Communications Network Design

lecture 09

Matthew Roughan
<matthew.roughan@adelaide.edu.au>

Discipline of Applied Mathematics
School of Mathematical Sciences
University of Adelaide

March 2, 2009

The Network Design Problem

In this lecture we consider a new optimization problem, the network design problem, where we can choose the network links (in contrast to routing where we only chose the routes across a given network). In this lecture we present some basics such as **star-like** topologies, **ring** topologies and the **travelling salesman's problem**.

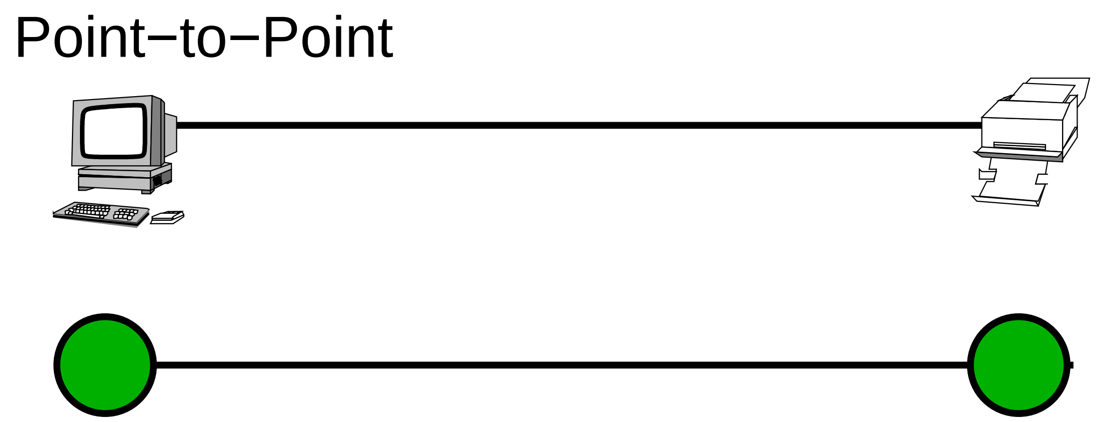
Network Design Problem

- the problems so far have concerned routing
  - network is given
  - we need to find optimal routing
- now we want to consider how to design the network
  - from scratch
  - routing is part of the design
- inputs
  - a set of nodes (locations)
  - forecasts of traffic demands

Example topologies

- point-to-point
- linear or bus
- ring
- hub and spoke or star
- double star
- fully connected (mesh) or complete topology or clique
- mesh
- (spanning) tree
- hybrid

Point-to-point



description: back-to-back connection of two nodes

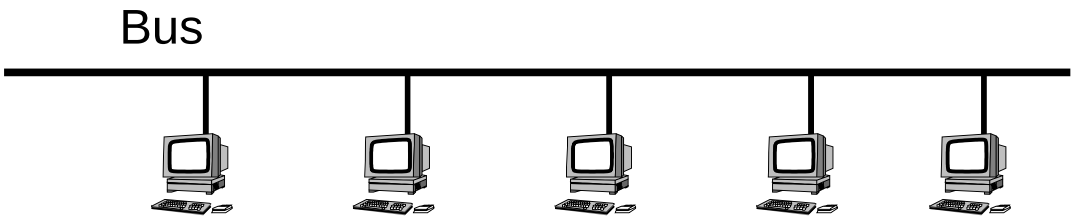
examples:

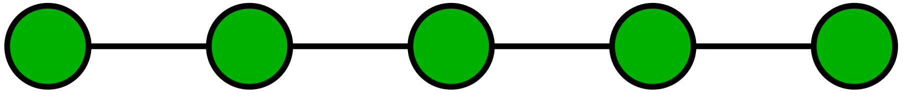
- (old fashioned) printer connection
- serial link
- PPP (Point-to-Point Protocol)

comments:

- used as a component of a larger network

Bus





description: a single line (the bus) to which all nodes are connected, and the nodes connect only to this bus.

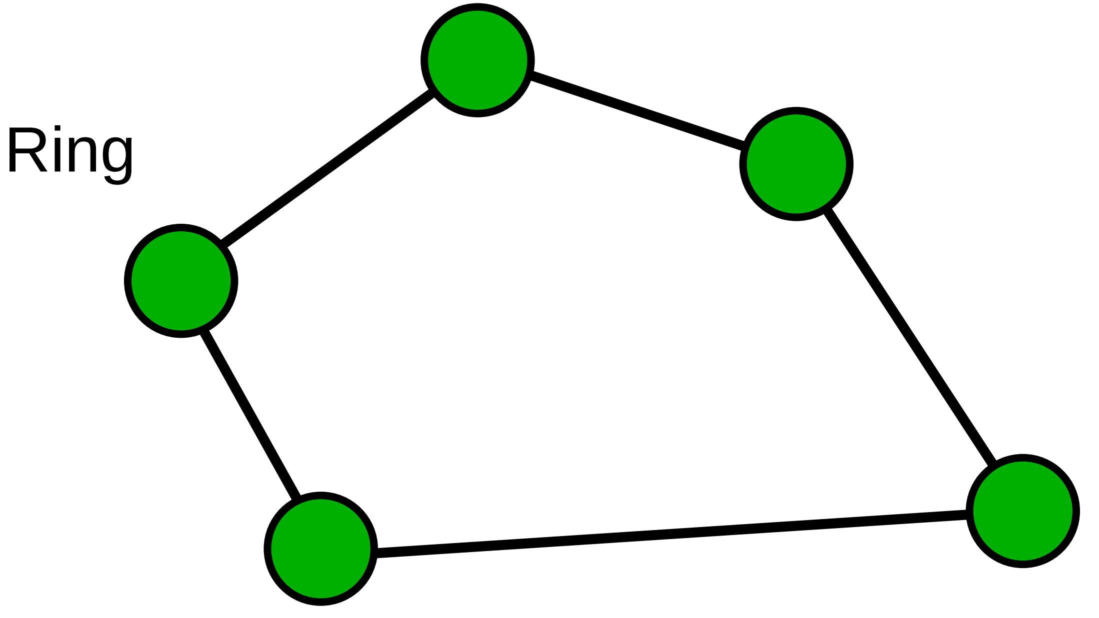
examples:

