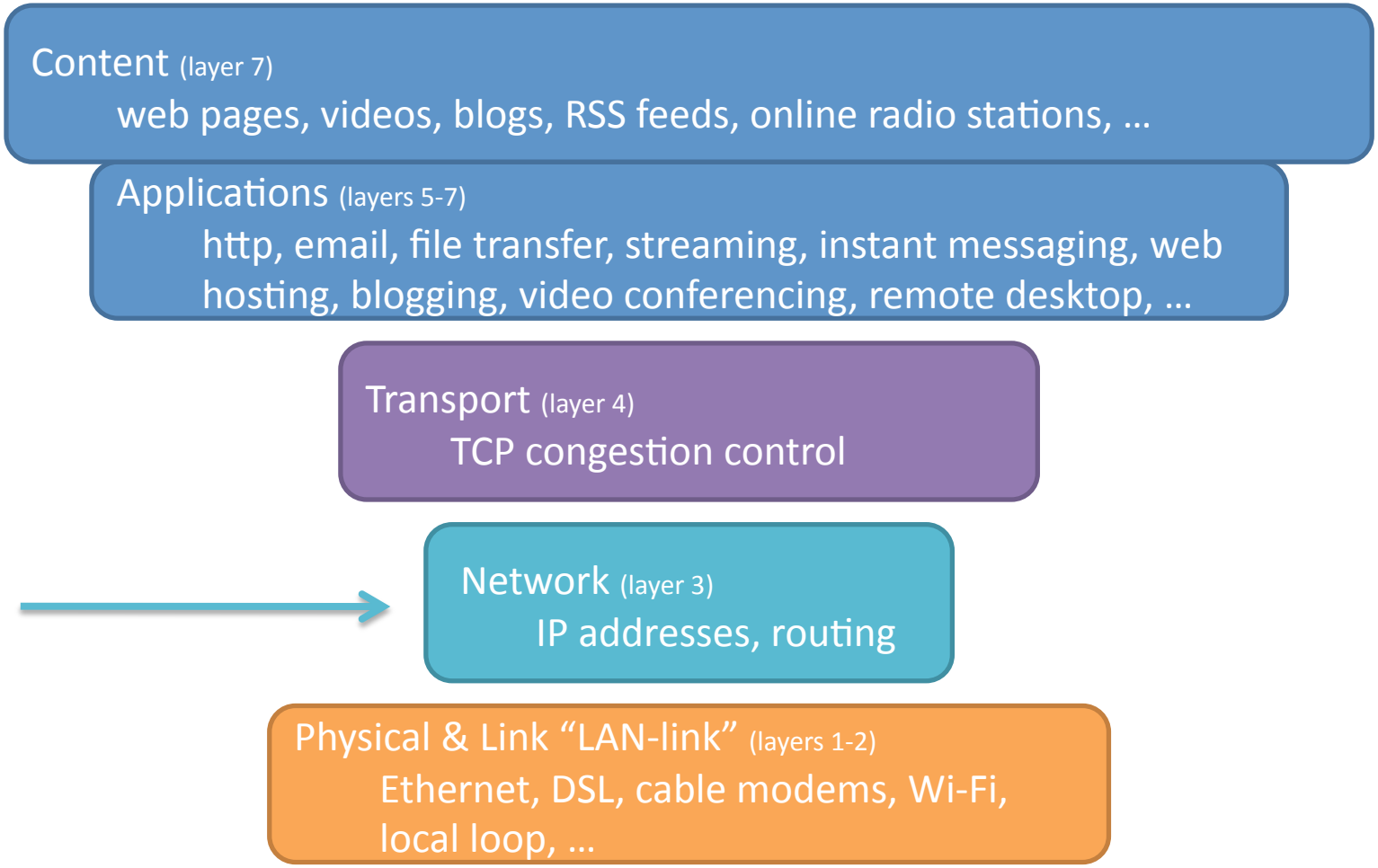
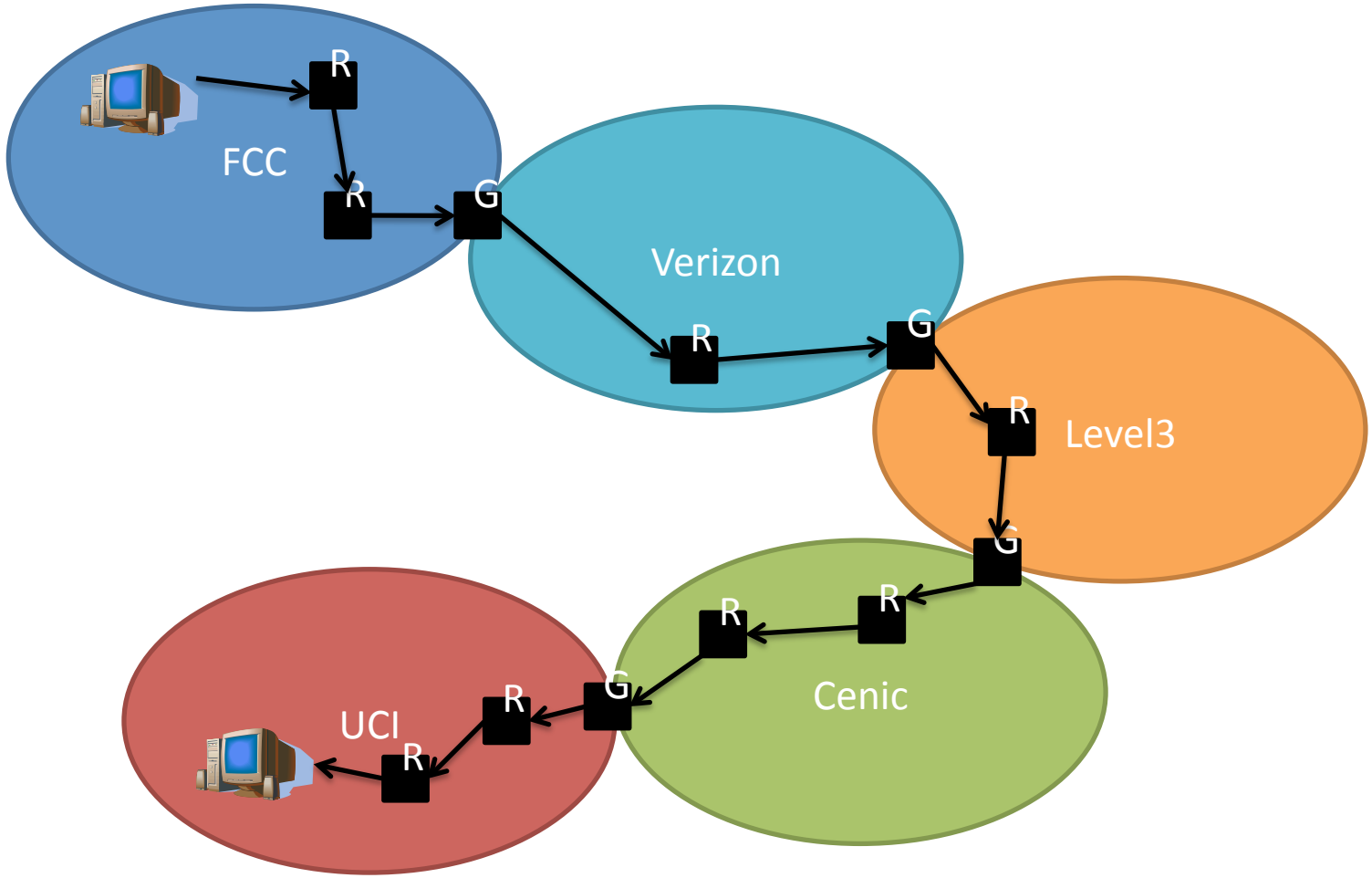
CS 232

Internet Protocol (IP)

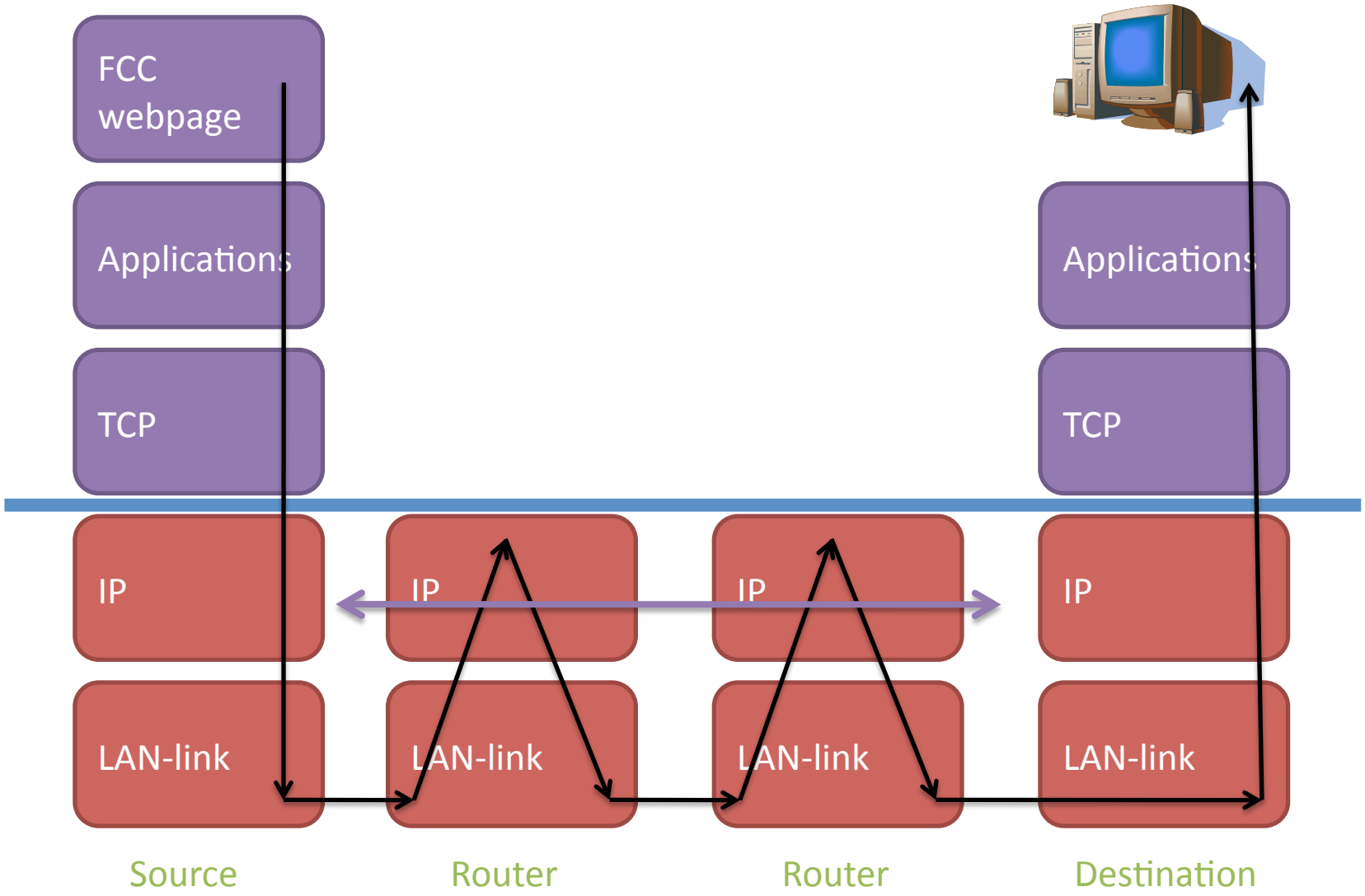
Internet Layering



Peering



IP: router-to-router-to-router



Network layer problems

- Routing
  - Determination of route
  - Implementation of routing
- Congestion
  - Routers may queue many packets, resulting in queuing delay
  - Routers may drop packets, resulting in packet loss

Routing

- Problem: Determination of route
  - Solution: Routers talk to each other and create routing tables
- Problem: Implementation of routing
  - Solution: Path determined at connection set-up
    - *if* want connection-oriented at network layer
  - Solution: Each router forwards packets from input links to output links

