Detecting network outages using different sources of data

TMA Experts Summit, Paris, France

Cristel Pelsser
University of Strasbourg / ICube

June, 2019

Some perspective on:

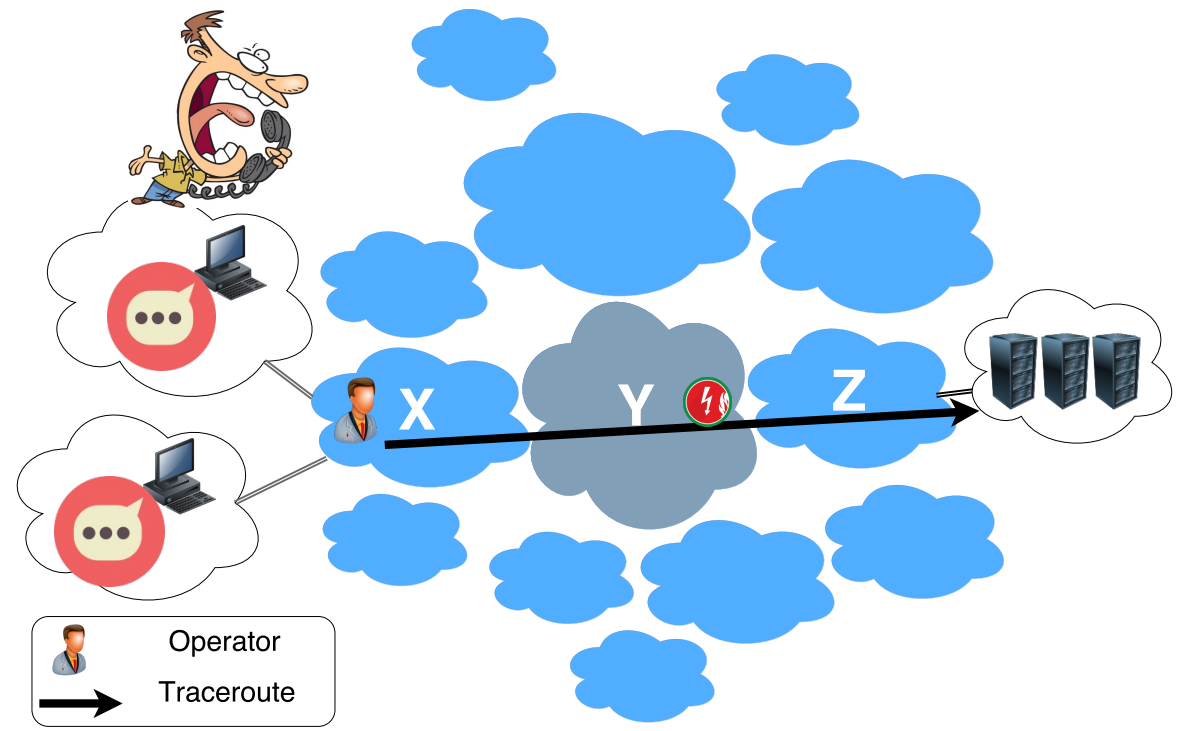
- From unsollicited traffic

  Detecting Outages using Internet Background Radiation. Andréas Guillot (U. Strasbourg), Romain Fontugne (IIJ), Philipp Winter (CAIDA), Pascal Mérindol (U. Strasbourg), Alistair King (CAIDA), Alberto Dainotti (CAIDA), Cristel Pelsser (U. Strasbourg). TMA 2019.
- From highly distributed permanent TCP connections

  Disco: Fast, Good, and Cheap Outage Detection. Anant Shah (Colorado State U.), Romain Fontugne (IIJ), Emile Aben (RIPE NCC), Cristel Pelsser (University of Strasbourg), Randy Bush (IIJ, Arrcus). TMA 2017.
- From large-scale traceroute measurements

  Pinpointing Anomalies in Large-Scale Traceroute Measurements. Romain Fontugne (IIJ), Emile Aben (RIPE NCC), Cristel Pelsser (University of Strasbourg), Randy Bush (IIJ, Arrcus). IMC 2017.

Understanding Internet health? (Motivation)



- To speedup failure identification and thus recovery
- To identify weak areas and thus guide network design

Understanding Internet health? (Problem 1)

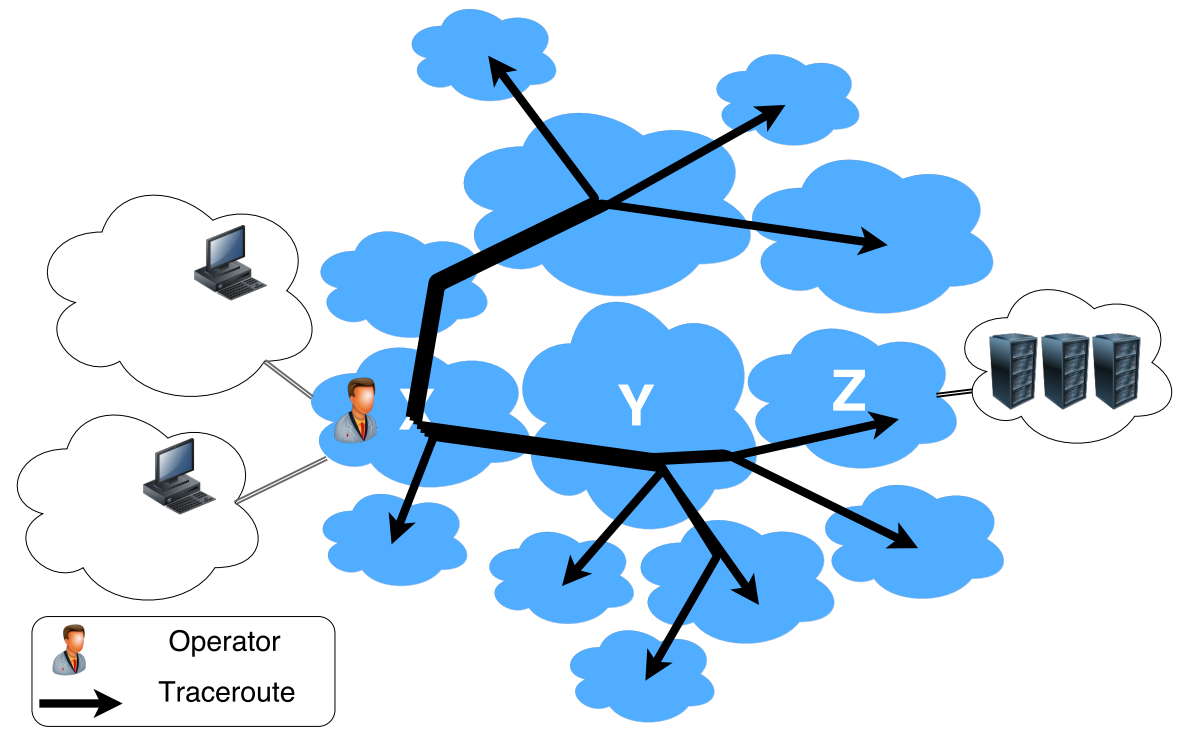
Manual observations and operations

- Traceroute / Ping / Operators' group mailing lists
- Time consuming
- Slow process
- Small visibility

→ **Our goal: Automaticaly pinpoint network disruptions (i.e. congestion and network disconnections)**

Understanding Internet health? (Problem 2)

A single viewpoint is not enough



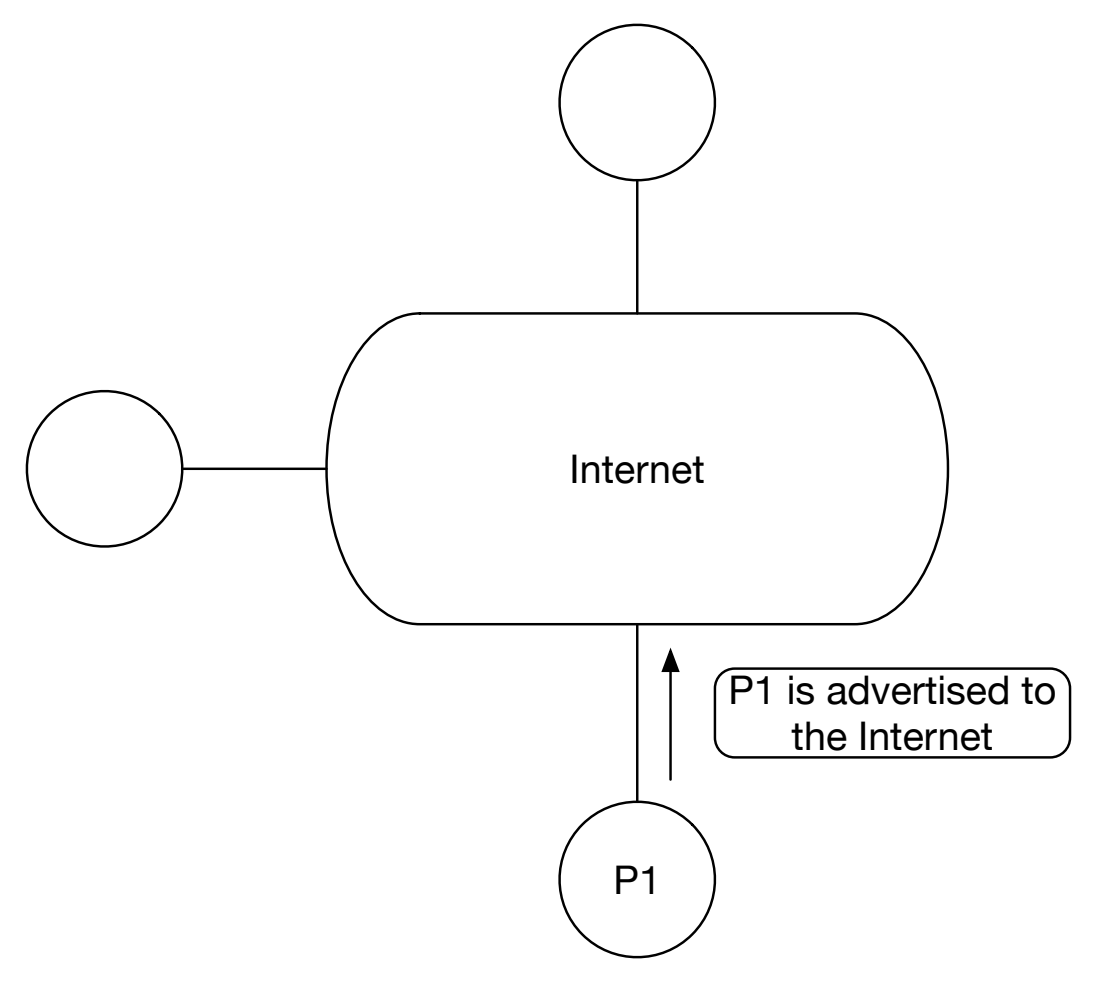
→ **Our goal: mine results from deployed platforms**

→ Cooperative and distributed approach

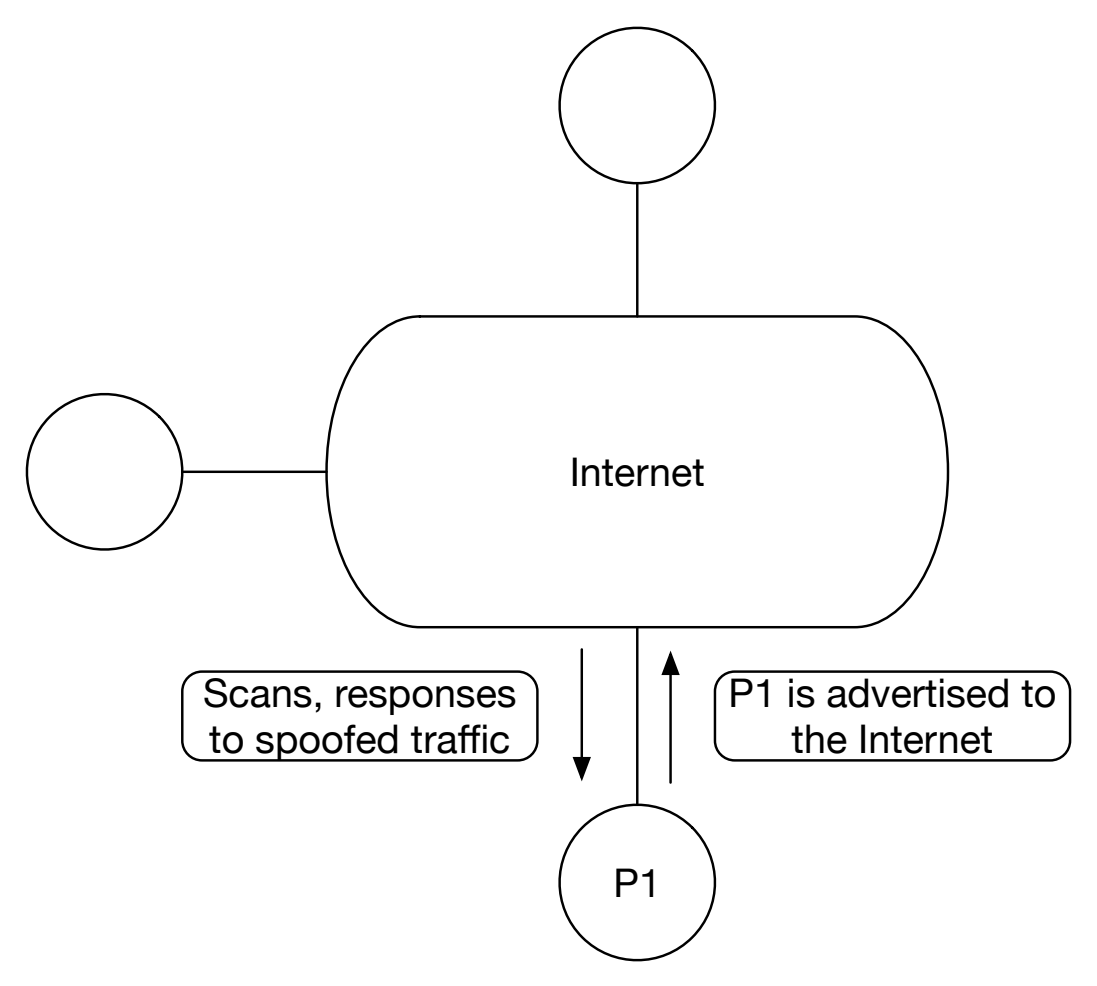
→ Using existing data, no added burden to the network

