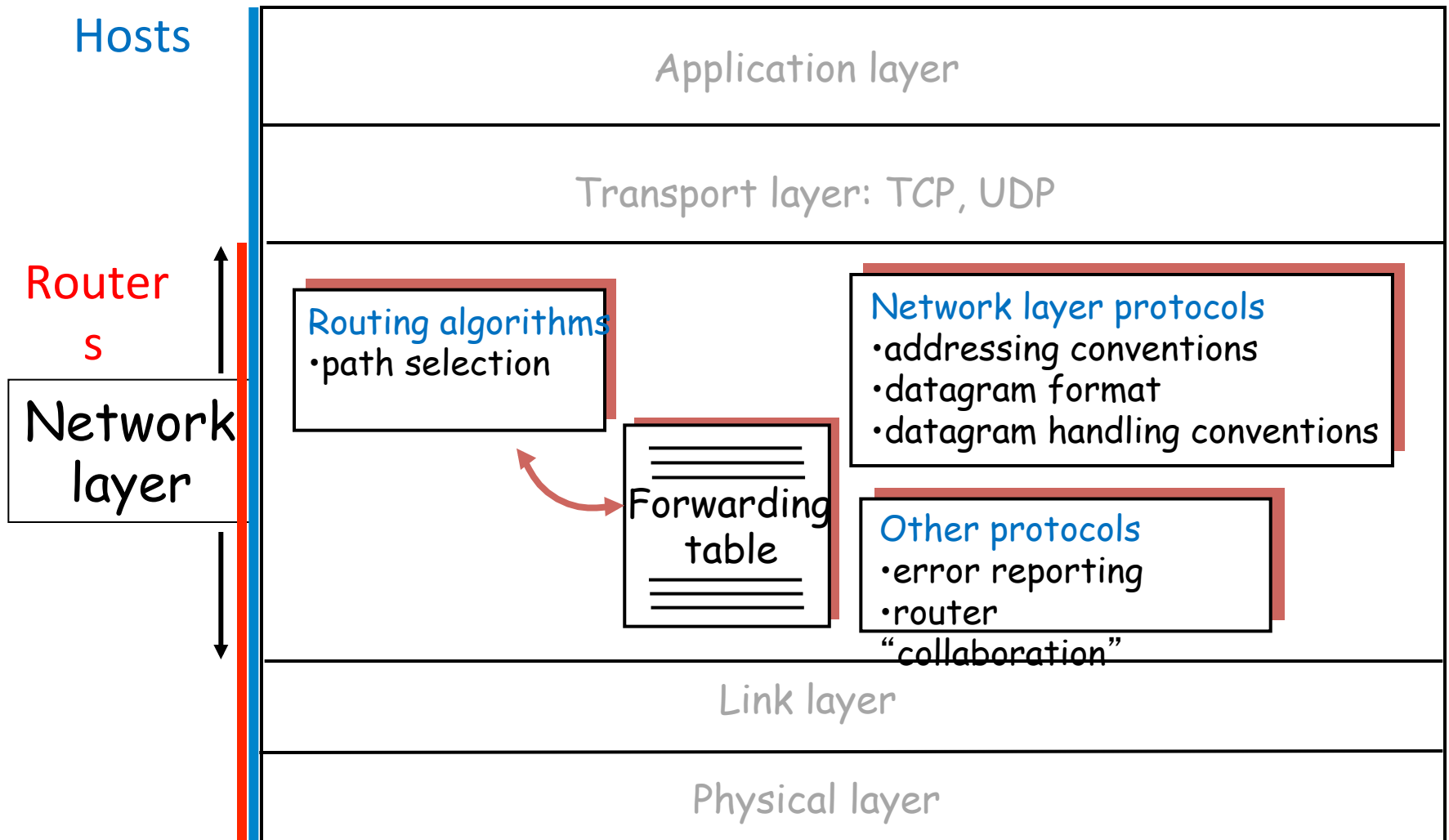


## The network layer

- hardware functions
  - routing and forwarding
- forwarding tables

**Note:** Many of the lecture slides are based on presentations that accompany *Computer Networking: A Top Down Approach*, 6<sup>th</sup> edition, by Jim Kurose & Keith Ross, Addison-Wesley, 2013.

