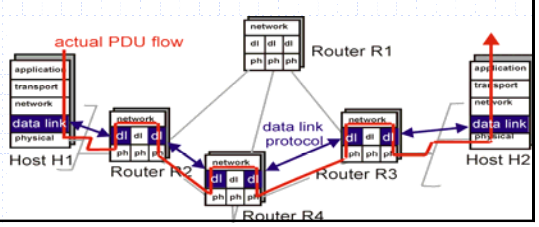
ECE453 – Introduction to Computer Networks

Lecture 9 - The Network Layer (I) - Routing

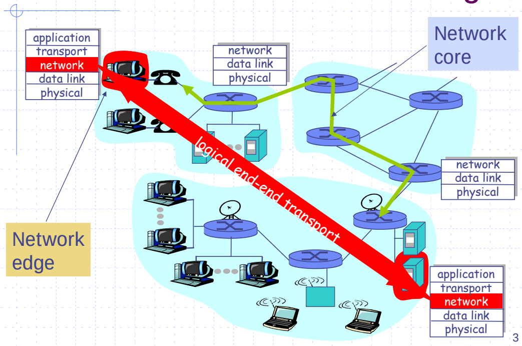
1

Recap

- Framing
- Error detection and correction
- Reliable or unreliable data transfer
- Channel allocation protocol



Network Core and Network Edge



3

Network Core – Information Transmission

- Circuit switching
  - Telephone system
- Message switching
  - Mail delivery
  - The message travels as a complete unit. At any one time, it completely exists in one place.
- Packet switching
  - The Internet

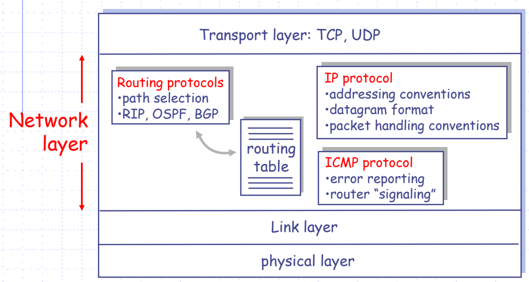
Design Issues

- Store-and-forward packet switching
- Services to the transport layer
  - Connection-oriented vs. Connectionless
  - Quality of service (QoS)

Network Layer Services

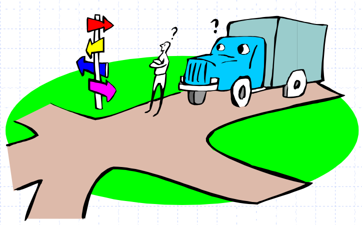
- Connectionless
  - Best-effort
  - No guarantee
  - The Internet
  - No advance setup is needed
  - Datagram subnet
- Connection-oriented
  - Guaranteed delivery
  - ATM
  - A path from the source router to the destination router is established before any data packets can be sent
  - Virtual circuit

The Internet Network Layer



7

Routing - In the Middle of Intersection …



8

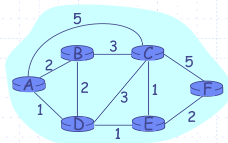
Different Strategies

- Nonadaptive algorithms (or static routing)
- Adaptive algorithms (or dynamic routing)
  - Global algorithm (have a global knowledge – a map)
  - Decentralized algorithm (get information only from neighbor)

9

Graph Abstraction for Routing Algorithms

- Graph nodes ←→ Routers
- Graph edge ←→ Physical links
- Edge weight ←→ Link cost
- Link cost
  - Delay, power consumption, congestion level, $cost, etc.
- Good path or optimal path
  - Minimum link cost



Two Fundamental Routing Algorithms

- Link state routing
  - A global algorithm
- Distance vector routing (or Bellman-Ford routing, Ford-Fulkerson routing)
  - A decentralized algorithm
  - The original ARPANET routing algorithm, replaced by LS routing in 1979

