SC/CSE 3213 Winter 2014

L4: Internet Routing and Controls

Sebastian Magierowski
York University

Outline

- Routing
- Error Detection
- Retransmission
- Congestion Control
- Flow Control
- Medium Access Control

Forwarding Tables

- How do routers know what to do with their packets?
- Their forwarding tables tell them:
- Forwarding: The process of taking a packet from an input and sending it out the appropriate output
- Forwarding tables need to contain every detail of a link

Destination	Interface	MAC Address
128.208.128.0/17	if0	8a:0c:1f:e4:6b:1c
128.208.0.0/18	if0	8a:0c:bb:e4:3b:a1
128.208.96.0/19	if2	8a:0c:7b:a9:b2:fc

- know the IP of where you want to send
- the router port from which you want to send
- the physical address of the destination (if applicable, not all protocols require this)

- They are often implemented in VLSI hardware
  - high-speed memories

Routing Tables

- Where do forwarding tables come from?
- From routing tables:
- Routing: The process of building the tables that determine the correct destinations for packets

Destination	Next Hop
128.208.128.0/17	171.69.245.10
128.208.0.0/18	171.69.245.10
128.208.96.0/19	178.45.23.124

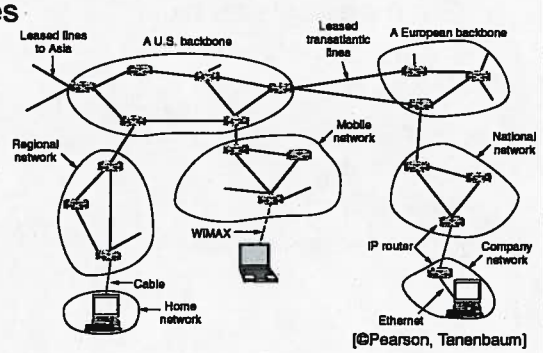
- Simpler than forwarding tables
  - typically just a data structure in a computer

but logically contain the information we ~~want to~~ need to move packets through network to destinations

Routing

how do you organize routing

- How do you build routing tables? How do you route?
- Routers run algorithms that update their knowledge of the network every few seconds to hours
- They do this by sending out queries for information and by responding to queries
- Routing seeks to find the cheapest path from any source to any destination
- Minimize link costs



[©Pearson, Tanenbaum]

first a reminder

Hierarchical Routing

- Routing over two-levels substantially cuts on complexity
  - Within AS
  - And between them
    - Region-region comms condensed to single router
    - Increased path length a common penalty (e.g. 1A to 5C)

sacrifice some efficiency

non-hierarchical

1A: Full Table

Dest.	Line	Hops
1A	–	–
1B	1B	1
1C	1C	1
2A	1B	2
2B	1B	3
2C	1B	3
2D	1B	4
3A	1C	3
3B	1C	2
4A	1C	3
4B	1C	4
4C	1C	4
5A	1C	4
5B	1C	5
5C	1B	5
5D	1C	6
5E	1C	5

hops to individuals

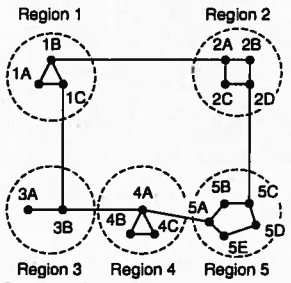
hierarchical

much smaller

1A: Hierarchical Table

Dest.	Line	Hops
1A	–	–
1B	1B	1
1C	1C	1
2	1B	2
3	1C	2
4	1C	3
5	1C	4

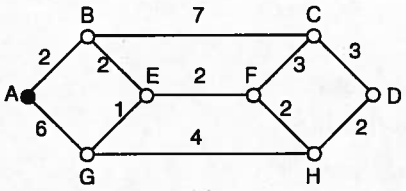
hops to group



[©Pearson, Tanenbaum]

Link Metrics (Costs)

- Routing often seeks to identify the shortest path between destinations in a graph, the smallest link metric
- Easiest is to just treat all links the same and just count hops (RIP: Routing Information Protocol, does this) ← Bellman-Ford, like BGP - distance-vect approach
- But many options exist
  - mean delay (latency)
  - distance
  - bandwidth
  - average traffic
  - communication cost
  - political/economic policy



[©Pearson, Tanenbaum]

