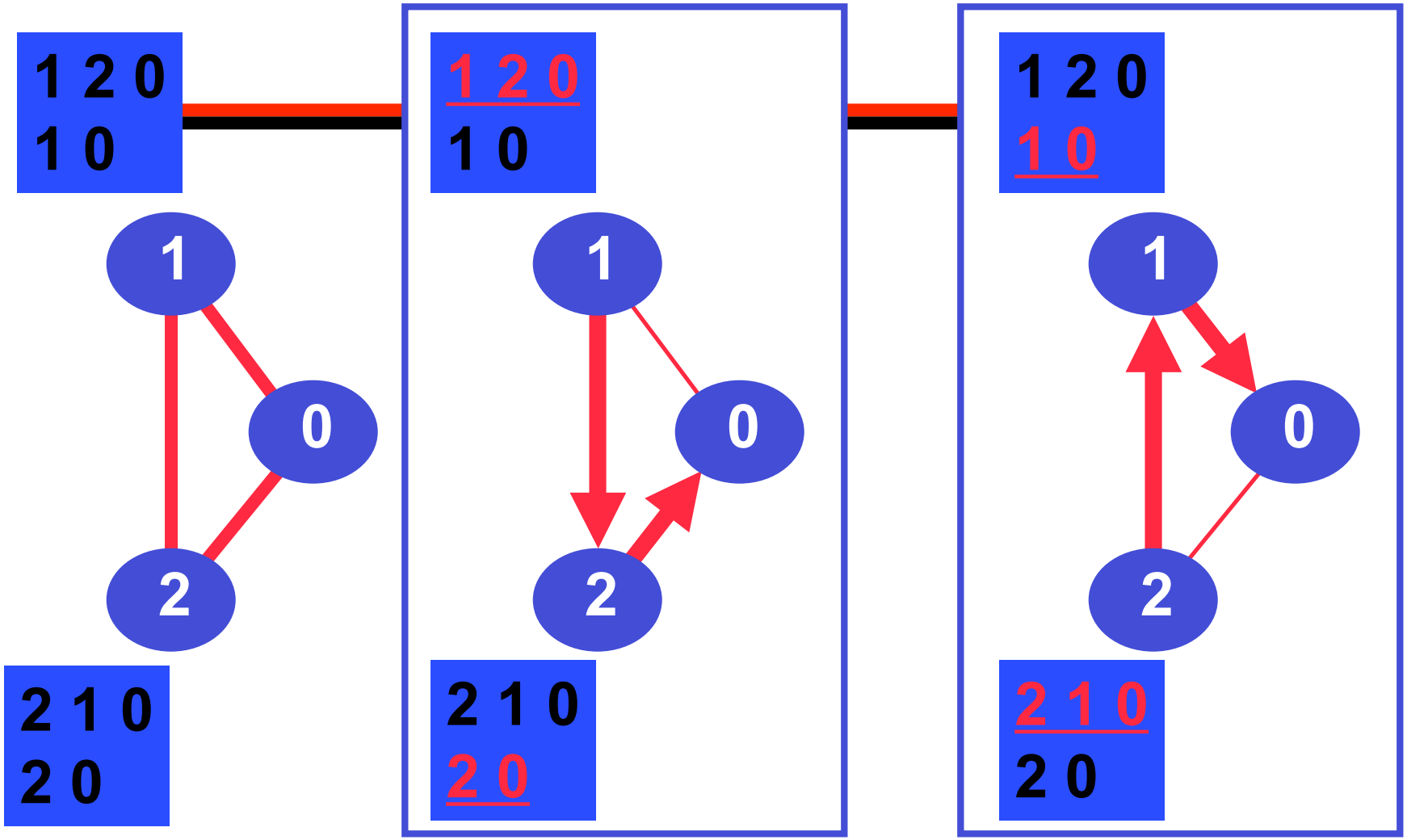
A *solution* is an assignment of permitted paths to each node such that

- node u's assigned path is either the null path or is a path uwP, where wP is assigned to node w and {u,w} is an edge in the graph,
- each node is assigned the highest ranked path among those consistent with the paths assigned to its neighbors.

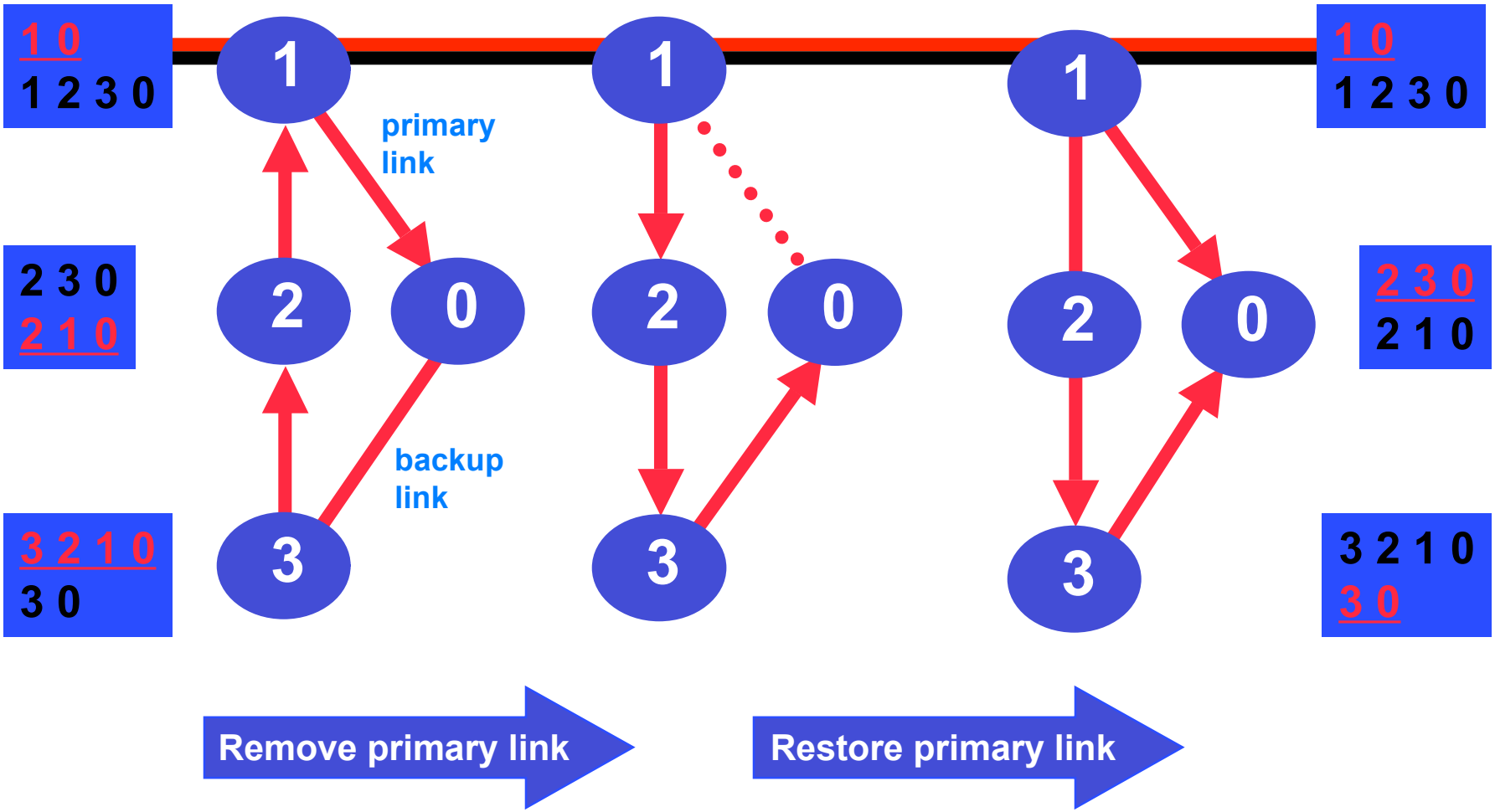


A Solution need not represent a shortest path tree, or a spanning tree.

