Routing

COMP 429

Kevin Scrivnor

Spring 2023

Previously

- We looked at

- IP addressing and how a campus LAN can be represented on the global Internet
- How subnetting and supernetting help with the routing table problem
- The idea that packets are forwarded by routers along some path

- This week we want to look at how the routing actually happens

- At this point, we can almost diagram how the Internet works

NAT/Traceroute Review

Explanation of things in the previous two labs

Routing and Forwarding

- The routing algorithm
  - Responsible for filling and maintaining the routing table
- Forwarding
  - The process of handling each packet as it arrives, looking up the outgoing line to use in the routing table

- Example from the last lab
  - Each NUC has a routing table (ip route)
  - This table can change if an interface is brought up or down
  - Pretty much statically generated based on the information you provide

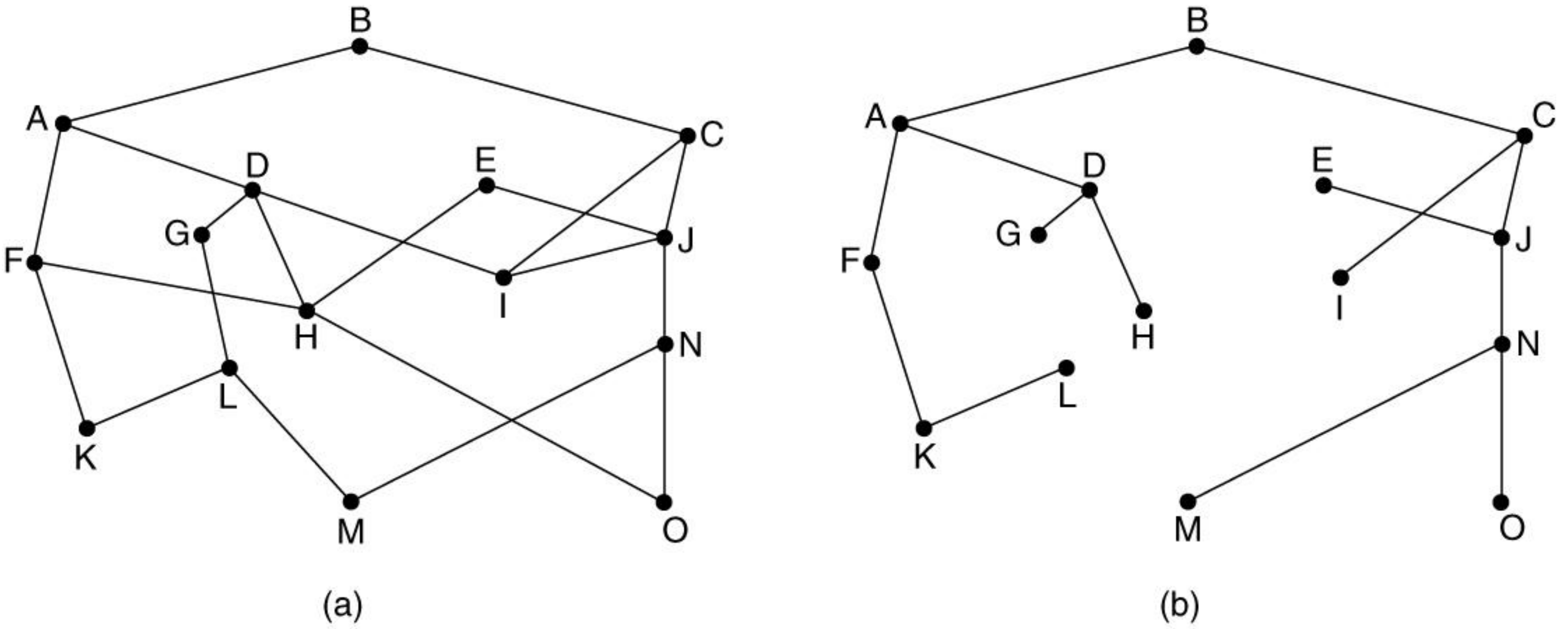
```
ETHERNET

IPv4 CONFIGURATION <Manual>
         Addresses 192.168.56.100/24________ <Remove>
                   <Add...>
           Gateway 192.168.56.1_____________
```

