Lecture 3: Multicast Routing

7COM1030 – Multicast and Multimedia Networking

Dr. Xianhui Che (Cherry)

x.che@herts.ac.uk

School of Computer Science

University of Hertfordshire, UK

Topics

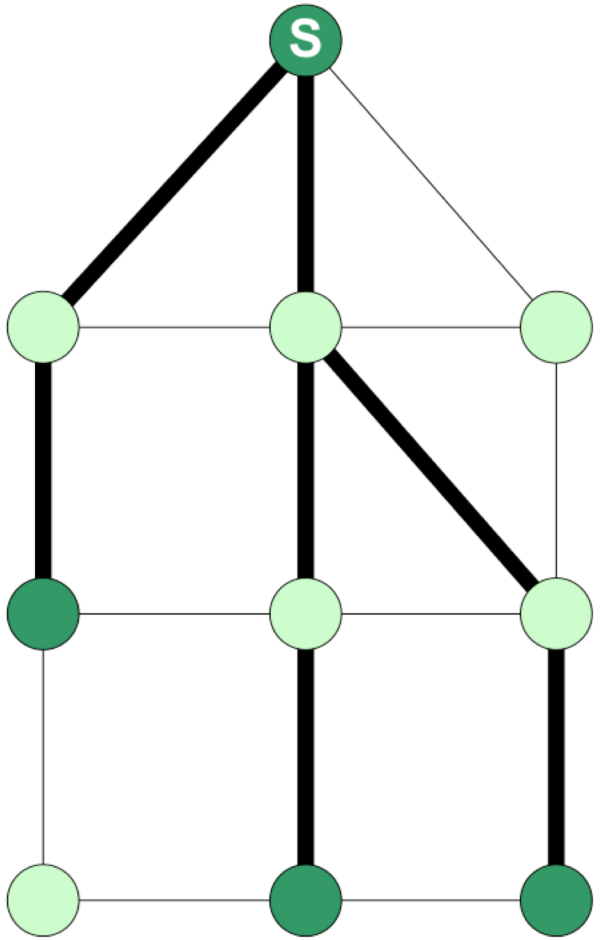
- Multicast Routing Solutions
- Source Based Trees
- Core Based Trees

Multicast Routing Protocols in the Internet

- **Distance Vector Multicast Routing Protocol (DVMRP):**
  - First multicast routing protocol
  - Implements flood-and-prune
- **Multicast Open Shortest Path First (MOSPF):**
  - Multicast extensions to OSPF. Each router calculates a shortest-path tree based on link state database
  - Not widely used
- **Core Based Tree (CBT):**
  - First core-based tree routing protocol
- **Protocol Independent Multicast (PIM):**
  - Runs in two modes: PIM Dense Mode (PIM-DM) and PIM Sparse Mode (PIM-SM).
  - PIM-DM builds source-based trees using flood-and-prune
  - PIM-SM builds core-based trees as well as source-based trees with explicit joins.

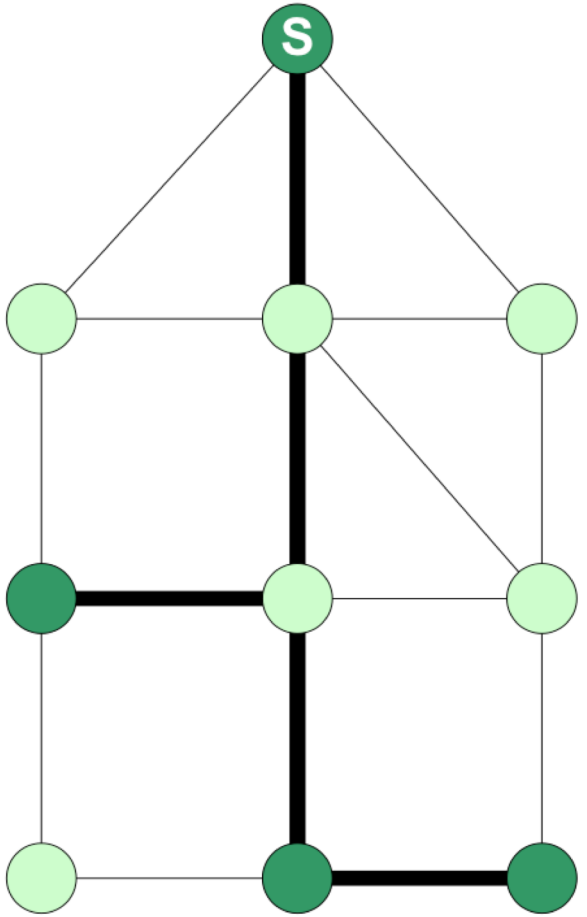
Multicast Routing Solution 1

- **Problem**: Embed a tree such that all multicast group members are connected by the tree
- **Solution 1: Shortest Path Tree or source-based tree**
  Build a tree that minimizes the path cost from the source to each receiver
  - *Good tree if there is a single sender*
  - *If there are multiple senders, need one tree per sender*
  - *Easy to compute*



Multicast Routing Solution 2

- **Problem**: Embed a tree such that all multicast group members are connected by the tree
- **Solution 2: Minimum-Cost Tree**
  Build a tree that minimizes the total cost of the edges
  - *Good solution if there are multiple senders*
  - *Very expensive to compute (not practical for more than 30 nodes)*



