CSE 222
Graduate Networking

Winter 2001

Lecture 13: Multicast

Geoffrey M. Voelker

Topic Overview

Three lectures discussing multicast routing:

1. Multicast as an IP layer service
   - Seminal paper by Deering
2. Multicast++
   - Protocol Independent Multicast (PIM)
   - Single-source multicast
3. Reliable Multicast
   - Tough problem, creative solutions

The Network Pariah

- In January 2001, CTSB (?) workshop on future network research directions
  - Situation: Room full of famous CS researchers
  - Q: What network research topic should be dropped?
  - A: "9 out of 10" → Multicast
  - …or systems that require multicast to work
- So why spend three days studying it?
  - Still need to know how it works, what the challenges are, why it hasn't succeeded as originally envisioned
  - Interesting routing problems, might be able to adapt algorithms/approaches to other problems

Lecture Overview

- Motivation
- Service model
  - Host groups
  - Group types
  - Communication environment
- Routing algorithms
  - Single spanning tree
  - Distance vector
  - Link state
  - Hierarchical

(Sorry, no pictures…)

Motivation for Multicast

Why provide multicast delivery?

- Efficient delivery to multiple destinations
  - When same data needs to be sent to multiple hosts
  - More efficient than unicasting copies to each destination
  - Replicated distributed systems (file system, database)
  - Video conferencing, collaborative applications
- Robust unknown destination delivery
  - Level of indirection between sender and receivers
  - Database, name server queries
  - Service discovery, location
  - Notifications, updates

Multicast Model

- Multicast model defines
  - Service interface – semantics that applications can rely upon
  - Multicast group types
  - Communication environment
- Multicast at the IP layer (IP Multicast)
  - Implemented in the network layer
  - Implies changes to the routers
- Assumption: fewer groups on a LAN than hosts
  - Seems odd

Multicast Service Interface

What is the model for using multicast?

- Host groups
  - Group of hosts that should receive a multicast packet
  - Named using a single multicast address
    - » Class D IP addresses (224.0.0.0 – 239.255.255.255)
  - Unlimited in size, arbitrary location, multiple senders
  - Transitory membership (arbitrary join and leave)
- Advantages
  - Similar addressing format as unicast packets
  - Sender does not need to know group members
  - Dynamic join/leave matches workload and network behavior
  - Can be implemented efficiently (?)

Group Types

What kind of multicast groups should be supported?

- Pervasive – directory services, netnews, etc.
  - Members on most networks
  - Long-lived, well-known addresses
  - Broadcast an option
- Sparse – video conferencing, wide-area games, etc.
  - Members on few networks, widely-scattered
  - Long-lived or transient
  - Requires selective delivery (defn. of sparse vs. pervasive)
- Local – distributed parallel apps, LAN games, etc.
  - Restricted to single network
  - Transient (order of execution of a single program)

Communication Environment

What is assumed about underlying networks?

- Internetworks
  - Multi-access networks connected in arbitrary topologies
  - LANs support intranetwork multicast
  - Global multicast addresses can be mapped to LAN addrs
- Delivery
  - Best-effort datagram reliability, same as unicast IP
  - What about reliability, flow control, congestion control, etc.?
- Scope
  - Limit spread of packets
  - Use TTLs

Performance Considerations

- Multicast algorithm performance depends on
  - Network utilization (message length and frequency)
  - Router state (space)
  - Router processing (cpu)
  - Host processing (cpu)
- Other performance goals
  - Loss probability and delay similar to unicast datagram service
  - Short join latency
    - » Delay between a join and delivery of first packet

Multicast Routing Algorithms

- Paper describes four algorithms for multicast
  - Single spanning tree
  - Distance vector
  - Link state
  - Hierarchical
- We'll go through each
- Some techniques are common throughout

Single Spanning Tree

- Bridged LANs
  - Broadcast too expensive
    - » All hosts on all segments have to process packets
  - Multicast limits transmission and processing
- Approach: Build a spanning tree
  - Routers are nodes, links are edges
  - Minimal transmission across segments
  - One spanning tree defines routing topology for all members
- Issues
  - How do routers know who is in a multicast group?
  - How do routers decide to forward multicast packets?

Group Membership

- Membership announcement for group G
  - Define an “all-bridges” group B that all bridges listen to
  - Hosts transmit “membership report” to B from G
    - » Analogous to bridges learning unicast hosts by watching packets
- Bridges create a multicast routing entry
  - Indexed by multicast address
  - Record links with members of the multicast group
    - » Note: Members not recorded, just links
  - Link entries are aged to garbage collect members
    - » No explicit “leave” message – performance implications?
  - Packets with multicast address are forwarded to all links (except incoming)

SST Performance

- Overhead is membership reports
  - Reporting time must be faster than expiration time
  - Tolerate packet losses
- Optimization (used in other algorithms, too)
  - Host sends report from G to G (instead of all-bridges B)
  - Bridge replaces to G with to B when forwarding to other links
- Why?
  - Other members on link see packets addressed to G
  - They suppress their own reports
  - Only need one packet per report interval for all hosts on link (for each multicast group)

Distance Vector

- Second algorithm based upon distance vector
  - Recall routers exchange forwarding tables
  - Examples: RIP, BGP
- Goal
  - Multicast packets delivered along shortest-path tree from sender to members of the multicast group
  - Note: Likely have different tree for different senders
- 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 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)