Two major classes of routing algorithms

- Nonadaptive algorithms
  - Statically generated offline and downloaded to the router later
  - Mostly useful in situations where the route is clear and obvious
- Adaptive algorithms
  - Change their routing tables based on changes in the topology or traffic
  - Uses many different metrics to determine best route
  - Number of hops
  - Load changes
  - Estimated transit time
  - Distance
  - Cost

The goal is to generate sink tree's for all nodes on the network

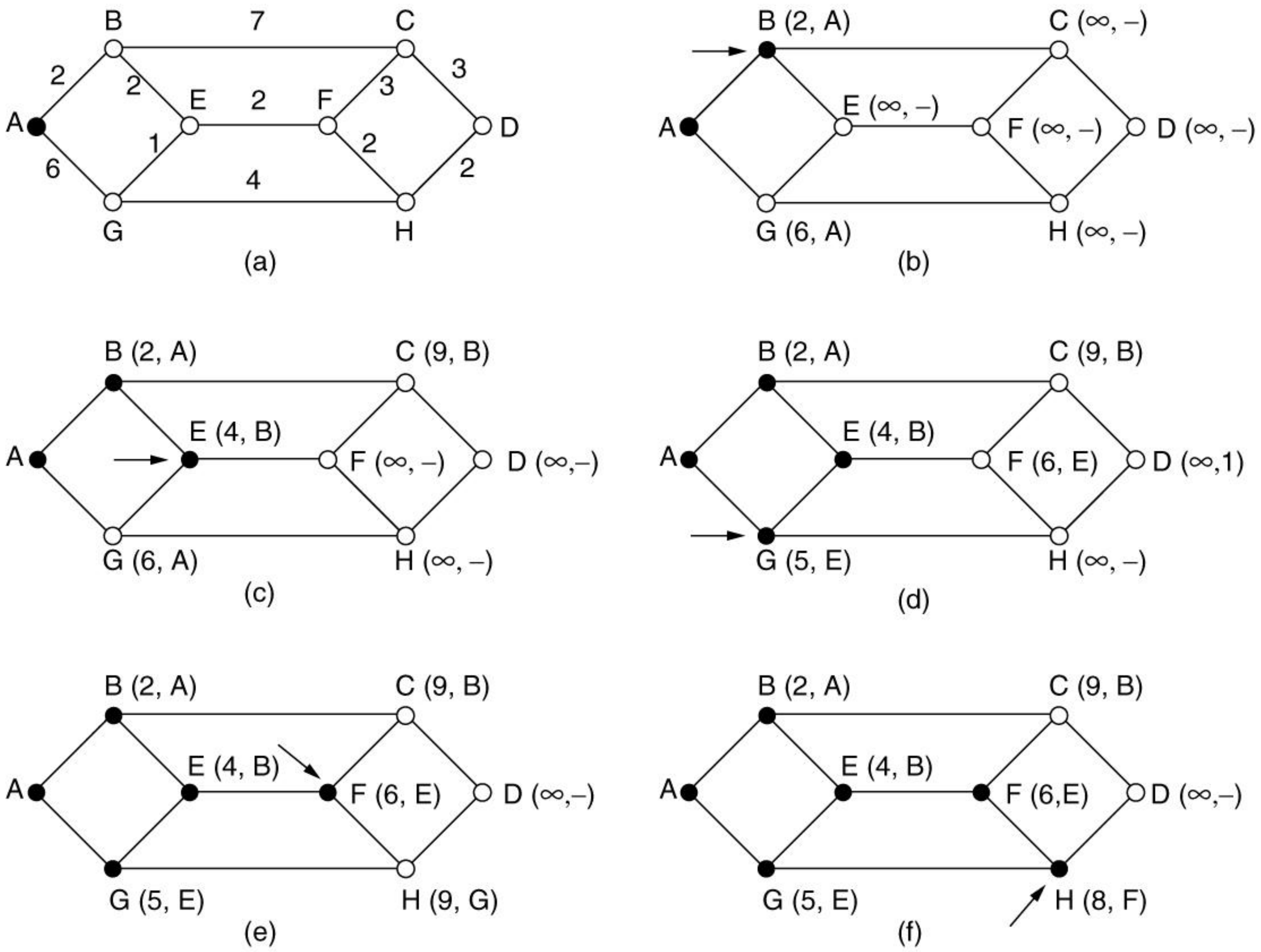
- Benchmark for a router algorithm



(a) A network. (b) A sink tree for router *B*.

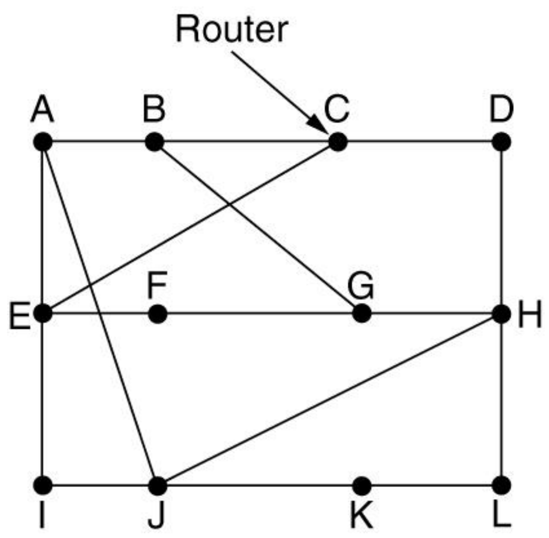
Shortest Path Algorithm

- First six steps of the shortest path algorithm
- The metric used here is distance in km
