Contents

Networks CS348 Notes

August 28, 2024

Lecture 1

Part I

General Ideas

1 Introduction

What is a computer network? A computer network is a group of interconnected devices that can exchange data and resources with each other. ~~Almost~~ all of today's devices are in one way or another connect the biggest computer network alias the Internet. There are several abstractions and nuances that enable the existence of such a huge structure.

A network consists of several end hosts which are systems that request/receive data using the network. These end hosts are connected using links which can directly connect the hosts together or more commonly connect multiple of them to switches/routers which can simplify the network while still providing connectivity among hosts.

1.1 Topologies

A group of hosts can be connected in multiple ways. The type of graph that is obtained from considering the hosts as nodes and links as edges is called the 'topology' of that network. Some examples include a bus where all hosts are connected to a common wire. Others include star topology where hosts are connected to a central host.

Links can also be classified on the basis of how many users can communicate across them.

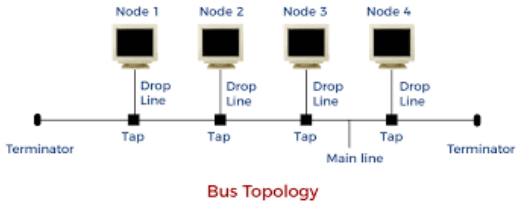


Figure 1: Bus Topology

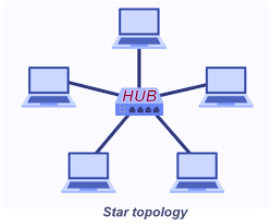


Figure 2: Star Topology

- **Simplex:** Only one user can talk across a link
- **Duplex:** Both users can communication *simultaneously* across a network.
- **Half Duplex:** Both users use the same link to communication but not simultaneously.

1.2 Switch

What exactly is a switch? The Switch is a network device that is used to segment the networks into different subnetworks called subnets. It can help simplify a network by grouping together lots of hosts into a sub-network. A switch has multiple incoming and outgoing links. It is capable of routing data from an incoming link to an appropriate outgoing link.

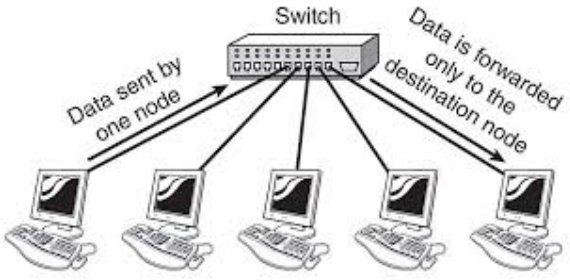


Figure 3: Example of a switch

2 Abstractions, Layering in a network

There is always the possibility to deal with the network as a single structure at once. That is to deal with the entire flow of data from the second a request is made by a 'user' and all the way till the request is serviced in one go.

However, this is very inconvenient and complicates the network in the sense that any change made to some part of the network can make or break the entire system. To

combat this the network is clearly split into layers where each layer operates relatively independently and only exposes parts of it that are necessary for the higher and lower layers in the network.

To better understand this let us take an example.

The most common request on the Internet is an HTTP request (ie) a request by a computer for a webpage. Let us see the flow of information when such a request is made.

1. **Application layer:** The URL is entered into a browser and then a user request is made. Then this URL is converted into the **IP**[1] address of the server that holds the page needed by the user.
2. **Transmission layer:** Now that we know the IP address from the application layer, the request for a page is sent to that address using the transmission layer. This layer sends the request message in manageable pieces to the network to be sent to the web server.
3. **Network layer:** Now these 'manageable pieces' need to be sent to the destination (ie) web server. The next router/link to which the message is to be sent is decided in this layer. These links 'talk' to each other in some sense and know where to send messages to reach the web server.
4. **Data Link layer:** The data link layer deals with splitting the message bit by bit and choosing the appropriate media to transfer them using (ie) Optic fiber, Wireless links etc..
5. **Physical layer:** Finally the physical layer deals with transmitting the actual bit signals over whatever media is chosen.

Now once the data is transferred to the physical layer of the web server, it climbs up in the layers till the application layer of the web server is reached. The response of the web server is transmitted back similarly. This by no means completely covers the functionality of each layer but rather gives a flavour of each layer's functions. It is easy to see how the abstraction is helpful as now the application layer has no need to worry about which media is used to transfer the bits and the Physical layer is oblivious to what message it is transferring.

The abstraction helps to simplify the structure of the network by helping us deal with one subproblem at a time.

Lecture 2

[1] will be dealt with later, assume it is some id for a computer

2.1 IP address

Each connected device has a unique identifier to describe it. This identifier is known as the IP address of a device.

The IP address has a hierarchical structure. An example of an IP address is '72.85.5.25'. This can be thought of as being similar to a postal address where the country of your address is the highest level at which location is specified. After this the state, city, area narrow down your location more and the message travels in an organised way from one 'level' to another.

