Complex-Network Modelling and Inference

Lecture 24: Network Tomography

Matthew Roughan

<matthew.roughan@adelaide.edu.au>

https://roughan.info/notes/Network_Modelling/

School of Mathematical Sciences,
University of Adelaide

January 14, 2025

Section 1

Network Inference Problems

Indirect Measurements

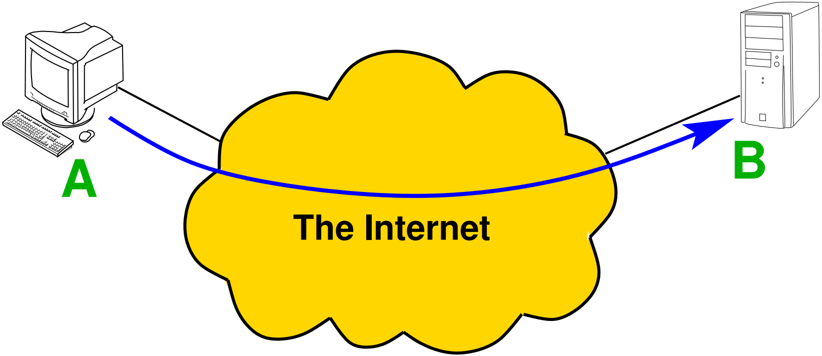
- Often we can't measure a network directly
  - don't have privileged access, *e.g.*, to routers
  - "actors" won't reliably report connections, *e.g.*, criminals
- Often we observe some proxy measurements of the network
  - instead of observing social relationships, we observe emails
- Sometimes, the proxy measurements don't even have the data we want, *i.e.*, edges

Measuring Network Performance

- We often want to know how well our (Internet) network is working
  - Internet stores packets in queues
  - hence delays
  - if queues over-flow, packets are *dropped*
- Performance metrics
  - *packet delay*
  - *packet loss rate*
  - packet jitter
  - packet reordering
  - throughput
- Network devices
  - are fairly "dumb", *i.e.*, they don't see or record their own performance, so how can we find this stuff out?
  - deliberately won't report dropped packets, *e.g.*, when they are deliberately *censoring* traffic

Active probes

- Active performance measurements
- Send probe packets from $A \rightarrow B$ across the network
- Measure, *e.g.*, the delays experienced by packets



Variations

There are lots of variations on this

- Round-trip v one-way
- What type of packet
- Passive variants
- ...

But the *key* idea is that we measure a performance metric along a whole path.

We could construct similar experiments in other *transport* networks

- delays of packages in the mail
- time for trucks to get to destinations

Question

Could we use these types of measurements somehow to reconstruct the network?

i.e., just using delays or lost packets from $A \to B$ etc, can we work out the network?

Inverse problems

- mostly in math classes we teach a technique, and then ask you to solve a problem using that technique
- In reality, problem solving involves determining which of the infinite set of available techniques, suits the problem
- This is the essence of inverse problems

Inverse problems

Characteristics

- forward problem:
  - logic is sequential: A therefore C
  - task is to use the model A to predict behaviour C
- inverse problem:
  - logic is reversed: C could result from A or B or something else?
  - very large class of possibilities
  - task is to determine which of A or B caused C
- modelling, in general, is an inverse problem
- we'll add some specifics here to make the problems soluble

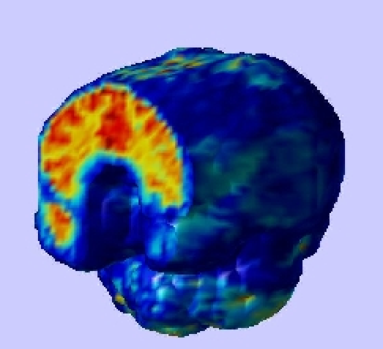
Example

- forward problem:
  - do the two sets of numbers A and B have the same sum, *i.e.*, is

$$\sum_{x \in A} x = \sum_{y \in B} y$$

- inverse problem:
  - given set of numbers C, can we divide it into two sets A and B that have the same sum
  - $\{1, 4, 5, 6, 9, 11, 14\}$

Example 2: Who put the CAT in CATscan?



- people don't like you cutting their head open!
- so indirect methods are used to peer inside
- Computer Axial *Tomography* (CAT)
    - Tomo- from the Greek **tomos** meaning "section"

Tomographic techniques are used in many areas:

- Ocean Acoustic Tomography http://www.oal.whoi.edu/tomo2.html
- Archaeology http://archaeology.huji.ac.il/ct/
- Medical Imaging http://www.triumf.ca/welcome/petscan.html
- Manufacturing http://www.tomography.umist.ac.uk/intro.shtml
- Seismology http://www.itso.ru/GEOTOMO/paper_moscow2003/index.html

There are many solution techniques.

