Network Technologies (TCP/IP Suite)

Umar Kalim
Dept. of Communication Systems Engineering

umar.kalim@niit.edu.pk
http://www.niit.edu.pk/~umarkalim

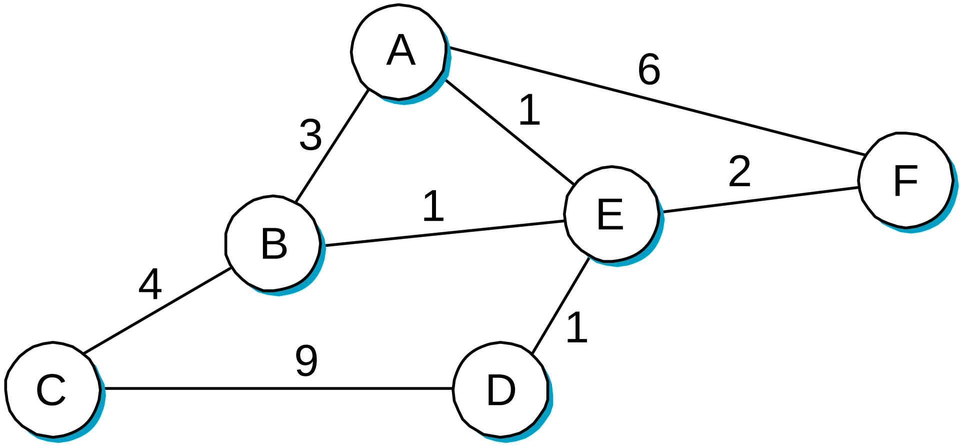
06/04/2007

Outline

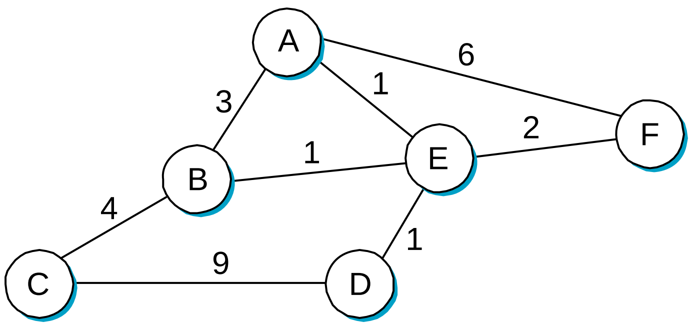
- Distance Vector Routing Protocol
  - RIP
- Link State Routing Algorithm
  - OSPF

Network Structure

- Administrative domains
  - Intra-domain routing protocols ~ Interior gateway protocols
  - Inter-domain routing protocols
- Network as a Graph



Network as a Graph

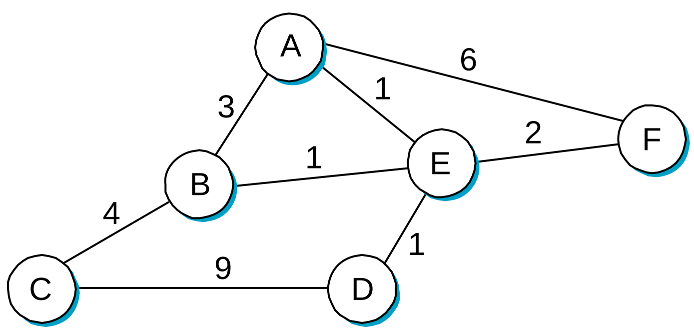


Routing is a problem of graph theory. You are given a set of nodes *V* and a set of edges *E* such that $e = (v_i, v_j)$ in *E* for v_i and v_j in V. These two sets define a graph $G=(V,E)$. You are also given a weight function *w*(.), which maps each *e* in *E* to some real valued weight.

