6.02 Quiz 3/Final Studying 5/13

This is just quiz 3

- modulation on

Need to practice networks, I made a lot of mistakes

And be tottally clear on modulation

Just make cheat sheet

Copy the modlation from the review sheet

Is it in time or freq domain?

- review lab code

- Think its fine

- converts to freq for limiting

  - Yeah [n] means timesteps!

I think in last recitation finally got modulation

- and complex exp.

- I should take anoter look at freq domain - not on quiz though

---

much more simpler than I realized

On a time domain graphs - diff freqs

5/14

Ok now modulation done

Networks

- go through lectures

I need to really understand the diff b/w the 2 algorithms

Ok grades posted

71.67/81

° The Quiz 3 is 19 remaining pts

But don't know letter cut offs
So info not useful

He didn't really provide the analysis steps

So what do the two look like when they are run?

Now the Stop + wait vs sliding window

- last big topic

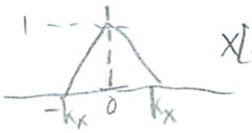
Need to practice Rates, Util, etc

6.02 Cheat Sheet 3

5/13

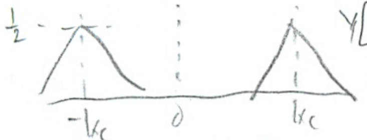
Modulation - Have a band assigned to you
Only transmit in that band

Have signal around baseband



Freq coeff

Then multiply each timestep by $\cos(\overset{\downarrow}{k_c}\frac{2\pi}{N}n)$



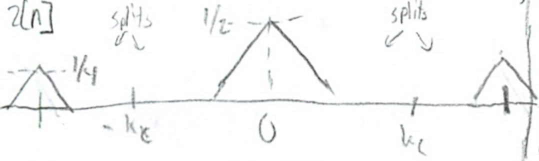
$Y[n] = x[n] \otimes \cos(k_c$

$$y[n] = \underbrace{\sum_{k=-k_x}^{k_x} a_k e^{jk\frac{2\pi}{N}\cdot n}}_{\text{Signal}} \cdot \underbrace{\frac{1}{2}e^{jk_c\frac{2\pi}{N}n} + \frac{1}{2}e^{-jk_c\frac{2\pi}{N}n}}_{\text{the cos}}$$

$$= \frac{1}{2}\sum_{k=-k_x}^{k_x} a_k e^{j(k+k_c)\frac{2\pi}{N}n} + \frac{1}{2}\sum_{k=-k_x}^{k_x} a_k e^{j(k-k_c)\frac{2\pi}{N}n}$$

This is then transmitted over a wire

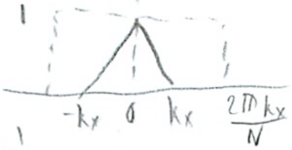
At reciever multiplied again by $\cos(k_c\frac{2\pi}{N}n)$



$$z[n] = y[n]\cdot\cos(k_c\frac{2\pi}{N}n)$$
$$= y[n]\cdot\left[\frac{1}{2}e^{jk_c\frac{2\pi}{N}n} + \frac{1}{2}e^{-jk_c\frac{2\pi}{N}n}\right]$$
$$= \left[\frac{1}{2}\sum a_k e^{j(k+k_c)\frac{2\pi}{N}n} + \frac{1}{2}\sum\cdots\right]\cdot\left[\frac{1}{2}e + \frac{1}{2}e\right]$$
$$= \underset{\text{discard}}{\frac{1}{4}\sum_{k=-k_x}^{k_x} a_k e^{j(k+2k_c)\frac{2\pi}{N}n}} + \underbrace{\frac{1}{2}\sum_{k=-k_x}^{k_x} a_k e^{jk\frac{2\pi}{N}n}}_{\text{keeping}} + \frac{1}{4}\sum_{k=-k_x}^{k_x} a_k e^{j(k-2k_c)\frac{2\pi}{N}n}$$