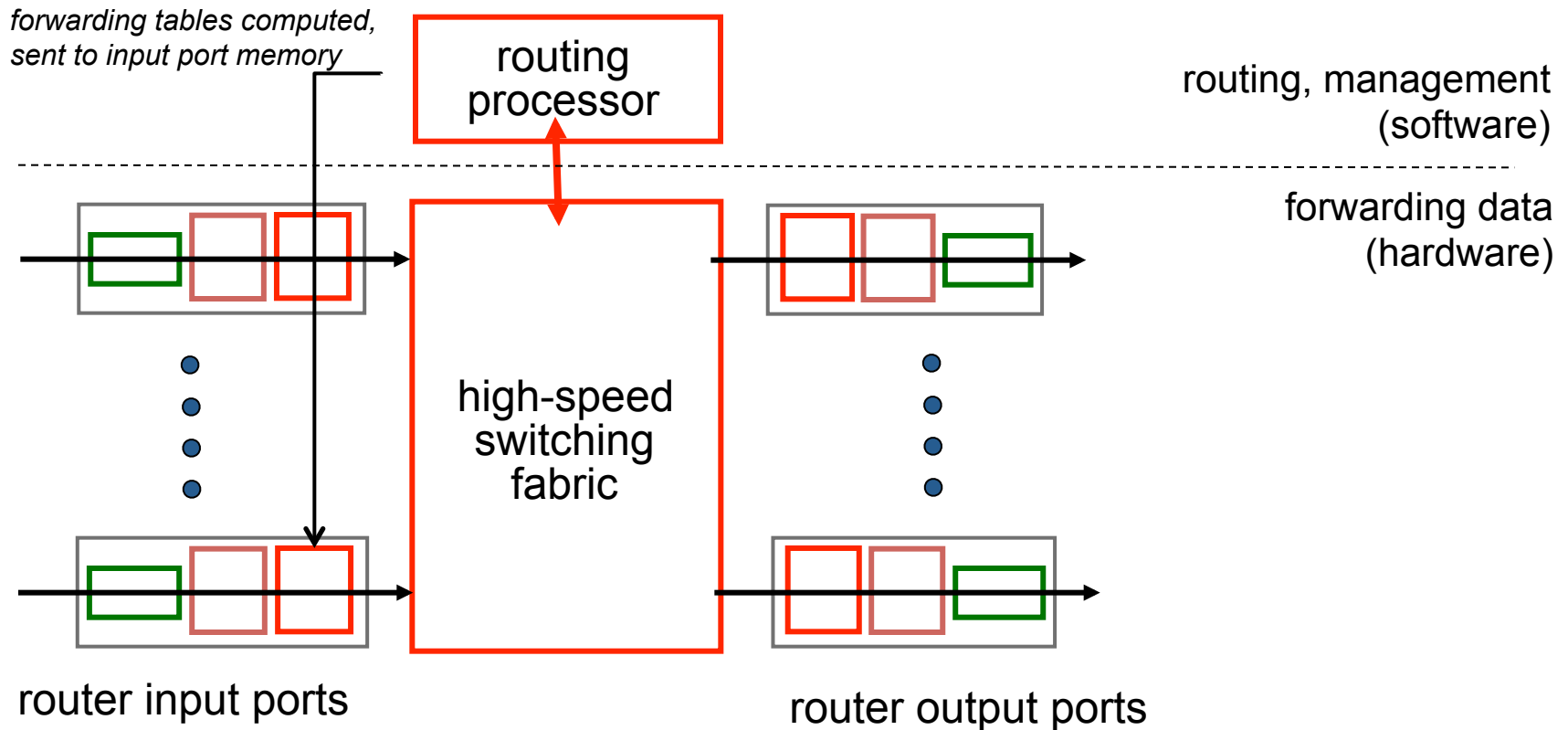
# The network layer functions



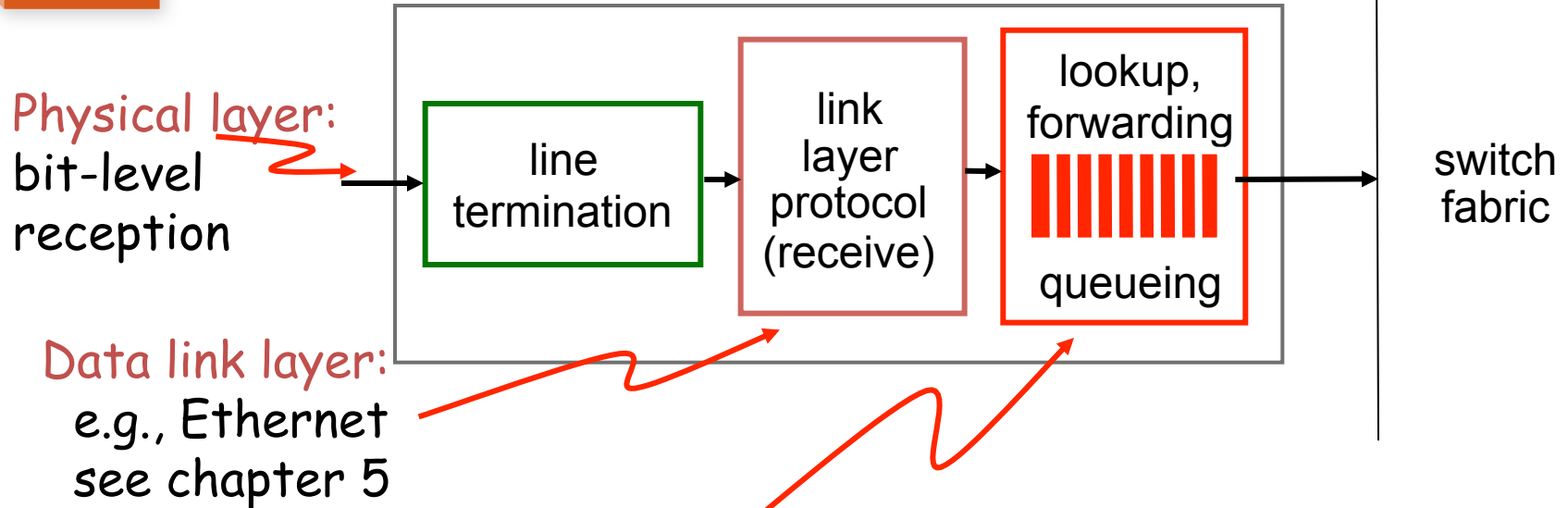
# Router architecture overview

Two key router functions: *routing* and *forwarding*

- Determine *route* (more later on algorithms/protocols)
- *forward* datagrams from incoming to outgoing link