In fact as discussed before each router at the Network layer transmits a message to a particular router which is chosen based on its IP address and the IP address of the destination. How is this rerouting done?

- **Readjusting weights:** The weights of each connection can be adjusted to change the shortest path to the destination
- **Longest Prefix Match (Practical):** There is a router table present which gives us the IP corresponding to a router. When a packet has to make the choice between routers it takes the router connected to the current router with the longest prefix match when compared to the destination address.

2.2 Message Granularity, Delays

The size of the message which is being sent via the network changes depending on which layer the transfer happens in. That is a message can be described in various granularities.

- **Application layer**: Application dependent, a video/ a webpage etc..
- **Transmission layer**: TCP splits the message into segments, UDP splits it into Datagrams
- **Network layer:** Transfers data as packets
- **Data Link layer:** Symbols
- **Physical layer:** Bit by Bit

As we saw in the last Lecture a router has some number of 'in' connections and 'out' connections which are connected together depending on which 'out' connection the message is supposed to be sent. This 'routing' is done by putting each incoming packets on queues corresponding to the outgoing connection we are supposed to send to.

This rerouting is done so that if the incoming rate to an out connection is greater than its outgoing rate we have the queue as a buffer.

Why do we just not make the queue very large to prevent these 'drop offs'

- **Cost:** Memory is not free so we need to be aware of the trade offs of increasing the queue size

- **End-End delay:** If the router has a buffer of say size Q_{max} and an output rate of c $bits/sec$. Now if the queue is almost full and we get a new packet put in at the very end, the packet takes $\frac{Q_{max}}{c}$ time to be put on the next connection. This is called the End-End delay and clearly a large queue increasing the worst case End-End delay

The total delay for a transmission of a packet through some K routers would be

$$Delay = \sum_{k=1}^{k=K} \frac{Q_{max}^k}{c^k} + S_d + T_d$$

S_d is called the speed of light delay and T_d is the transmission delay. In most applications End-End delay is the significant bottle neck for the whole delay. Infact in some applications we prefer dropping packets inorder to not have high delays[2]

2.3 Layering and Design Protocols

Any subproblem is handled by some protocol corresponding to the layer we are at.

We have divided networks into 5 layers. Specifically Application layer, Transmission Layer, Network Layer, Data Link Layer , Physical Layer. This abstraction enables users to interact only with the layer they are concerned with in that layer without having to deal with the network as a whole.

Some advantages of layering networks are:

- **Ease of development:** Only certain problems need to be dealt with at each layer
- **Debugging:** Ease in fixing new problems in each layer independently
- **Flexibility of Physical technologies, Applications:** As an example whatsapp as an application only deals with that layer of the network. It doesn't interfere/have to deal with the particular intricacies of the physical technology used by their users to connect to the network.
- **Ease of Modification:** We need to change only a particular layer to address problems associated with it. There is no fear of breaking the system due to modifications made to said layer.
- **Choices at each layer:** Each layer can use multiple media without breaking compatability with the system.

Lecture 3

Disadvantages of layering networks:

[2]Think of voice call where a delay would lead to not being able to communicate anything

1. There is some opaqueness about other layers. As an example let's say a packet is sent from a source to destination using a sequence of routers. If a packet is dropped midway, TCP makes an assumption that they were dropped due to a full queue. Infact there can even be wrong assumptions that a packet was dropped[3]. This mainly arises from the fact that the routers have little to no communication going with the protocol of a higher layer.
2. There is redundancy at each level of the network. **TCP** handles retransmission, but even **MAC** handles that. Let's say that a packet transmission attempt in a wireless link failed. The MAC will make sure that retransmission happens. But this is also taken care by TCP which is redundant.
3. Transmission is suboptimal. The sender of the message is unable to specify guarantees they want in the delivery of said packets. This is referred to as a 'Best Effort' system where the network has no guarantees regarding the quality of service

2.4 Latency Metrics

1. **One-Way delay:** If a packet is sent out at time t_0 and it reaches the destination at t_1, then the *one way delay* of the packet is $t_1 - t_0$. However measuring one day delay is difficult since it takes the direct difference in times measured at the source and destination. This difference may drift apart over time due to both systems operating at different clock cycles.
2. **Round-Trip Time:** Once a packet is recieved by the target it sends back an acknowledgment message. The time taken from sending the messages to recieving the acknowledgment is its *round trip time*.
3. **Jitter:** Jitter measures the variability in the latencies associated with the sending some k packets. Let jitter be J. It can be written as

$$e_k = |d_{k+1} - d_k|$$
$$J = \frac{1}{n-1} \sum_{k=1}^{n} e_k$$

Lecture 4

2.5 Headers

There is some meta data about each packet/module of data generated when a message is sent from one layer to the below layer. This metadata is added as a header to the

[3]example for another reason is interference in wireless links

message itself that is shared with the next layer.