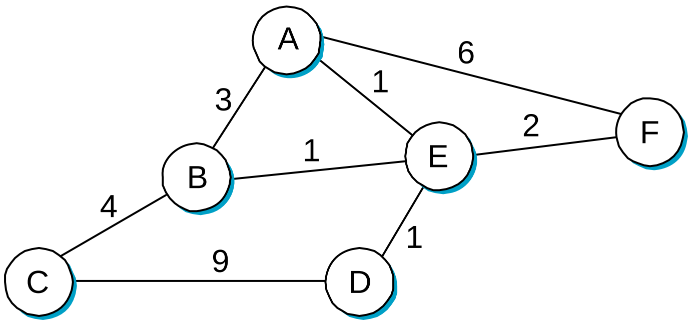
The problem is to find the lowest-cost path between any two nodes, where the cost of a path equals the sum of the weights of all edges in the path. This can be re-stated as the ***all-pairs shortest path problem***.

Network Structure

- Network as a Graph
  - Each edge has a cost ~ indication of the desirability of the link
- Basic Routing Problem ~ determine the lowest-cost path
  - Cost = sum of costs of all links forming the path



Proposal



Each node discovers its neighbors and tells one specific node (the *leader*) what they find. The leader runs an algorithm to solve the all-pairs shortest path problem, computing routing tables for each node. The leader communicates the routing table to each node, which stores it in non-volatile memory in each of the nodes.

Question: What are the problems with this method?

Limitations of static routing

- Routing can be done statically at the time of deployment, yet there are drawbacks
  - It does not deal with node or link failures
  - It does not consider the addition of new nodes or links
  - It implies that edge costs do not change
- Thus we require a *dynamic* and *distributed* routing algorithm
  - Why distributed?

Centralized solutions do not scale

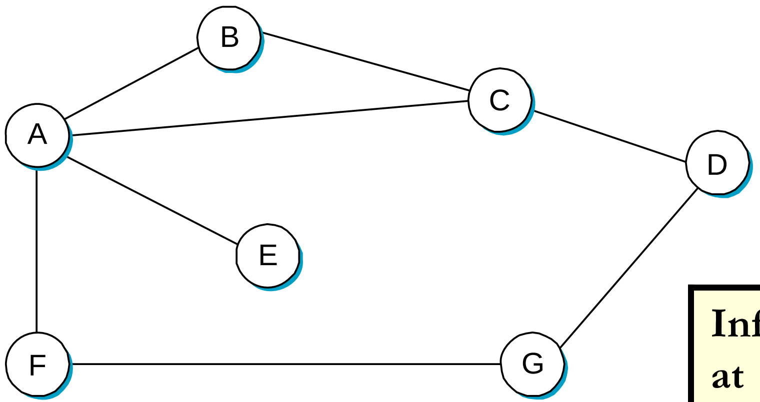
Distributed Approach Wins

- GGP - Predecessor of RIP
- No distinction between hosts and routers
- Attempts to keep track of the load in the network

Routing Algorithms - the Design Goals

- Optimally
- Simplicity/Low overhead
- Robustness/ Stability
- Rapid Convergence
- Flexibility

Distance Vector Routing



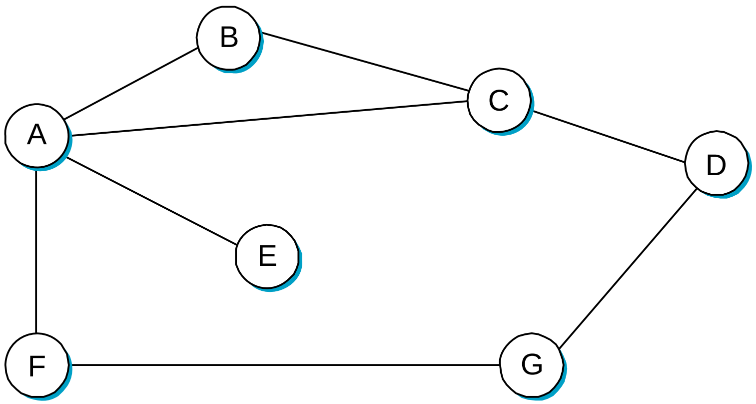
A's view: vector

Dest	Cost	NextHop
B	1	B
C	1	C
D	∞	-
E	1	E
F	1	F
G	∞	-

Global view

Info at node	Distance to node						
	A	**B**	**C**	**D**	**E**	**F**	**G**
A	0	1	1	∞	1	1	∞
B	1	0	1	∞	∞	∞	∞
C	1	1	0	1	∞	∞	∞
D	∞	∞	1	0	∞	∞	1
E	1	∞	∞	∞	0	∞	∞
F	1	∞	∞	∞	∞	0	1
G	∞	∞	∞	1	∞	1	0

Distance Vector Routing



Every node starts by building it's own local view of what nodes are 1 hop away.

