Networks Structure and Dynamics
13. Dynamics of the Internet topology

Maximilien Danisch, Lionel Tabourier

LIP6 – CNRS and Sorbonne Université

first_name.last_name@lip6.fr

January 19^{th} 2020

Outline

References for this course

Fasts dynamics in Internet topology: observations and first explanations - *Magnien, Ouédraogo, Valadon, Latapy - 2009*

A Radar for the internet - *Latapy, Magnien, Ouédraogo - 2011*

Towards realistic modeling of IP-level routing topology dynamics - *Magnien, Medem, Kirgizov, Tarissan - 2013*

Unexpected properties

Important unexpected property

Heterogeneous degree distribution

Consequences on . . .

- robustness
- spreading phenomena
- . . .

Unexpected properties

Important unexpected property

Heterogeneous degree distribution

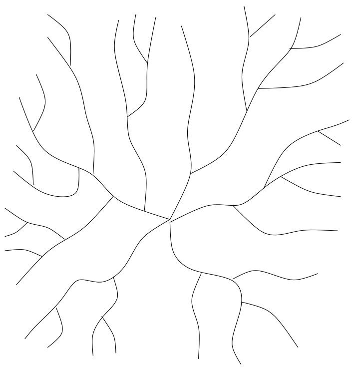
Consequences on . . .

- robustness
- spreading phenomena
- . . .

Led to works on the cartography of the Internet at the IP-level that we discuss here

Observed vs real properties (with traceroutes)

One source, a lot of destinations:



Observed vs real properties (with traceroutes)

- bias on the structure observed: missing links

⇒ theoretical and empirical works to . . .

- . . . assess the bias
- . . . assess some properties in an unbiased way

Summary

Cartography at the IP level

Long and expensive

Trend during 2010's

- Massive data
  increase number of sources and destinations, distributed measurements. . .
- Improve the quality of the measurement

What about the dynamical aspect?

"Measuring" the dynamics | Context and approach
Analysis of the dynamics | Tracetree
Modelling the dynamics | Measurements

Summary

Cartography at the IP level

Long and expensive

Trend during 2010's

- Massive data
  increase number of sources and destinations, distributed measurements. . .
- Improve the quality of the measurement

What about the dynamical aspect?

7/44

The approach in this course

Focus on what one machine sees of the Internet

- easier to measure (in terms of time and load)
  simple and efficient measure $\Rightarrow$ easy to repeat
- easier to interpret

$\Rightarrow$ ego-centered view

Radar

One source, several destinations, periodic measurements

"Measuring" the dynamics | Context and approach
Analysis of the dynamics | Tracetree
Modelling the dynamics | Measurements

The approach in this course

Focus on what one machine sees of the Internet

- easier to measure (in terms of time and load)
  simple and efficient measure $\Rightarrow$ easy to repeat
- easier to interpret

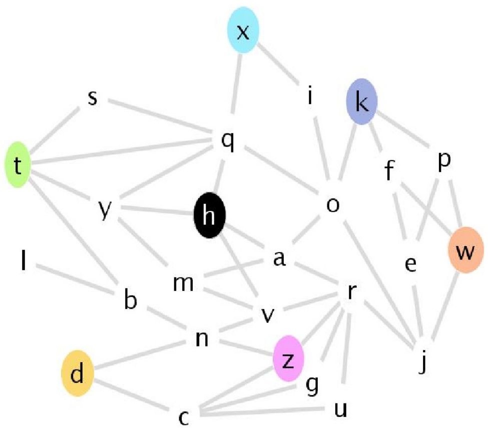
$\Rightarrow$ ego-centered view

Radar

One source, several destinations, periodic measurements

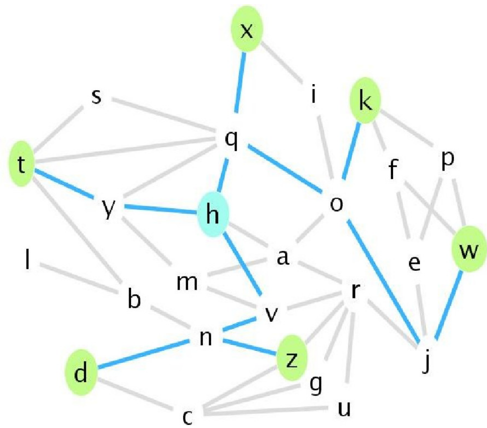
8/44

Ego-centered view

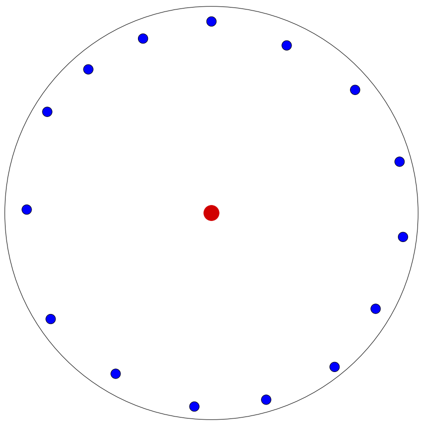


"Measuring" the dynamics
Analysis of the dynamics
Modelling the dynamics
Context and approach
Tracetree
Measurements

Ego-centered view

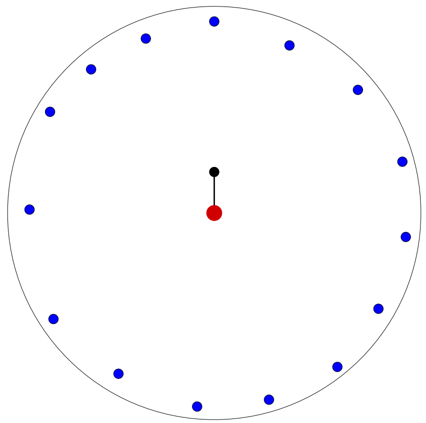


Measuring with traceroute

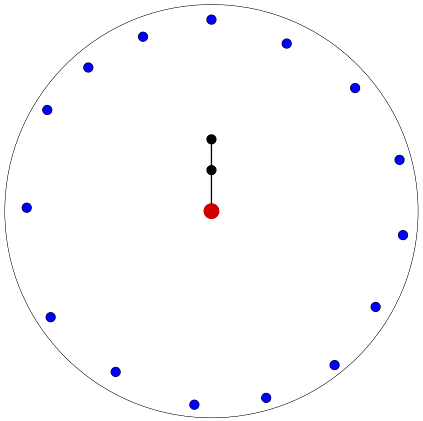


"Measuring" the dynamics
Analysis of the dynamics
Modelling the dynamics
Context and approach
Tracetree
Measurements

