Network systems

Notes on the lectures

Week 3

Media, encoding and framing

There are three main types of physical media; electrical, with light and electromagnetic.

A coaxial cable (coax cable) consists of an inner wire for the signal, and three layers of shielding and insulation. A twisted pair cable prevents noise from an electric and/or magnetic field, by constantly twisting the positive and negative cables.

Optical fiber knows three forms:

- Step index fiber. Multiple lights are beamed in the fiber, all at a different angle. They end at the same angles at the end of the fiber.
- Graded index fiber. Multiple lights are beamed in the fiber, all at a different angle. Instead of bouncing against the walls, they rather circle around the middle like a sine wave.
- Single mode fiber. One light travels through the fiber.

Radio communication is communication using electromagnetic waves, propagating freely through the air.

Analog channels in general have a few properties:

- Their bandwidth is limited on the number of signal changes per second;
- The signal-to-noise ratio indicates how strong the signal is, compared to the noise;
- Possibly they have other characteristics, depending on the medium.

The channel's capacity is indicated by $C = B * \log_2(1 + \frac{S}{N})$, with C*apacity* in bits/s, B*andwidth* in Hz and S*ignal*/N*oise* ratio simply as a ratio.

To prevent errors in signal handling, different approaches exist.

- NRZ: Directly send the bits, with a high signal corresponding to a 1, and a low signal corresponding to a 0.
- Manchester: Transition the signal at every bit. Up if it's a 0, down if it's a 1.
- NRZI-M: Transition the signal if there's a 1.
- NRZI-S: Transition the signal if there's a 0.
- *X*B*Y*B encoding: Every X bits of data are encoded using predefined corresponding Y-bit strings.

To indicate the start or end of a packet, other techniques exist.

- Wait a long time between frames.

- Insert a bit / byte count at the beginning. Only possible if there's already a byte-stream. Doesn't withstand byte loss or corruption.
- Flags, with bit or byte stuffing. Reserves one of the possible bytes to indicate the start / end of a frame. Bit stuffing: agree on the flag to be 0111110, and agree on a system to detect genuine strings of 5 1s.
- Violating the physical layer. Do something that shouldn't occur, like a third signal level, in 4B5B-encoding: use on of the unused patterns, and with Manchester-encoding: use 00 or 11.
- Using very strict timing.

Medium Access Control

A communication network can be point-to-point, meaning there's one sender and one receiver, but also broadcast, where multiple devices share one link. Examples of point-to-point networks are ADSL, point-to-point ethernet links and Fiber to the Home. Examples of broadcast networks are old-fashioned Ethernet, cable Internet access and wireless (Wi-Fi, Bluetooth, GSM, 3G, LTE).

Medium Access Protocols have been designed to manage these networks. That's because, in a broadcast network, multiple transmissions may occur at a time, and that may cause collisions. MAPs determine how the nodes in a network share their channel. MAPs run in the network itself, they don't use a separate channel. Furthermore, not all nodes in a channel might be able to receive each other's messages, and they might not be aware of collisions happening in the network.

Channel partitioning

The channel is divided into smaller "pieces" (time slots, frequencies, code). Those pieces are allocated for exclusive use to one node.

TDMA (Time Division Multiple Access)

The channel access goes by rounds, where each node gets a fixed length time slot in each round. Unused slots stay idle.

FDMA (Frequency Division Multiple Access)

The channel's spectrum is divided into frequency bands, and each station gets assigned a fixed frequency band. The unused frequency bands stay idle.

CDMA (Code Division Multiple Access)

The data is XOR'd with a sequence of chips, with more than 1 chips per bit. The received signal is again XOR'd with the same code. If the codes are chosen properly, multiple nodes can use the channel simultaneously.

Because TDMA, FDMA and CDMA require setups, most implementations require that the clients communicate about this in a special channel for initial communication.

Taking turns

Here, the nodes take turns to send their data. Nodes that have more data to send, may take longer turns.

Polling
A master node can invite slaves to transmit in turn. This technique is used with dumb slave devices, like Bluetooth. The polling creates an overhead, and the master is a single point of failure.