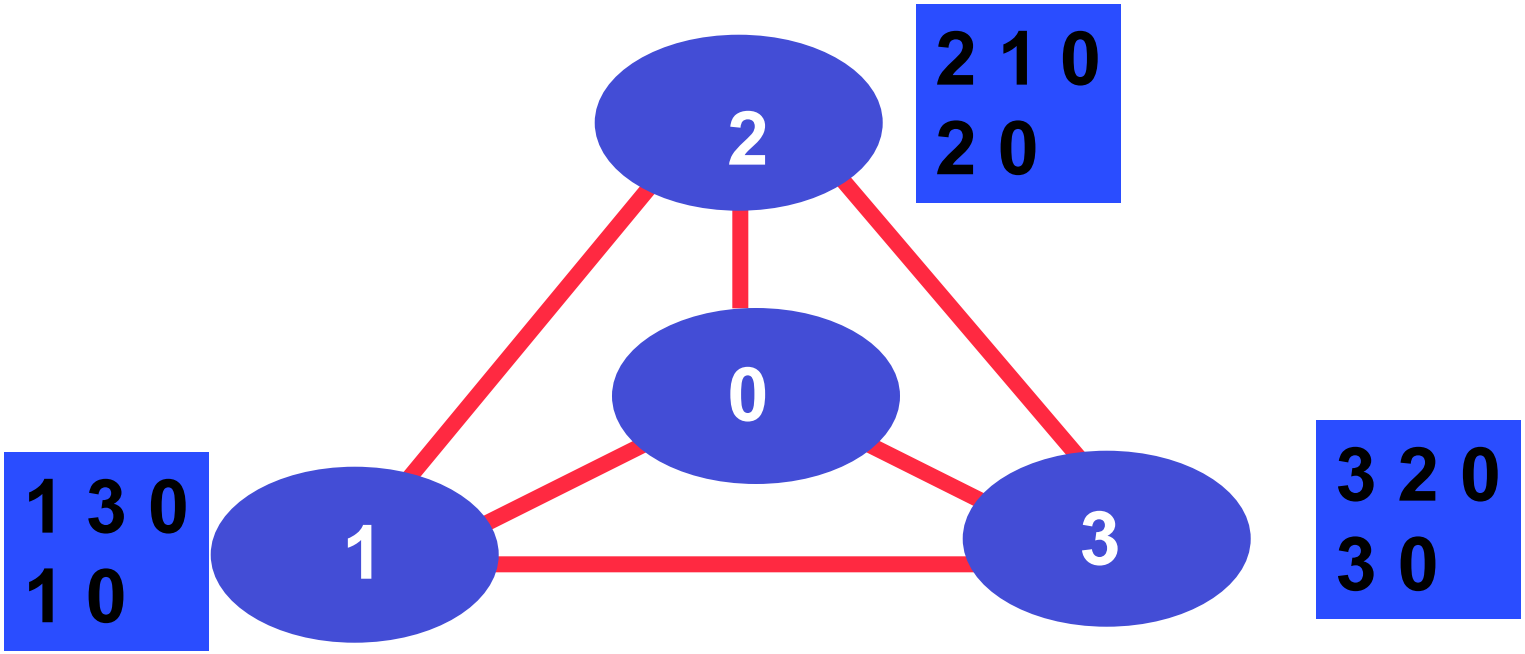
An SPP may have multiple solutions



Multiple solutions can result in "Route Triggering"



BAD GADGET : No Solution



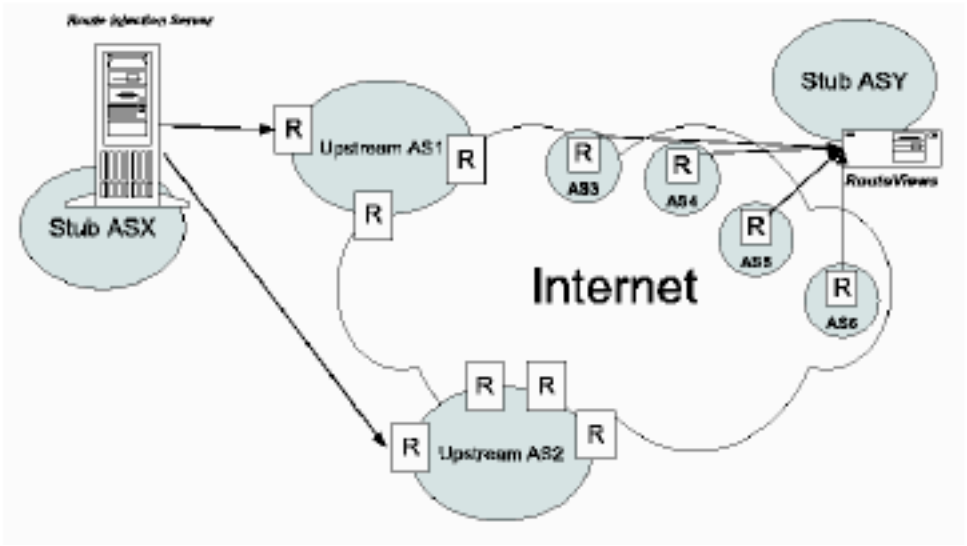
Persistent Route Oscillations in Inter-Domain Routing. Kannan Varadhan, Ramesh Govindan, and Deborah Estrin. Computer Networks, Jan. 2000

Labovitz 00 (slides by S. Savage)

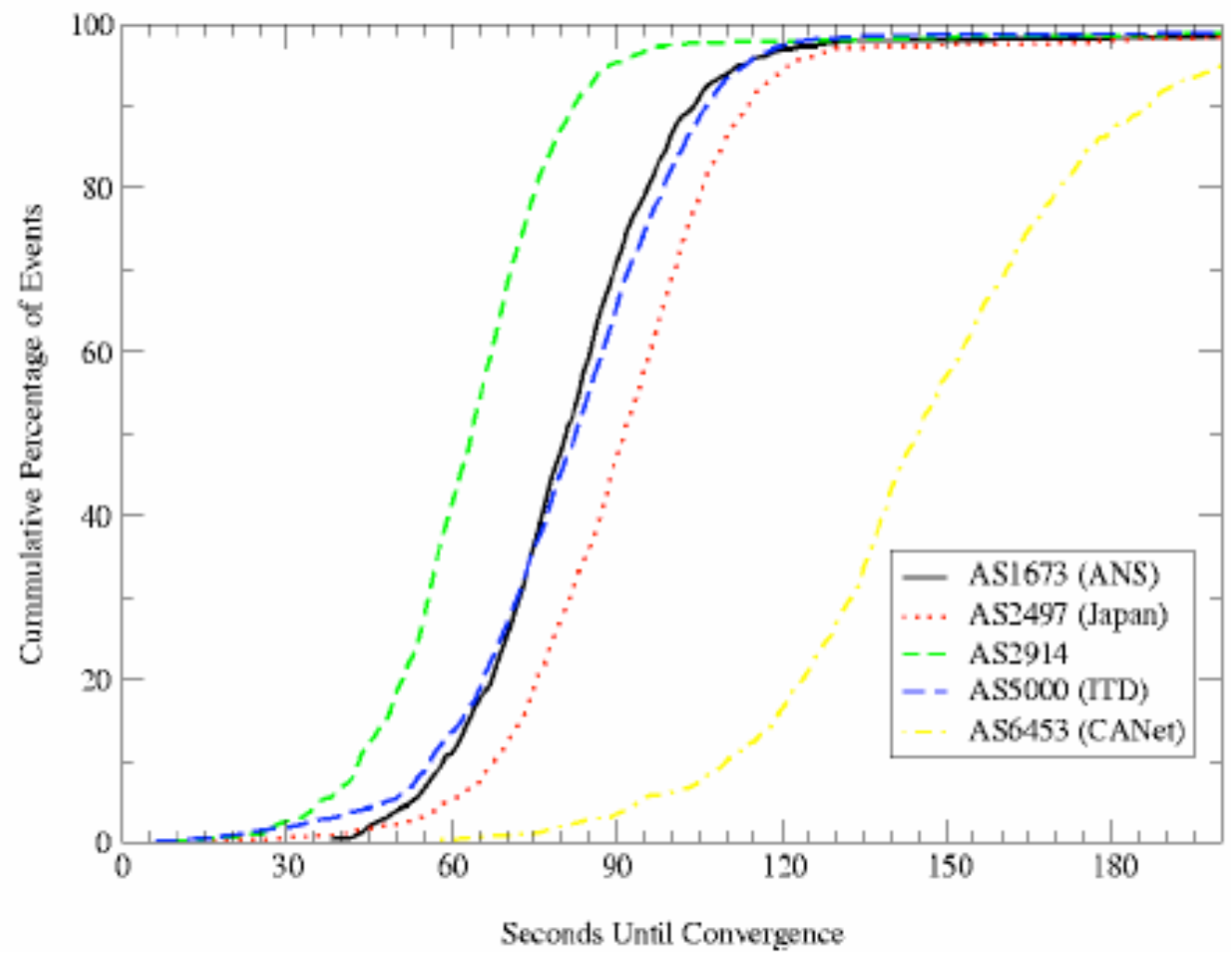
- When a node/link fails, how quickly can BGP recover?
  - Common wisdom
    - Very fast, a few milliseconds
    - Route withdrawals sent immediately
  - ASPATH loop detection eliminates problems with DV
  - Reality somewhat different…
- Meta-issue
  - Does BGP ever converge?
  - Griffin et al00 show that with unconstrained policies it doesn't have to and its NP-hard to tell if it does
  - However, Labovitz et al, deal with constrained policies that do converge