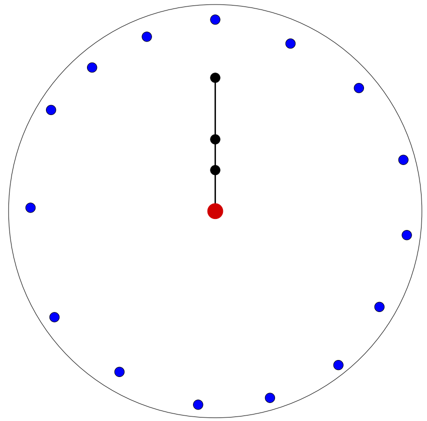
Measuring with traceroute



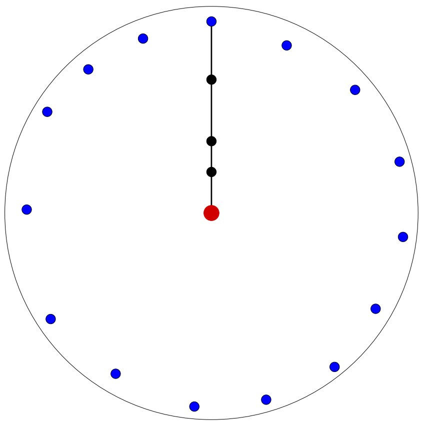
Measuring with traceroute



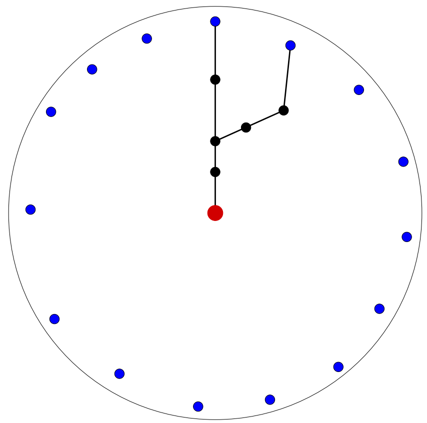
Measuring with traceroute



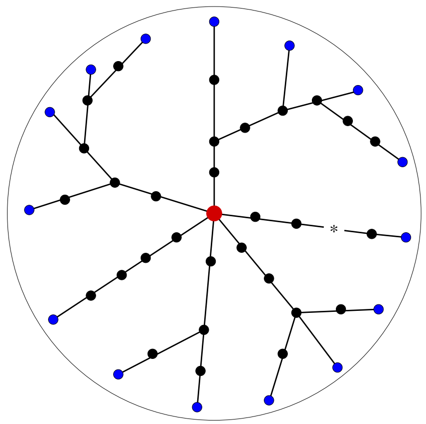
Measuring with traceroute



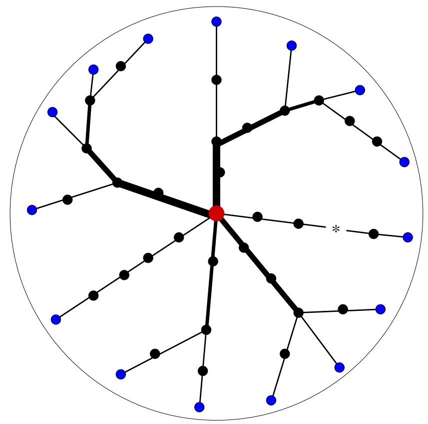
Measuring with traceroute



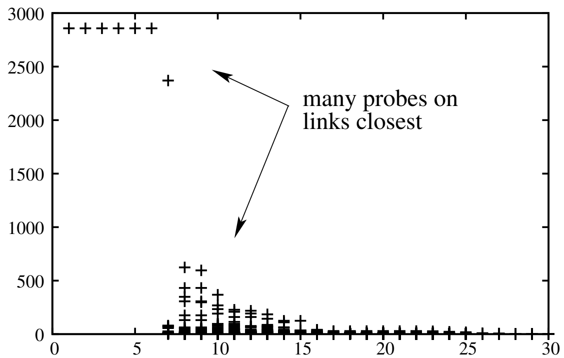
Measuring with traceroute



Unbalanced probing



Unbalanced probing



x = distance to source, y = # probes per link

Limits of the traceroute approach

- Unbalanced probing
  $\Rightarrow$ information redundancy
- Obtained view is not a tree
  $\Rightarrow$ make the analysis more complicated

$\Rightarrow$ need for a dedicated tool

tracetree

Limits of the traceroute approach

- Unbalanced probing
  ⇒ information redundancy
- Obtained view is not a tree
  ⇒ make the analysis more complicated

⇒ need for a dedicated tool

tracetree

"Measuring" the dynamics | Context and approach
Analysis of the dynamics | Tracetree
Modelling the dynamics | Measurements

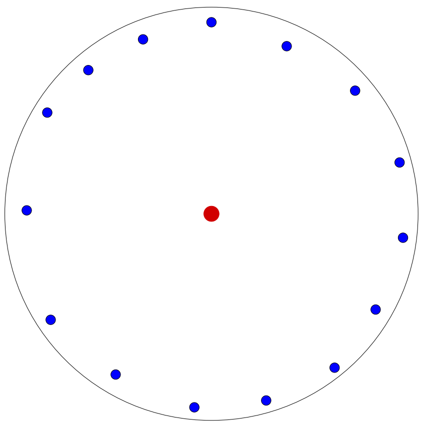
Limits of the traceroute approach

- Unbalanced probing
  ⇒ information redundancy
- Obtained view is not a tree
  ⇒ make the analysis more complicated