Multicast Routing in Practice

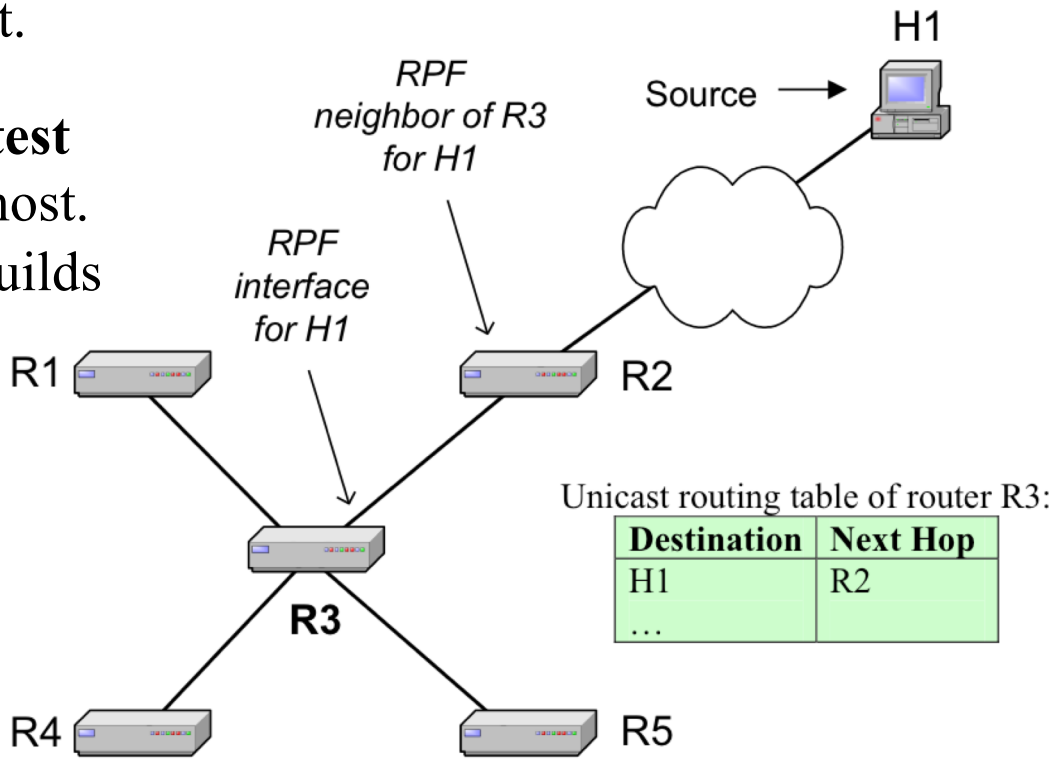
- Routing Protocols implement one of two approaches:
  - **Source Based Tree:**
    - Essentially implements Solution 1.
    - Builds one shortest path tree for each sender
    - Tree is built from receiver to the sender

      → reverse shortest path / reverse path forwarding
  - **Core-based Tree:**
    - Build a single distribution tree that is shared by all senders
    - Does not use Solution 2 (because it is too expensive)
    - Selects one router as a "core" (also called "rendezvous point")
    - All receivers build a shortest path to the core

      → reverse shortest path /reverse path forwarding

Reverse Path Forwarding (RPF)

- RPF builds a shortest path tree in a distributed fashion by taking advantage of the unicast routing tables.
- **Main concept:** Given the address of the root of the tree (e.g., the sending host), a router selects as its upstream neighbor in the tree, which is the next-hop neighbor for forwarding unicast packets to the root.
- This concept leads to a **reverse shortest path** from any router to the sending host. The union of reverse shortest paths builds a **reverse shortest path tree**.



Unicast routing table of router R3:

Destination	Next Hop
H1	R2
…	

Multicast Routing table

- Routing table entries for source-based trees and for core-based trees are different
  - **Source-based tree**: (Source, Group) or (S, G) entry.
  - **Core-based tree:** (*, G) entry.

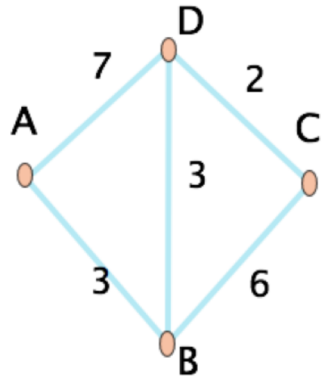
Source IP address	Multicast group	Incoming interface (RPF interface)	Outgoing interface list
S1	G1	I1	I2, I3
*	G2	I2	I1, I3

Recap of Unicast Routing Protocols

- Depending on the routing protocol, different factors may be taken into account, e.g. number of hops, latency, bandwidth, etc.
- Shortest path routing aims to find a path that has the least amount of cost.
- There are two types algorithms to update the routing tables.

Category	Link State	Distance Vector
Algorithm	Dijkstra	Bellman-Ford
Example	OSPF	RIP

Distance Vector Example



(1) First exchange for all nodes to find best 1-hop routes

To	A says	B says	C says	D says
A	0	∞	∞	∞
B	∞	0	∞	∞
C	∞	∞	0	∞
D	∞	∞	∞	0

→

A learns Cost	A learns Next	B learns Cost	B learns Next	C learns Cost	C learns Next	D learns Cost	D learns Next
0	--	3	A	∞	--	7	A
3	B	0	--	6	B	3	B
∞	--	6	C	0	--	2	C
7	D	3	D	2	D	0	--

(2) Second exchange for all nodes to find best 2-hop routes

To	A says	B says	C says	D says
A	0	3	∞	7
B	3	0	6	3
C	∞	6	0	2
D	7	3	2	0

→

A learns Cost	A learns Next	B learns Cost	B learns Next	C learns Cost	C learns Next	D learns Cost	D learns Next
0	--	3	A	9	B	6	B
3	B	0	--	5	D	3	B
9	D	5	D	0	--	2	C
6	B	3	D	2	D	0	--

(3) Third exchange for all nodes to find best 3-hop routes

To	A says	B says	C says	D says
A	0	3	9	6
B	3	0	5	3
C	9	5	0	2
D	6	3	2	0

→

A learns Cost	A learns Next	B learns Cost	B learns Next	C learns Cost	C learns Next	D learns Cost	D learns Next
0	--	3	A	8	D	6	B
3	B	0	--	5	D	3	B
8	B	5	D	0	--	2	C
6	B	3	D	2	D	0	--

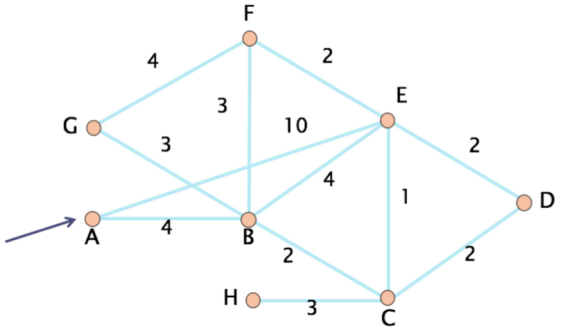
(4) Fourth and subsequent exchanges; converged

