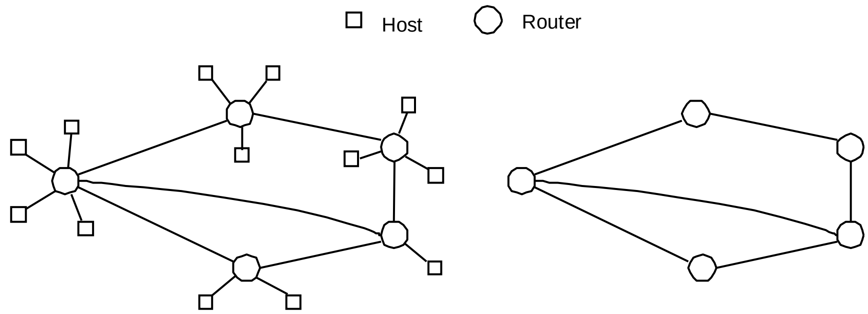
Network Layer, Part 2
Routing

Terminology

- User machines, (e.g., your PCs) are usually called
  - **hosts** by the internet community or
  - **end systems** in the OSI reference model
- User machines access the (switching-based) network by connecting to switching devices, called
  - **routers** in the Internet,
  - **switches** in ATM networks, or
  - **intermediate systems** in the OSI reference model

❑ The interconnection of network nodes, that is, the topology of the network, can be described by a graph:



(a) Network topology with hosts

(b) After omitting hosts

❑ Since only routers are involved in routing activities, we omit hosts in this talk.

The Routing Problem

❑ Given a pair of source *s* and destination *d*, find a path *P* to reach *d* from *s*.

❑ Typically, *P* is a shortest *s*-to-*d* path.

❑ What constitutes "shortest" depends on the definition of distance:

– One can count the number of "hops" to reach *d* from *s*.

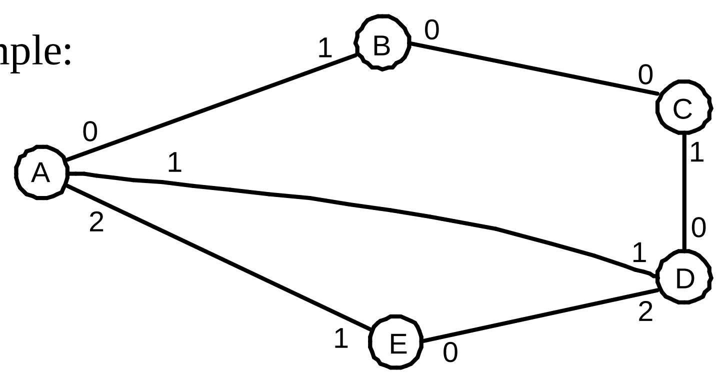
– Or we can use more sophisticated metrics, such as delays, monetary costs, reciprocal of bandwidth, and so forth

Routing Method I:

Distance Vector Routing

- Each router maintains a vector of distances to others. Initially, each router knows only the existence of itself (it does not know the rest of the network.)
- An entry (*Y*,*p*,*c*) in the vector of router *X* means that *X* can reach *Y* via port *p* with cost *c*.

Example:



Initial distance vectors at the five routers:

A	B	C	D	E
A -1 0	B -1 0	C -1 0	D -1 0	E -1 0

- Every router periodically sends its distance vector to its neighboring routers.
  - Assuming that router A sends its vector first, we have the following distance vectors:

A	B	C	D	E
A -1 0	A 1 1 B -1 0	C -1 0	A 1 1 D -1 0	A 1 1 E -1 0

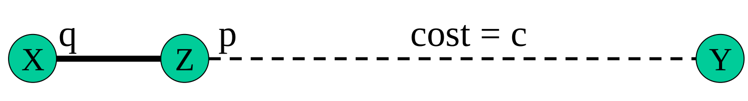
  - Next *C* sends its vector, we have

A	B	C	D	E
A -1 0	A 1 1 B -1 0 C 0 1	C -1 0	A 1 1 C 0 1 D -1 0	A 1 1 E -1 0

Processing Incoming Vectors

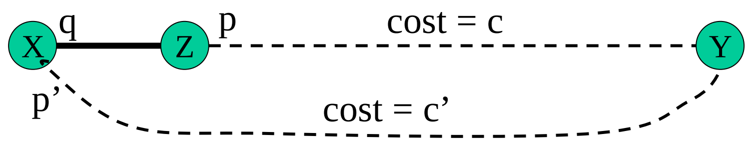
- Assume that a router *X* receives via port *q* the vector of its neighbor *Z.*
- *X* will process each entry in the vector one by one.
- Let (*Y*,*p*,*c*) be an entry in Z's vector.