- For instance, A is 2km away from B, but 6km away from G
- If the metric were average throughput, the shortest path would be the fastest path



Distance Vector Routing

- J would **update** its **routing table** based on its **neighbors** information.
- The neighboring nodes to J are A, I, H, and K.
- Delay from *J to A is 8*, *I is 10*, *H is 12*, *K is 6* msecs.
- A’s table explained:
  - **12 msec to B**, **25 msec to C** and so on.



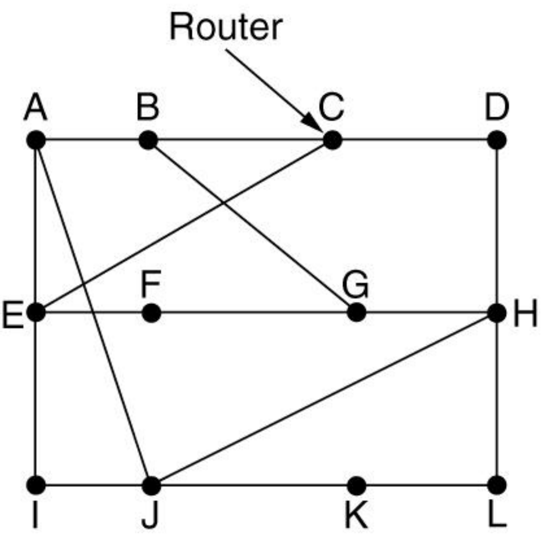
(a)

To	A	I	H	K	New estimated delay from J	Line
A	0	24	20	21	8	A
B	12	36	31	28	20	A
C	25	18	19	36	28	I
D	40	27	8	24	20	H
E	14	7	30	22	17	I
F	23	20	19	40	30	I
G	18	31	6	31	18	H
H	17	20	0	19	12	H
I	21	0	14	22	10	I
J	9	11	7	10	0	–
K	24	22	22	0	6	K
L	29	33	9	9	15	K
	JA delay is 8	JI delay is 10	JH delay is 12	JK delay is 6	New routing table for J	