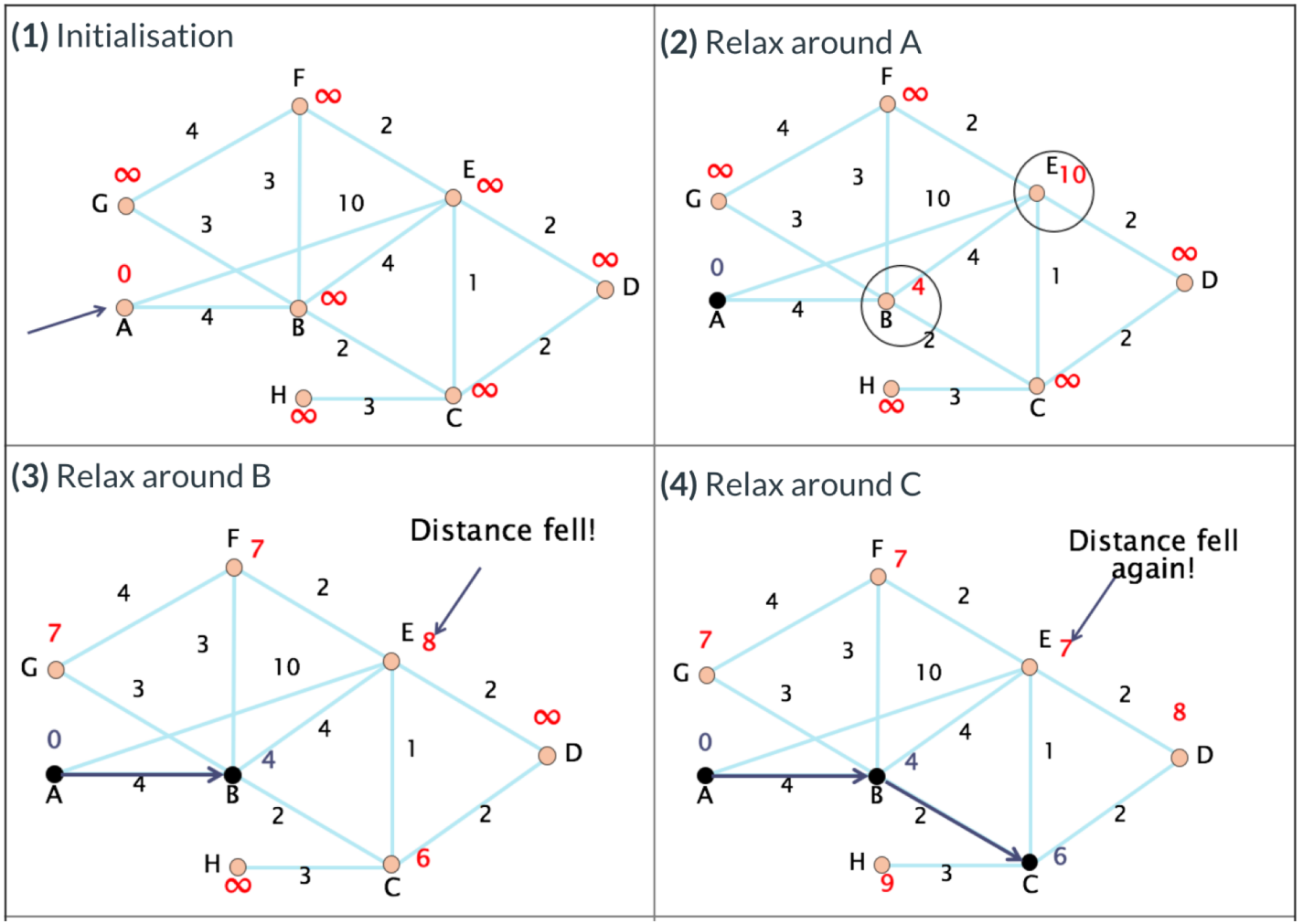
To	A says	B says	C says	D says
A	0	3	8	6
B	3	0	5	3
C	8	5	0	2
D	6	3	2	0

→

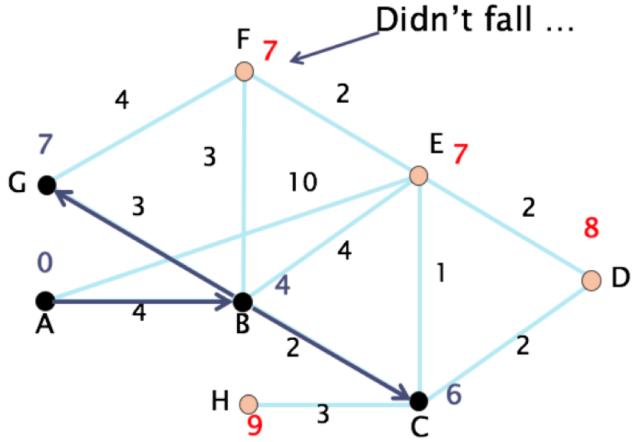
A learns Cost	A learns Next	B learns Cost	B learns Next	C learns Cost	C learns Next	D learns Cost	D learns Next
0	--	3	A	8	D	6	B
3	B	0	--	5	D	3	B
8	B	5	D	0	--	2	C
6	B	3	D	2	D	0	--

