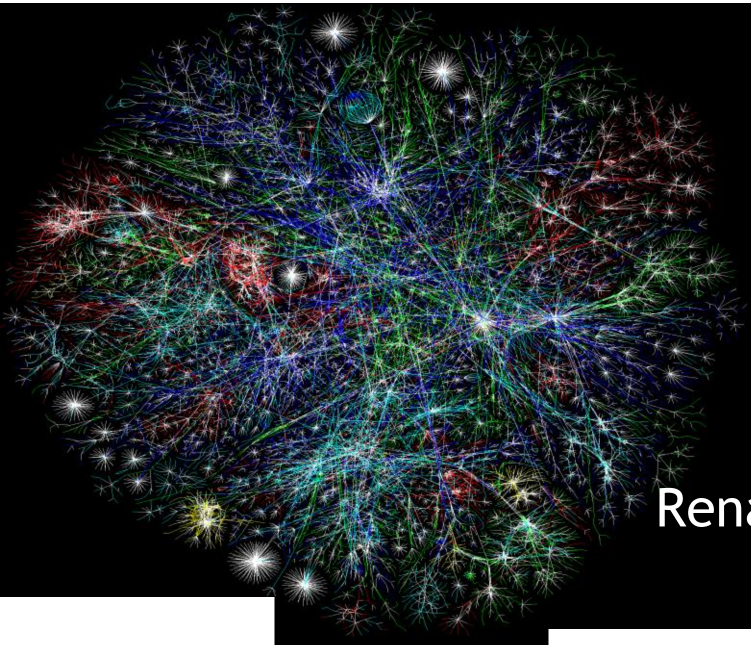


Predicting and Tracking Internet Path Changes

Ítalo Cunha

Renata Teixeira, Darryl Veitch, and Christophe Diot

Problem statement

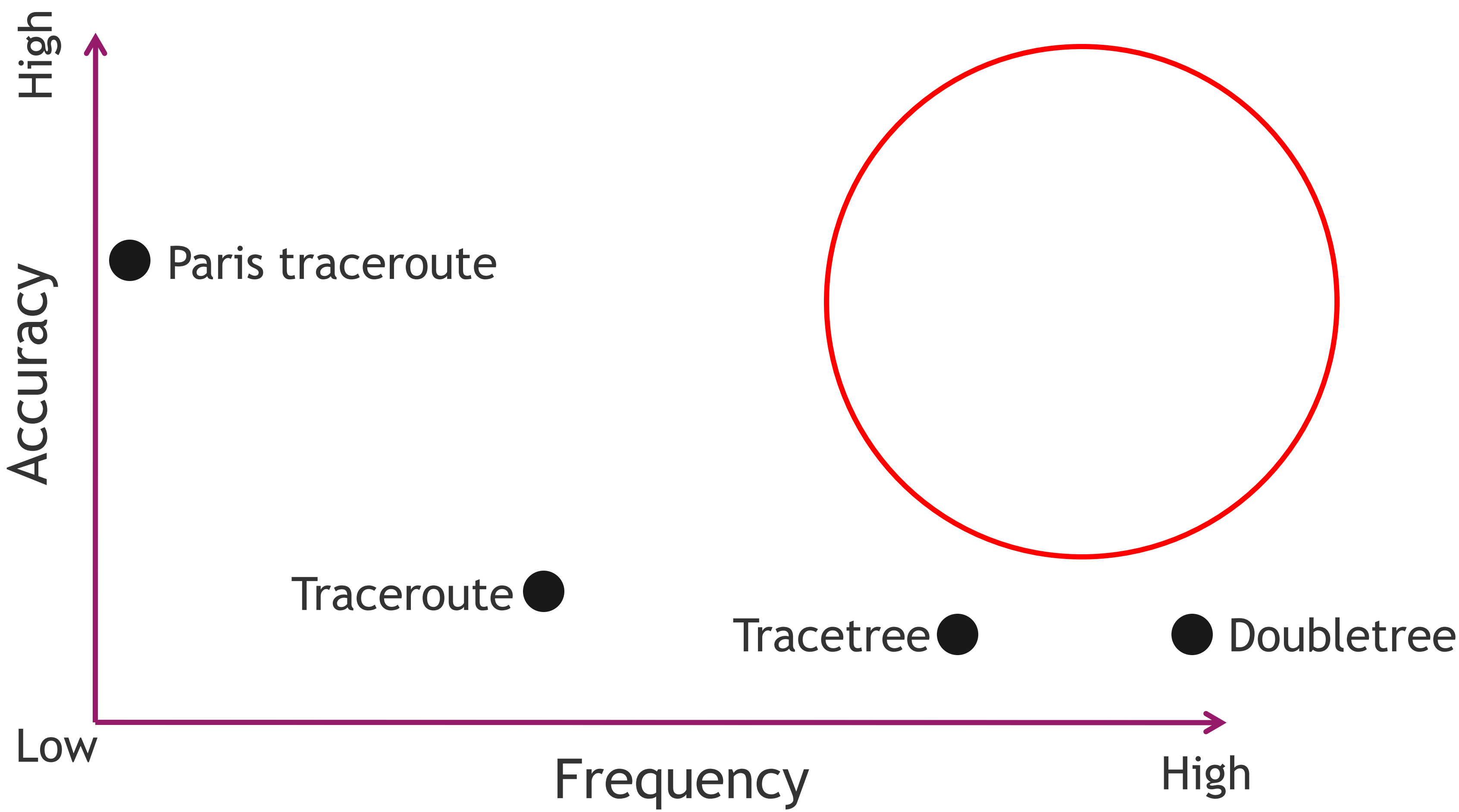
Goal: track large number of paths

Current approach: traceroute-style measurements

Challenges

- Cannot measure frequently enough to detect all changes
  - Network and system limitations
- Accurate measurements require extra probes
  - Identify all paths under load balancing

Frequent vs. accurate measurements



technicolor

Approach

Observation: Internet paths are mostly stable

- Current techniques waste probes

Probe according to path stability

Separate tasks of change *detection* and change *remapping*

- Use lightweight probing to detect changes faster
- Remap with Paris traceroute to get accurate path measurements

Contributions

NN4: Predicting Internet path changes

- Distinguish between stable and unstable paths

DTrack: Tracking Internet path changes

- Lightweight probing process to detect changes
- Allocates more probes to unstable paths

technicolor