Then discard other pieces w/ LPF

And double (gain) on center piece



Done!
- this was perfect case

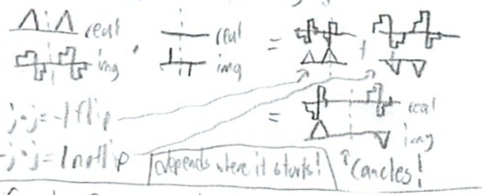
$e^{j\varphi} = \cos(\varphi) + j\sin(\varphi)$

$\cos(\varphi) = \frac{1}{2}e^{j\varphi} + \frac{1}{2}e^{-j\varphi}$

$\sin(\varphi) = \frac{1}{2j}e^{j\varphi} - \frac{1}{2j}e^{-j\varphi}$

Sin is flipped when $j\cdot j = -1$

Notice flips real/imag negative area!



Can be freq or phase error

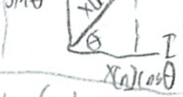
$\cos((\Omega+\epsilon)n)$ $\cos(\Omega n+\epsilon)$

results in amp scaling of $\cos(\epsilon)$

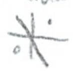
delays look like phase errors

To fix: quadrature demodulation

- demodulate cos, sin seperatly
- $w[n] = I[n] + jQ[n]$
$= \sqrt{I[n]^2 + Q[n]^2}$



DPSK This was thing where it needs to draw line during training
or look at the differential



More complex - can have whole constellation w/ diff radii

Networks global multi-hop networks
- reliable, scalable, peformance, cost, security
- redundent, degradation, not failure

abstract away layers

simplex - one way communication

Half duplex - bidirectional, 1 way at a time

Full-duplex - simultaneous two way comm

Fully connected every point 1 to 1
Throughput $O(N^2)$ Latency $O(1)$ Cost $O(N^2)$

Star one central node - high prone to failure
Throughput $O(N)$ Latency $O(1)$ Cost $O(N)$

Circuit vs Packet switching

Time Divison Mutiplexing - reserve you a slot
C bits/sec channel, each R bits/sec
So C/R # of slots - bad for bursts

So go packet switching - best effort delivery
- good at bursty traffic
- best routing changes automatically
- have a queue

Little's Law ↑ queue size ↑ delay

$N = \lambda D$

N: mean # packets in queue, A/T over time
λ: rate P/T
D: mean delay per packet A/P

A: area under curve

T = time looking at

Say ammusement park
N = # people in line
λ = # people ride takes every minute
D = wait time for each rider

Depends where you define system
Line or Line + Ride
depends where you want to look

Want to find the min cost route: DV vs LS

in both periodic "HELLO" packets - 1. hop to get list of neighbors

2. Advertise 3. Integrate

DV Sends its routing table (dest, cost) to its imediate neighbors
final whole

Bellman-Ford (Integration)

For each entry recieved, add in used link cost

Are we currently using that neighbor?
- if yes update cost
- if no, if lower cost, update table

So routing table is
- for each dest, next link and total cost

if link breaks, ∞ costs should propagate
- as updates that neighbor
- if Hello says that link has failed

code kept sep lists for next links, costs

Sometimes get routing loop as packets sent back + forth till reach ∞ threshold

Path vector send entire path, so can check for duplicate entries

as packets go hop by hop figure out next best place to go

Shortest path X → Y via Z, There must be a shortest path X → Z

Proof on induction of # walks on shortest paths

Converge time: proportional to largest # hops considering all min cost paths

LS

- sends info about its links to its neighbors
- (neighbors, link cost) to them
- Not DV's final dest, cost to final dest
- if ad has higher serial # than before, forward it along all links

Each node should have every packet's ad

Then run Dikstara

- initally nodeset with all nodes
- have a table of costs and table of paths like before
- Find nodeset w/ smallest $\overset{sp}{\text{cost}}$ → call u
  - remove from nodeset
  - For v in u's neighbors
    - d = spcost(v) + cost (u,v)
    - if d < spcost(v) ← previously stored
      - sp cost(v) = d
      - routes[v] = routes[u] ← link

note routes is dict of links that this node must take next for each dest

↑ repeat, new u

Complexity N = # nodes L = # links

Finding u (N times): linear O(N) heap O(logN)

Updating SPcost O(L) since each link 2x

$O(N^2 + L)$ or $O(N\log N + L)$

Same where packet forwarded next link and then next link decides where to send

But LS means it is started in right dir.