Link State Example

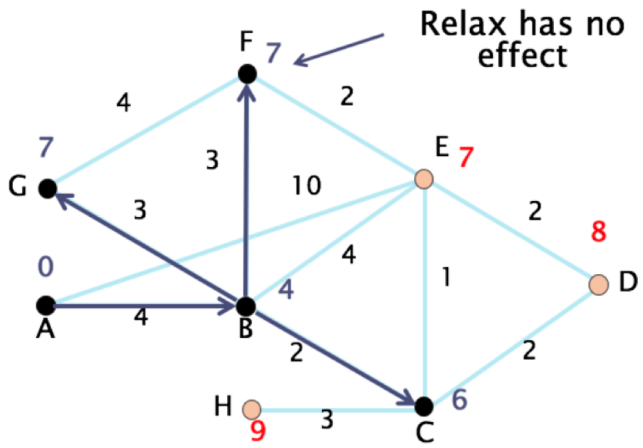




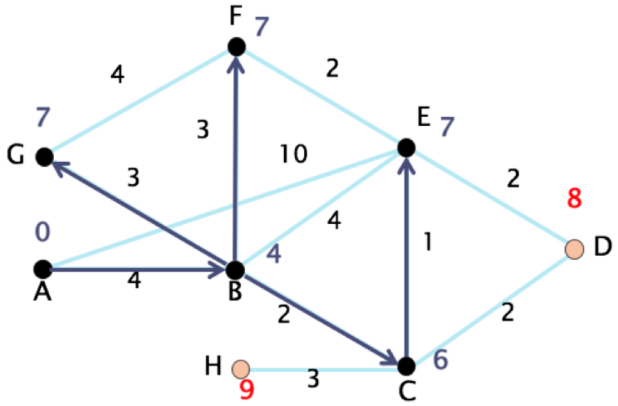
(5) Relax around G (say)



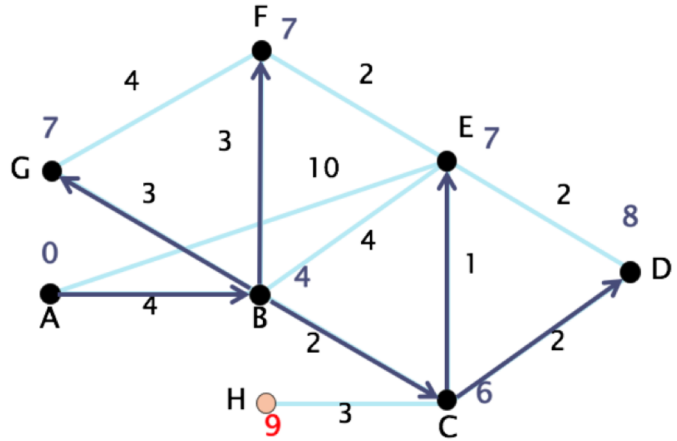
(6) Relax around F (say)