Outage detection from unsollicited traffic

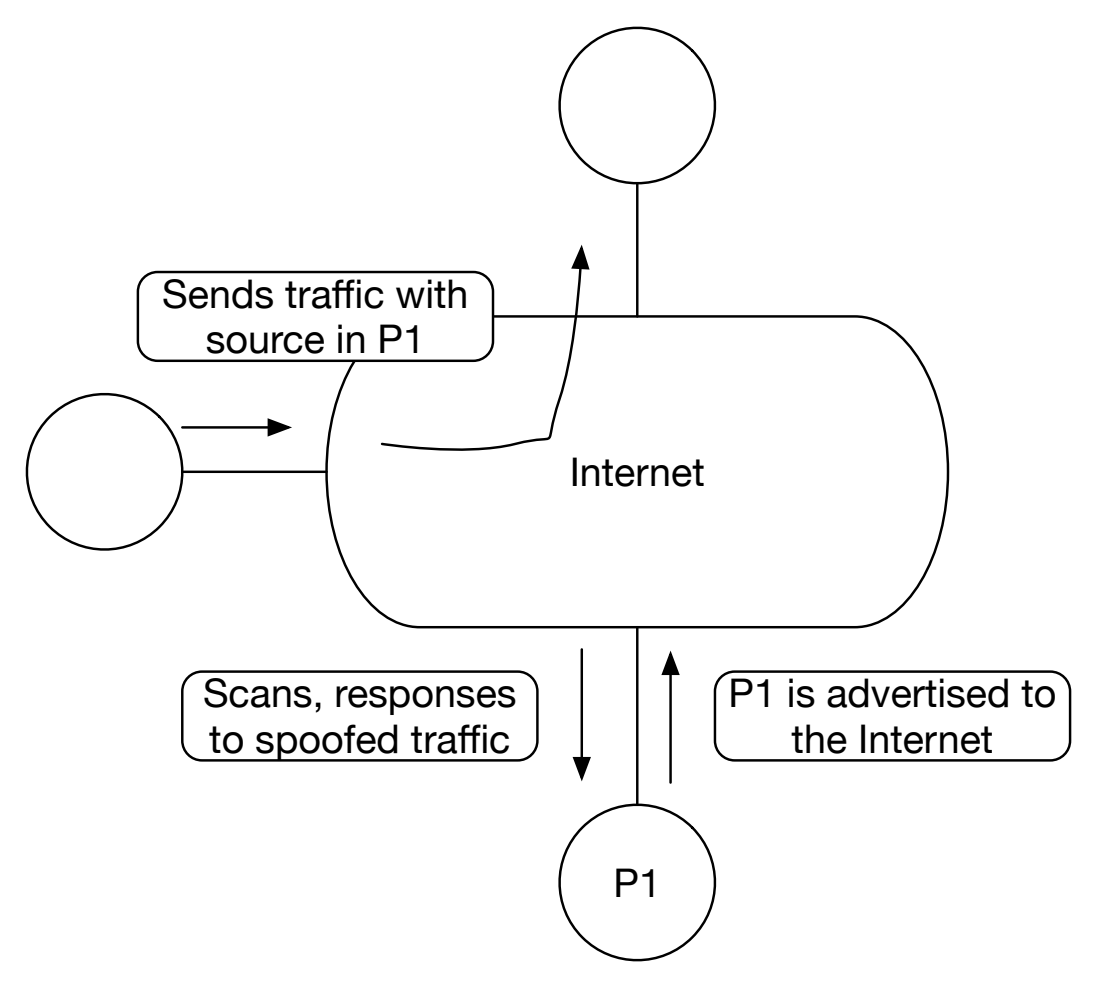
Dataset: Internet Background Radiation



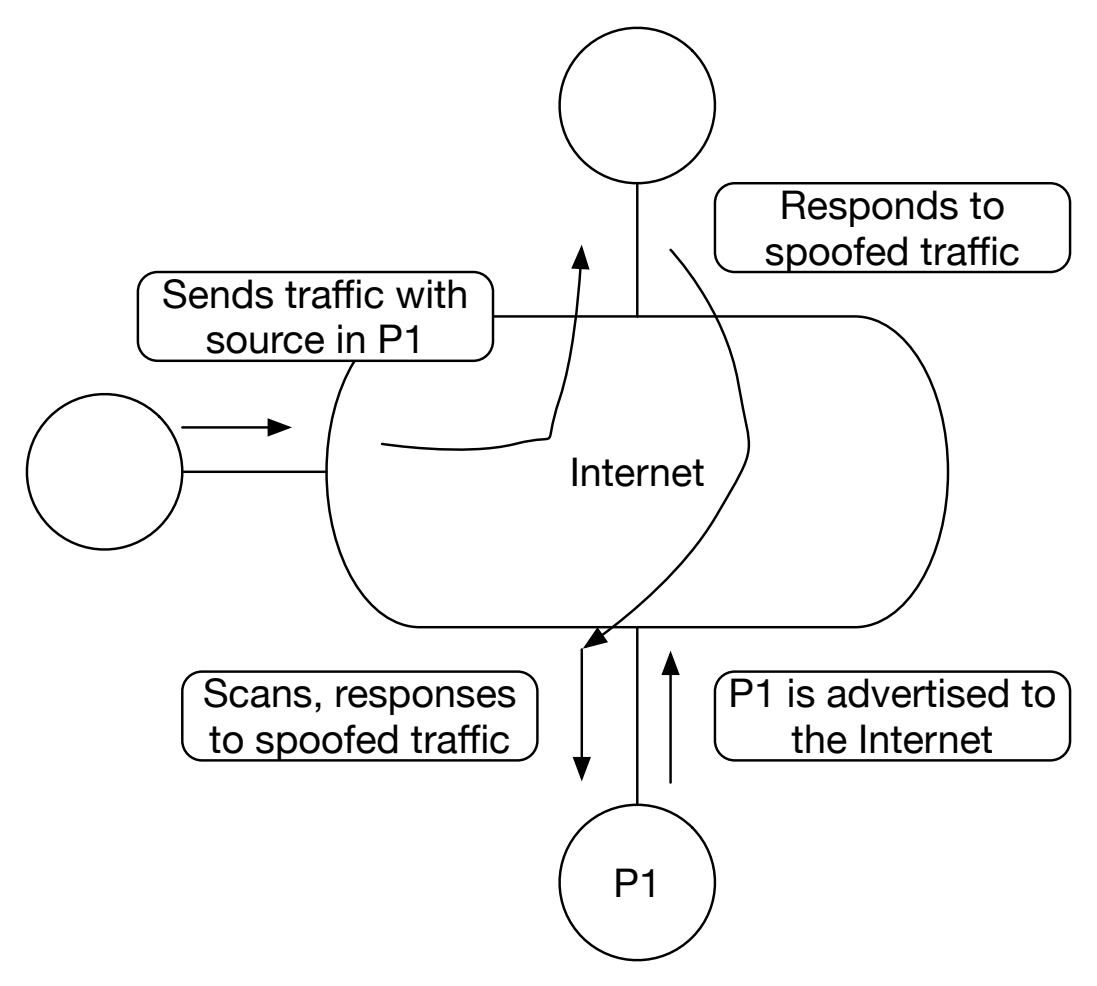
Dataset: Internet Background Radiation



Spoofed traffic



Spoofed traffic



Dataset: IP count time-series (per country or AS)

Use cases: Attacks, Censorship, Local outages detection

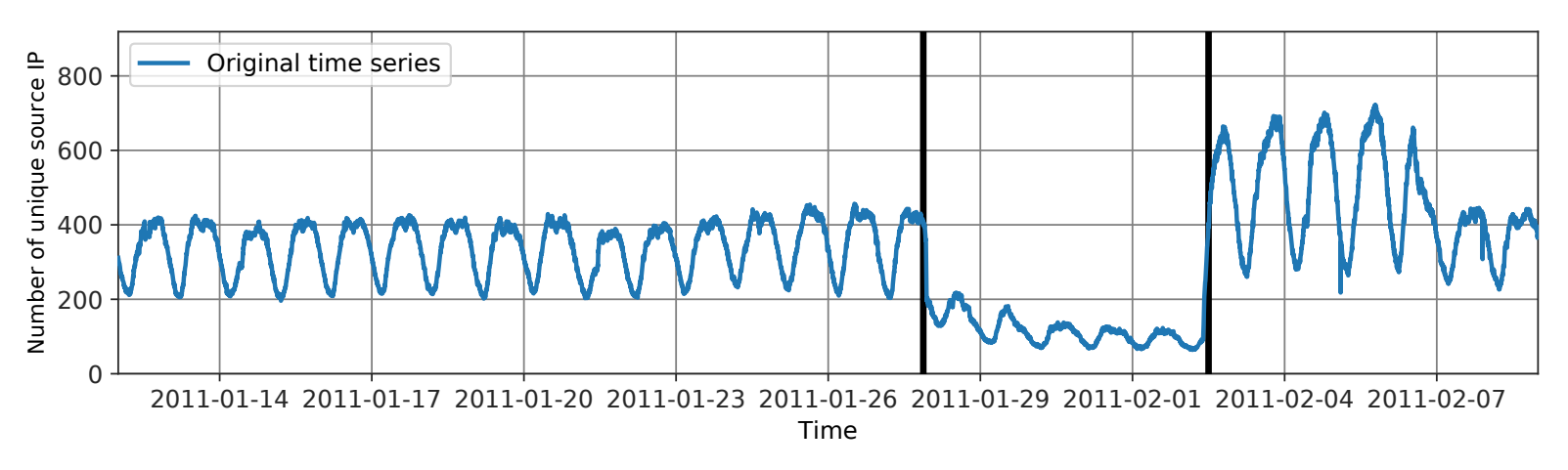
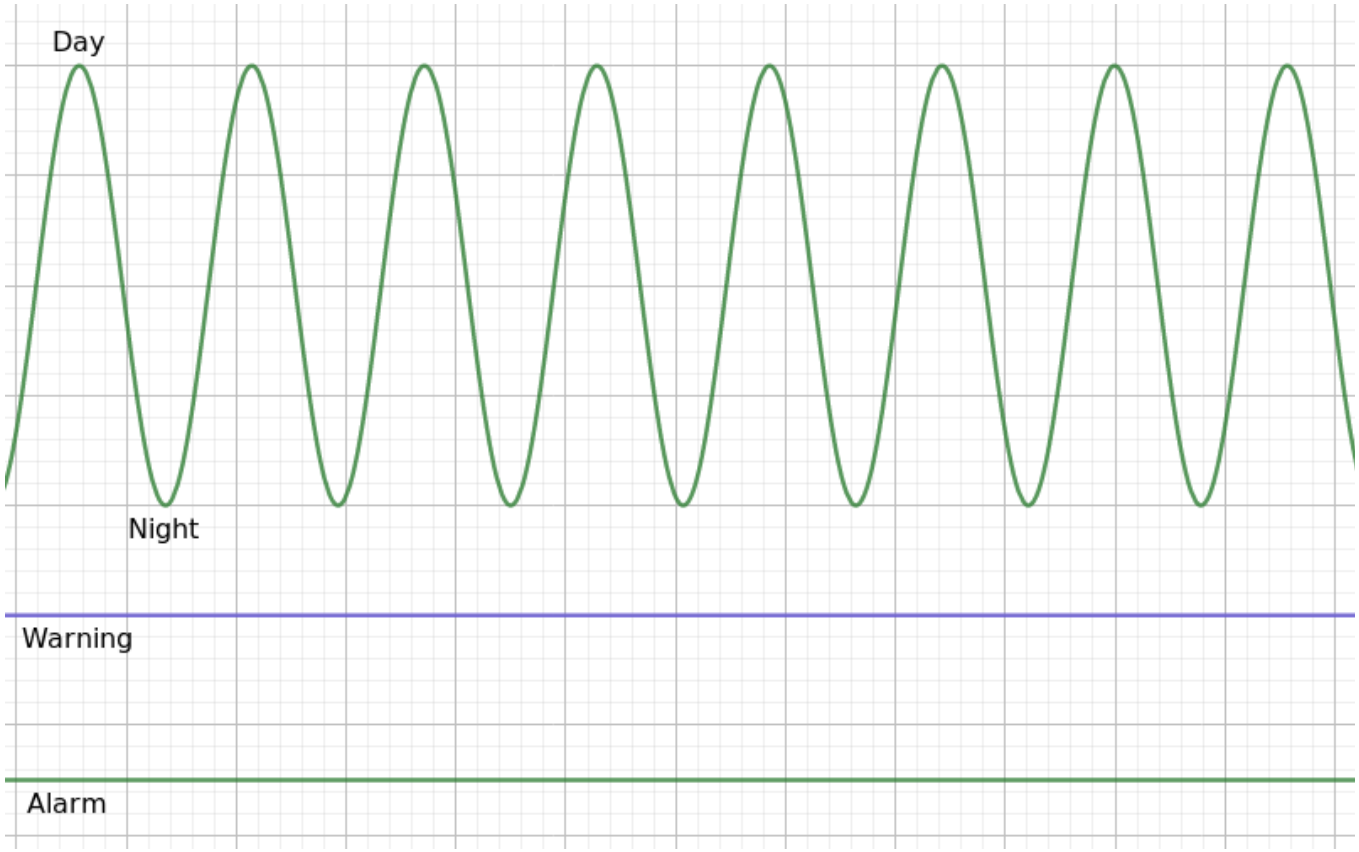


Figure 1: Egyptian revolution

⇒ More than 60 000 time series in the CAIDA telescope data.

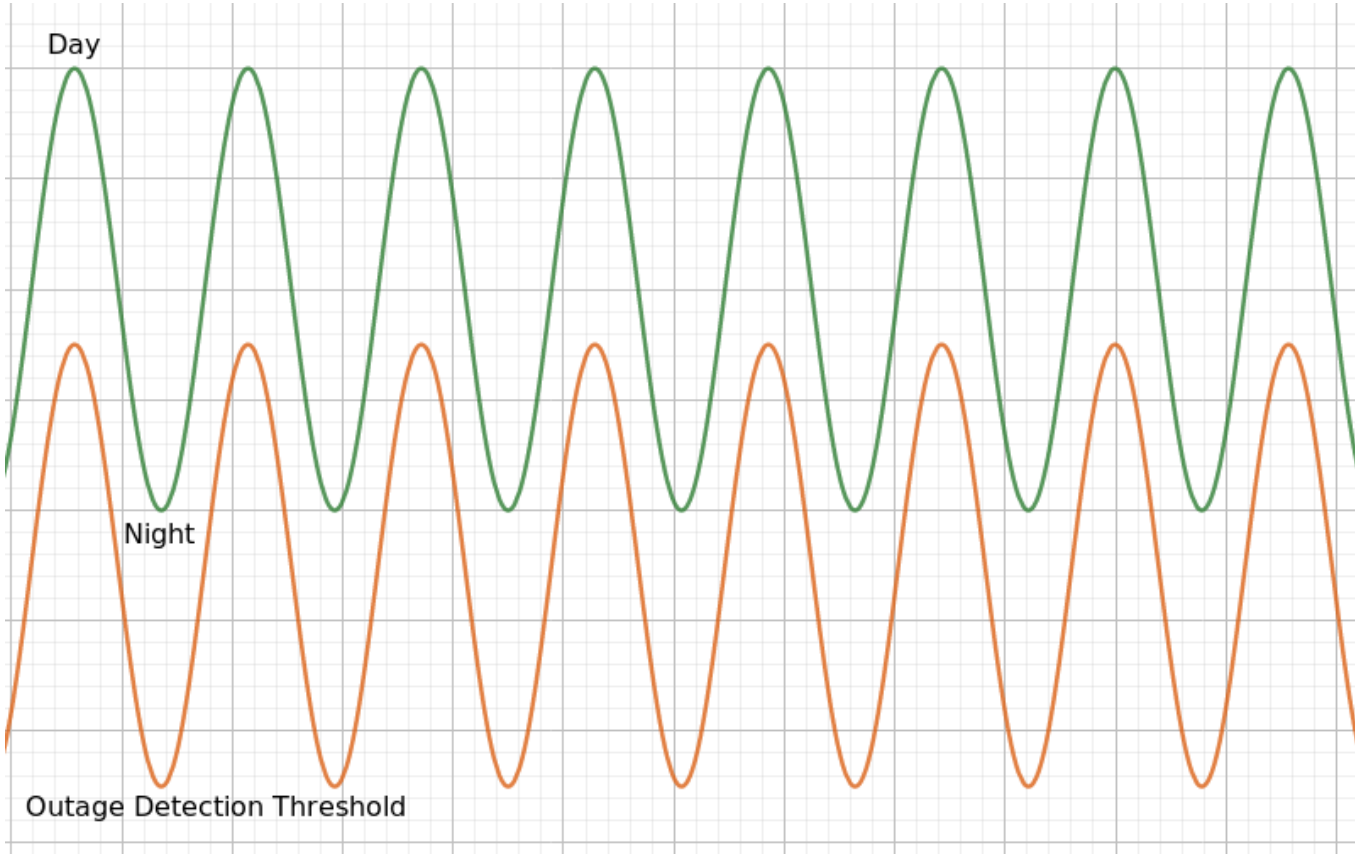
We use drops in the time series are indicators of an outage.