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 parent forwards broadcast packets
- How to decide parent and children routers for link?
  - All routers see forwarding tables of all other routers on the same link, hence all can locally agree on shortest-path router

Truncated RPB (TRPB)

- Problem: Broadcast is not multicast
  - Broadcast only good for small internetworks, infrequent sends
- Observation: Only need to forward to networks that have multicast group members
  - "Prune" broadcast tree
- Approach: Use membership reports to do pruning
  - Members of G send reports back up tree towards S
- Problem: Does not scale well
  - Need reports for all groups over all trees: |G|*|S|

TRPB (2)

- Fix: Only prune non-member leaf networks
  - Now need to identify leaf networks that have no members
- Leaves
  - Child links that no other router uses to reach source S
  - Every router needs to know next-hop destinations in forwarding tables in all other routers attached to that link
    - » Can use "infinity" value used by split horizon for next-hop ident.
- Membership
  - Use membership reports from SST (but do not forward)
- Overhead
  - |G| report messages per link
  - Bitmap for each router entry identifying leaves

Reverse Path Multicast (RPM)

- Problem: Still broadcasting up to leaf networks
- Idea: Instead of actively building tree, use reports to actively prune tree
  - TRPB initially starts with a pruned tree, and then uses reports to identify leaf networks with members to extend tree
  - Why not do the reverse?
  - Start with a full broadcast tree to all links (RPB), and then actively prune tree back
  - Use non-membership reports (NMRs) to do pruning
- Called reverse path multicast (RPM)
  - Note that it starts with broadcast, then refines to a multicast

Non-Membership Reports

- Who sends?
  - Routers with child links that are leaves with no members
- Where sent?
  - To router one hop back to source S
- Recursive
  - Routers that receive NMRs from all child links generate NMRs
  - Non-membership propagates back to source
- Expiration and cancellation
  - NMRs expire to reconnect a link to multicast tree
  - Routers cancel NMRs to immediate rejoin tree (new members)
  - Why have expire if there is cancellation? (Think GC)

RPM Overhead

- Overhead of TRPB plus
  - Space, processing, and network utilization of NMRs
  - Entirely workload dependent
  - Use aggregation to reduce router space requirements

Link State

- Third algorithm based upon link state routing
  - All routers flood forwarding tables, calc shortest path locally
  - OSPF, ISIS
- Idea: Add set of groups with members on link to routing state
  - Flood this to all routers as well
  - Each router can compute shortest path multicast for all groups
  - Trigger flood when groups appear/disappear on a link
  - Use membership reports to notify routers of groups on links
  - Only need one router to monitor/react to reports

Link State Overhead

- Computing and storing multicast trees
  - Expensive for all senders and all groups
  - Idea: Use a cache of computed multicast trees
  - Recompute on demand, use LRU to expire cache entries
    - » Results in delay for first packet from a new source to a group
  - Flush cache when topology changes
- Group membership volatility
  - Triggers flooding
- Very workload dependent
  - Tradeoff space (cache) for router cpu (recompute trees)
  - Hope group membership changes are infrequent or sparse
  - Can aggregate changes by delaying updates

Hierarchical Multicast

- Can use hierarchy to scale algorithms to large internetworks
  - Treat subdomains (subnetworks) as links in routing algorithms
  - Hitch: all superdomain routers need to be in groups
  - Solution: create a “wild card” group to which all superdomain routers join, modify routing algorithms to send to wild card instead of next-hop router
  - Only needed for transit subdomains, leaves only need one superdomain router

MBone

- MBone: Multicast Backbone for the Internet
  - Overlay network on top of unicast IP
    - » Video conferencing (vic), shared whiteboard (wb)
  - Multicast routers maintain state, tunnnel multicast packets to other multicast routers using unicast addresses
    - » Distance vector multicast routing protocol (DVMRP)
  - Internet Group Management Protocol (IGMP)
    - » Membership reports (e.g., join, but no leave)
  - All-hosts: 224.0.0.1
    - » Keep-alive sent by routers to hosts
  - All-routers: 224.0.0.2
    - » Reply to routers from hosts
  - Network Time Protocol: 224.0.1.1

Discussion

- Performance
  - In addition this paper lacks any performance criteria and actual testing results.
- Router state
  - IP Multicast requires routers to maintain per group state, which not only violates the "stateless" architectural principle of the original design, but also introduces high complexity and serious scaling constraints at the IP layer.
- Vulnerability
  - However, a particular question of mine is that the membership reporting allows a host to join any group freely, but how can we restrict the membership of a multicast group to certain hosts.

Discussion (2)

- Naming
  - IP Multicast requires every group to dynamically obtain a globally unique address from the multicast address space (224-239/8) and it is difficult to ensure this in a scalable, distributed and consistent fashion.
- Transport features
  - I believe that for simple best-effort datagram protocols such as UDP the presented solutions are appropriate but none address the real complexity of allowing a byte-stream oriented protocol such as TCP…Even conceding that the TCP model is inappropriate for multicast a mechanism for timeout/retransmition is still required to provide a reliable datagram service which is necessary for many applications.

Discussion (3)

- Deployment
  - Unfortunately, the extension of routing algorithms discussed here requires that changes be made to the innards of the network.
- Link state
  - I wonder about the case when organizations like ABCnews wants to try video multicast and it is a long lived source, but the viewers (which I presume are group members) come and go at very high frequencies.

For Next Time...

- Ch. 4.4.3, two multicast papers
- Final project info