Routing Types

- Methods of building routing tables?
- They do this by sending out queries for information and by responding to queries
  - Intradomain Routing
    - Routing within an AS ← Intermediate System
    - OSFP (Campus), IS-IS (ISPs)
  - Interdomain Routing
    - Routing between AS
    - BGP

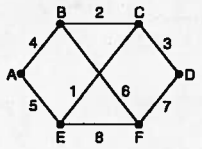
link state routing

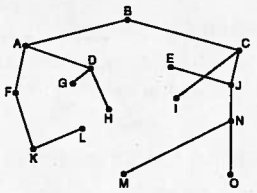
distance vector routing

OPEN shortest path first

Intradomain Routing

- OSPF: Open Shortest Path First
  - Link-state routing
  - Every router builds up a knowledge of the whole network topology
    - 1) Find link quality to all next-hop neighbors (local view formed)
    - 2) Send this information throughout the whole network (global view formed)
    - 3) Compute shortest path to every router (Dijkstra's algorithm)
- Details…





[©Pearson, Tanenbaum]

Intradomain Routing

getting to know who you are connected to

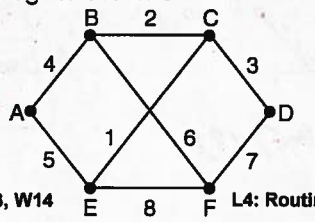
- Learning about neighbours
  - When router is introduced it sends out HELLO packet on each line
  - Router on other line sends back its name

point-to-point

find numbers

- Setting link costs
  - Connecting routers can construct costs by sending their bandwidth limits
  - Delay can also be constructed by sending ECHO packets
- Building link state packets
  - Aggregate the info

build packet containing all data learned



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

Link State Packets

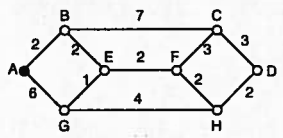
Intradomain Routing

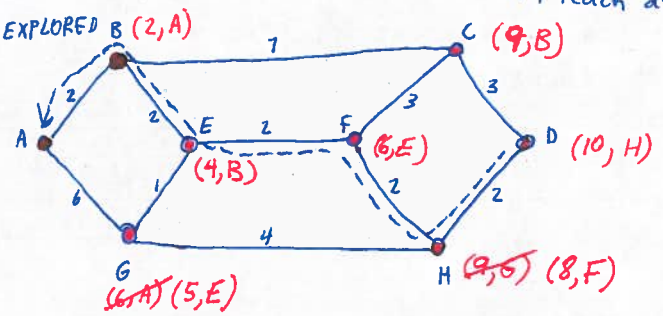
- **Distributing link state packets**
  - Trickiest part
    - All routers must get packets quickly and reliably
    - If routers have different versions of topology odd behaviour will result
  - Use flooding to distribute link state packets
  - Every incoming packet is sent out on every outgoing line
    - Except the one it arrived on
    - How do you keep from swamping the network?
  - Packets have sequence numbers
    - Each time you (the source) send out a new packet increment its sequence number, k
    - Routers keep track of (*source router*, $k_{largest}$) pairs
    - If incoming packet has $k < k_{largest}$ discard it
    - 32-bit k's would take 137 to loop at 1 packet per second
    - Age field is decremented in case router goes down

CSE 3213, W14 L4: Routing & Control 11

Shortest-Path Finding

- **Now apply Dijkstra's algorithm to find shortest path**





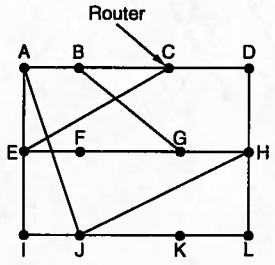
CSE 3213, W14 L4: Routing & Control 12

Interdomain Routing

- BGP: Border Gateway Protocol
  - Path vector routing (a form of distance vector routing)
  - Exchanging data with neighbours to incrementally form a global view of the network

table "vector" holding best known distance to each destination

i.e. each router has this a global view



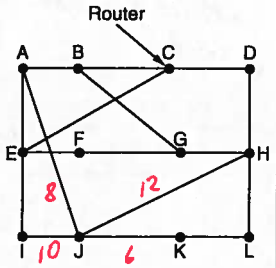
[©Pearson, Tanenbaum]

Vector Tables

- Each router maintains a vector table (e.g. J)
  - For each router (destination) in the network keeps track of...
    - The next hop it should take
    - The total (estimated) distance to the destination