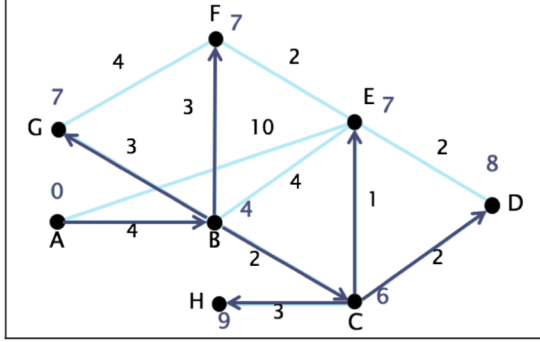
(7) Relax around E



(8) Relax around D



(9) Finally, H... done

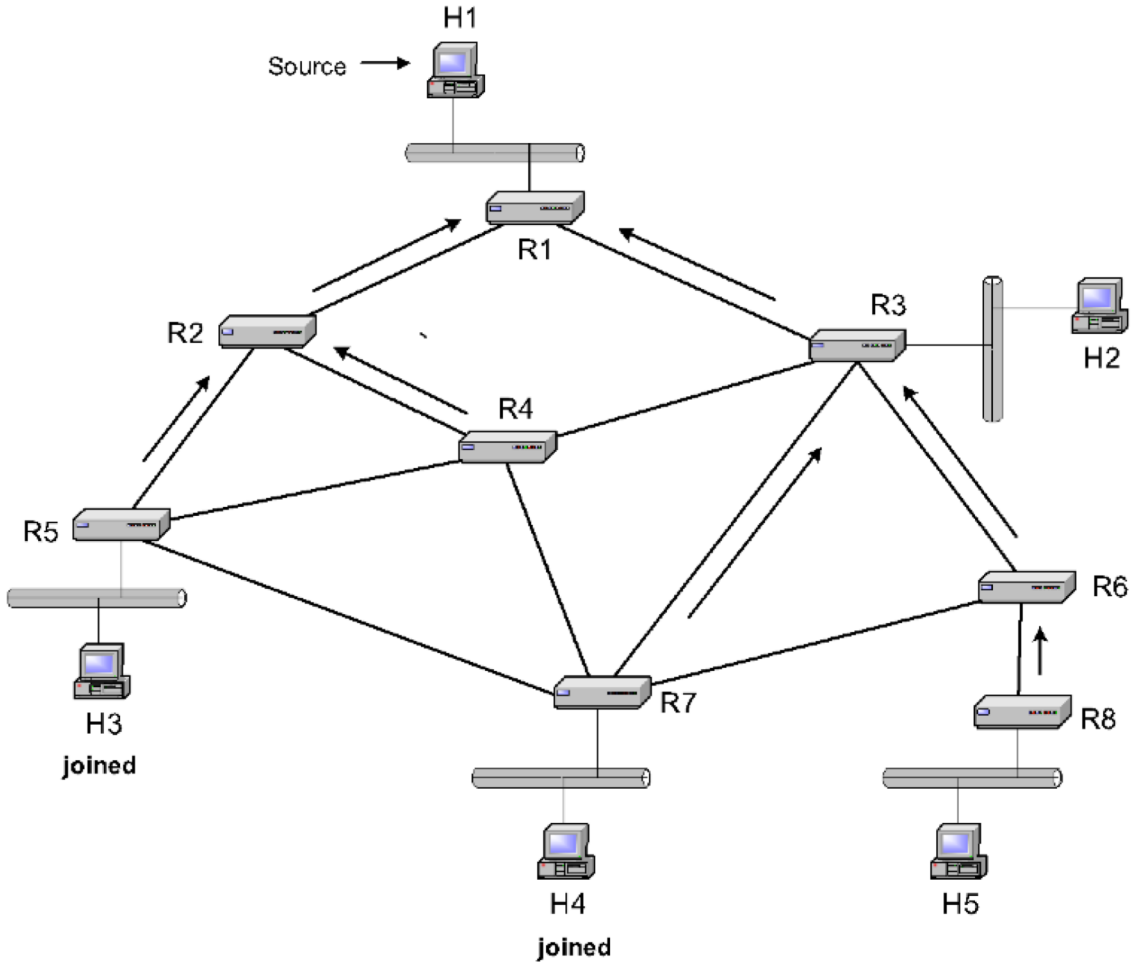


Topics

- Multicast Routing Solutions
- Source Based Trees
- Core Based Trees

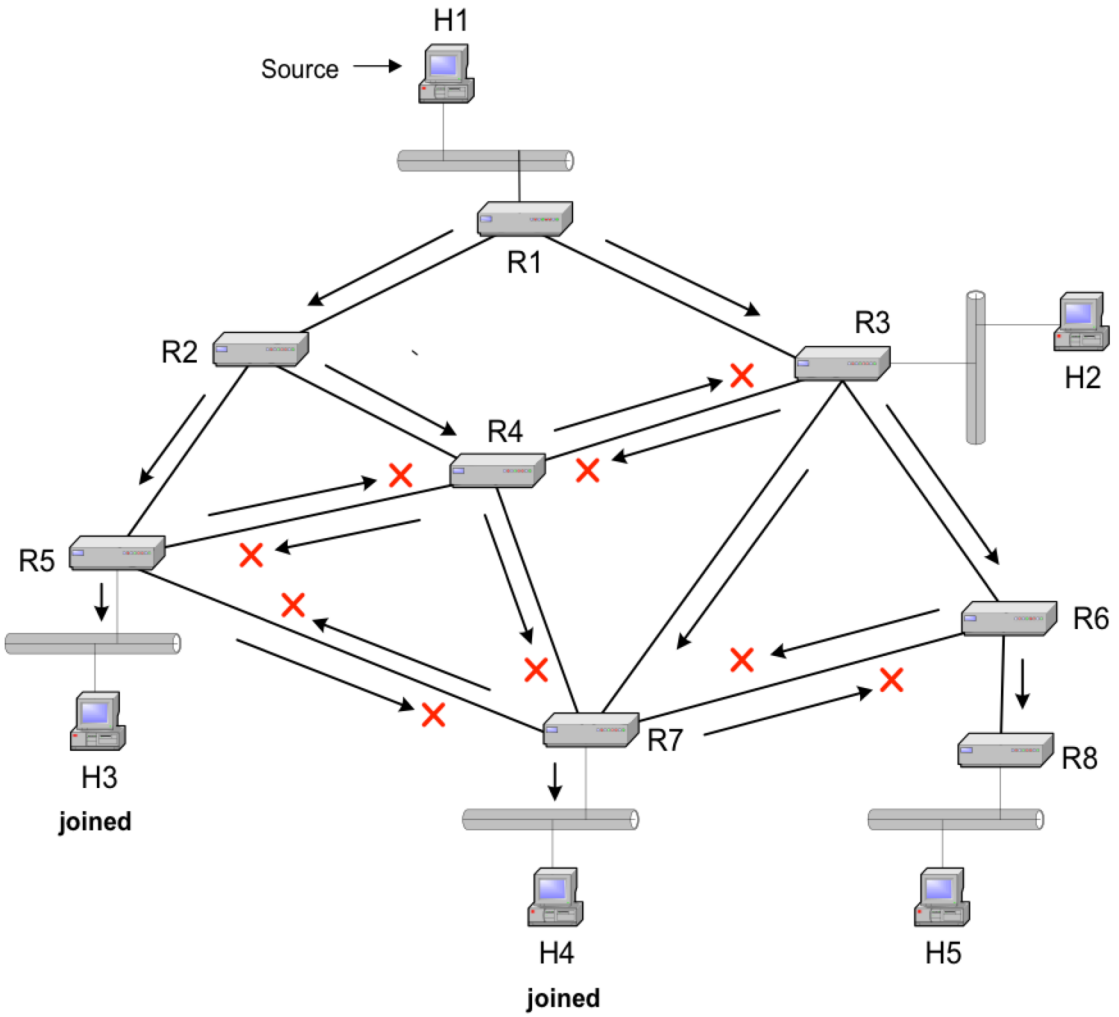
Source-Based Tree

- Set routing tables according to RPF
- Flood-and-Prune



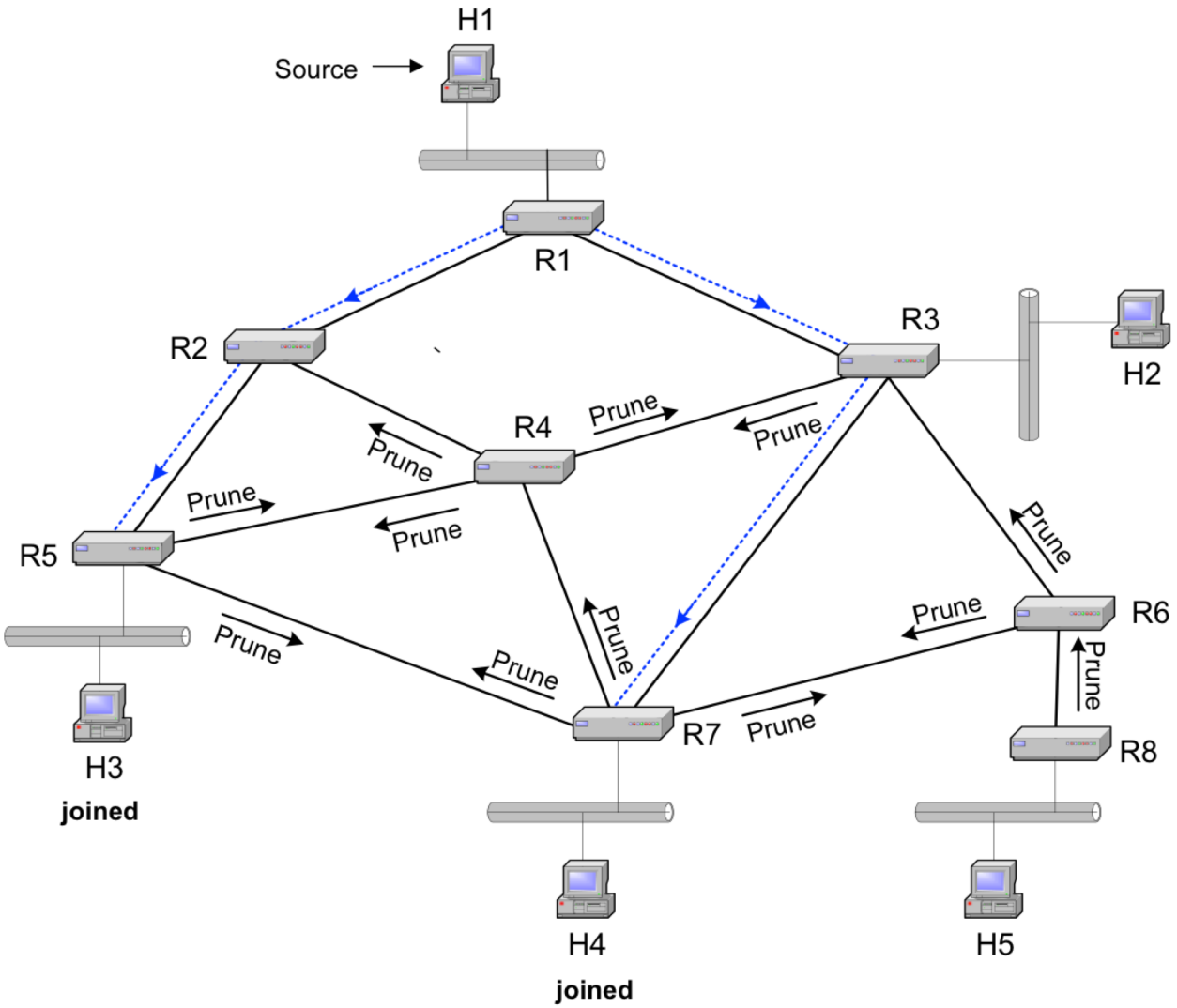
Flooding