Predicting path changes

Prediction goals

- Time until the next change
- Number of changes in a time interval
- Whether a path will change in a time interval

Identify path features that can help with prediction

- Features must be computable from traceroute measurements
  - Characteristics of the current path
  - Characteristics of the last path change
  - Behavior of the path in the recent past

technicolor

Feature selection

Use RuleFit to identify the relative importance of features

1. Fraction of time path was active in the past (prevalence)
2. Number of changes in the past
3. Number of previous occurrences of the current path instance
4. Path age

Four most important features carry all the predictive information

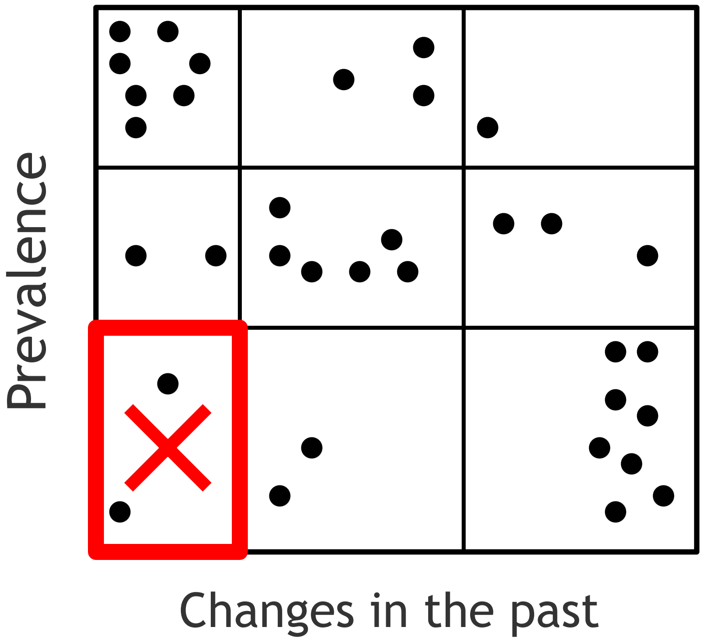
technicolor

NN4 predictor

RuleFit is CPU-intensive and hard to integrate in other systems

NN4 is based on the nearest-neighbor scheme

- Compute neighbors by partitioning the path feature “state-space”
  - Boundaries computed from feature distributions
- Prediction computed as the average behavior of all neighbors