- physical structure of 10Base2 Ethernet
- logical structure of 10BaseT Ethernet with a hub

comments:

- design often matches a building (corridors)
- no redundancy (failures effect whole network)

Ring



description: Every node has exactly two branches connected to it, so that they form a (logical) ring.

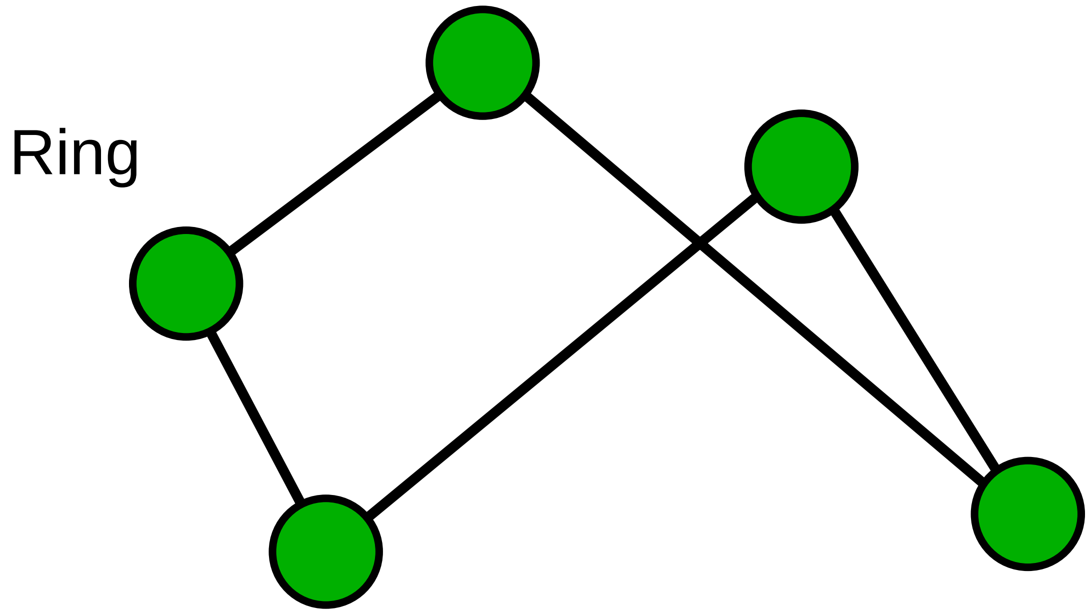
example:

- SONET, FDDI, Token Ring

comments:

- two paths provide some redundancy (a dual ring)

Ring



description: Every node has exactly two branches connected to it, so that they form a (logical) ring.

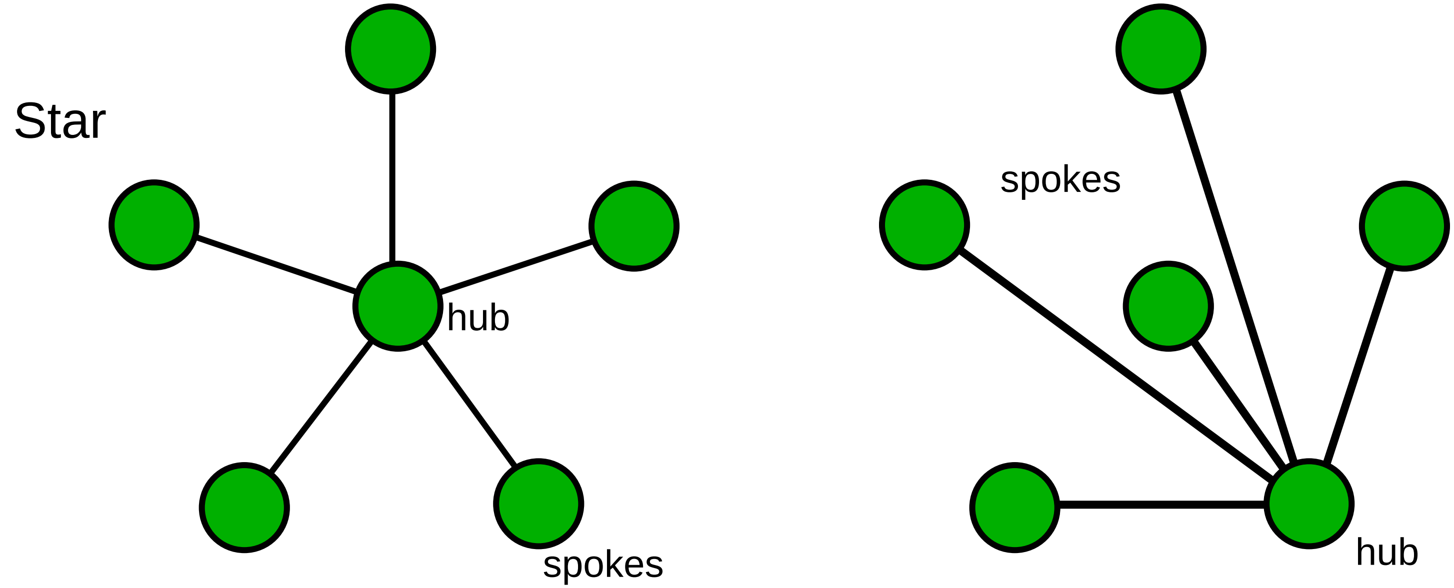
example:

- SONET, FDDI, Token Ring

comments:

- two paths provide some redundancy (a dual ring)

Star



description: peripheral (spoke) nodes are connected to a central (hub) node. All communications is via the hub.

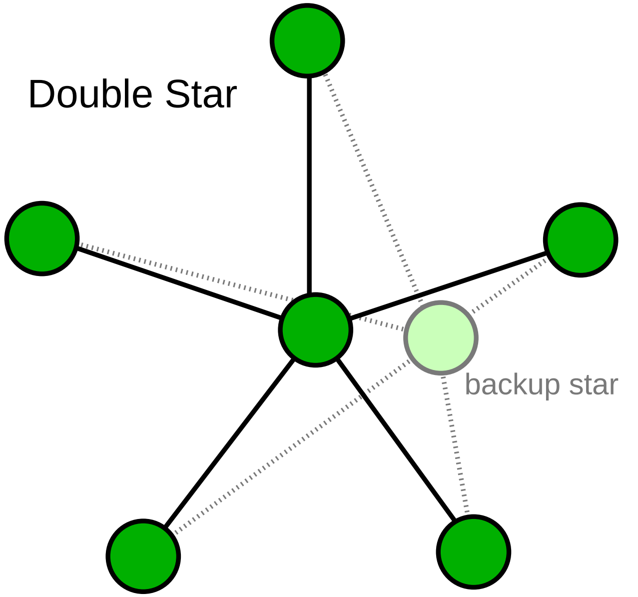
examples:

- physical topology of 10BaseT Ethernet with a hub
- logical topology of 10BaseT Ethernet with a switch

comments:

- hub node failures are critical

Double star



description: two stars, with two hubs, effectively, one is a redundant backup for failures.

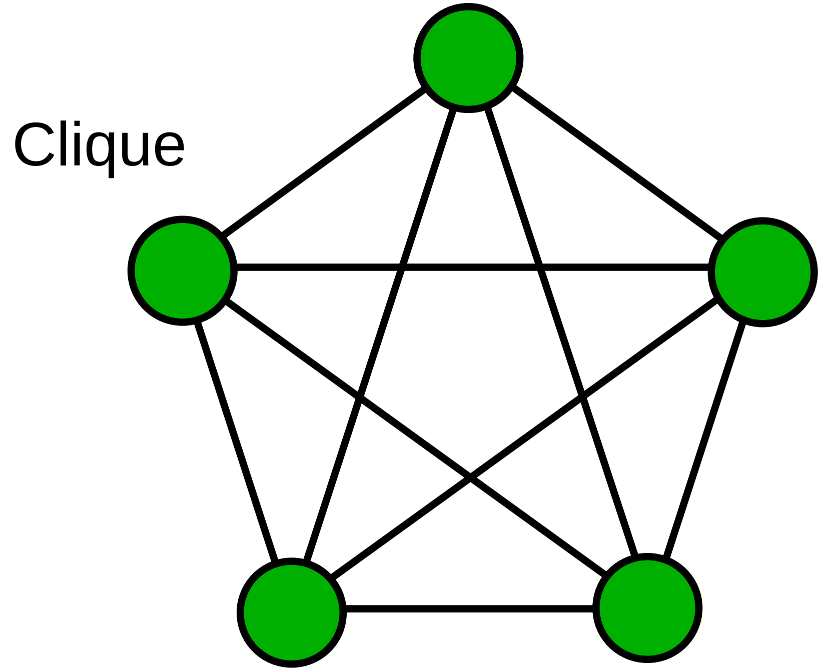
example:

- used for many networks

comments:

- stars are sensitive to failures of hub, or links
- robust to a failure of hub, or single link

Fully connected



description: every node directly connected to every other node (also called a clique).