# Input Port Functions

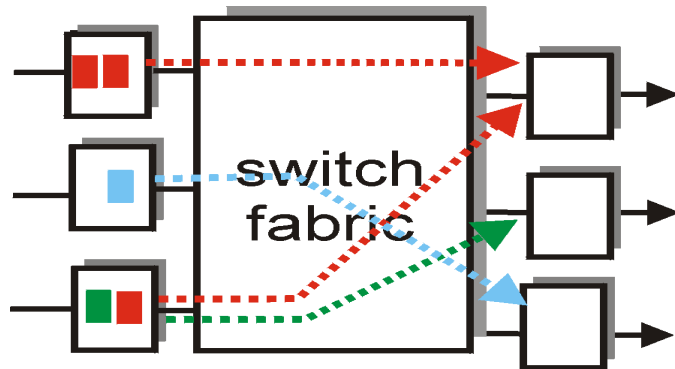


## Network layer (in):

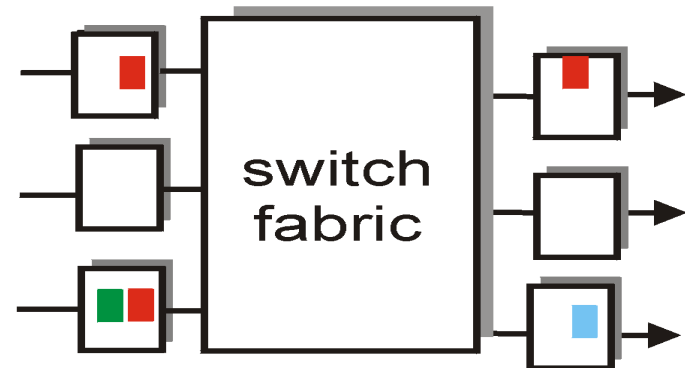
- **Queuing**: if datagrams arrive faster than forwarding rate into switch fabric
- **Lookup**: given datagram destination, lookup output port using forwarding table in input port memory
- **Forwarding**: forward to appropriate output port

# Input Port Queuing

- Fabric slower than combined input ports
  - queuing may occur at any of the input ports
  - output port contention
  - **Head-of-the-Line (HOL) blocking:** queued datagram at front of queue prevents others in queue from moving forward
- *queuing delay and loss due to input buffer overflow*

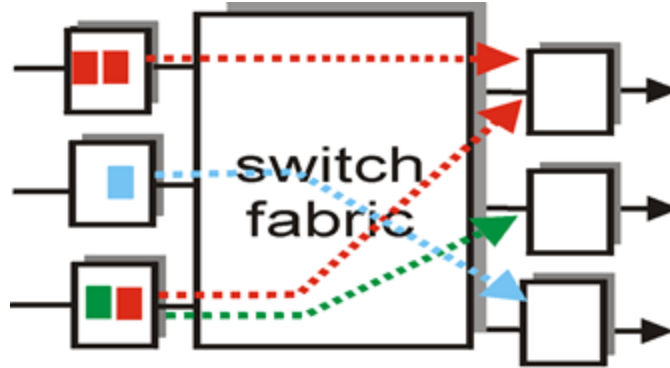


output port contention  
at time t - only one red  
packet can be transferred



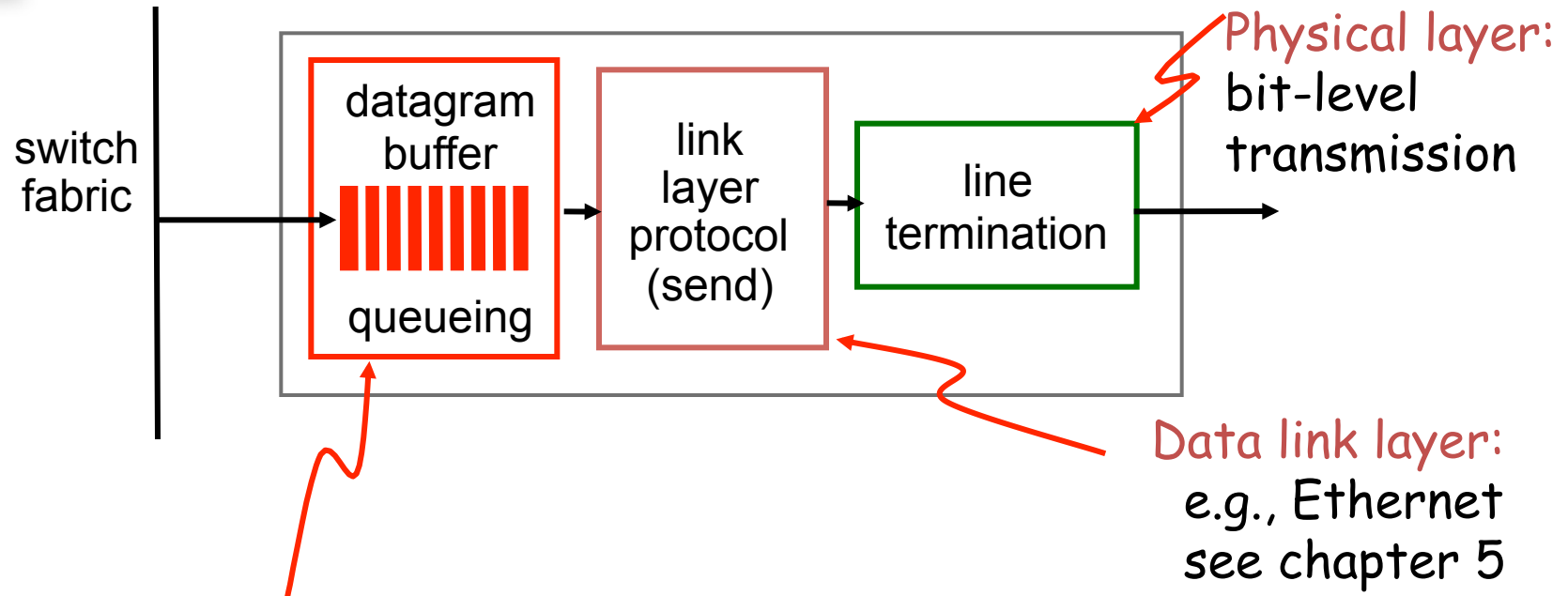
green packet  
experiences HOL blocking

# Switching fabric functions



- Transfer datagram to selected output port
- Prevent collisions

# Output Port Functions

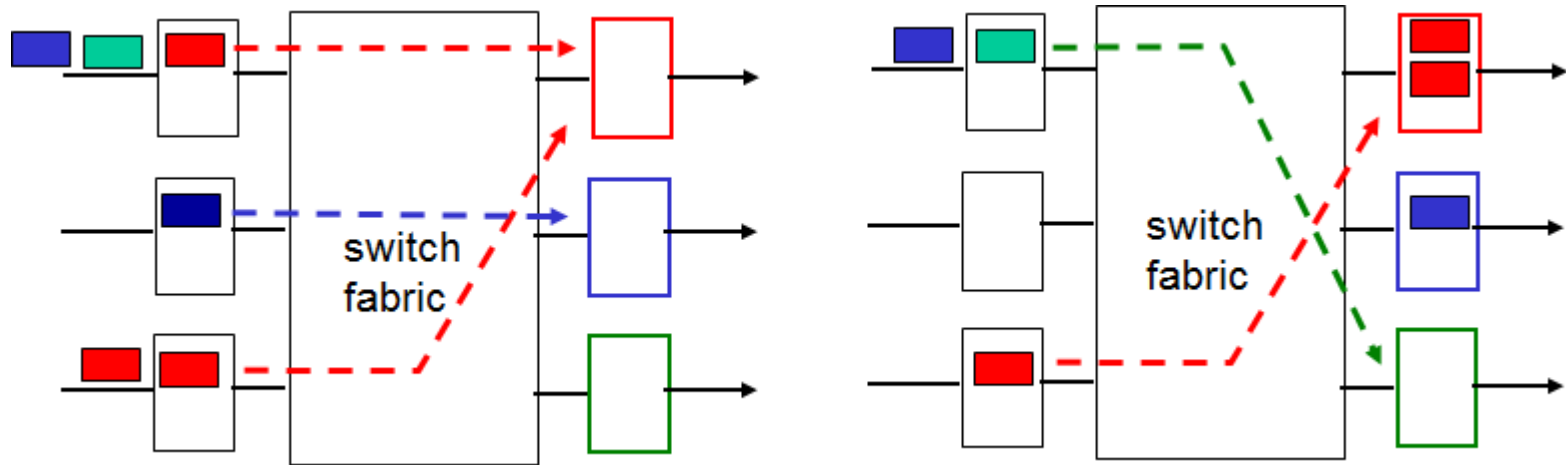


## Network layer (out):

- *Queuing* required (at each output port) if datagrams arrive from fabric faster than the link's transmission rate

# Output port queueing

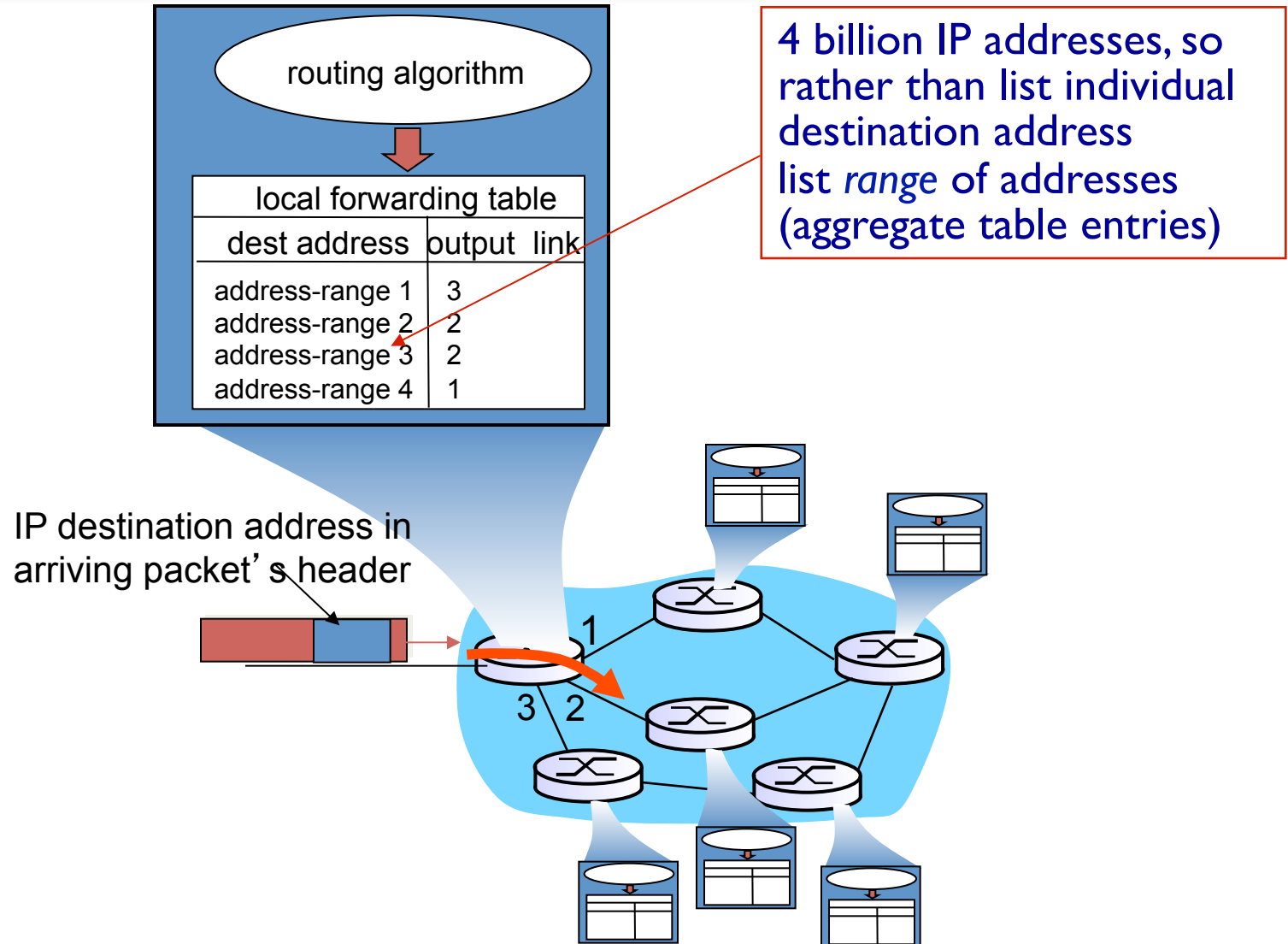
- Output link's transmission rate slower than switching fabric
  - queuing may occur at any of the output ports
  - overloading of individual output ports
- *queuing delay and loss due to output port buffer overflow*



Too much traffic on one output port



# Datagram forwarding table



# Forwarding table

IPv4 has 4 billion possible entries

## Examples:

<u>Destination Address Range</u>	<u>Output Link</u>
11001000 00010111 00010000 00000000 through 11001000 00010111 00010111 11111111	0
11001000 00010111 00011000 00000000 through 11001000 00010111 00011000 11111111	1
11001000 00010111 00011001 00000000 through 11001000 00010111 00011111 11111111	2
otherwise	3

# Longest prefix matching

## Forwarding Table

<u>Prefix Match</u>	<u>Link Interface</u>
11001000 00010111 00010	0
11001000 00010111 00011000	1
11001000 00010111 00011	2
otherwise	3

## Examples:

Destination address: 200.23.24.170

11001000 00010111 00011000 10101010

Which interface?

1

Destination address: 200.23.25.155

11001000 00010111 00011001 10011011

Which interface?

2

Destination address: 200.23.22.161

11001000 00010111 00010110 10100001

Which interface?

0

Destination address: 200.23.15.153

11001000 00010111 00001111 10011001

Which interface?

3

- routing, forwarding
- Router architecture
  - input/output ports
  - switching fabric
  - queuing (at input/output ports)
  - routing (forwarding) tables
- longest prefix matching