DIPARTIMENTO
DI INFORMATICA
Via Salaria 113 - 00198, Roma

SAPIENZA
UNIVERSITÀ DI ROMA

Chapter 4
Network Layer

Reti di Elaboratori
Corso di Laurea in Informatica
Università degli Studi di Roma "La Sapienza"
Canale A-L

Prof.ssa Chiara Petrioli

Parte di queste slide sono state prese dal materiale associato al libro *Computer Networking: A Top Down Approach* , 5th edition.

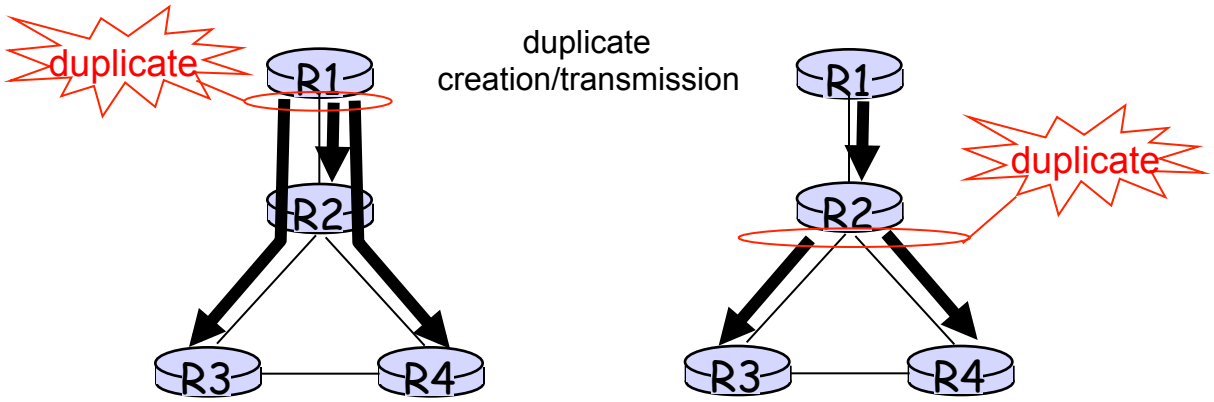
Thanks also to Antonio Capone, Politecnico di Milano, Giuseppe Bianchi and Francesco LoPresti, Un. di Roma Tor Vergata

Chapter 4: Network Layer

- 4. 1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What's inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing

Broadcast Routing

- deliver packets from source to all other nodes
- source duplication is inefficient:

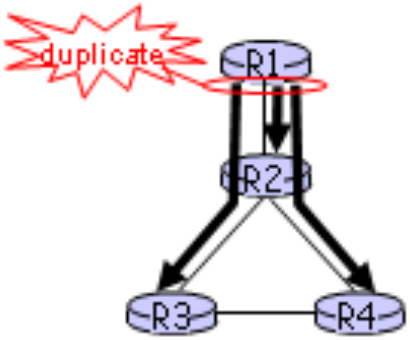


source duplication

in-network duplication

- source duplication: how does source determine recipient addresses?

Unicast ad N vie

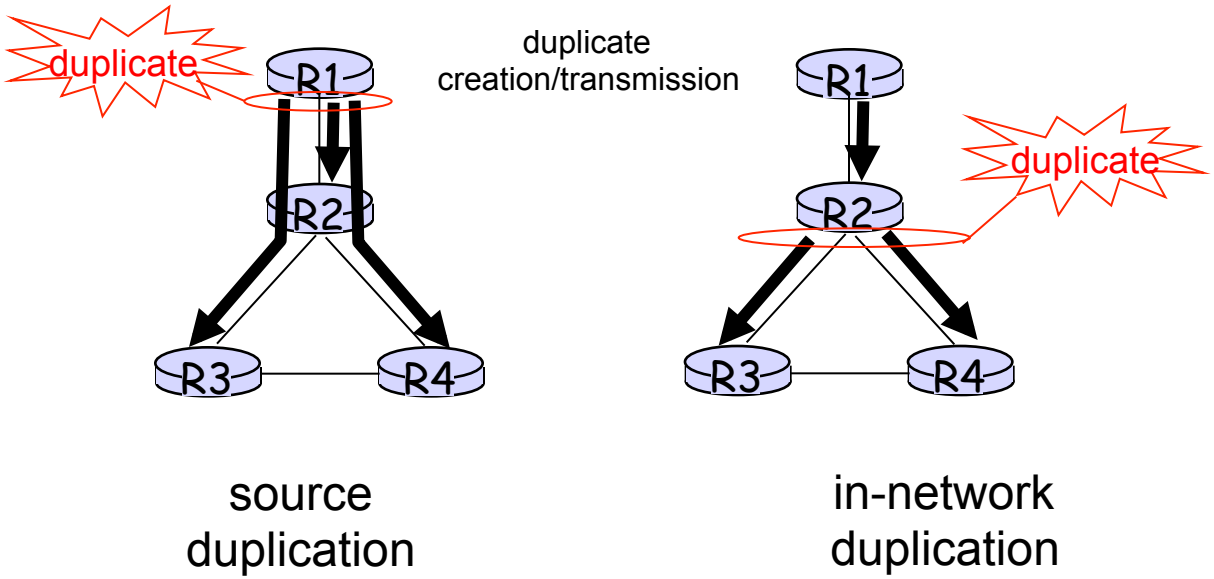


source duplication

- Inefficiente
  - Un singolo collegamento attraversato da N copie del messaggio se il nodo origine è connesso al resto della rete tramite un unico collegamento
- Indirizzi di tutte le destinazioni devono essere noti al mittente
  - altri meccanismi protocollari sono richiesti
- Broadcast può essere usato per inoltrare informazioni di topologia in una situazione in cui le rotte non sono ancora note
  - es. OSPF