Case 1: *Y* is not found in *X*'s vector



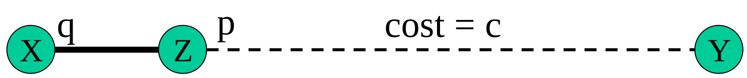
  - *X* creates an entry (*Y*,*q*,*c*+1) in its vector

Case 2: An entry (*Y*,*p'*,*c'*), where *p'* is not *q*, already exists in X



– If c'>c+1, then X replaces (*Y*,*p'*,*c'*) with (*Y*,*q*,*c*+1); otherwise, do nothing.

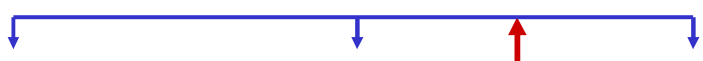
Case 3: an entry (*Y*,*q*,*c'*) is already in X's



– Replace (*Y*,*q*,*c'*) with (*Y*,*q*,*c*+1)

Continuing the Example

- Next, *D* sends its vector.



A	B	C	D	E
A -1 0	A 1 1 B -1 0 C 0 1	C -1 0	A 1 1 C 0 1 D -1 0	A 1 1 E -1 0

- We have

A	B	C	D	E
A -1 0 C 1 2 D 1 1	A 1 1 B -1 0 C 0 1	A 1 2 C -1 0 D 1 1	A 1 1 C 0 1 D -1 0	A 1 1 C 0 2 D 0 1 E -1 0

• After *E* sends its vector (to *A* and *D*), we have

A	B	C	D	E
A -1 0	A 1 1	A 1 2	A 1 1	A 1 1
C 1 2	B -1 0	C -1 0	C 0 1	C 0 2
D 1 1	C 0 1	D 1 1	D -1 0	D 0 1
E 2 1			E 2 1	E -1 0

• Next, *B* sends vector (to *A* and *C*), we have

A	B	C	D	E
A -1 0	A 1 1	A 1 2	A 1 1	A 1 1
B 0 1	B -1 0	B 0 1	C 0 1	C 0 2
C 1 2	C 0 1	C -1 0	D -1 0	D 0 1
D 1 1		D 1 1	E 2 1	E -1 0
E 2 1				

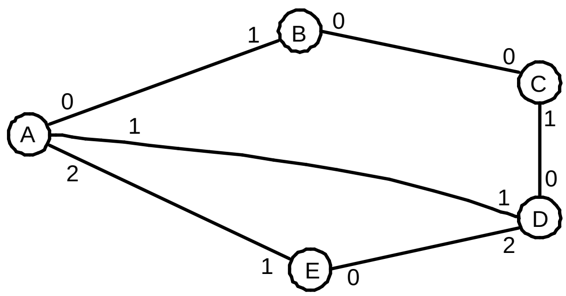
• Assuming that *A* sends its vector again, we have

A	B	C	D	E
A -1 0	A 1 1	A 1 2	A 1 1	A 1 1
B 0 1	B -1 0	B 0 1	B 1 2	B 1 2
C 1 2	C 0 1	C -1 0	C 0 1	C 0 2
D 1 1	D 1 2	D 1 1	D -1 0	D 0 1
E 2 1	E 1 2		E 2 1	E -1 0

• Routing tables/vectors are completed after *D* sends its vector again; see next page.

Complete Routing Tables

A			B			C			D			E		
A	-1	0	A	1	1	A	1	2	A	1	1	A	1	1
B	0	1	B	-1	0	B	0	1	B	1	2	B	1	2
C	1	2	C	0	1	C	-1	0	C	0	1	C	0	2
D	1	1	D	1	2	D	1	1	D	-1	0	D	0	1
E	2	1	E	1	2	E	1	2	E	2	1	E	-1	0



Exercise

- The distance vector of Router X is:

(A,3,5) (B,0,2) (C,0,1) (W,2,4) (X,-1,0) (Y, 1,1)

- X receives via port 0 a vector

(A,3,2) (B,0,4) (C,-1,0) (W,1,4), (X,4,1) (Z,2,2)

- Give the vector of X after processing the arriving vector.