Token passing
A control token is passed from one node to the next sequentially. The token is a single point of failure, and created an overhead.

Random access

The simplest of approaches: the channel isn't divided, the nodes simply random try to send their data and try to recover from collisions. The random access MAC protocols specify when to transmit, how to detect collisions and how to recover from them.

Slotted ALOHA
All frames in ALOHA are of the same size, and the time is divided into equal size slots. The nodes only start transmitting at the start of each slot, and they are synchronized. If 2 or more nodes transmit simultaneously, this is detected by all the nodes. If a node detects that its transmit has failed, it will try again in the next frame with a certain probability. The best transmission probability for slotted ALOHA is $\frac{1}{N}$, with N being the number of clients. As the number of clients approaches infinity, the maximum efficiency of slotted ALOHA approaches $\frac{1}{e}$.

In slotted ALOHA, a single active node can continuously transmit at the full rate of the channel. It's also highly decentralized; only the slots and the nodes need to be in sync. It's very simple. However, the collisions cause a waste of slots. Slots might also be idle.

Pure ALOHA
This is the same as aloha, except that it doesn't have synchronized time slots. This makes the probability of a collision higher. The maximum efficiency of pure ALOHA is very low; about 18%.

The best transmission probability for slotted ALOHA is $\frac{1}{(2N-1)}$, with N being the number of clients. As the number of clients approaches infinity, the maximum efficiency of slotted ALOHA approaches $\frac{1}{(2e)}$.

CSMA (Carrier Sense Multiple Access)
A fairly simple protocol, where a frame is only sent if the channel is sensed to be idle. Otherwise, it's withheld until the channel is idle. Two nodes might not be able to hear each other, and thus might not be able to detect an upcoming collision. If there is a collision, the entire packet transmission time is wasted, as CSMA clients do not care about collisions.

CSMA/CD (Carrier Sense Multiple Access with Collision Detection)
This is an extension of CSMA, where the clients do care about collisions and stop transmitting when they detect one. The collisions of CSMA are easily detected, thanks to the physical medium. This is

possible in wired LANs, but difficult in wireless LANs, because not all nodes may be able to hear each other.

CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance)
Another extension of CSMA, which prevents collisions rather than solving them after the fact. This is done by sending acknowledgements after a successful data transfer.

RTS / CTS (Request to Send / Clear to Send)
A protocol designed for wireless networks, where senders send a request to send message to their intended receiver, before starting the actual (big) transfer. The receiver then sends a clear to send message, indicating how long the sender may send. All other nodes receiving a RTS or CTS back off during the transmission. Collisions may still occur during the control messages, but the channel waste is significantly reduced.

Week 4
Switching and Basic Internetworking
To figure out where a packet needs to go, multiple approaches are possible.

Datagram
Every packet contains the complete destination address, and every router and/or switch knows the next hop for each destination. Every packet is also forwarded independently. All switches have a forwarding table, indicating which destination should go through which port.

Virtual circuit
All switches and routers have decided on virtual circuit identifiers (VCI), and have tables of incoming and outgoing VCIs. Virtual circuits can be predefined by the administrator (PVC, Permanent Virtual Circuit) or on-the-fly, by the source node (SVC, Switched Virtual Circuit). In the latter setup, signaling is used to set up the tables. These tables are populated when a packet travels a new, untraveled route (host A → host B, unknown route).

Source routing
In source routing, the route a packet should take is indicated in the headers of said packet.

	Datagram	Virtual Circuit Switching
Properties	• A host can send data anywhere at any time; • Complete destination and source address carried in the header of every packet; • Switch should be able to determine outgoing link for every address in the network; • Packets are switched independently, and may follow different routes; • In a large network, a routing algorithm is needed to fill a forwarding table;	• RTT delay before sending data; • Only VCI needed in header of data packet, source / data address only needed during setup; • Switch only needs entries in forwarding table for established VCs; • If a link or node fails, a new VC needs to be set up; • In a large network, a routing algorithm is needed to decide on outgoing link for set-up message;
Usage	• Ethernet switches; • IP routers;	• X.25, obsolete; • ATM (Asynchronous Transfer Mode), obsolete; • MPLS (Multi-Protocol Label Switching), used in core of the Internet;

Interconnecting Ethernet LANs

To interconnect LANs, a hub, bridge or ethernet switch may be used. A hub is a repeater for electronic signals. The connected LANs form a single “collision domain”. A bridge interconnects two LANs, and leaves them in their own “collision domain”. A bridge automatically fills its forwarding table, based on previously received frames. Finally, an ethernet switch is simply a bridge with many ports.