Broadcast Routing

- deliver packets from source to all other nodes
- source duplication is inefficient:



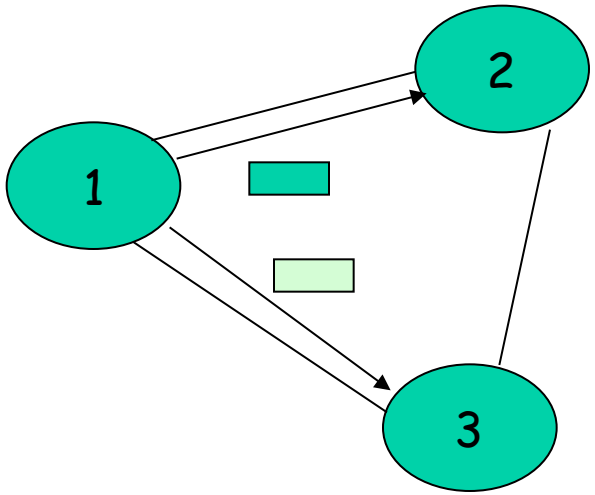
source duplication

in-network duplication

- source duplication: how does source determine recipient addresses?

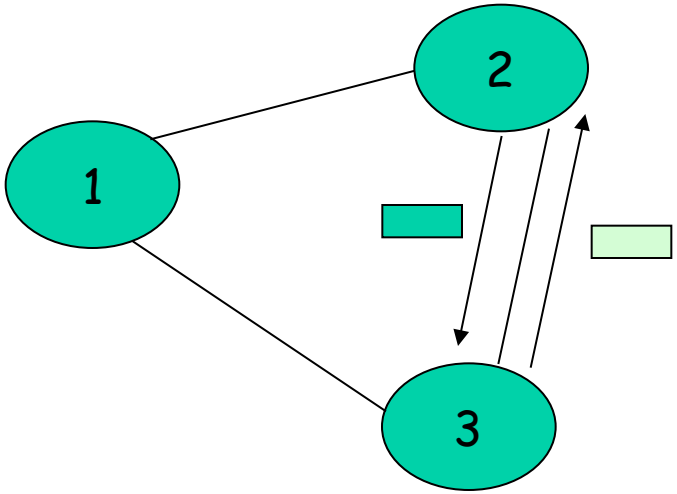
In-network duplication

- flooding: when node receives brdcst pckt, sends copy to all neighbors EXCEPT the one from which the pckt was received
  - Problems: cycles & broadcast storm



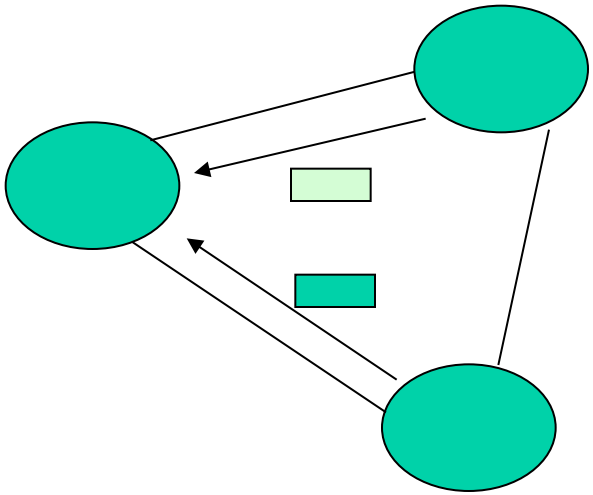
In-network duplication

- flooding: when node receives brdcst pckt, sends copy to all neighbors
  - Problems: cycles & broadcast storm



