

Computer Networking Supervision 1

Daniel Chatfield

November 24, 2014

Topic 01 - Introduction / Foundation

1. *Concepts in (computer) networking.*

Consider a communication network consisting of a room full of people, where one or more people are exchanging thoughts with one or more others by talking.

- (a) For each of the abstract terms node, channel, entity, layer, transmission (the act thereof), coding, addressing and multiplexing, identify one or more corresponding concrete components or activities within the system. If the correspondence is not exact, give the closest approximation you can, and explain why it is not exact. [If you're unfamiliar with any of the abstract terms, you may need to look ahead in the lecture notes, or in one of the course texts.]

Answer:

node A person

channel The air between people (sound waves are produced and travel from one person to another).

entity A person?

layer *not sure*

transmission Speaking

coding Converting the message that someone wants to convey into sentences.

addressing Saying a name before speaking.

multiplexing Multiple conversations at once, or taking it in turns to speak in a group conversation. In the first example it doesn't match a traditional network exactly as when multiple people speak they do so at the same time and use the same frequencies, in a

traditional network it would not be possible to distinguish between different sources. The second activity is an example of *time division multiplexing*

(b) Compare and contrast the network with a *shared-media wireless ethernet* and a *single ethernet* link, for the following channel criteria:

- physical medium
- total capacity
- maximum user-to-user capacity
- medium access control
- geographical area
- failure modes

Answer:

The *room-full-of-people* network, like the wireless ethernet, has no physical media.

Both the *shared-media wireless ethernet* and *single ethernet* networks have a constant *bandwidth*, the bandwidth of the *room-full-of-people* network is dependant on factors like proximity of people, how loud people are talking, how fast people can talk and be understood.

Not sure about user-to-user capacity

The *shared media wireless ethernet* and *single ethernet* networks do not inherently have any access control as all packets (assuming that that is how data is sent) can be snooped by all users, regardless of whether they are the intended recipient. Security can be layered on top of this by requiring that an authenticated session be established between a user and the network, thus preventing the other users from being able to understand the data they can intercept. (e.g. WPA). Alternatively, the data can be encrypted by the sender (e.g. TLS) and then the network itself cannot understand the data either. The *room-full-of-people* network more closely resembles the *shared media wireless* network as people can communicate at the same time. In terms of sending data, all 3 networks have to have some kind of arbitration to determine what can be transmitted at any one time. The *room-full-of-people* network can accommodate multiple people talking at once where as with the single ethernet packets will get trampled and have to be retransmitted. In contrast with the other networks it is unlikely for a user to be able to imitate another user as in a real world setting you can see who is talking.

The geographical area of the *SWME* network is limited by the wireless range of the access point, the range of the people network is limited by the distance at which a person can be understood. The single ethernet network is limited by the distance at which the signal to noise ratio becomes insufficient to distinguish between signal and noise.

Not sure about failure modes

- (c) Suggest one way in which *administration* (or “management”) of a room-full-of-people network may be performed. State whether it is distributed, centralised or a mixture of the two. [Hint: consider the example of a private party.]

Answer:

An administrator could require that people put there hands up when they want to speak and then select people on a needs basis. Alternatively it could be a speed dating type setup where people move around in a predetermined fashion.

The first example is centralised, the second distributed.

2. Multiplexing basics

- (a) Give an example of multiplexing in a real system. Using your example, explain the relationship between multiplexing and coding. Explain your example’s *policy* (concurrency-control) on access to the lower-layer channel, and how this is agreed.

Answer:

Using a cashier at a bank. The concurrency-control policy is based upon *first come first served* and a queue forms.

- (b) Explain the similarity and distinction between *frequency division multiplexing* and *wave division multiplexing*, giving an example of each.

Answer:

Both are ways of multiplexing a number of carrier streams onto a single signal. *Frequency division multiplexing* typically refers to radio carriers where as *wave division multiplexing* refers to an optical carrier.

- (c) What kind of traffic is suited to *synchronous time-division multiplexing*? Define the term *circuit*. Give an example of a *circuit* which is *not* implemented using time-division multiplexing.

Answer:

Digital audio or audio visual broadcasts (e.g. GSM, Digital TV).

A circuit is a discrete path between two or more nodes in the network along which signals can be carried.

Analogue radio is implemented using *frequency-division multiplexing*.

- (d) Give three ways in which asynchronous time-division multiplexing is more complex than synchronous time-division multiplexing. In what circumstances does asynchronous TDM make more efficient use of the lower-layer channel than synchronous TDM?

Answer:

- It requires a buffer for when more than one carrier wants to transmit data.
- It requires address data to be attached so the demultiplexer can tell which carrier transmitted it as this can no longer be derived solely from the time of transmission

Asynchronous TDM makes sense when the rate at which carriers want to transmit data isn't consistent, with synchronous TDM the timeslot is simply wasted but with asynchronous TDM the time-slot is allocated on a "who needs it" basis.

In synchronous TDM, each frame consists of a set of time slots, and each source is assigned one slot per frame. In a particular frame, if a source doesn't have any data, then that time slot goes empty (and is thus wasted).

In asynchronous TDM, the time slots are not pre-assigned - instead slots are allotted to sources "on demand", depending on the availability of data from different sources. A time slot is only unused if none of the data sources has any data to put in it.

- (e) Explain why contention policies are inherently more complex on a shared media link than a point-to-point link using asynchronous TDM. Give an example of each.

Answer:

On a point-to-point link using asynchronous TDM collisions are avoided as the multiplexer chooses what data to include in the frame and only transmits one frame at a particular time.

With a shared media link a node first listens to check that the channel

is idle and then transmits the frame, this helps prevent collisions but doesn't eliminate them and thus the node has to continue listening to see if there is a collision and then has to retransmit (at some random time later).