⇒ need for a dedicated tool

tracetree

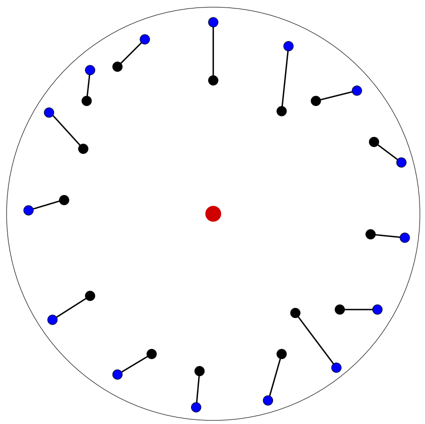
Tracetree



Backward measurement, in parallel, stop when paths connect

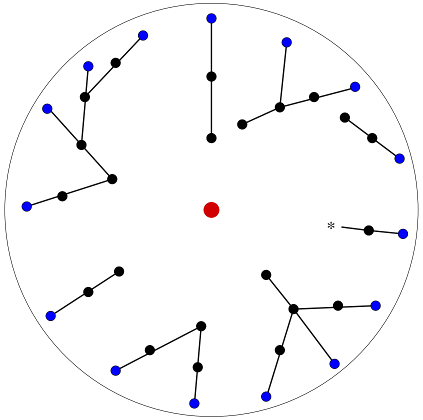
"Measuring" the dynamics | Context and approach
Analysis of the dynamics | Tracetree
Modelling the dynamics | Measurements

Tracetree



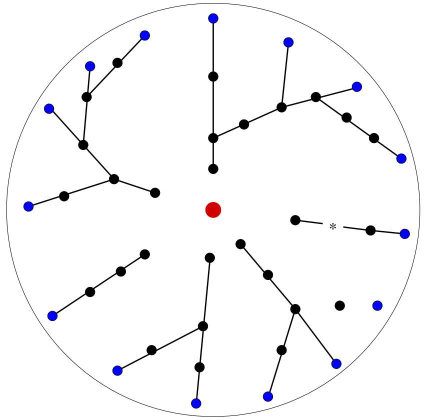
Backward measurement, in parallel, stop when paths connect

Tracetree



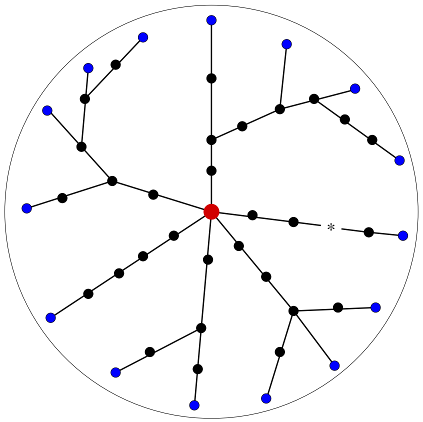
Backward measurement, in parallel, stop when paths connect

Tracetree



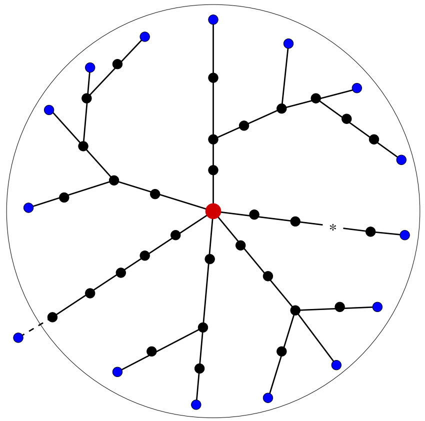
Backward measurement, in parallel, stop when paths connect

Tracetree



Backward measurement, in parallel, stop when paths connect

Tracetree



force a radius limit (= TTL max):
Probe until a certain distance to the source (typically 20 to 30)

Tracetree and Radar

Tracetree

- One packet per link
- Uniform (in terms of load) and fast measurement
- Tree

⟶ available *C* code
`http://www.complexnetworks.fr/programs/`

Radar

- One source
- A set of destinations
- Periodical tracetree measurements

But Tracetree same problems as traceroute for meas. quality

"Measuring" the dynamics
Analysis of the dynamics
Modelling the dynamics
Context and approach
Tracetree
Measurements

Tracetree and Radar

Tracetree

- One packet per link
- Uniform (in terms of load) and fast measurement
- Tree

⟶ available *C* code
`http://www.complexnetworks.fr/programs/`

Radar

- One source
- A set of destinations
- Periodical tracetree measurements

But Tracetree same problems as traceroute for meas. quality

"Measuring" the dynamics
Analysis of the dynamics
Modelling the dynamics
Context and approach
Tracetree
Measurements

Tracetree and Radar

Tracetree

- One packet per link
- Uniform (in terms of load) and fast measurement
- Tree

⟶ available *C* code
`http://www.complexnetworks.fr/programs/`

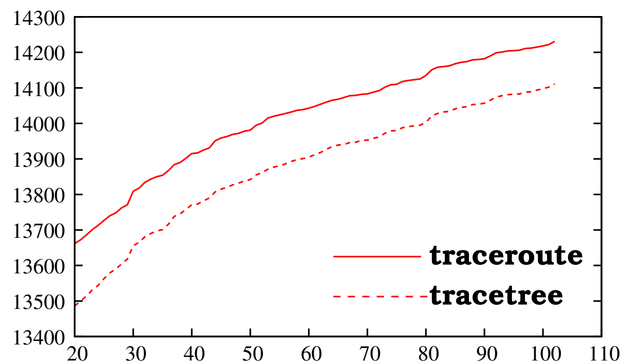
Radar

- One source
- A set of destinations
- Periodical tracetree measurements

But Tracetree same problems as traceroute for meas. quality

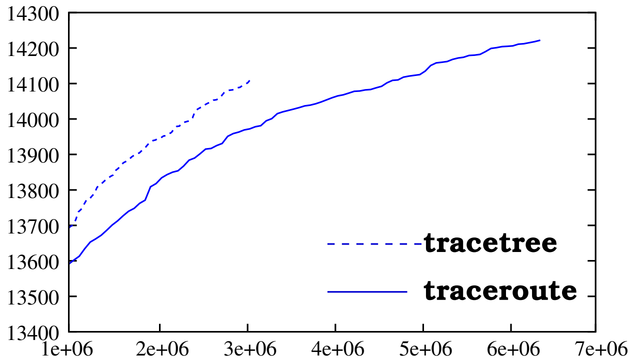
"Measuring" the dynamics
Analysis of the dynamics
Modelling the dynamics
Context and approach
Tracetree
Measurements