When a message goes to another layer below this header is not tampered with at all. Rather the next header is just layered on like an onion on top of the previous header.

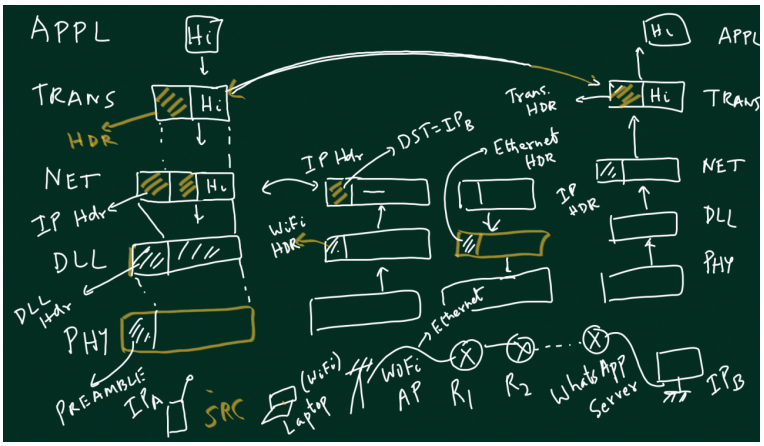


Figure 4: How headers are built and removed

3 Quality of service

There are some metrics to determine the quality of the service provided by the network.

- **Latency:** Delay from the dispatch of a packet till it reaches destination
- **Throughput:** Amount of data that can be sent in given time
- **Bandwidth:** Amount of data that can be sent at the same time in network

Part II

Physical Layer

1 Physical Media

1. **Twisted pair cables:** It consits of a pair of cables twisted together (to cancel out magnetic fields created by loops). The larger the number of cables and the more the twisting the better it is.
   There are different categories of cable each having its own data transmission rates.
2. **Co-axial Medium:** Co-axial cables consist of a series of wires covered by a wire mesh to deal with magnetic fields. Depending on the thickness of the mesh data transmission rates vary.

3. **Optic Fibre:**
   - Single Mode: The optical fiber is so thin that only a single ray of light can cleanly pass through it
   - Multiple Mode: A thicker fiber which is capable of transferring multiple rays at the same time

   Which of those is better? One may think it is multimode since it can transmit multiple pulses together. However the rays in the multimode gets interferred with each other if they are placed closer together.
   Thus there is a need to delay the throughput in multimode inorder to make sure the signals received are coherent.
   Overall single mode turns out to be better.

Lecture 5

1.1 Attenuation

Attenuation is the loss of power when it is transmitted over a Physical media.

$$\text{Attenuation} = 10\log(\frac{P_{in}}{P_{out}})$$

Attenuation has units as dB/decibels.

Some examples on calculating Attenuation:

1. $$\frac{P_{in}}{P_{out}} = 2$$

   $$\begin{aligned}\text{Attenuation} &= 10\log_{10} 2 \\ &= 3dB\end{aligned}$$

2. There are two identical wires which cause an Attenuation of 3dB. The wires are connected and a signal is passed through the combined wire. Assume P1 is the power left after the signal passed through one of the wires

   $$10\log_{10}\frac{P_{in}}{P_1} + 10\log_{10}\frac{P_1}{P_{out}} = 10\log_{10}\frac{P_{in}}{P_{out}}$$

   Thus total attenuation is the sum of attenuation of both wires.

Another thing to be noted is that power is directly proportional to the square of the amplitude of a signal. Thus Attenuation can also be expressed as:

$$\begin{aligned}\text{Attenuation} &= 10 \log_{10} \frac{(A_{in})^2}{(A_{out})^2} \\ &= 20 \log_{10} \frac{A_{in}}{A_{out}}\end{aligned}$$

1.2 Absolute Power in decibel scale

1 mW(milli Watt) is kept as the reference to express absolute power in the decibel scale. That is power of P Watts can be written as $10 \log_{10} \frac{P}{10^{-3}}$.

Absolute power has no significance to describe the quality of a transmission on its own. What does matter is Recieved power relative to Noise power.