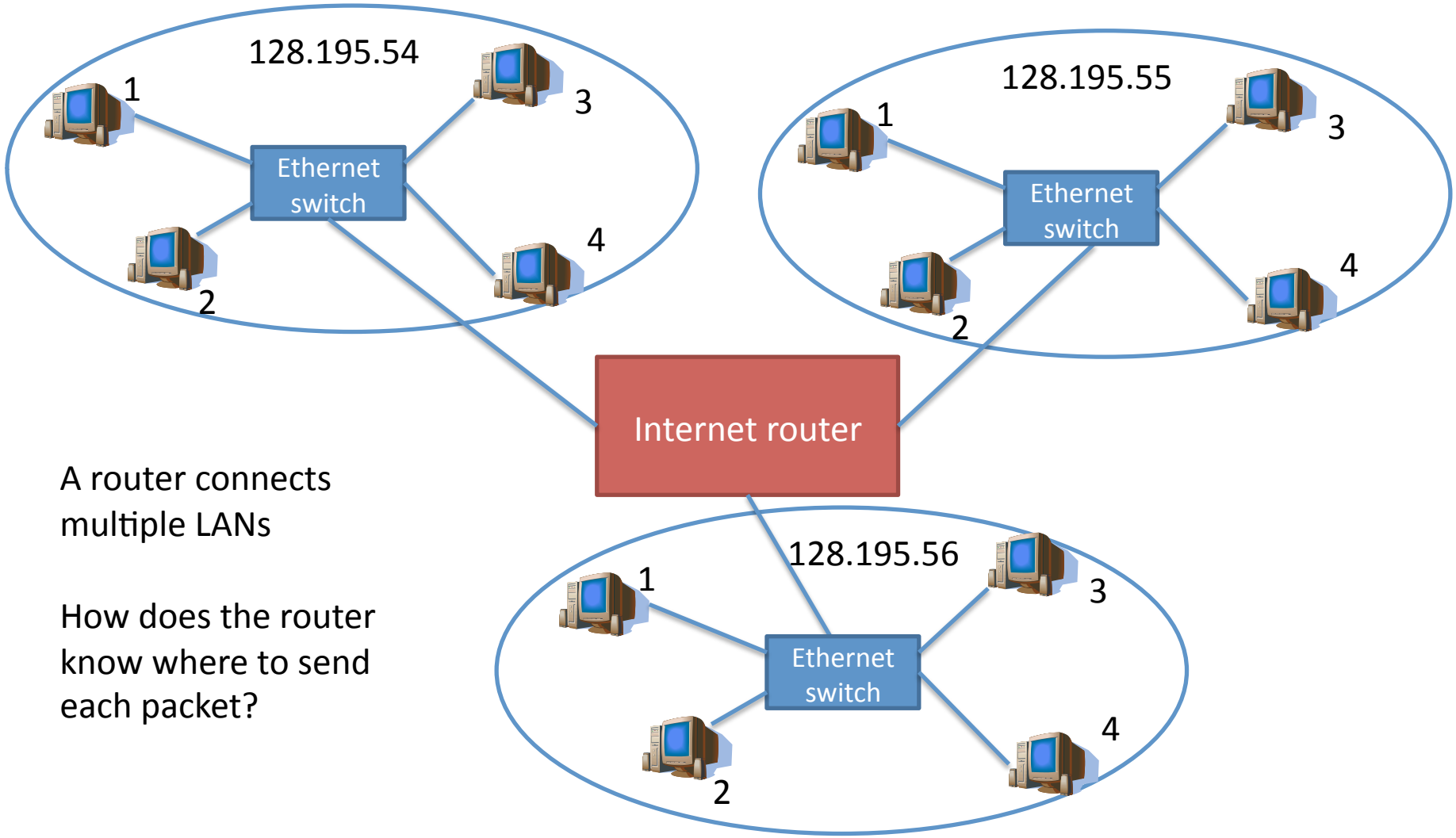
Congestion

- Problem: Routers may queue many packets, resulting in queuing delay
  - Solution: packet scheduling algorithms
  - Solution: reserve resources, *if* connection set-up
- Problem: Routers may drop packets, resulting in packet loss
  - Solution: packet dropping policies
  - Solution: reserve resources, *if* connection set-up

Interface to transport layer

- Source side: receive segments from transport layer
  - Note which transport layer protocol
  - Note application & transport choice: packet scheduling & dropping priorities
- Destination side: hand segment to indicated transport layer protocol
  - if this is a hop that has a transport layer
    - i.e. (usually) only if this is the destination

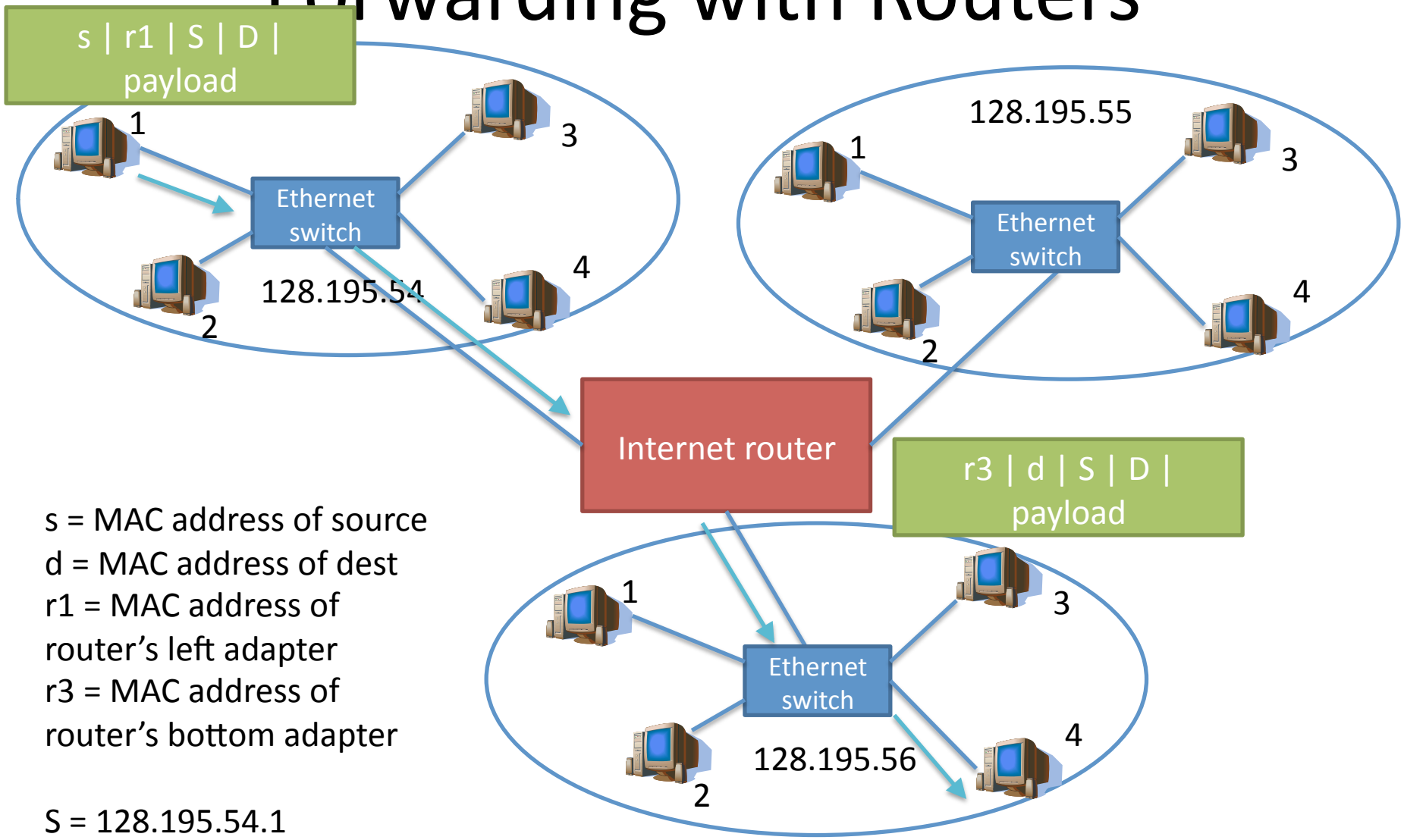
Routers



A router connects multiple LANs

How does the router know where to send each packet?

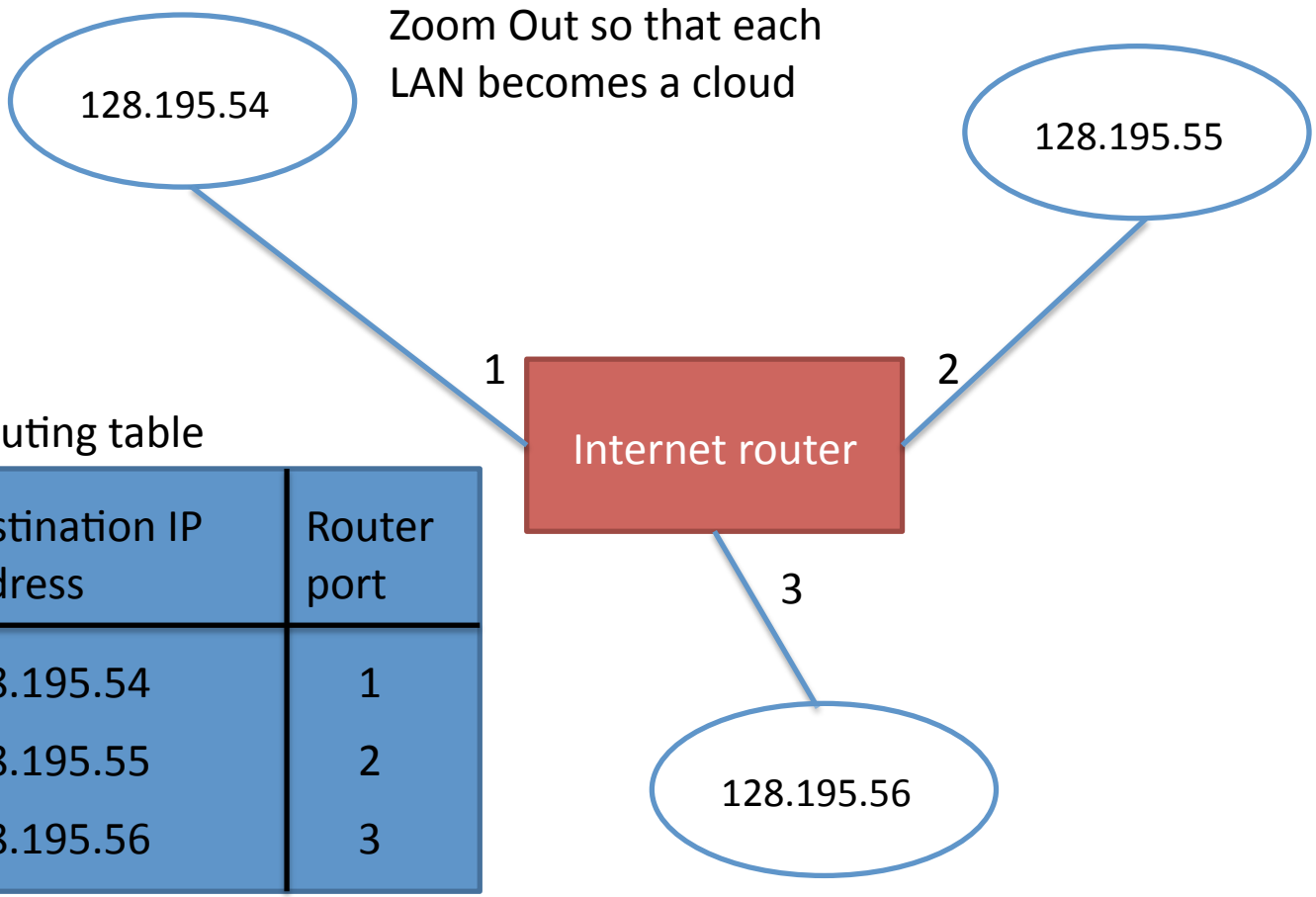
Forwarding with Routers



s = MAC address of source
d = MAC address of dest
r1 = MAC address of router's left adapter
r3 = MAC address of router's bottom adapter

S = 128.195.54.1
D = 128.195.56.4

Routing tables



Routing table

Destination IP address	Router port
128.195.54	1
128.195.55	2
128.195.56	3

Forwarding vs. routing

- Forwarding: move packets from router's input to appropriate router output
- Routing: determine route taken by packets from source to destination

Forwarding tables in datagram packet switched networks

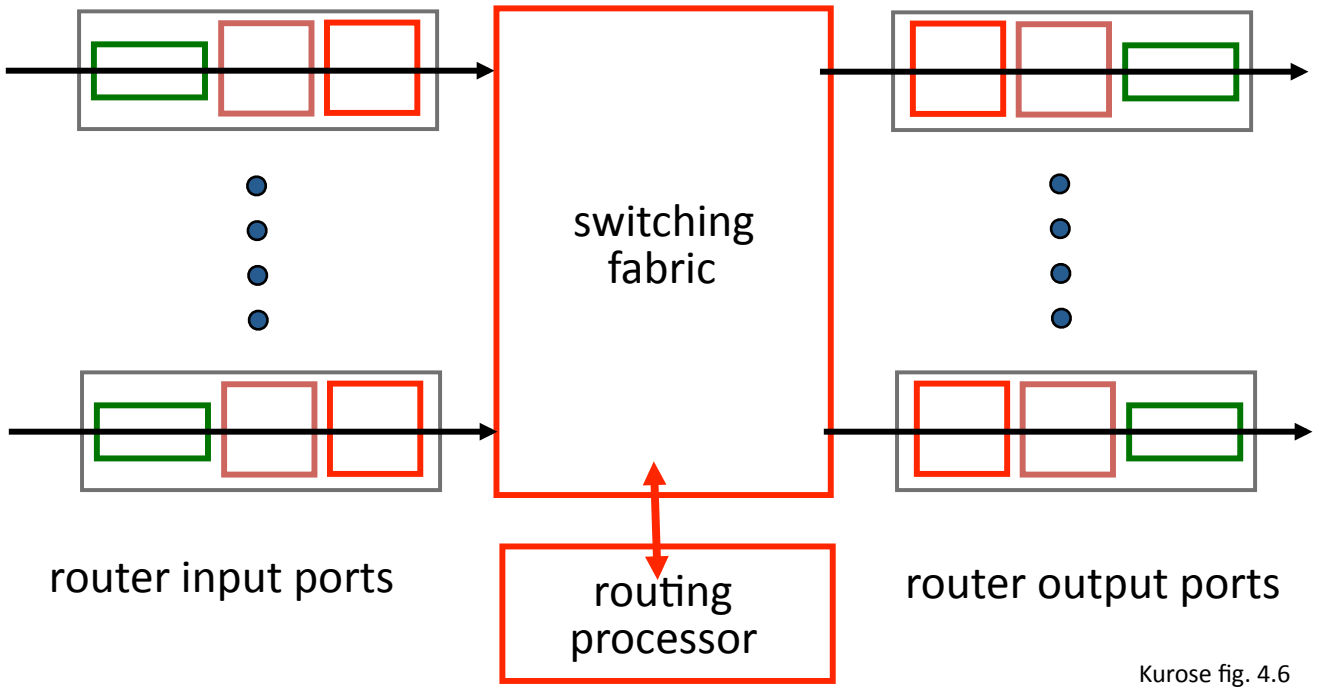
Longest prefix matching

when looking for forwarding table entry for given destination address, use *longest* address prefix that matches destination address.