FastMapping data

Frequent path measurements

- 5 times faster than Paris traceroute

Complete information about routers performing load balancing

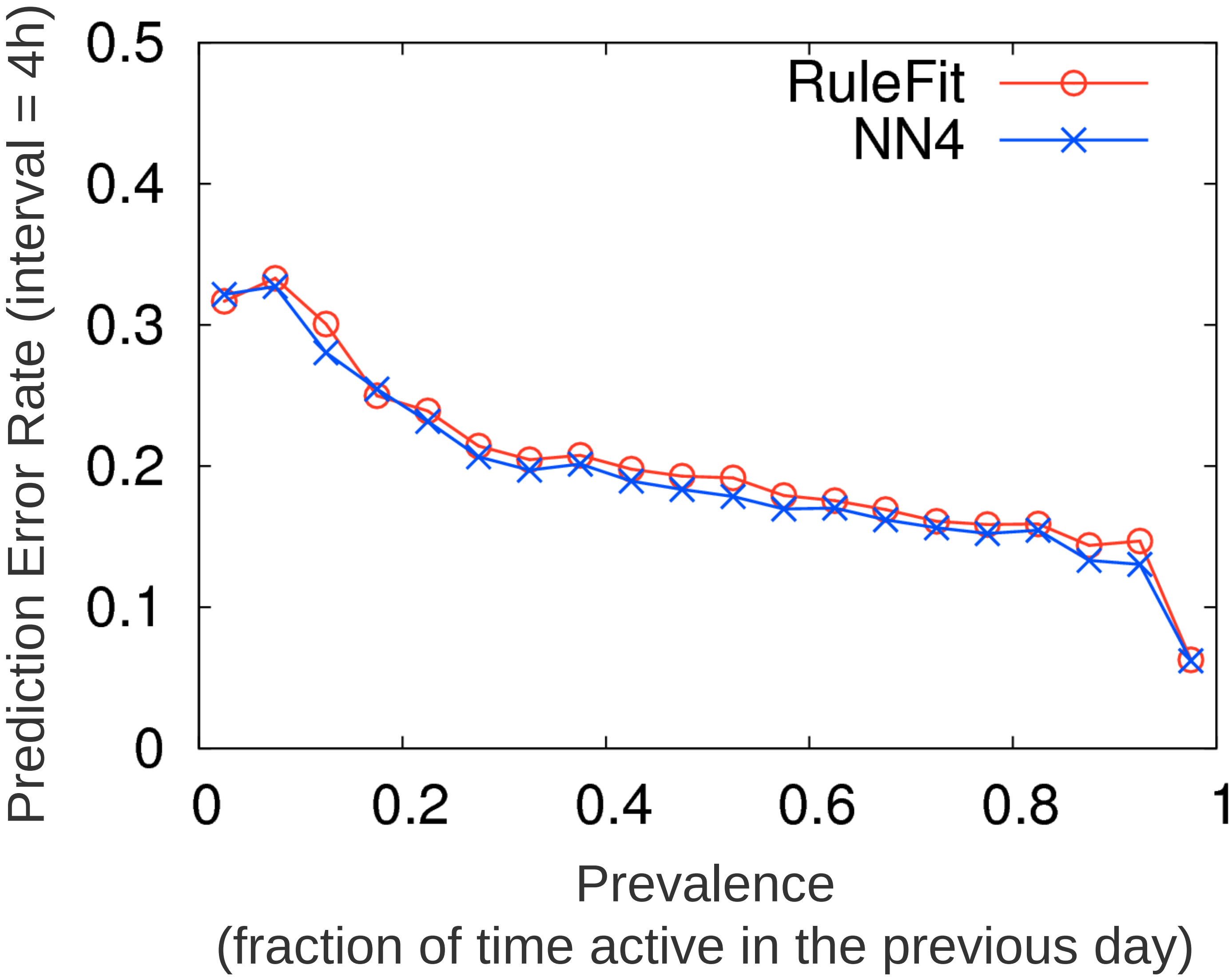
- Required to differentiate load balancing from routing changes