tracetree vs traceroute: empirical comparison



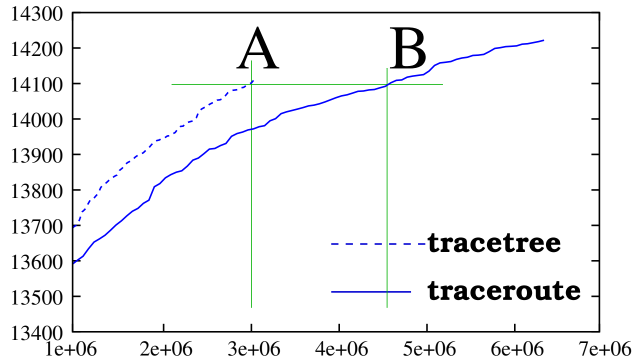
x=nb of rounds of measurements, y=nb of IP addresses discovered

tracetree vs traceroute: empirical comparison



x=nb of packets sent, y=nb of IP addresses discovered

tracetree vs traceroute: empirical comparison



x=nb of packets sent, y=nb of IP addresses discovered

Setting the experiments parameters

Measurement parameters

- Which source, which destinations?
- How many destinations?
- Which delay between consecutive rounds?
- ...

Finding a good trade-off

- High frequency
- Large measurements
- Low network load

Approach: parameters changed one by one
controlled measurement

Setting the experiments parameters

Measurement parameters

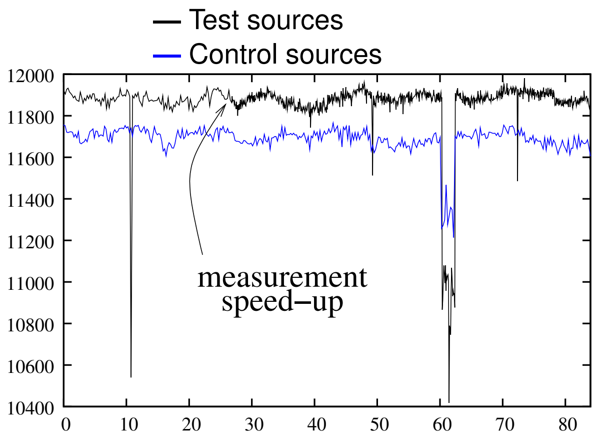
- Which source, which destinations?
- How many destinations?
- Which delay between consecutive rounds?
- ...

Finding a good trade-off

- High frequency
- Large measurements
- Low network load

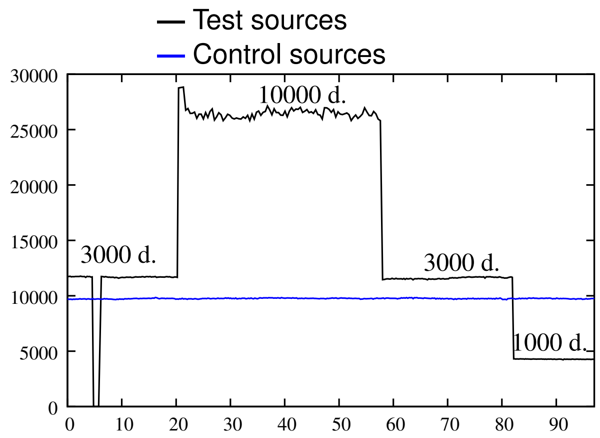
Approach: parameters changed one by one
controlled measurement

Parameters: frequency



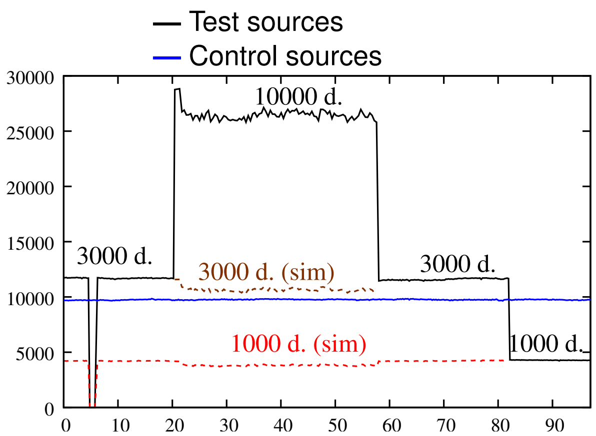
x=# hours, y=# IPs

Parameters: number of destinations



x=# hours, y=# IP

Parameters: number of destinations



x=# hours, y=# IP

Our measurements

Two parameter sets:

- normal: 3000 destinations, 10 min. delay between rounds, max TTL 30, ... $\sim$ 100 rounds / day
- fast: 1000 destinations, 1 min. delay between rounds, max TTL 15, ... $>$ 800 rounds / day, $\sim$ 36 / hour

Sources: PlanetLab and others ($>$ 100)

Destinations: random, answered once to ping

several months of uninterrupted measurement
dataset available for study
www-complexnetworks.lip6.fr/~latapy/Radar/

Our measurements

Two parameter sets:

- normal: 3000 destinations, 10 min. delay between rounds, max TTL 30, ... $\sim$ 100 rounds / day
- fast: 1000 destinations, 1 min. delay between rounds, max TTL 15, ... $>$ 800 rounds / day, $\sim$ 36 / hour

Sources: PlanetLab and others ($>$ 100)

Destinations: random, answered once to ping

several months of uninterrupted measurement
dataset available for study
`www-complexnetworks.lip6.fr/~latapy/Radar/`

Outline

Analysis

Which dynamics?

Goal:

- better understanding
- model, simulate
- detect events
- ...