Vectors received from J's four neighbors

(b)

Distance Vector Routing Example



(a)

To	A	I	H	K	New estimated delay from J	Line
A	0	24	20	21	8	A
B	12	36	31	28	20	A
C	25	18	19	36	28	I
D	40	27	8	24	20	H
E	14	7	30	22	17	I
F	23	20	19	40	30	I
G	18	31	6	31	18	H
H	17	20	0	19	12	H
I	21	0	14	22	10	I
J	9	11	7	10	0	–
K	24	22	22	0	6	K
L	29	33	9	9	15	K
	JA delay is 8	JI delay is 10	JH delay is 12	JK delay is 6	New routing table for J	

Vectors received from J's four neighbors

(b)

- J computes a new route to each node. Let's say J -> G.
- **J** can get to **A in 8 msec** and **A** can get to **G in 18 msec**, so it would take (8 + 18) = **26 msec** to get to G.
- Compute for the rest of the neighbors:
  - J -> G *through A* is 8 + 18 = **26**
  - J -> G *through I* is 10 + 31 = **41**
  - J -> G *through H* is 12 + 6 = **18**
  - J -> G *through K* is 6 + 31 = **37**
  - **Clearly, H is the fastest route.**

Count to infinity problem

- Assume A is initially down.
- (a) A comes **up** late, good news propagates fast!
- (b) A goes **down** again, bad news propagates fast!
- Issue: X tells Y it has a path but Y has no way of knowing if itself is part of that path.

A	B	C	D	E	
	•	•	•	•	Initially
	1	•	•	•	After 1 exchange
	1	2	•	•	After 2 exchanges
	1	2	3	•	After 3 exchanges
	1	2	3	4	After 4 exchanges

(a)

A	B	C	D	E	
	1	2	3	4	Initially
	3	2	3	4	After 1 exchange
	3	4	3	4	After 2 exchanges
	5	4	5	4	After 3 exchanges
	5	6	5	6	After 4 exchanges
	7	6	7	6	After 5 exchanges
	7	8	7	8	After 6 exchanges
		⋮			
	•	•	•	•	

(b)

Link State Routing

- ARPANET used Distance Vector Routing until 1979
  - The problems with count-to-infinity and the time it took to converge on a route would take too long
- Variants of the Link State Routing are called
  - IS-IS Intermediate System-Intermediate System
  - OSPF Open Shortest Path First
  - These variants are widely used today
- In essence the algorithm does the following
  1. Discover neighbors and learn network address
  2. Set the distance/cost metric to each neighbor
  3. Construct a packet telling all it has learned
  4. Send/receive this packet to/from other routers
  5. Compute the shortest path to each router

Discovering Neighbors

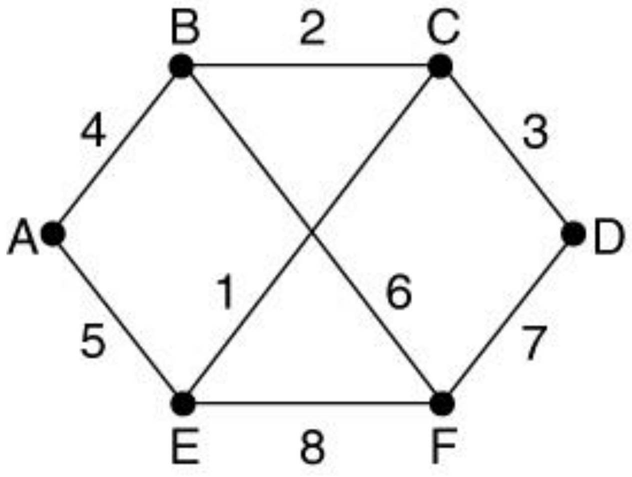
- When the router is first brought online, it constructs a HELLO packet
- The packet is then broadcast to all the neighboring routers
- The router on the other end replies giving its name