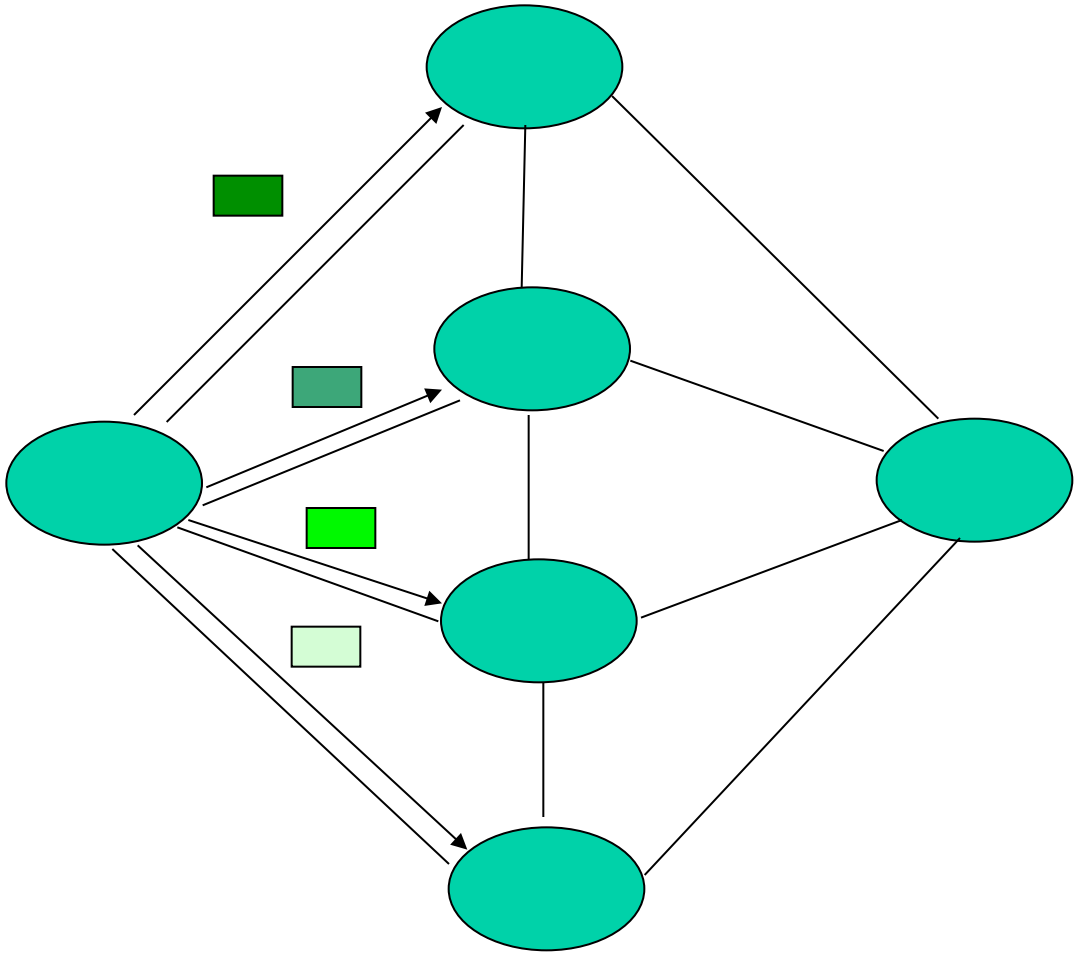
In-network duplication

- flooding: when node receives brdcst pckt, sends copy to all neighbors
  - Problems: cycles & broadcast storm

