CS4450

Computer Networks: Architecture and Protocols

Lecture 2
Sharing Networks: “Circuits” and “Packets”

Rachit Agarwal
Announcements

• Please read course policies on the webpage.

• You should all be registered on Piazza by now.

• I do not expect you to read notes/slides before lecture

• We will decide on office hours in Feb first week
  • Your schedule will be clearer
Goal of Today’s Lecture

• Learn about:
  • Two important performance metrics:
    • Bandwidth
    • Delay, or latency
    • Why are these important?

• Two ways of sharing networks:
  • Circuit switching
  • Packet switching

• Why do current computer networks use packet switching?
But first, Recap from last lecture
Recap: what is a computer network?

A set of network elements connected together, that implement a set of protocols for the purpose of sharing resources at the end hosts

• Three important components:
  • Core infrastructure:
    • A set of network elements connected together
  • Protocols:
    • Needed to use the network
  • Purpose:
    • Sharing resources at the end hosts (computing devices)
What is a computer network?

A set of network elements connected together, that implement a set of protocols for the purpose of sharing resources at the end hosts.
Recap: what do computer networks do?

A computer network delivers data between the end points

• **One and only one task:** Delivering the data

• This delivery is done by:
  • Chopping the data into **packets**
  • Sending individual packets across the network
  • Reconstructing the data at the end points
Recap: what do computer networks look like?

Three Basic components

• **End hosts**: they send/receive packets

• **Switches/Routers**: they forward packets

• **Links**: connect end hosts to switches, and switches to each other
Let's make the picture simpler for today's lecture
What is a computer network?

A set of network elements connected together, that implement a set of protocols for the purpose of sharing resources at the end hosts.
What is a computer network?

A set of network elements connected together, that implement a set of protocols for the purpose of sharing resources at the end hosts.
A computer network can be abstractly represented as a graph.
Many mechanisms underneath!

- Locating the destination: Naming, addressing
- Finding a path to the destination: Routing
- Sending data to the destination: Forwarding
- Failures, reliability, etc.: Distributed routing and congestion control

Will take the entire course to learn these
A computer network can be abstractly represented as a graph.
Today’s focus: sharing the network (graph)
Today’s lecture: sharing computer networks

1. What does network sharing mean?

2. What are the performance metrics?

3. What are the various mechanisms for sharing networks?

4. Why “packets” and “flows”?

5. Understanding bandwidth and latency for packets
What does network sharing mean?
The problem of sharing networks

- Must support many “users” at the same time
- Each user wants to use the network (send and receive data)
- Limited resources

Fundamental question:
  - How does network decide which resource to allocate to which user at any given point of time?
What are the performance metrics?
Performance metrics in computer networks!

- **Bandwidth**: Number of bits sent per second (bits per second, or bps)
  - Depends on
    - Hardware
    - Network traffic conditions
    - ....

- **Delay**: Time for all bits to go from source to destination (seconds)
  - Depends on
    - Hardware
    - Distance
    - Traffic from other sources
    - ....

- Many other performance metrics (reliability, etc.)
  - We will come back to other metrics later ...
What are the various mechanisms for sharing networks?
Group Exercise 1:

How would you design a sharing mechanism?

Hint:
Think about sharing any resource (think of a road)
Two approaches to sharing networks

- Reservations
- On demand
Two approaches to sharing networks

- **First: Reservations**
  - Reserve bandwidth needed in advance
  - Set up *circuits* and send data over that circuit

- How much bandwidth to reserve?
  - Applications may generate data at rate varying over time
    - 100MB in first second
    - 10MB in second second ...
  - Must reserve for peak bandwidth (100MB)
Circuit switching: Implementing reservations since ...

Telephone networks

• One of the many approaches to implementing reservations

• **Mechanism:**
  • Source sends a reservation request for peak demand to destination
  • Switches/routers establish a “circuit”
  • Source sends data
  • Source sends a “teardown circuit” message
Circuit switching: an example (red request fails)

Source

Request = 100Mbps

Bandwidth = 100Mbps

Destination

Facebook

Google
Circuit switching: another example (red request succeeds)

Source

Request = 10Mbps

Bandwidth = 100Mbps

Destination

Request = 10Mbps

Destination
Circuit switching and failures

• Circuit is established

• Link fails along path (!!!!!!!)
  • First time we have seen failures making our life complicated.
  • Remember this moment.
  • Its gonna happen, over and over again.

• Must establish new circuit

Circuit switching doesn’t route around failures!!
Circuit switching summary

• **Goods:**
  • Predictable performance
  • Reliable delivery (assuming no failures)
  • Simple forwarding mechanism

• **Not-so-goods**
  • Handling failures
  • Resource underutilization
  • Blocked connections
  • Connection set up overheads
  • Per-connection state in switches (scalability problem)
Two approaches to sharing networks

• Second: On demand (also known as “best effort”)
  • Designed specifically for the Internet
  • Break data into packets
  • Send packets when you have them
  • Hope for the best …
Packet switching: an example
Packets

• Packets carry data (are bag of bits):
  • Header: meaningful to network (and network stack)
    • can be multiple headers
  • Body: meaningful only to application
  • More discussion in next lecture

• Body can be bits in a file, image, whatever
  • can have its own application “header”

• What information goes in the header?
What must headers contain to enable network functionality?

• Packets must describe where it should be sent
  • Requires an address for the destination host
    • can be multiple headers

• Packets must describe where its coming from
  • why?
    • Acknowledgments, etc.

• Thats the only way a router/switch can know what to do with the packet
Packet switching: what does a switch look like
Packet switching summary

• Goods:
  • Possible to gracefully handle failures
  • No resource underutilization
    • A source can send more if others don’t use resources
  • No blocked connection problem
  • No per-connection state
  • No set-up cost

• Not-so-goods:
  • Unpredictable bandwidth availability
  • Unpredictable delay/latency
  • Packet header overhead
Circuits vs packets

• Pros for circuits:
  • Better application performance (reserved bandwidth)
  • More predictable and understandable (w/o failures)

• Pros for packets:
  • Better resource utilization
  • Easier recovery from failures
  • Faster startup to first packet delivered
Summary of network sharing
Statistical multiplexing

- **Statistical multiplexing**: combining demands to share resources efficiently

- Long history in computer science
  - Processes on an OS (vs every process has own core)
  - Cloud computing (vs every one has own datacenter)

- Based on the premise that:
  - Peak of aggregate load is $\ll$ aggregate of peak load

- Therefore, it is better to share resources than to strictly partition them ...
Two approaches to sharing networks

Both embody statistical multiplexing

• Reservation: sharing at connection level
  • Resources shared between connections currently in system
  • Reserve the peak demand for a flow

• On-demand: sharing at packet level
  • Resources shared between packets currently in system
  • Resources given out on packet-by-packet basis
  • No reservation of resources
Understanding delay/latency
Packet Delay/Latency

• Consists of four components
  • Transmission delay (hardware properties)
  • Propagation delay (hardware properties, distance)
  • Queueing delay (traffic, switch internals)
  • Processing delay (end hosts)

• First, consider transmission and propagation delays

• Then queueing delay

• Ignore processing delays
Transmission delay

• How long does it take to push all the bits of a packet into a link?

• Packet size / Transmission rate of the link
  • Transmission rate = Share of Bandwidth

• Example:
  • Packet size = 1000Byte
  • Bandwidth = 100Mbps
  • $1000 \times 8 / 100 \times 1024 \times 1024$ seconds $\sim 76.3\text{us}$
Propagation delay

- How long does it take to move one bit from one end of a link to the other?

- Link length / Propagation speed of link
  - Propagation speed ~ some fraction of speed of light

- Example:
  - Length = 30,000 meters
  - Delay = $30 \times 1000 / 3 \times 100,000,000$ second = 100us
Group Exercise 2:
How long does it take for a packet on a link?

Constraints:
- Packet size = 1000Byte
- Rate = 100Mbps
- Length = 30,000m