Done in a Hierarchy so it scales

kilo	1000	centi	.01
mega	10^6	milli	.001
giga	10^9	micro	10^{-6}
tera	10^{12}	nano	10^{-9}

Packets may become lost on the way

So sometimes need to retransmit

may also arrive out of order, duplicate

will need to fix

Stop + wait - Simple version

- Send packet
- when ACK recieved, send next
- if no Ack in timeout window, retransmit
  - chose timeout so no spurious retx
  - would be RTT if things were perfect
  - but are highly random
  - So Chebchev $P(|X-\mu| \geq k\sigma) \leq \frac{1}{k^2}$
  - pick k, so small prob
- Smooth it w/ SRTT

$SRTT[n] = \alpha\, RTT[n] + (1-\alpha)\, SRTT[n-1]$

α is .125 in TCP, use $.1 \leq \alpha \leq .25$

- Can also use deviation

$dev[n] = |RTT[n] - SRTT[n-1]|$

$SRTTDev[n] = \beta\, dev[n] + (1-\beta)\, SRTTDev[n-1]$

- $timeout[n] = SRTT[n] + k \cdot SRTTdev(n)$ ↑ k=4 for TCP

← to have low tail for spurious Retx

Throughput = 1/T ← time b/w sucessful deliveries

T = RTT if things are perfect

Prob packet is lost $L = 1-(1-p)^N$ ← # links ↑ prob loss per link

So when mess up

$T = (1-L)RTT + L(\text{timeout} + T)$

$= RTT + \frac{L}{1-L}\text{timeout} = RTT + \frac{P(\text{loss})}{P(\text{success})} \cdot RTO$

Suppose RTT same for every packet so timeout = RTT

$T = RTT + \frac{L}{1-L} RTT = \frac{1}{1-L} RTT$

$\text{Throughput} = \frac{1-L}{RTT} = \frac{(1-p)^N}{RTT}$

So max throughput = 10%!

$RTT = \underbrace{\sum \frac{S}{R}}_{\text{data}} + \underbrace{\sum \frac{k}{R}}_{\text{Ack}}$

n = # hops

R = rate of channel

S = size data in bits

k = size ack in bits

$\text{Utilization} = \frac{\text{Throughput}}{\text{Max Throughput} \leftarrow \text{bitrate}} = \frac{\text{data rate}}{\text{link rate}}$

$\text{Max throughput} = \frac{1}{\text{time/packet}} = \frac{1}{\frac{S}{R}} = \frac{R}{S}$

Watch kilo, mega byte/bit!

$\text{Rate} = \frac{\text{bits}}{\text{time}}$ How long to transmit x bits in X seconds?

$\# \text{ trans for sucessful delivery} = \frac{1}{1-L}$

Improve w/ Sliding Window

- allow W packets "in the air" at once
- not really a window - I think
- if unacked list < W, transmit
- each packet has its own timestamp, for timeout
- Reciever has buffer so packets delivered in order

Want W just right

- too small: low throughput
- too large: queues fill up - lots of latency
  - Set $W = B \cdot RTT_{min}$ ↑ rate of slowest link - in packets/sec!
  - use RTT_{min} - not including queue delays since feedback loop would ↑ W to ∞ as queue lenghts ↑
  - Or slighty larger so busy even w/ losses

$\text{Throughput} = \frac{W}{RTT} = \text{effective departure rate}$ | Really $= \min\left(\frac{W}{RTT}, B\right)$

- limited by B = rate of slowest link
  - find by ↑ W till packets dropping

Can double window size | 2W | depends what limiting factor is

Cut RTT/2 | RTT/2

Double B | 2B

if p link fails = p_i

Then $1 - \prod_{i=1}^{N}(1-p_i) = P(\text{packet fails})$

have already

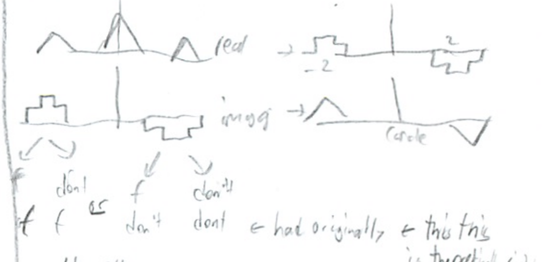
If W_{set} > W supposed to be $(B \cdot RTT_{min})$

then queue is needed

$\text{queue delay} = \frac{W_{set} - W_{supposed}}{B = \text{bottleneck link rate}}$

