15-441 Computer Networking
Lecture 11 – Routing

Peter Steenkiste

Departments of Computer Science and Electrical and Computer Engineering

15-441 Networking, Spring 2008

http://www.cs.cmu.edu/~dga/15-441/S08

Outline

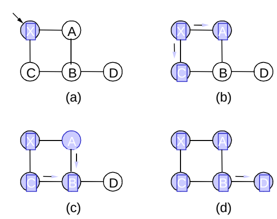
- Link State
- OSPF
- IP Multicast Service Basics
- Host/Router Interaction
- MOSPF/DVMRP

Link State Protocol Concept

- Every node gets complete copy of graph
  - » Every node "floods" network with data about its outgoing links
- Every node computes routes to every other node
  - » Using single-source, shortest-path algorithm
- Process performed whenever needed
  - » When connections die / reappear

Sending Link States by Flooding

- X Wants to Send Information
  - » Sends on all outgoing links
- When Node Y Receives Information from Z
  - » Send on all links other than Z

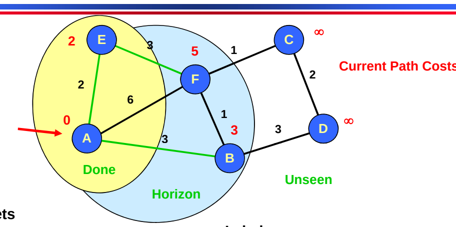


- Need to stop propagation
  - » Use sequence number to recognize and discard old packets
  - » Also: limit hop count or time in the network

Dijkstra's Algorithm

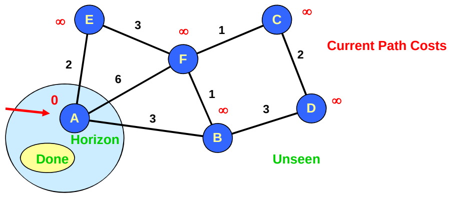
- Given
  - » Graph with source node s and edge costs c(u,v)
  - » Determine least cost path from s to every node v
- Shortest Path First Algorithm
  - » Traverse graph in order of least cost from source

Dijkstra's Algorithm: Concept



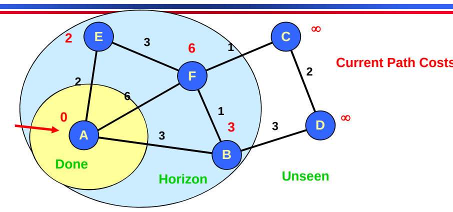
- Node Sets
  - » Done
    - –Already have least cost path to it
  - » Horizon:
    - –Reachable in 1 hop from node in Done
  - » Unseen:
    - –Cannot reach directly from node in Done
- Label
  - » d(v) = path cost
    - – From s to v
- Path
  - » Keep track of last link in path

Dijkstra's Algorithm: Initially



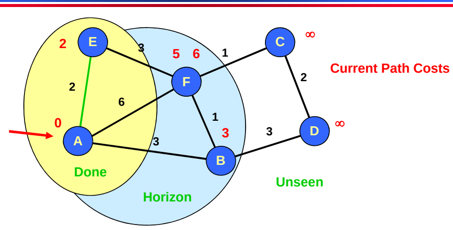
- No nodes done
- Source in horizon

Dijkstra's Algorithm: Initially



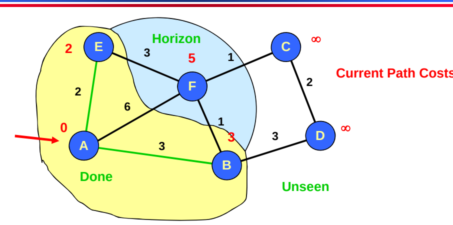
- d(v) to node A shown in red
  - » Only consider links from done nodes

Dijkstra's Algorithm



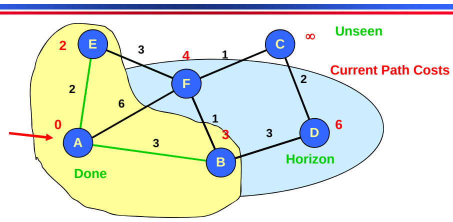
- Select node v in horizon with minimum d(v)
- Add link used to add node to shortest path tree
- Update d(v) information