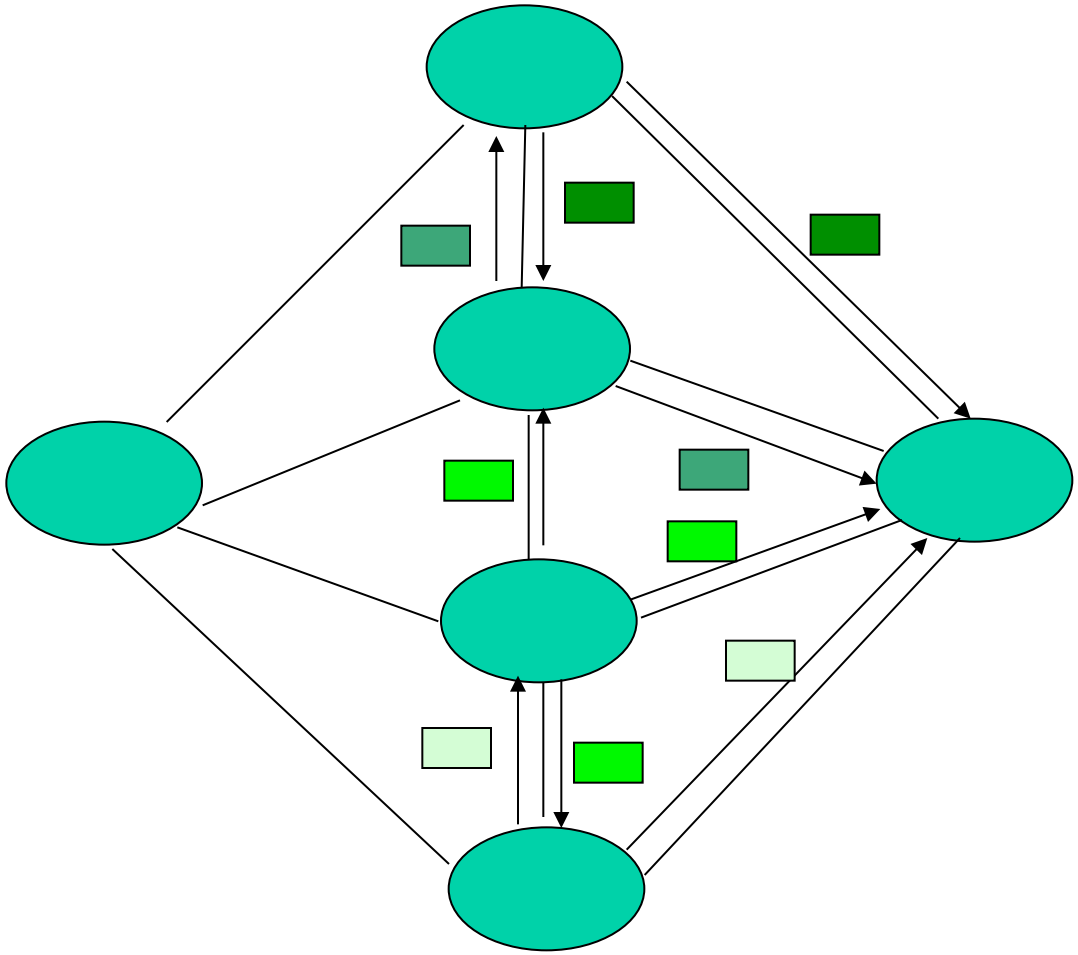


E ricominciamo come nella prima situazione
Bisogna saper distinguere tra quando mandiamo un nuovo messaggio e quando stiamo ritrasmettendo qualcosa che abbiamo già visto
→ Sequence numbers!

Broacast storm

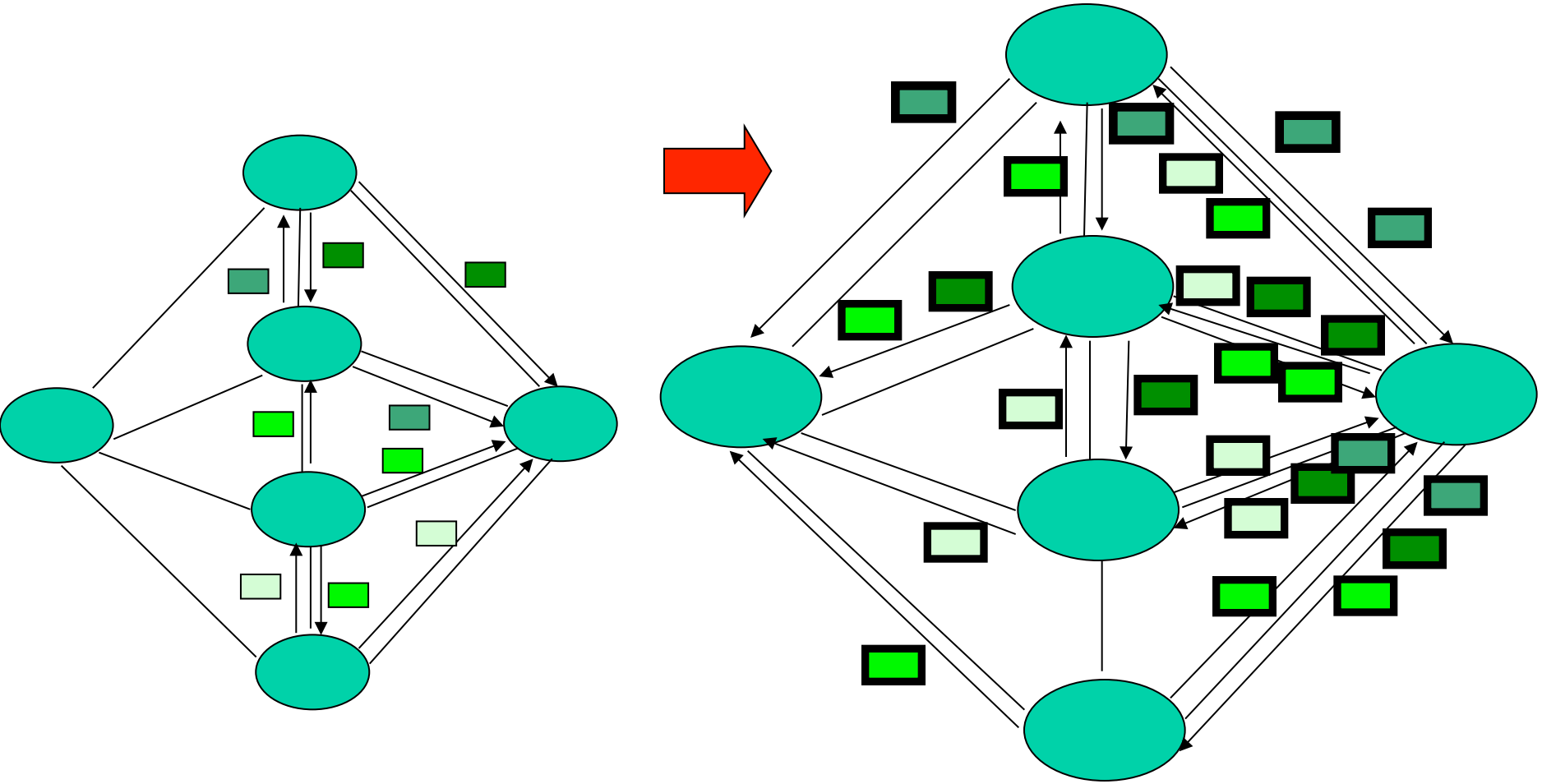


Broacast storm



Broacast storm

Il numero di pacchetti in rete cresce significativamente!!