$RTT = RTT_{min} + \text{queue delay}$

$\text{Throughput} = \frac{W}{RTT} = \text{same as } \frac{W_{suppost}}{RTT_{min}} = B!$ ← note just B!



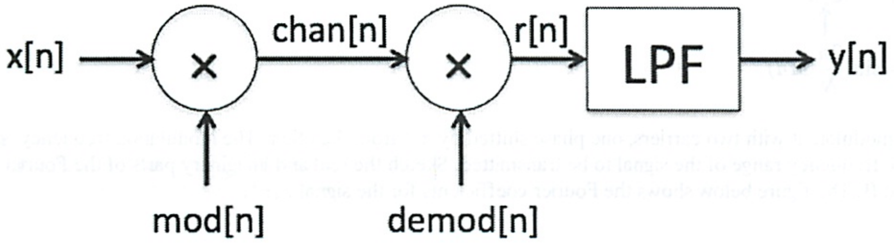
Hopefully less confused

Show All Answers Hide All Answers

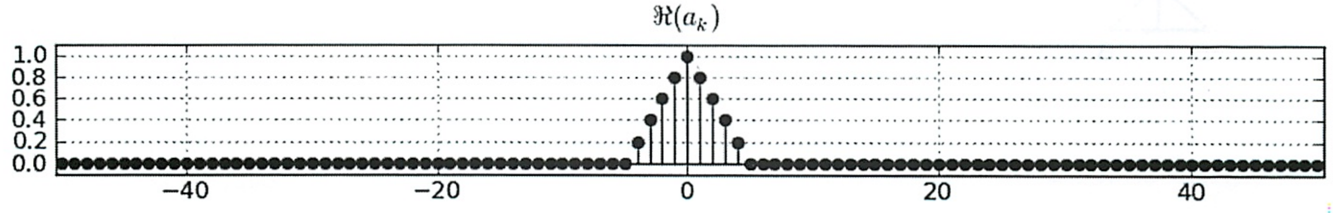
6.02 Tutorial Problems: Modulation

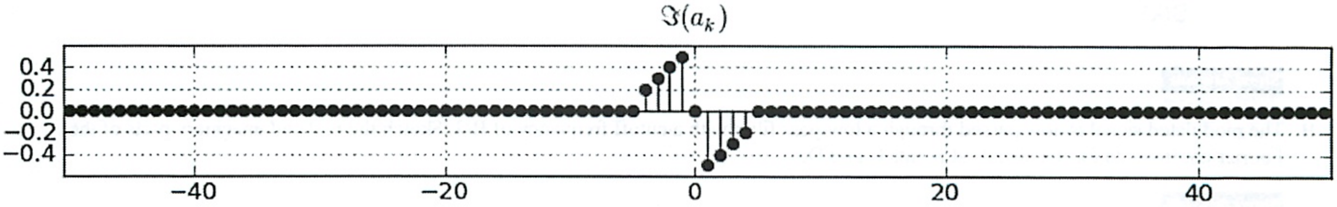
Problem 1.

Here's our "standard" modulation-demodulation system diagram: at the transmitter, signal x[n] is modulated by signal mod[n] and the result (chan[n]) to the receiver where the incoming signal is demodulated by demod[n] and sent through a low-pass filter to produce y[n].



Suppose the band-limited signal x[n] has the spectral coefficients shown below, where N=101:





A. Let mod[n] = cos(15(2π/101)n). Sketch a plot of the spectral coefficients for chan[n], showing both the real and imaginary components. Please label center frequencies and peak amplitudes.