Dijkstra's Algorithm



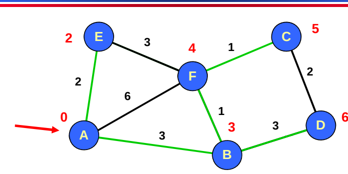
- Repeat…

Dijkstra's Algorithm



- Update d(v) values
  - » Can cause addition of new nodes to horizon

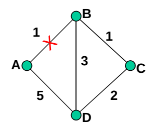
Dijkstra's Algorithm



- Final tree shown in green

Link State Characteristics

- With consistent LSDBs*, all nodes compute consistent loop-free paths
- Can still have transient loops
  - » Routers may update database at slightly different times



Packet from C→A may loop around BDC if B knows about failure and C & D do not

OSPF Routing Protocol

- Open
  - » Open standard created by IETF
- Shortest-path first
  - » Another name for Dijkstra's algorithm
- More prevalent than RIP

OSPF Reliable Flooding

- Transmit link state advertisements
  - » Originating router
    - – Typically, minimum IP address for router
  - » Link ID
    - – ID of router at other end of link
  - » Metric
    - – Cost of link
  - » Link-state age
    - – Incremented each second
    - – Packet expires when reaches 3600
  - » Sequence number
    - – Incremented each time sending new link information

OSPF Flooding Operation

- Node X Receives LSA from Node Y
  - » With Sequence Number q
  - » Looks for entry with same origin/link ID
- Cases
  - » No entry present
    - – Add entry, propagate to all neighbors other than Y
  - » Entry present with sequence number p < q
    - – Update entry, propagate to all neighbors other than Y
  - » Entry present with sequence number p > q
    - – Send entry back to Y
    - – To tell Y that it has out-of-date information
  - » Entry present with sequence number p = q
    - – Ignore it

Flooding Issues

- When should it be performed
  - » Periodically
  - » When status of link changes
    - – Detected by connected node
- What happens when router goes down & back up
  - » Sequence number reset to 0
    - – Other routers may have entries with higher sequence numbers
  - » Router will send out LSAs with number 0
  - » Will get back LSAs with last valid sequence number p
  - » Router sets sequence number to p+1 & resends

Adoption of OSPF

- RIP viewed as outmoded
  - » Good when networks small and routers had limited memory & computational power
- OSPF Advantages
  - » Fast convergence when configuration changes

Comparison of LS and DV Algorithms

Message complexity
- **LS:** with n nodes, E links, O(nE) messages
- **DV:** exchange between neighbors only

Speed of Convergence
- **LS:** Complex computation
  - But…can forward before computation
  - may have oscillations
- **DV**: convergence time varies
  - may be routing loops
  - count-to-infinity problem
  - (faster with triggered updates)

Space requirements:
- LS maintains entire topology
- DV maintains only neighbor state

Comparison of LS and DV Algorithms

Robustness:

LS:
- node can advertise incorrect *link* cost
- each node computes only its *own* table

DV:
- DV node can advertise incorrect *path* cost
- each node's table used by others
  - errors propagate thru network
- Other tradeoffs
  - Making LSP flood reliable

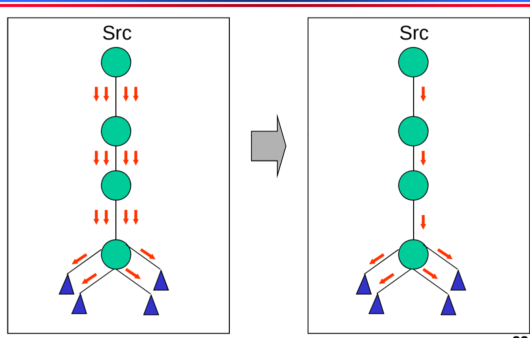
Outline

- Link State
- OSPF
- IP Multicast Service Basics
- Host/Router Interaction
- MOSPF/DVMRP

Multicast Routing

- Unicast: one source to one destination
- Multicast: one source to many destinations
- Main goal: efficient data distribution

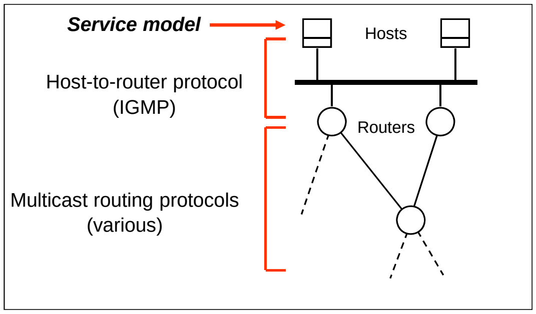
Multicast – Efficient Data Distribution



Example Applications

- Broadcast audio/video
- Push-based systems
- Software distribution
- Web-cache updates
- Teleconferencing (audio, video, shared whiteboard, text editor)
- Multi-player games
- Server/service location
- Other distributed applications