Destination Address Range	Link interface
11001000 00010111 00010*** *********	0
11001000 00010111 00011000 *********	1
11001000 00010111 00011*** *********	2
otherwise	3

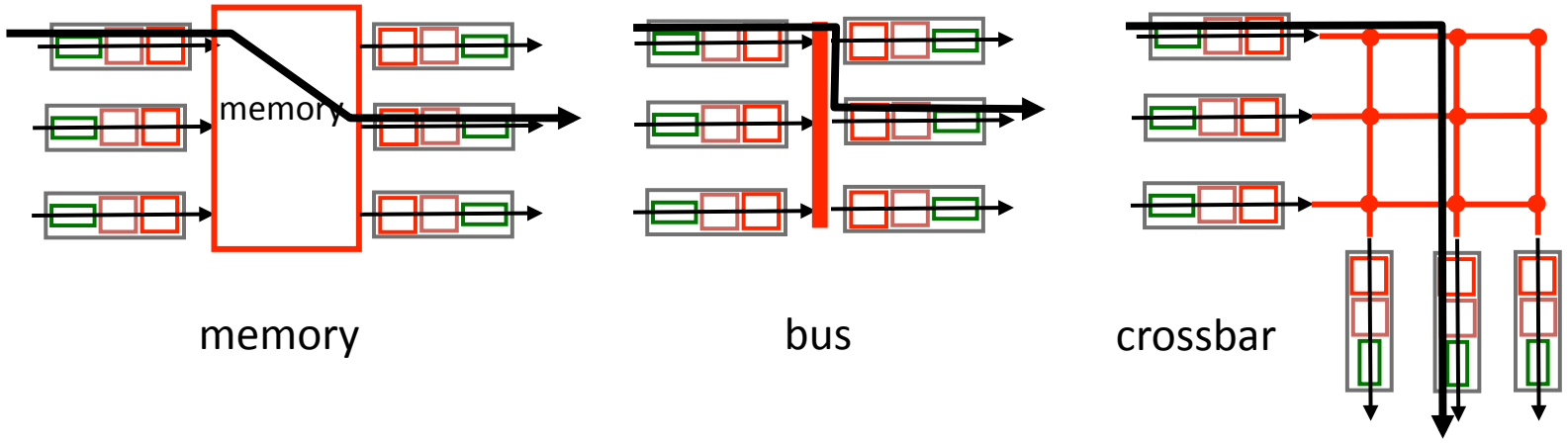
Kurose

Router architecture



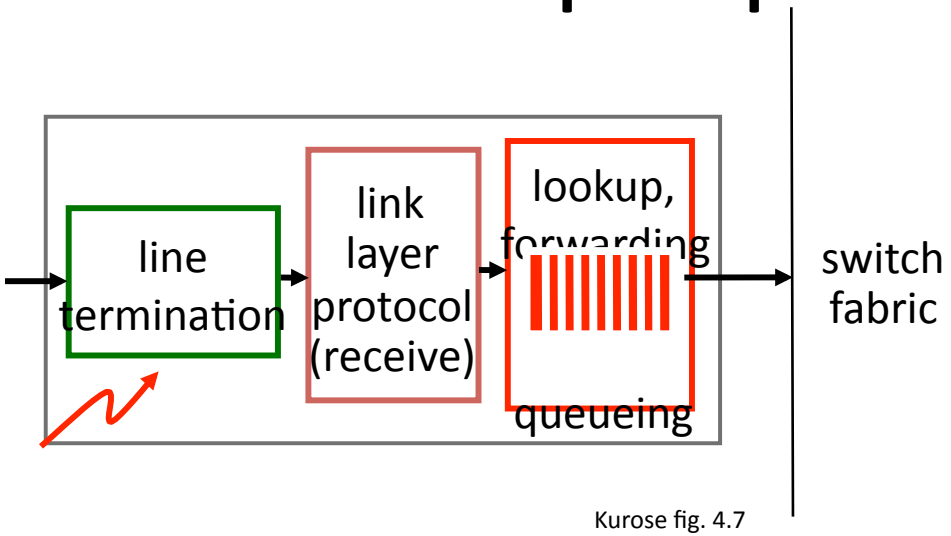
Kurose fig. 4.6

Switching fabric



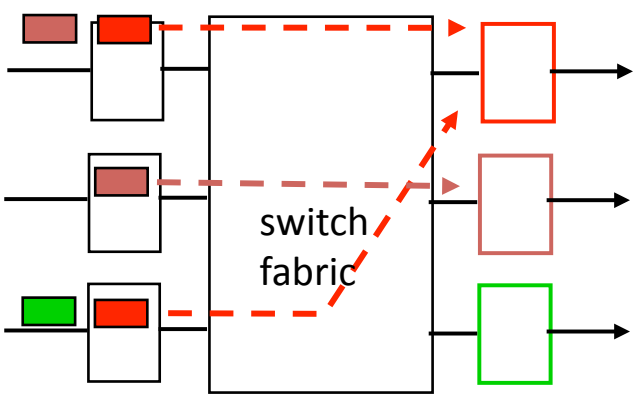
Kurose fig. 4.8

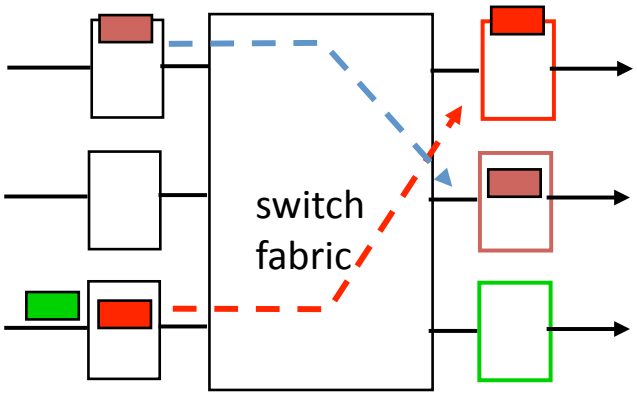
Input port queuing



Kurose fig. 4.7

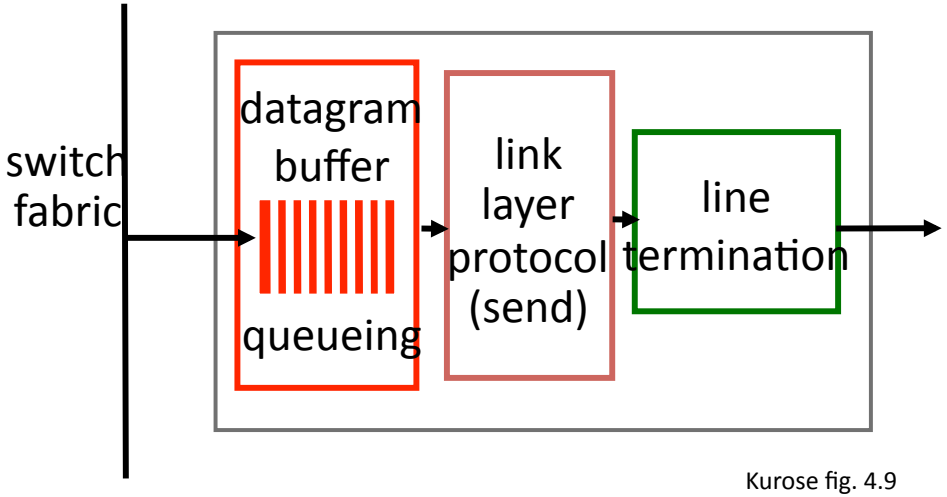
- input port queuing
  - when packets arrive faster than switch fabric takes them in
- Head of Line blocking





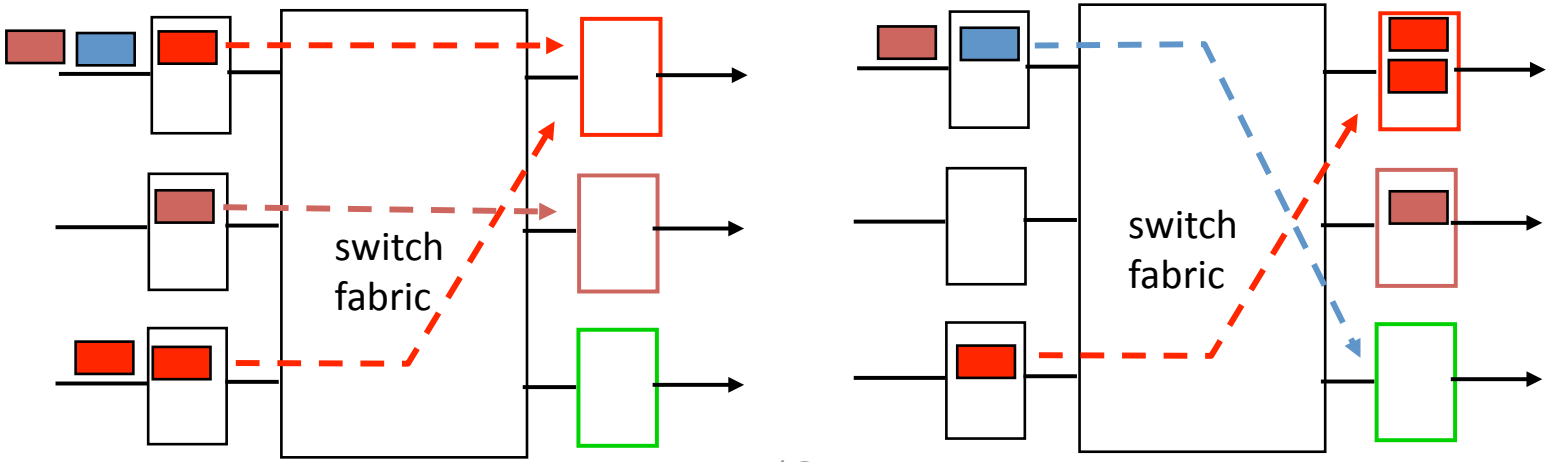
Kurose fig. 4.11

Output port queuing



Kurose fig. 4.9

- output port queuing
  - when switch fabric outputs packets faster than link transmission rate



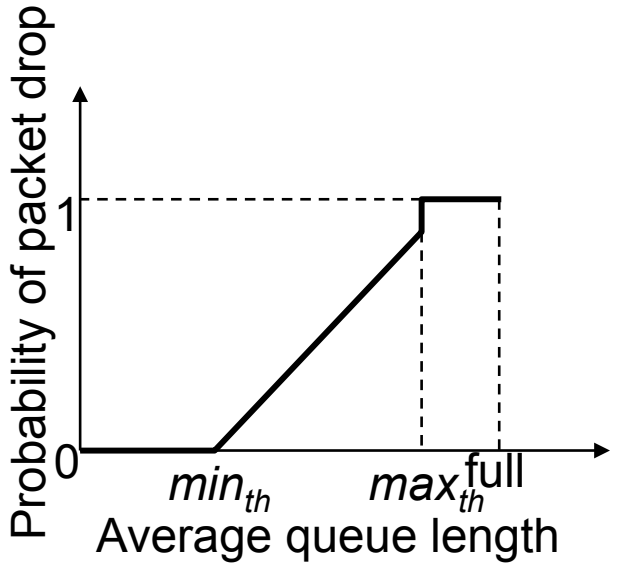
Kurose fig. 4.10

Packet scheduling

- simplest: First In First Out (FIFO)
- later: various types of priority queues and bandwidth allocation