Active Route Measurement

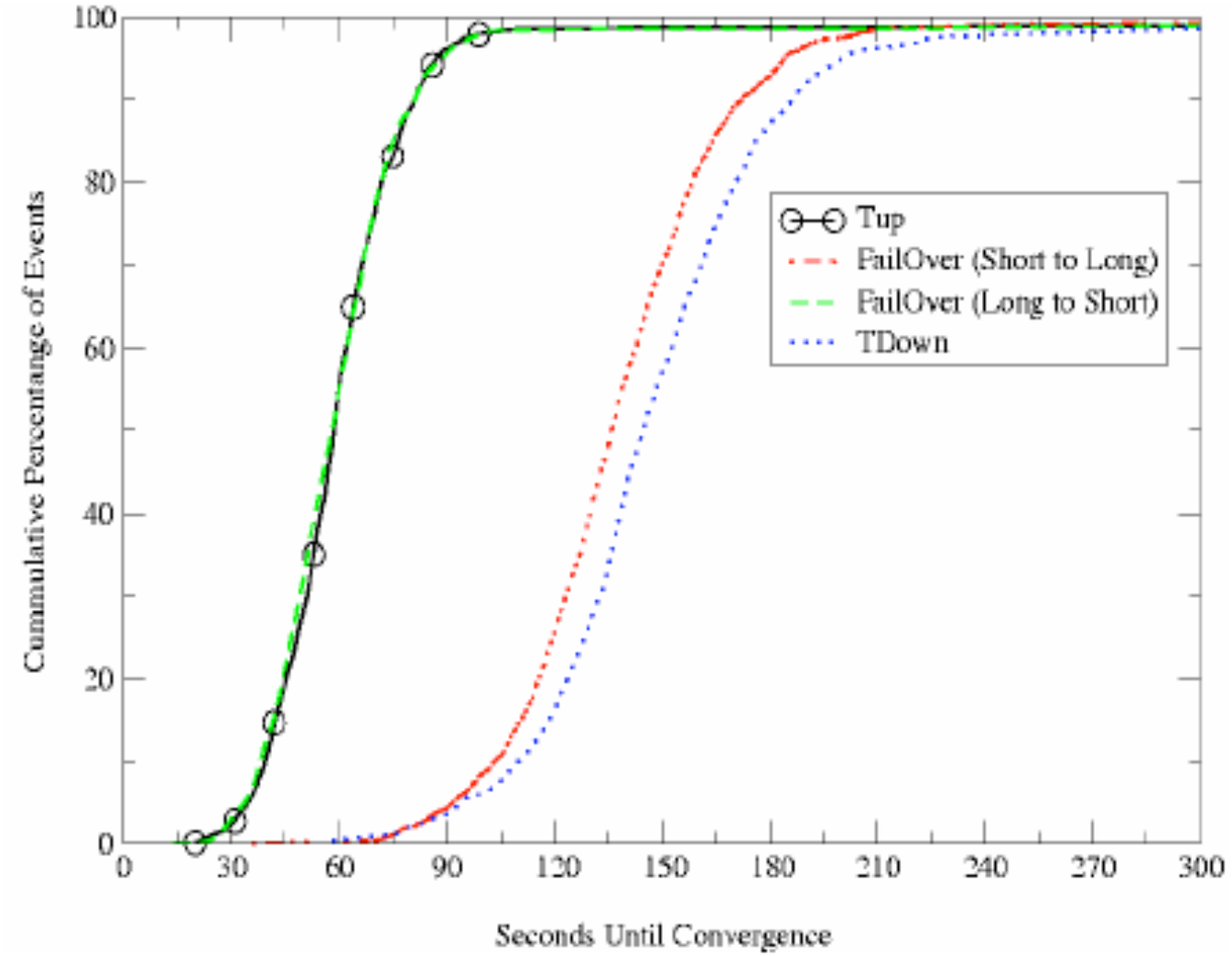
- **Inject routes into geographically and topologically points in the network**
- **Periodically fail and change these routes**
- **Time events using ICMP echos (ping), HTTP GET and time-synchronized monitoring machines**



Time to converge after route failure



Time to Repair or Failover



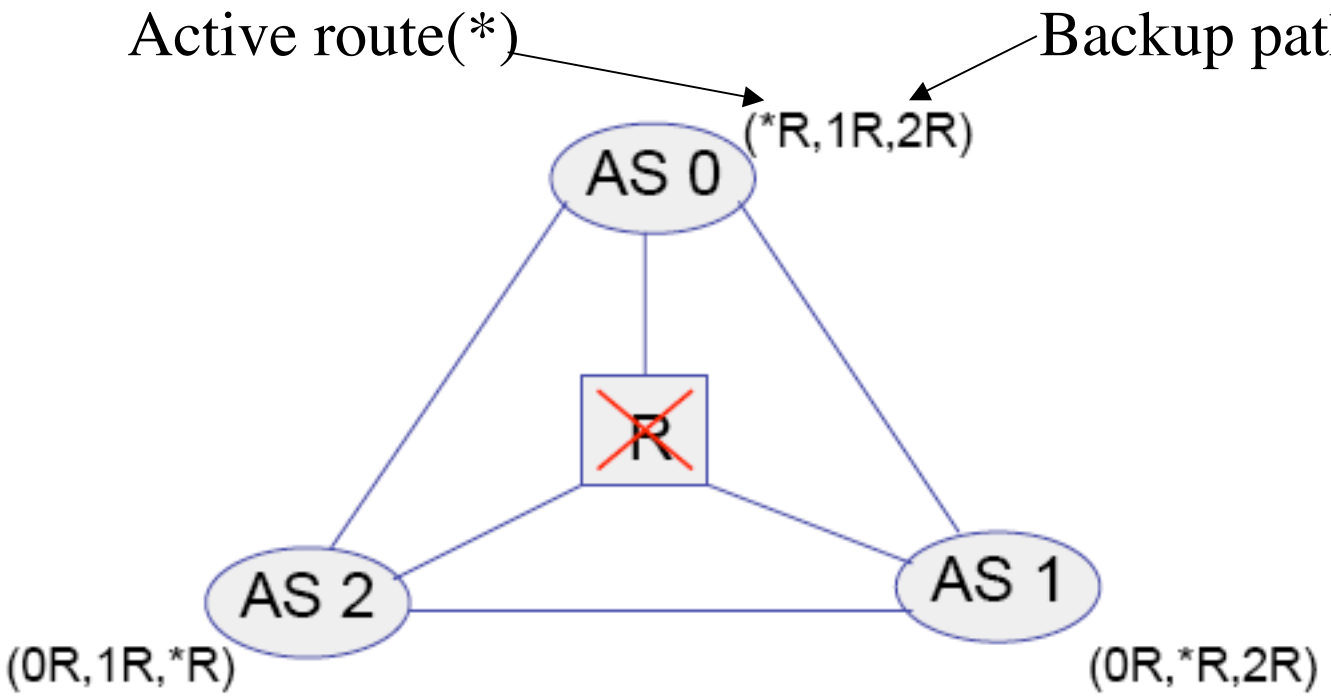
Observations

- Repairs (Tup time) exhibit similar convergence properties as long-short ASPath fail-over
- Failures (Tdown) and short-long fail-overs (e.g. primary to secondary paths) also similar
  - Slower than a repair (bad news doesn't travel fast, DV protocols)
  - 60% take > 2 minutes

Why “long” failovers?