Current methodology used by IODA



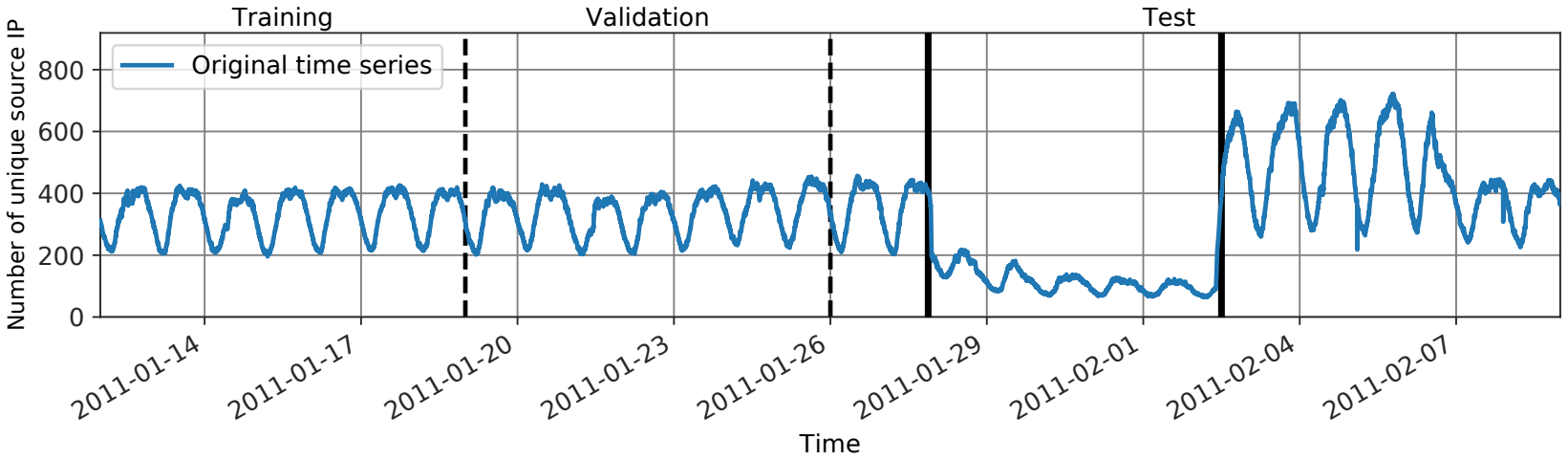
Detecting outages using **fixed thresholds**

Our goal

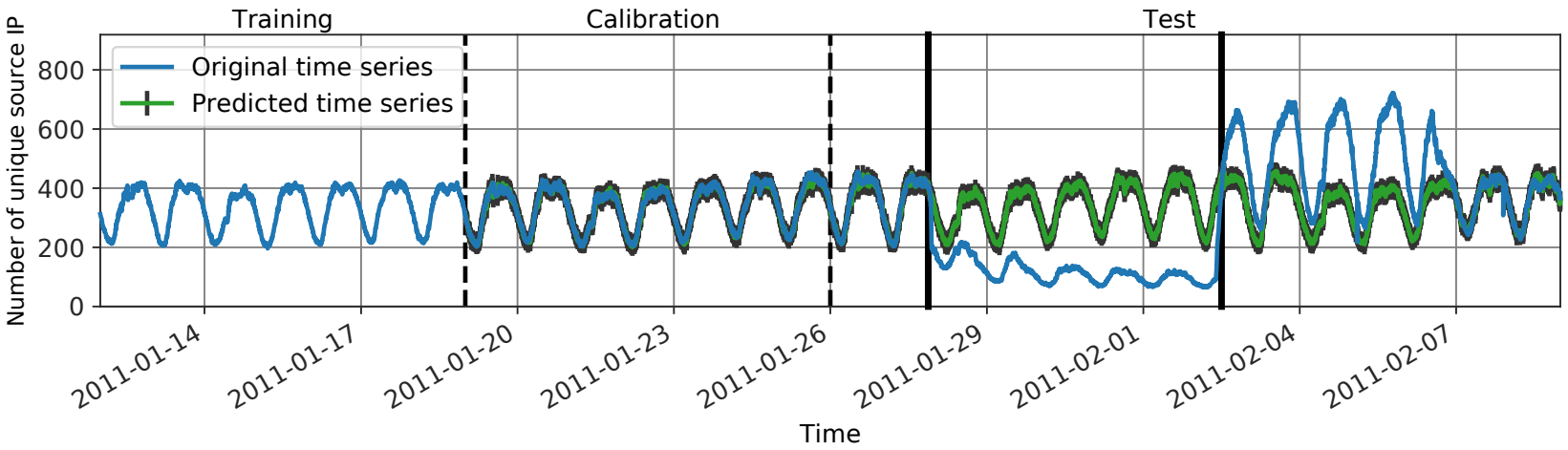


Detecting outages using **dynamic thresholds**

Outage detection process

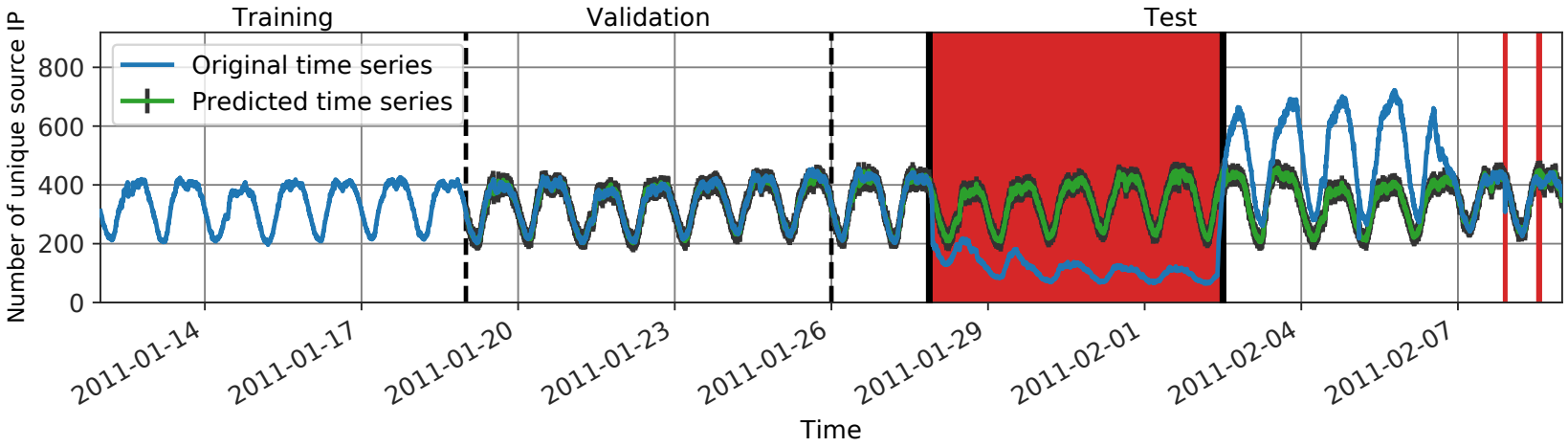


Outage detection process



Prediction and confidence interval

Outage detection process



- When the real data is outside the prediction interval, we raise an alarm.
- We want a prediction model that is robust to the seasonality and noise in the data $\rightarrow$ We use the SARIMA model[1].

[1] More details on the methodology on wednesday.

Validation: ground truth

Characteristics

- 130 known outages
- Multiple spatial scales
  - Countries
  - Regions
  - Autonomous Systems
- Multiple durations (from an hour to a week)
- Multiple causes (intentional or non intentional)

Evaluating our solution

Objectives

- Identifying the minimal number of IP addresses
- Identifying a good *threshold*

Threshold

- TPR of 90% and FPR of 2%

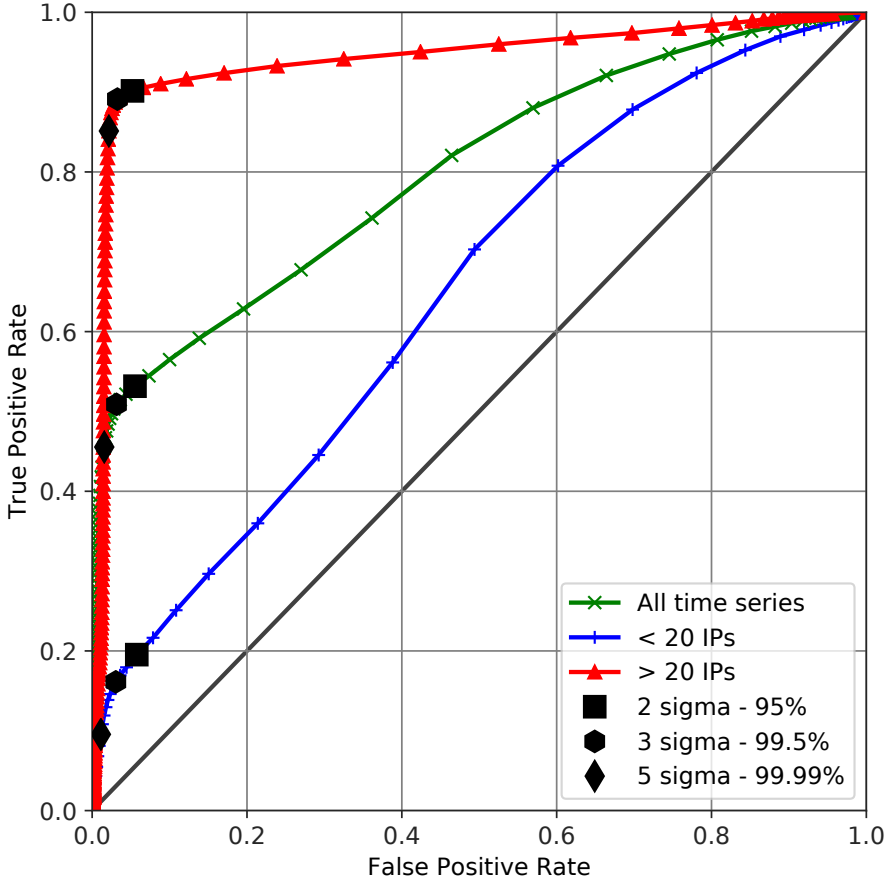
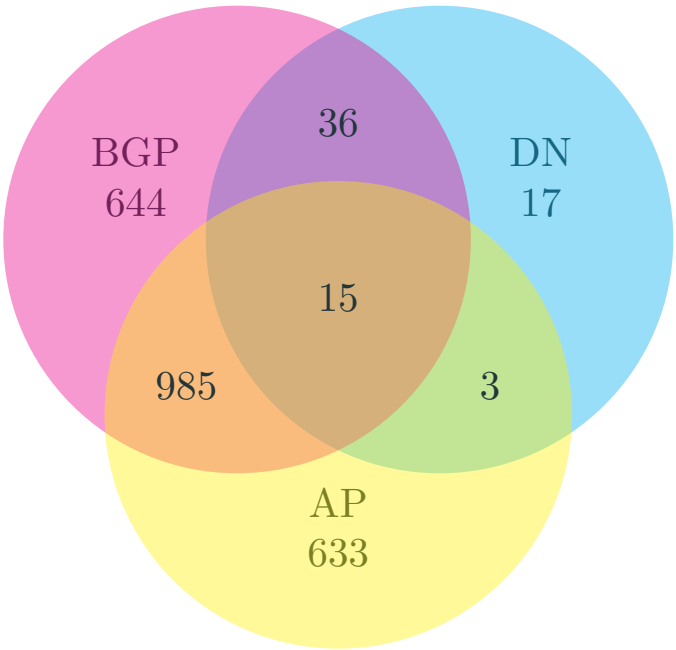
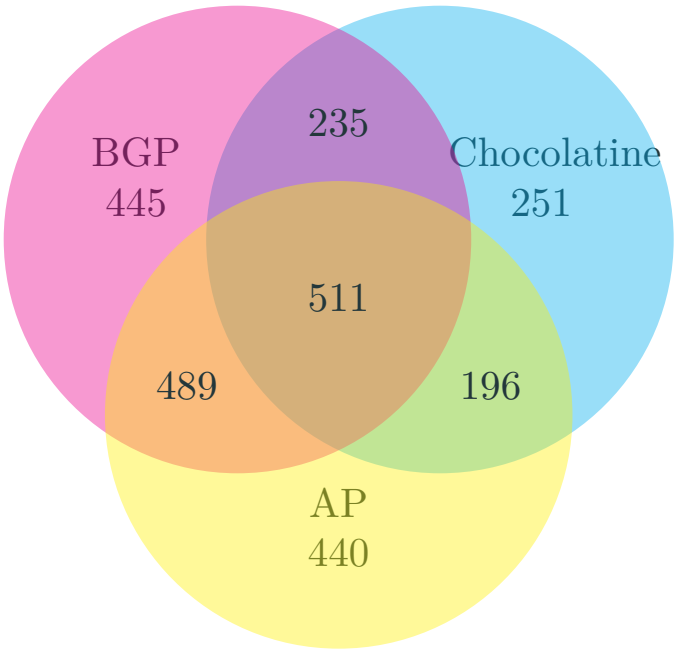


Figure 2: ROC curve

Comparing our proposal (Chocolatine) to CAIDA's tools

- More events detected than the simplistic thresholding technique (DN)
- Higher overlap with other detection techniques
- Not a complete overlap
  → difference in dataset coverage
  → different sensitivities to outages



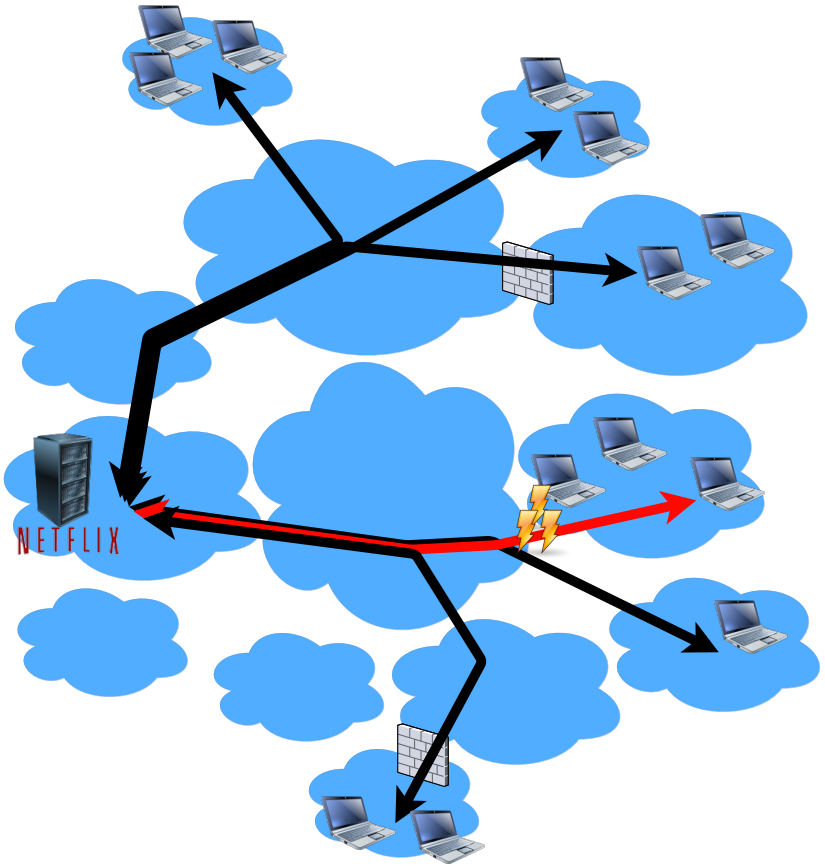


Outage detection from highly distributed permanent TCP connections

Proposed Approach

Disco:

- Monitor long-running TCP connections and synchronous disconnections from related network/area
- We apply Disco on RIPE Atlas data, where probes are widely distributed at the edge and behind NATs/CGNs providing visibility Trinocular may not have



→ **Outage = synchronous disconnections from the same topological/geographical area**