IP Multicast Architecture



Logical Naming

- Single name/address maps to logically related set of destinations
  - » Destination set = multicast group
- Key challenge: scalability
  - » Single name/address independent of group growth or changes

Multicast Router Responsibilities

- Learn of the existence of multicast groups (through advertisement)
- Identify links with group members
- Establish state to route packets
  - » Replicate packets on appropriate interfaces
  - » Routing entry:

Src, incoming interface	List of outgoing interfaces

IP Multicast Service Model (rfc1112)

- Each group identified by a single IP address
- Groups may be of any size
- Members of groups may be located anywhere in the Internet
- Members of groups can join and leave at will
- Senders need not be members
- Group membership not known explicitly
- Analogy:
  - » Each multicast address is like a radio frequency, on which anyone can transmit, and to which anyone can tune-in.

IP Multicast Addresses

- Class D IP addresses
  - » 224.0.0.0 – 239.255.255.255

1 1 1 0	Group ID

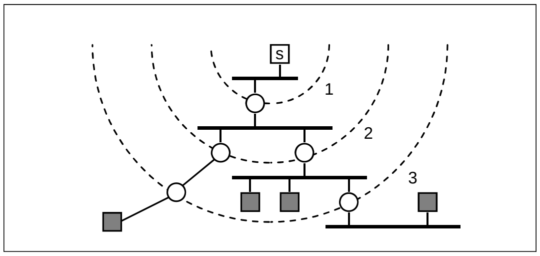
- How to allocated these addresses?
  - » Well-known multicast addresses, assigned by IANA
  - » Transient multicast addresses, assigned and reclaimed dynamically, e.g., by “sdr” program

IP Multicast API

- Sending – same as before
- Receiving – two new operations
  - » Join-IP-Multicast-Group(group-address, interface)
  - » Leave-IP-Multicast-Group(group-address, interface)
  - » Receive multicast packets for joined groups via normal IP-Receive operation
  - » Implemented using socket options

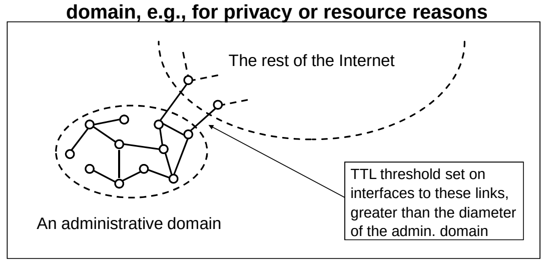
Multicast Scope Control – Small TTLs

- TTL expanding-ring search to reach or find a nearby subset of a group



Multicast Scope Control – Large TTLs

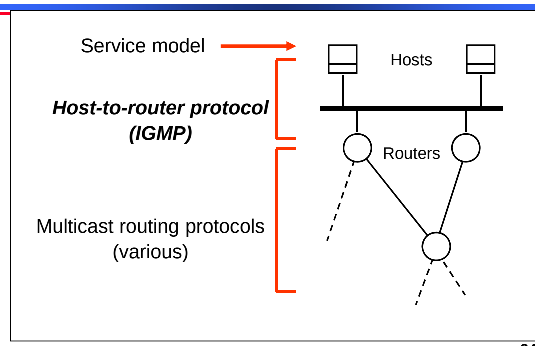
- Administrative TTL Boundaries to keep multicast traffic within an administrative domain, e.g., for privacy or resource reasons



Overview

- IP Multicast Service Basics
- Host/Router Interaction
- MOSPF/DVMRP

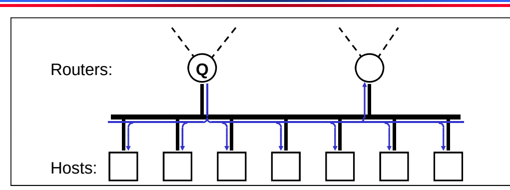
IP Multicast Architecture



Internet Group Management Protocol

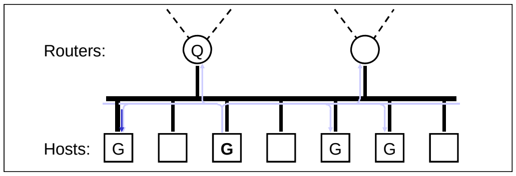
- End system to router protocol is IGMP
- Each host keeps track of which mcast groups are subscribed to
  - » Socket API informs IGMP process of all joins
- Objective is to keep router up-to-date with group membership of entire LAN
  - » Routers need not know who all the members are, only that members exist

How IGMP Works



- On each link, one router is elected the “querier”
- Querier periodically sends a Membership Query message to the all-systems group (224.0.0.1), with TTL = 1
- On receipt, hosts start random timers (between 0 and 10 seconds) for each multicast group to which they belong

How IGMP Works (cont.)