1.3 Frequency vs attenuation

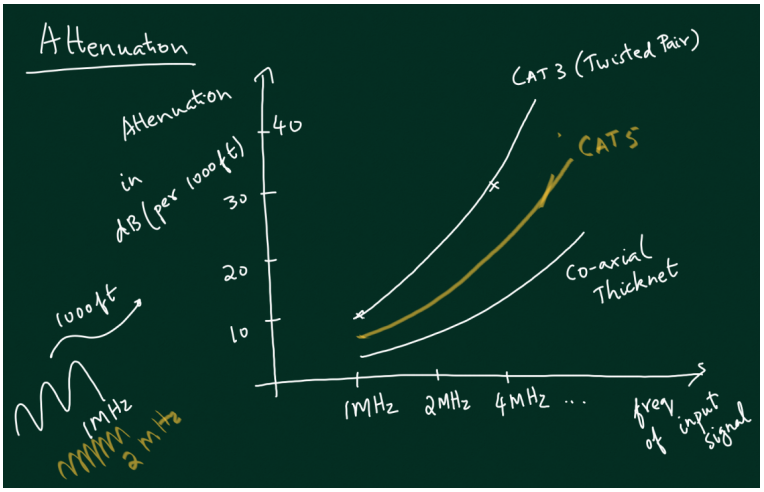


Figure 5: Attenuation vs Length of cable

Clearly, attenuation increases with the frequency of the signal sent through it. This does not make sense considering the wire to have only resistance. Thus it is clear that the wire also has some inductance associated to make this behaviour happen.

However, the situation is a bit different for optical fibres.

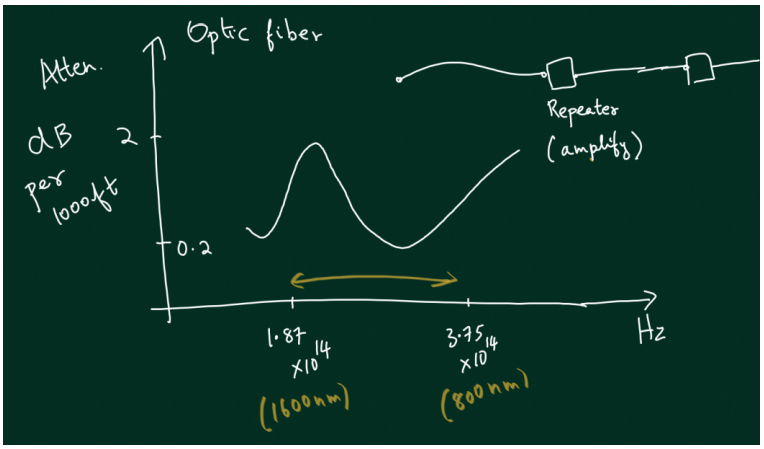


Figure 6: Attenuation vs Length of optic fibre

Here the attenuation shows non-linear behaviour which suggest some capacitative behaviour along with the inductance.

To prevent attenuation from decreasing signal quality, repeaters are used to boost up the signal. They are placed at calculative distances to maximise their advantage.

Different signals can be sent across the same channel using a different Frequency. But the speed at which data can be sent decreases as the frequency range of different messages or **Bandwidth** increases. This can be calculated using Claude Shanon's definition of entropy.

1.4 Attenuation in Wireless signals

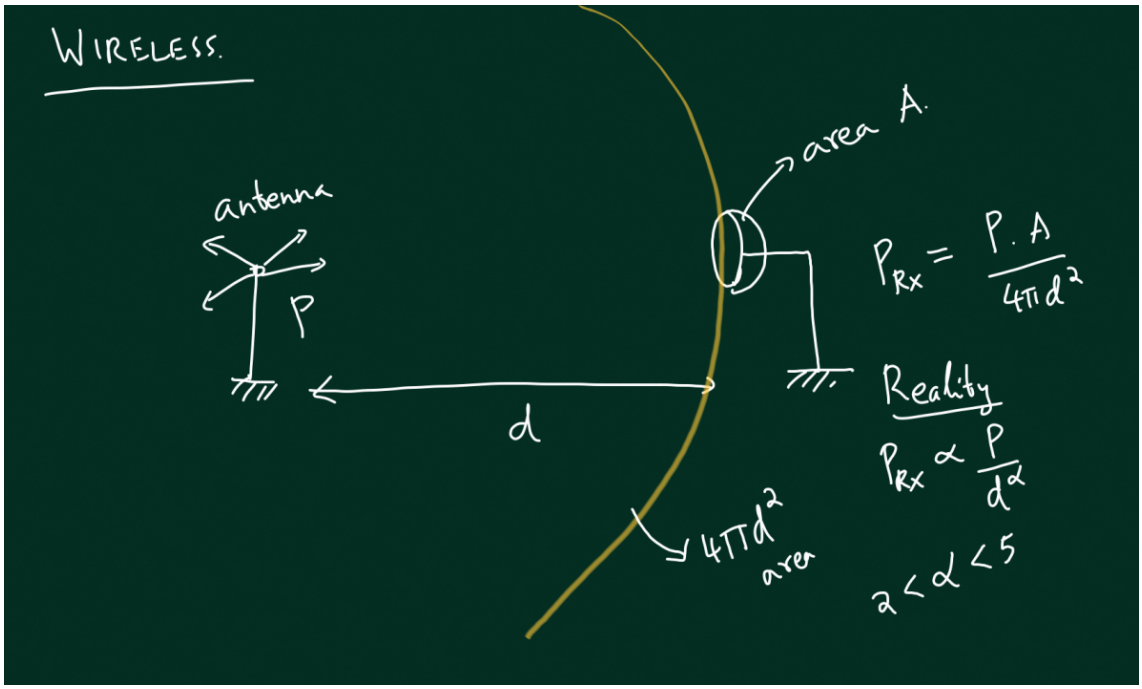


Figure 7: Attenuation in wireless communication

The power obtained depends on the area it is recieved from and the area it has spread over. Power obtained (ie) P_{Rx} can be written as