How to answer to this question?
No out-of-the-shelf method

Data

Valid questions for any dynamical network

Focus on the radar normal dataset

- 3000 destinations
- one round every 15 minutes (100 rounds per day)

Study of a round

Variations as a function of:

- the source of the radar
- the moment of the data collection

General trends

- $\sim$ 12 000 IP
- $\sim$ 12 000 "stars"
- most nodes are at distance 13-18

Study of a round

Variations as a function of:

- the source of the radar
- the moment of the data collection

General trends

- $\sim$ 12 000 IP
- $\sim$ 12 000 "stars"
- most nodes are at distance 13-18

We always observe the same kind of behaviors

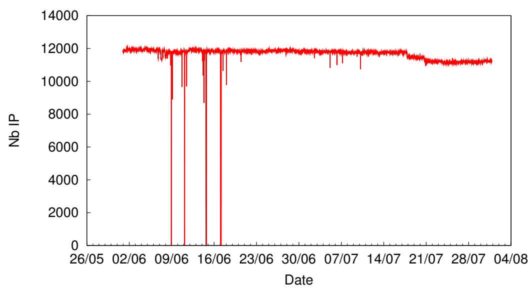
Simple properties through time

First approach: evolution of simple properties through time

Properties investigated

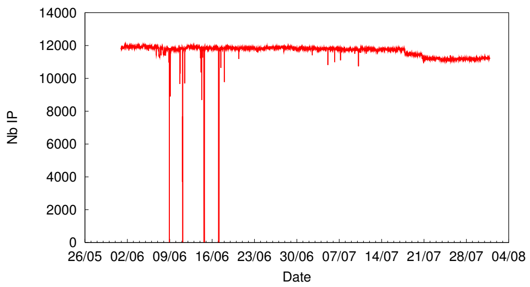
- number of IPs
- number of unanswered probes
- duration of rounds
- ...

Simple properties through time



- source in Japan
- two months of measurement
- $\sim$ 6 000 rounds

Simple properties through time



- number of IP seen more or less stable
- a few sudden drops (downward pikes) $\rightarrow$ disconnections?
- no upward pikes

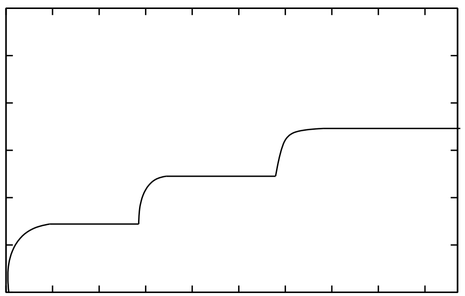
Stabilization?

Expected behavior?

x = number of rounds
y = number of distinct IP addresses from the beginning

Stabilization?

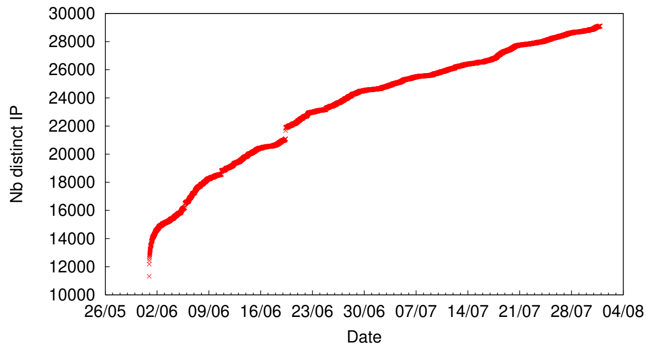
Expected behavior?



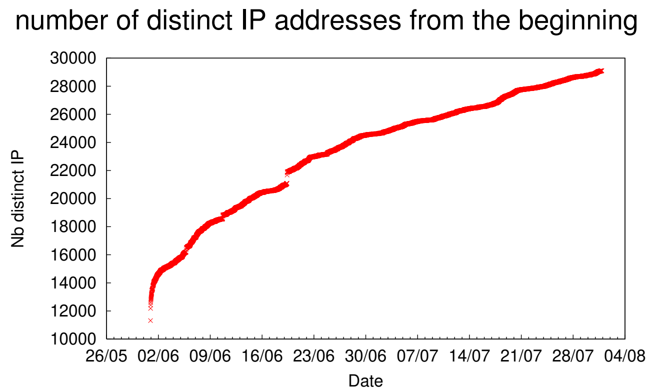
x = number of rounds
y = number of distinct IP addresses from the beginning

Stabilization?

number of distinct IP addresses from the beginning

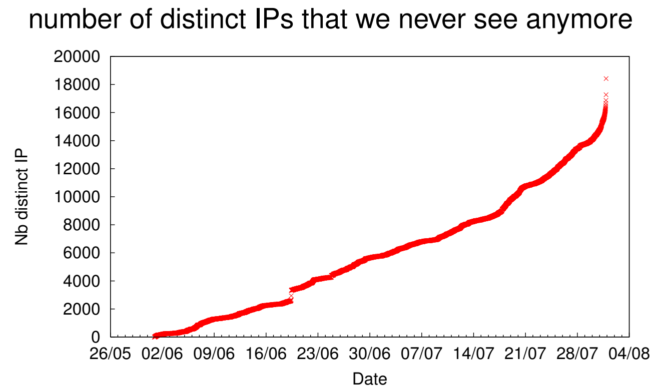


Stabilization?

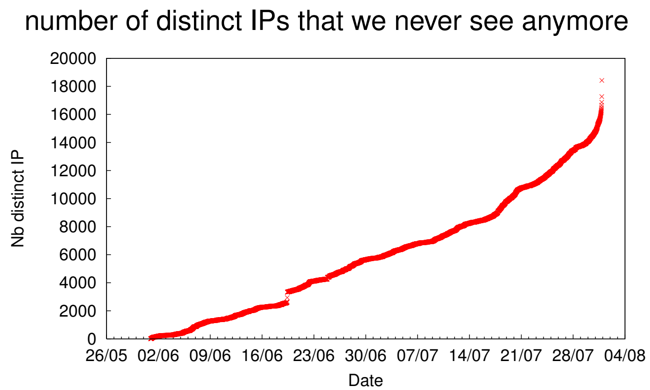


New IPs observed continuously

Disappearances



Disappearances



Continuous renewal of IPs

Possible explanations

Surprising growth... how to explain?