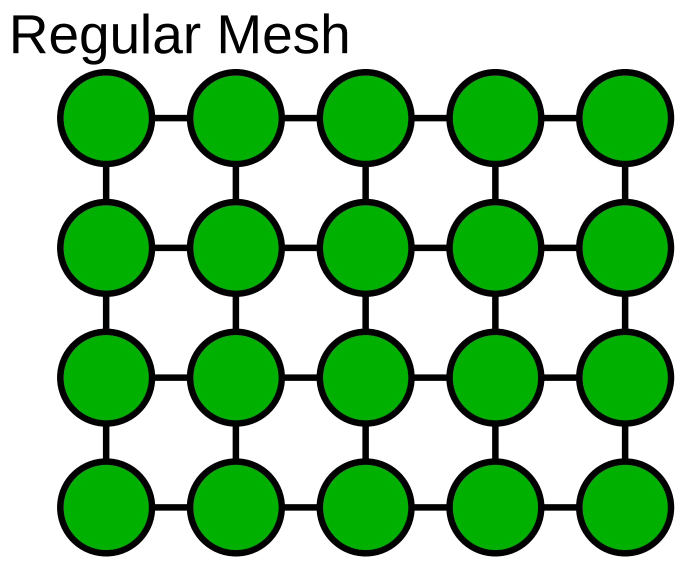
example:

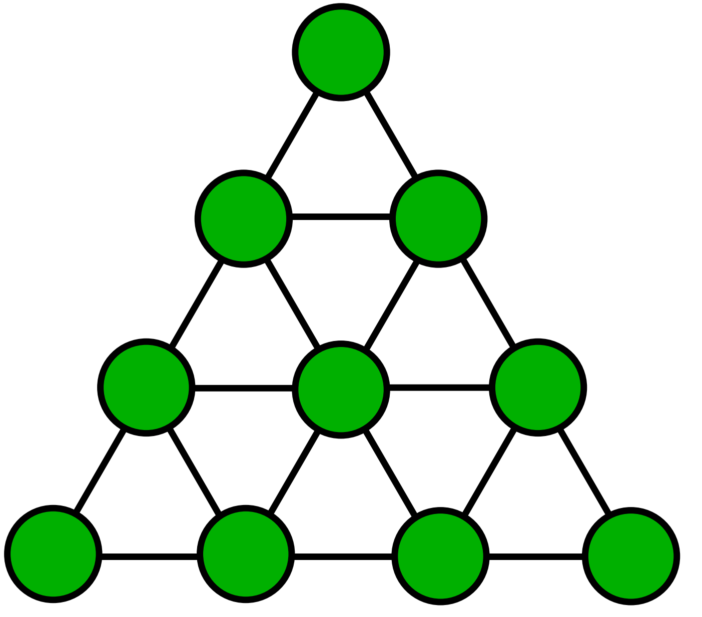
- frame relay network (at a logical level)

comments:

- very robust to failures

Mesh





description:

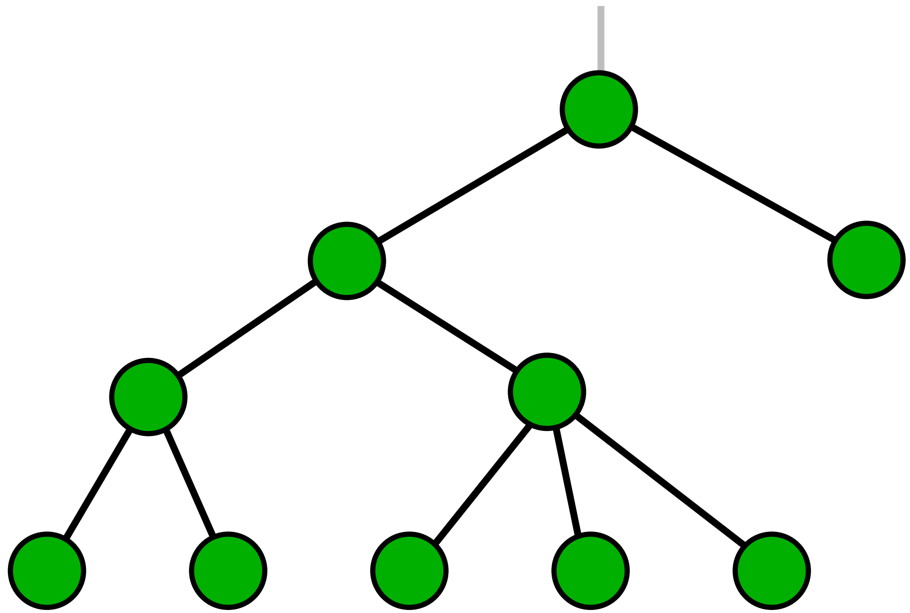
example:

- many real networks are somewhat meshy

comments:

- somewhere between clique, and star
- robust to failures

Tree



description: nodes are arranged as a tree (no loops)