Assumptions / Design Choices

Rely on TCP disconnects

- Hence the granularity of detection is dependent on TCP timeouts

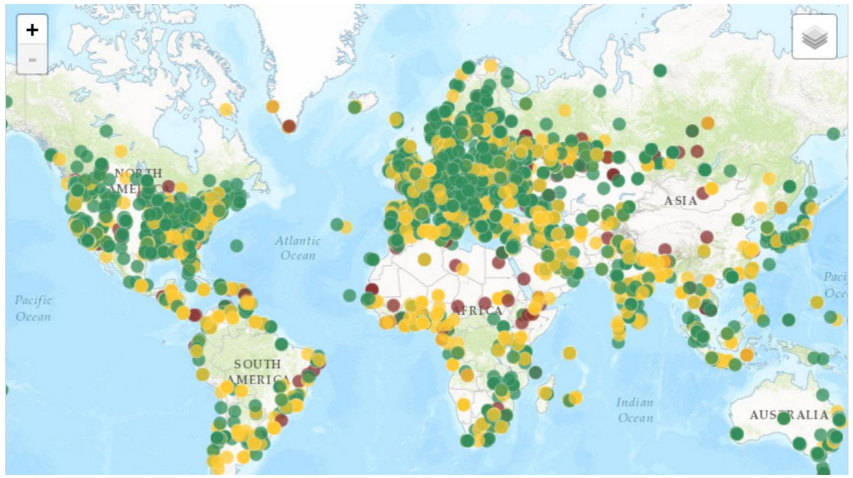
Bursts of disconnections are indicators of interesting outage

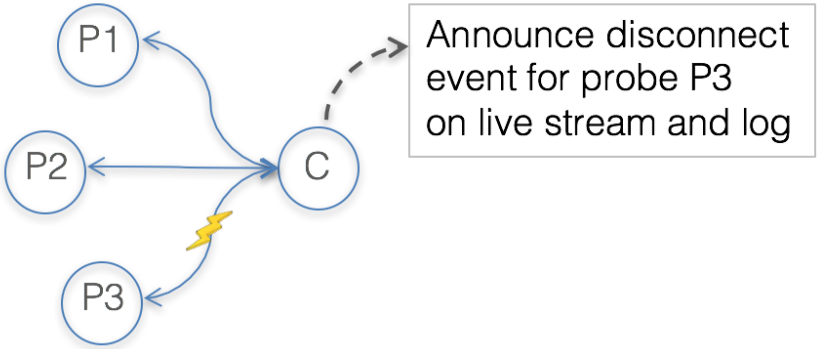
- While there might be non bursty outages that are interesting, Disco is designed to detect large synchronous disconnections

Proposed System: Disco & Atlas

RIPE Atlas platform

- 10k probes worldwide
- Persistent connections with RIPE controllers
- Continuous traceroute measurements (see outages from inside)

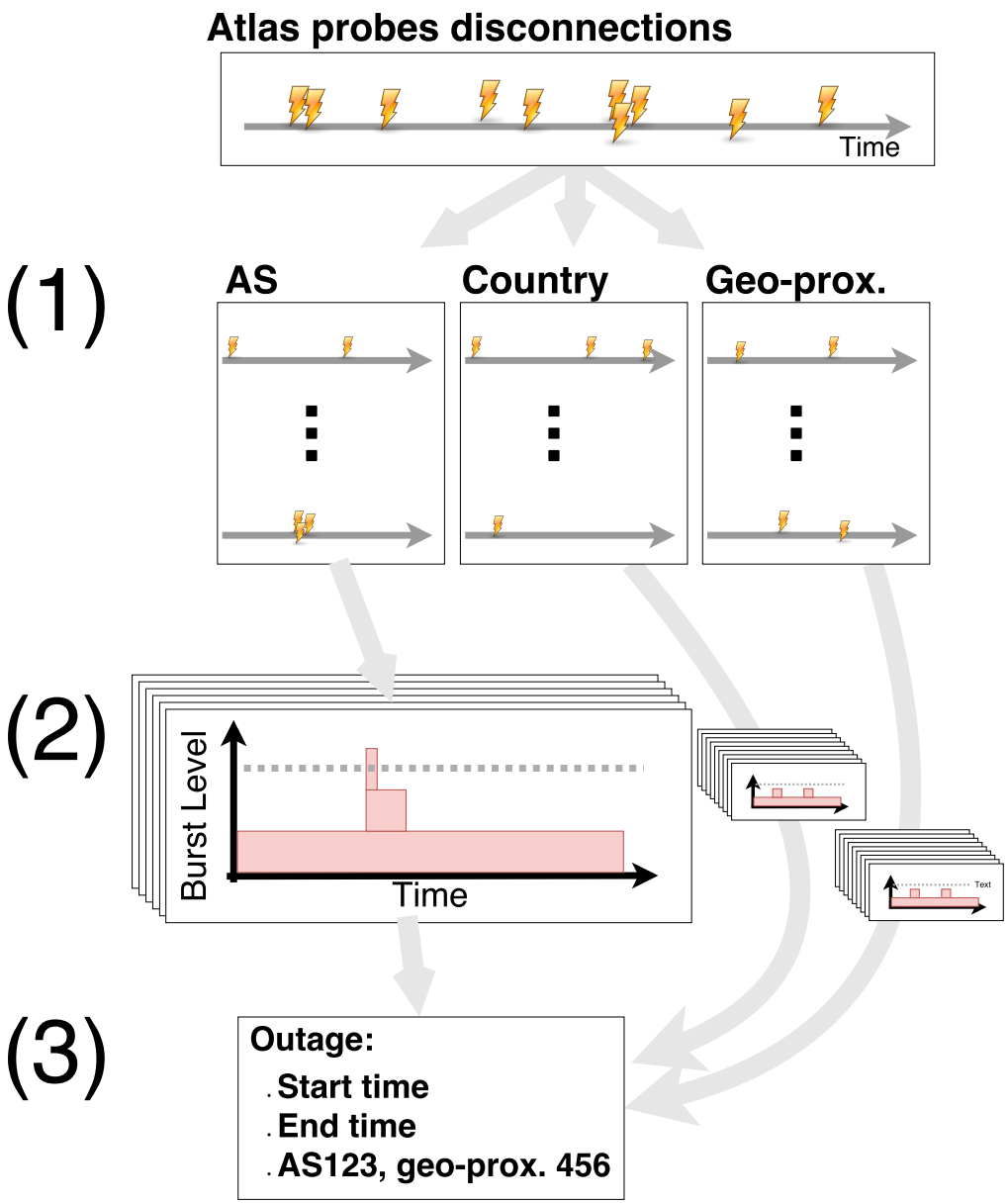




→ **Dataset: Stream of probe connection/disconnections (from 2011 to 2016)**

Disco Overview

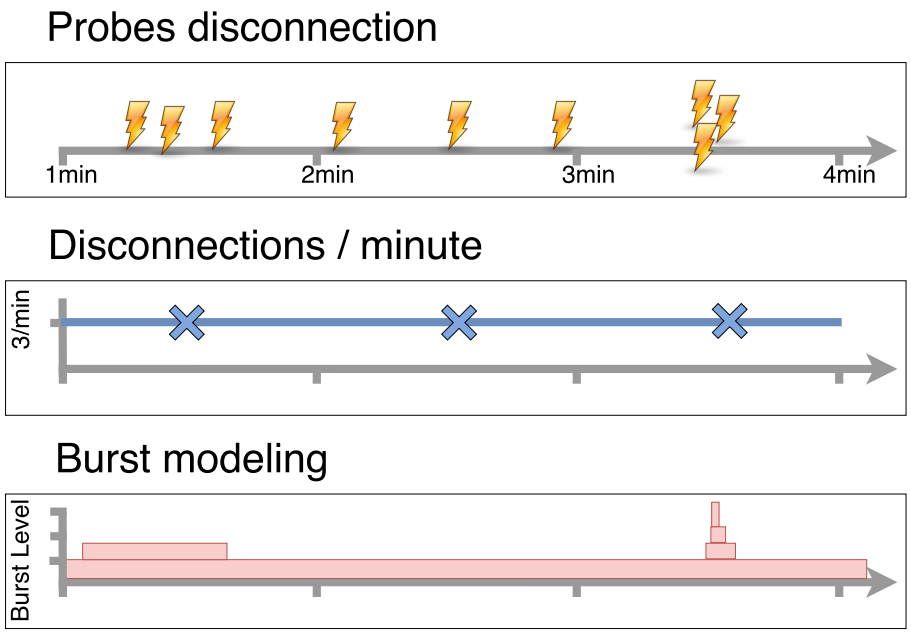
1. Split disconnection stream in sub-streams (AS, country, geo-proximate 50km radius)
2. Burst modeling and outage detection
3. Aggregation and outage reporting



Why Burst Modeling?

Goal: How to find synchronous disconnections?

- Time series conceal temporal characteristics
- Burst model estimates disconnections arrival rate at any time

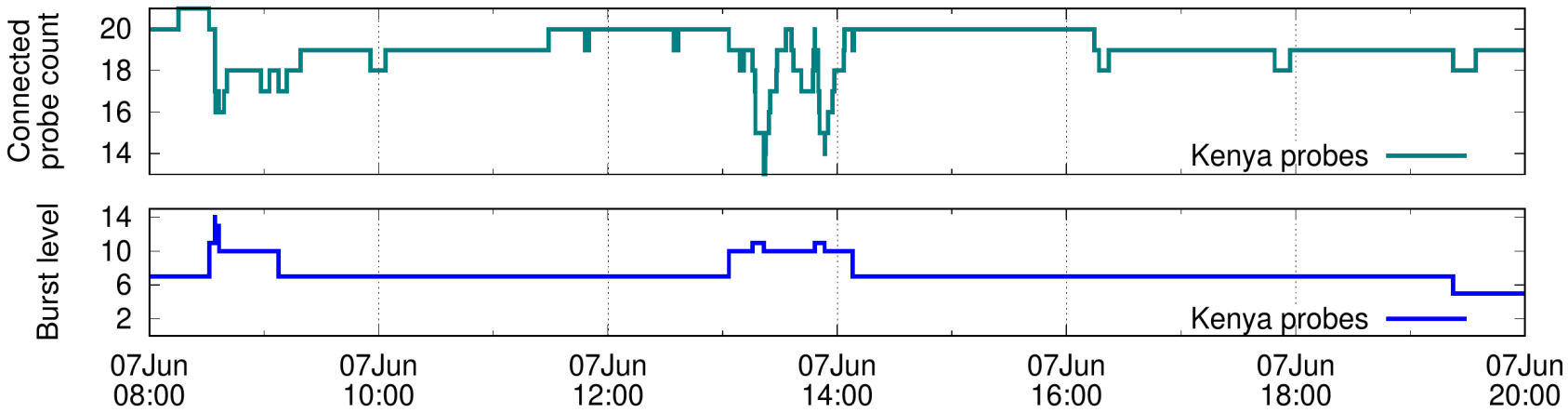


Implementation: Kleinberg burst model[2]

[2] J. Kleinberg. "Bursty and hierarchical structure in streams", Data Mining and Knowledge Discovery, 2003.

Burst modeling: Example

- Monkey causes blackout in Kenya at 8:30 UTC June, 7th 2016
- Same day RIPE rebooted controllers



Results

Outage detection:

- Atlas probes disconnections from 2011 to 2016
- Disco found 443 significant outages

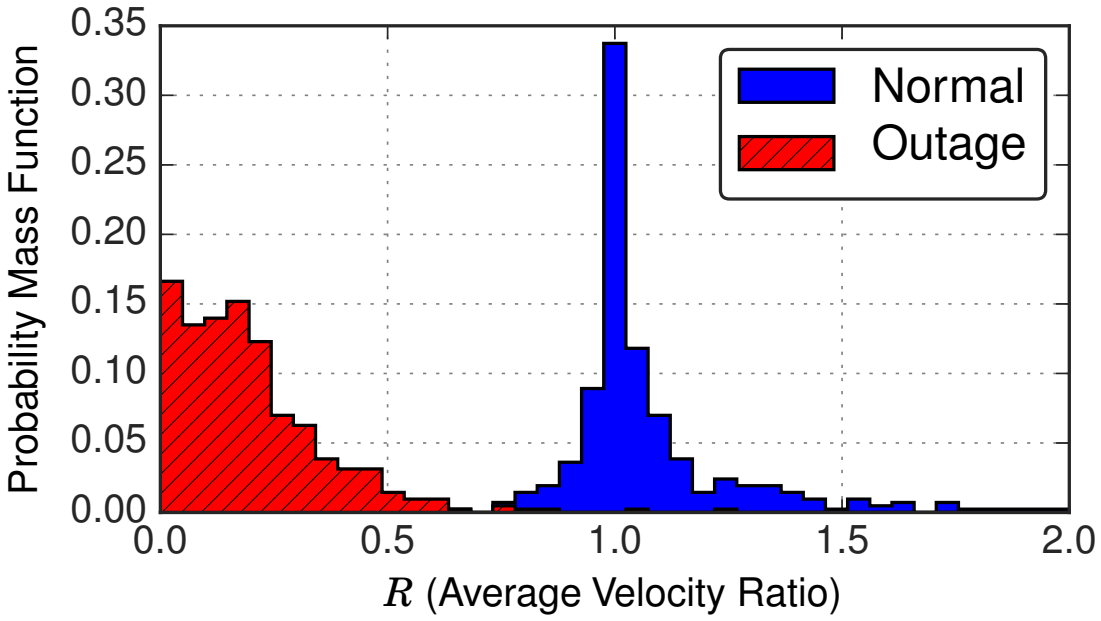
Outage characterization and validation:

- Traceroute results from probes (buffered if no connectivity)
- Outage detection results from Trinocular

Validation (Traceroute)

Comparison to traceroutes:

- Probes in detected outages can reach traceroutes destination?
  $\rightarrow$ Velocity ratio: proportion of completed traceroutes in given time



$\rightarrow$ **Velocity ratio ≤ 0.5 for 95% of detected outages**

Validation (Trinocular)

Comparison to Trinocular (2015):

- Disco found 53 outages in 2015
- Corresponding to 851 /24s (only 43% is responsive to ICMP)

Results for /24s reported by Disco and pinged by Trinocular:

- 33/53 are also found by Trinocular
- 9/53 are missed by Trinocular (avg time of outages $<$ 1hr)
- Other outages are partially detected by Trinocular

23 outages found by Trinocular are missed by Disco

- Disconnections are not very bursty in these cases

$\rightarrow$ **Disco's precision: 95%, recall: 67%**

Outage detection from large-scale traceroute measurements

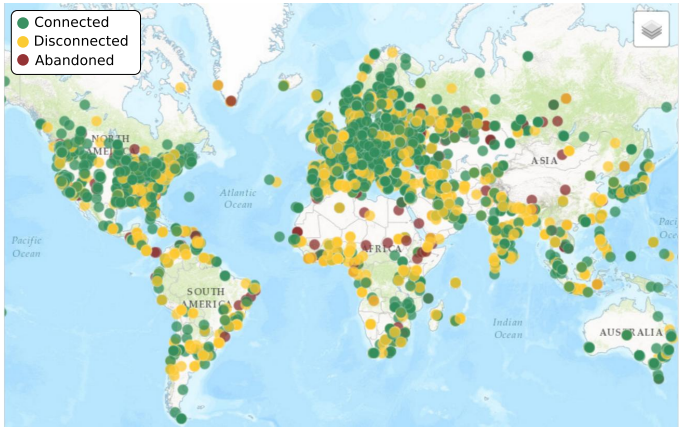
Dataset: RIPE Atlas traceroutes

Two repetitive large-scale measurements

- *Builtin*: traceroute every 30 minutes to all DNS root servers ($\approx$ 500 server instances)
- *Anchoring*: traceroute every 15 minutes to 189 collaborative servers

Analyzed dataset

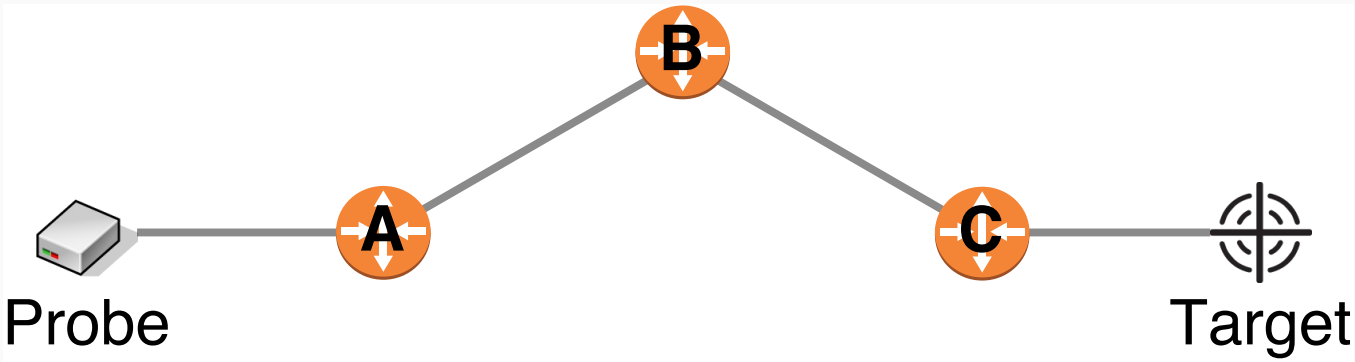
- May to December 2015
- 2.8 billion IPv4 traceroutes
- 1.2 billion IPv6 traceroutes



Monitor delays with traceroute?