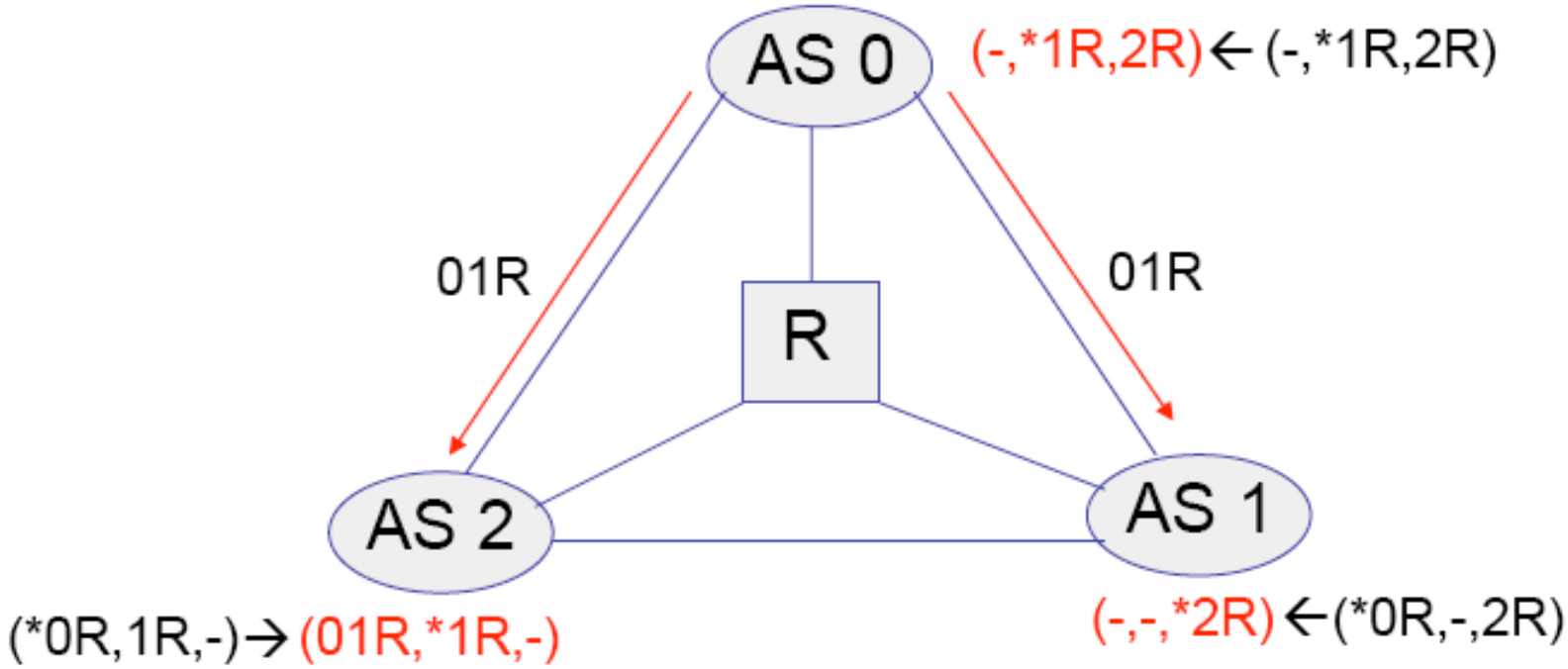
- **Route oscillation**
  - If policy just uses ASPATH length, then looks like a DV protocol; can still oscillate
  - Loop prevention doesn’t prevent this
  - Can explore every possible path through network → (n-1)! Combinations
- **Timers**
  - 30MinAdver timer makes router wait between updates
  - Adds artificial “rounds” to propagation speed
- **Loop detection**
  - Waits to send routes that have loops in them (prevailing BGP implementation only does receiver based loop detection)
- **Typical Internet failover times**
  - New/shorter link → 60 seconds: simple replacement at nodes
  - Down link → 180 seconds: search of possible options
  - Longer link → 120 seconds: replacement or search based on length

Example BGP Oscillation (step 1)



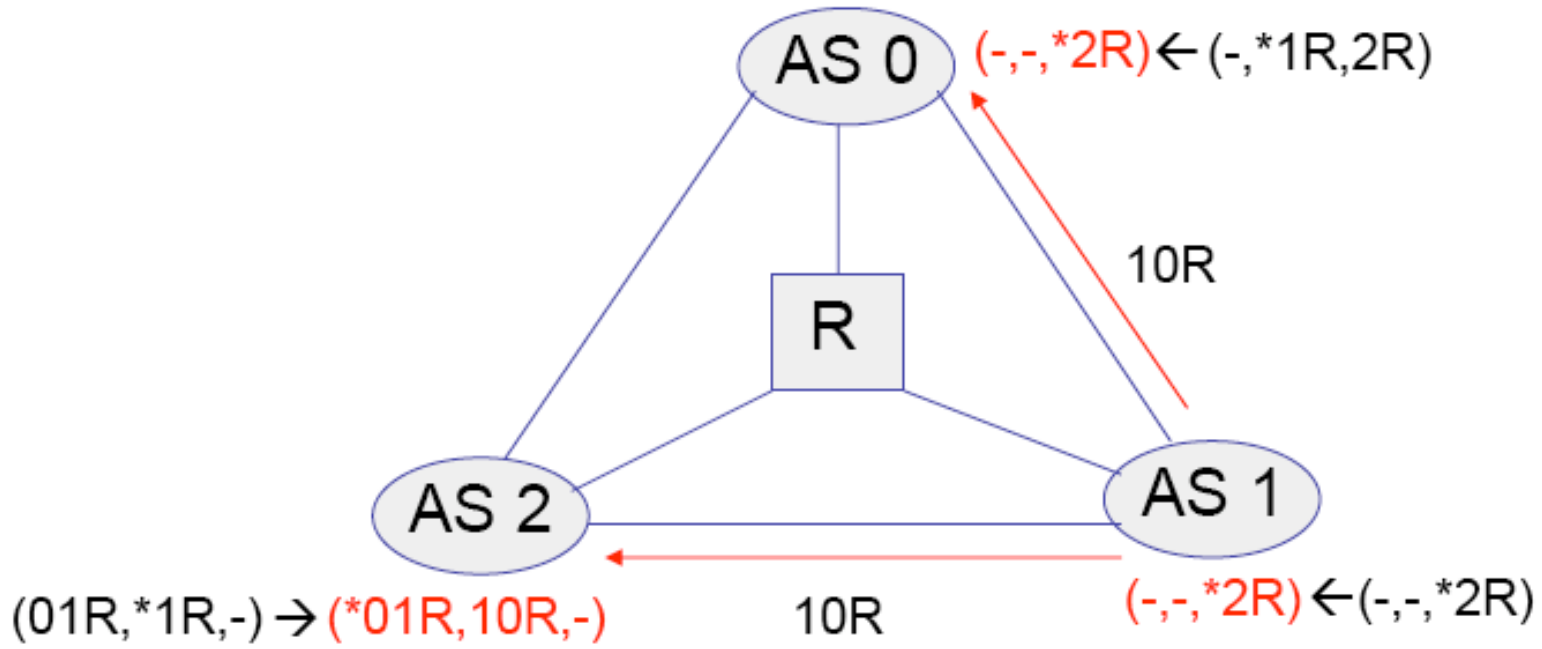
Example BGP Oscillation (step 2)