Handling Link Failures

- When router X notices the failure of port *p*, it changes in its vector all the entries (*Y*,*p*,*c*) to (*Y*,*p*,∞).
- Assuming that ***A-D*** link has failed, we have

A			B			C			D			E		
A	-1	0	A	1	1	A	1	2	A	1	∞	A	1	1
B	0	1	B	-1	0	B	0	1	B	1	∞	B	1	2
C	1	∞	C	0	1	C	-1	0	C	0	1	C	0	2
D	1	∞	D	1	2	D	1	1	D	-1	0	D	0	1
E	2	1	E	1	2	E	1	2	E	2	1	E	-1	0

- *A* will learn a new path to get to *C* after *B* sends its vector

Discussion

- Distance vector routing can be considered as a distributed implementation of a shortest-path computation algorithm, the **Bellman-Ford Algorithm**.
- Examples of distance vector routing protocol include the **RIP** (Routing Information Protocol) and RIP II of the Internet.
  - they have been in use since the beginning the Internet.

Problems with Distance Vector Routing

- Slow convergence
  - after a change in network topology (failures of links, additions of new links/routers, etc.), it may take more than 10 minutes for routing tables to stabilize
- Transient routing loops

Transient Routing Loops

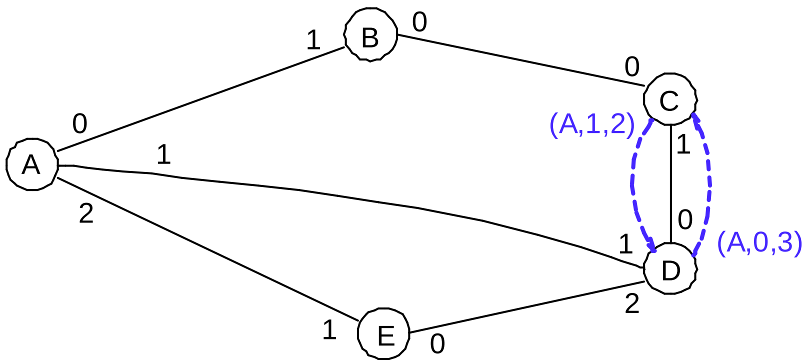
- Consider again the handling the failure of the *A*-*D* link
- From page 14, we have

A			B			C			D			E		
A	-1	0	A	1	1	**A**	**1**	**2**	A	1	**∞**	A	1	1
B	0	1	B	-1	0	B	0	1	B	1	∞	B	1	2
C	1	∞	C	0	1	C	-1	0	C	0	1	C	0	2
D	1	∞	D	1	2	D	1	1	D	-1	0	D	0	1
E	2	1	E	1	2	E	1	2	E	2	1	E	-1	0

- Assume that *C* sends its vector before others.
- Let us focus on the entries regarding *A* at *C* and *D*.

Example

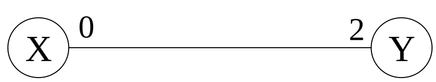
- Consider a packet from D to A.



- The packet gets trapped in a loop between D and C.

Exercise

- Consider two neighboring routers X and Y



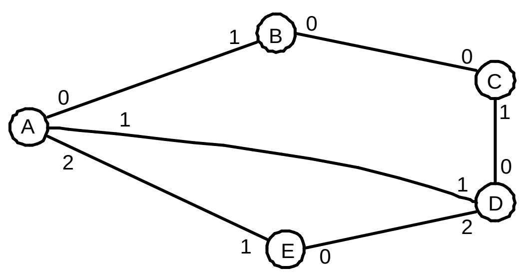
- Vector of
  - X: (A,3,2)(B,1,1)(C,0,4)(X,-1,0)(Y,0,1)
  - Y: (A,2,7)(B,0,3)(C,2,3)(X,2,1)(Y,-1,0)
- Identify a routing loop.

Routing Method II: Link State Routing

- Using a special-purpose broadcast algorithm, every router periodically broadcasts to all other routers a list of its local links.
  - such a broadcast message is termed a **link state advertisement** (LSA)
  - the broadcast method, discussed later, is called **flooding**

CS 656 21

Example:



- the LSA from *A* contains links *A*-*B*, *A*-*D*, and *A*-*E*
- the LSA from *B* contains links *A*-*B* and *B*-*C*
- the LSA from *C* contains links *B*-*C* and *C*-*D*
- the LSA from *D* contains links *C*-*D*, *A*-*D*, and *D*-*E*
- the LSA from *E* contains links *A*-*E* and *D*-*E*