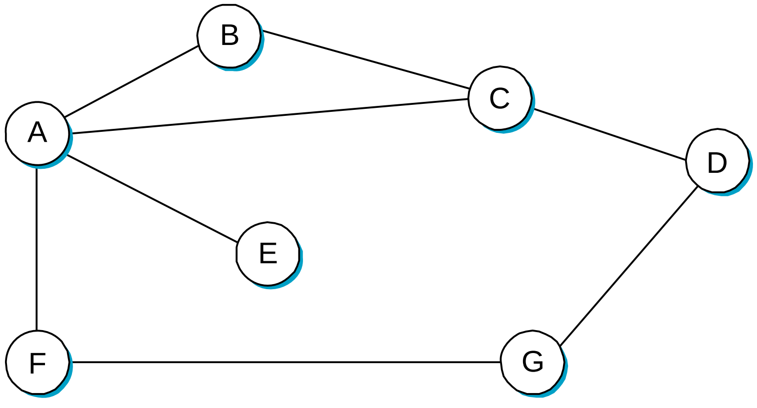
Next, every node sends its vector to its directly connected neighbors.

F tells A that it can reach G at cost 1. A knows it can reach F at cost 1, so it updated its own vector to indicate that it can reach G at cost 2. If A were to discover another route to G at a cost higher than 2, it would ignore it and leave its vector as it is.

After a few iterations of these exchanges, the routing table ***converges*** to a consistent state.

Question: How would this method deal with link failures?

Distance Vector Routing

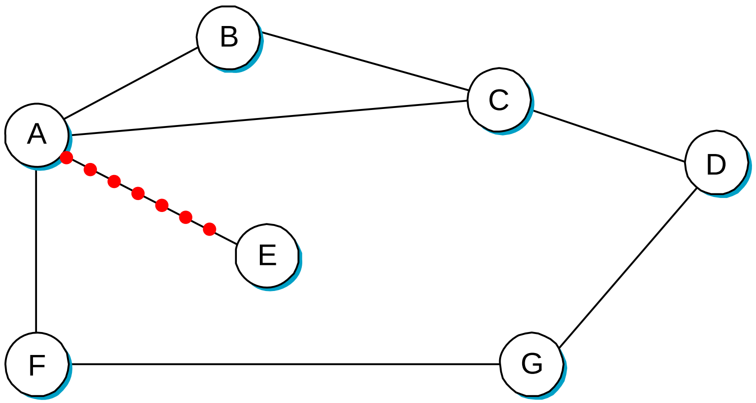


Periodic updates: Every *t* seconds, send your local info to your neighbors. This allows other nodes to know that you are running.

Triggered updates: Every time you learn new information from a neighbor that leads you to update your local vector, you send the recomputed vector to all your neighbors.

Question: How can you detect that a node has failed?

The Count-to-infinity Problem



The routing table doesn't *converge* or *stabilize*.

Link (A,E) goes down. A periodic update kicks in and A advertises that its distance to E is infinity. At about the same time, B and C tell each other that they can reach E in 2 hops. B knows now that it can't communicate with E via A, but concludes that it can send traffic to C which says it can reach A in 2 hops. Now, B is thinking that its distance to E is 3 hops and it lets A know about that. Argh, now A is going to think it can reach E in 4 hops and it will tell C of this "fact"… Where does this end?

“Solutions” for 2-node loops

Count-to-infinity: Cap infinity at some maximum number of hops that allow a packet to go all the way across the network.

Reduce time to convergence: Split-horizon – when a node sends a routing update to its neighbors, it does not send the routes it learned from a neighbor back to that neighbor.