Network Tomography

The CATscan example is a lot like our network measurements

- Indirect measurements
- We want to understand structure inside

Idea spawned a large area of research called “Network Tomography” [Var96, kcMM99, CHNY02]

Network Tomography

There are many variants, but we will think about only two.

1. Tree-based, (almost) deterministic tomography
2. Stochastic tomography on general networks
   Only if we have some spare time after the break.

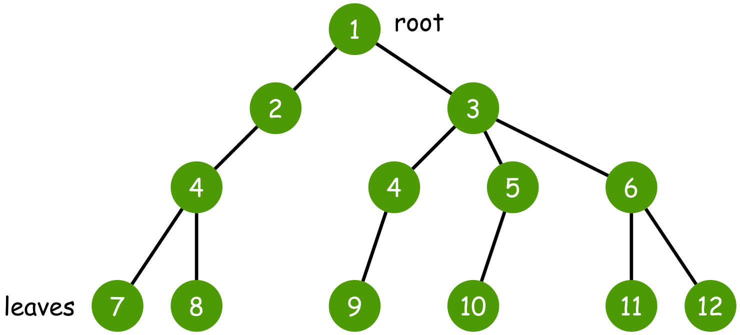
Section 2

Tree-based, deterministic tomography

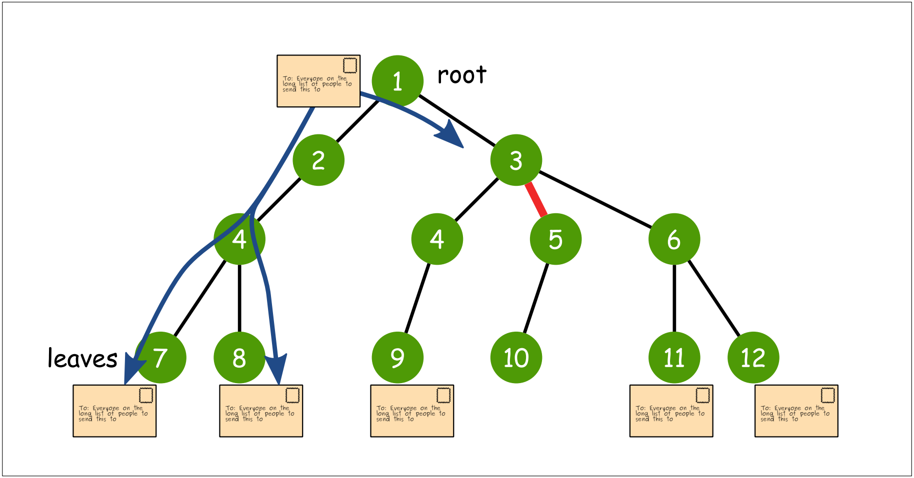
Tree-based

- Many networks are trees
- Even when the network itself is not a tree, remember that shortest-path routing forms trees (from a single source to all destinations, or visa versa)
- Assume some links or nodes are "blockages," and we want to find these
- Assume we have a *multicast* mechanism
  - a way to send a message from the root of the tree to all the leaves
  - ideally, all messages are simultaneous so we have an atomic measurement
  - we could approximate multicast in various ways (sending lots of smaller messages together) if we don't actually have such a mechanism
- Assume we can record who receives the message

Multicast



Multicast



To: Everyone on the long list of people to send this to
1
root
2
3
4
4
5
6
leaves
7
8
9
10
11
12
To: Everyone on the long list of people to send this to
To: Everyone on the long list of people to send this to
To: Everyone on the long list of people to send this to
To: Everyone on the long list of people to send this to
To: Everyone on the long list of people to send this to

Tree-based

- Starting point: given a tree, can we work out where blockages are?
  - Find an "explanation" for observations?
    - if the mechanism is correct, then there should be such an explanation, but can we find it without enumerating all possibilities?
    - is that still true if there is noise in the measurements?
  - Is there a unique explanation?
    - look at the figure carefully
- Then: can we choose between trees?

SAT

Definition (SAT)

A (Boolean) *satisfiability* (SAT) problem has n Boolean variables $x_1, \ldots, x_n$ and a Boolean formula ϕ involving the variables. The question is whether there is an assignment (of TRUE and FALSE) to the variables, such that $\phi(x_1, \ldots, x_n) = TRUE$, *i.e.*, we satisfy the formula.