$$P_{Rx} = \frac{P.A}{4\pi d^2}$$

In reality the equation turns out to be proportional to $\frac{P}{d^\alpha}$ where alpha can vary from 2 to 5.

Why is $\alpha > 2$? It is due to interference and diffraction. If a big object obstructs a wave it may bend around the obstacle, so it is non-trivial to note where signals will be weak. Similarly when waves take longer paths and reflect off surfaces to reach a location they can interfere destructively to decrease signal power further (Multi-path).

This is why there are random locations with good signals and others very close by with bad signals.

One point is to note is that due to interference the attenuation in wireless transmission is much much more than wired transmissions. Thus, some frequency bands are licensed by different service providers and it is agreed that they will use those frequencies for transmission.

What about WiFi? Well, there are some bands which are unlicensed and can be used without any premium. However, here again we have to worry about interference.

1.4.1 MIMO

Multi Input Multi Output (MIMO) is a larger tower with several antennas to transmit signals. Why do many different antennas help? Virtue of having several antennas different signals can be sent on the antennas to strategically make the signals interfere to only send a beam in a particular direction. Thus it uses multi-path communication to its advantage.

2 Signalling

Now signals have to formualted and transmitted bit by bit. There are two ways to formulate signals bit by bit. Assume a '1' is described by +5V and '0' is described by -5V.

- **Non-Return to Zero:** To send a signal like '101', +5v and then -5v and then +5v is sent one after another
- **Return to Zero:** To send a signal like '101', +5v and then the signal 'returns' to 0, then -5v and again returns to zero after some time and then +5v is sent.

What are some of the issues with Non-return to zero?

- The number of bits sent can only be calculated using the time interval of the signal which is needed for one bit. The issue is that the clocks of the sender and the reciever need not be in sync.
  So the receiver may percieve a different signal.
  This problem is rectified by *Return to zero* since the wave form indicates the number of bits sent. The tradeoff here is that a higher frequency is needed to send that waveform which implies higher attenuation.
- Another issue is Baseline wander. When a signal is amplified, non-ideally a DC-offset is induced in the signal. If the offset is severe enough along with the noise then a '0' could be mistaken to be a 1.
  To deal with this a High-Pass filter is used. A **HPF** removes all low frequency signals which includes the DC offset. Infact since only a particular provider's signals are to be recieved, a bandpass filter is used to filter out signals not falling in the desired range.

Apart from this a big issue is that regardless of the method used, lets say that a sequence of 1s is sent as a signal. Now the average signal sent becomes 1 and as the signal gets amplified an offset is created. Even if this is corrected by removing the offset now the entire signal is lost.

Thus, to prevent this the average of the signal sent must be zero.

Too many issues phew :/

How to deal with all this?

2.1 Manchester coding

Encode a bit differently depending on the signal. Take an xor of the clock and the bit to be sent.

- There is a signal transition for every bit period
- The average signal per bit period is 0

Data	Clock	Encoding
0	0	0
1	0	1
1	1	0
0	1	1

Table 1: Manchester Encoding

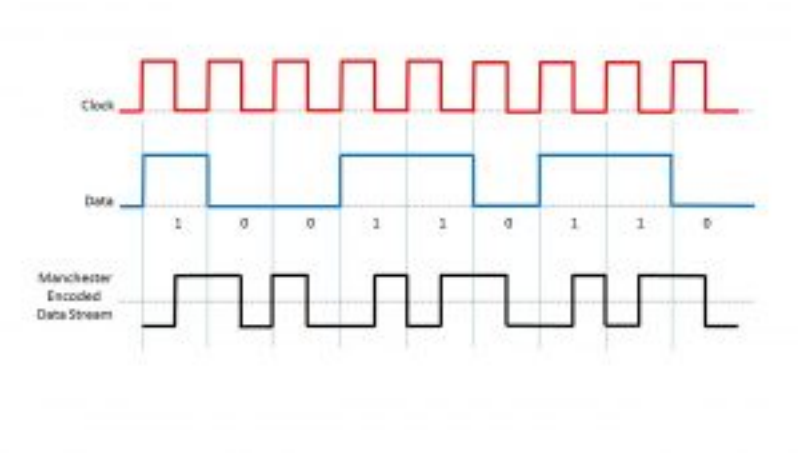


Figure 8: Manchester Encoding

How to determine the polarity of the signal sent if the voltages are switched. The last few bits of the preamble can be dedicated to determining the polarity of the signal. Say they are 111, if 111 is received the polarity would be correct and else polarity needs to be flipped.