Packet dropping

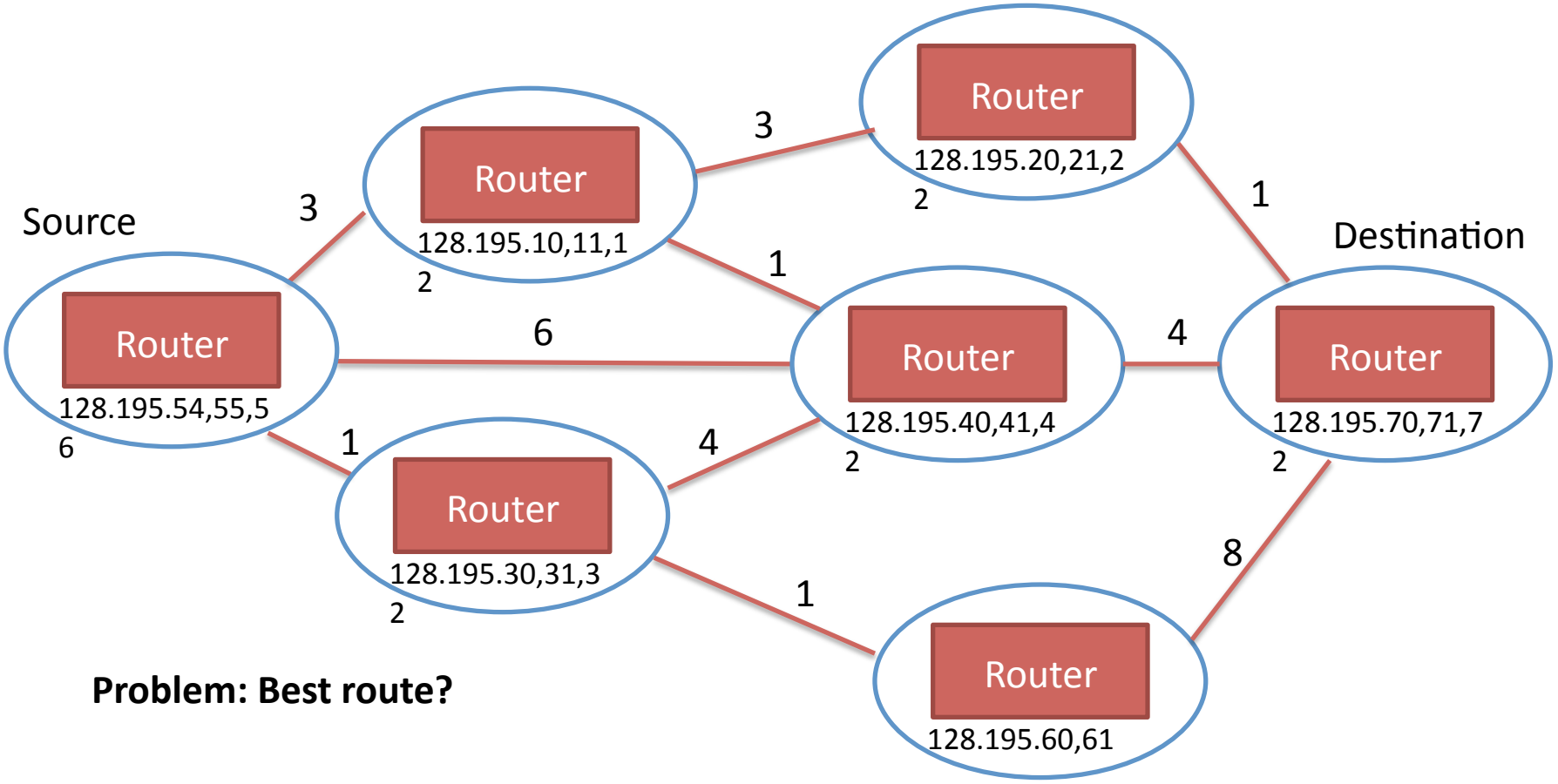
- simplest:
  - drop packet only when buffer is full "drop-tail"
- intermediate:
  - drop packet when buffer is full
  - and sometimes when buffer is almost full
  - probability of drop is a function of queue occupancy
  - when used in conjunction with TCP, "random early detection" can smooth flow control window size changes
- later:
  - drop packet according to priority



Leon-Garcia fig. 7.49

Routing

Routing within an organization



Problem: Best route?

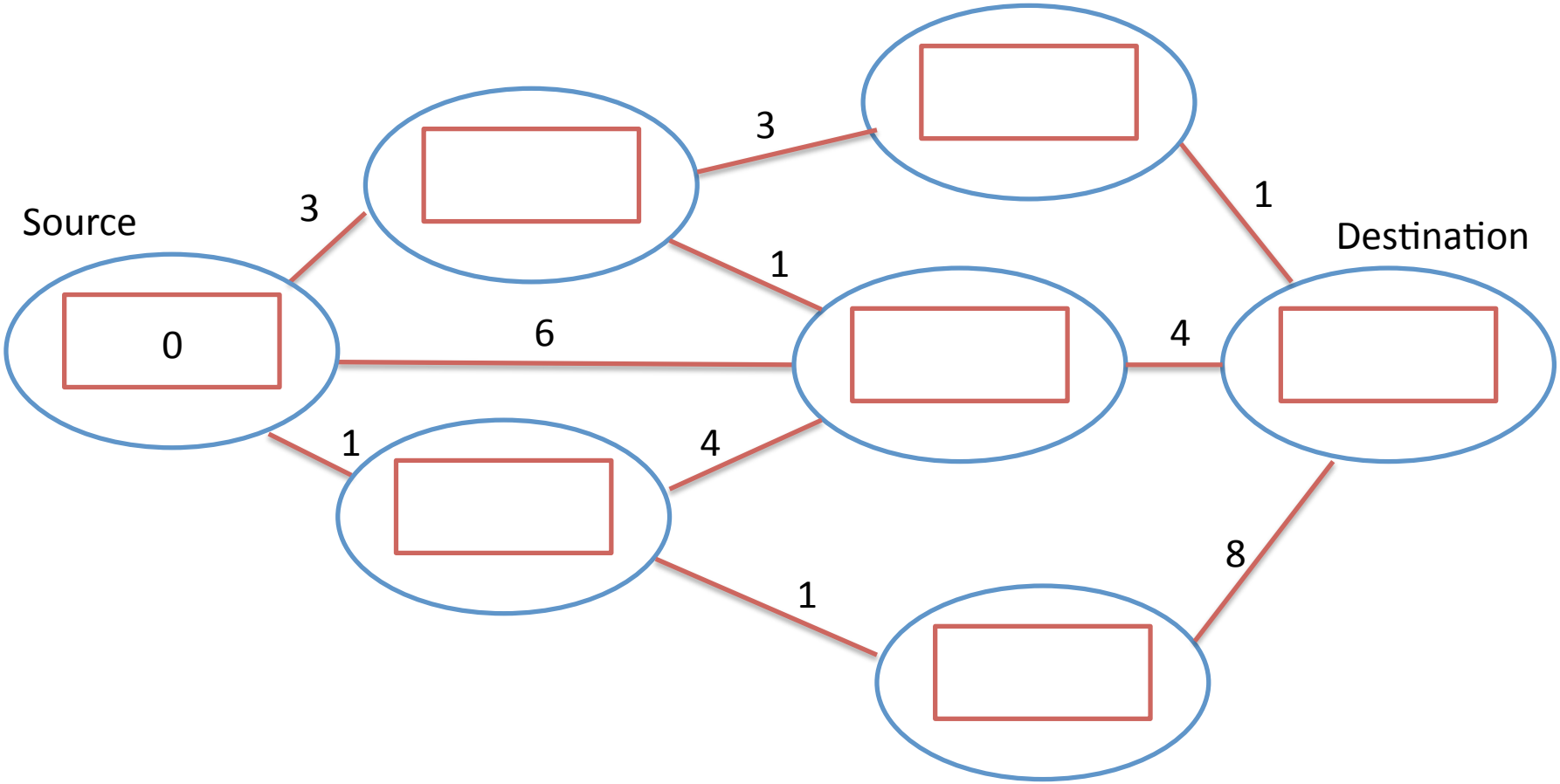
OSPF ("Open Shortest Path First")

- Step 1: Exchange information
  - Each router within the organization talks to each other
  - Sends their list of IP addresses
  - May send estimate of delay on outgoing links
  - Forwards each received routing message to all other routers it connects to
    - up to a maximum # of hops

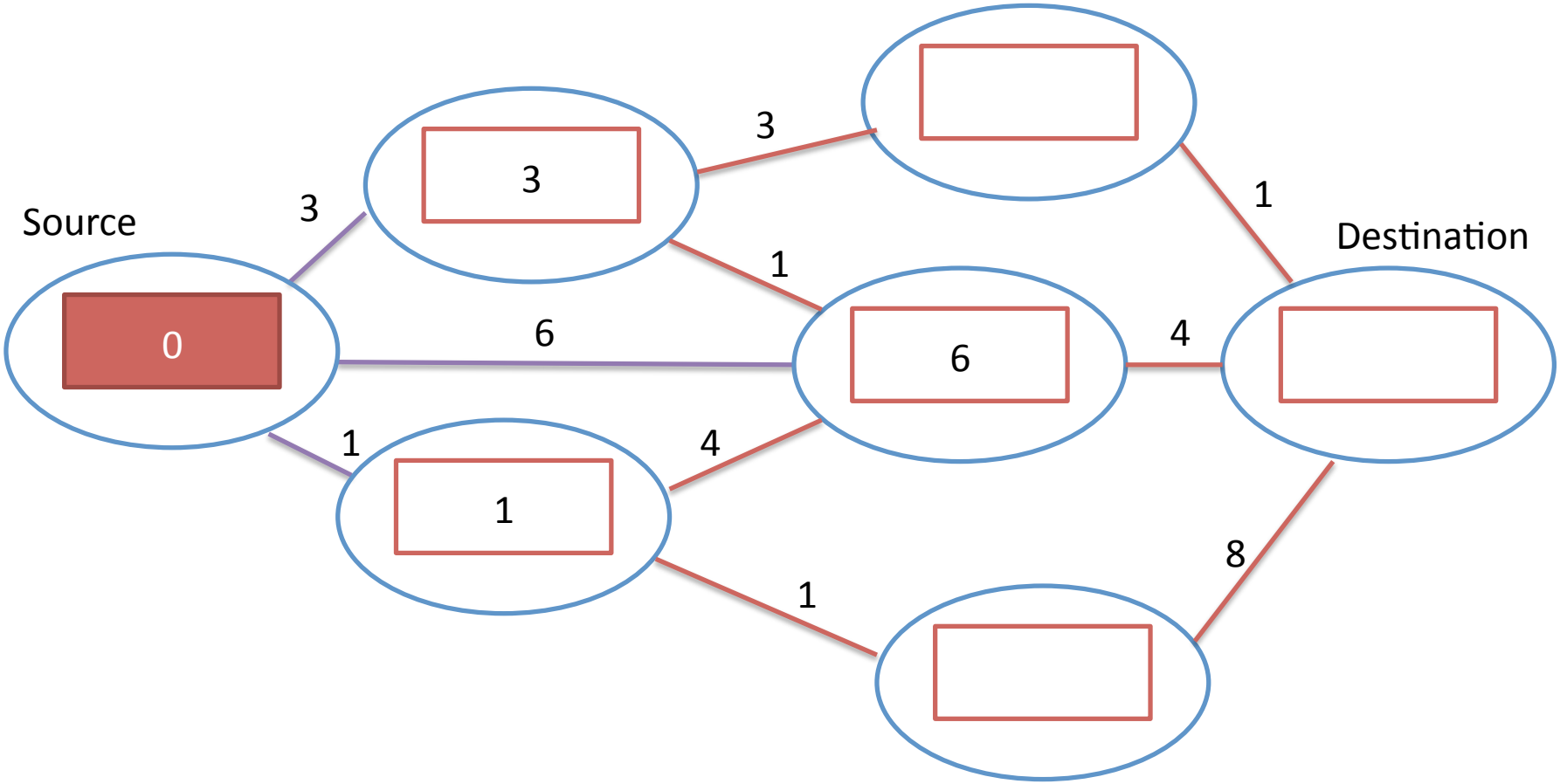
OSPF

- Step 2: Route calculation
  - After step 1, can draw topology picture (2 slides back)
  - Runs a "shortest path" algorithm to find the best route to each IP address
- Review of Shortest Path algorithm
  - Initial label = infinity (or blank) for all nodes other than source, zero for source; nodes unmarked
  - Pick unmarked node with smallest label
  - Update its children and mark it; mark/unmark shortest path links