- When a host's timer for group G expires, it sends a Membership Report <u>to group G</u>, with TTL = 1
- Other members of G hear the report and stop their timers
- Routers hear <u>all</u> reports, and time out non-responding groups

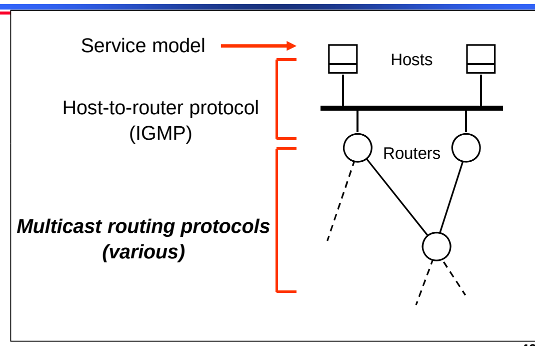
How IGMP Works (cont.)

- Note that, in normal case, only one report message per group present is sent in response to a query
  - » Power of randomization + suppression
- Query interval is typically 60-90 seconds
- When a host first joins a group, it sends one or two immediate reports, instead of waiting for a query

Overview

- IP Multicast Service Basics
- Host/Router Interaction
- MOSPF/DVMRP

IP Multicast Architecture



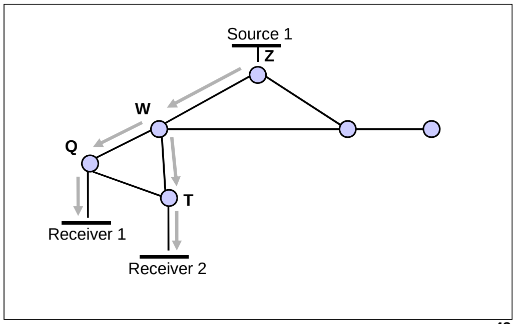
Routing Techniques

- Basic objective – build distribution tree for multicast packets
- Flood and prune
  - » Begin by flooding traffic to entire network
  - » Prune branches with no receivers
  - » Examples: DVMRP, PIM-DM
  - » *Unwanted state where there are no receivers*
- Link-state multicast protocols
  - » Routers advertise groups for which they have receivers to entire network
  - » Compute trees on demand
  - » Example: MOSPF
  - » *Unwanted state where there are no senders*

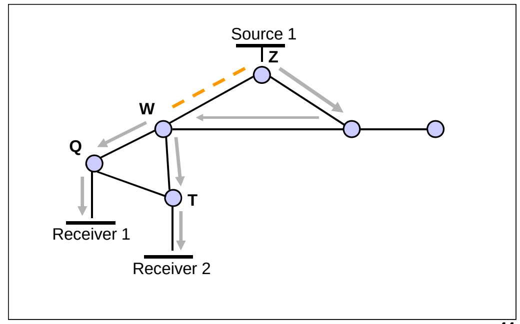
Multicast OSPF (MOSPF)

- Add-on to OSPF (Open Shortest-Path First, a link-state, intra-domain routing protocol)
- Multicast-capable routers flag link state routing advertisements
- Link-state packets include multicast group addresses to which local members have joined
- Routing algorithm augmented to compute shortest-path distribution tree from a source to any set of destinations

