Broadcast Routing

- Deliver packets from source to all other nodes
- Source duplication is inefficient:
  - Source duplication: how does source determine recipient addresses?

![Diagram showing source and in-network duplication]
**In-network duplication**

- Flooding: when node receives broadcast packet, sends copy to all neighbors
  - Problems: cycles & broadcast storm
- Controlled flooding: node only broadcasts packet if it hasn't broadcast the same packet before
  - Node keeps track of packet IDs already broadcasted
  - Or reverse path forwarding (RPF): only forward packet if it arrived on shortest path between node and source
- Spanning tree
  - No redundant packets received by any node

---

**Spanning Tree**

- First construct a spanning tree
- Nodes forward copies only along spanning tree

![Diagram of spanning tree](a) Broadcast initiated at A

![Diagram of spanning tree](b) Broadcast initiated at D
Spanning Tree: Creation

- Center node
- Each node sends unicast join message to center node
  - Message forwarded until it arrives at a node already belonging to spanning tree

(a) Stepwise construction of spanning tree  
(b) Constructed spanning tree

Multicast Routing: Problem Statement

- **Goal**: find a tree (or trees) connecting routers having local mcast group members
  - tree: not all paths between routers used
  - source-based: different tree from each sender to rcvrs
  - shared-tree: same tree used by all group members

Shared tree  
Source-based trees

Stan Kurkovsky
Multicast routing

- Address indirection
  - A single ID is used for addressing the group of receivers
- Multicast group
- IGMP: Internet Group Management Protocol
  - Hosts inform routers about joining/leaving multicast groups
  - Operates locally
  - Actual multicast routing is achieved by complimentary network layer protocols
    - PIM, DVMRP, MOSPF

Approaches for building mcast trees

Approaches:
- **source-based tree**: one tree per source
  - shortest path trees
  - reverse path forwarding
- **group-shared tree**: group uses one tree
  - minimal spanning (Steiner)
  - center-based trees

- we first look at basic approaches, then specific protocols adopting these approaches
**Shortest Path Tree**

- mcast forwarding tree: tree of shortest path routes from source to all receivers
- Dijkstra's algorithm

![Diagram of Shortest Path Tree]

**Reverse Path Forwarding**

- rely on router's knowledge of unicast shortest path from it to sender
- each router has simple forwarding behavior:

  \[
  \text{if}(\text{mcast datagram received on incoming link on shortest path back to center})
  \]  
  \[
  \text{then} \quad \text{flood datagram onto all outgoing links}
  \]  
  \[
  \text{else} \quad \text{ignore datagram}
  \]
Reverse Path Forwarding: example

- result is a source-specific reverse SPT
  - may be a bad choice with asymmetric links

Reverse Path Forwarding: pruning

- forwarding tree contains subtrees with no mcast group members
  - no need to forward datagrams down subtree
  - "prune" msgs sent upstream by router with no downstream group members
**Shared-Tree: Steiner Tree**

- **Steiner Tree**: minimum cost tree connecting all routers with attached group members
- problem is NP-complete
- excellent heuristics exists
- not used in practice:
  - computational complexity
  - information about entire network needed
  - monolithic: rerun whenever a router needs to join/leave

---

**Center-based trees**

- single delivery tree shared by all
- one router identified as "center" of tree
- to join:
  - edge router sends unicast join-msg addressed to center router
  - join-msg "processed" by intermediate routers and forwarded towards center
  - join-msg either hits existing tree branch for this center, or arrives at center
  - path taken by join-msg becomes new branch of tree for this router

- Suppose R6 chosen as center:

![Diagram](image-url)
**Internet Multicasting Routing: DVMRP**

- **DVMRP**: distance vector multicast routing protocol, RFC1075
- **flood and prune**: reverse path forwarding, source-based tree
  - RPF tree based on DVMRP's own routing tables constructed by communicating DVMRP routers
  - no assumptions about underlying unicast
  - initial datagram to mcast group flooded everywhere via RPF
  - routers not wanting group: send upstream prune msgs
- **soft state**: DVMRP router periodically (1 min.) "forgets" branches are pruned:
  - mcast data again flows down unpruned branch
  - downstream router: reprune or else continue to receive data
- routers can quickly regraft to tree
  - following IGMP join at leaf
- odds and ends
  - commonly implemented in commercial routers
  - Mbone routing done using DVMRP

**Tunneling**

Q: How to connect “islands” of multicast routers in a “sea” of unicast routers?

- mcast datagram encapsulated inside "normal" (non-multicast-addressed) datagram
- normal IP datagram sent thru “tunnel” via regular IP unicast to receiving mcast router
- receiving mcast router unencapsulates to get mcast datagram
PIM: Protocol Independent Multicast

- not dependent on any specific underlying unicast routing algorithm (works with all)
- two different multicast distribution scenarios:
  - **Dense:**
    - group members densely packed, in “close” proximity.
    - bandwidth more plentiful
    - Consequences:
      - group membership by routers assumed until routers explicitly prune
      - data-driven construction on mcast tree (e.g., RPF)
      - bandwidth and non-group-router processing profligate
  - **Sparse:**
    - # networks with group members small wrt # interconnected networks
    - group members “widely dispersed”
    - bandwidth not plentiful
    - Consequences:
      - no membership until routers explicitly join
      - receiver-driven construction of mcast tree (e.g., center-based)
      - bandwidth and non-group-router processing conservative

PIM - Dense Mode

flood-and-prune RPF, similar to DVMRP but

- underlying unicast protocol provides RPF info for incoming datagram
- less complicated (less efficient) downstream flood than DVMRP reduces reliance on underlying routing algorithm
- has protocol mechanism for router to detect it is a leaf-node router
PIM - Sparse Mode

- center-based approach
- router sends *join* msg to rendezvous point (RP)
  - intermediate routers update state and forward *join*
- after joining via RP, router can switch to source-specific tree
  - increased performance: less concentration, shorter paths

**sender(s):**
- unicast data to RP, which distributes down RP-rooted tree
- RP can extend mcast tree upstream to source
- RP can send *stop* msg if no attached receivers
  - "no one is listening!"

![Diagram](image-url)