However, bit-errors can make this scheme breakdown quickly.

2.2 Differential Manchester Coding

Another more error-free option is to encode the messages with Differential Manchester encoding. Rules:

1. **0 bit:** The voltage in the first half of the time period is different from the voltage in the second half of the **Previous** time period.
2. **1 bit:** The voltage in the first half of the time period is same as the voltage in the second half of the **Previous** time period.

3 Phase Modulation in wireless channels

Wireless signals can be used to represent bits in several ways. Frequency, amplitude modulation (ie) encoding the message in values of frequency, amplitude is common.

However, another popular option is to use phase modulation, (ie) the message is put into the phase of the signal that is sent.

Unrelated side track: Since each company has been allocated a band of frequencies available to them. This implies that the fourier transform of the signal sent almost completely lies within the appropriate frequency range.

Assume that f_0 is the centre of this band and the 'Bandwidth' is 2Δ. How do we make sure the signal's recieved has a frequency decomposition lying only in the given range? One option is to use a bandpass filter at the receiver end to obtain only the signals in some range.

Take a signal like $s(t) = A\cos(2\pi ft + \phi)$. The signal recieved would be $\alpha s(t-\delta) + n(t)$. α is due to attenuation, δ is the propogation delay, n(t) is white noise[4](after bandpass filter).

Side track over.

3.1 Signals as vector spaces

Signals can actually form a vector space. Each signal is a 'vector' offset from the axis corresponding to the phase and having length proportional to amplitude.

The angle between these vectors can be found using their inner product

$$< a(t), b(t) >= \int_0^T a(t)b(t)\partial t$$

This can be represented as a 2-d diagram where one axis represents the sin component

[4]White noise is a signal that has energy at all frequencies and cannot be filtered out

of the wave and another the cos component. A phase shifted wave can be represented as a combination of sin, cos wave. The length of the vector in this domain is proportional to amplitude.

This is called the constellation diagram.

How to get the constellation diagram for a given signal? We can dot the given signal with e_x and e_y (ie) unit normal vectors of this vector space to get each component of the wave.

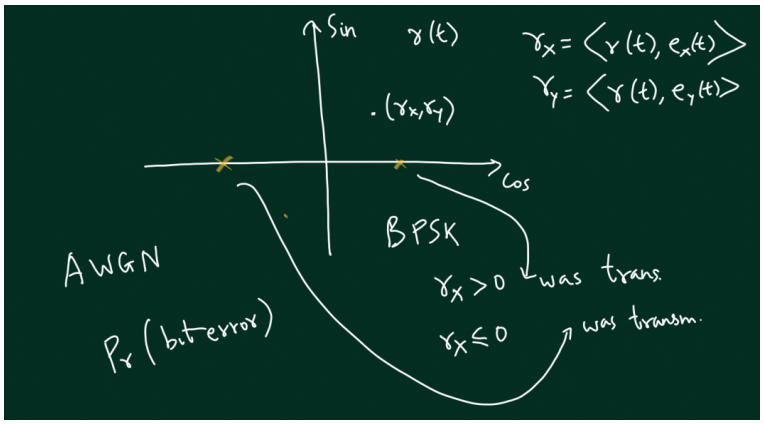
Thus given a signal with a phase ϕ its 2d representation can be found to be put on the constellation diagram.

3.2 BPSK encoding

Now how do we encode bits using the constellation diagram?

One option is to allocate a region in the 2-d vector space for '1' and another for '0'

Assume that the +x axis points denote signals for bit as '1' and -x axis points denote signals for bit as '0'.



This is called Binary Phase Shift Keying. Note that even with attenuation and white noise the amount the received signal shifts in a constellation diagram is lesser and thus unlikely to cause a zero to be interpreted as a 1.

3.3 QPSK encoding

The issue with BSPK is that a single signal/wave can only transmit one bit of information. If we can partition the 2-d plane into say 4 regions and use a similar logic to now assign a 2-bit encoding to each region, we can double our throughput.