1 and 2 receive announcement from 0



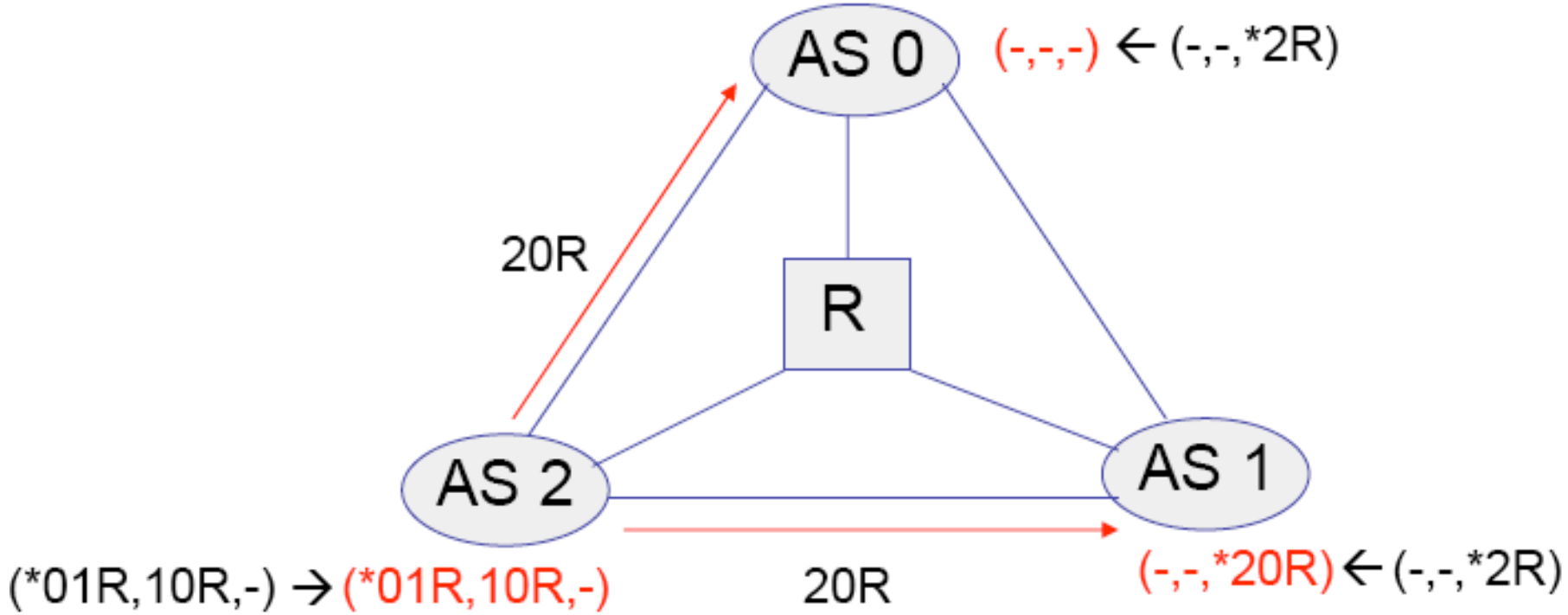
Example BGP Oscillation (step 3)

0 and 2 receive announcement from 1



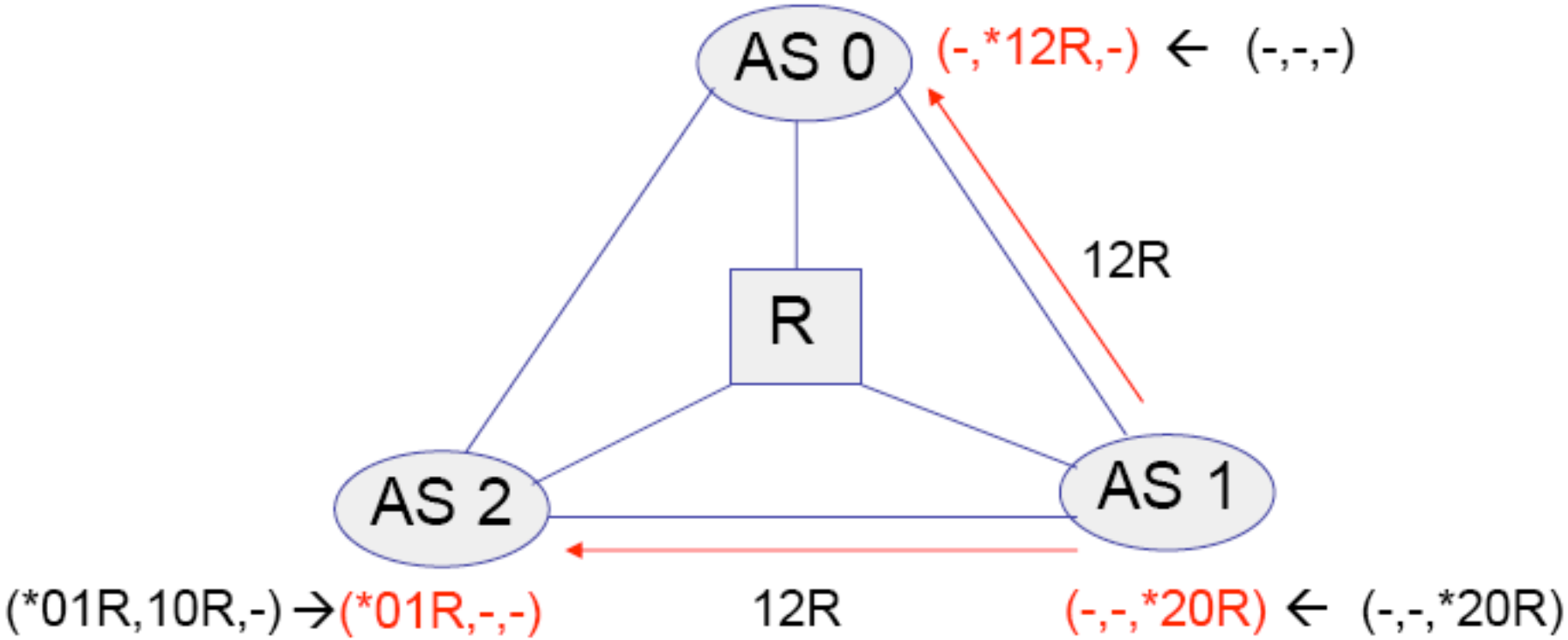
Example BGP Oscillation (step 4)

0 and 1 receive announcement from 2



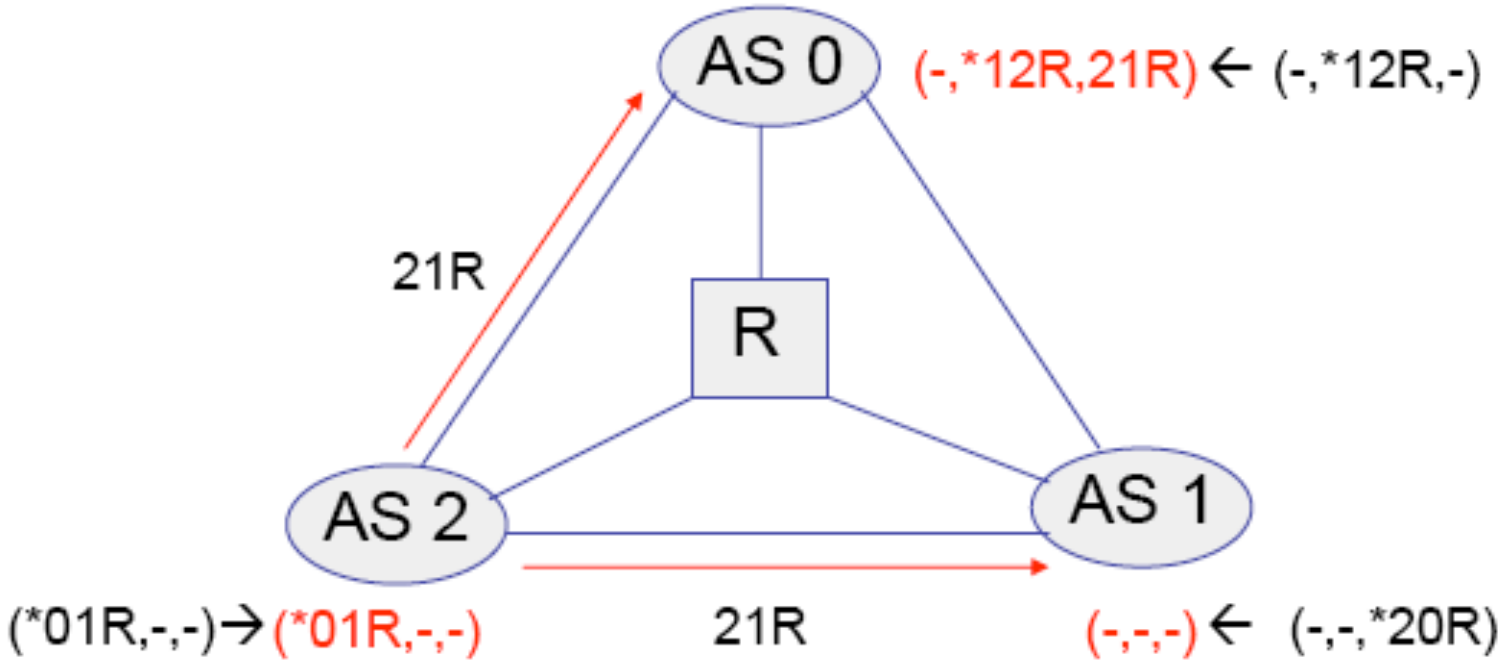
Example BGP Oscillation (step 5)

0 and 2 receive announcement from 1



Example Oscillation (step 6)