- Forward packets on all non-RPF interfaces
- Receiver drops packets not received on RPF interface



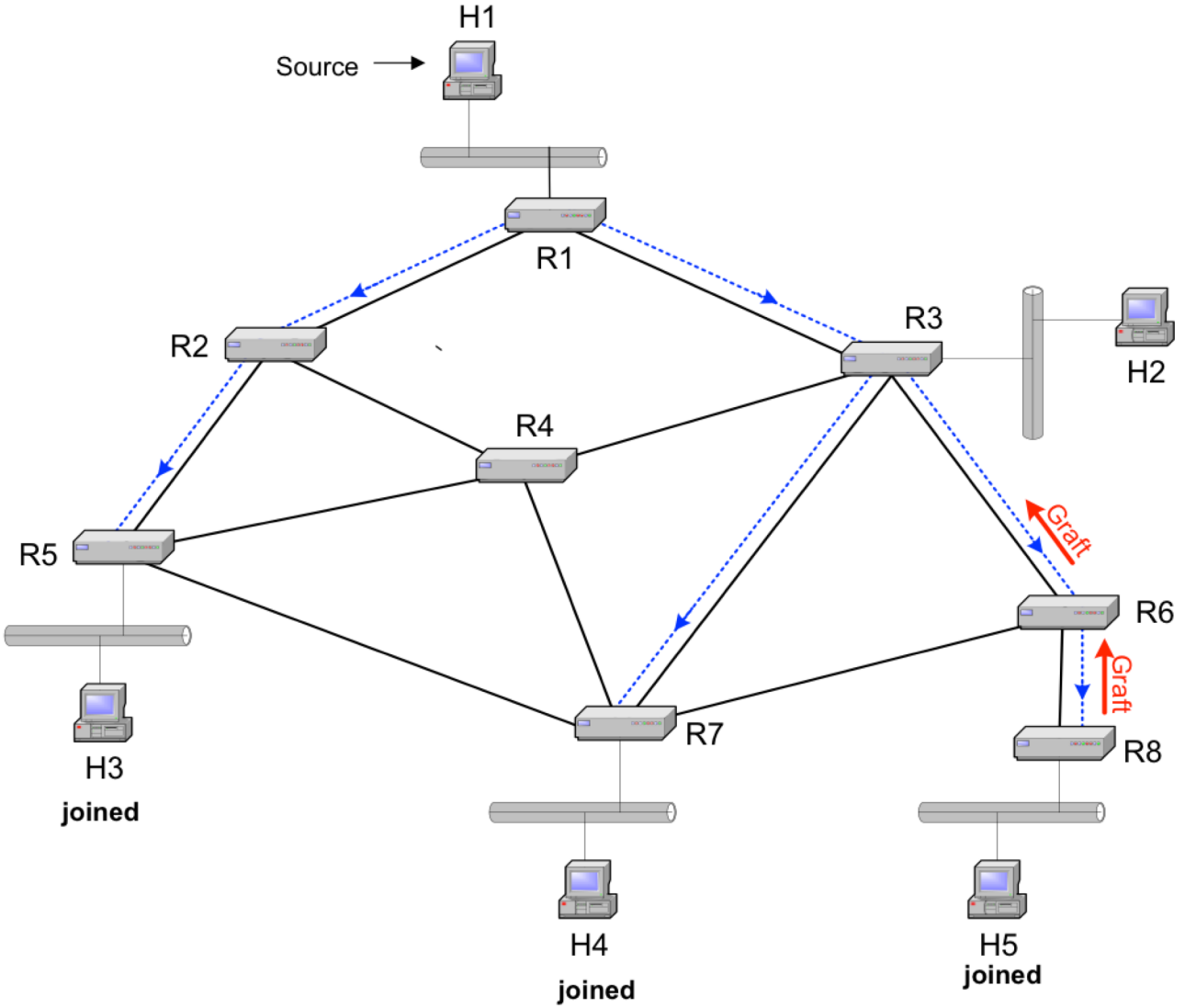
Pruning

- Prune message temporarily disables a routing table entry
- **Effect**: Removes a link from the multicast tree
  No multicast messages are sent on a pruned link
- Prune message is sent in response to a multicast packet
- *Question: Why is routing table only temporarily disabled?*