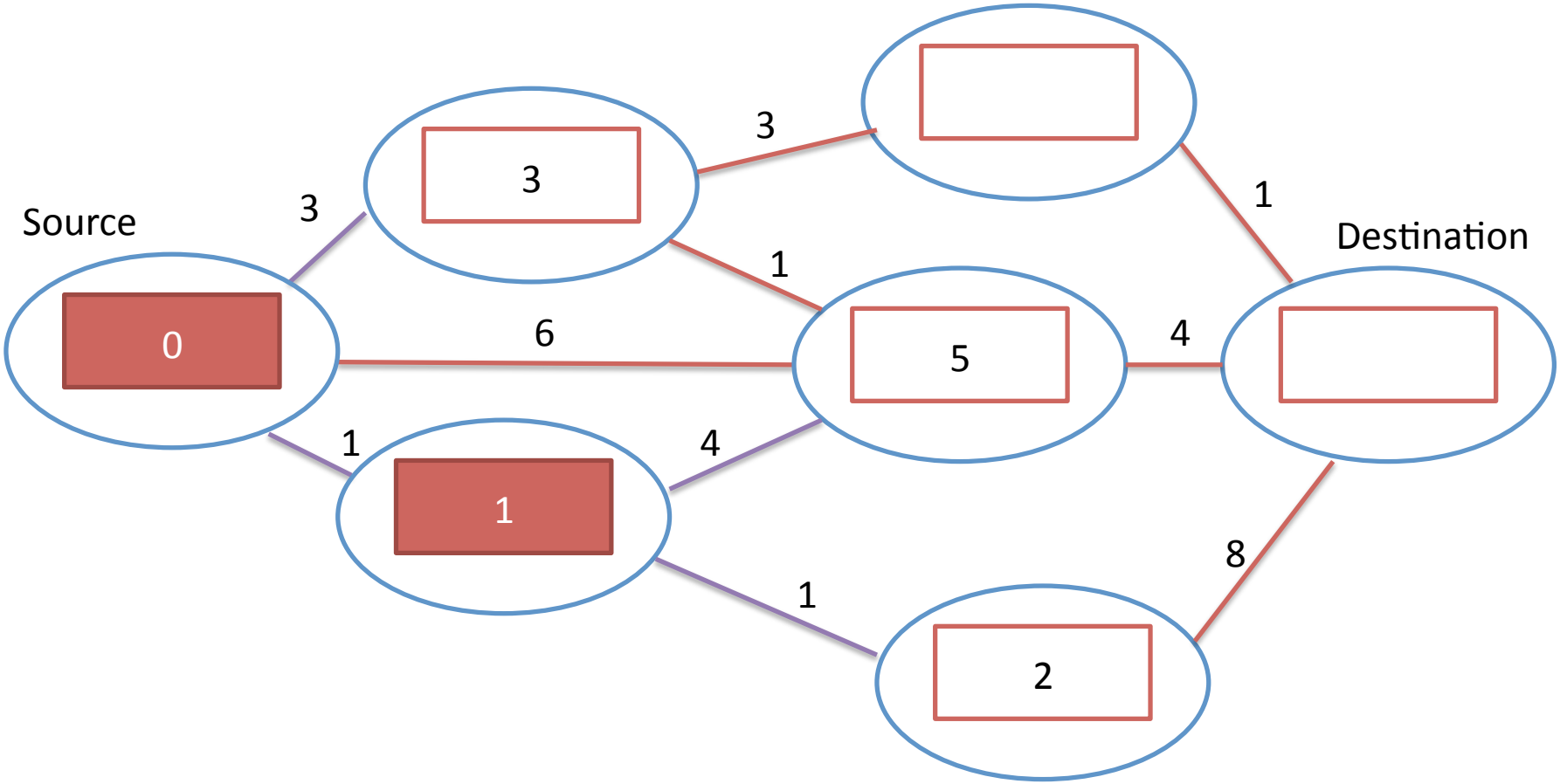
OSPF



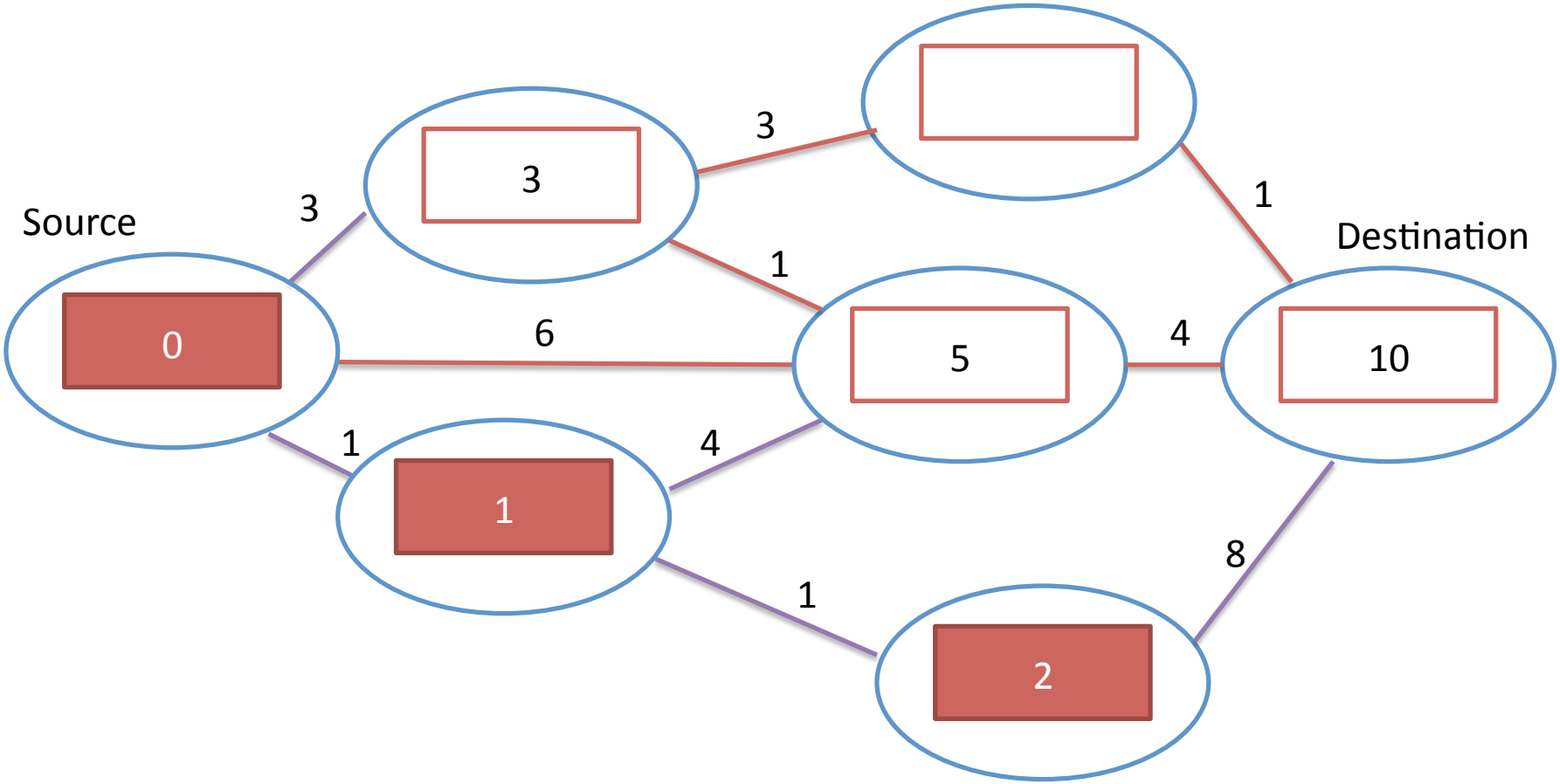
OSPF



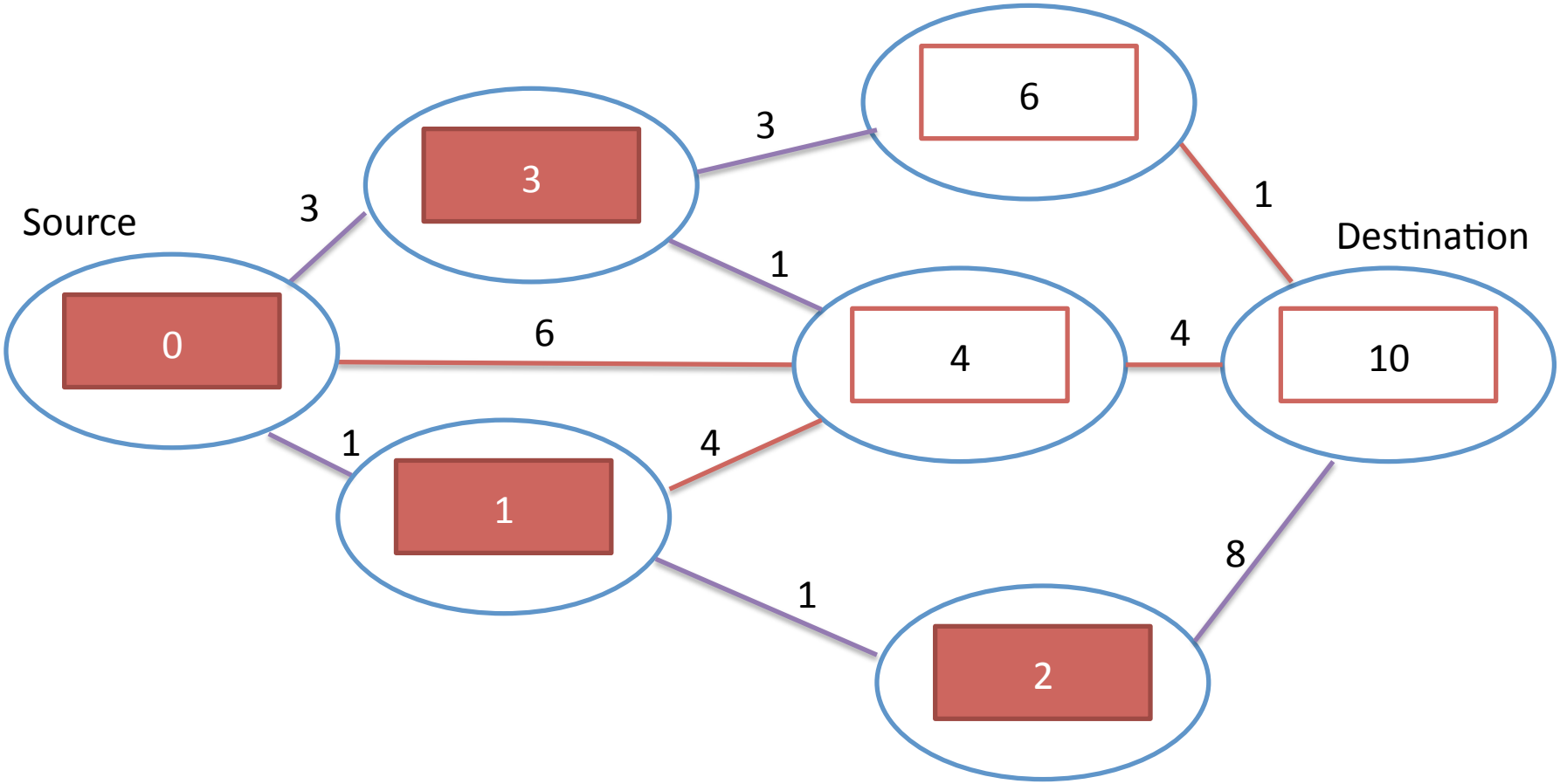
OSPF



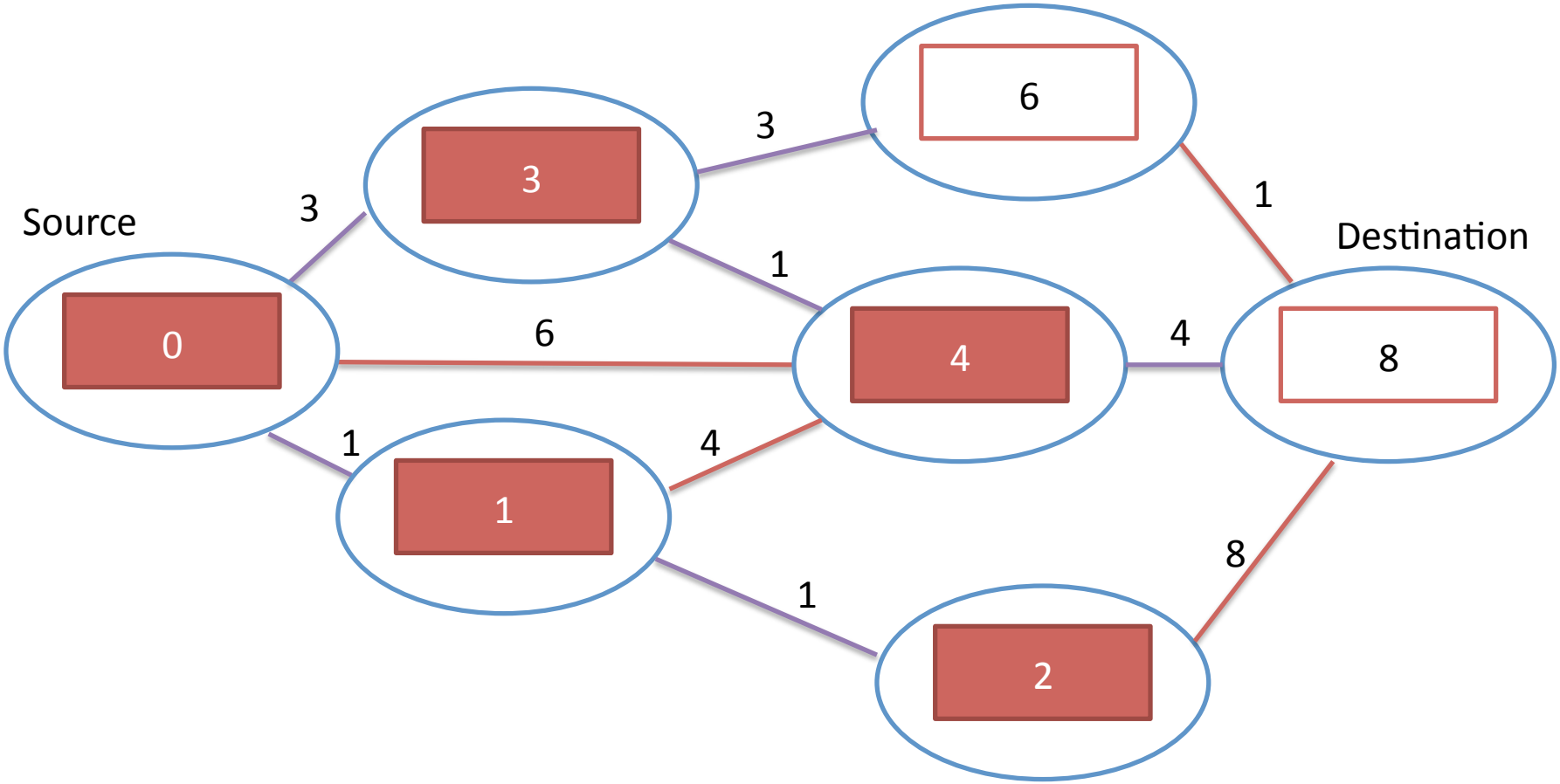
OSPF



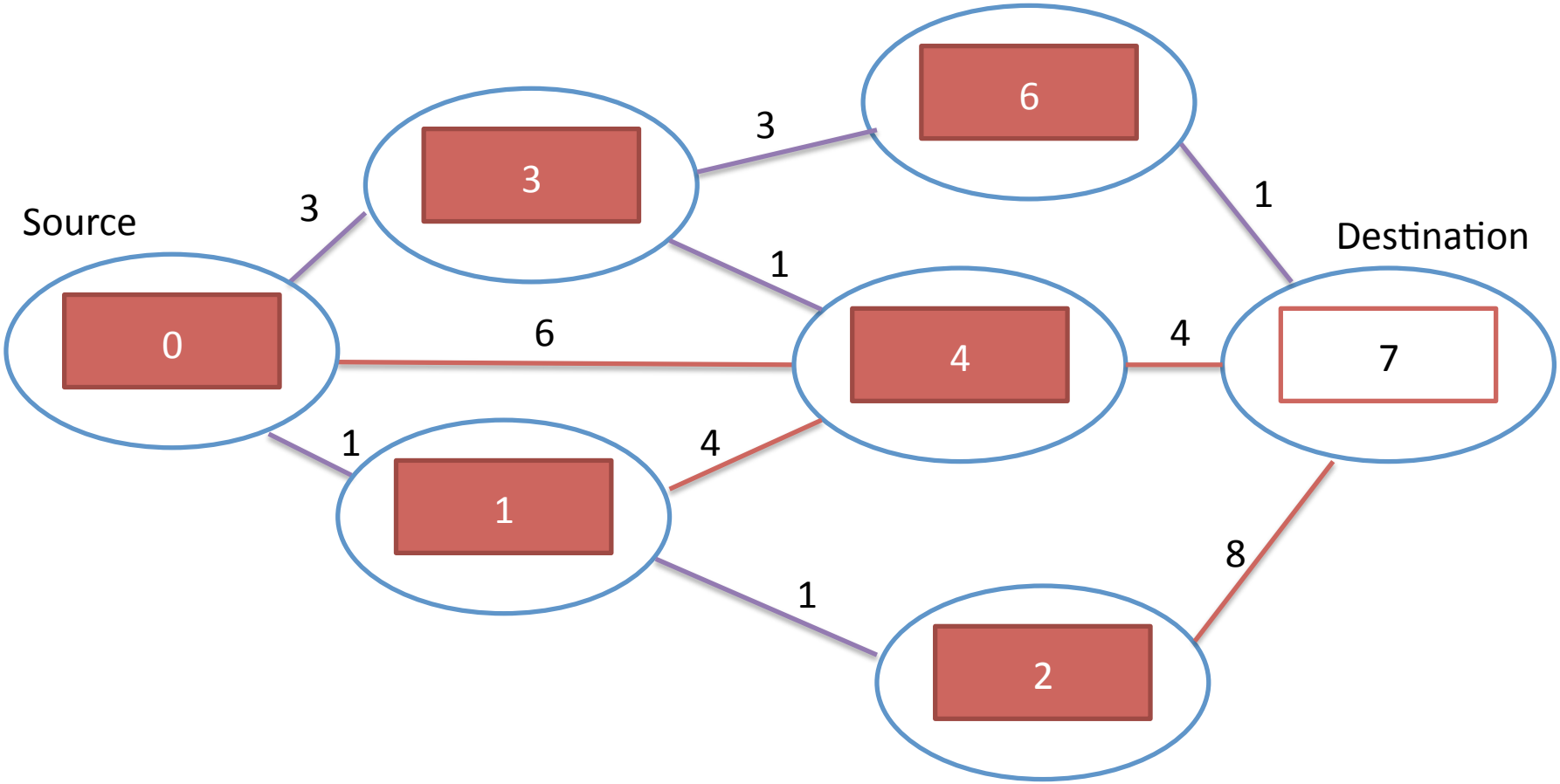
OSPF



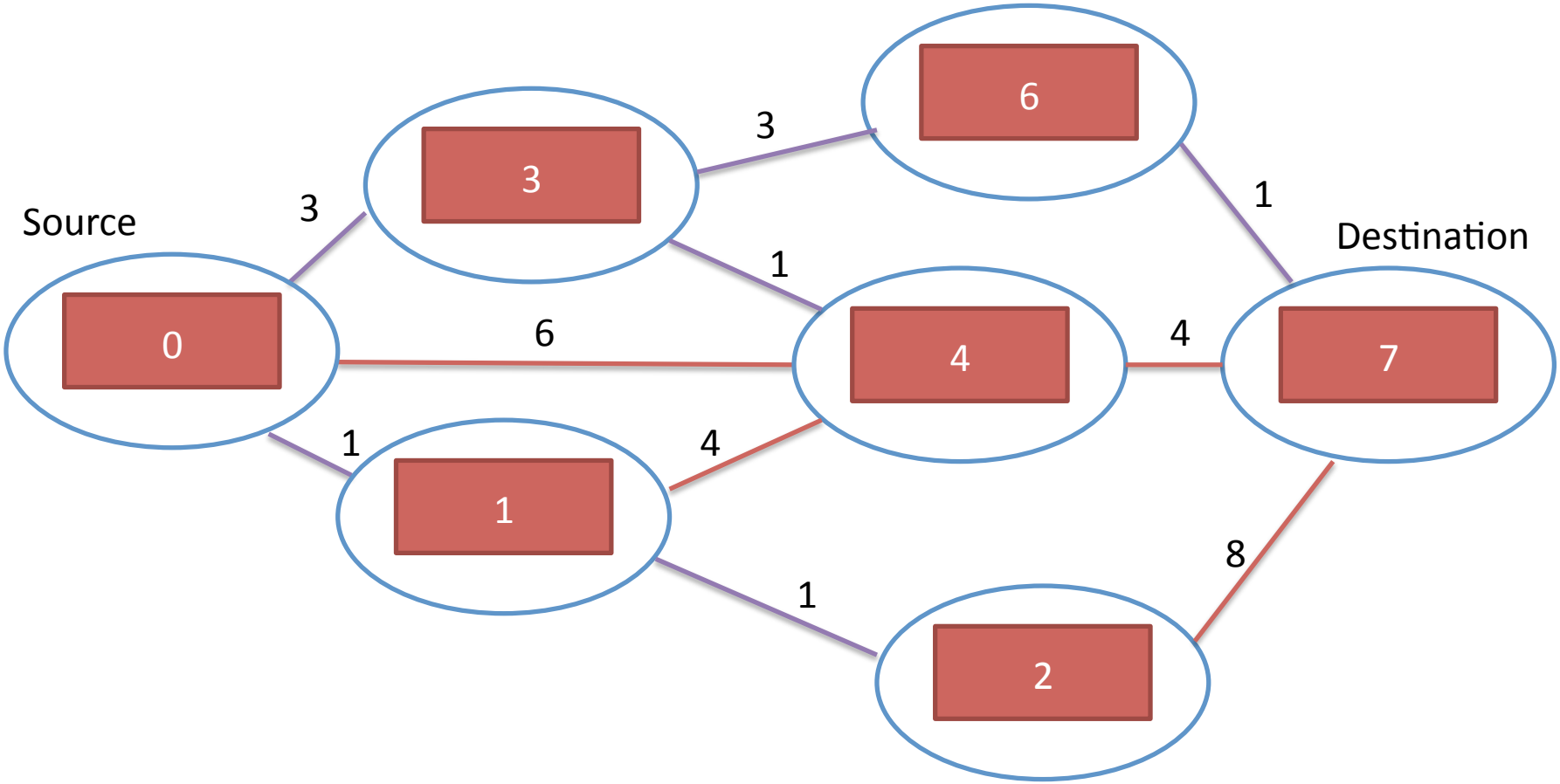
OSPF



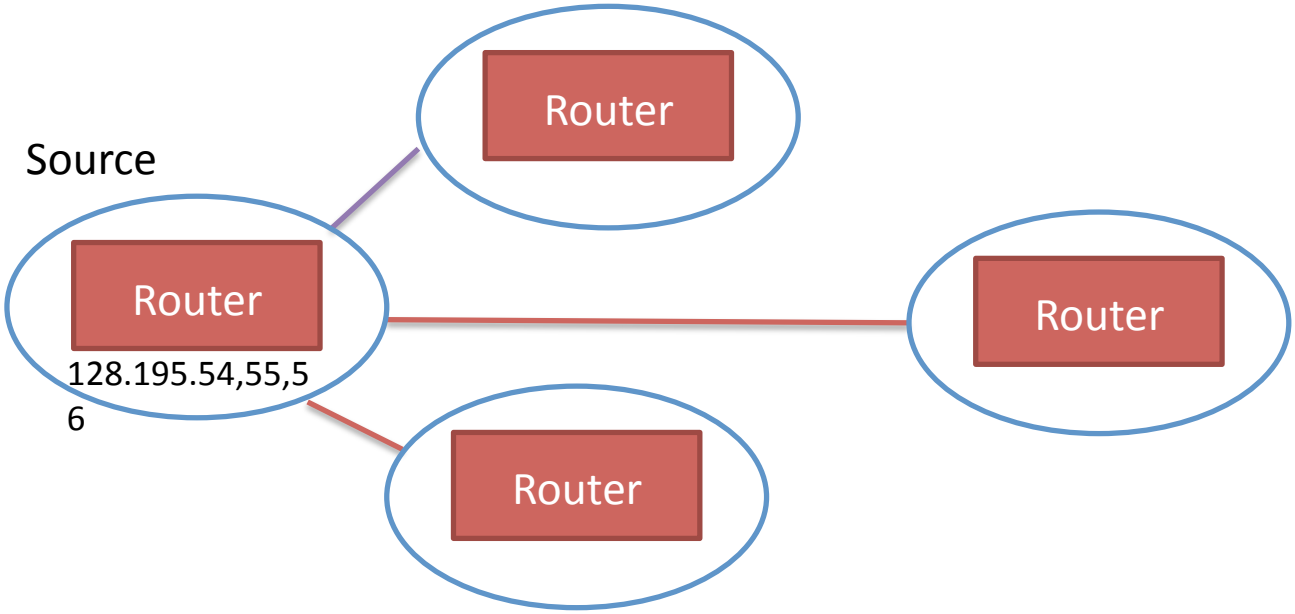
OSPF



OSPF

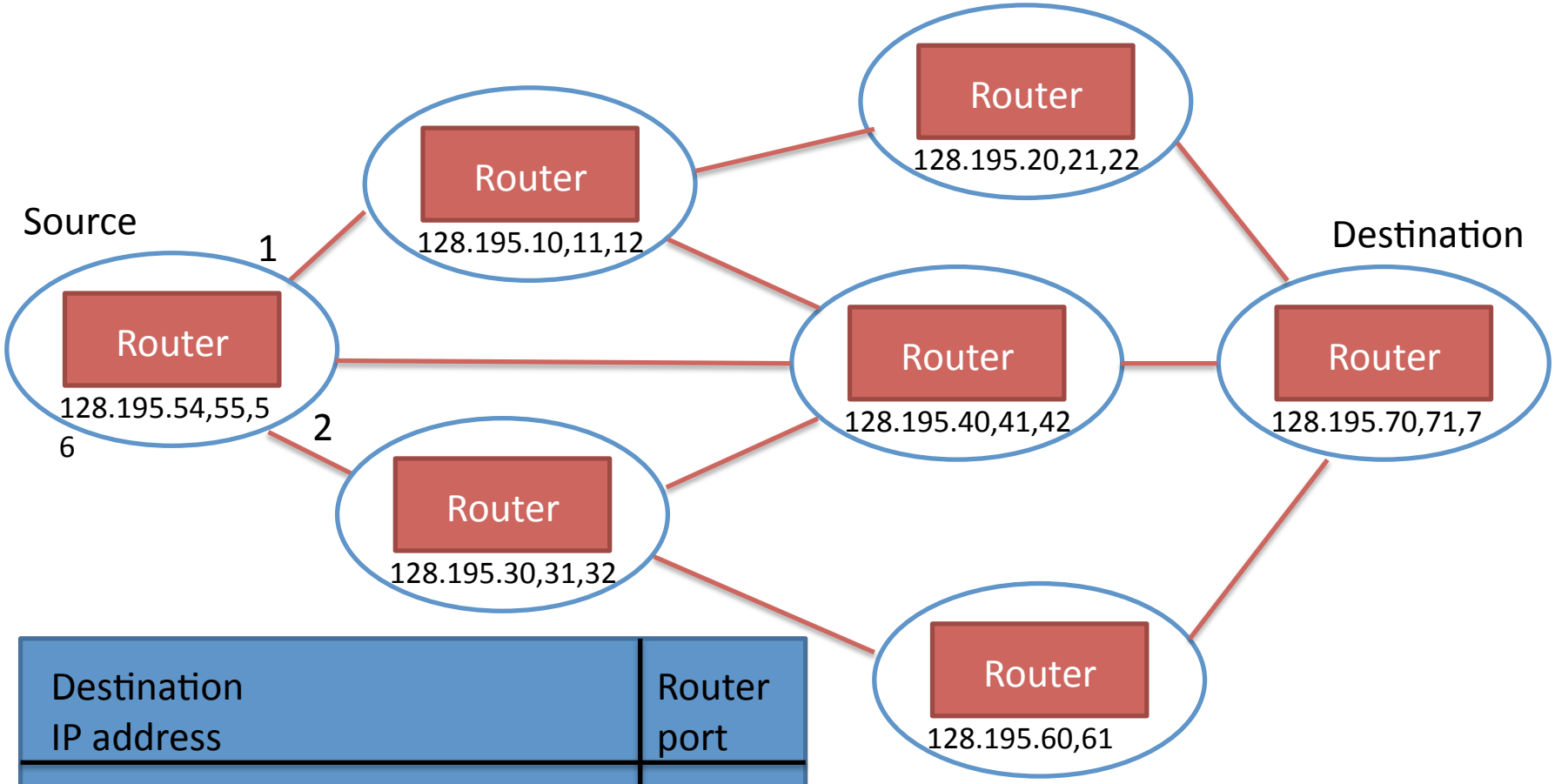


OSPF



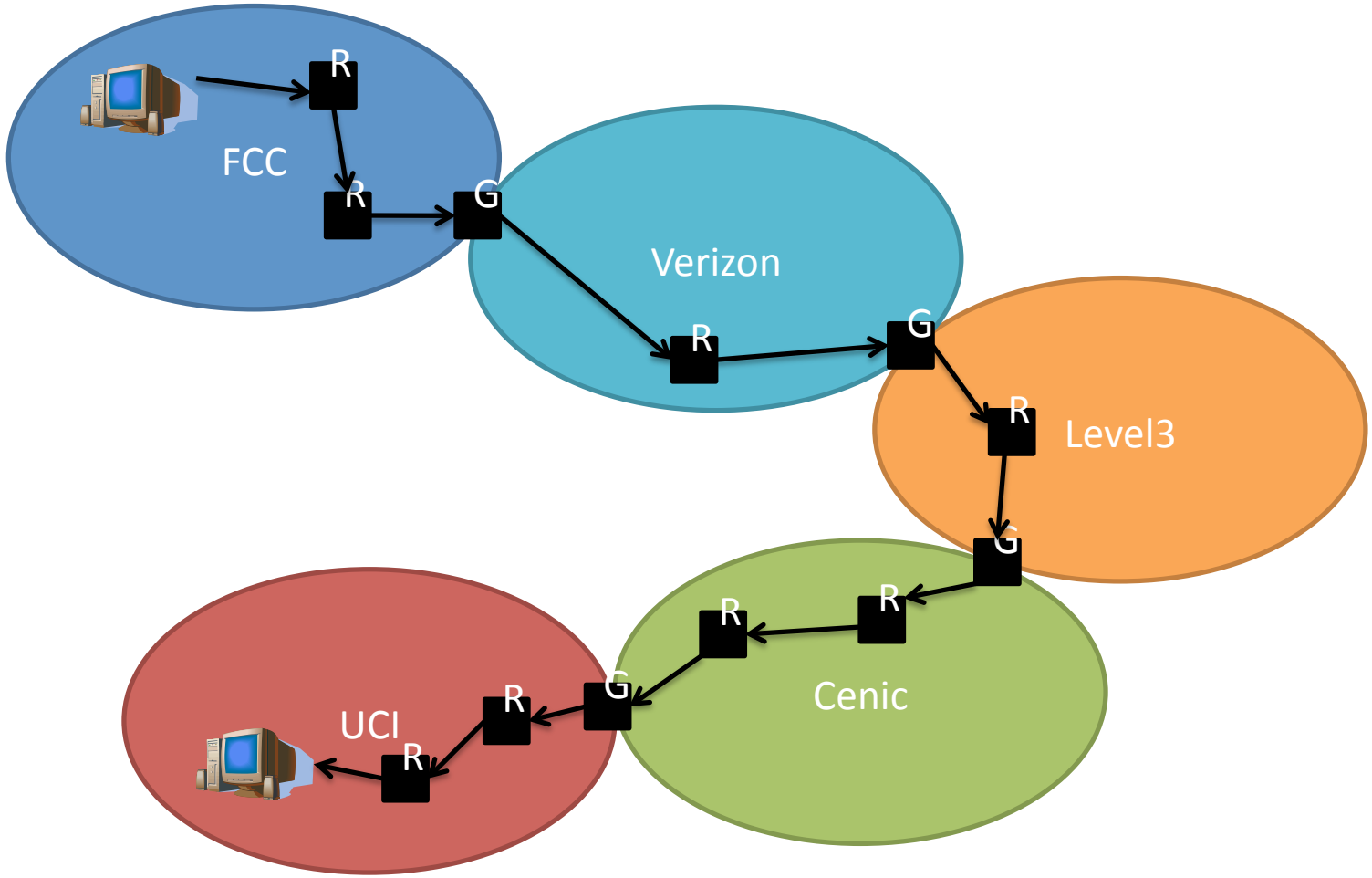
Only need put the first hop for each route into the routing table

Routing within an organization