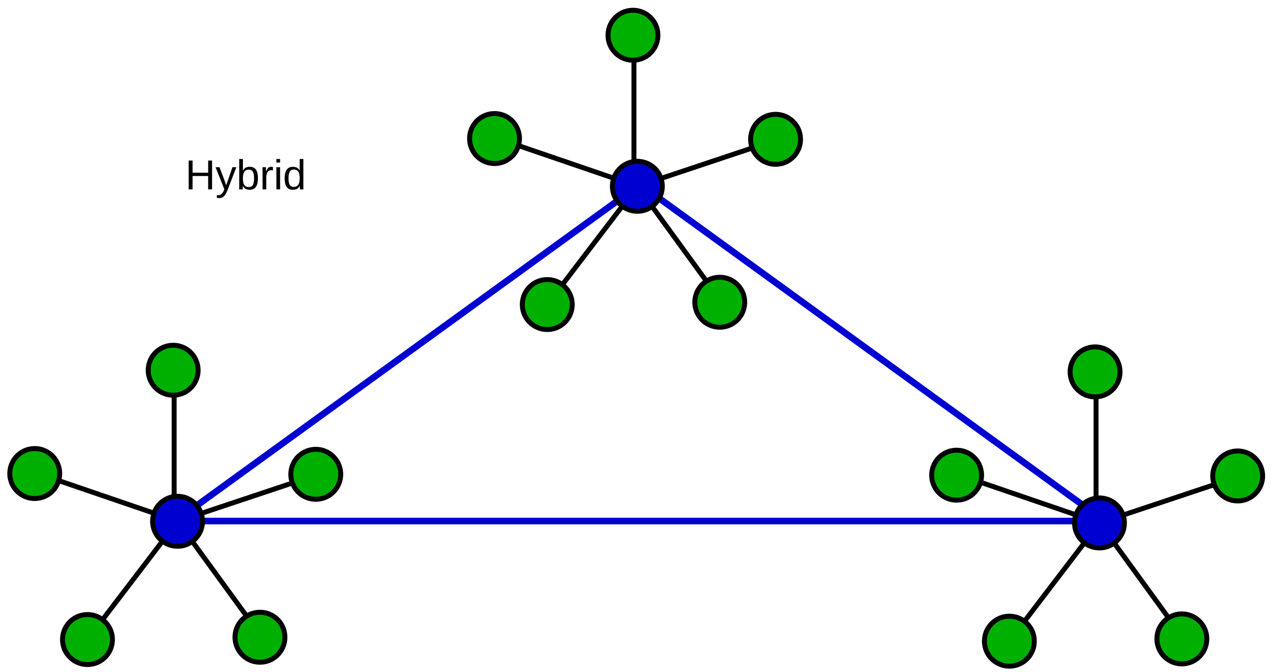
examples:

- shortest path trees in routing
- spanning tree protocol (for switched Ethernets)

comments:

- sensitive to failures

Hybrid



description: A combination of any two or more network topologies in such a way that the resulting network does not have one of the standard forms.

comments:

- a tree connected to a tree is still a tree network
- example is a hierachical network (as above)

Notation recap

Mostly as before (lecture 6)

- A **network** is a graph $G(N,E)$, with **nodes** $N = \{1,2,\ldots n\}$ and **links** $E \subseteq N \times N$
- Offered traffic between O-D pair (p,q) is t_{pq}
- The set of all **paths** in $G(N,E)$ is $P = \cup_{[p,q] \in K} P_{pq}$
- Each link $e \in E$ has
  - a **capacity**, denoted by $r_e (\geq 0)$
  - a **distance** $d_e (\geq 0)$
  - a **load** $f_e (\geq 0)$
- The vector $\mathbf{x} = (x_\mu : \mu \in P)$ is called the **routing**

$$f_e = \sum_{\mu \in P : e \in \mu} x_\mu$$

Primitive network design

- assume network nodes and edges are given

$$G = (N, E)$$

- find optimal routing $\mathbf{x}$, ignoring capacity constraints

Formulation: minimize $C(\mathbf{f})$ s.t.

$$\begin{aligned} f_e &= \sum_{\mu \in P : e \in \mu} x_\mu, & \forall e \in E \\ x_\mu &\geq 0, & \forall \mu \in P \\ \sum_{\mu \in P_{pq}} x_\mu &= t_{pq}, & \forall [p,q] \in K \end{aligned}$$

- use loads given by routing to obtain capacities, e.g.