Setting Link Costs

- Typically geographical distance isn't an issue
- Numbering system may be used where 10 Gbps is 1 and 100 Mbps line costs 10
- An ECHO packet can be sent to the neighbor
  - Dividing how long it takes to get the reply by 2 gives you one way latency time

Building the Link State Packets

- Identity, sequence number, age, list of neighbors
- Question: when to build/send?



(a)

Link State Packets

A	B	C	D	E	F
Seq.	Seq.	Seq.	Seq.	Seq.	Seq.
Age	Age	Age	Age	Age	Age
B 4	A 4	B 2	C 3	A 5	B 6
E 5	C 2	D 3	F 7	C 1	D 7
	F 6	E 1		F 8	E 8

(b)

- (a) A network. (b) The link state packets for this network.

Distributing The Packets

- This is actually the hardest part
- The idea is to use flooding, but flooding has some issues
    - Duplicates
    - Loops
    - Infinite packets
- Counteracting flooding
    - Each LS packet has a sequence number
    - Routers keep track of each sequence number, source router
    - When a new LS packet arrives, it compares it to a list of packets already seen
        - If it is new, forwards to all lines except the one it came in on
        - If is a duplicate or the sequence number is lower than the highest that has arrived, it is discarded

Issues

- Sequence numbers are 32 bit
  - What does the router do with LS packets that have a lower sequence number than the highest that has arrived?
  - Solution: in reality, this is unlikely to ever occur
  - If a router sends an LS packet every second then it wrap after

(2^32) seconds =

136.102208 years

- What if a router crashes? It would start its sequence over again at 0
  - Which would then be rejected by its neighboring routers

- If a sequence number gets corrupted and 65,540 gets sent instead of a 4 (just a 1-bit error), then packets 5-65,540 will be rejected

Solution: age

- A router will discard information after a certain amount of time has passed
- The age field also is decremented each time a router forwards the packet
  - (Similar to IP TTL)
  - If a packet gets lost, then eventually its age will reach 0 and no longer be forwarded
  - Prevents packets from looping around forever

- Acknowledgements are sent for LS packets as well to avoid errors

Computing New Routes

- Once the LS packets have arrived, the router has all the information needed to compute routes

- In the world
  - ISPs generally use a link state routing algorithm called IS-IS or OSPF

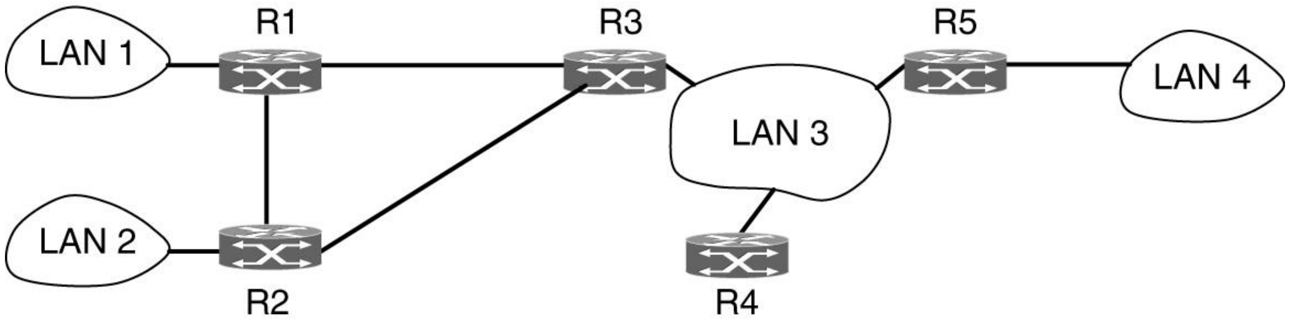
- General problems
  - A router that says it has a route but doesn't actually
  - Forgets it has a route that still exists
  - Fails to forward or corrupts LS packets
  - Many outages lately have been router misconfigurations