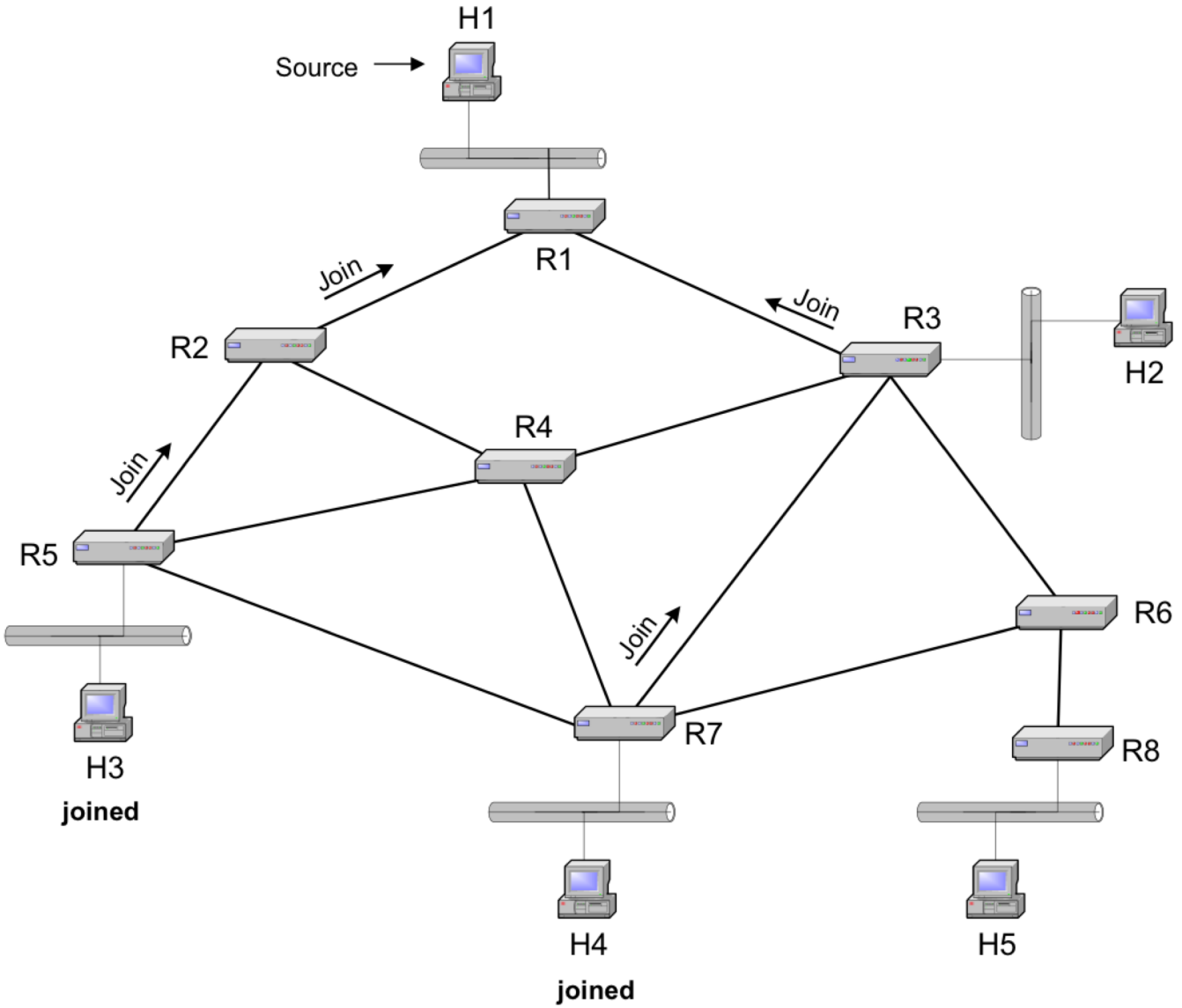
Grafting

- When a receiver joins, one needs to re-activate a pruned routing table entry
- Sending a *Graft* message disables prune, and re-activates routing table entry.



Alternative: Explicit Join

- This only works if the receiver knows the source
- Receiver sends a *Join* message to RPF neighbor
- Join message creates (S,G) routing table entry
- *Join* message is passed on