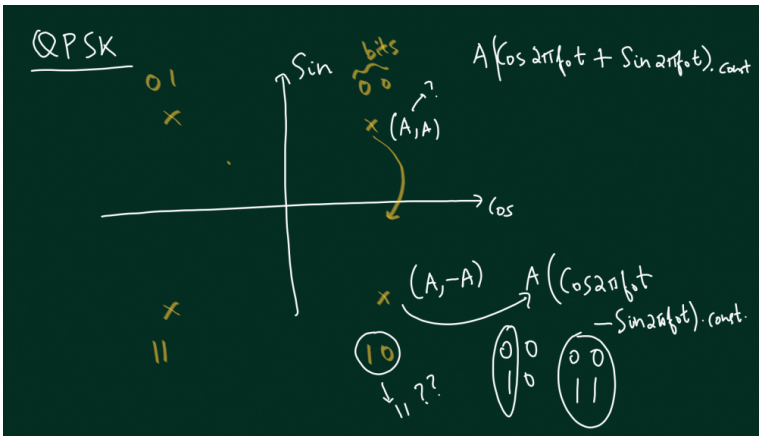


Figure 9: Diagram to show bit encoding of each region

This is called Quadrature Phase Shift Keying. One point to be noted is that while assigning bits to region it is preferable to assign encoding such that strings which differ in both bits are diagonally opposite regions so that even if a misinterpretation does happen its more likely to be a one bit error.

This idea of more regions being able to encode more bits can be expanded more however the more regions it is split into, the more error prone the encoding becomes due the regions being more packed.

Part III

Data Link Layer

1 Introduction

The Data link layer deals with preparing a packet to be sent across the physical medium and also with error correction in received messages.

Its exact functionalities are:

- **Framing:** This deals with packaging packets into units called frames by adding headers with the destination MAC address and also some other information, along with redundancies to detect errors.
- **Error Detection and correction:** When a message is sent there are definite error imposed due to environment. This layer deals with detecting and correcting such errors by adding some redundancy in these messages.
- **Medium Access:** Deals with managing the shared medium, prevent collisions

(where two devices transmit simultaneously), and ensuring fair access for all devices on the network.

1.1 Bit stuffing

Some networks keep transmitting data in the order of frame, sequence, frame.... A sequence is a filer message sent between two frames. It is the bitstring '01111110'.

The receiver knows to ignore sequences. However, now the issue is what if an actual message to be sent has some substring of the from '01111110'. To deal with this when data is sent every time '11111' (5 1's) appears in a sequence the next bit is put to be 0. This 0 is removes when the receiver reads a sequence of 5 1's and at the same time sequences cannot appear since a sequence of 6 1's is impossible. This insertion of 0's is called bit stuffing.

1.2 Cyclic redundancy check

Now what if there are bit-flips in the message when it is transmitted?

There is a need to be able to identify and preferable correct some errors. For this purpose some extra bits are appended to the end of the message. This is called the Cyclic redundancy check(CRC).

What are some features we would like in CRC?

- Want to be easily able to detect a large variety of errors
- Creation/verification of this CRC must be efficient
- For any n-bit message some k bit CRC should be computable

1.3 Coding Theory

The process of coming with schemes to create such CRC's is a field called coding theory. The basic idea is as follows

- Any n-bit bitstring is mapped to an n+k bitstring
- Since there are only 2^n valid bit strings in the n+k bitstring domain an error that results in an invalid string can help us detect an error
- Infact given that both sender, receiver know the coding scheme we should be able to map the 'defective' n+k bitstring back to the original message.

1.3.1 Hamming Distance

Hamming distance between two bitstrings(same length) is the number of bit flips needed to transform one to the other. The minimum Hamming distance of a code is the minimum hamming distance between all valid bitstring in the n+k bitstring domain explained before.

1.3.2 Error Detection:

If the minimum Hamming distance is N(for a coding scheme), then any message with less than $N-1$ bit errors(but atleast 1) can never be a valid bitstring. If that was the case min hamming distance for the code is less than N, which cannot be true.

1.3.3 Error Correction:

If the minimum Hamming distance is 2N + 1 (for a coding scheme), then any message with less than $N-1$ bit errors(but atleast 1) can be mapped to the corresponding corrected bitstring accurately.

This bitstring will be the valid bitstring closest to the one with errors. If that was not the case min hamming distance for the code is less than 2N+1, since the bitstring closest to the one with errors and the corrected bitstring will have a hamming distance less than 2N+1.

1.4 Galois Theory

A Galois field, named after the mathematician Évariste Galois, is a finite field that contains a set number of elements and supports operations of addition, subtraction, multiplication, and division (excluding division by zero).

We define a galois field where addition is replaced by xor.

The claim is that for any given data bit string, the remainder obtained when the data(appended with k zeros) is divided[5] by a special k+1 bit generator gives us a CRC.

Example:

Data is '110110', generator is '1101'. Note that here there is no carry in the addition, just bit by bit addition.