$$r_e = f_e, \ \forall e \in E$$

More generally

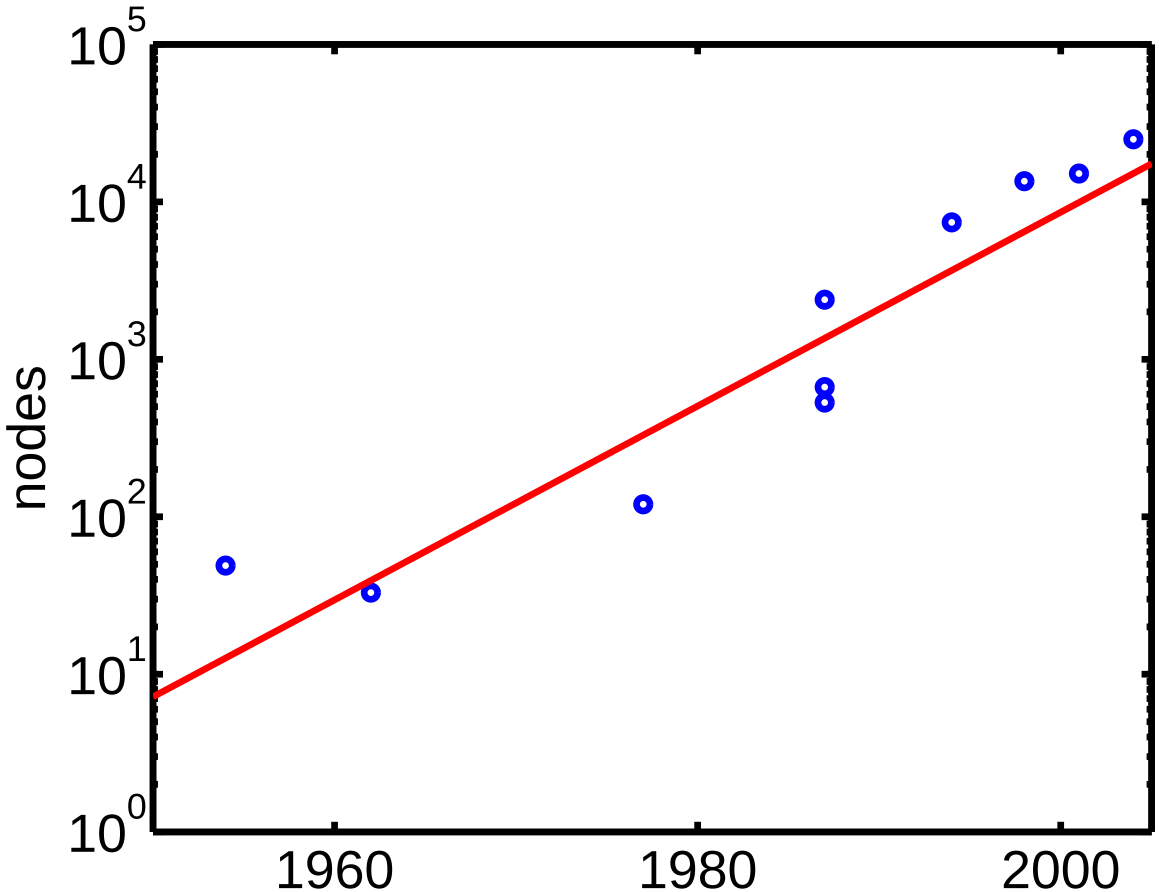
- only network nodes are given
- we must decide edges as well as nodes
- routing is part of this
  - often assume shortest (physical) path routing
- in other design problems, even the nodes aren't given
  - e.g. cellular mobile phone network
  - we are not considering these cases in this course
- costs include
  - construction costs based on capacities r_e
  - performance costs (e.g. delays, reliability, ...) based on r_e and f_e

Minimal cost ring

- minimum cost path that visits each node exactly once, and returns to the start
- consider case where cost is linear in distance
  - minimum cost ring is the shortest ring
  - traveling salesman problem [1, 2, 3]
    - find the shortest tour between N nodes
    - e.g. a travelling salesman has to visit N cities (exactly once each), with the minimum travel distance, and return to his start point.
  - NP-complete or NP-hard (Non-Polynomial)
    - settle P versus NP problem and fetch a $1,000,000 prize

Travelling Salesman Computations

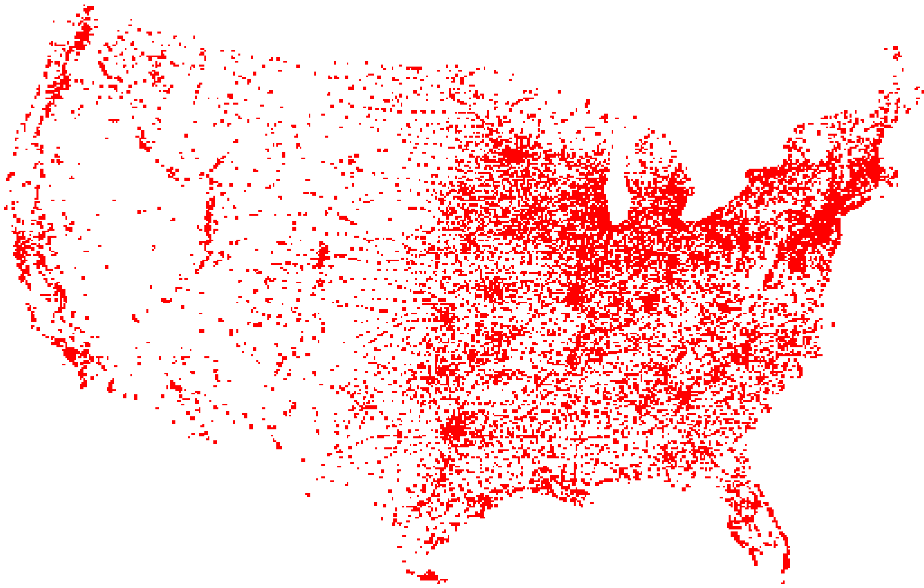
largest solvable problem has doubled in ∼5 years [4]