Controlled flooding

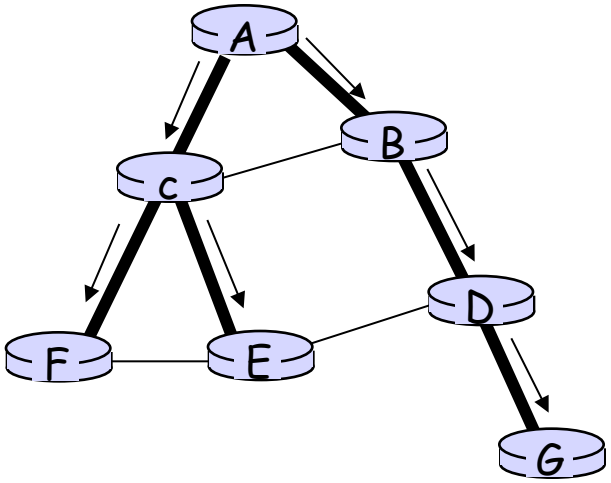
- Il nodo origine pone il proprio indirizzo ed il numero di sequenza nei pacchetti che invia in broadcast
- Ciascun nodo mantiene una lista di ID origine, SEQN per i broadcast ricevuti, trasmesso o inoltrato
- Se riceve un pacchetto broadcast per prima cosa verifica se <ID, SEQN> compare nella lista dei pacchetti già gestiti
  - Se si scarta
  - Altrimenti riinvia su tutte le interfacce tranne quella da cui ha ricevuto

Controlled flooding, altre opzioni

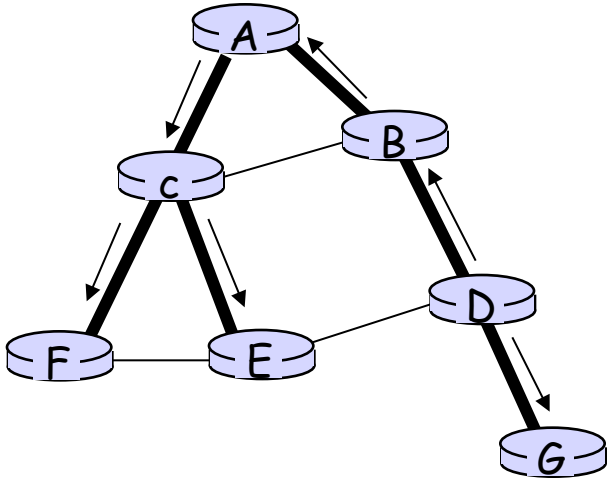
- Reverse path forwarding (RPF): only forward pckt (on all links but the one from which the packet was received) if it arrived on shortest path between node and source

Spanning Tree

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



(a) Broadcast initiated at A



(b) Broadcast initiated at D

Kruskal algorithm

- Kruskal's algorithm:

  Sort the edges of G in increasing order by length

  Keep a subgraph S of G, initially empty
  For each edge e in sorted order

  If the endpoints of e are disconnected in S then add e to S

  Return S

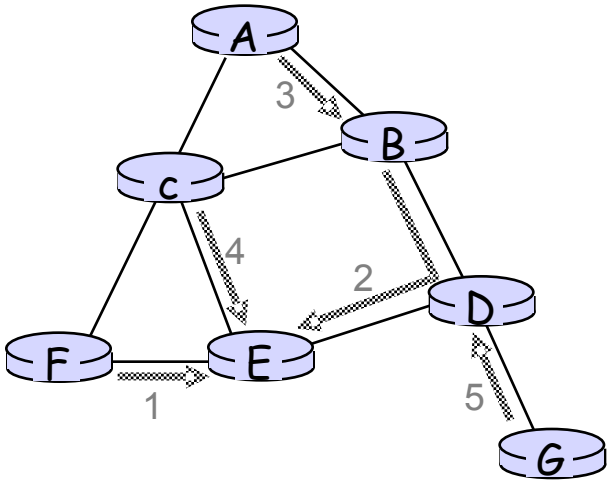
Prim's Algorithm

- Prim's algorithm:

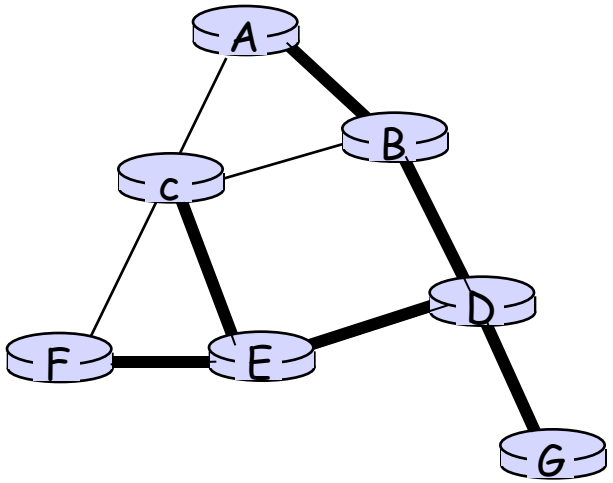
```
let T be a single vertex x
while (T has fewer than n vertices)  {
        Find the smallest edge connecting T to G-T
        Add it to T
}
```