Show Answer

B. Let demod[n] = cos(15(2π/101)n), i.e., a sinusuoid of the same frequency and phase as mod[n]. Sketch a plot of the spectral coefficients for r[n], showing both the real and imaginary components. Please label center frequencies and peak amplitudes.

Show Answer

C. Give the appropriate cutoff frequency and gain for the low-pass filter LPF so that y[n] is identical to x[n].

Show Answer

D. Let mod[n] = sin(15(2π/101)n), i.e., like part (A) except the modulating signal is a sine instead of a cosine. Sketch a plot of the spectral coefficients for chan[n], showing both the real and imaginary components. Please label center frequencies and peak amplitudes.

Show Answer

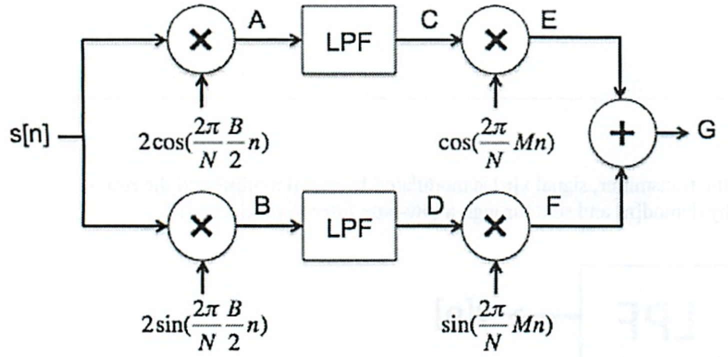
E. Let demod[n] = cos(15(2π/101)n), i.e., a sinusuoid of the same frequency and but π/2 out of phase with mod[n]. Sketch a plot of the spectral coefficients for r[n], showing both the real and imaginary components. Please label center frequencies and peak amplitudes.

Show Answer

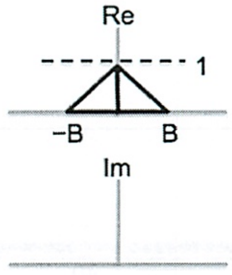
F. Assuming the parameters of the LPF are set as determined in part (C), describe y[n].

Show Answer

Problem 2. Single-sideband (SSB) modulation is a modulation technique designed to minimize the amount of footprint used to transmit an amplitude modulated signal. Here's one way to implement an SSB transmitter.


A
LPF
C
E
s[n]
$2\cos(\frac{2\pi}{N}\frac{B}{2}n)$
$\cos(\frac{2\pi}{N}Mn)$
+
G
B
LPF
D
F
$2\sin(\frac{2\pi}{N}\frac{B}{2}n)$
$\sin(\frac{2\pi}{N}Mn)$

A. Starting with a band-limited signal s[n], modulate it with two carriers, one phase shifted by $\pi/2$ from the other. The modulation frequency is chosen to be B/2, i.e., in the middle of the frequency range of the signal to be transmitted. Sketch the real and imaginary parts of the Fourier coefficients for the signals at points A and B. The figure below shows the Fourier coefficients for the signal s[n].



S(k)

Show Answer

B. The modulated signal is now passed through a low-pass filter with a cutoff frequency of B/2. Sketch the real and imaginary parts of the Fourier coefficients for the signals at points C and D.

Show Answer

C. The signal is modulated once again to shift it up to the desired transmission frequency. Sketch the real and imaginary parts of the Fourier coefficients for the signals at points E and F.

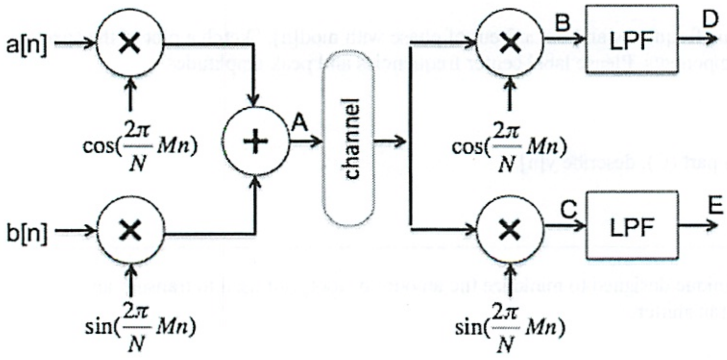
Show Answer

D. Finally the two signals are summed to produce the signal to be sent over the air. Sketch the real and imaginary parts of the Fourier coefficients for the signal at point G.