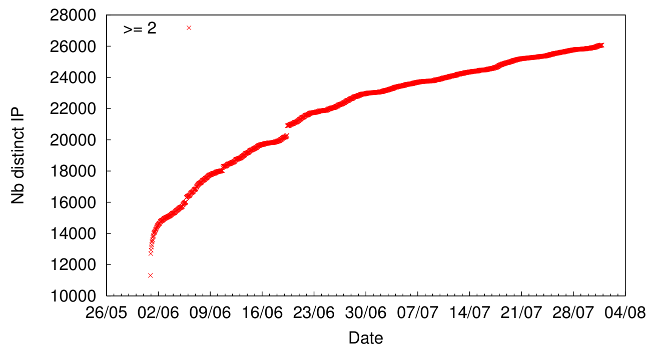
Possible explanations

Surprising growth... how to explain?

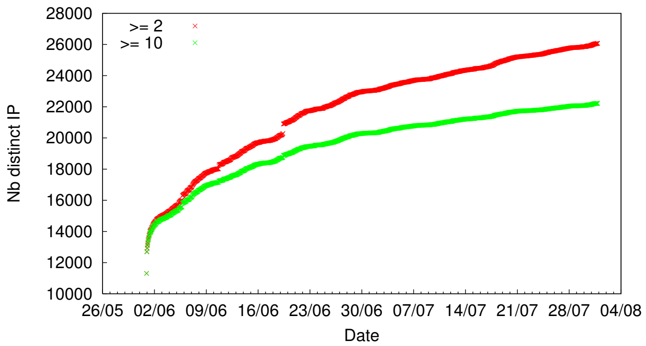
A "natural" explanations

- backbone of permanent IPs +
  "random" IPs (appears once then disappears)

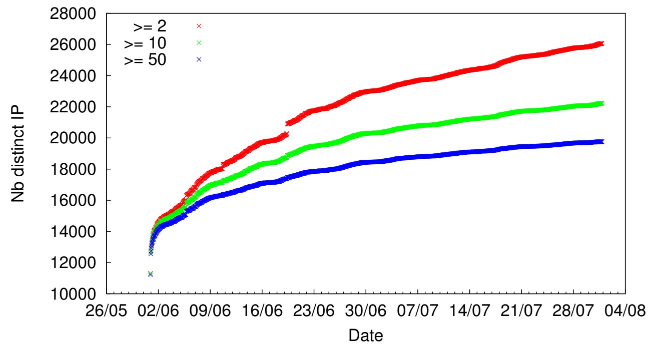
IPs seen only once?



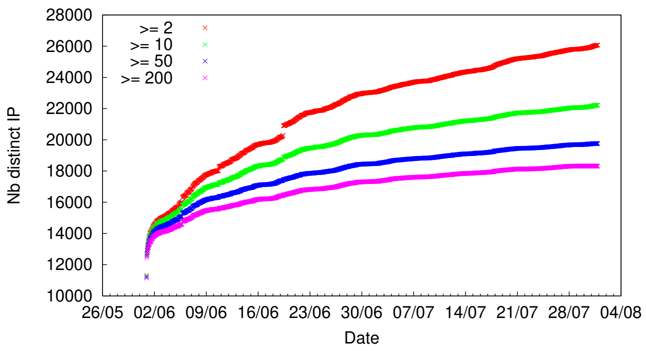
IPs seen only once?



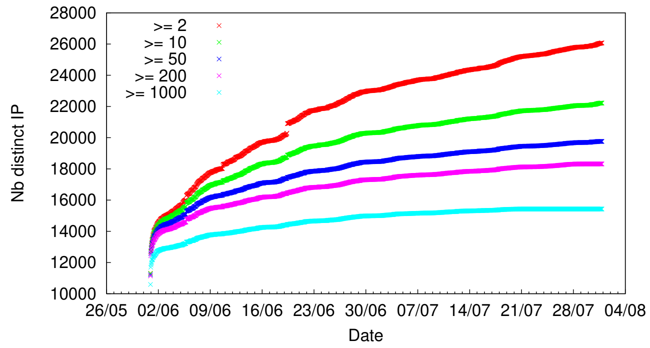
IPs seen only once?



IPs seen only once?



IPs seen only once?



Recurring new IPs...
Rules out the hypothesis of random IPs

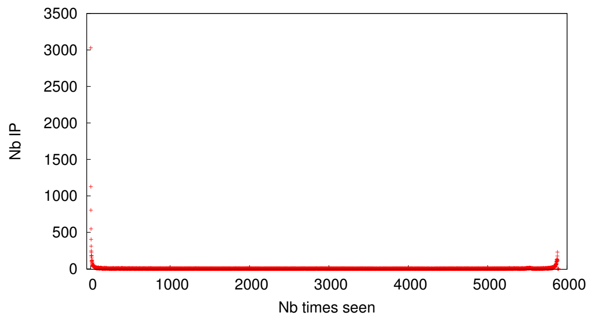
Presence of an IP

Distribution of the number of presences

All IPs seen during the measurement ($\sim$ 29 000)
For each IP: number of rounds where it is seen
$\longrightarrow$ Distribution

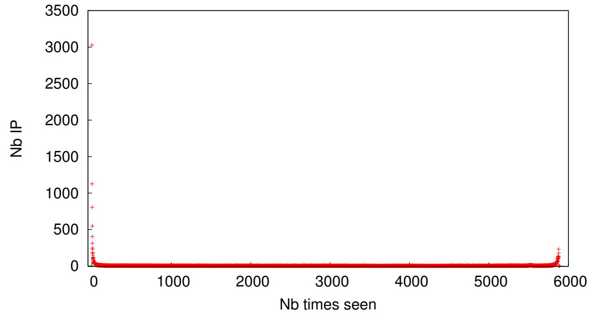
Presence of an IP

2 months $\sim$ 6000 rounds



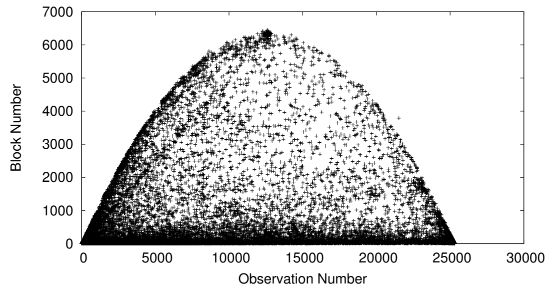
Presence of an IP

2 months ~ 6000 rounds

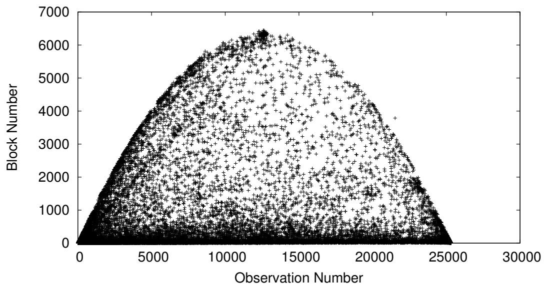


- A lot of ephemeral IPs (seen very few times)
- Substantial number of stable IPs (seen ~ every round)