Major Router Outages

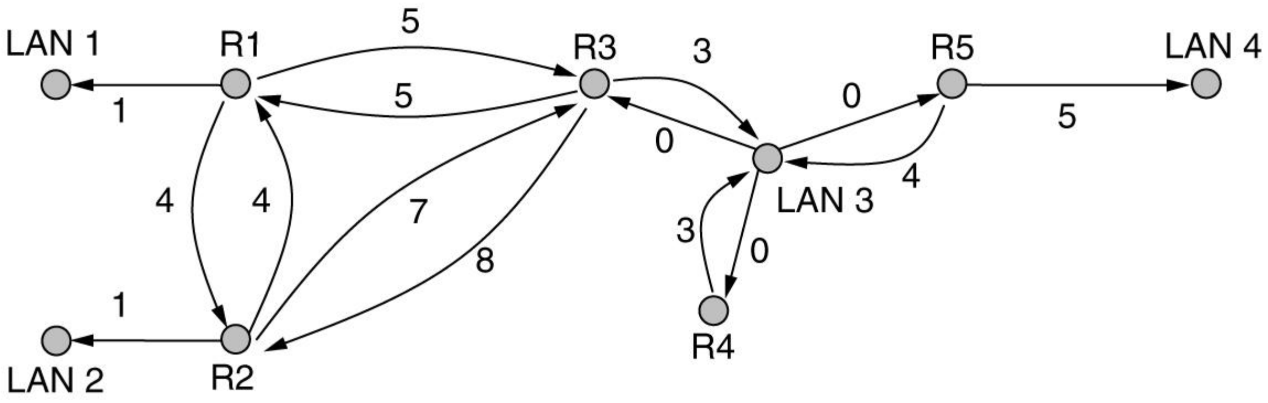
- Cloudflare, Jun 21 2022
  - https://blog.cloudflare.com/cloudflare-outage-on-june-21-2022/
- Facebook, Oct 4 2021
  - https://engineering.fb.com/2021/10/04/networking-traffic/outage/
  - https://blog.cloudflare.com/october-2021-facebook-outage/
  - https://blog.cloudflare.com/during-the-facebook-outage/
- That time Verizon took out the Internet, Jun 24 2019
  - https://blog.cloudflare.com/how-verizon-and-a-bgp-optimizer-knocked-large-parts-of-the-internet-offline-today/

OSPF – Interior Gateway Routing Protocol

- The university as an autonomous system (AS)
  - OSPF likely used as the internal gateway routing protocol
  - IS-IS is usually used by ISPs



(a)