Example 1:

One variable x_1 and Boolean formula

$$\phi(\mathrm{x}) = x1 \wedge \neg x1$$

where $\wedge$ = AND and $\neg$ = NOT, is *not satisfiable* because

$$\begin{aligned} TRUE\ AND\ NOT\ TRUE &= FALSE \\ FALSE\ AND\ NOT\ FALSE &= FALSE \end{aligned}$$

so there is no value of x_1 that leads to $\phi(x_1) = TRUE$.

SAT

Example 2:
Three variables x_1, x_2 and x_3 and Boolean formula

$$\phi(\mathrm{x}) = (x1 \vee \neg x2) \wedge (\neg x1 \vee x2 \vee x3) \wedge \neg x1$$

where

$$\begin{aligned} \vee &= \text{OR} \\ \wedge &= \text{AND} \\ \neg &= \text{NOT} \end{aligned}$$

is satisfied by $x1 = FALSE$, $x2 = FALSE$, and $x3$ arbitrarily.

Recast multicast problem as SAT

There are approaches to try to solve the multicast-tree problem directly, but it is more appealing to convert it into a SAT problem because

- it is a more general framework, *i.e.,* we could include other constraints into the problem
- it is a hugely studied problem, and there are very good SAT-solvers out there in free-software land
  http://www.maxsat.udl.cat/16/results/index.html

Recast multicast problem as SAT

- Each edge forms a variable x_{ij}

$$x_{ij} = \begin{cases} TRUE, & \text{if } e_{ij} \text{ is good,} \\ FALSE, & \text{if } e_{ij} \text{ is bad,} \end{cases}$$

- Each path to a successful delivery defines an expression

$$AND_{e \in P} x_e$$

- Each path to a failed delivery defines an expression

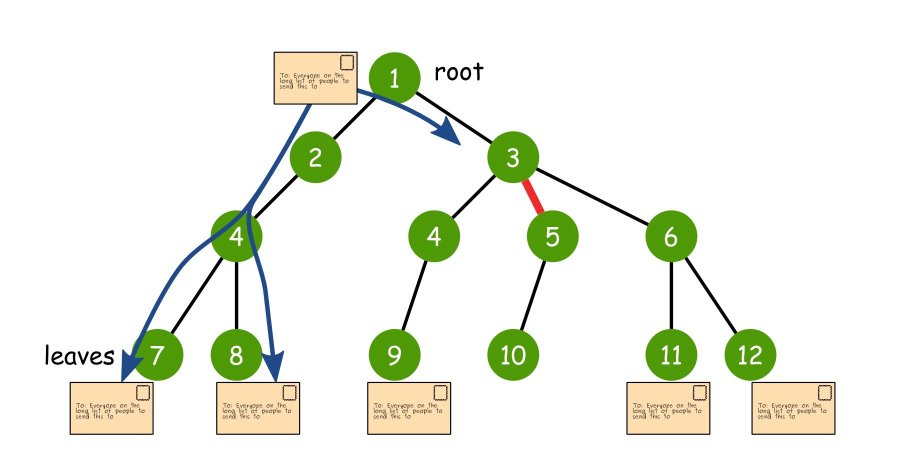
$$\neg AND_{e \in P} x_e$$

- The overall expression is an AND over all of these

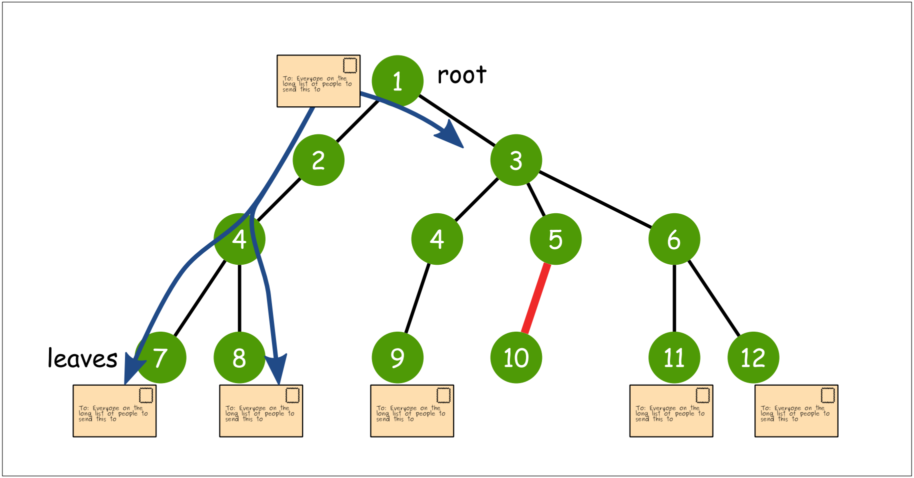
SAT

- SAT is a *decision* problem
  - it just asks us to find at least one solution
  - it's still NP-complete (the first known such)
- We need a little more than just a decision
  - # SAT or Sharp-SAT is the problem of counting all of the solutions
  - there are other variants