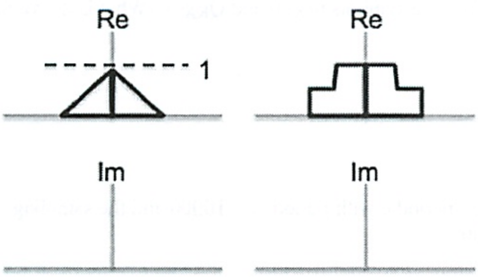
Show Answer

Problem 3.

We learned in lecture that if we modulate a signal with a cosine of a particular frequency and then demodulate with a sine of the same frequency and pass the result through a low-pass filter, we get nothing! We can use this effect to our advantage -- here's a modulation/demodulation scheme that sends two independent signals over a single channel using the same frequency band:



A. The Fourier coefficients for signals `a[n]` and `b[n]` are shown below. Sketch the real and imaginary parts of the Fourier coefficients for the signal at point A.



Spectrum of a[n] Spectrum of b[n]

Show Answer

B. Sketch the real and imaginary parts of Fourier coefficients for signal at point B, right after we demodulate the combined signal using a cosine. The result is passed through a low-pass filter with a cutoff of M; compare the signal at point D to the two input signals and summarize your findings.

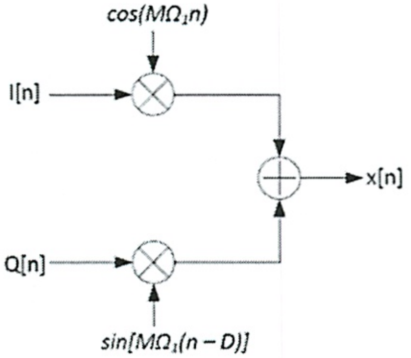
Show Answer

C. Sketch the real and imaginary parts of Fourier coefficients for signal at point C, right after we demodulate the combined signal using a sine. The result is passed through a low-pass filter with a cutoff of M; compare the signal at point E to the two input signals and summarize your findings.

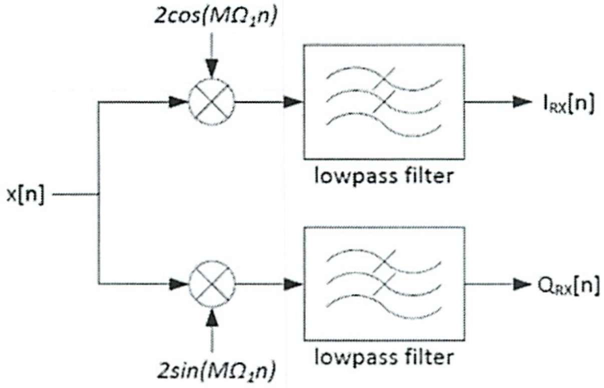
Show Answer

Problem 4.

The diagram for a broken IQ transmitter is shown below. Assume N=1024 and M=64.



The IQ receiver was designed assuming the transmitter was working correctly:



A. The broken transmitter sent the symbols I=1 and Q=1. However, the receiver received the symbols I_{RX}=0.617 and Q_{RX}=0.924. What is the value of D?

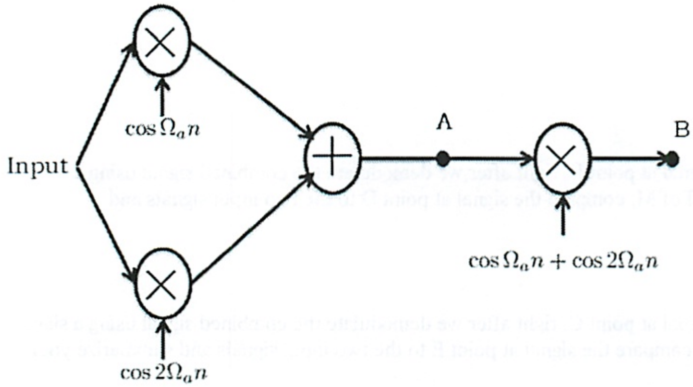
Show Answer

B. The broken transmitter sent the symbols I=1 and Q=1. However, the receiver received the symbols I_{RX}=0 and Q_{RX}=0. What is the value of D?