Traceroute to "www.target.com"

```
~$ traceroute www.target.com
traceroute to target, 30 hops max, 60 byte packets
 1  A       0.775 ms  0.779 ms  0.874 ms
 2  B       0.351 ms  0.365 ms  0.364 ms
 3  C       2.833 ms  3.201 ms  3.546 ms
 4  Target  3.447 ms  3.863 ms  3.872 ms
```



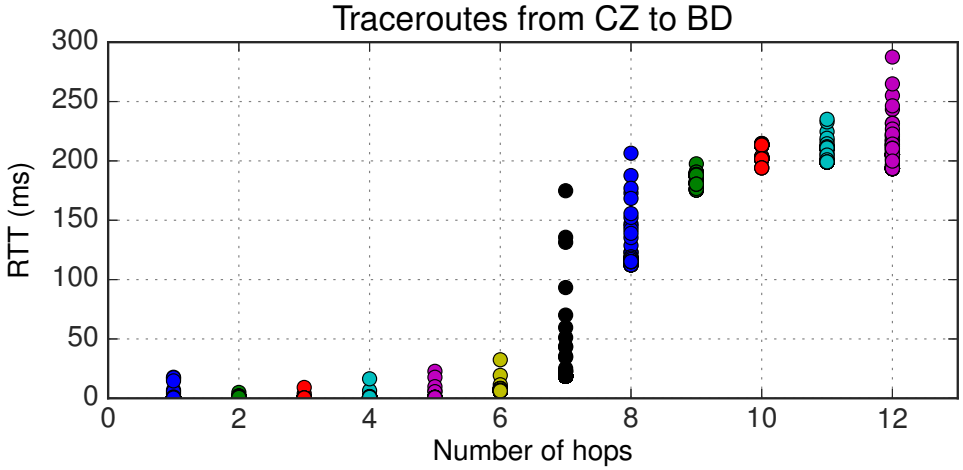
Round Trip Time (RTT) between B and C?

Report abnormal RTT between B and C?

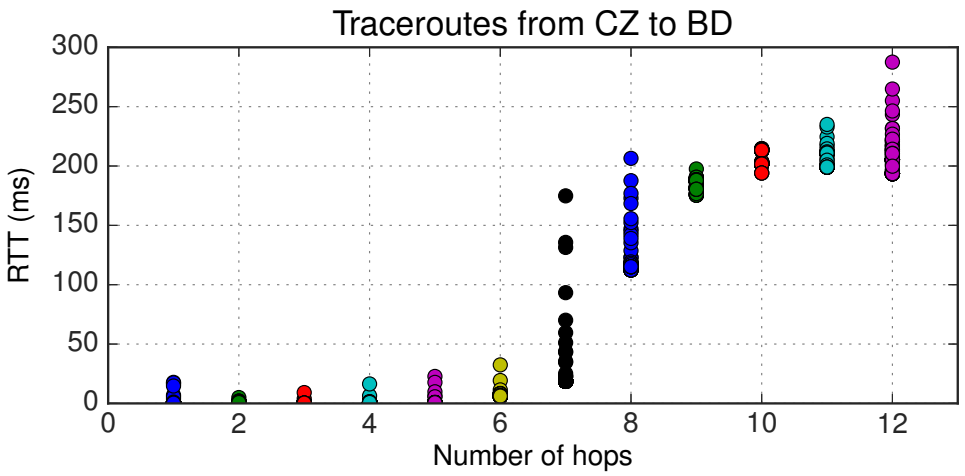
Monitor delays with traceroute?

Challenges:

- Noisy data

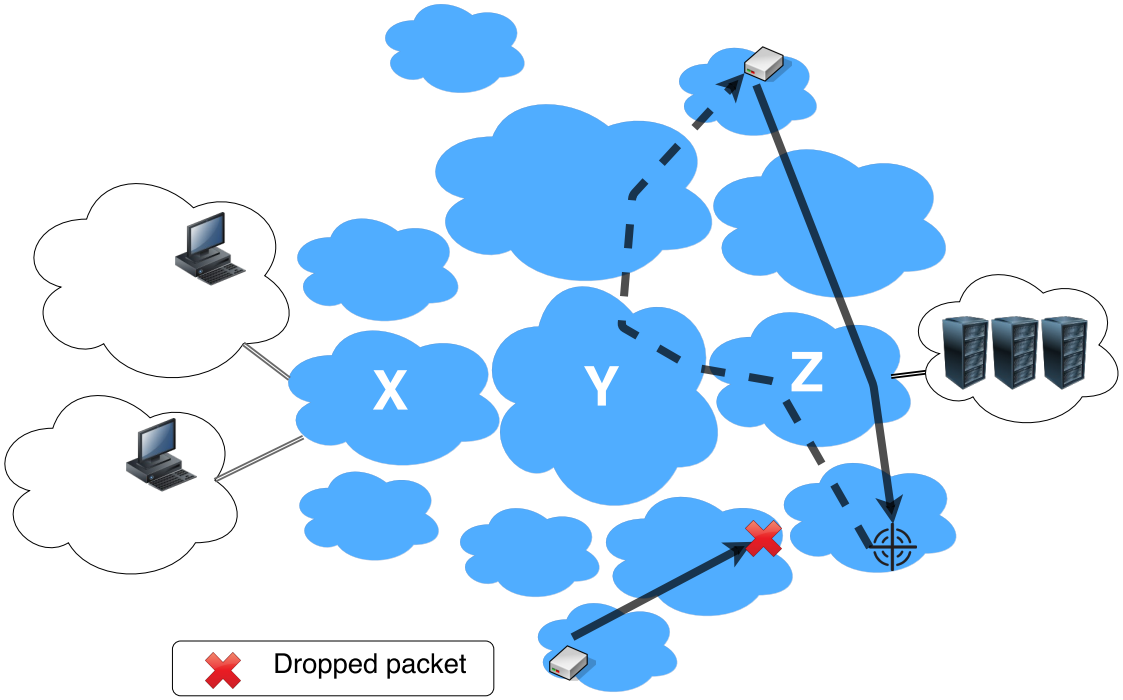


Monitor delays with traceroute?



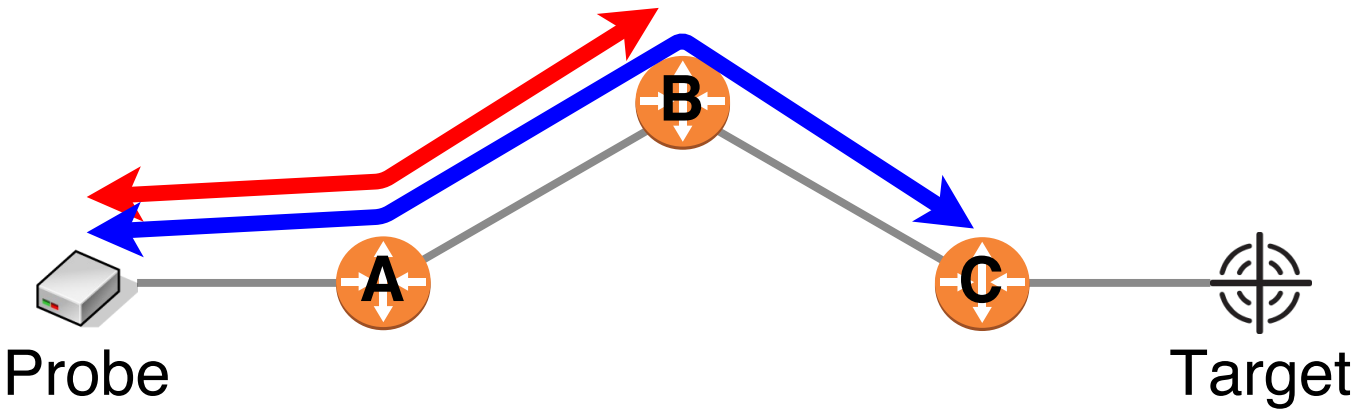
Challenges:

- Noisy data
- Traffic asymmetry



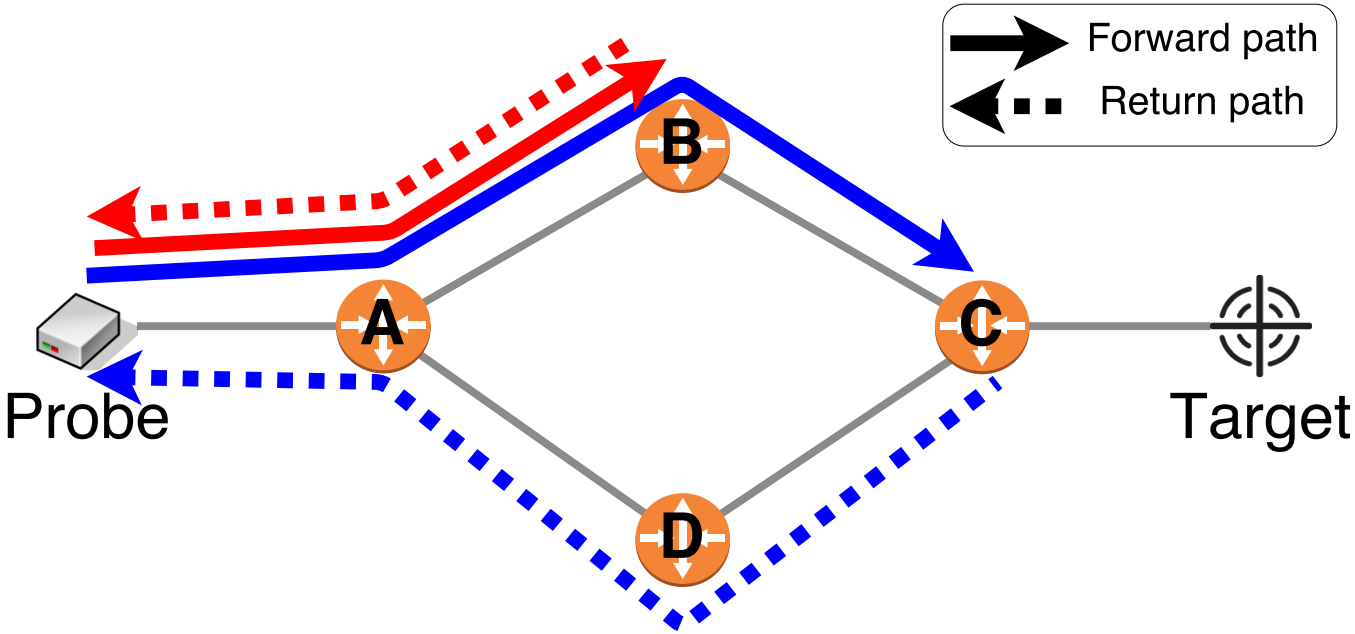
What is the RTT between B and C?

```
~$ traceroute www.target.com
traceroute to target, 30 hops max, 60 byte packets
 1  A       0.775 ms  0.779 ms  0.874 ms
 2  B       0.351 ms  0.365 ms  0.364 ms
 3  C       2.833 ms  3.201 ms  3.546 ms
 4  Target  3.447 ms  3.863 ms  3.872 ms
```



$$RTT_C - RTT_B = RTT_{CB}?$$

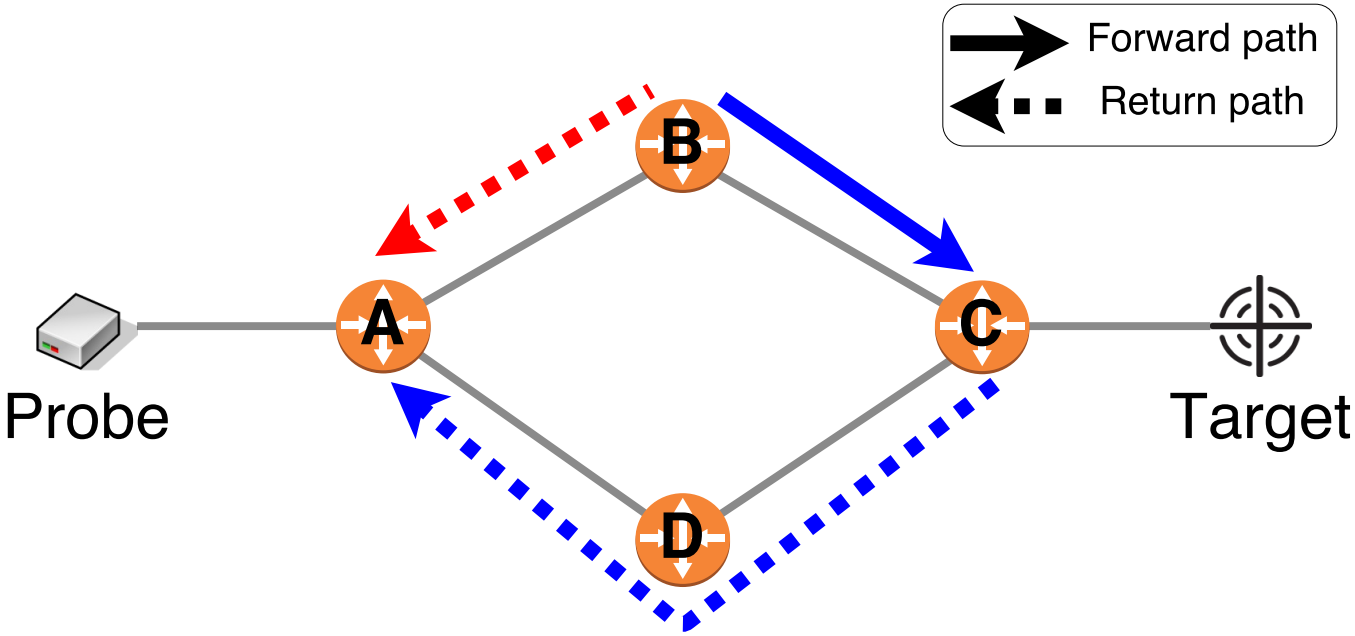
What is the RTT between B and C?



RTT_C - $RTT_B = RTT_{CB}$?