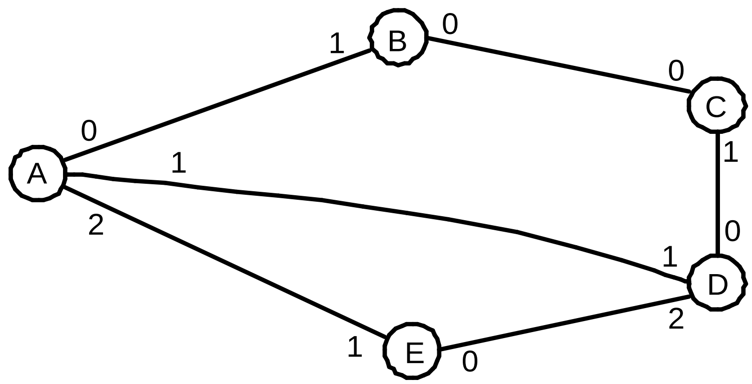
CS 656 22

- Through LSAs from other routers, every router maintains a complete map of the entire network.
- Using a shortest-path algorithm, each router *X* computes the shortest paths from *X* to all other routers.
  - For instance, the Dijkstra's open shortest path first algorithm with complexity $O(\boldsymbol{N}^2)$.
- When detecting failure/recovery/addition of its local links, a router immediately broadcasts a new LSA.
  - In response, all routers re-compute shortest paths

Flooding

- The source (router) of an LSA sends the LSA to all its neighboring nodes.
- When a node receives the LSA the first time, it updates its local network map accordingly and forwards the LSA via all interfaces except the one on which the LSA arrives.
- The node discards subsequent duplicates of the LSA.
- All LSA forwarding steps are reliable, using a stop-and-wait protocol.

Flooding in Operation



Discussion

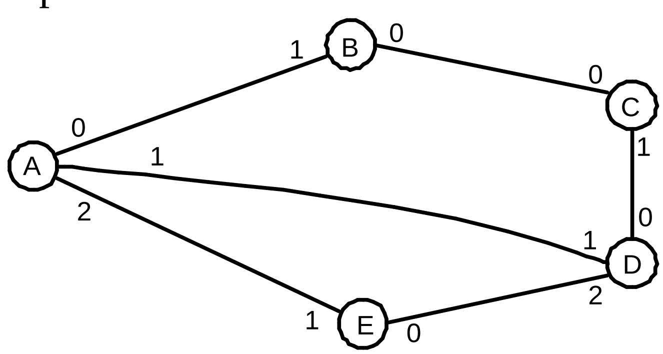
- Link state routing protocols:
  - **OSPF** (Open Shortest Path First) for the Internet
  - **IS-IS** (Intermediate System to Intermediate System) from ISO
  - **PNNI** (Private Network to Network Interface) for ATM networks
- Major Advantage
  - fast convergence after network changes

Disadvantages

- Limitations in scalability due to
  - the maintenance of the entire network map at every router
  - the periodic broadcast from every router
  - shortest path computations at every router
- Complexity
  - it takes 100 pages to describe OSPF; RIP needs only 20

Routing Method III: Path Vector Routing

- Each router maintains a vector of complete paths to all other routers.
- Example:



- Initial path vectors at the five routers:

A	B	C	D	E
A ( )	B ( )	C ( )	D ( )	E ( )

- Router *A* sends its vector first; we have

A	B	C	D	E
A ( )	A (A) B ( )	C ( )	A (A) D ( )	A (A) E ( )

- Next, router *C* sends its vector; we have

A	B	C	D	E
A ( )	A (A) B ( ) C (C)	C ( )	A (A) C (C) D ( )	A (A) E ( )

- Then, *D* sends its vector:

A	B	C	D	E
A ( ) C (D, C) D (D)	A (A) B ( ) C (C)	A (D, A) C ( ) D (D)	A (A) C (C) D ( )	A (A) C (D, C) D (D) E ( )

- After *E* sends its vector, we have

A	B	C	D	E
A ( ) C (D, C) D (D) E (E)	A (A) B ( ) C (C)	A (D, A) C ( ) D (D)	A (A) C (C) D ( ) E (E)	A (A) C (D, C) D (D) E ( )

- After *B* sends its vector, we have

A	B	C	D	E
A ( )	A (A)	A (D, A)	A (A)	A (A)
B (B)	B ( )	B (B)	C (C)	C (D, C)
C (D, C)	C (C)	C ( )	D ( )	D (D)
D (D)		D (D)	E (E)	E ( )
E (E)				

- Assuming that *A* sends its vector again,

A	B	C	D	E
A ( )	A (A)	A (D, A)	A (A)	A (A)
B (B)	B ( )	B (B)	B (A, B)	B (A, B)
C (D, C)	C (C)	C ( )	C (C)	C (D, C)
D (D)	D (A, D)	D (D)	D ( )	D (D)
E (E)	E (A, E)		E (E)	E ( )

Discussion

- Path vector routing is designed for use by the **BGP** (Border Gateway Protocol) of the Internet.
- One major advantage is its prevention of routing loops (by inspecting paths).
- Path vector routing also enables **policy routing**,
  - the selection of paths based on non-technical issues.