[5]Not exactly but similar operation

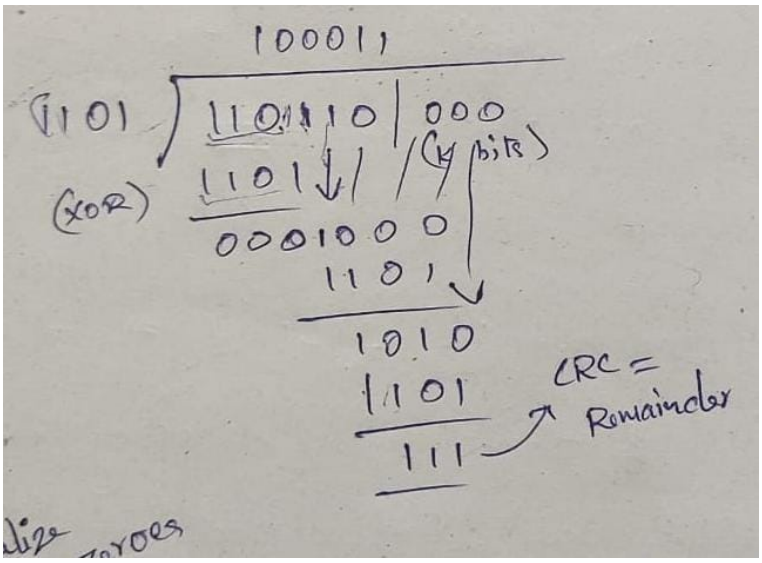


Figure 10: Example for CRC generation

So the algorithm is similar to long division and just uses additions one after another instead of subtraction.

This algorithm can in fact be simulated by a simple logic circuit.

How can this CRC be used for error correction at the receiver side?

- If the received message on divison with the generator does not give 000 as a remainder then there is an error
- Take the data part of the message and append 000 to it and perform division with the generator. If the remainder obtained from this does not match with the CRC there was an error in the transmission

2 Polynomial Arithmetic

As done before in Galois fields, division is still defined using Addition (xor), multiplication (normal).

But encoding is now being done as the coeffiicent of Polynomials. For example, something like '110110' is encoded as $x^5 + x^4 + x^2 + x$. In this context what does it mean for $A(x)$ to be divisble by $D(x)$, basically

$$A(x) = B(x)D(x)$$

Let's try to understand how to use this now,

Say $P(x)$ is the message to be sent (clear from before Polynomial is equivalent to

bitstrings), lets say some bits get flipped in the Polynomial. The bits that get flipped corresponds to a polynomial $E(x)$.

So the total received message can be denoted as $P(x) + E(x)$. Now if the generator is known across both devices which have the message and it also divides $P(x)$, then error detection can be done by checking if $P(x) + E(x)$ is divisible by $C(x)$.

For this method to work, $E(x)$ cannot be divisible by $C(x)$. But is this the case?

- **Single Error Bit:** E(x) = x^i for some i, Suppose $C(x) = x^k + 1$ then $C(x)$ cannot divide $E(x)$,
  To see why think of a divisor for this operation, say $D(x)$

$$C(x)D(x) = x^i \tag{1}$$
$$(x^n + \cdots + 1)(x^m + \cdots + x^q) = x^i \tag{2}$$

  The above is not possible.
- **Two Bit Errors:** $E(x) = x^j + x^i$, (j > i) $E(x) = x^i(x^{j-i} + 1)$
  Suppose $C(x)$ is of the form $x^k + \cdots + 1$
  Now again the problem reduces to if $C(x)$ can divide $P(x)+E(x)$, which basically is asking if it can divide $E(x)$.
  Order of a Polynomial: The smallest 'r' such that $C(x)$ divides $x^r + 1$ is called its order.
  Methods are known to find $C(x)$ such that their orders are very high. **Example:** k = 16, $C(x) = x^{16} + \cdots + 1$, then we can find $c(x)$ s.t it will not divide any $x^p + 1$, if $p < 2^{16} - 1$.
- **Odd number of Errors**
  1. If $C(x) = (1 + x)(...)(...)$, this can catch all odd number of errors.
  2. If $C(x)$ has even number of terms, it can also capture all odd number of errors.

  Why is this true?
  1. Assume that a Polynomial $E(x)$ has an odd number of terms, $E(1)$ is '1', since addition is xor. Similary $C(1)$ is 0. Since the term $1 + 1$ becomes 0. Thus $E(x)$*$D(x) = C(x)$ can never be true since the equation fails for $x = 1$.
  2. Similar argument applies to argue that $C(1)$ is 0 when it has even number of terms.
- **Burst of Errors** Many times, due to interference we have a consecutive sequence of bits which get flipped.
  Thus
$$E(x) = x^{i+l-1} + x^{i+l-2} + \cdots + x^i$$
  , which can also be written as $(x^i)(x^{l-1} + x^{l-2} + \cdots + 1)$
  If our carry is of form $C(x) = x^k + x^{k-1} + x^{k-2} + \cdots + 1$. It can detect all burst errors of length less than l, since it cannot divide the error term. If you are not convinced think of a dividend $D(x)$ then $E(x) = C(x)D(x)$, which is not possible.

2.1 Some Polynomial CRC's

CRC-32: $x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$