Reduce time to convergence: Split-horizon with poison reverse – communicate back to the sending neighbor but “poison” the route with negative information (infinity) so that it doesn’t end up used as intermediate node in a route.

RIP - Routing Information Protocol

- first used in XNS (Xerox Network Systems)
- designed as a component of the networking code for the BSD release of UNIX
  - incorporated in program “routed” (rote management daemon)
- documented in rfc 1058

RIP - Characteristics

- the metric is a hop-count
  - The value of 1 to 15 is used (16 denotes infinity)
- supports point-to-point links and broadcast networks
  - doesn't support CIDR
- used only in IP networks
  - at first the intention was to be used in variety of networks

RIP - Characteristics

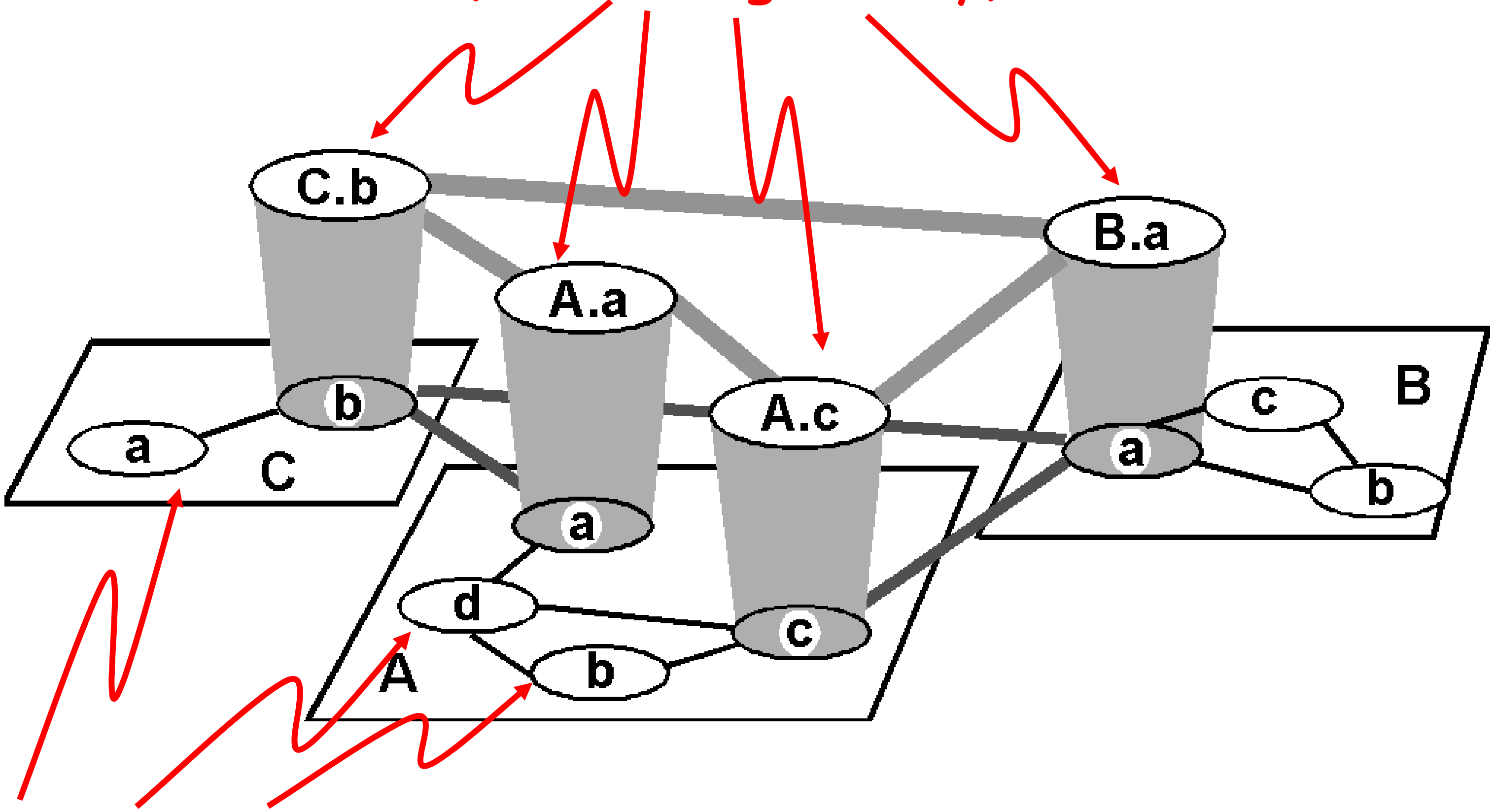
- packets are sent every 30 seconds or faster when necessary
- route is considered down if not refreshed within 180 sec. (distance set to infinity)
- two kinds of messages
  - request
  - response