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

Multicasting

- Molte applicazioni richiedono il trasferimento di pacchetti da uno o più mittenti ad un gruppo di destinatari
  - trasferimento di un aggiornamento SW su un gruppo di macchine
  - streaming (audio/video) ad un gruppo di utenti o studenti
  - applicazioni con dati condivisi (lavagna elettronica condivisa da più utenti)
  - aggiornamento di dati (adnamento di borsa)
  - giochi multi-player interattivi
  - …

Indirizzamento Multicast

- L'identificatore che rappresenta un gruppo multicast è un indirizzo IP multicast di classe D
- Come ci si affilia ad un indirizzo multicast? Come vengono gestiti i cambiamenti dinamici (join/remove) nel gruppo?
  - Gestione dinamica del gruppo OLTRE a
  - Algoritmi per la consegna delle informazioni ad un gruppo multicast

IGMP Internet Group Management Protocol

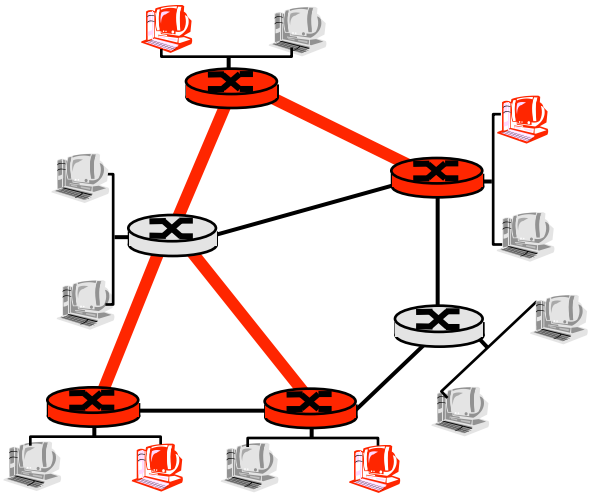
- Messaggi incapsulati in datagrammi IP, con IP protocol number 2
- Mandati con TTL a 1
- Messaggi IGMP
  - Type (8bit) Query (richiesta dal router)/ Membership Report (risposta dagli host)/ Leave group (ma anche possible timeout + mancata risposta alla richiesta del router→ soft state)
- Max Response Time (per rispondere ad una query)
- Checksum
- Group Address (0 se si manda una general query, indirizzo IP del gruppo nel caso di una group specific query con cui si richiede chi sia affiliato a quel gruppo)

IGMP Internet Group Management Protocol

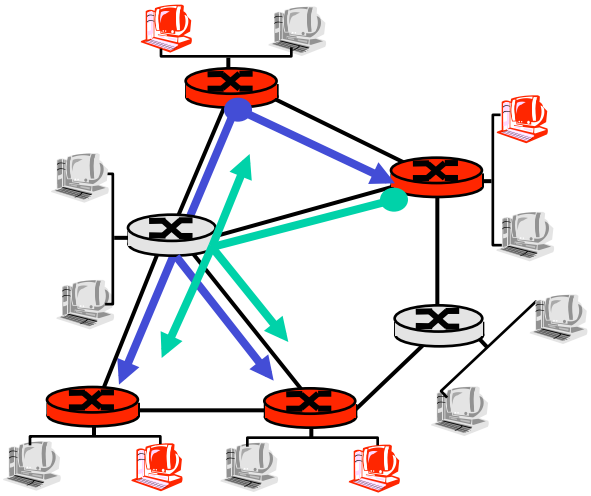
- IGMP consente ad un router di imparare quali gruppi multicast hanno affiliati sulle sottoreti connesse a ciascuna delle loro interfacce
- Un router multicast tiene una lista per ciascuna sottorete dei multicast group (multicast group membership→ almeno un elemento del gruppo fa parte della sottorete) con un timer per membership
  - la membership deve essere aggiornata da report inviati prima della scadenza del timer
  - può essere anche aggiornata tramite messaggi di leave espliciti