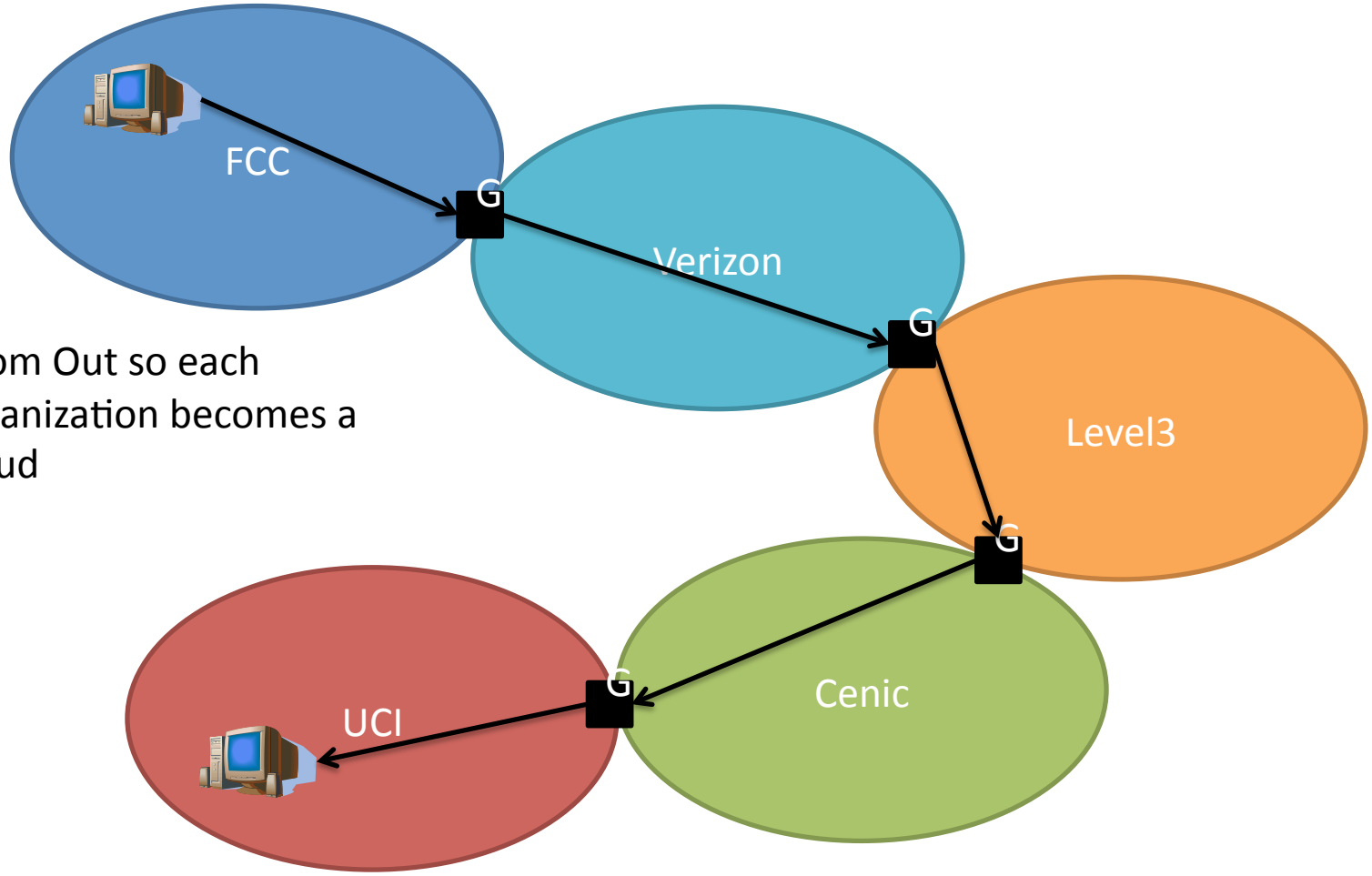


Destination IP address	Router port
128.195.10,11,12,20,21,22,40,…	1
128.195.30,31,32,60,61	2

Internet routing



Routing between organizations



Zoom Out so each organization becomes a cloud

Gateways

- "Autonomous System (AS):
  - organization
  - or part of one
- "gateway"
  - router that connects to another AS
  - "boundary router"
    - sometimes called "border router"

Service Level Agreement (SLA)

- Decision between two ISPs to interconnect
  - legal decision
- contract (SLA) specifies:
  - which traffic an ISP will accept
  - any charges
  - may also specify Service Level Specification (SLS):
    - delay
    - loss
    - availability

Hierarchical routing

- Why not use a single routing algorithm everywhere?
  - complexity
  - autonomy
- Routing within an Autonomous Systems
  - "interior gateway protocol"
  - "intra-domain routing"
- Routing between Autonomous Systems
  - "exterior gateway protocol"
  - "inter-domain routing"

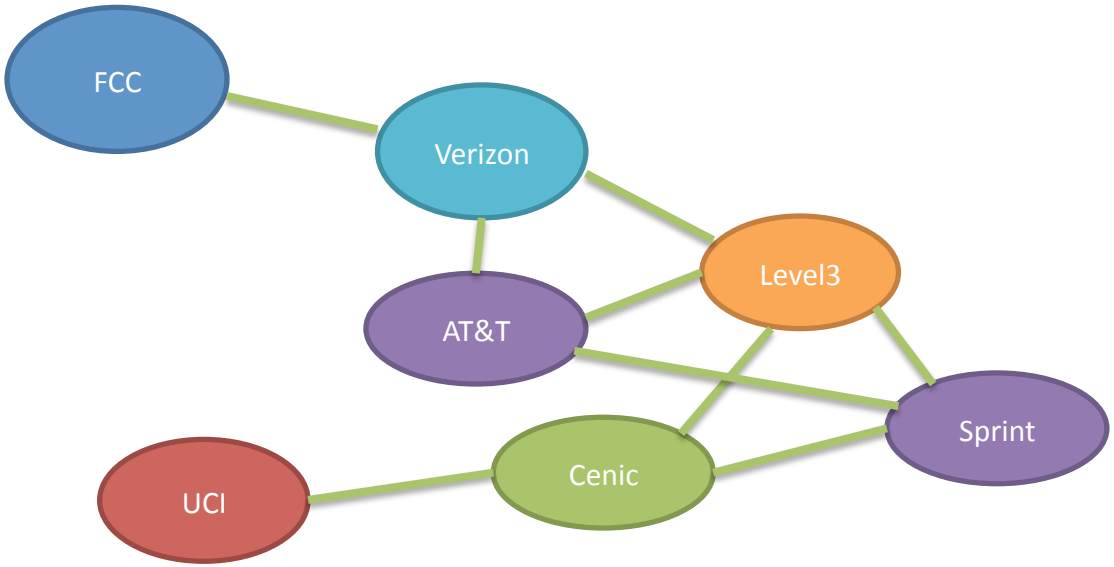
Border Gateway Protocol (BGP)