70 PlanetLab hosts probing 1000 destinations

5 weeks of data starting September 1st, 2010

Dataset covers 7942 ASes and 97% of the large ASes

NNN4 performance



NN4: summary

NN4 is lightweight, easy to integrate, and as accurate as RuleFit

Prediction is not highly accurate

- It is possible to distinguish unstable from stable paths

DTrack

Goal: Given a probing budget, detect as many changes as possible

Allocates probing rates *per path* using NN4's predictions

Targets probes along each path

- Reduce redundant probes at shared links
- Spread probes over time

technicolor

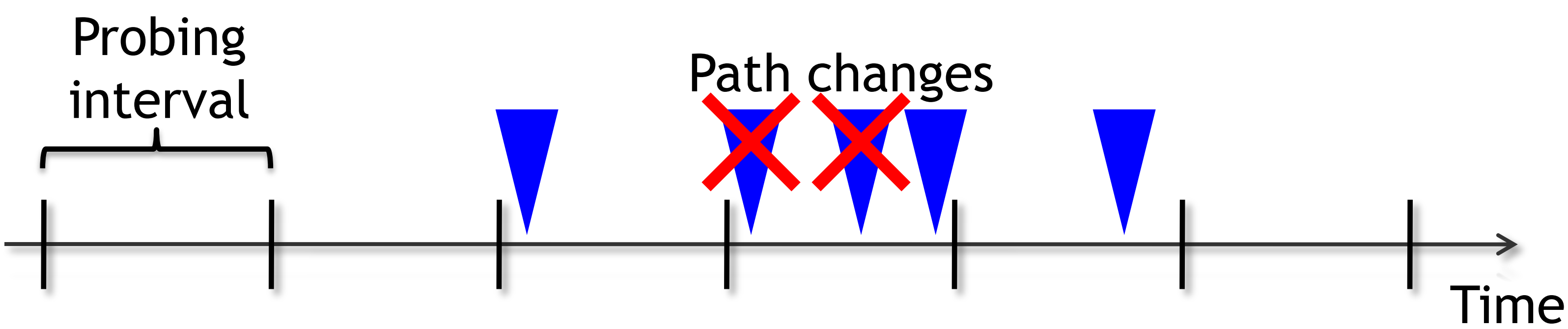
Probe rate allocation

Allocate rates that minimize total number of missed changes

Model changes in each path as a Poisson process

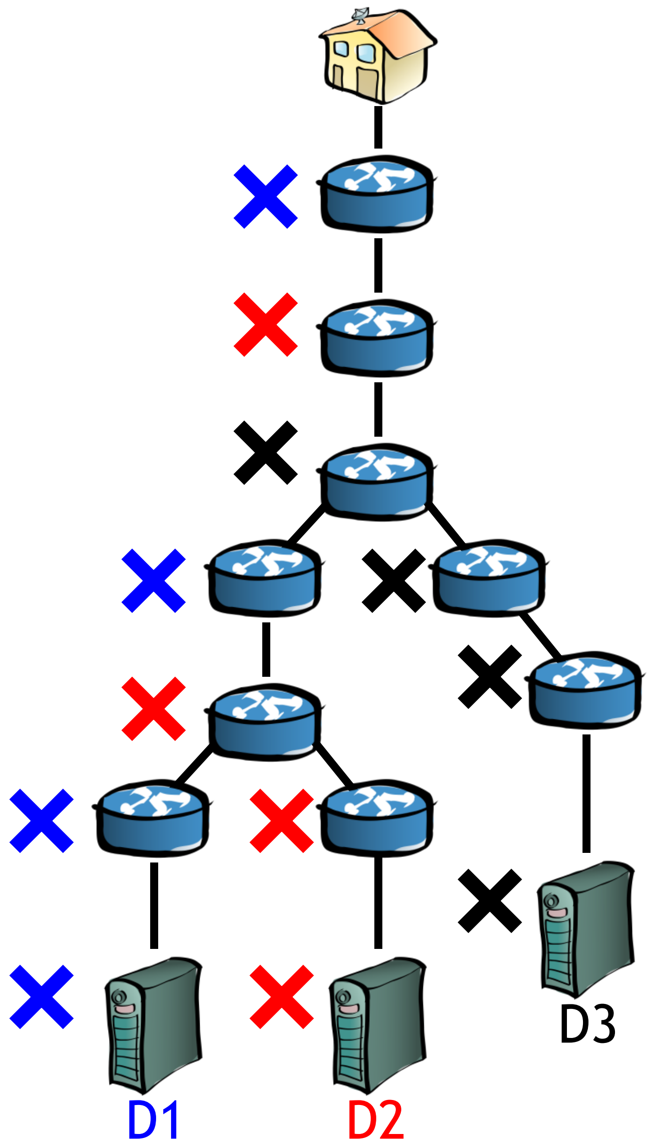
- Estimate the rate of changes using NN4

