Multicast Routing

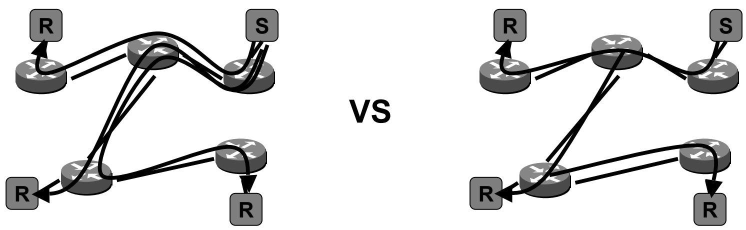
CSE 561 Lecture 13, Spring 2002.
David Wetherall

Overview

- Multicast goals and service model
- Multicast Routing
  - Dense: Distance Vector / Link State
  - Sparse: Shared tree
- Limiters (compare to end-system multicast)
  - Scalability
  - Deployment issues
  - Operational/Economic issues
  - Applications?

Motivation

- Efficient delivery to multiple destinations (e.g. video broadcast)



- Service discovery; communication with a layer of indirection
  - Publish / subscribe communications model
  - Don’t need to know destinations

Multicast on shared LAN

- Efficient multicast is straightforward
  - the medium is broadcast
- How do we add a layer of indirection?
  - Define new multicast addresses to represent groups
  - Let hosts join / leave receiver groups as they please by filtering incoming packets according to local group membership
  - Allow anyone to send to a multicast address
- Much of Internet multicast can be viewed as trying to replicate this success in the wide area … it gets hard!

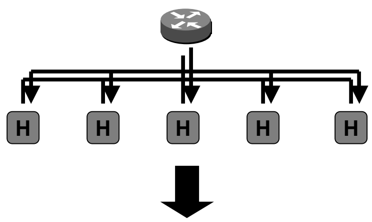
IP Multicast service model

- Communications based on groups
  - Special IP addresses (class D) for “multicast groups”
  - Anyone can join / leave group anytime
  - Anyone can send to group anytime (even non-members)
- Unreliable datagram service
  - Extension to unicast IP
  - Group membership not visible to hosts
- Scoping to limit spread of packets
  - In the wide-area, can set TTL low to reach “nearby” members

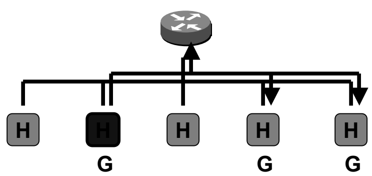
Internet Group Management Protocol (IGMP)

- **By internet convention, hosts don’t participate in routing**
  - **IGMP used to communicate group membership between hosts and routers**
- Soft-state protocol
  - Hosts explicitly inform their router about membership
  - Must periodically refresh membership report
  - Routers implicitly timeout groups that aren’t refreshed
  - Why isn’t explicit “leave group” message sufficient?
- Implemented in most of today’s routers and switches

How IGMP works (roughly)



- Router broadcasts *membership query* to 224.0.01 (all-systems group) with ttl=1
- Hosts start random timer (0-10 sec) or each group they have joined



- When a host's timer expires for group G, send *membership report* to group G, with ttl=1
- When a member of G hears a report, they reset their timer for G
- Router times out groups that are not "refreshed" by some host's report

Multicast routing

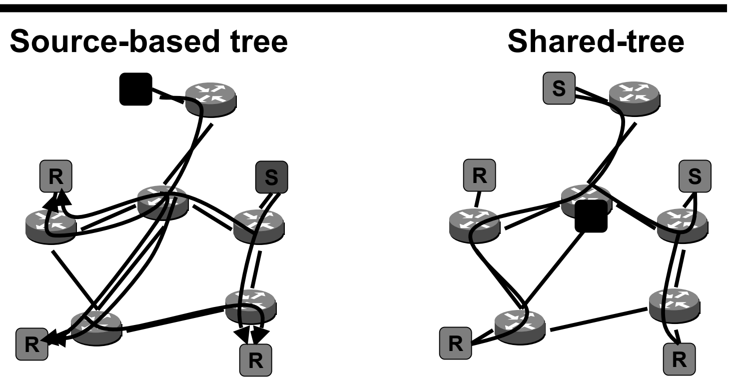
- **Goal: build distribution tree for multicast packets**
  - Efficient tree (ideally, shortest path)
  - Low join / leave latency