- No!
- Traffic is asymmetric
- RTT_B and RTT_C take **different return paths!**

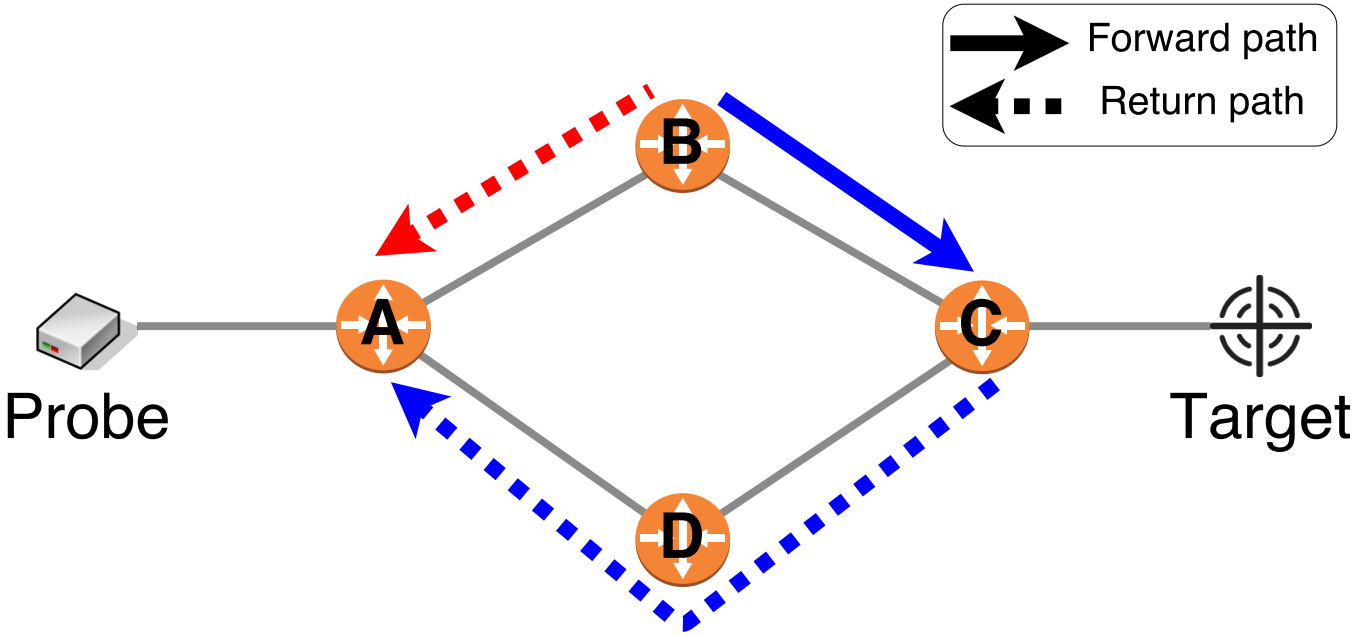
What is the RTT between B and C?



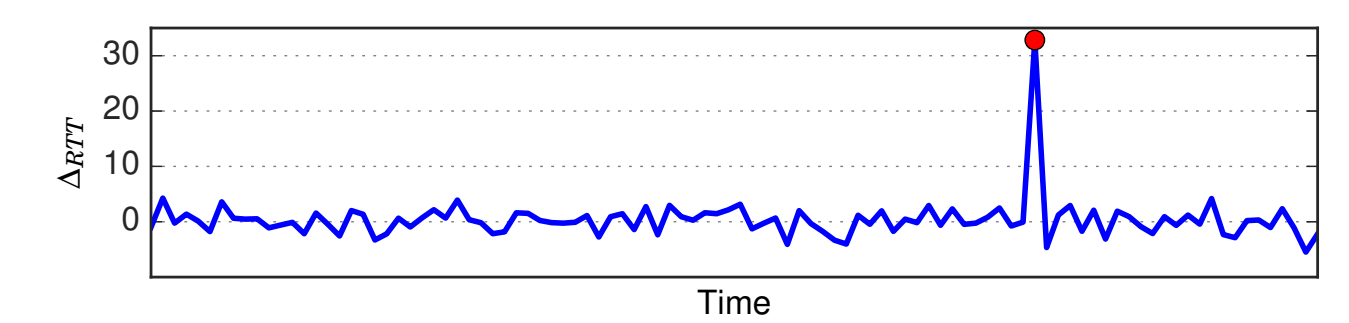
RTT_C - RTT_B = RTT_{CB}?

- No!
- Traffic is asymmetric
- RTT_B and RTT_C take **different return paths!**
- **Differential RTT**: $\Delta_{CB} = RTT_C - RTT_B = d_{BC} + e_p$

Problem with differential RTT



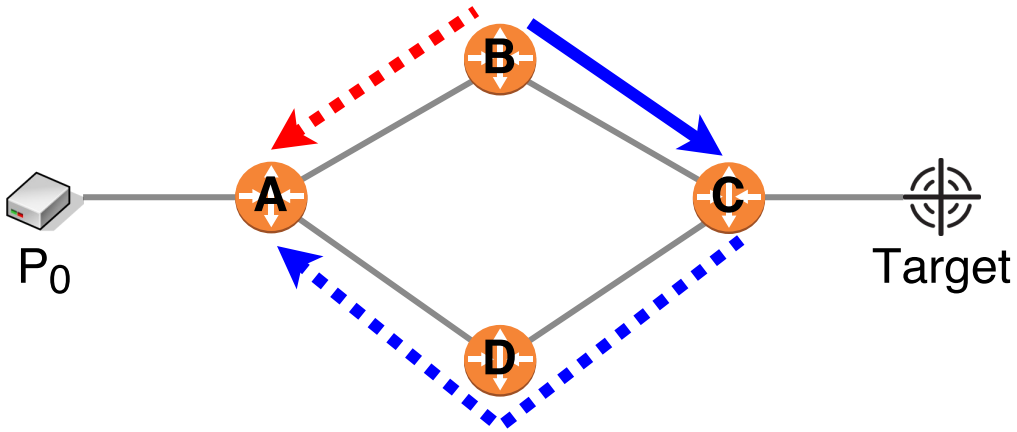
Monitoring Δ_{CB} over time:



→ Delay change on BC? CD? DA? BA???

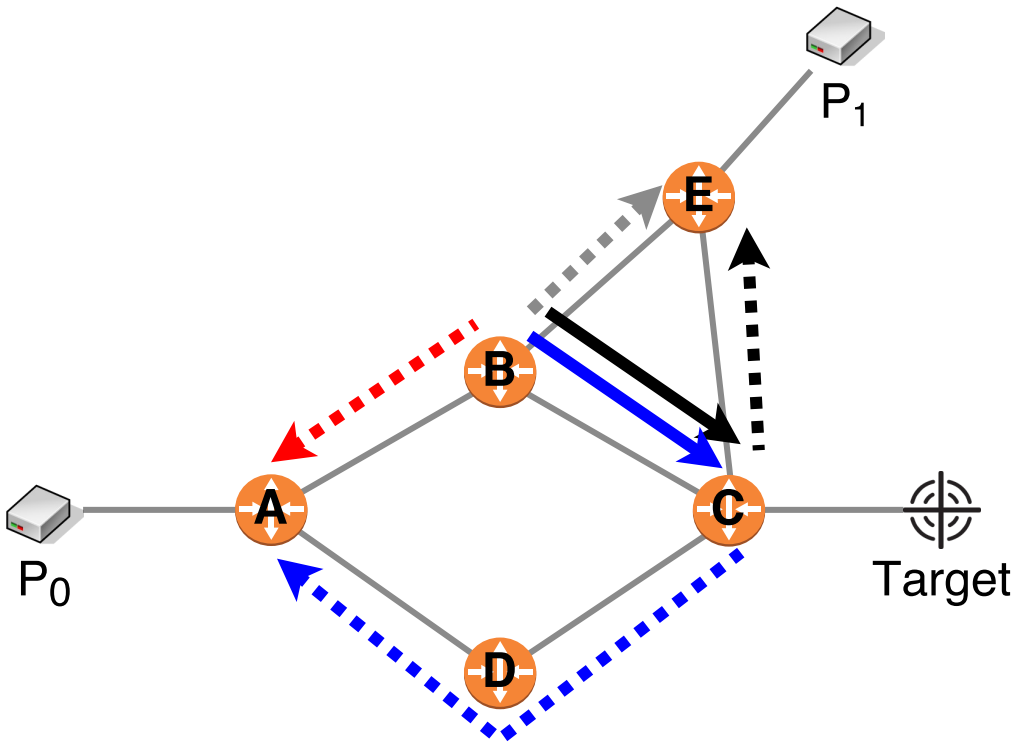
Proposed Approach: Use probes with different return paths

Differential RTT: $\Delta_{CB} = x_0$



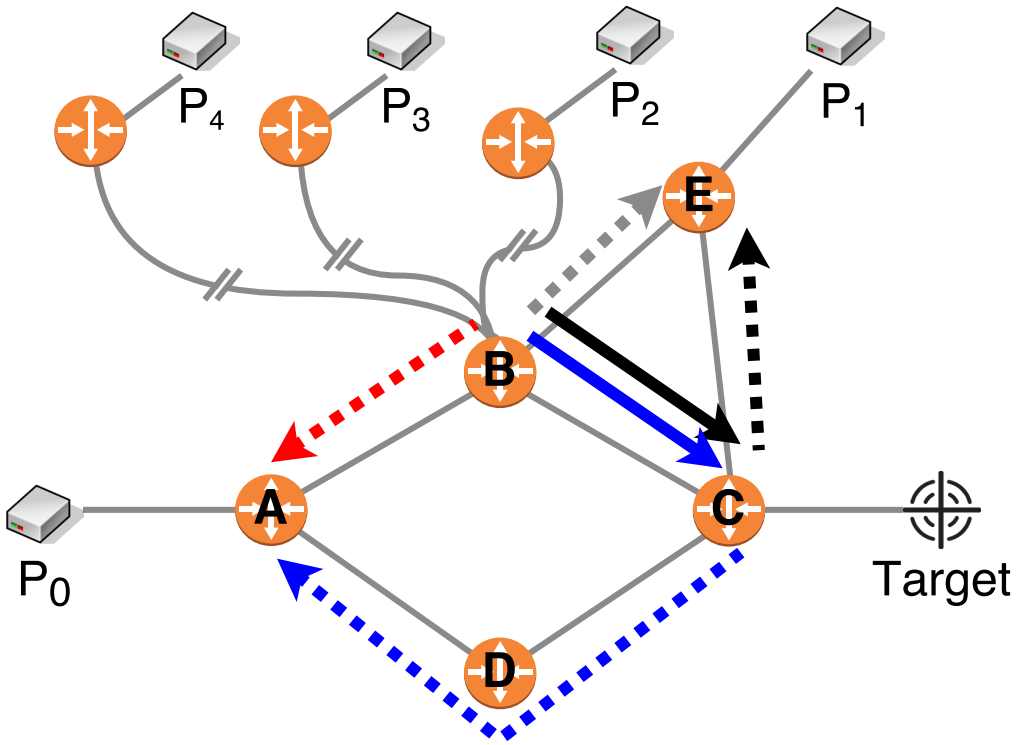
Proposed Approach: Use probes with different return paths

Differential RTT: $\Delta_{CB} = \{x_0, x_1\}$



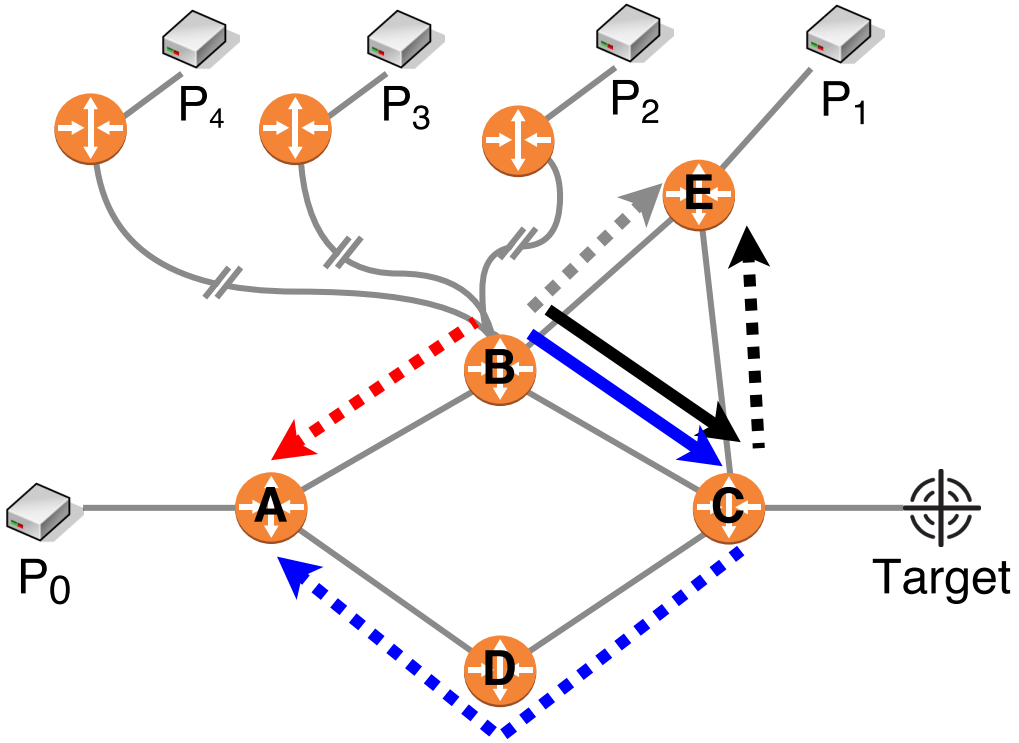
Proposed Approach: Use probes with different return paths

Differential RTT: $\Delta_{CB} = \{x_0, x_1, x_2, x_3, x_4\}$



Proposed Approach: Use probes with different return paths

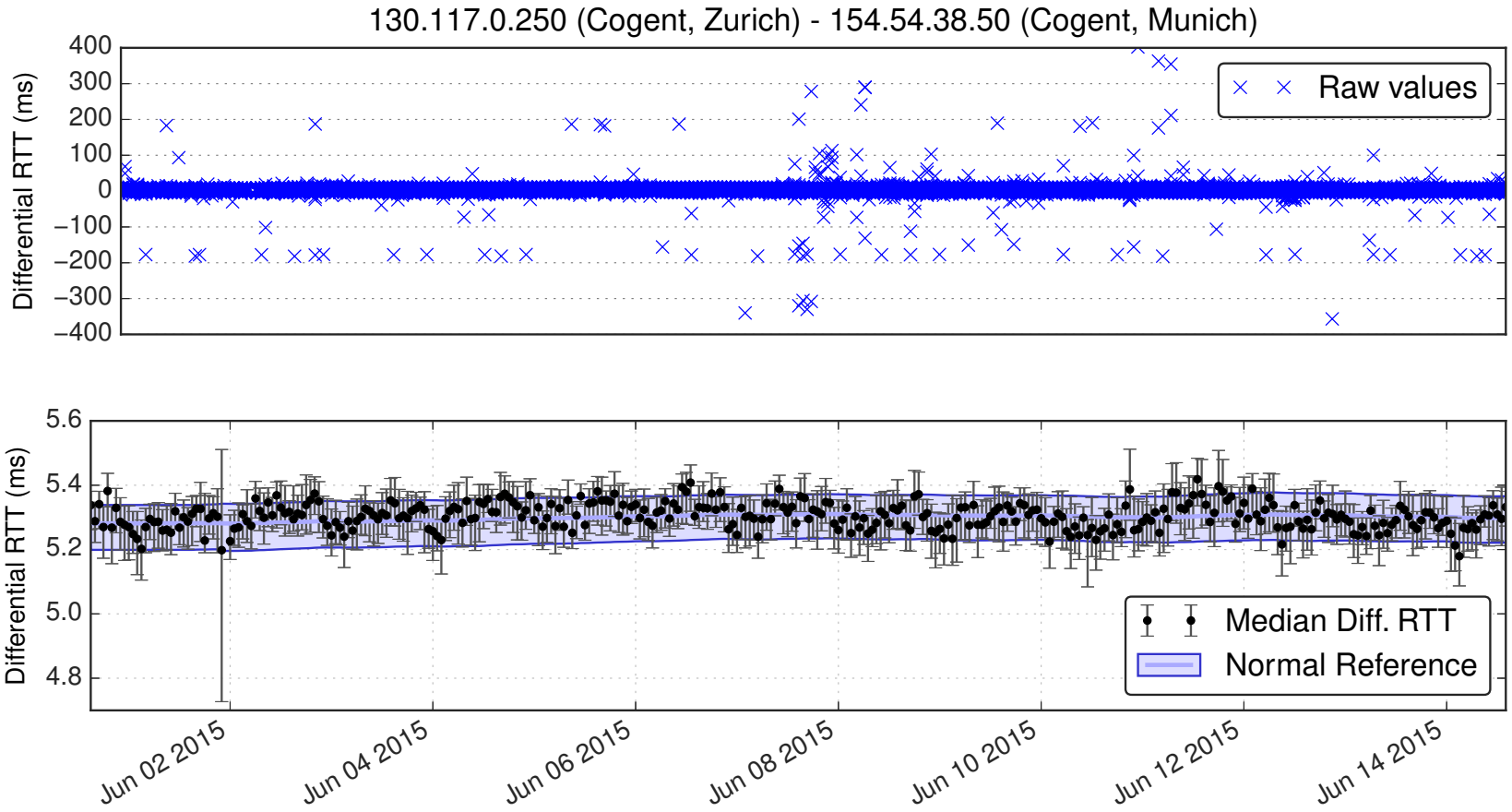
Differential RTT: $\Delta_{CB} = \{x_0, x_1, x_2, x_3, x_4\}$