Compute missed changes as function of probing rate



technicolor

Probe targeting overview



technicolor

Evaluation

Method

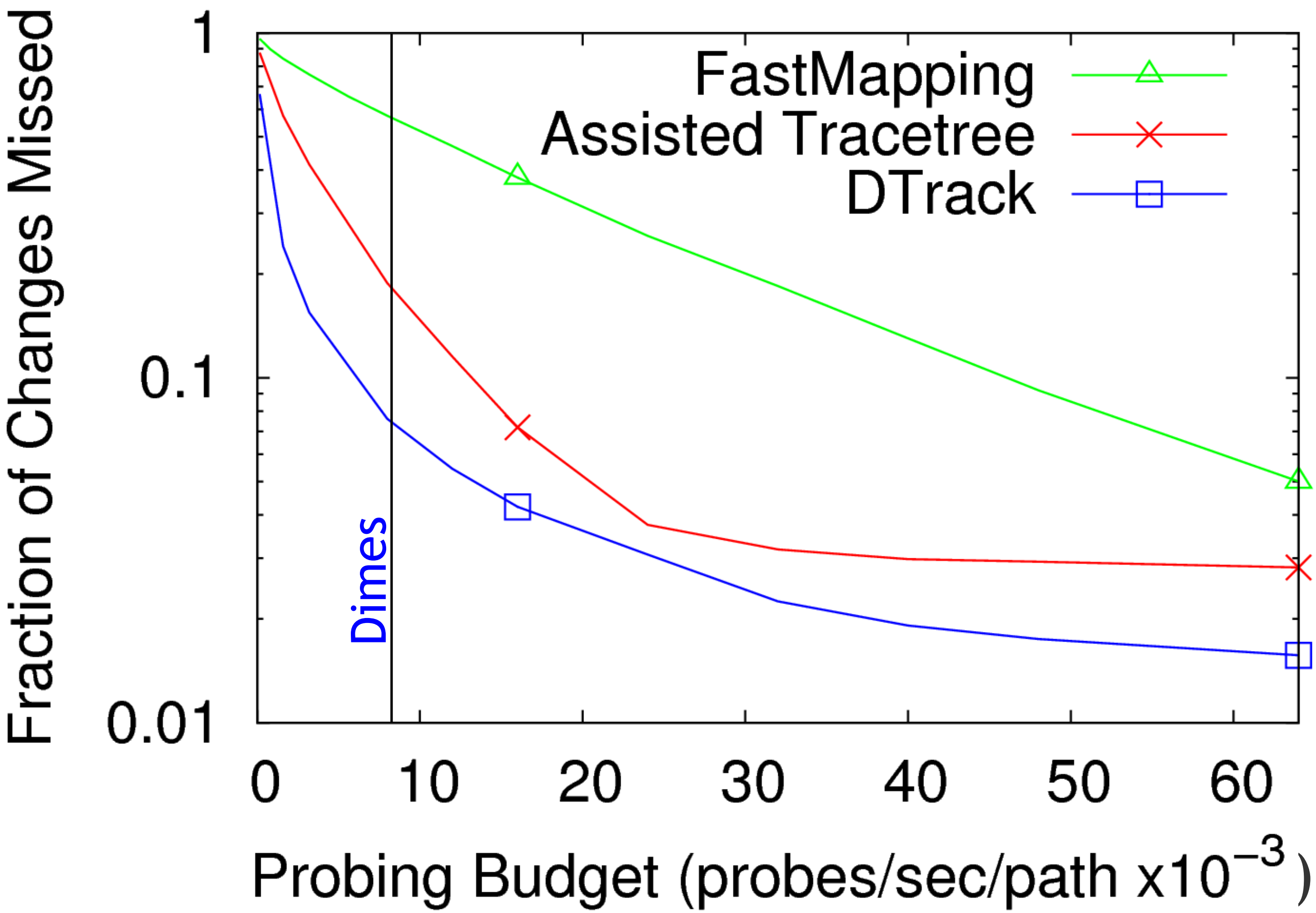
- Trace-driven simulations using the FastMapping dataset