Learning bridge

A bridge learns, by learning the source address of incoming frames and using it to choose on which port to send outgoing frames. Upon receiving a packet, the source’s address and port are stored in the switch table. This is used later on to decide where a packet should go. If a destination is unknown in the switch table, the packet is simply forwarded to all ports on the bridge.

Spanning-tree algorithm

In the spanning-tree algorithm, the key concept is to automatically switch off some ports, ensuring no loops in the network are possible, while all nodes remain reachable. Every bridge has a unique identifier, and the bridge with the lowest ID becomes the root of the network. Until they learn otherwise, every bridge assumes its the root. The bridge occasionally sends out configuration messages, and other bridges relay these. The bridges use these to find the shortest path to the root and construct a spanning tree.

In the control messages, each bridge broadcasts its ID, what it thinks is the root, and the root path cost (RPC), the amount of hops between itself and what it thinks is the root. This information is sufficient for all bridges to figure out what the “real” root bridge will be. When the root bridge is undecided, i.e. two bridges have a different root, but equal RPC, the bridge with the lowest ID is chosen as the root.

Even after the root is decided, the bridges continue to send (if they're the root) or forward the configuration messages. When a bridge or link fails, the spanning-tree can be reconfigured.

Bridges only support interconnecting LANs with the same address format. Furthermore, it's not very scalable. First, the root bridge may become a bottleneck at high traffic. The forwarding tables may become large, and the routes may be sub-optimal. Finally, flooding the initial packet wastes resources.

Internetworking

An Internet is a collection of networks, interconnected to provide some sort of host-to-host packet delivery service. In *the* Internet, the Internet Protocol (IP) is used to accomplish this.

IP uses a global addressing scheme, which provides a way to identify all hosts in an Internet. IP uses datagram delivery, with best effort. Packets may be lost, delayed, delivered out of order or duplicated. However, it can use any underlying network (OSI layer model).

Because IP runs on any underlying network, it must respect that network's specification, most importantly the MTU (maximum transfer size). If an IP datagram's size exceeds the MTU of a link-level frame, the IP datagram is fragmented into smaller IP packets that do fit in the MTU. They are only reassembled at the final destination. Fragmentation is managed at the IP level using bits in the header.

IPv4 addresses are 32 bits, and generally displayed with a dotted decimal notation. To forward an IP datagram, the router reads the network part of the address. If the interface address and destination address are part of the same network, then the packet is immediately forwarded to the destination host. Otherwise, it's forwarded to the next-hop router, which is stored in the forwarding table.

IP networks are divided into class A, B, C and D networks. They consist of 16M, 65k and 254 hosts, respectively. Class D is reserved for multicast.

Apart from networks, another level in the IP hierarchy is used; subnetting. Here, the first *n* bits of an IP address indicate which subnet the IP address is part of. For example, 192.168.1.1 is part of the 192.168 subnet. This is indicated with 192.168.0.0/16, or 192.168.0.0 subnet mask 255.255.0.0. The subnet numbers and masks are then stored in the aforementioned forwarding table. This is the CIDR (Classless InterDomain Routing). IP reservation is done with CIDR in favor of network numbers nowadays, as it minimizes the amount of necessary entries in forwarding tables. If a packet's destination address matches multiple entries, the most specific one, i.e. the one with the longest prefix (/24 over /20) must be chosen.

Internetworking and Link State Routing

DHCP (Dynamic Host Configuration Protocol) is a protocol that allows hosts to dynamically obtain their IP addresses from the network server, whenever it joins the network. It can renew it's lease on the address in use, and it allows the reuse of older addresses. A DHCP client only keeps its address while it's connected.