Median Δ_{CB}:

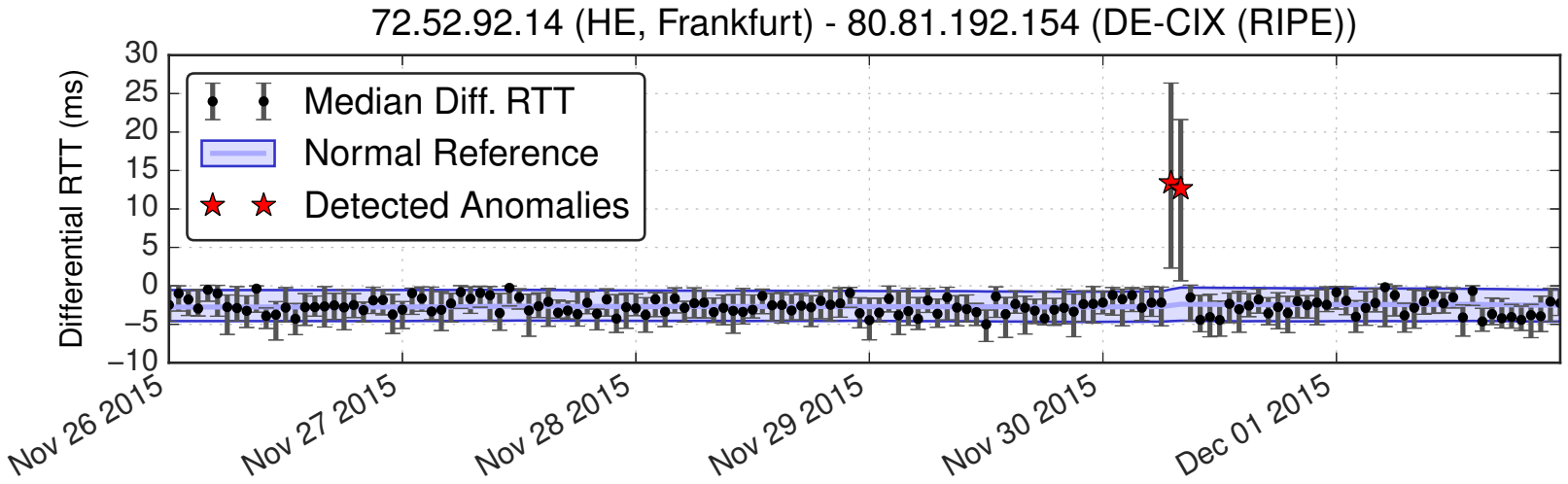
- Stable if a few return paths delay change
- Fluctuate if delay on BC changes

Median Diff. RTT: Tier1 link, 2 weeks of data, 95 probes



- **Stable** despite noisy RTTs (not true for average)
- Normally distributed

Detecting congestion



Significant RTT changes:

Confidence interval not overlapping with the normal reference

Results

Analyzed dataset

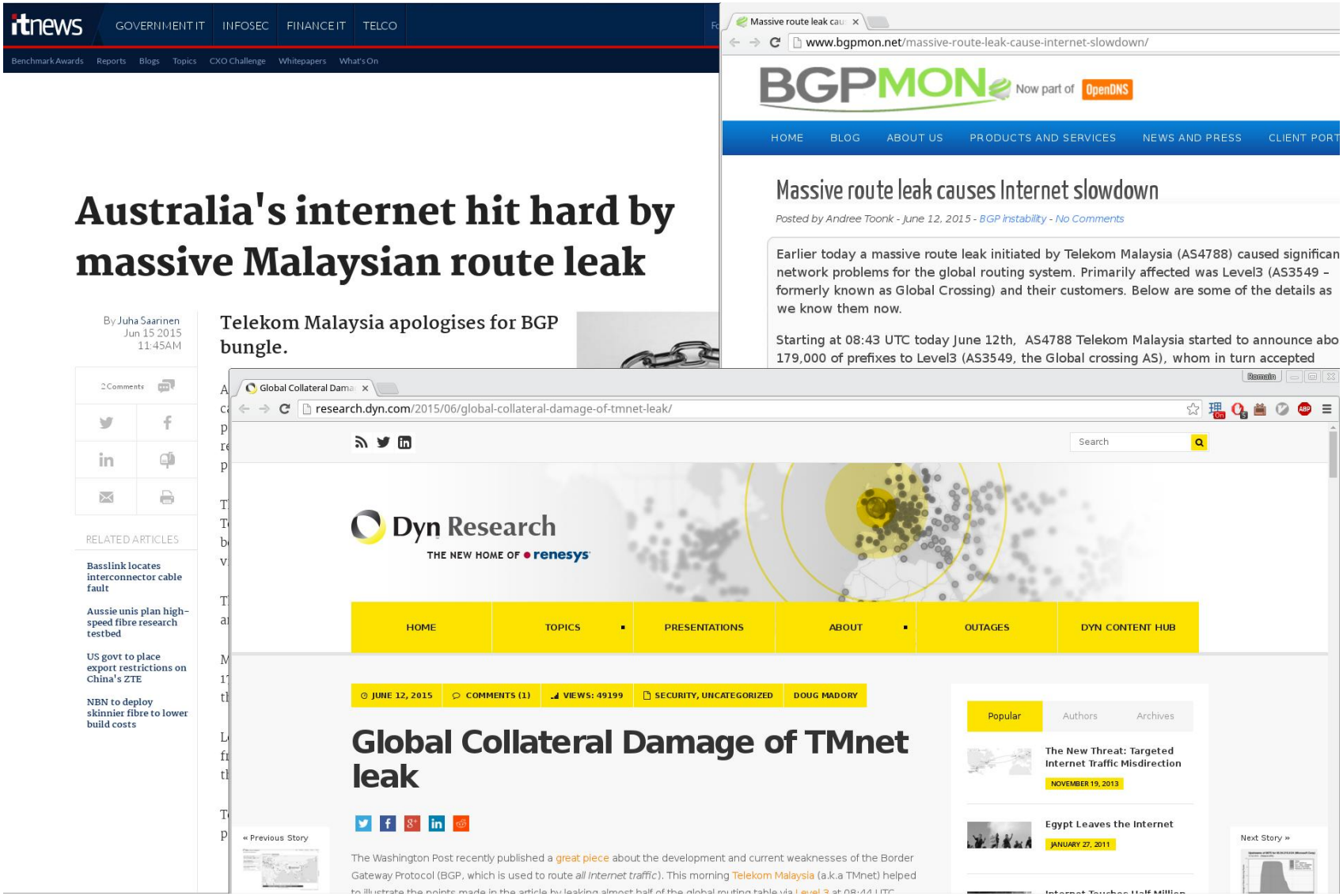
- Atlas *builtin/anchoring* measurements
- From May to Dec. 2015
- 2.8 billion IPv4 traceroutes
- 1.2 billion IPv6 traceroutes

- Observed 262k IPv4 and 42k IPv6 links (core links)

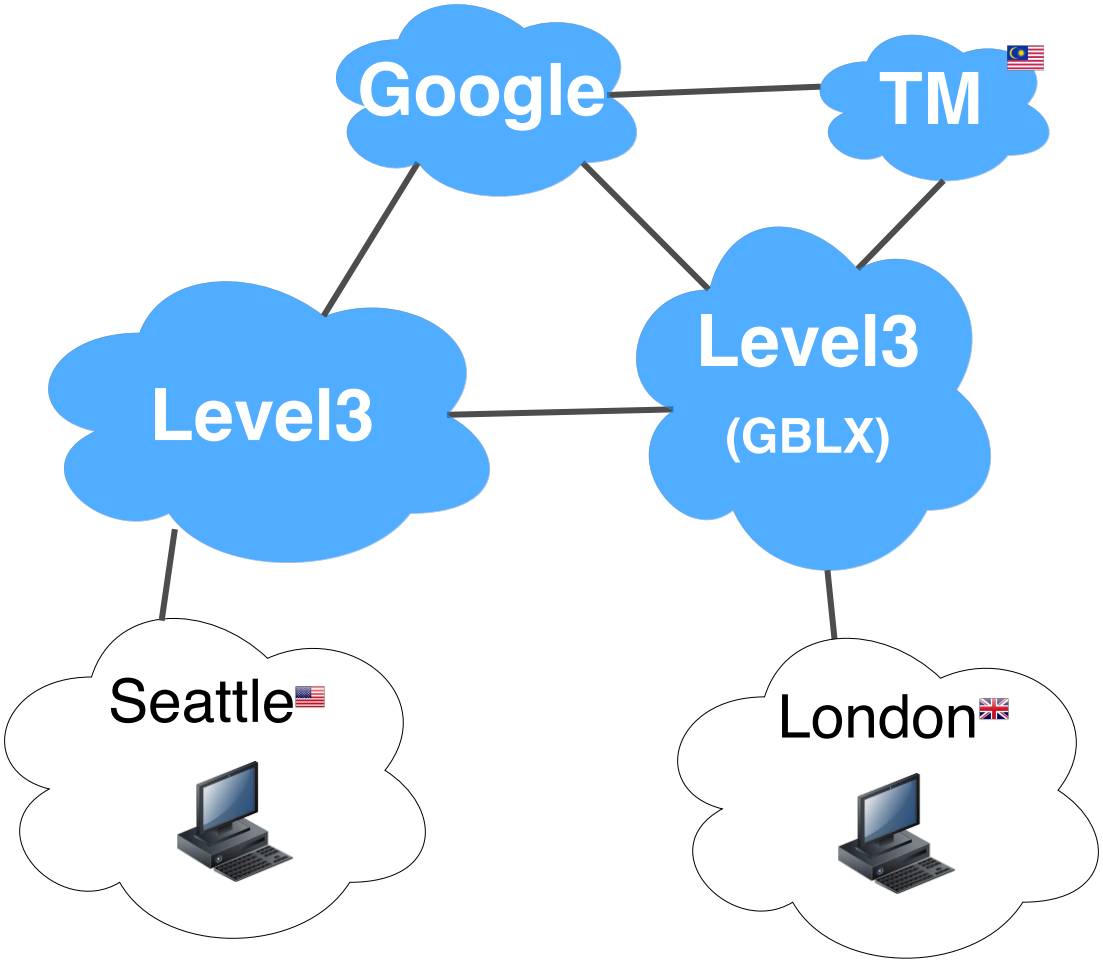
We found a lot of congested links!