- Several approaches
  - Distance Vector / Link State
    - Leverage existing unicast routing protocols
  - Shared tree
    - Unicast / multicast hybrids

Multicast routing taxonomy

- **Source-based tree (Dense mode)**
  - *Separate shortest path tree for each source*
  - **Flood and prune** (DVMRP, PIM-DM)
    - Send multicast traffic everywhere
    - Prune edges that are not actively subscribed to group
  - **Link-state** (MOSPF)
    - Routers flood groups they would like to receive
    - Compute shortest-path trees on demand
- **Shared tree** (CBT, PIM-SM) **(Sparse Mode)**
  - *Single distributed tree shared among all sources*
  - Specify rendezvous point (RP) for group
  - Senders send packets to RP, receivers join at RP
  - RP multicasts to receivers; Fix-up tree for optimization

Source-based vs Shared



- **Efficient trees; low delay, even load**
- **Per-source state in routers (S,G)**
- **Good for dense-area multicast**

- **Higher delay, skewed load, SPOF**
- **Per-group state only (G)**
- **Efficient for sparse-area multicast**

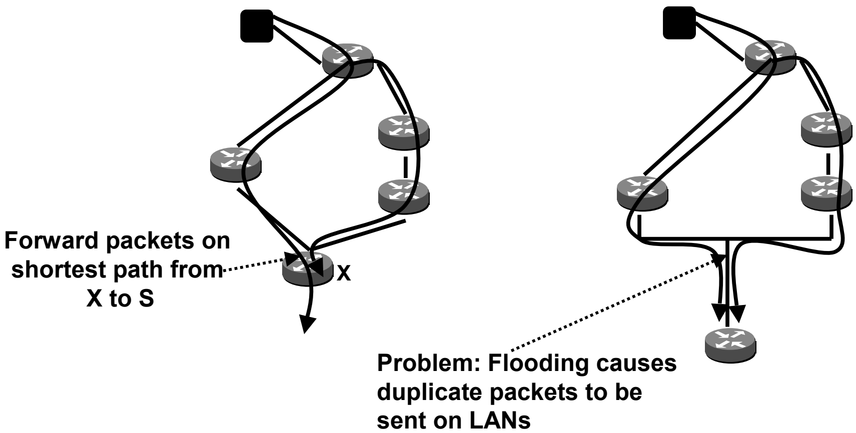
Flood and Prune (DV)

- Extensions to unicast distance vector algorithm
- Goal
  - Multicast packets delivered along shortest-path tree from sender to members of the multicast group
  - Likely have different tree for different senders
- Distance Vector Multicast Routing (DVMRP) developed as a progression of algorithms
  - Reverse Path Flooding (RPF)
  - Reverse Path Broadcast (RPB)
  - Truncated Reverse Path Broadcasting (TRPB)
  - Reverse Path Multicast (RPM)

Reverse Path Flooding (RPF)

- Observation: Shortest-path multicast tree is subtree of shortest-path broadcast tree
- Approach: Use shortest-path broadcast tree
- Use reverse path to determine shortest path
  - Router forwards a packet from S if received from the shortest-path link to S
  - Exactly what is in entry in forwarding table
    - To reach S along shortest path, use link L
    - If received packet from S on L, it came along shortest path
- How are packets forwarded?
  - Flooding – forward packets to multicast address out to all links except incoming link (hence reverse path flooding)

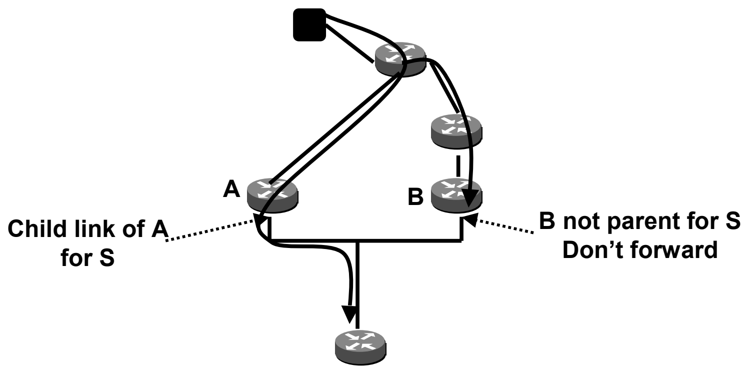
Example:
Reverse Path Forwarding



Solution:
Reverse Path Broadcast (RPB)

- Flooding vs. broadcast
  - With flooding, a single packet can be sent along an individual link multiple times
    - Each router attached to link can potentially forward same packet
  - RPB sends a packet along a link at most once
- Approach: Define parent and child routers for each link
  - Relative to each link and each source S
  - Router is a parent for link if it has minimum path to S
  - All other routers on the link are children
  - Only forward on child links for S
- How to decide parent and children routers for link?
  - In routing updates; router determines if is parent

Example:
Reverse Path Broadcasting



Truncated RPB (TRPB)

- Problem: Broadcast is not multicast
  - Broadcast only good for small internetworks, infrequent sends
- Approach: don't forward packets to networks that aren't group members
  - Identify leaves
    - Child links not used by any other routers to reach S
    - Send periodic updates about next-hops to S
  - Detect group membership
    - Multicast group membership locally (i.e. IGMP)
  - Only add links to leaves that are group members

Reverse Path Multicast (RPM)

- Problem: Still broadcasting up to leaf networks
- Idea: Instead of actively building tree, use reports to actively prune tree

- Start with a full broadcast tree to all links (RPB),
- Prune (S,G) at leaf if it has no members
  – Send Non-Membership Report (NMR) to prev-hop for S
- If all children of router R prune (S,G)
  – Send NMR for (S,G) to parent of R
- Soft-state management (must refresh NMR or rejoin)
- New group member sends graft (anti-prune) message

Link State

- Use existing link-state routing algorithm (e.g. OSPF)
- Idea: include active groups in LSPs
  – Each router can compute shortest path tree from source to all destinations for any group
  – Trigger new flood on group membership change

- Performance issues
  – Expensive to precompute all (S,G) trees
  – Keep cache of trees and compute new trees on demand when new (S,G) packet arrives
  – Workload / topology dependant

- Best known example: MOSPF

Shared tree approaches

- Unicast packets to Rendezvous Point (RP), which multicasts packet on shared tree
- Tree construction
  - Receivers send join messages to RP
  - Intermediate routers install state to create per-group tree
  - Key advantage is routers only store O(G) state
  - Potential optimizations: reroute to source-specific trees for local group members or high data-rate sources
  - Example: CBT, PIM-SM
- Issues
  - Delay, fault tolerance, RP selection

IP Multicast today

- IP Multicast has generated 1000s of papers, but has not been widely deployed in the Internet…
- Why?
  - Scalability
  - General deployment difficulties (Mbone)
  - Inter-domain multicast complexity
  - Economics of multi-source multicast
  - Applications?

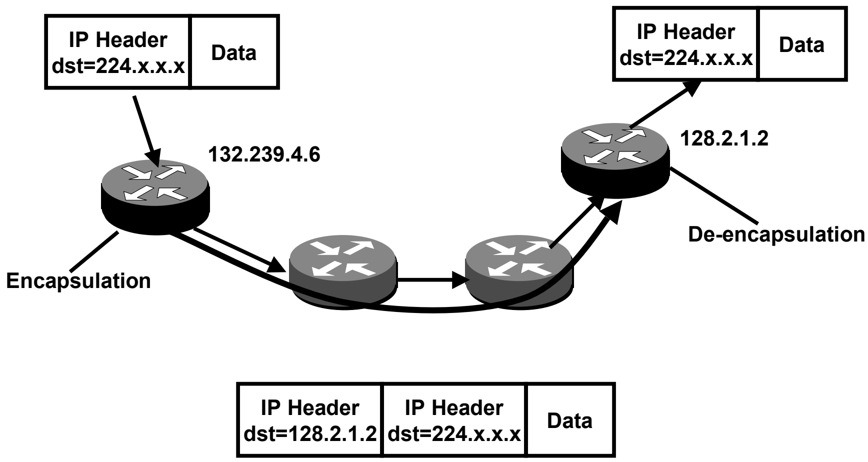
Scalability

- How much state does a router need for multicast?
  - Dense mode O(senders * groups)
  - Sparse mode O(groups)
  - Compare to O(#networks) for unicast …
- Problem: can't aggregate multicast addresses in the same way as unicast addresses – no hierarchy
- Also address allocation: which address to use for a new group?
  - No standard but must be globally unique
  - Global random selection
  - Per-domain addressing (MASC, GLOP)

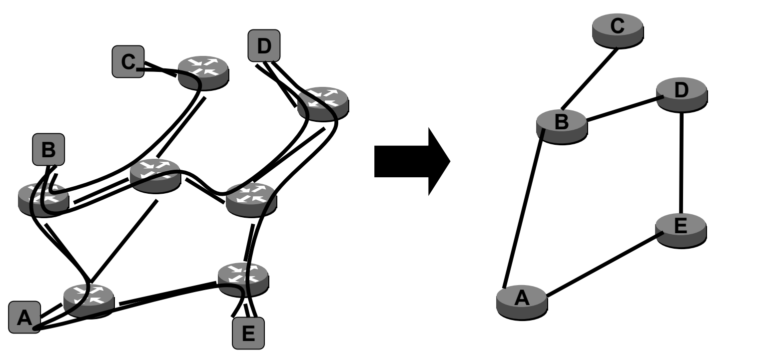
Multicast evolution

- How to deploy a new network-layer service?
  - Difficult to change router software
  - Difficult to change all routers
- Mbone (tunneling)
  - Special multicast routers (built from PCs / Workstations)
  - Construct virtual topology between them (overlay)
  - Run routing protocol over virtual topology
  - Virtual point-to-point links called ***tunnels***
    - Multicast traffic encapsulated in IP datagrams
    - Multicast routers forward over tunnels according to computed virtual next-hop

Tunneling



Virtual overlay network



Real topology with tunnels **Virtual overlay topology**

Mbone Pro/Con

- Success story
  - Multicast video to 20 sites in 1992
  - Easy to deploy, no explicit router support
  - Ran DVMRP and had 100s of routers
- Drawbacks
  - Manual tunnel creation / maintenance
  - Inefficient
  - No routing policy (single tree)
  - Why would an ISP deploy a new mbone node?

Operational / Economic issues

- Billing model
  - Inconsistent with input-rate-based billing
  - No group management (how big is group?)
- ISP router migration cycle
  - Can't afford new routers on edge
- Group management
  - Who is in the group? Who can send? Security
- Domain independence
  - Do I want my customers MC controlled by an RP in a competitors domain?
  - Why run an RP for which I have no senders or receivers?
- Complexity, e.g., multicast address allocation

Proposal: Single source multicast

- Reduce complexity and match ISP economic needs by limiting group to single source
- Example: EXPRESS [Holbrook and Cheriton99]
  - Root of tree at source, all receives use RPM to join at source
  - Use src and dst addresses to define group (src, channel)
  - Recursive CountQuery message to count group members
  - Closed groups (authentication to subscribe)
- Also Simple Multicast (Perlman etc.)
- And even more extreme is End-System multicast ☺