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

Approaches for building mcast trees

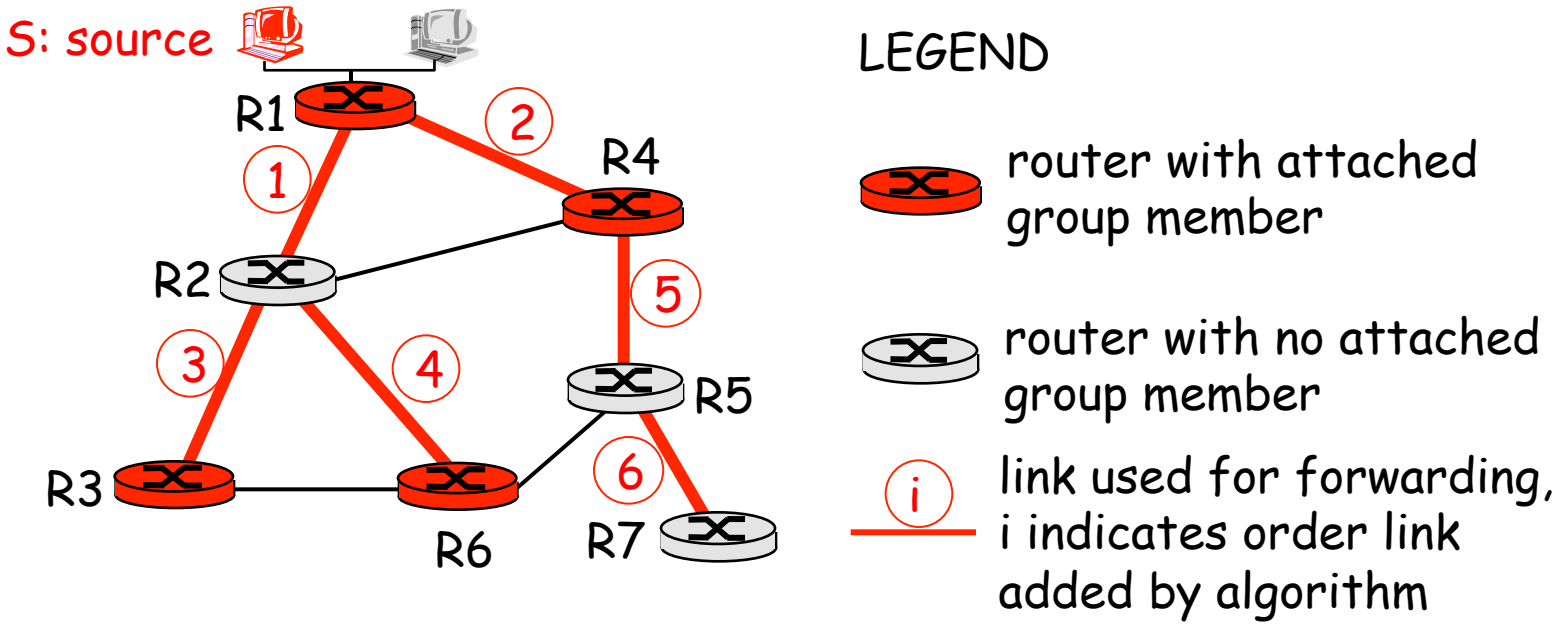
Approaches:

- 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



Reverse Path Forwarding

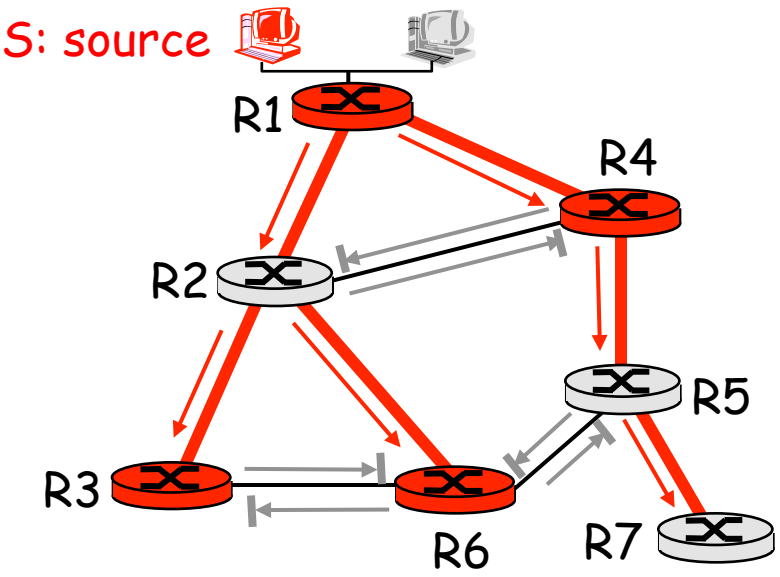
- rely on router's knowledge of unicast shortest path from it to sender
- each router has simple forwarding behavior:

if (mcast datagram received on incoming link on shortest path back to center)