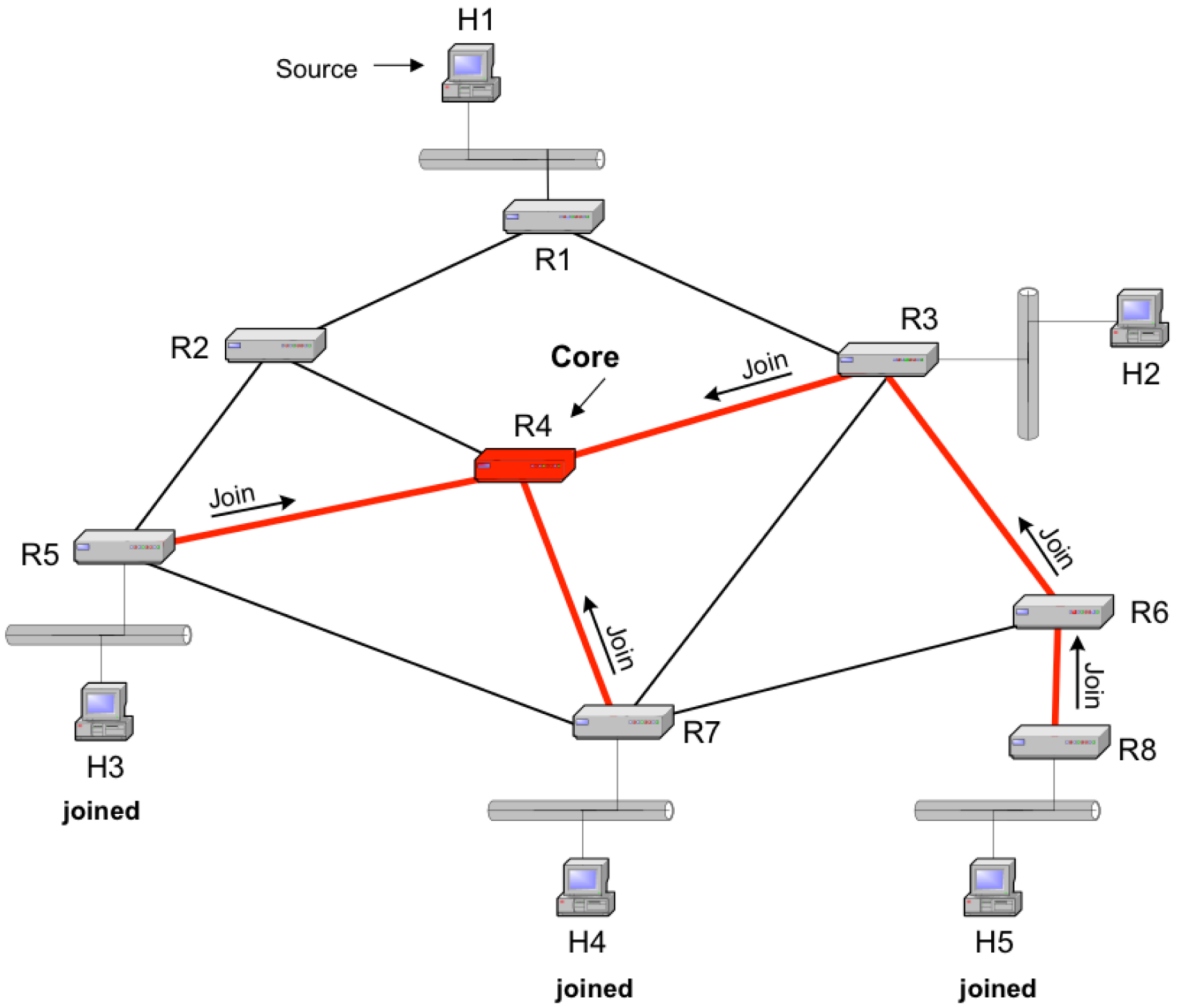


Topics

- Multicast Routing Solutions
- Source Based Trees
- Core Based Trees

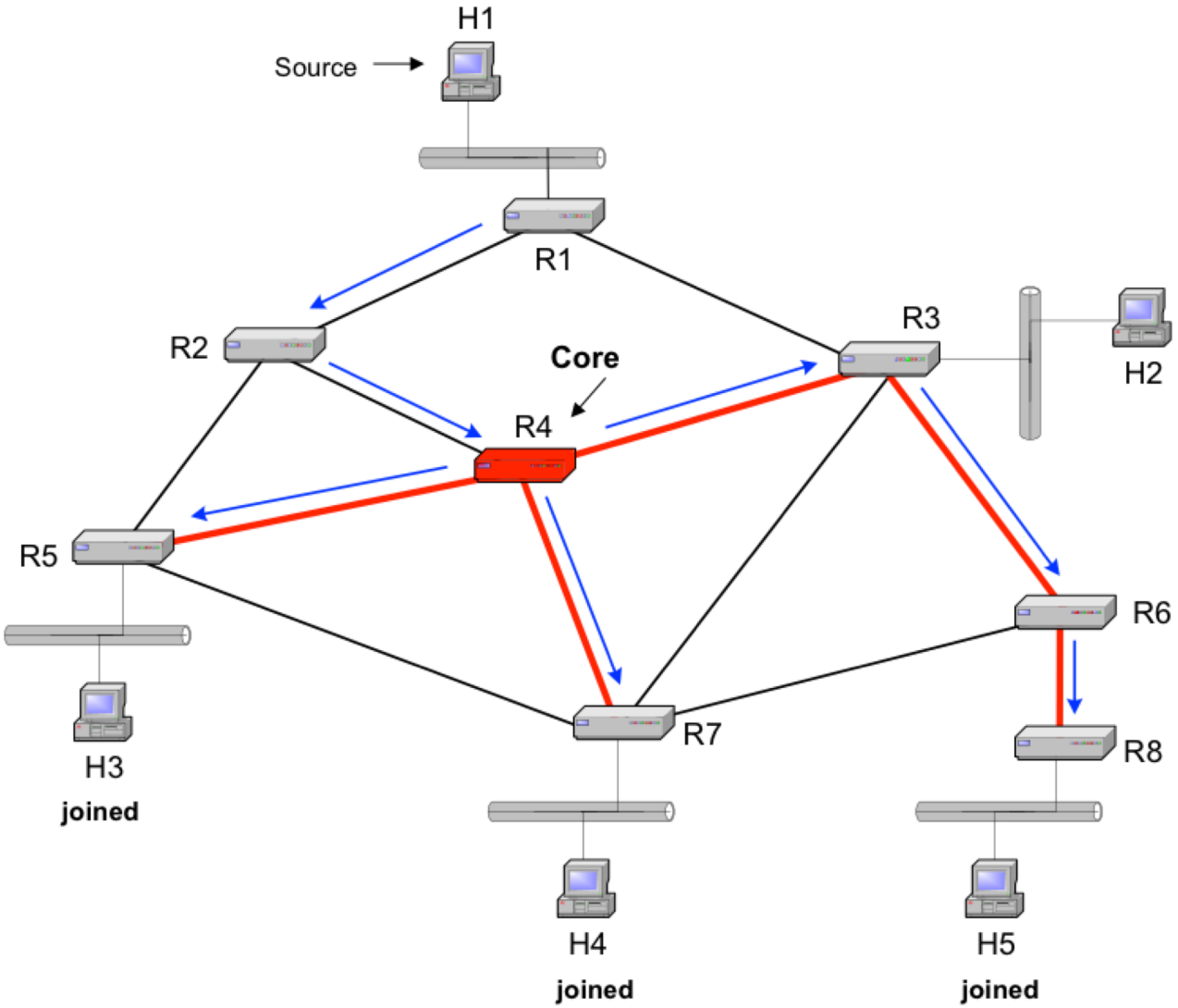
Core-Based Tree

- Core is called rendezvous-point **(RP)**
- Receivers know RP (statically configured or dynamically elected)
- Receiver sends a *Join* message to RPF neighbour with respect to core
- *Join* message creates (*, G) routing table entry



Core-Based Tree

- Source sends data to the core
- Core forwards data according to routing table entry



Questions?

- **Email:** x.che@herts.ac.uk
- **Office:** LB218
- **Tel:** 01707 286206