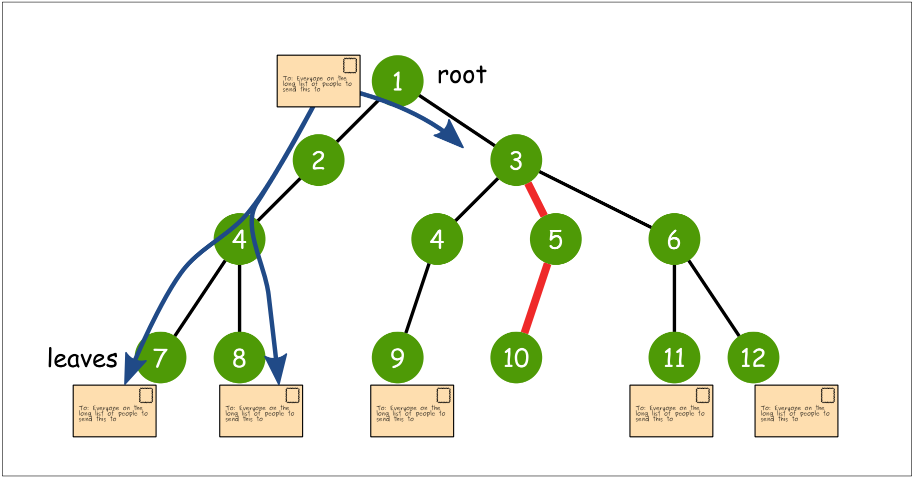
Non-uniqueness



Non-uniqueness



Non-uniqueness



Non-uniqueness

What can we do?

- Ockham's Razor

Ockham's razor

> *Pluralitas non est ponenda sine neccesitate*
>
> William of Ockham (ca. 1285-1349)

- "Plurality should not be posited without necessity."
- alternative versions
  - "Entia non sunt multiplicanda praeter necessitatem", or "Entities should not be multiplied beyond necessity"
  - "in vain we do by many which can be done by means of fewer"
  - "if two things are sufficient for the purpose of truth, it is superfluous to suppose another"
  - Principle of Parsimony

Quidquid latine dictum sit, altum viditur.

Non-uniqueness

What can we do?

- Ockham's Razor
- Use "churn"

Uniqueness via Churn + and Application

Application: locating censorship on the WWW [CNRG17]

- Internet is a key mode of free speech, and open dissemination of information, but not all governments agree with those ideas, and not all corporations want to provide open access
- We know some Internet content is censored
  - often it is done by "breaking" the network
- Can we detect where censorship is happening?

Censorship model

- *Nodes* are "autonomous systems"
  - think of them as a network operator like Telstra
  - nodes are where the censorship happens (not edges)
- *Edges* are the connections between ASs
  - note that there can be many physical edges, but they are represented by one logical edge
- *Measurements:* observe from a "vantage point" outwards (effectively creating a tree)
- Assumptions
  - not all traffic is censored
  - so we can see the routes

Churn

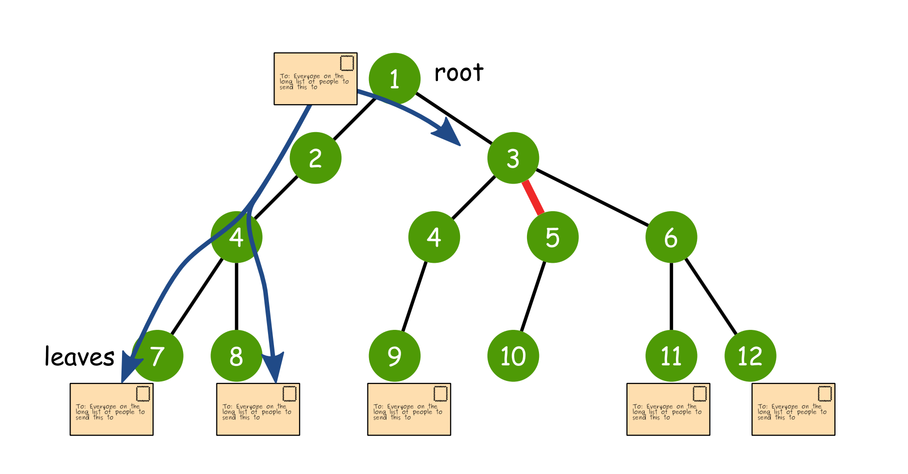
Internet routing "churns", *i.e.*, it changes regularly

- normally this is a problem
- here it is an advantage

Simply, as routes change, the measurements will change, and we get more constraints. More constraints means we are more likely to get a unique solution.