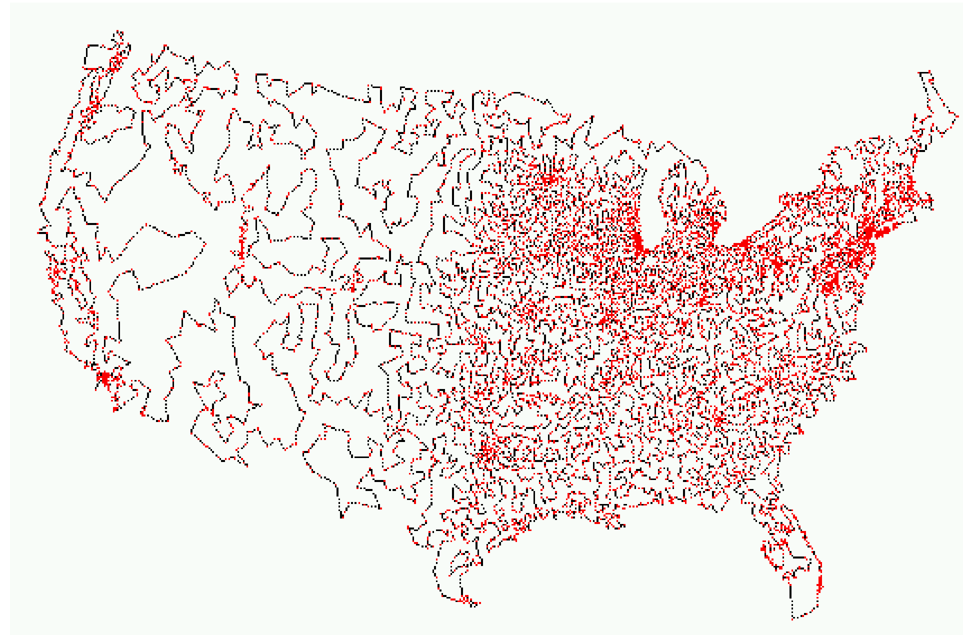


Current, can do $\sim 20,000$ nodes which is big enough for most networks, but not fast, or easy.

Travelling Salesman Example

13,509 nodes



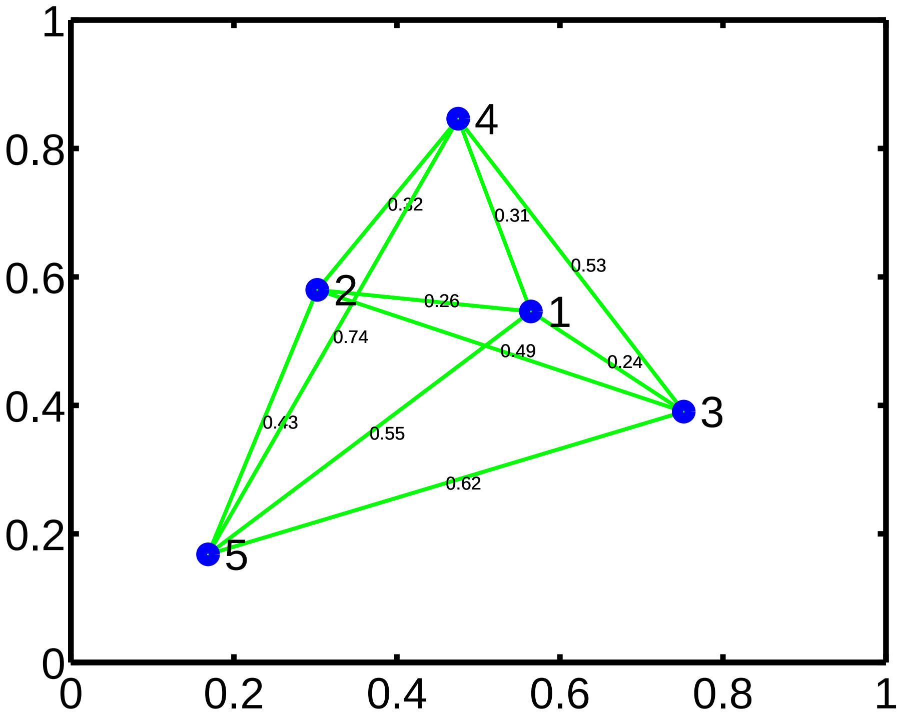


http://www.tsp.gatech.edu/gallery/idata/usa13509.html

Minimal cost star

- all we need to do is choose the hub
- assume cost are linear in distances
- either compute or are given the distances between each pair of nodes
- simple calculate all column (or row) sums, and find the minimum
  - this gives the hub
  - only one routing is possible
  - compute capacities as for primitive case above
- complexity $O(N^2)$ which is pretty good
  - compared to NP-hard