Let's look at one example

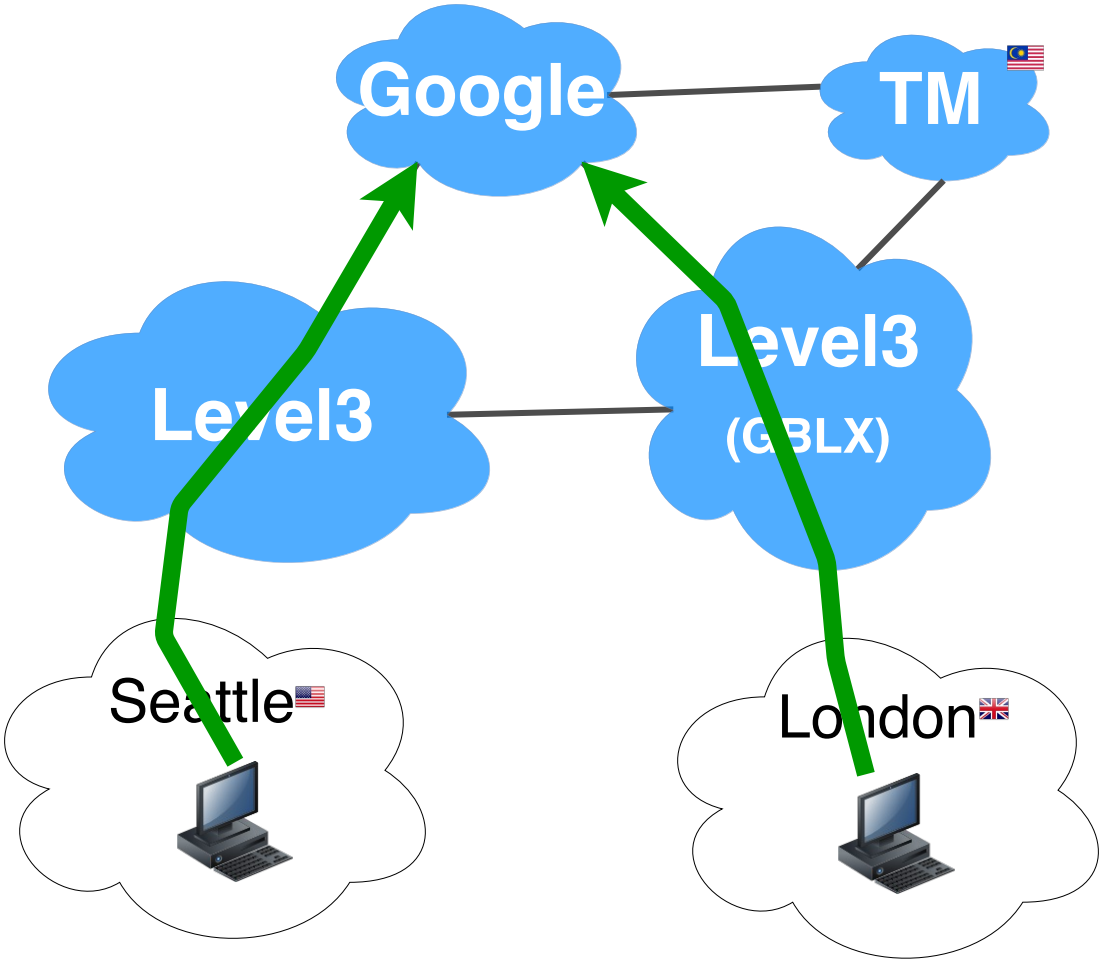
Study case: Telekom Malaysia BGP leak



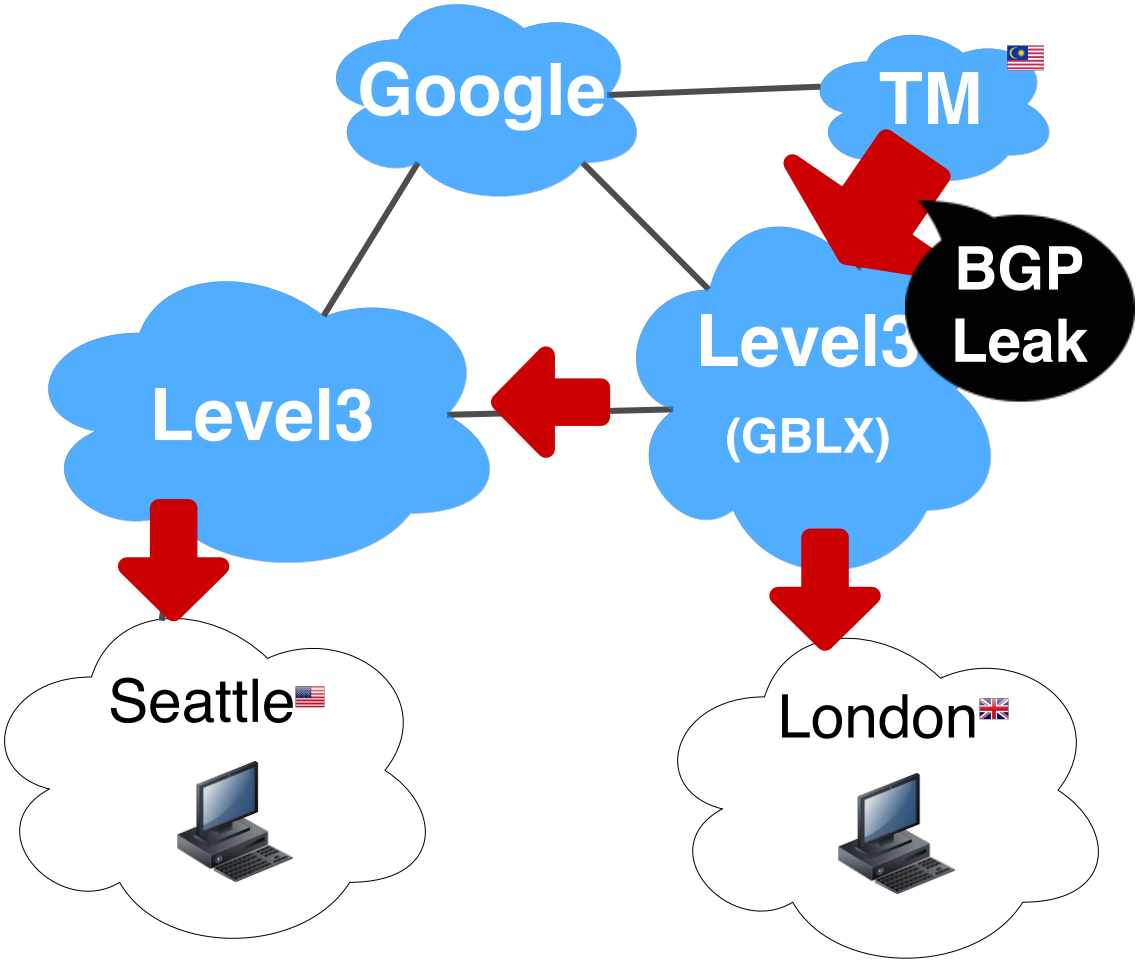
Study case: Telekom Malaysia BGP leak



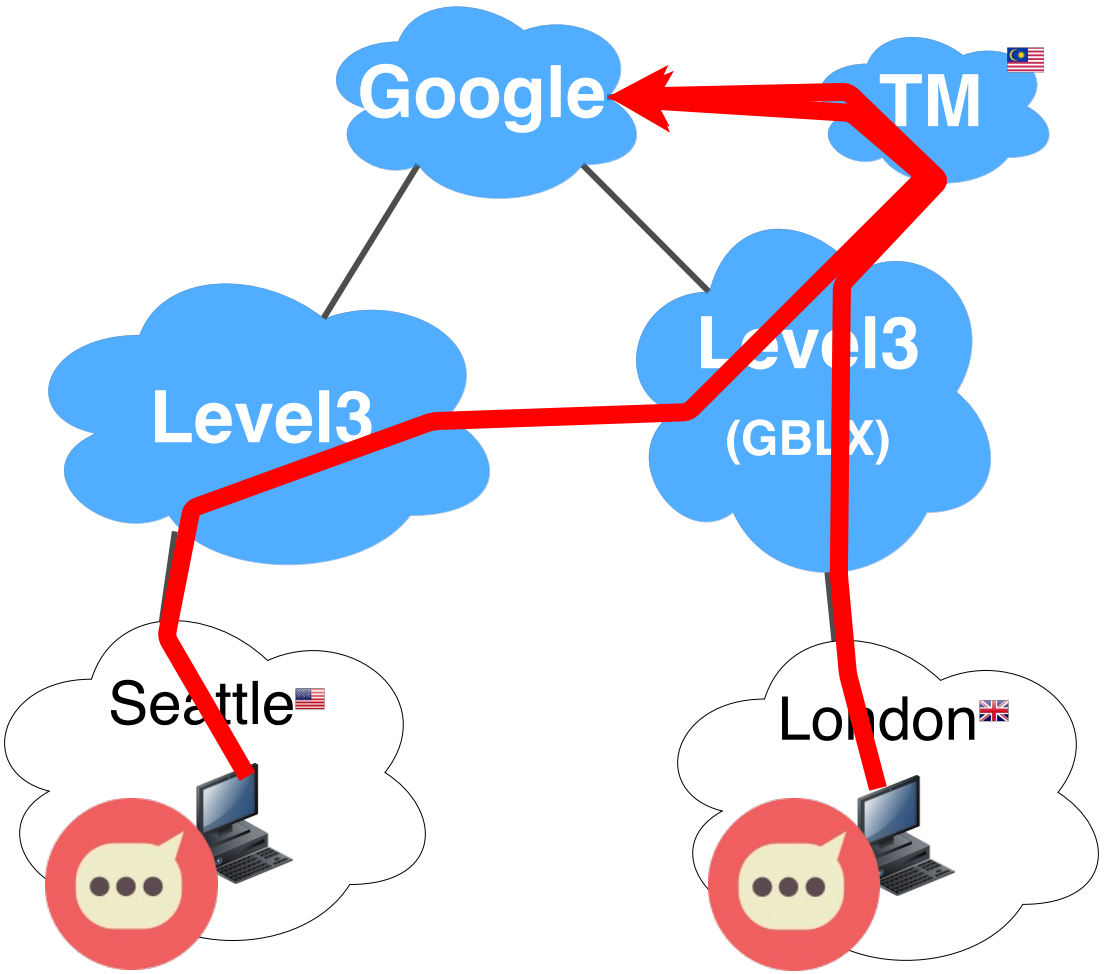
Study case: Telekom Malaysia BGP leak



Study case: Telekom Malaysia BGP leak



Study case: Telekom Malaysia BGP leak

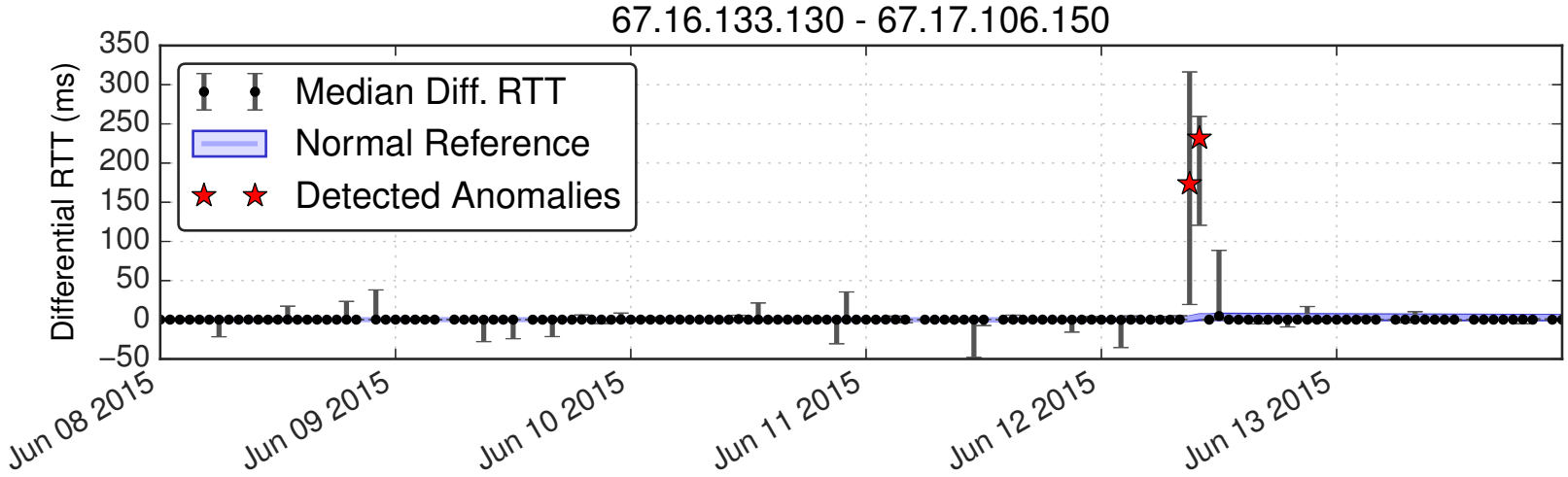


Not only with Google... but about **170k prefixes!**

Congestion in Level3

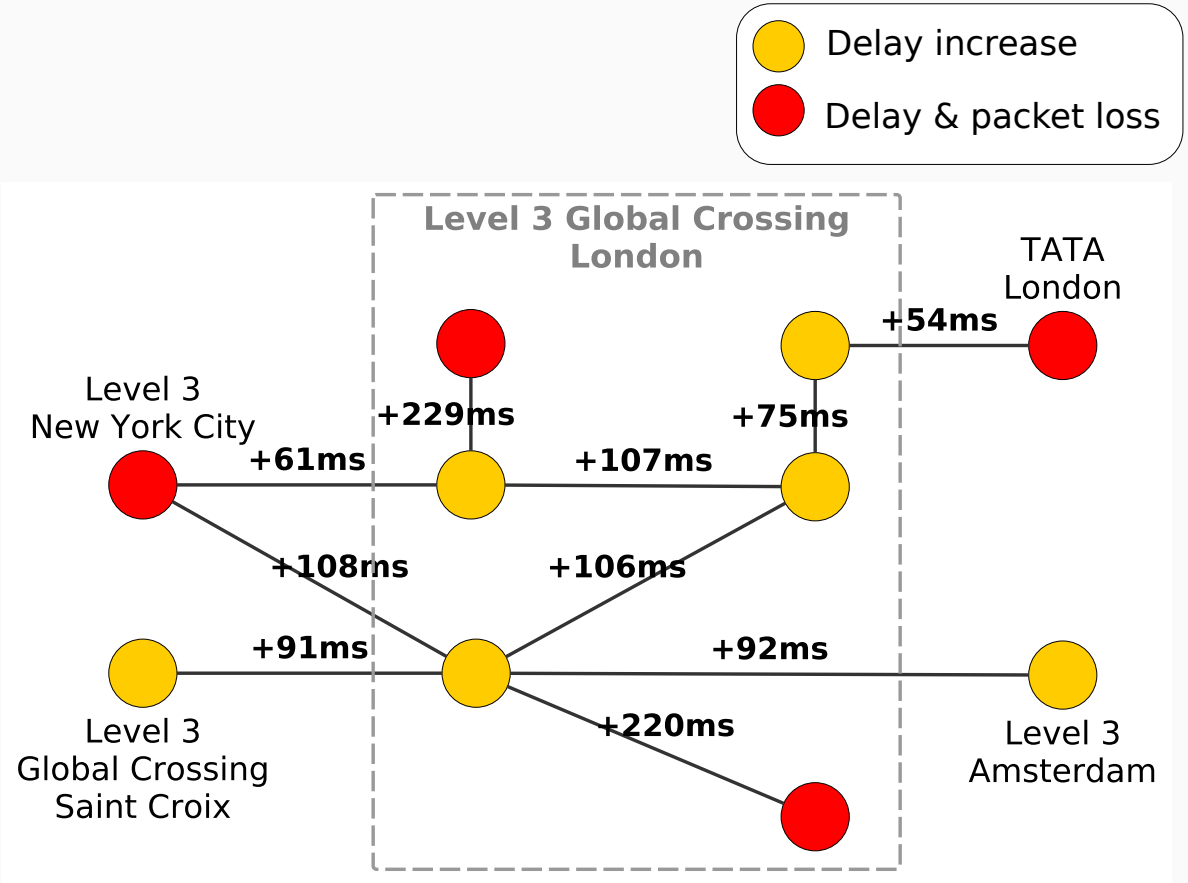
Rerouted traffic has congested Level3 (120 reported links)

- Example: 229ms increase between two routers in London!



Congestion in Level3

Reported links in London:



→ Traffic staying within UK/Europe may also be altered

But why did we look at that?

Per-AS alarm for delay

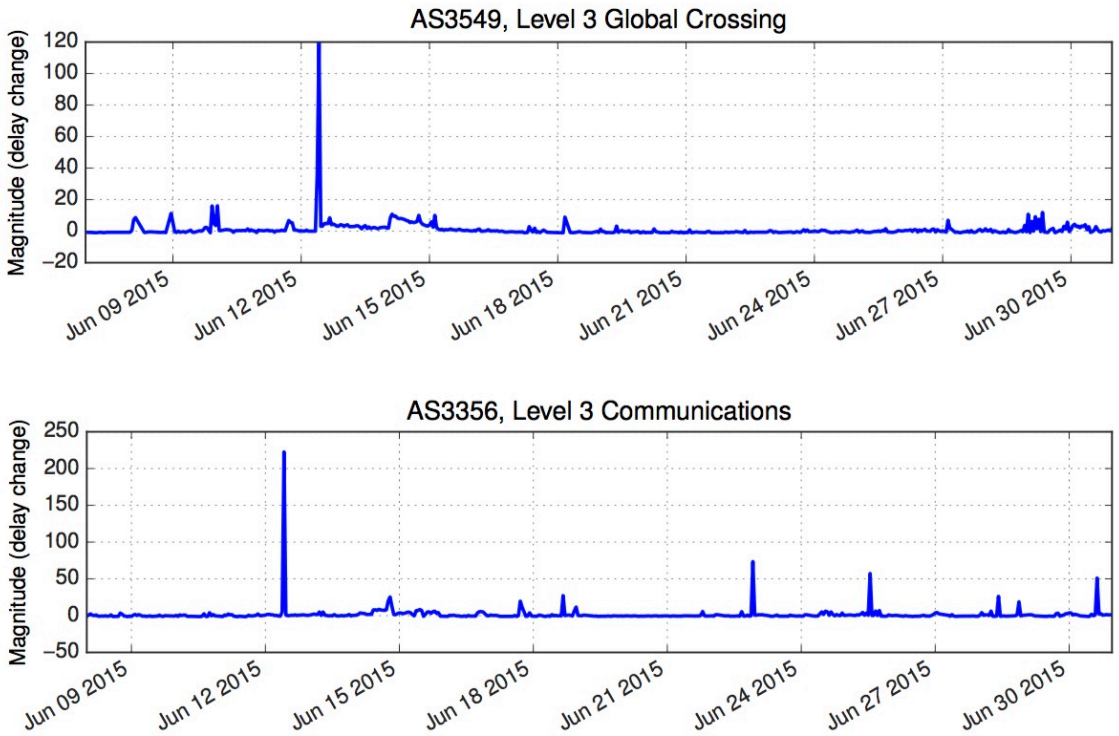


Figure 8: Delay change magnitude for all monitored IP addresses in two Level(3) ASs.

Conclusions and perspectives (1)

We proposed 3 different techniques to detect outages for 3 different sources of data

- Each source of data has its own coverage
  - Core links (congestion and failures)
  - Prefix, country, region, AS disconnections

Conclusions and perspectives (1)

We proposed 3 different techniques to detect outages for 3 different sources of data

- Each source of data has its own coverage
  - Core links (congestion and failures)
  - Prefix, country, region, AS disconnections
- Each source of data has its own noise, properties
  - Identifying the suitable model is a challenge

Conclusions and perspectives (2)

There is no substancial, state of the art ground truth to validate the results. We resort to

- the comparison of different techniques with different coverages
- evaluations on the basis of partial ground truth
- characterizations of the detected outages based on the detection algorithm used

Turn this

Into this

http://ihr.iijlab.net