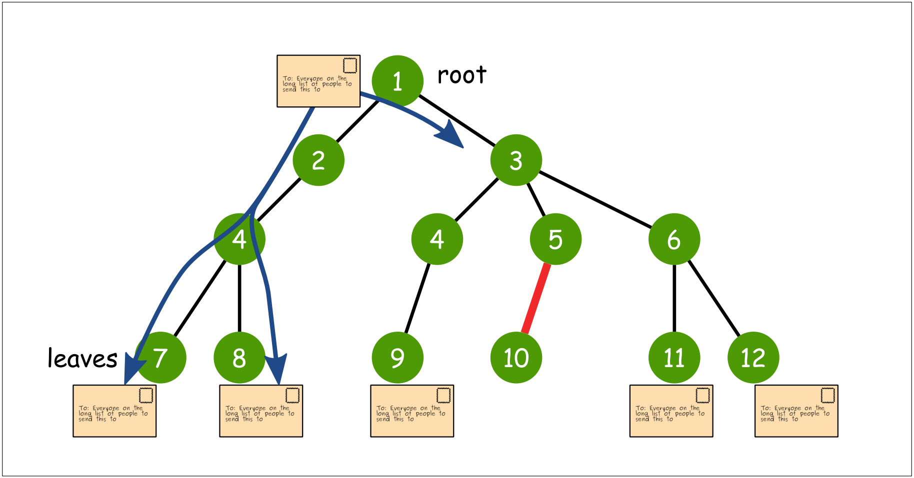
Churn

Case 1 (the real case)



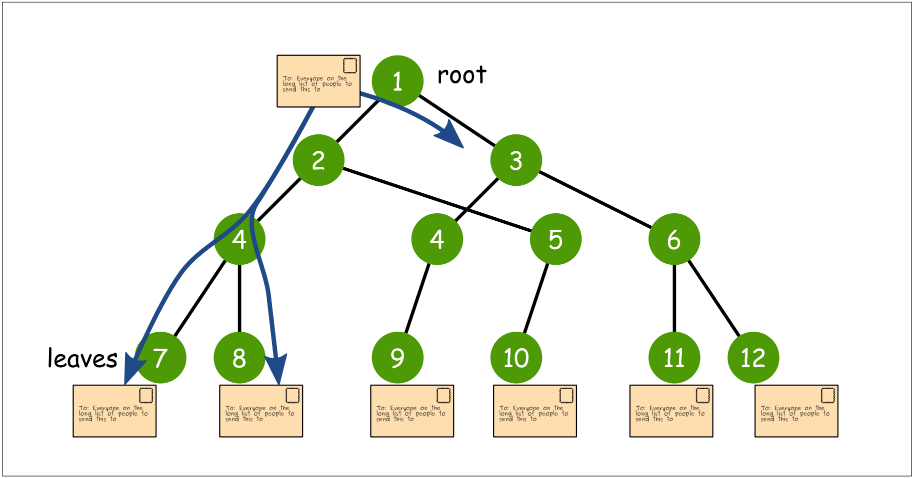
Churn

Case 2 (alternative hypothesis)



Churn

Routing change



Churn

[CNRG17] showed that in the censorship problem churn could reduces uncertainty in the number of censoring ASs by 95%

Tree-inference

- The above assumed we knew the routing/tree
- What can we do if we don't? Can we infer the tree?
- Not from a single experiment, but if we can conduct many we might have some hope
    - look into approaches next

Further reading I

- M. Coates, A. Hero, R. Nowak, and B. Yu, *Internet tomography*, IEEE Signal Processing Magazine (2002).
- Shinyoung Cho, Rishab Nithyanand, Abbas Razaghpanah, and Phillipa Gill, *A churn for the better: Localizing censorship using network-level path churn and network tomography*, CoNext, December 2017.
- k.c. claffy, T.E. Monk, and D. McRobb, *Internet tomography*, Nature: web matters (1999), `http://www.nature.com/nature/webmatters/tomog/tomog.html?foxtrotcallback=true`.
- Y. Vardi, *Network tomography: estimating source-destination traffic intensities from link data*, J. Am. Statist. Assoc. **91** (1996), no. 433, 365–377.