Multicast in Multistage Networks

Jon Turner
Computer Science & Engineering
Washington University

www.arl.wustl.edu/~jst
Embedded Multicast Routing Tables

- Multicast tables in later stages for dynamic routing
  - determine local branching
  - Indexed by global Multicast Identifier (MI)
- For static routing, tables in all stages
- Output-side lookup for per-copy information
Maintaining Routing Tables

- **Dynamic routing case**
  - each center stage table requires entry for every MI
  - for 10^6 MIs and 16 port switch elements need 2 MB tables
  - later stages require less redundancy
  - hash lookup can be more space-efficient in later stages
  - change to a multicast may change many tables
  - hardware support for control message replication can help

- **Static routing case**
  - routing tables needed in all stages
  - each switch element sees only a small fraction of MIs
  - hash lookup allows storage of only the required routes

- **High speed lookup required**
  - for 16×10 Gb/s switch element, about 3 ns per access
Nonblocking for Static Routing

- **Strong nonblocking condition**
  - system never blocks when adding branch to existing flow
  - spcdup required is impractically large

- **Practical alternative**
  - system never blocks when adding new multicast flow
  - can extend existing flows be re-routing

- **Fat trcc networks can be re-routably nonblocking with moderate speedup**
  - key is to constrain branching in “upward” direction
  - different branching constraints yield different speedup
  - two pass alternative requires smaller speedup
    - route new multicast flow through most lightly loaded relay
Multicast Servers

- Multicast originals sent to server ports, which make copies and forward to destinations
  » network sees only unicast, simplifying traffic regulation
- Requires \((\text{fanout}) \times (\text{flow rate}) \leq (\text{switch bandwidth})\)
  » 10 Gb/s internal links can support 10 Mb/s flows with fanout of 1000
    • inability to handle high bandwidth, high fanout applications may not matter in practice
  » potential for wasted capacity due to bw fragmentation
  » implies known flow rates and/or re-routing as needed
- Can tailor to expected multicast traffic
  » if 10% of outgoing traffic is multicast, need about one server for every 10 outgoing links
Binary Tree Multicast

- Multiple passes thru interconnect
  - binary copying on each pass
  - packet header includes two output port numbers
  - no lookup in interconnect
- Relay points recycle packets
  - perform lookup to provide next two destinations
- Requires only small speedup
- Variant with binary copying at inputs and relays
  - use with VOQ-based traffic regulation
Adding an Endpoint

- Pick a shallow leaf e and recycling port z
- Make e and new leaf f children of z
- Make z a child of e's former parent
- Note: two table entries are changed.
  - independent of switch size and connection fanout
- Number of passes limited to $\log_2(\text{max fanout})$
Dropping an Endpoint

- Let c be leaf to be removed
  - if c has grandparent, let y be its parent, x its grandparent and d its sibling; in x’s VXT entry, replace y with d
  - if c is the child of tree root and has sibling with children, redirect root’s pointers to the sibling’s children
  - if c is the child of tree root and has no sibling, or its sibling is childless, simply remove it
- In all cases, only one table entry changed
Speedup Requirement

- Let $S$ be switch speedup and let $B$ be max multicast session bandwidth (as fraction of external link speed).
- $\text{(# internal tree nodes)} < \text{(# leaves)}$, so total recycling bandwidth less than output bandwidth
  - since total exiting traffic is $\leq n$, there must always be some recycling port with load $< 1$
  - result: if $S \geq 2 \cdot B$, there is always a recycling port that can accommodate a new session
- If $\delta$ is fraction of exiting traffic in multicast sessions, it’s enough to have $S > 1 + \delta + B$
  - note that required speedup independent of $n$
  - examples: If $\delta = B = 1$, a $3x$ speedup is required. If $B = 1/16$ and $\delta = 0.2$, a speedup of 1.26 is enough
Recycling Port Requirements

- Moderate number of ports suffice in most cases
- Adjust capacity used for multicast as needs change
Variants

Instead of recycling at all ports, dedicate \( h \) ports to recycling – this system is nonblocking if VOQ-based traffic regulation for multicast

If copying done at inputs and relays ports, can use

\[
S \geq \frac{(n-k)k}{h} + B \text{ or } \frac{h \geq (6n - B)}{8n - B} + B
\]

- drawback is larger speedup requirement
- acceptable impact for typical multicast fractions
- if all ports used for recycling, need \( S \geq 1+2n-B \)
Many-to-Many Multicast

- Multiple one-to-many multicasts scale poorly
  - $m$-way multicast requires $m(m-2)$ routing table entries
  - adding endpoint requires updating $3m-1$ entries
- Single shared tree much more efficient
  - $2m-1$ entries, adding endpoint updating 3 entries
- Usually want to suppress “upstream copies”
  - requires inclusion of source field in internal cell header