then flood datagram onto all outgoing links

else ignore datagram

Reverse Path Forwarding: example



LEGEND

router with attached group member

router with no attached group member

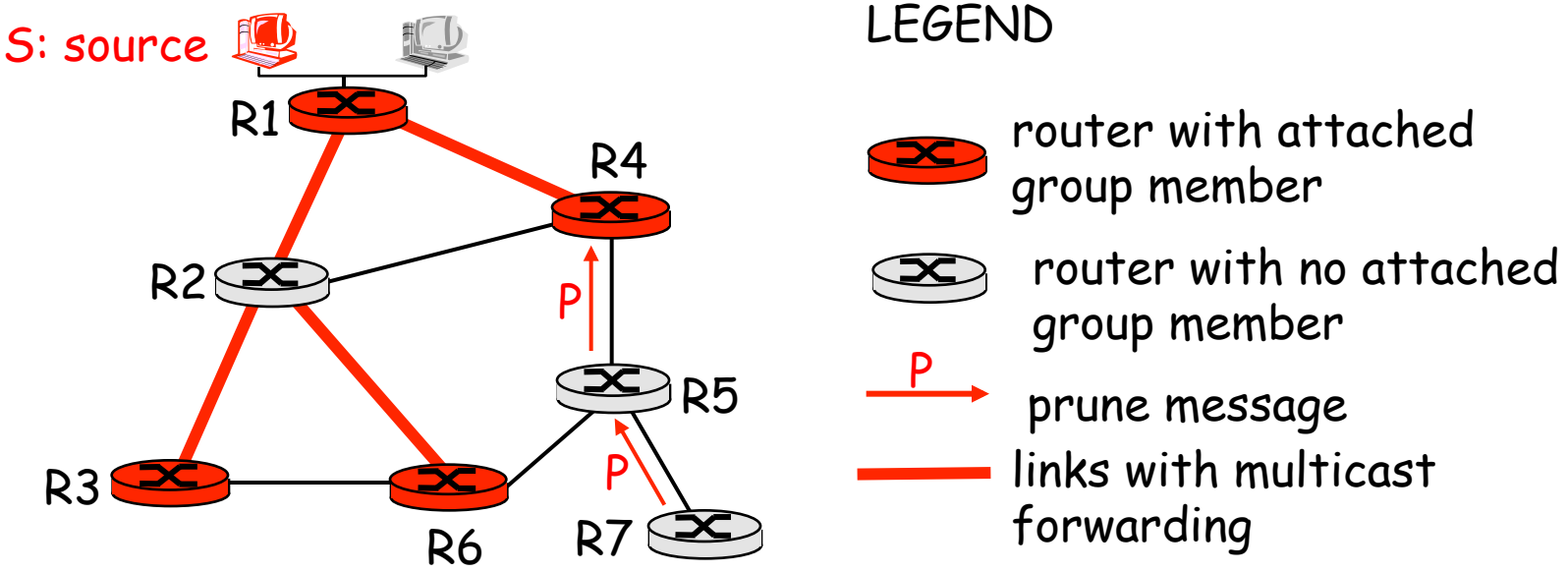
datagram will be forwarded

datagram will not be forwarded

- 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

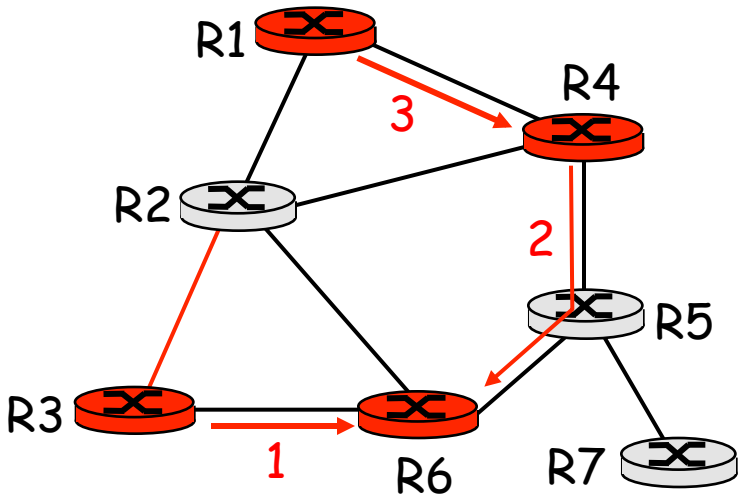
- Steiner Tree: minimum cost tree connecting all routers with attached group members
- problem is NP-complete
- excellent heuristics exists
- not used in practice:
  - computational complexity
  - information about entire network needed
  - monolithic: rerun whenever a router needs to join/leave

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

Center-based trees: an example

Suppose R6 chosen as center:



LEGEND

router with attached group member

router with no attached group member

1 path order in which join messages generated

Internet Multicasting Routing: DVMRP

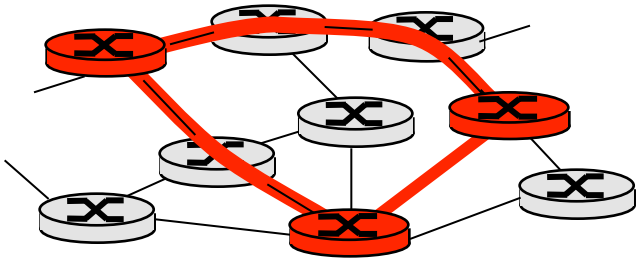
- DVMRP: distance vector multicast routing protocol, RFC1075
- *flood and prune:* reverse path forwarding, source-based tree
  - RPF tree based on DVMRP's own routing tables constructed by communicating DVMRP routers
  - no assumptions about underlying unicast
  - initial datagram to mcast group flooded everywhere via RPF
  - routers not wanting group: send upstream prune msgs

DVMRP: continued…

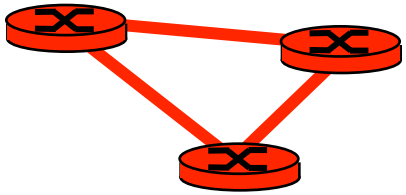
- *soft state:* DVMRP router periodically (1 min.) "forgets" branches are pruned:
  - mcast data again flows down unpruned branch
  - downstream router: reprune or else continue to receive data
- routers can quickly regraft to tree
  - following IGMP join at leaf
- odds and ends
  - commonly implemented in commercial routers
  - Mbone routing done using DVMRP

Tunneling

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



physical topology



logical topology

- 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

Chapter 4: summary

- 4. 1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What's inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing