Computer Networks

Lecture 1

Types of data flow

1. **Simplex**
- One device transmit while another device receive.
- Transmitter takes over capacity of channel.
2. **Half-duplex**
- Each device can transmit and receive but not at the same time.
- Current transmitter takes over capacity of channel.
3. **Full-duplex**
- Both devices can transmit and receive at the same time.
- Both devices share capacity of channel.

Types of connection

1. **Point-to-point connection**
- Provides a dedicated link between two devices.
- Capacity of channel is reserved for these two devices.
2. **Multipoint connection**
- Multiple devices share a link.
- They share capacity of channel either at the same time or take turn.

Physical topology

1. **Mesh topology**
- Each device has a dedicated point-to-point link to all other devices.
- N devices have n(n-1)/2 links.
- Each device has n-1 I/O ports.
- One unusable link doesn't stop transmission.
2. **Star topology**
- Each device has a dedicated point-to-point link only to a central controller/hub.
- The hub acts as an exchange between nodes.
3. **Bus topology**
- A multipoint connection; a long cable acts as backbone to link all devices.
- One fault can stop transmission.
4. **Ring topology**
- Each device has a dedicated point-to-point link only with two devices on either side of it.
- A signal is passed along the ring in one direction, from device to device, until it reaches the destination.
- One disabled station can stop transmission.

Protocol

- Set of rules governing data communication.
- **Syntax**: structure/format of data.

- **Semantics**: meaning of each portion of bits.
- **Timing**: when and how fast to send data.

Lecture 2

OSI model

1. **Physical layer** transmit bits across medium.
2. **Data link layer** deliver frames from node-to-node.
3. **Network layer** deliver packets from source-to-destination host.
4. **Transport layer** deliver message from process-to-process.
5. **Session layer** establish, manage, and terminate sessions.
6. **Presentation layer** translate, encrypt, and compress data.
7. **Application layer** provide services to user.

Addresses in TCP/IP (and OSI)

1. Host-to-network [**physical, data link**] – **physical address**.
- 48 bits (6 bytes) in 12 hex digit. E.g. 07:01:02:01:2C:4B.
2. Internet [**network**] – **logical address**.
- IPv4. 32 bits in binary or decimal.
3. **Transport** – **port address**.
- 16 bit in one decimal number. E.g. 753.
4. **Application** – **specific address**.
- E.g. abc@gmail.com.

Lecture 3

Analog signal

1. Have infinite number of values.
2. **Period**, $T = 1/f$. Time to complete one cycle.
- **Frequency**, $f = 1/T$ (in kHz). Number of period in one second.
- **Phase** is position of waveform relative to time zero.
3. **Composite signal** is the sum of sine waves with different frequency / phase / amplitude.
- **Spectrum of signal** is the sine waves that make up the signal.
4. **Bandwidth of medium**
- Range of frequencies that a medium can pass (Maximum frequency – Minimum frequency).
- Property of medium.
5. **Bandwidth of signal**
- Maximum frequency of signal – Minimum frequency of signal.
- Width of frequency spectrum of signal.

Digital signal

1. Have limited number of values.
- Is aperiodic. Composite signal with infinite frequency/bandwidth.

2. **Bit interval** is time to send one bit.
- **Bit rate** is number of bit intervals per second, bps.
- **Bit rate** and **bandwidth** are proportional to each other.

Data rate

1. Depends on bandwidth available, levels of signals can use and quality of channel (level of noise).
2. **Noiseless** channel
- **Nyquist bit rate** = 2 * bandwidth * log2L where L is number of signal level.
3. **Noisy** channel
- **Shanon capacity** (Max bit rate) = bandwidth * log2(1+SNR) where SNR is signal-to-noise ratio.

Transmission impairment

1. **Attenuation** is loss of energy through medium resistance.
- **Decibel (dB)** measures relative strengths of two signals or a signal at two points.
- dB = 10log10(P2/P1) where P1 and P2 are powers of signal at points 1 and 2.
- Positive if signal amplified, negative if signal attenuated.
2. **Distortion** is signal changes form.
- Occurs on composite signal which is made up of different frequencies.
- Each signal component has its own propagation speed thus has its own delay.
3. **Noise** is external energy corrupt signal.
- **Thermal** is random motion of electrons cause extra signal.
- **Induced** comes from motors/appliances.
- **Crosstalk** is the effect of one wire on other.
- **Impulse** is a spike comes from power lines, lightning etc.

Lecture 5

Types of error

1. **Single-bit error** is one bit in data unit has changed.
2. **Burst error** is multiple bits in data unit have changed.

Error detection

1. Uses the concept of **redundancy** which means adding extra bits to detect error.
2. **Parity check**
- Adds a redundant bit called **parity bit** to data unit so that total number of 1s is even.
- Can detect all single-bit errors.
- Can detect burst errors that have odd number of errors.
3. **Two-dimensional parity check**
- Adds a redundant data unit to n data units.
- A redundancy of n bits can detect a burst error of n bits.
- Cannot detect an error if two bits in two data units are damaged at the same positions.
4. **Cyclic Redundancy Check (CRC)**

- Adds a sequence of redundant bits called **CRC remainder** derived from binary division to data unit.
- CRC remainder must have one less bit than divisor. After being added, the data unit must be exactly divisible by divisor.
- At destination, incoming data unit is divided by the same divisor. A remainder means data unit is damaged.
- Can detect all burst errors that affect an odd number of bits.
- Can also detect all burst errors of length less than degree of polynomial.

Lecture 6

Data link layer

- **Flow control** is procedures to restrict amount of data that sender can send before waiting for acknowledgment (ACK); to prevent data congestion.
- **Error control** is methods to detect errors and retransmit frames based on ARQ.
- **Automatic Repeat Request (ARQ)** is retransmit specific frames if error is detected in an exchange.

Types of ARQ

(Those marked with – are general to ARQ.)

1. **Stop-and-Wait ARQ**

- Sender sends a frame and waits for an ACK from receiver before sending next frame.
- Data frames and ACK frames are numbered alternately. ACK number is number of next expected frame.
  E.g. data frame 0 is acknowledged by ACK1; receiver receive data frame 0 and expect data frame 1.
- Receiver will discard lost/damaged/out-of-order frame and not send ACK.
- Sender has control variable S = number of recently sent frame.
  Receiver has control variable R = number of next frame expected.
- Sender starts timer when sends a frame.
  If ACK is not received within allocated time period, sender assume frame is lost/damaged and resend it.
  Receiver send ACK if frame arrives correctly.
- **Piggybacking** is a method to combine a data frame with ACK to save bandwidth.

2. **Go-Back-N ARQ**

- Send multiple frames at the same time before receiving ACK.
  If got error, retransmission begins with last unacknowledged frame (even if subsequent frames have arrived correctly).
  Discard duplicate frames.
- Use **sliding windows**.
  m = number of bits for sequence number. Sequence numbers range from 0 to 2^m – 1, thus size of sender window is at most 2^(m–1). Sender window includes unsent frames.
  Size of receiver window is always 1 (looking for one specific sequence number).

3. **Selective-Repeat ARQ**

- Send multiple frames at the same time before receiving ACK.
  If got error, retransmit only unacknowledged frame.

- Use **sliding windows**.
  Size of sender and receiver window are at most one half of 2^m (thus receiver window is looking for a range of sequence numbers).
- Defines **negative acknowledgement (NAK)** that reports sequence number of damaged frame before timer expires.

HDLC

1. A protocol that implements ARQ mechanisms.
2. Supports communication over point-to-point or multiple-point links.
3. Provides two mode of communication which are NRM and ABM.
4. **Normal Response Mode (NRM)**
   - Station configuration is unbalanced.
   - One primary station send commands and multiple secondary stations respond.
   - Used for point-to-point and multiple-point links.
5. **Asynchronous Balanced Mode (ABM)**
   - Station configuration is balanced.
   - Each station functions as both primary and secondary.
   - Used for point-to-point link.
6. HDLC frames
   - Each frame may contain up to six fields: beginning flag, address, control, information (for I-frame and U-frame), frame check sequence (FCS), and ending flag.
   - **Information frame (I-frame)** transport user data and control information related to user data.
   - **Supervisory frame (S-frame)** transport control information.
   - **Unnumbered frame (U-frame)** transport information to manage link. Reserved for system management.

Lecture 7

Multiple access protocol

- When nodes use a common link called multipoint link, we need a **multiple access protocol** to coordinate access to the link.
- Random-access protocols and controlled-access protocols.

Random-access protocols

- Each station can send frame anytime. This may cause collision when multiple stations send at the same time.

1. **Multiple access** (MA) (ALOHA protocol)

- Station sends a frame when it has a frame to send.
- After sending frame, station waits for ACK.
- If doesn't receive ACK during allocated time, station assume frame is lost and resend after a random amount of time.

2. **Carrier sense multiple access (CSMA)**

- Station checks state of medium before sending.

- Collision still exists because of propagation delay; a medium may be idle because propagation by another station has not yet reached.

- **Persistence strategy**: procedures for station to sense a busy medium.
- **Nonpersistent strategy**: a station that has a frame to send senses the line.
  Sends if line is idle. Else wait a random period of time and senses again.
  Reduces collision because it is unlikely for stations to wait same amount of time and retry again simultaneously.
  Reduces network efficiency if medium is idle when there are stations that have frames to send.

3. Carrier sense multiple access with **collision detection (CSMA/CD)**

- Adds a procedure to CSMA algorithm to detect collision.
- Any station can send frame.
- The station monitors medium to see if transmission was successful.
- If collision occurs, station waits and resends the frame.
- Used in Ethernet.

4. Carrier sense multiple access with **collision avoidance (CSMA/CA)**

- Adds a procedure to CSMA algorithm to avoid collision.
- After line is idle, station waits an IFG (interframe gap) amount of time. Then waits a random amount of time. Finally, sends the frame.
- Used in wireless LANs.

Controlled-access protocols

- Stations consult each other to find which station has the right to send.
- A station can send only after authorised by other stations.

1. Reservation access

- Station needs to make reservation before sending data.
- Time is divided into intervals. In each interval, a reservation frame precedes the data frame sent in that interval.
- N reservation minislots in reservation frame for N stations. Each minislot belongs to a station.
  When a station needs to send data frame, it makes a reservation in its own minislot.
  The stations that made reservations can send data frames after reservation frame.

2. Polling

- **Primary** station controls the link; **secondary** stations follow its instructions.
- **Primary** station determines which station to use channel at a given time.
- **Polling**:
  If **primary** wants to receive data, it asks **secondaries** if they have anything to send.
  Secondary will respond with either NAK or a data frame.
  If NAK, **primary** polls the next **secondary** until it finds one with data to send.
  If data frame, **primary** reads the frame and return ACK.
- **Selecting**:
  If **primary** wants to send data, it tell **secondaries** to get ready to receive.
  Before sending data, **primary** creates and transmits a select (SEL) frame which includes address of intended **secondary**.

3. Token-passing method

- Station can send data when it has a special frame called **token**.
- Stations are arranged around a ring.

- Each station has a **predecessor** and a **successor**. Frames come from **predecessor** and go to **successor**.
- When no data are being sent, a **token** go around the ring.
- If station needs to send data, it waits for **token**. When the station gets **token**, it send frames and finally release **token** to be used by **successor** station.

Lecture 8

Connecting devices

- Connect LANs or segments of LANs.
- Operate at different layer of Internet model.

Repeater

- Operates in **physical** layer.
- Receives signal before it becomes corrupted and regenerates the original bit pattern.
- Since **amplifier** cannot differentiate intended signal and noise and it amplifies anything, a **repeater** must be placed to receive signal before noise change meaning of any bit.
- Has no filtering capability; forwards every frame.

Bridge

- Operates in **physical** and **data link** layers.
- As a **physical** level device, it regenerates signal it receives.
- As a **data link** device, it checks **physical (MAC) addresses** (source and destination) contained in the frame.
- Has filtering capability; checks destination address of frame and decides if frame should be forwarded to a port or dropped using **bridge table** that maps addresses to ports.

Forwarding table in bridge

1. **Static table** map addresses to ports manually.
2. **Dynamic table** map addresses to ports automatically.

- To do this, the **bridge** inspect both destination and source addresses.
- Use destination address for forwarding (table lookup).
- Use source address for adding entries to table and for update purpose.

STP (Spanning tree protocol)

- System admin adds redundant bridges (more than one bridge between a pair of LANs) to make system more reliable. If a bridge fails, another bridge takes over.
  However redundancy can create loops in system.

- **Bridges** use **STP** algorithm to create a loopless topology.
  In a bridged LAN, this creates a topology in which each LAN can be reached from any other LAN through one path only (no loop).
- STP process:
  1. Every bridge has a built-in ID. Elect bridge with **smallest ID** as **root bridge**.
  2. Mark one port of each bridge (except root bridge) as **root port**, the port with **least-cost** path from bridge to root bridge.
     Least-cost path may mean **minimum number of hops** (from bridge to LAN) or path with **minimum delay** or **maximum bandwidth**.
     If two ports have the same least-cost value, just chooses one.
  3. Choose a **designated bridge** for each LAN, which has **least-cost** path between LAN and root bridge.
     Mark the corresponding port that connects LAN to designated bridge as **designated port**.
     If two bridges have same least-cost value, chooses the one with **smallest ID**.
  4. Mark **root port** and **designated port** as **forwarding ports**, the others as **blocking ports**.
- The bridges send **BPDUs** (Bridge Protocol Data Units) to each other to update spanning tree when there is a change in network such as bridge failure, addition or deletion.

Wireless LANs

1. **Piconet**
- A Bluetooth network.
- Can have up to eight stations: one station as **master**, the rest as **slave**.
2. **Scatternet**
- Combination of multiple piconets.
- A **slave** station in one piconet can become **master** in another piconet.
- This station can receive messages from **master** in first piconet as a **slave**, and act as a **master** to deliver messages to **slaves** in second piconet.

Lecture 9

IPv4

1. 32 bits long. X.y.z.t/n where /n defines mask.
2. To find **first address** in block, set rightmost 32-n bits to 0s. Used as network address that represent organisation to rest of world.
- To find **last address**, set rightmost 32-n bits to 1s.
- Number of address = 2^(32-n).
3. Leftmost n bits (prefix) define network, rightmost 32-n bits define host.
4. **Network address translation (NAT)** allows private network to use a set of private addresses for internal communication and a set of global Internet addresses for external communication. Uses translation tables to route messages.

IPv6

1. 128 bits long. ABCD : ABCD : ABCD : ABCD : ABCD : ABCD : ABCD : ABCD.
2. Solve address depletion in IPv4.

ARP (Address Resolution Protocol)

1. Translate **logical** address to **physical** address.
2. IP packet contains physical and IP (logical) addresses of sender and IP address of receiver.
3. Since sender doesn't know physical address of receiver, an **ARP query** is broadcast over the network.
4. Every host on network receives the **ARP query** packet.
5. But only the intended recipient recognises its IP address and sends back an **ARP response** packet containing its **physical** address.

DHCP (Dynamic Host Configuration Protocol)

- Translate **physical** address to **logical** address.
- There are two scenarios in which a host knows it physical address, but needs to know its logical address.

1. A diskless station is just booted.
   The station can find its physical address by checking its interface, but it doesn't know its IP address.
2. An organisation doesn't have enough IP addresses to assign each station; it needs to assign IP addresses on demand.
   The station can send its physical address and ask for an IP address for a short time lease.

- Provides static and dynamic address allocation that can be manual or automatic.

1. **Static address allocation**:
   A DHCP **client** can request a **static** address from a DHCP server.
   A DHCP **server** has a database that **statically** binds **physical** addresses to **IP** addresses.
2. **Dynamic address allocation**:
   DHCP has a second database with a pool of available IP addresses.
   When a **client** requests for a temporary IP address, it assigns an available IP address for a negotiable period of time.

ICMP (Internet Control Message Protocol)

1. Report error message to original source and send query message.
2. **Error message** is reporting problem that a router/host at destination may encounter when it processes an IP packet.
3. **Query message** is helping a host to get specific information from a router/host.

Types of communication

1. **Unicast** communication: one source sends packet to one destination.
2. **Multicast** communication: one source sends packet to multiple destinations.

IGMP (Internet Group Management Protocol)

- For multicasting, we need multicast routers that can route multicast packets.

1. IGMP gives multicast routers information about membership status of hosts/routers connected to network.
2. Helps multicast routers create and update a list of loyal members related to each router interface.

Lecture 10

Types of delivery

- **Network** layer handles delivery of **packets**.

1. **Direct delivery**

- Receiver and sender are on the same network.

2. **Indirect delivery**

- Receiver and sender are not on the same network, thus **packets** go from router to router until it reaches receiver.

Forwarding

1. Places **packet** in its route to its destination.
2. Requires every host or router to have a **routing table** to route IP packets.
3. Host sends packet; router forwards packet.

Types of routing table

1. **Static routing table** contains manually entered information.
2. **Dynamic routing table** is updated automatically using dynamic protocols such as RIP, OSPF.

Types of routing protocol

- Rules and procedures that let routers inform each other of changes.
- **Autonomous system (AS)** is a group of networks and routers under authority of one administration.

1. **Intradomain routing**:

- Routing inside an AS.
- Each AS can choose one or more to handle routing inside AS.
- **Distance vector (RIP)** and **link state (OSPF)**.

2. **Interdomain routing**:

- Routing between ASs.
- Only one interdomain routing protocol handles routing between ASs.
- **Path vector (BGP)**.

Distance vector routing

1. **Least-cost** route between two nodes is a route with minimum distance.
2. Each node maintains a table containing minimum distances to other nodes.
3. The table also guides packet to desired note by showing next stop in route (**next-hop routing**).
4. Each node shares **routing table** with immediate neighbours periodically and when there is a change.

5. Updating routing table in distance vector routing
   - Receiving node modifies received table by adding cost between itself and sending node to each value in second column of received table.
   - Add name of sending node to each row of received table as third column (next-node column).
   - Receiving node compares each row of its old table with corresponding row of modified received table and chooses the row with smallest cost.
6. Is instable thus a network using this protocol can become unstable.

RIP (Routing Information Protocol)

- **Intradomain routing** protocol inside an AS.
- Implement **distance vector routing** with some considerations.

1. Routers have routing tables; networks do not.
2. Destination in routing table is a network; first column defines a network address.
3. Uses a metric called hop count. Metric is cost to deliver packet through a network.
4. Infinity is defined as 16, thus any route cannot have more than 15 hops.
5. Next-node column defines address of router to which the packet is sent to reach its destination.

Link state routing

1. Each node has entire topology which contains list of nodes and links, how they are connected including type, cost (metric) and condition of links (up or down).
2. Each node uses **Dijkstra's algorithm** to compute **shortest path** to other nodes and thus builds its routing table.
3. Routing table creation:
   - Each node creates link state packet which contains information about neighbour nodes and metric for each neighbour node.
   - **Flooding**: send this packet to all other routers.
   - Compute **shortest path** to each node using **Dijkstra's algorithm**.
   - Calculation of a routing table based on shortest path tree.

Lecture 11

Transport layer

1. Deliver **packet** from **process-to-process**.
2. Needs **socket address**, a combination of **IP address** and **port number**, at each end.

Types of communication mode

1. **Connectionless service**:
   - **Packets** are sent without need for connection establishment or release.
   - **Packets** are not numbered, they may be delayed/lost/out of order.
   - E.g. **UDP**.
2. **Connection-oriented service**:
   - Connection is first established between sender and receiver before data transfer.