A presence block = consecutive presences of an IP

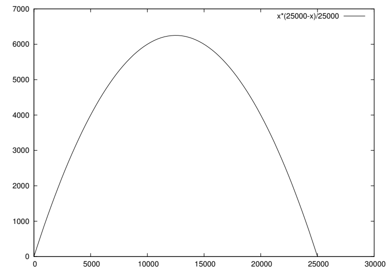


A presence block = consecutive presences of an IP



Delimiting triangle:
Nb blocks of presence $\leq$ min (nb presences, nb absences)

A presence block = consecutive presences of an IP

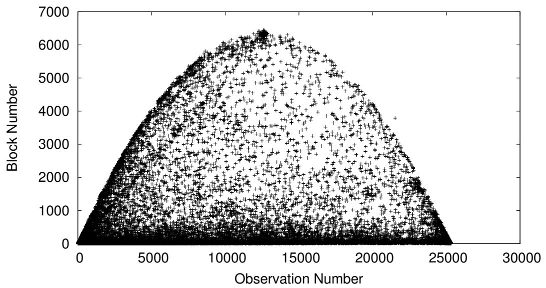


Parabola:
Nb of blocks expected on average for this number of presences

$$P(presence) = \frac{x}{25000}\ ,\ P(appearance) \propto \frac{x(25\,000-x)}{25\,000}$$

as appearance corresponds to absence then presence

A presence block = consecutive presences of an IP



Two classes of IP

random (parabola) and stable (1 block) dominate

Outline

How to explain the dynamics?

Approach in the following:

1. Choose observable features from the analysis (see above)
2. Identify underlying mechanisms related to them
3. Model these mechanisms
4. Validate using simulations

Several mechanisms may be responsible for routing changes:

"Measuring" the dynamics
Analysis of the dynamics
Modelling the dynamics
Impact of the dynamics
Modelling
Simulations

How to explain the dynamics?

Approach in the following:

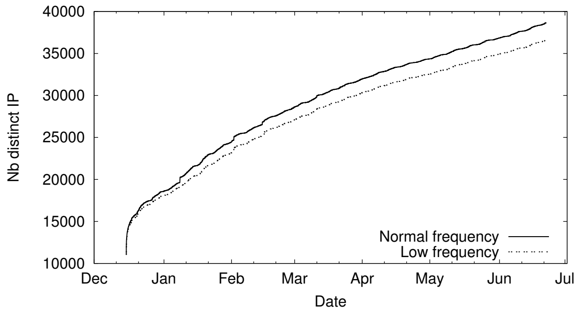
1. Choose observable features from the analysis (see above)
2. Identify underlying mechanisms related to them
3. Model these mechanisms
4. Validate using simulations

Several mechanisms may be responsible for routing changes:

- load-balancing
- routing topology modifications

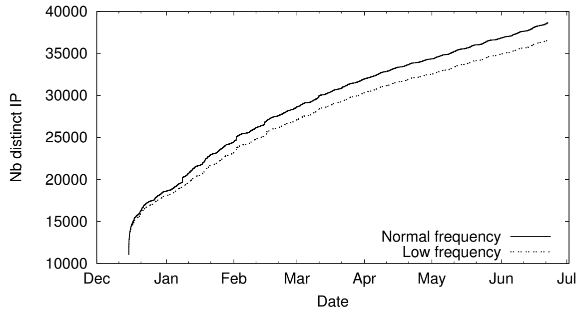
Do these mechanisms really impact our measurements?

Impact of the frequency in regard to time



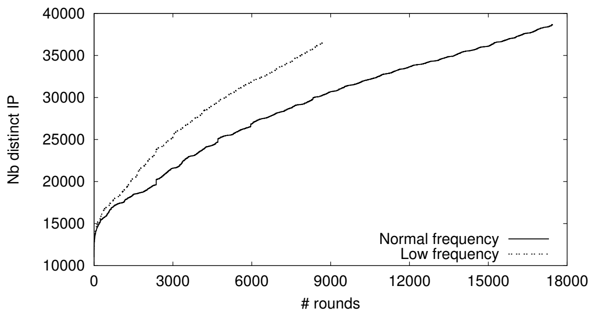
The higher the rate of measurement, the more IPs are seen

Impact of the frequency in regard to time



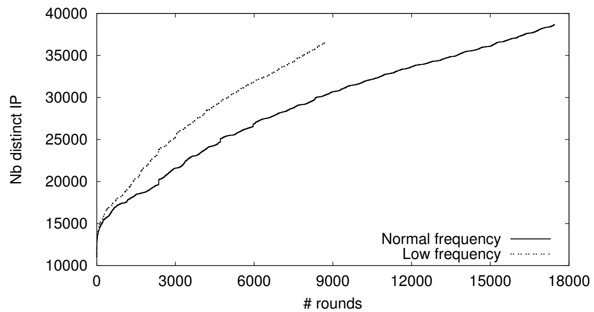
The higher the rate of measurement, the more IPs are seen
$\Rightarrow$ load-balancing

Impact of the frequency in regard to # of rounds



The longer the time between measurements, the more IPs seen

Impact of the frequency in regard to # of rounds



The longer the time between measurements, the more IPs seen
$\Rightarrow$ changes of the topology

Modelling the dynamics

We are looking for a (simple) model to account for the load-balancing and for the modifications of the routing topology

Load-balancing

Traceroute measurement as a shortest path
How to adapt the BFS to account for the load-balancing?

Topology

Which graph model? Which re-arrangement rules?
Random nodes? Random links?

Modelling traceroute and load-balancing

Several possibilities to choose a shortest path.