11

A Link State Routing Algorithm

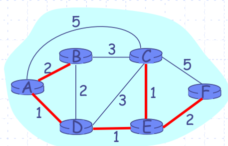
Dijkstra's algorithm

- Net topology, link costs known to all nodes
  - accomplished via "link state broadcast"
  - all nodes have the same info
- computes least cost paths from the source to all other nodes
- Iterative algorithm

12

Dijkstra's Algorithm: An Example

Step	start N	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
0	A	2,A	5,A	1,A	infinity	infinity
1	AD	2,A	4,D		2,D	infinity
2	ADE	2,A	3,E			4,E
3	ADEB		3,E			4,E
4	ADEBC					4,E
5	ADEBCF					



13

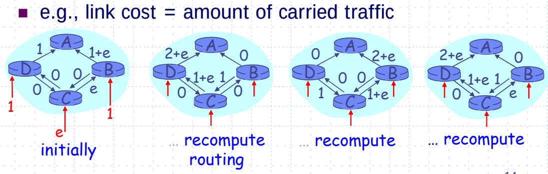
Dijkstra's Algorithm: Discussion

Algorithm complexity: n nodes

- each iteration: need to check all nodes, not in N
- n*(n+1)/2 comparisons: O(n**2)
- more efficient implementations possible: O(nlogn)

Oscillations possible:

- e.g., link cost = amount of carried traffic



14

DV Routing

- Each router maintains a distance table
- Initialization
  - 0 for itself
  - Infinity for non-neighbor
  - Link cost for neighbor
- Message exchange between neighbors
  - When a neighbor first comes up
  - When information changes (e.g. change in link cost)
- Distance vector calculation
  - Minimize cost to each destination

cost to destination via

$D^E()$	A	B	D
A	1	14	5
B	7	8	5
C	6	9	4
D	4	11	2

destination

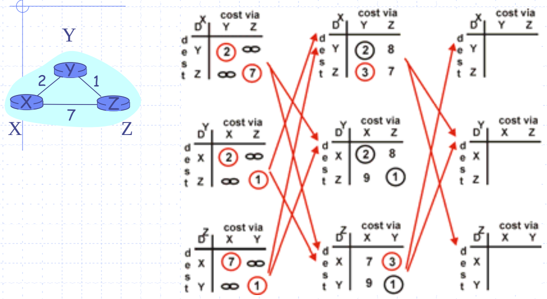
$D^X(Y,Z)$ = distance *from* X *to* Y, *via* Z as next hop

= $c(X,Z) + \min_w\{D^Z(Y,w)\}$

Iterative routing
Distributed routing

15

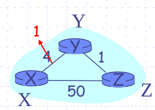
DV: Example

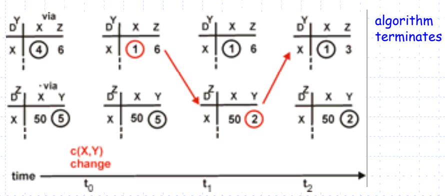


16

DV: Link Cost Changes

"good news travels fast"



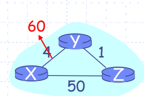


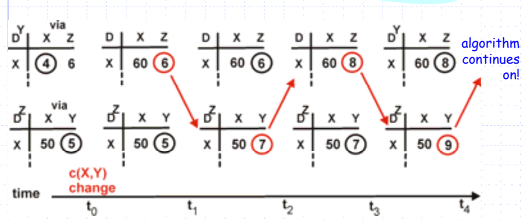
17

DV: Link Cost Changes

"bad news travels slow"

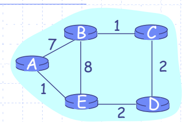
Count to infinity problem





18

Distance Table: An Example



$$D^E(C,D) = c(E,D) + \min_w \{D^D(C,w)\} = 2+2 = 4$$

$$D^E(A,D) = c(E,D) + \min_w \{D^D(A,w)\} = 2+3 = 5$$ loop!

$$D^E(A,B) = c(E,B) + \min_w \{D^B(A,w)\} = 8+6 = 14$$ loop!

cost to destination via

destination: $D^E()$	A	B	D
A	1	14	5
B	7	8	5
C	6	9	4
D	4	11	2

19