Routing in the Internet

- The Global Internet consists of Autonomous Systems (AS) interconnected with each other:
  - Stub AS: small corporation: one connection to other AS's
  - Multihomed AS: large corporation (no transit): multiple connections to other AS's
  - Transit AS: provider, hooking many AS's together

Internet AS Hierarchy

Inter-AS border (exterior gateway) routers



Intra-AS interior routers

Intra-AS Routing

- Also known as Interior Gateway Protocols (IGP)
- Most common Intra-AS routing protocols:
  - RIP: Routing Information Protocol
  - OSPF: Open Shortest Path First
  - IGRP: Interior Gateway Routing Protocol (Cisco proprietary)

Link State (OSPF)

- Reliable Flooding for dissemination
  - *A nodes ends its link-state information out on all of its directly connected links which in turn forward it on all of their directly connected links.*
  - Link State Packet (LSP)
    - ID of node creating the packet
    - List of directly connected neighbours + cost
    - A sequence number
    - TTL
  - Example
    - Small seq. # ~ old LSP
    - The fact that an LSP is not sent back on the link from which it is received allows to put an end to the flooding

Link State (OSPF)

- LSPs are generated in the case of a change in topology or due to timer expiry
- Hello packet to test if a node is alive or not
- Design goals
  - Newest information must be flooded ASAP
    - Minimize overhead
      - How?
    - Seq # & TTL ensure correctness of information
- Route Calculation ~ Dijkstra's Algorithm

Link State (OSPF)

- Characteristics
  - Stabilizes quickly
    - Promptly Responds to topology changes
  - Information stored at each node is quite large
    - Thus potential problems with scalability

- Difference in DV and LS?

OSPF (Open Shortest Path First)

- "open": publicly available
- Uses Link State algorithm
  - LS packet dissemination
  - Topology map at each node
  - Route computation using Dijkstra's algorithm

- OSPF advertisement carries one entry per neighbor router
- Advertisements disseminated to entire AS (via flooding)
  - Carried in OSPF messages directly over IP (rather than TCP or UDP

Summary

Distance Vector

- Router maintains a routing table that lists known networks, direction (vector) to each network, and the distance to each network. (The meaning of "distance" is based on the routing protocol metric.)
- Router periodically (every 30 seconds, for example) transmits the routing table via a broadcast packet that reaches all other routers on the local segments. (To conserve bandwidth, some advanced distance-vector protocols don't send periodically but, rather, only when there is a change.)
- Router updates the routing table, if necessary, based on received broadcasts.
- If router receives information about a new route to a network, the router determines if the advertising router has a shorter path. If it does, this router updates its table to say that packets should be sent in the direction of the advertising router. ("Shorter" is defined by the metric.)

Link State

- Routers send updates only when a link changes state
- Neighbors propagate the change to their neighbors
- Routers update their database if necessary
- Link-state algorithms keep a database of routers and links between them.
- Link-state algorithms think of the internetwork as a graph instead of a list.
- When changes occur, link-state algorithms apply Dijkstra's shortest-path algorithm to find the shortest path between any two nodes.
- Unlike distance-vector protocols, a simple comparison of distance isn't sufficient. The router must run the shortest-path algorithm, which can be CPU-intensive.

Questions?

That's all for today!