Overview

- Route advertisement
- Route calculation
- Coordination with OSPF

BGP route advertisement

- Exchange information amongst Autonomous Systems
  - Boundary routers talk to each other



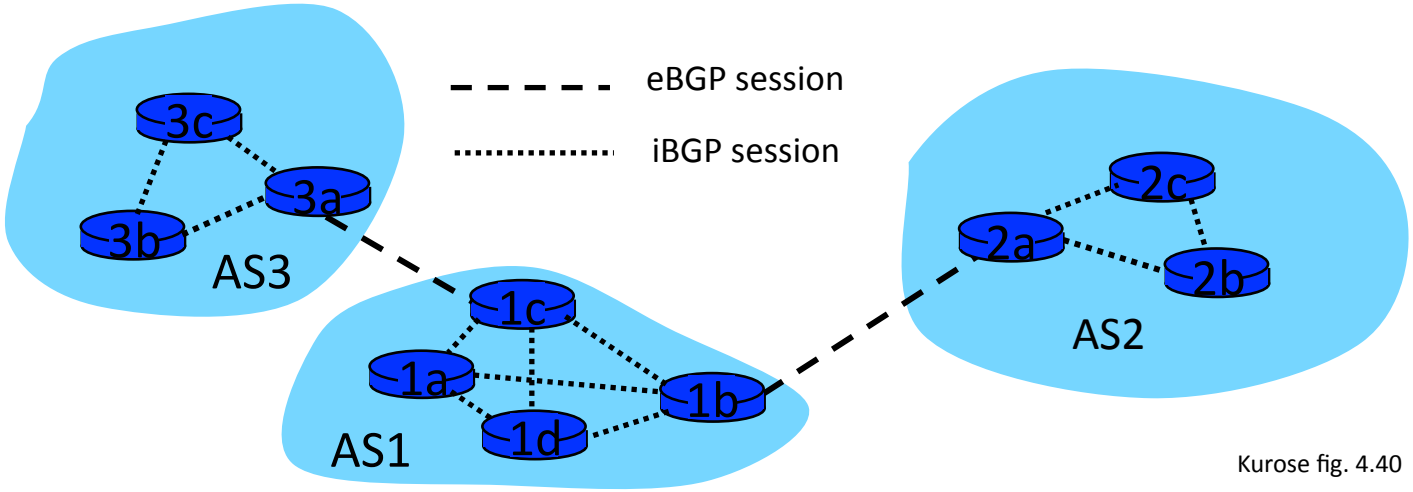
BGP route advertisement

- Creating route advertisements
  - Send a list of IP addresses within its Autonomous System
    - aggregated into IP prefixes
  - Send the routes it uses
    - for each IP prefix:
      - link corresponding to first AS hop
  - May send performance metrics

BGP route calculation

- Route calculation
  - Using received route advertisements, draw AS-level topology graph
    - only includes routes currently used by each AS
  - If there are multiple possible routes to an IP prefix, use AS policies, including:
    - cost of using path
    - length of path (at AS level)
    - number of IP hops within AS on route

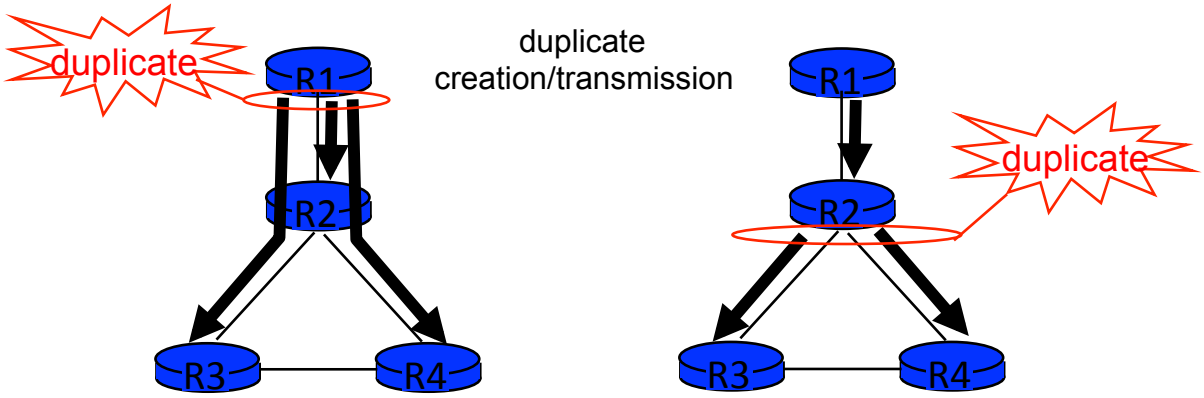
BGP coordination with OSPF



Kurose fig. 4.40

- Boundary routers send their routing decisions to all other routers within Autonomous System (iBGP)
- Routing table thus contains:
  - OSPF routes for prefixes inside AS
  - combined OSPF/BGP routes for prefixes outside AS

Broadcast routing



Kurose fig. 4.43

source duplication

in-network duplication

- In-network duplication:
  - potentially reduces number of copies of message
  - allows sender to broadcast without knowing complete list of destinations
  - requires
    - computation of broadcast routes
    - broadcast-capable routers

Broadcast routing: flooding

- Flooding:
  - send a copy of broadcast packet to all neighbors (other than on link packet was received)
  - problem: endless duplication of each packet
- Controls on flooding:
  - TTL
  - sequence number
  - forwarding router ID

Broadcast routing: reverse path forwarding

- Forward broadcast packet only if arrived on link on shortest path tree back to source of packet
  - look in routing table for source

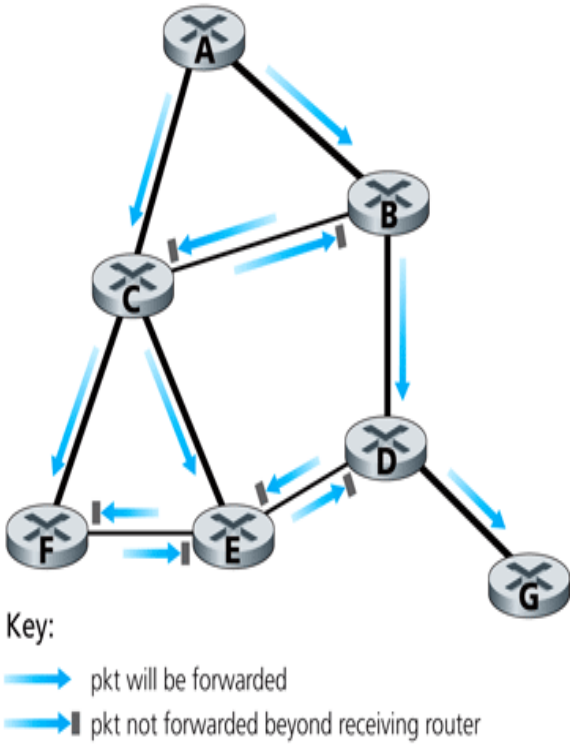
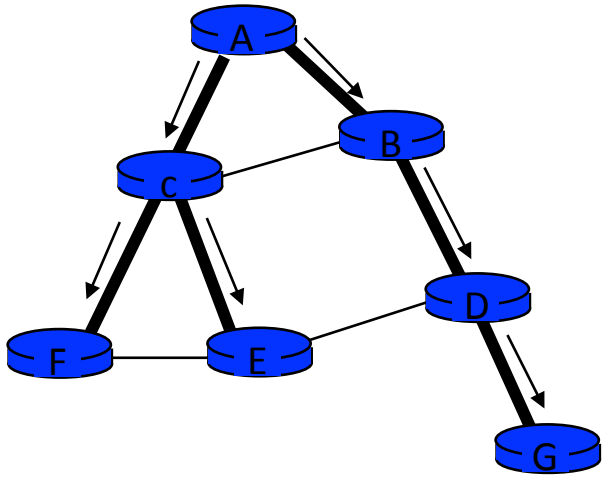


Figure 4.45 ♦ Reverse path forwarding

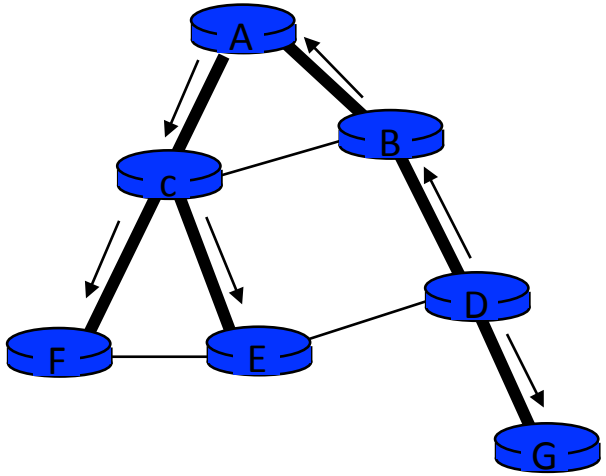
Kurose fig. 4.44

Broadcast routing: spanning tree

- Create a spanning tree
- Tree may be re-used for broadcast from nodes other than root



(a) Broadcast initiated at A



(b) Broadcast initiated at D

Kurose fig. 4.45

Multicast routing

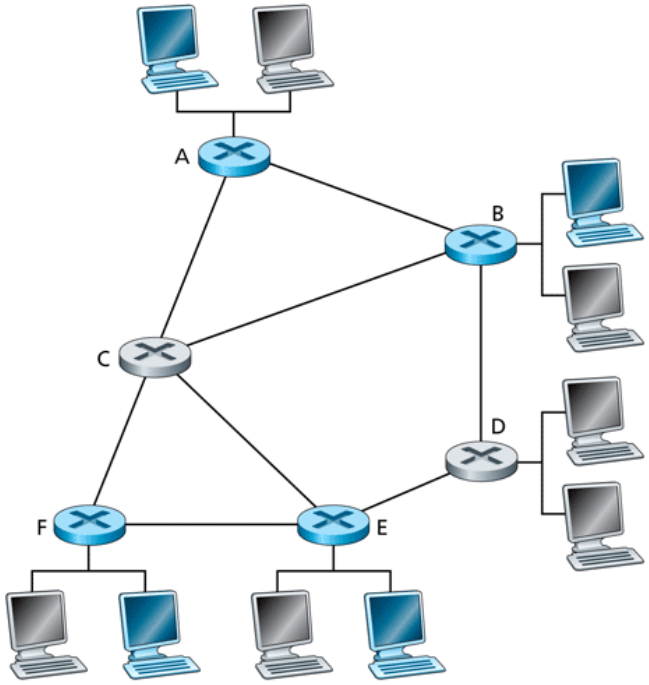


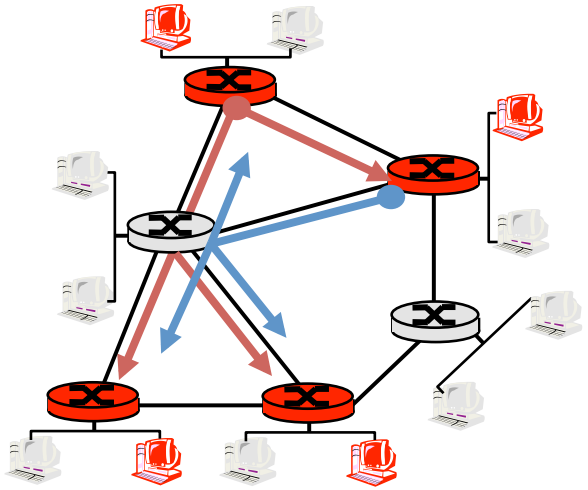
Figure 4.50 ♦ Multicast hosts, their attached routers, and other routers

Kurose fig. 4.49

- subset of hosts are subscribed
- routers:
  - subset with subscribed hosts
  - plus those required to form routes
- common applications:
  - 1 host is source, all others destination
  - all hosts are sources, all hosts are destinations

Multicast routing: source-based trees

- each source uses a spanning tree with itself as the root
  - e.g. use reverse-path forwarding to construct the tree
  - and "prune" the routers on the tree that don't lead to subscribing hosts



Kurose