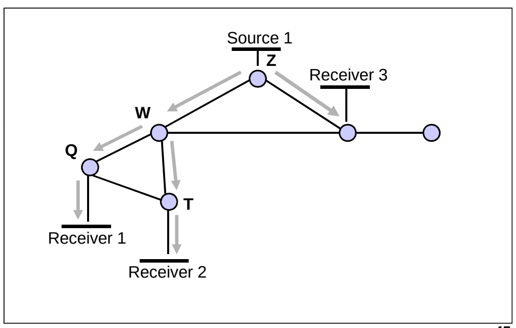
Example



Link Failure/Topology Change



Membership Change



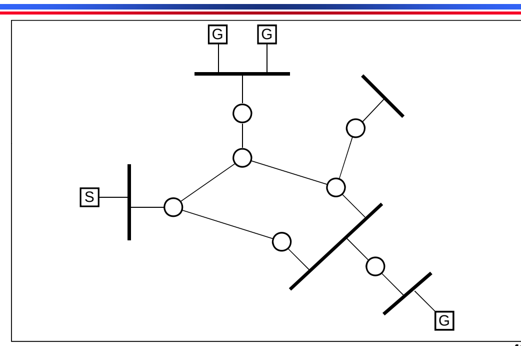
Impact on Route Computation

- Cannot pre-compute multicast trees for all possible sources
- Compute on demand when first packet from a source S to a group G arrives
- New link-state advertisement
  - » May lead to addition or deletion of outgoing interfaces if it contains different group addresses
  - » May lead to re-computation of entire tree if links are changed

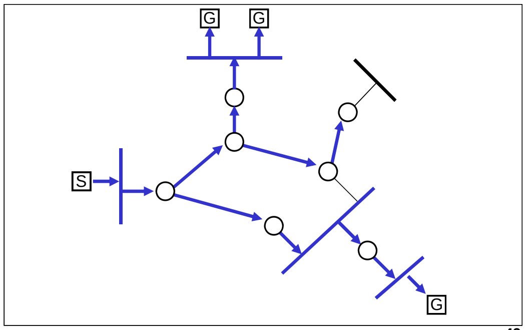
Distance-Vector Multicast Routing

- DVMRP consists of two major components:
  - » A conventional distance-vector routing protocol (like RIP)
  - » A protocol for determining how to forward multicast packets, based on the routing table
- DVMRP router forwards a packet if
  - » The packet arrived from the link used to reach the source of the packet (reverse path forwarding check – RPF)
  - » If downstream links have not pruned the tree

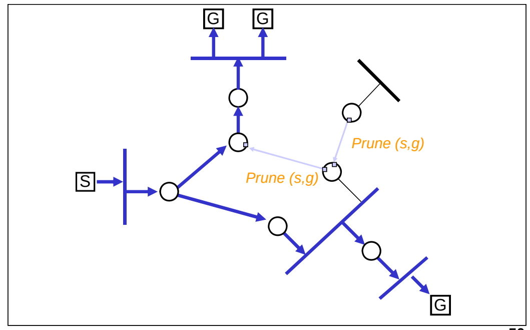
Example Topology



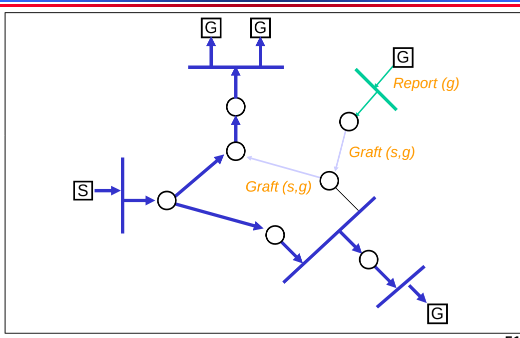
Broadcast with Truncation



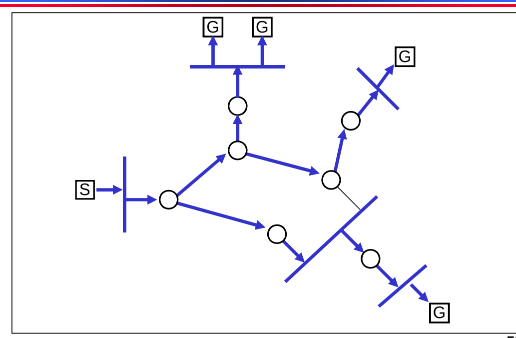
Prune



Graft



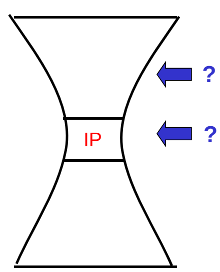
Steady State



Failure of IP Multicast

- Not widely deployed even after 15 years!
  - » Use carefully – e.g., on LAN or campus, rarely over WAN
- Various failings
  - » Scalability of routing protocols
  - » Hard to manage
  - » Hard to implement TCP equivalent
  - » Hard to get applications to use IP Multicast without existing wide deployment
  - » Hard to get router vendors to support functionality and hard to get ISPs to configure routers to enable

Supporting Multicast on the Internet



At which layer should multicast be implemented?

Alternative: application layer multicast – “peer-to-peer”