Given a DAG of shortest paths from a source to a destination, we can:

- either choose one path among all possible paths
- or for each of the possible following nodes, we choose one

Modelling traceroute and load-balancing

Several possibilities to choose a shortest path.

Given a DAG of shortest paths from a source to a destination, we can:

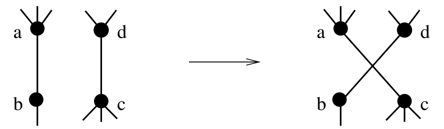
- either choose one path among all possible paths
- or for each of the possible following nodes, we choose one

Benefit of the second choice: local rule at the level of the node

Implementation: random BFS

Modelling the topology modifications

We choose to focus on the links, considering rewiring two pairs of connected nodes:

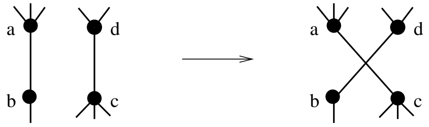


Benefit: no change in the graph degree distribution

Drawback: cannot handle appearances and disappearances of nodes in the graph

Modelling the topology modifications

We choose to focus on the links, considering rewiring two pairs of connected nodes:



Benefit: no change in the graph degree distribution

Drawback: cannot handle appearances and disappearances of nodes in the graph

Validate using simulations

Simulations approach:

1. Generate an initial graph: ER, BA, CM, others. . . => G_1
2. Simulate a measurement: extract a routing tree (random BFS) => T_1
3. Simulate a topology modification: swap => G_2
4. Iterate step 2: => $T_2; \ldots; T_n$

Questions:

- Do we observe the same phenomena in $T_1; \ldots; T_n$?
- What is the impact of the model parameters?

Validate using simulations

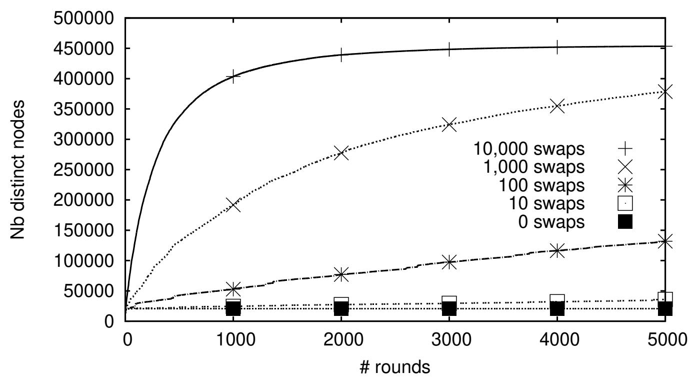
Simulations approach:

1. Generate an initial graph: ER, BA, CM, others. . . => G_1
2. Simulate a measurement: extract a routing tree (random BFS) => T_1
3. Simulate a topology modification: swap => G_2
4. Iterate step 2: => $T_2; \ldots; T_n$

Questions:

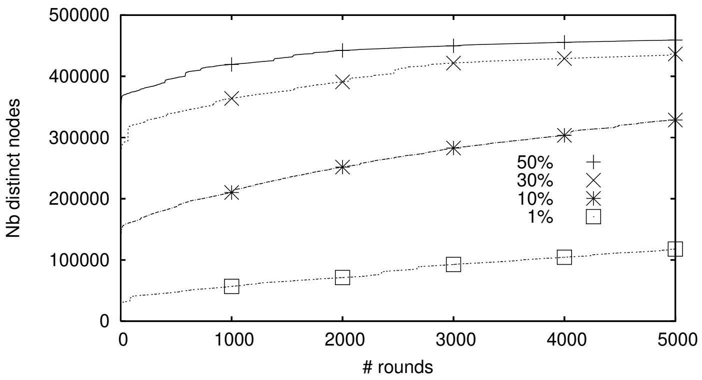
- Do we observe the same phenomena in $T_1; \ldots; T_n$?
- What is the impact of the model parameters?

Impact of the number of swaps



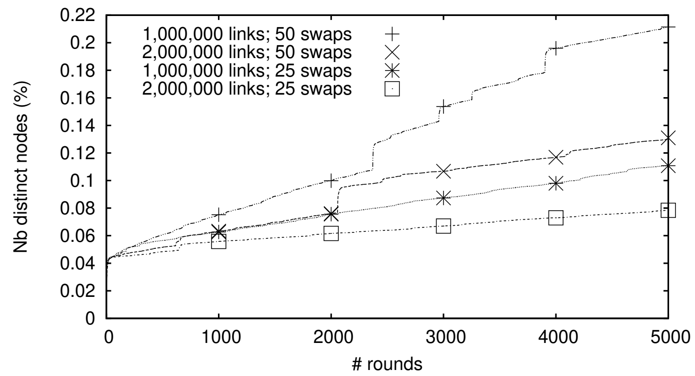
$n = 500,000$; $m = 1,000,000$; $3,000$ destinations

Impact of the number of destinations



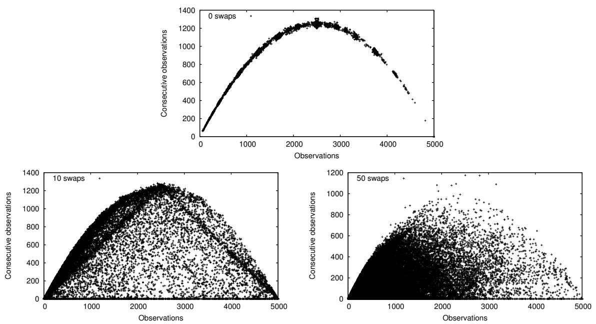
$n = 500,000$; $m = 1,000,000$; 50 swaps

Relation between the # of links and the # of swaps



$n = 500,000$; $3,000$ destinations

The parabola observation



$n = 500,000$; $m = 1,000,000$; $3,000$ destinations

Conclusions on the simulations

Global conclusions:

- we account qualitatively for the dynamics observed
- the chosen models allow to replicate the observations
- some constraints on the parameters allow to identify invariant in the dynamics
- we do not account for stable IPs

Possible follow ups:

- test other topologies
- integrate appearances and disappearances of nodes
- obtain analytical results to have a better understanding of the curves observed