0 and 1 receive announcement from 2



BGP impact on flow

- Slides by Sharad Agarwal

Traffic Collection

- Packet-level data
  - Ingress, OC-12
  - TCP/IP header convert to PDF
  - GPS synched timestamp

  – Analyzed 6 traces
    - 2 PoPs, 3 days, 6 different ASes

BGP Collection

- Zebra BGP collectors
  - 3 PoPs (including traffic collection)
  - iBGP RRC & eBGP sessions

- Focus on iBGP updates
  - How data packets exit Sprint
  - Reflects eBGP updates, internal policy & IGP changes

Traffic Analysis Method

- Correlate iBGP & traffic
  - Find egress PoP for each data packet
    - Longest prefix match, router to PoP map
    - Ingress link to multiple egress PoP fan-out
  - Identify traffic variability due to BGP updates
    - Static BGP table + data packets
      - Shows variability due to other factors
    - Dynamic BGP table + data packets

Overview

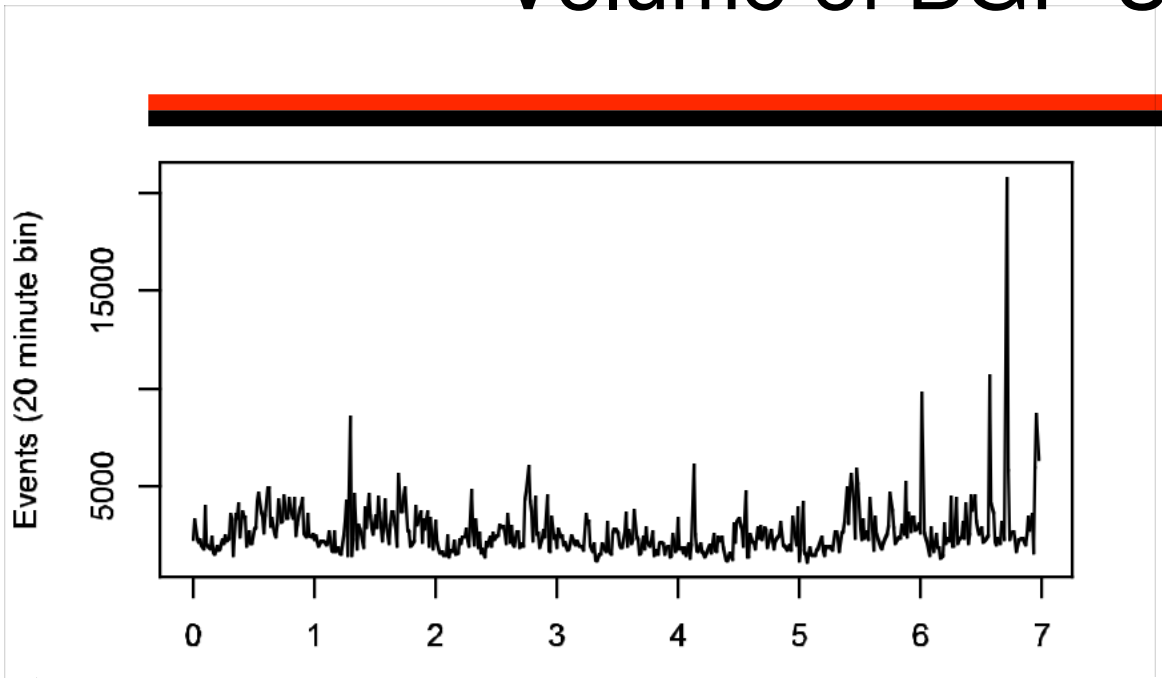
- Problem statement
- Methodology
  - Data collection & analysis
- Results
  - Likely causes
- Conclusions & implications

Results Presented

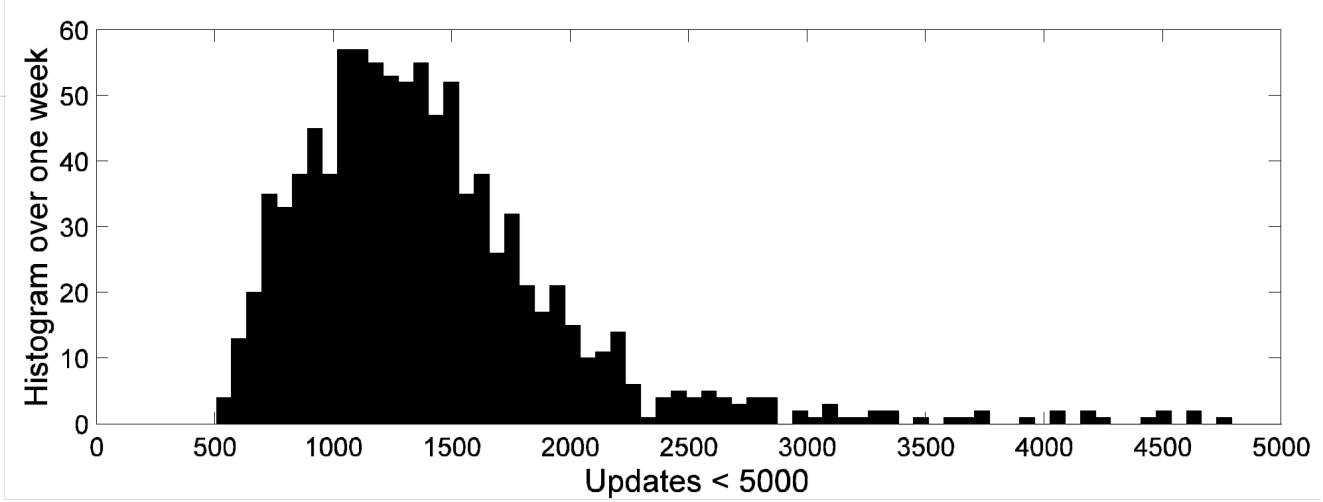
- 22 hour packet trace from Aug 6 2002
  - 112 Mbps average link utilization

- Traffic fan-out approximations
  - 2,649,315,251 packets
  - BGP table calculated every 20 minutes
  - Addresses carrying 99% of traffic
    - ~30,000 destination addrs of 200,000