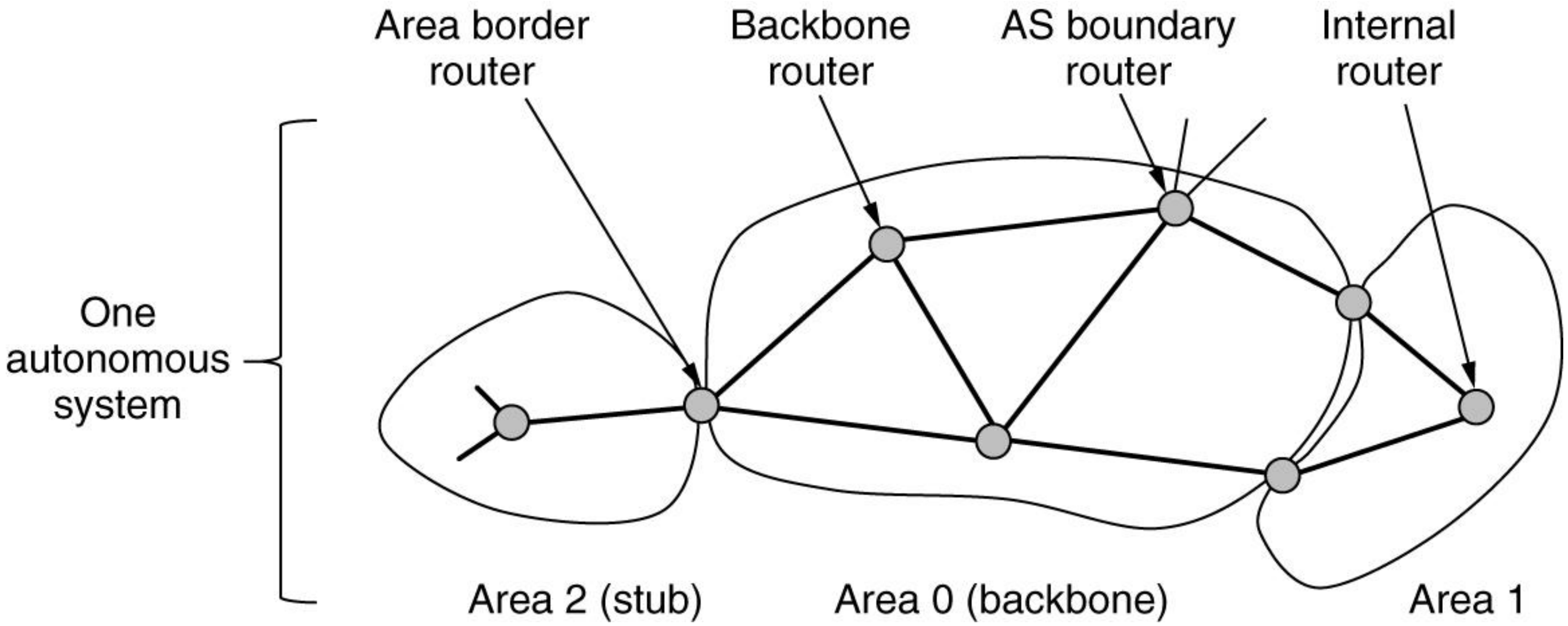


(b)

- (a) An autonomous system. (b) A graph representation of (a).

Area 0: the backbone

- Every AS has an Area 0, the routers in this area are called backbone routers
- The idea is to get from one area (on campus) to any other area on campus
  - Traffic from Sierra Hall to Ojai Hall travels through some AS0
- Routers between two areas are called area border routers



OSPF

- OSPF works by exchanging link state messages between *adjacent* routers
  - (not necessarily neighboring routers)
  - It is not efficient to have every router talk to every other router
- One router in an area is the designated router
  - Adjacent to all routers in the area
  - With another router configured as a duplicate for backup
- Periodically LINK STATE UPDATE message are sent to all the adjacent routers

Message type	Description
Hello	Used to discover who the neighbors are
Link state update	Provides the sender's costs to its neighbors
Link state ack	Acknowledges link state update
Database description	Announces which updates the sender has
Link state request	Requests information from the partner

Traceroute (from lab on campus to google) showing various regions

```
$ traceroute -a google.com
traceroute to google.com (216.58.193.206), 64 hops max
 1  [AS0] 10.31.0.1 (10.31.0.1)
 2  [AS0] internal-gp.csuci.edu (10.121.0.50)
 3  [AS2152] 209.129.116.196 (209.129.116.196)
 4  [AS2152] 137.164.41.217 (137.164.41.217)
 5  [AS2152] 137.164.11.36 (137.164.11.36)
 6  [AS15169] 74.125.49.165 (74.125.49.165)
 7  [AS15169] 108.170.247.161 (108.170.247.161)
 8  [AS15169] 209.85.143.159 (209.85.143.159)
 9  [AS15169] lax02s23-in-f14.1e100.net (216.58.193.206)
```