Minimal cost star: example

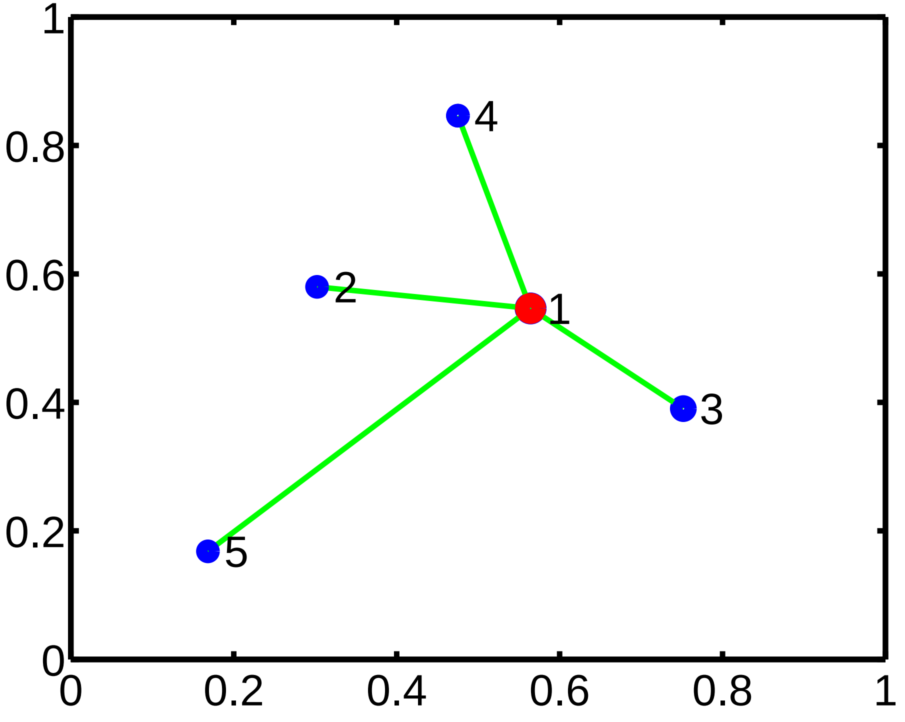


distances

$$\begin{pmatrix} 0.00 & 0.26 & 0.24 & 0.31 & 0.55 \\ 0.26 & 0.00 & 0.49 & 0.32 & 0.43 \\ 0.24 & 0.49 & 0.00 & 0.53 & 0.62 \\ 0.31 & 0.32 & 0.53 & 0.00 & 0.74 \\ 0.55 & 0.43 & 0.62 & 0.74 & 0.00 \end{pmatrix}$$

sums

Minimal cost star: example



distances

$$\begin{pmatrix} 0.00 & 0.26 & 0.24 & 0.31 & 0.55 \\ 0.26 & 0.00 & 0.49 & 0.32 & 0.43 \\ 0.24 & 0.49 & 0.00 & 0.53 & 0.62 \\ 0.31 & 0.32 & 0.53 & 0.00 & 0.74 \\ 0.55 & 0.43 & 0.62 & 0.74 & 0.00 \end{pmatrix}$$

sums 1.37 1.50 1.89 1.91 2.35

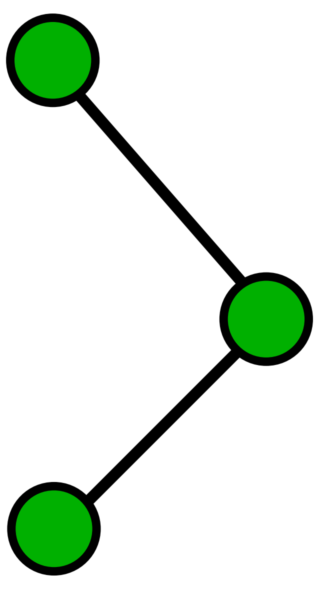
- Node 1 is has the minimal column sum.
- Hence Node 1 is the hub

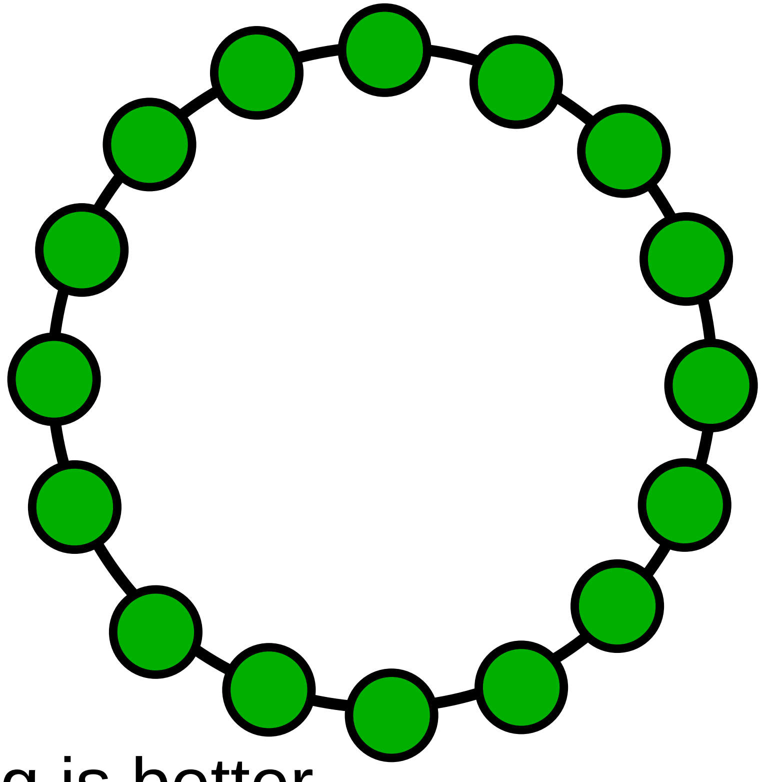
Minimal cost star

- stars are used a lot
  - particularly at layer-2
    - Ethernets commonly use stars (at some level)
      - put stars together to get a tree
  - good where traffic matrix is not known
    - see later for why
- note often dual stars for reliability
  - backup star may be passive or active
    - active = load sharing
- not just used in comm.s networks
  - hub airports in the US

Which is better

- both very simple (conceptually)
- very different computationally
- a star or a ring can be better in some cases
- neither is truly optimal





References

[1] A. Schrijver, “On the history of combinatorial optimization (till 1960).” http://homepages.cwi.nl/~lex/.

[2] “Traveling salesman problem.” http://www.tsp.gatech.edu/index.html.

[3] D. Applegate, R. Bixby, V. Chvátal, and W. Cook, Computational Combinatorial Optimization, ch. TSP cuts which do not conform to the template paradigm, pp. 261–304. Springer, 2001.

[4] “Milestones in the solution of TSP instances.” http://www.tsp.gatech.edu/history/milestone.html.