Performance metrics

- Number of missed changes
- Change detection delay

Compare against FastMapping and Tracetree

Number of changes missed



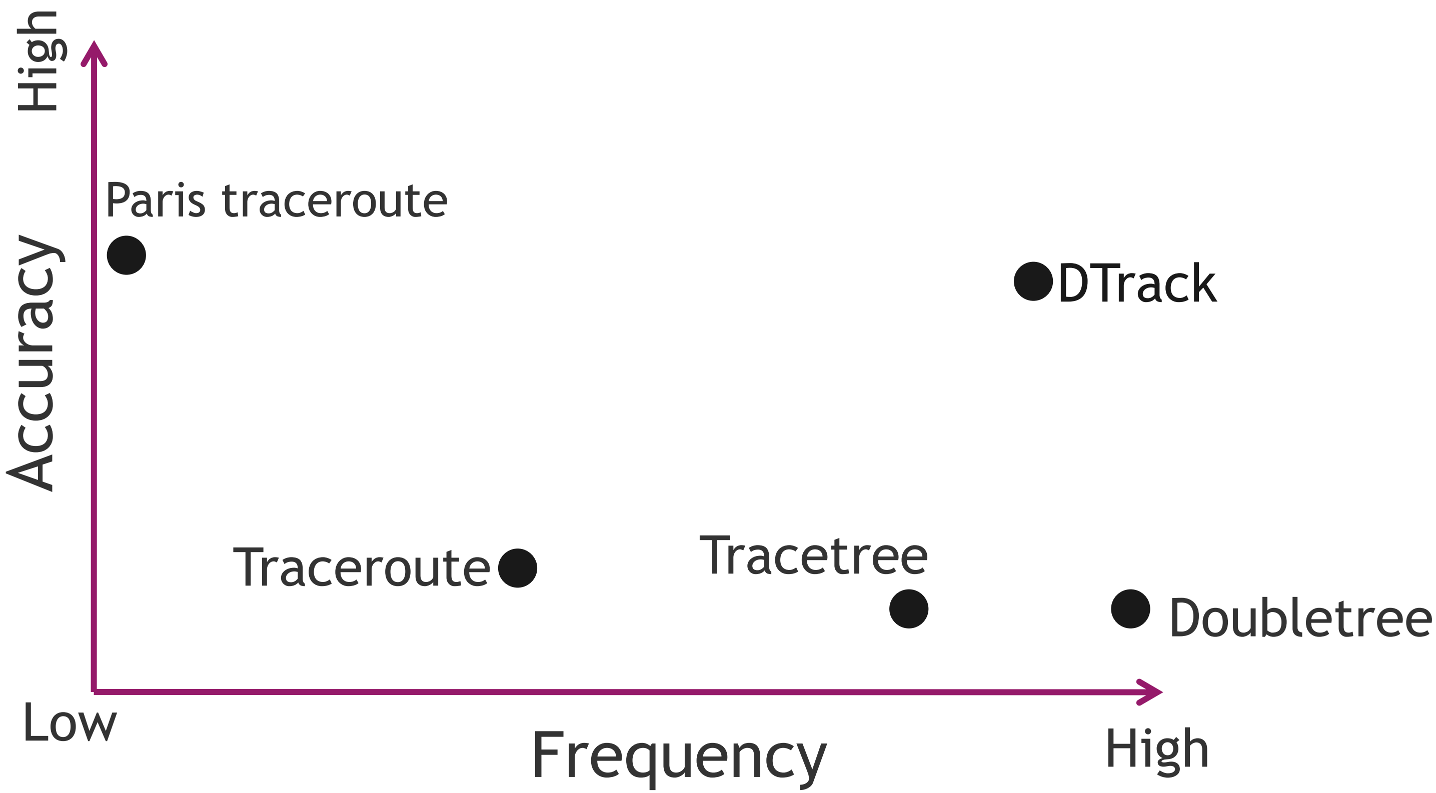
technicolor

Conclusion

NN4: A lightweight predictor of path changes

- Distinguishes stable and unstable paths

DTrack detects more changes than the current state-of-the-art



technicolor

Future work

Deploy DTrack on gateways

Improve NN4's prediction accuracy

- Use extra information like BGP updates

Extend DTrack

- Reduce remapping cost
- Coordinate probing across multiple monitors

Thank you!

Questions?