- At the end, release connection.
- E.g. **TCP**.

UDP (User Datagram Protocol)

1. Unreliable **connectionless** transport protocol.
2. Provides process-to-process communication.

TCP (Transmission Control Protocol)

1. **Connection-oriented** protocol.
2. Creates a virtual connection between two TCPs to send data.
3. Uses **flow** and **error control**.
4. Bytes of data being transferred in each connection are **numbered**. Numbering starts with a randomly generated number.
   - Value in **sequence number** field of a segment defines number of first data byte contained in that segment.
   - Value in **acknowledgment** field in a segment defines number of next byte a party expects to receive. Acknowledgment number is cumulative.
5. Suppose a TCP connection is transferring a file of 5000 bytes. The first byte is numbered 10,001. What are the sequence numbers for each segment if data are sent in five segments, each carrying 1000 bytes?
   Answer:
   Segment 1 – Sequence Number: 10,001 (range: 10,001 to 11,000)
   Segment 2 – Sequence Number: 11,001 (range: 11,001 to 12,000)
   Segment 3 – Sequence Number: 12,001 (range: 12,001 to 13,000)
   Segment 4 – Sequence Number: 13,001 (range: 13,001 to 14,000)
   Segment 5 – Sequence Number: 14,001 (range: 14,001 to 15,000)

TCP connection

1. Create a virtual path between source and destination. All segments of a message are sent over this virtual path.
2. **TCP** transmits data in **full-duplex** mode and requires three phase: connection establishment, data transfer, and connection termination.
3. **Connection establishment** is called **three-way handshaking**.
4. After establish connection, bidirectional **data transfer** takes place. **Client** and **server** can both send data and acknowledgment.
5. Both **client** and **server** can **close connection**, usually by **client**.
6. Uses **sliding window** to make transmission efficient and control flow of data to prevent data congestion at destination.
   - Size of window = min(rwnd, cwnd) where rwnd = receiver window and cwnd = congestion window.
7. What is value of receiver window (rwnd) for host A if receiver, host B, has a buffer size of 5000 bytes and 1000 bytes of received and unprocessed data?
   Answer:
   Value of rwnd = 5000 – 1000 = 4000.

8. What is size of window if value of rwnd is 3000 bytes and value of cwnd is 3500 bytes?
   Answer:
   min(rwnd, cwnd) = 3000 bytes.

Lecture 12

Congestion

1. Occurs if packet load on network is greater than capacity of network.
2. **Congestion control** keeps load below capacity.
3. **Open-loop** congestion control is **prevention**; **closed-loop** congestion control is **removal**.

Back pressure

1. **Closed-loop** congestion control.
2. A **congested node** stops receiving data from immediate upstream node(s).
3. Node-node congestion control that starts with a node and propagates in opposite direction of data flow, to the source.
4. Node sends **choke packet** to source to inform it of congestion.
5. Warning is from a node to its upstream node; choke packet is from node to source directly.

Congestion control in TCP

- Sender has knowledge of receiver's window size (rwnd) and congestion window size (cwnd). Actual window size = min(rwnd, cwnd).
- TCP handles congestion based on three phases: slow-start, congestion avoidance and congestion detection.

1. **Slow-start phase**: size of congestion window **increase exponentially** until it reaches a threshold.
2. **Congestion avoidance phase**: size of congestion window **increase additively** until it detects a congestion.
3. **Congestion detection phase**: when congestion is detected, threshold value is dropped to one-half (**multiplicative decrease**).

- If detection is by time-out, sender goes back to **slow-start**.
- If detection is by three ACKs, sender goes back to **congestion avoidance**.

QoS techniques

1. **Resource reservation**
   - A flow data needs resources. QoS is improved if these resources are reserve beforehand.
   - Employ a QoS model called **Integrated Services** which depends heavily on resource reservation.
2. **Admission control**
   - Used by router/switch to accept/reject a flow based on **flow specifications**.
   - Before router accepts a flow for processing, it checks **flow specifications** to see if its capacity and its previous commitments to other flows can handle the new flow.