DHCP works quite simple. The client broadcasts a "DHCP discover" packet, to which the DHCP server responds with "DHCP offer". The client requests an offer by sending a "DHCP request", and the server acknowledges the request with "DHCP ack".

DHCP is a flexible protocol, allowing more than one IP address to be returned. The DHCP server might inform the client about the address of the first-hop router, the name and IP address of the DNS server, and the network mask of the network it's connecting to.

Hosts on the Internet are identified by their MAC address on the hardware level, and an IP address is used for internetworking. Hosts might have a domain name, which is a human-readable name for an IP address. Resolving domain names to IP addresses is done using DNS (Domain Name System), while MAC addresses can be resolved from IP addresses using ARP (Address Resolution Protocol).

Both the link-layer and network-layer addressing schemes are used when communicating with hosts in another LAN. A host might create an IP packet going to a certain host outside its network, encapsulated in a link-layer frame going to the next hop in the route to the destination. Between the source and the destination, the link-layer frame is altered, while the IP frame stays intact.

ARP uses a cache of known IP – MAC combinations, called the ARP table. It may be integrated with the forwarding table. The ARP table contains mappings consisting of the IP address, the MAC address and the time to live (the time after which the entry may be considered invalid).

If a host wants to send a datagram to a certain IP address, but does not know the MAC address of the destination, it sends an ARP query packet to everyone in the network. The destination then replies with its MAC address, using unicast. The source then caches this information in the ARP table.

DNS is a distributed database, implemented in a hierarchy of many name servers. It is an application-layer protocol, and it is not essential for communication between hosts. It exists, simply to help humans find certain IP addresses on a network. It works like a phone book: you can remember everybody's phone number, but it is easier (for humans) to look them up by name.

DNS is decentralized, because otherwise it would not scale. There would be a single point of failure, which would have to handle an amazing amount of traffic.

DNS records consist of a name, a value, a type, a class and a time to live. An A-type record indicates that the human-friendly domain name *name* resolves to the IP address *value*. A CNAME record indicates an alias, where the domain name *name* is actually the domain name *value*. An NS record indicates the authoritative name server of a domain, indicated by the *value*. Finally, an MX-record states the mail server *value* for a certain domain *name*.

The DNS system consists of the root DNS servers, followed by top-level domain name servers (like the servers for .com, .org, .nl), followed by the name servers for separate domain names. This hierarchy may continue downward. A DNS request starts out at the lowest tier of the DNS system. If a local DNS server cannot resolve a domain name, it contacts its parent DNS server, and so on. The root DNS servers contain very few information; they only indicate where to find the servers associated with top-level domains like .com and .nl.

Every DNS server caches previously requested records from their parents, ensuring that the root servers are not flooded with requests. The time to live of a DNS record decides how long a child DNS server should keep the information. In practice, a lot of servers higher up in the hierarchy don't look up the domain name that the child requested, but rather inform the client of a server that might now the answer.

Routing

The big difference between routing and forwarding, is that routing algorithms figure out an end-to-end path between two hosts, and forwarding tables figure out the next step in a route.

Every network can be seen as a graph, with hosts as nodes, and the links between them as edges. The challenge of a routing algorithm is to find the path between any two nodes with the lowest cost, where the cost is the sum of the costs of all the links that make up the path. Sometimes this means that a route via one or more other hosts might be a better idea than directly connecting to the host, even if the source and destination are directly connected. Every node in a network is able to learn who its neighbors are, and what the costs are of the links to its neighbors.

Link State Routing

This routing algorithm floods the (local) link-state information of a node to all other nodes. A local algorithm is then used to calculate the route to all destinations. Every node creates a link state packet (LSP), with the ID of the node creating it, the cost of the link to each directly connected neighbor, and the time-to-live of the LSP, uniquely identified by the sequence number. It recreates such an LSP with an increased sequence number whenever the TTL expires, if the link with a neighbor has disappeared, or if the link has (dis)appeared.

Those LSPs are flooded to all nodes in the network, using acknowledgements and retransmissions where needed. Nodes keep the most recent version of an LSP for every node. They forward their stored, valid LSPs to every neighbor except the origin. The TTL of an LSP is decreased on every forward, dropping it when the TTL becomes 0.