Destination	Next Hop	Total Distance
A	A	8
B	~~B~~ A	15
C	H	12
D	A	17
E	I	22

- It can start a basic network table by talking to its neighbours
  - JA=8, JI=10, JH=12, JK=6



Vector Table Updates

- Every period T each router shares its vector table with each of its neighbours, X, and their distance to destination Y
  - Looks at their distance, estimates XY
    - Neighbour X's estimate to destination Y
    - If my estimate to Y, RY > XY + RX...
    - ...update my row for Y to
      - next hop: X
      - distance: XY + RX
- For example what happens to J's vector table with...
  - JA=8, JI=10, JH=12, JK=6
- This technique is called the distributed Bellman-Ford algorithm

To	A	I	H	K
A	0	24	20	21
B	12	36	31	28
C	25	18	19	36
D	40	27	8	24
E	14	7	30	22
F	23	20	19	40
G	18	31	6	31
H	17	20	0	19
I	21	0	14	22
J	9	11	7	10
K	24	22	22	0
L	29	33	9	9
J to...	8	10	12	16

Distance Vector Convergence

- Good news travels fast
  - Delay metric is number of hops
  - A is down initially
  - But when it comes up...
  - ...each exchange propagates the news in a linear fashion

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

- Bad news travels slow
  - A suddenly goes down
  - B does not hear from A...
  - ...but C thinks it is 2 hops away
    - B thinks it can get to A from C
  - But B & D think they are 3 away
    - So C updates to 4, etc., etc.
  - Distance = 1 + min(neighbour)
    - Slow count to infinity

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

Routing Summary

- Intradomain
  - OSPF
  - link-state
    - global communication
    - local computation
  - Runs in network layer
    - acknowledged IP
  - Tends to have high memory requirements
    - Keeping track of each router's link state
  - More computation in implementing graph search
- Interdomain
  - BGP
  - distance vector
    - local communication
    - global computation
  - Runs in application layer
    - utilizes TCP
  - Slow at pruning out bad links
    - Bad news travels slowly (count-to-infinity problem)

Error Detection

- Physical links, router and host hardware can corrupt messages
- Routers should have the ability to detect errors
- Many approaches are used at many levels
  - Line coding at the physical layer (will discuss layers in L4)
    - Turbo codes
    - Reed-Solomon codes
    - LDPC codes
- At the data link and network layers
  - Use header checksums
  - A parity scheme

Retransmission of Erroneous Information

- What do you do when you detect an error?
- You can just forget about the erroneous packet
  - System doesn't breakdown, but information is lost
- Or you can request that the same packet be retransmitted
  - Foundation of a reliable delivery service
  - Such a scheme is called: Automatic Repeat reQuest (ARQ)
    - Units send messages and expect acknowledgments
    - A number of strategies are employed to make this approach reliable and efficient
  - Present in both the data link and transport layers
  - In the Internet this is typically carried out by the hosts not the routers

Congestion Control

- What happens if many hosts send packets through one link?
- Router buffer is overwhelmed and must start discarding packets
- Hosts don't get acknowledgments and thus slow down the rate at which they send information

(so you are typically controlling this with the end points)

Flow Control

- A fast transmitter can overwhelm a slow receiver
- In this case receiver indicates (in acknowledgments) to the transmitter how much buffer space it has remaining
- Transmitter doesn't send unless it is aware of enough buffer space in receiver
  - Implemented within ARQ

automatic repeat request

→ here there is explicit comms between hosts about state of exchange

→ in congestion control we react to circumstances (i.e. no ack)

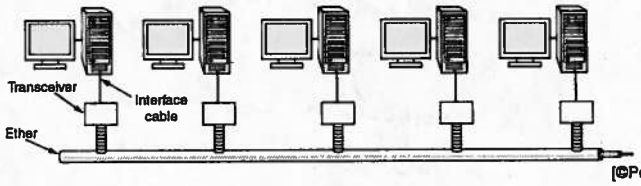
} at least this is a basic thing

Congestion & Flow Control Summary

- Congestion Control
  - Prevents overflowing the router buffers
  - Concerned with network internals
- Flow Control
  - Prevents overflowing the destination buffer
  - Concerned with end-to-end operation

Medium Access Control (MAC)

- Multiple hosts try to communicate over one medium
  - one wire
  - one radio channel
- How do the units organize their behaviour in order to achieve useful communication?
  - This is the job of the MAC
- This is more the job of LAN and less a specific Internet function



[©Pearson, Tanenbaum]