Policy Routing

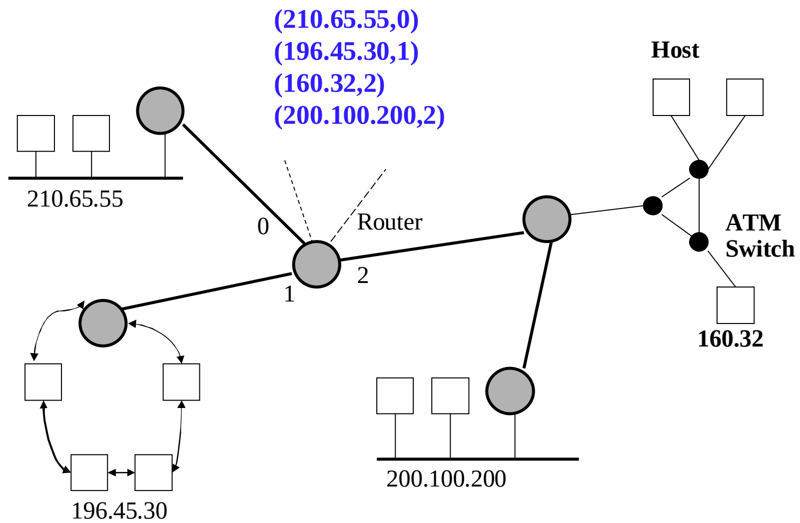
- A router belonging to ISP (Internet Service Provider) *A* may want to reject a path that uses routers belonging to its rival ISP *B*, even if the path is the shortest.
- A router of FBI must reject paths that involve routers controlled by a hostile country.
- Such filtering is achieved by scanning a path before accepting it.

IP Routing Tables

- IP routing protocols do not deal with destination addresses; they deal with destination network IDs.
  - in distance vector routing, each vector entry is of the form (netid, interface, cost).

- The routing table of a router comprises a list of network-ID to interface mappings.
- When a packet arrives at a router, the destination network ID is used to look up the routing table to determine via which interface to forward the packet.

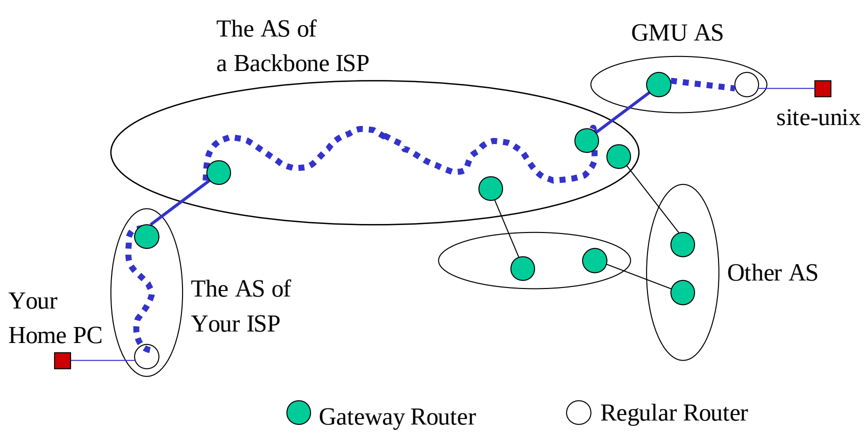
Routing Table Example



Internet Routing Hierarchy

- The Internet is divided into Autonomous Systems (**AS**).
- Each AS comprises the routers and networks governed by a single administration authority.
- Routing occurs at two levels
  - Intra-AS routing, also called internal gateway routing
  - Inter-AS routing, also called external gateway routing

Two-Level Routing of the Internet



- If the destination of a given packet is inside the source AS, then the packet is routed by intra-AS routing protocols
  - for example, RIP and OSPF
- If the destination is outside the source AS, then the packet is routed through a sequence of intermediate ASes to reach the destination AS, using inter-AS routing protocols
  - such as BGP
  - Inside each AS, intra-AS routing is used