An example of a shared media link is WiFi. An example of a point to point link would be a PSTN line.

- (f) Ethernet is an asynchronous TDM system with a random-access contention policy. With reference to collision detection, give one reason why shared media Ethernets have a maximum length, and suggest how this might be calculated from the bitrate of the link and the minimum packet length. Give an example of a link which might use asynchronous TDM but where collision detection is not feasible.

Answer:

A node (A) has to be sure that when it has finished transmitting a frame it is not possible that another node (B) started transmitting just before signal A got to it and that signal B has not got back yet.

From the bitrate and the minimum frame length you can calculate the minimum time it takes to transmit a frame. Multiplying this by the speed of light gives you the maximum distance from one node to another and back again.

E.g. for ethernet the minimum frame length is $64B$ ($512bits$), at $100Mbps$ this takes $5.12\mu s$ to transmit. Multiplying half this by the speed of light gives a maximum round trip distance of $1536m$ and thus a maximum network length $\approx 750m$.

- (g) Using the example of a two-way radio conversation, explain token loss and token duplication in token-based asynchronous TDM.

Answer:

Not sure about this

- (h) Explain how slotted systems for asynchronous TDM are different from synchronous TDM. In what case are the two equivalent?

Answer:

In asynchronous TDM the slots are filled by demand rather than being preassigned to a data source. If all data sources continually want to transmit then they are equivalent.

- (i) The telephone network is an example of a reservation system. Explain what is happening when a caller dials a telephone number. Give two reasons why the delay between dialing a number and being connected (i.e. hearing the remote ringer) is variable.

Answer:

When a caller dials a number a circuit is negotiated between the caller and the called party and then they have exclusive use of this line. The delay is variable as the route taken is not constant and it might need to wait for a circuit to become available.

Topic 02 - Architecture and Internet

4. *Standards - so many to choose from and Internet Philosophy*

- (f) Comparing (data) delays in packet and circuit-switched networks. Compare the time it takes to transfer a file of data under circuit-switching and packet-switching. Consider a network consisting of n links in a row, each of B bandwidth and latency L .

Circuit-switching At time $t = 0$, the first node sends out a circuit reservation packet (of size R) which is sent to the second node, which then receives the full packet and then forwards it to the next node. This is continued at each node, until the reservation packet arrives at the last node (after traversing n links). After this reservation message is processed at the last node (the destination), the last node sends back a reservation confirmation message (also of size R) back to the first hop. Because the circuit is established before this confirmation is sent, this packet need not be processed at each node; instead, the bits flow through the nodes without any delay. Once the confirmation message is received at the first node (the source), the source immediately starts sending the file (which is of size F) at the full bandwidth of the link.

Note that when the file is transferred, the data is not stored-and-forwarded at any of the intermediate nodes but is just passed through without delay.

Also, we ignore the teardown message, since it is only sent after the file arrives.

- i. Assuming no problems in transmission along the way, at what time does the last bit of the file arrive at the last node (the destination)?

Answer:

It takes $L + \frac{R}{B}$ to transmit the circuit reservation packet through each of the n links as each node has to process the packet first and thus transmission of the packet onto the next link stalls until the last bit of the packet has been received by the node.

The transmission of the confirmation packet takes $\frac{R}{B} + L$ time to travel through the first link but then just L to travel through all subsequent links (as the bits can just flow straight through without stalling).

Once the circuit is setup it takes $\frac{F}{B}$ time before the last bit enters the first link and then nL time before it reaches the destination.

Putting this together you get: $t = 3nL + \frac{2R+F}{B}$

Packet-switching Here the file is broken into Q packets of size D , each with header size H and payload size P . Since the entire file must be carried, $Q \times P = F$. At time $t = 0$, the source (the first node) sends the first packet, which is stored-and-forwarded at each of the subsequent nodes until it reaches the destination (the last node). As soon as the source finishes sending the first packet, it sends the second packet (at full link bandwidth). Note that the source does not wait until the first packet arrives at the next node before starting the next transmission, it starts sending the next packet as soon as it has finished transmitting the previous packet. We assume that a node can immediately send a packet out on the next link as soon as the last bit has arrived from the previous link (i.e., there is no time required to process the packet before sending it on the next link)

- ii. Assuming no packet drops or other errors, at what time does the last bit of the file arrive at the destination?

Answer:

The first packet is subject to both the regular link latency and also additional latency caused by the fact that all the nodes (except the last) have to store the packet before they can transmit it ($nL + \frac{(n-1)D}{B}$).

The only additional time is the time it takes to transmit all the packets ($\frac{QD}{B}$).

Putting this together you get: $t = nL + \frac{(n-1+Q)D}{B}$

Not sure what the relevance of giving us the header and payload size is unless it is only the header that has to be processed by the intermediary nodes but the question suggests that this is

not the case.

In the following questions, we refer to cases where some quantities are big. By that we mean consider the limit where that quantity becomes infinitely large or infinitesimally small. Note that some quantities are linked (i.e., if the payload P gets smaller, the number of packets Q must get larger to keep $Q \times P = F$). For each question, the answer could be either: circuit-switched is faster, or packet-switched is faster.

Even if you didn't get the formulae above completely correct, you should understand how these perform relative to each other in the limit. Use this as a way to check your answers for the two previous questions.

- iii. If the file size F is very large, which is faster? (Assume that the header size H has not changed.)

Answer:

Circuit switched

- iv. If the payloads become small (but the header size remains constant), which is faster?

Answer:

Circuit switched

- v. If the bandwidth B is very large, which is faster? And by what ratio (in the limit)?

Answer:

Packet switched (3 times as fast).