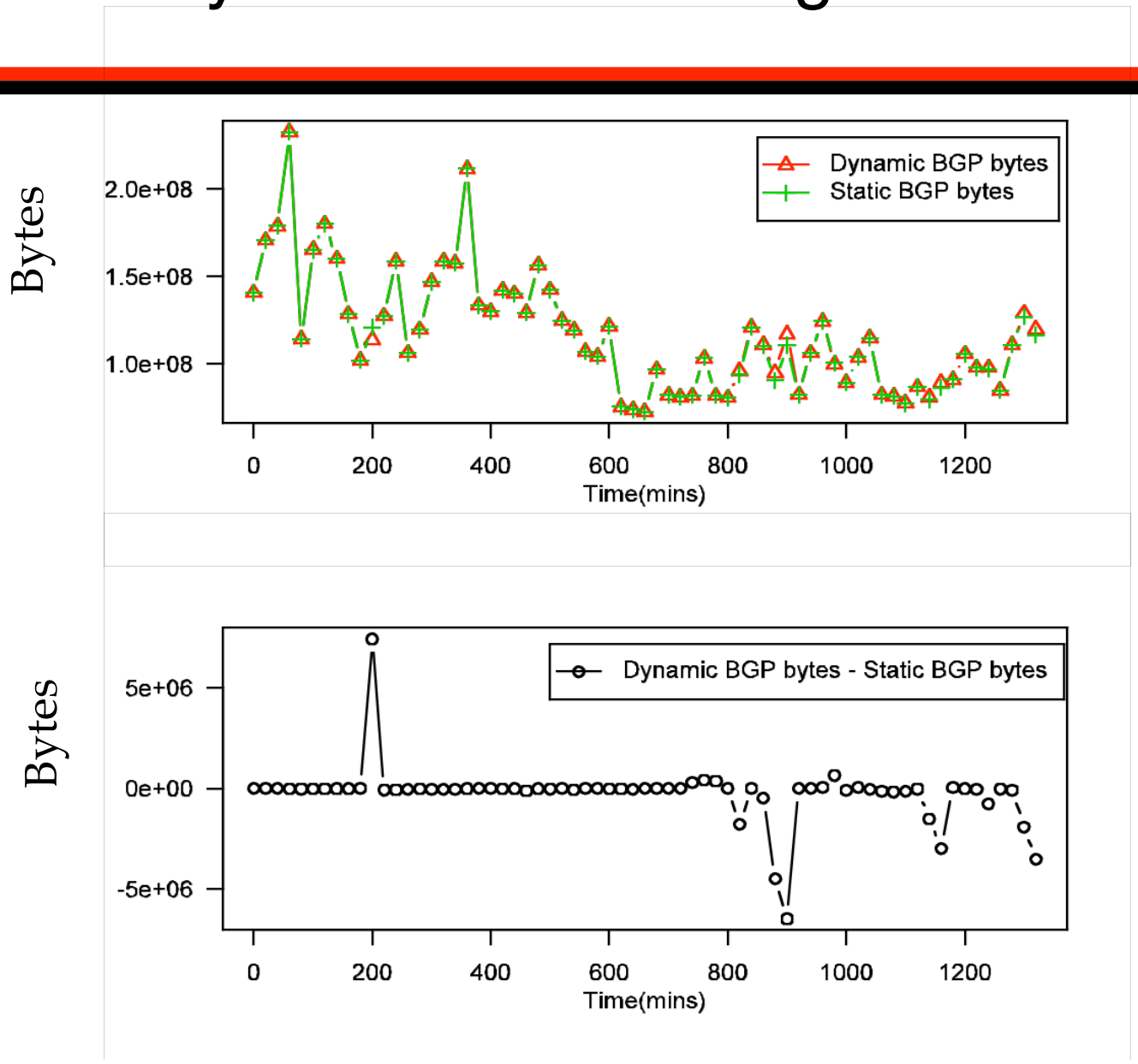
Volume of BGP Updates



- iBGP
  - Mean 1330/10min
  - Max 93320/10min
- Spikes
  - Maintenance?



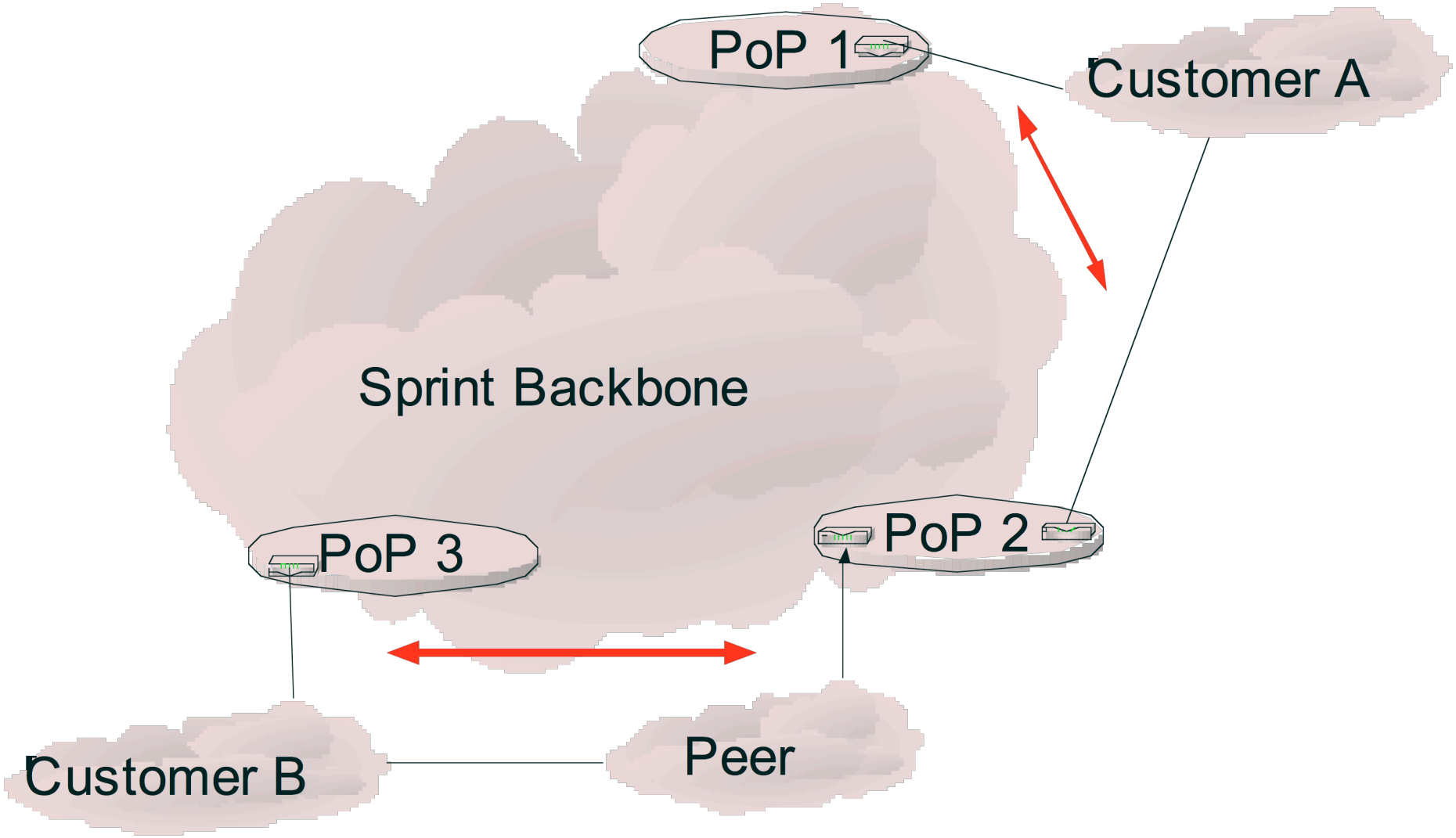
Variability in Traffic to an Egress PoP



Summary of Results

- 1~3% of variability in traffic to all PoPs
  - Representative of other traces
- Specific sources of variability
  - Networks with multiple links/paths to Sprint
  - BGP updates cause shift between inter-AS paths
    - Causes shift between intra-Sprint paths

Case Study



Few ASes Involved

- Single next-hop AS in 47% of all traffic shifts
- Single last-hop AS in 46% of all traffic shifts

- All egress PoP shifts happen once or twice for a destination

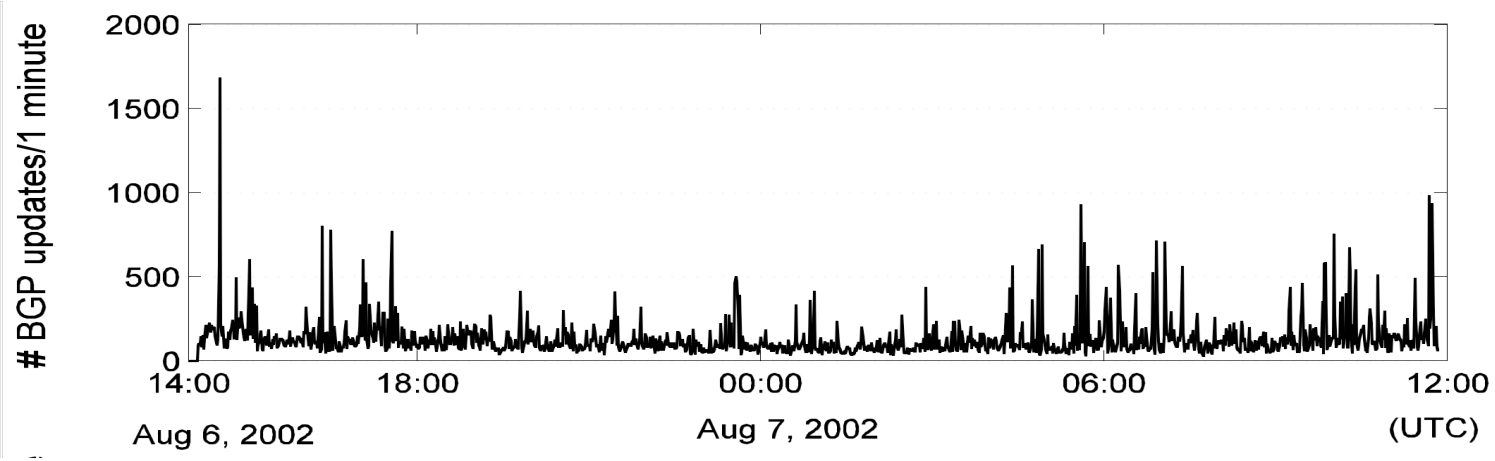
- But only 1% traffic. What of other traffic?

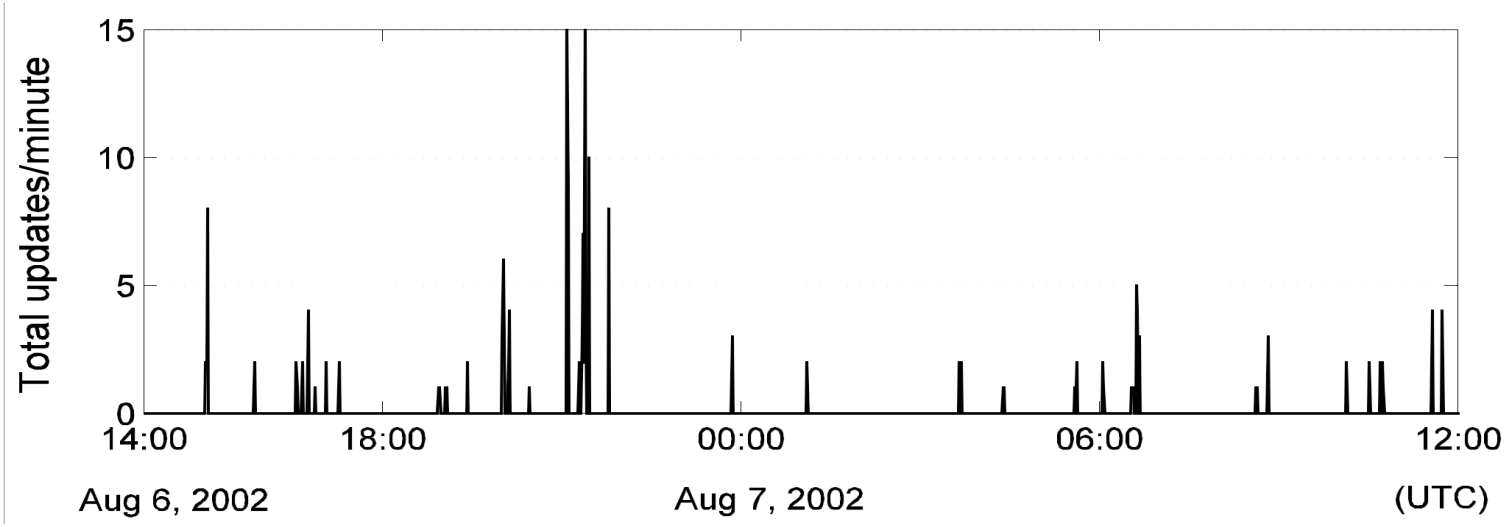
Backbone Traffic Characteristics

- Heavy hitters prevalent
  - ~200,000 destination addresses → 100% traffic
  - ~15 % of destination addresses → 99% traffic
  - ~1.5 % of destination addresses → 80% traffic

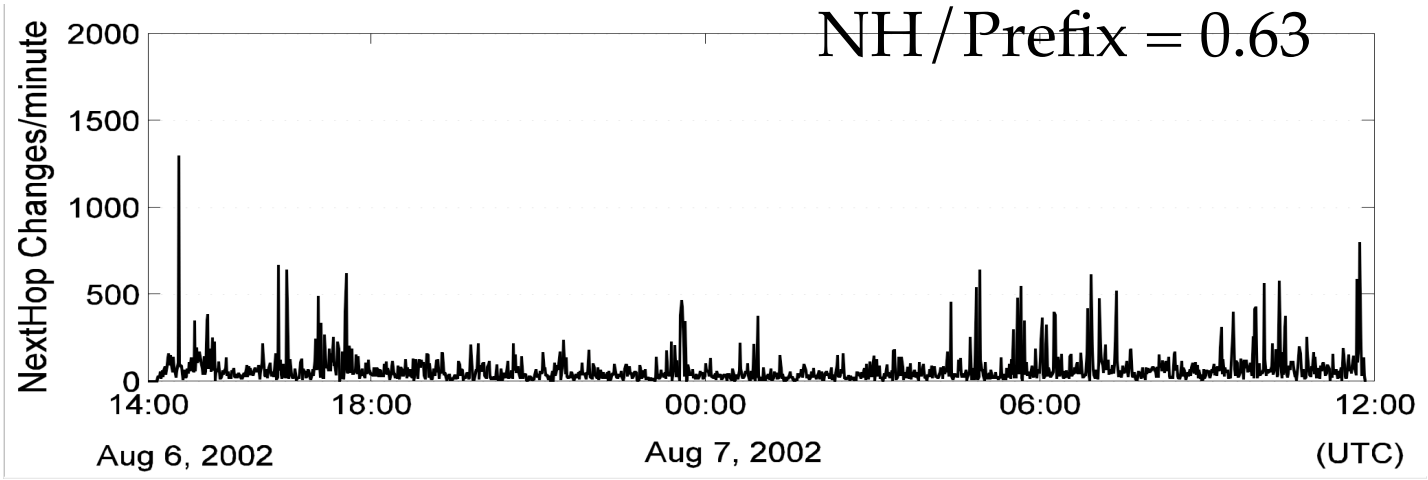
- Which updates affect heavy hitters?
- Which of these change egress PoP?

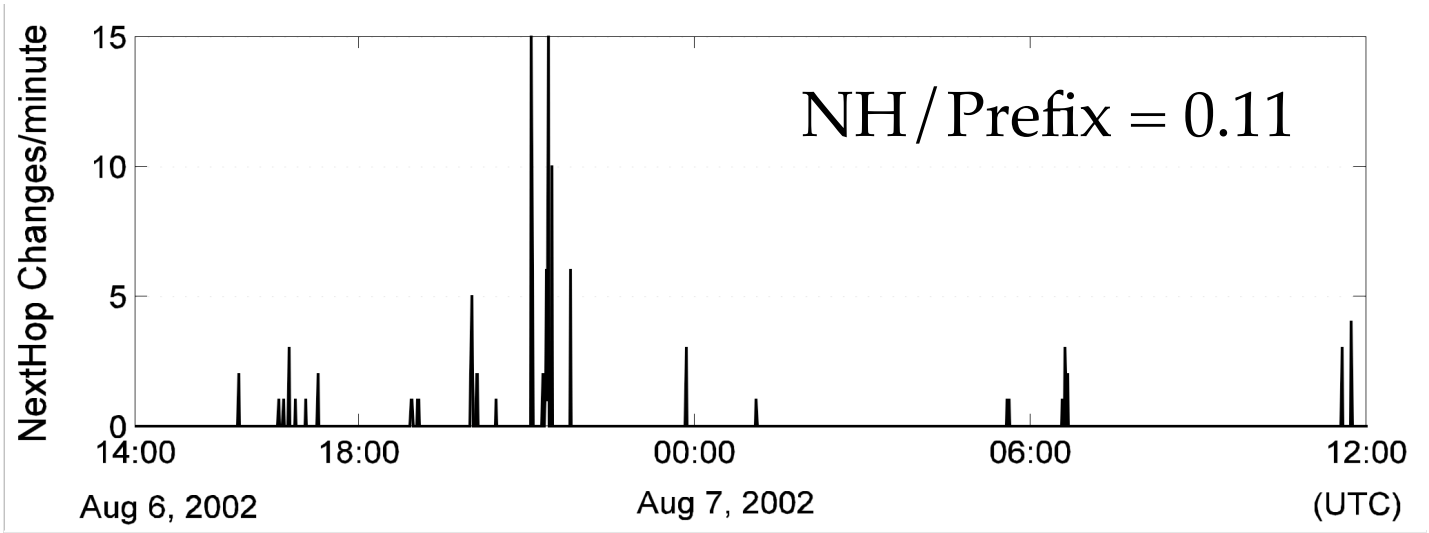
BGP Updates : Heavy Hitters





0.05% change Nexthop for Heavy Hitters





Conclusions

- BGP updates hardly affect intra-Sprint traffic fan-out
  - AT&T[Rexford02]: stable traffic → stable prefixes
  - Why?
    - Standard route filtering?
    - Stable prefixes *attract* stable traffic?
    - Layer3 protection switching and engineering?
  - Why so many other BGP updates?
  - Cause analysis : need all BGP sessions!

Implications

- BGP doesn’t cause latency variation in Sprint
  - Good for applications
- BGP doesn’t make link loads more dynamic
  - Provisioning / traffic engineering easier
  - Traffic matrix less variable
    - But still inherent variations in traffic

http://ipmon.sprint.com

Data from analysis of traces and routing

Research papers and technical reports

Contact Info