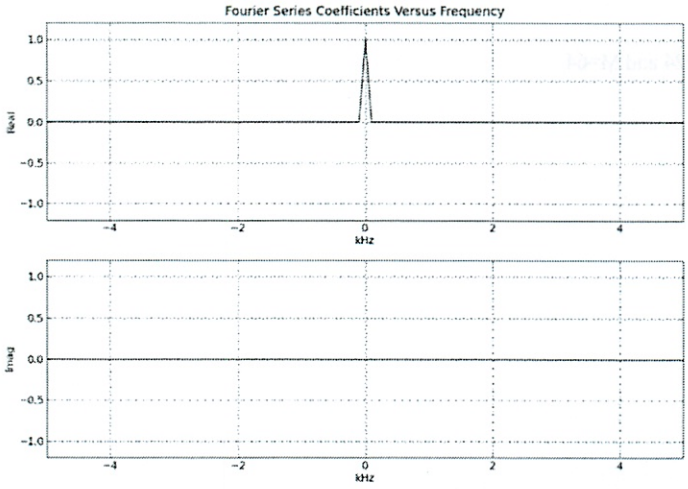
Show Answer

Problem 5.

Consider the simple modulation-demodulation system below, where all signals are assumed periodic with period N = 10000 and the sampling frequency, fs, is 10000 samples per second. In addition, Ωa = 2π(fa/fs) = (1000*2π)/10000.



The Fourier Series coefficients versus frequency for the input to the modulation-demodulation system are plotted below for the case N=10000 and fs=10000. Note that the Fourier coefficients are nonzero only for -100 ≤ k ≤ 100.

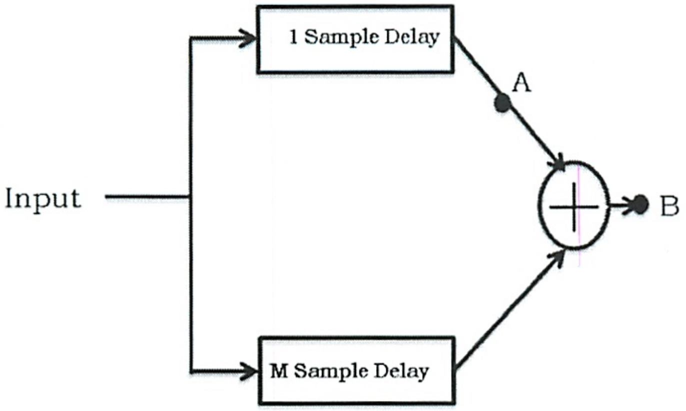


Please plot the Fourier series coefficients versus frequency for the signals at location A and B in the above diagram. Be sure to label key features such as values and coefficient indices for peaks.

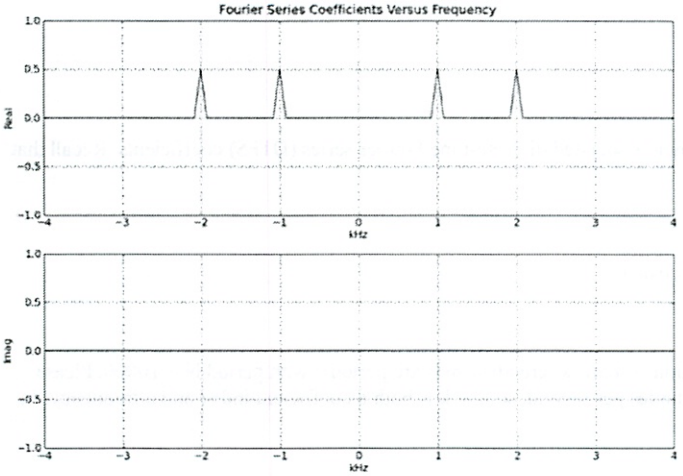
Show Answer

Problem 6.

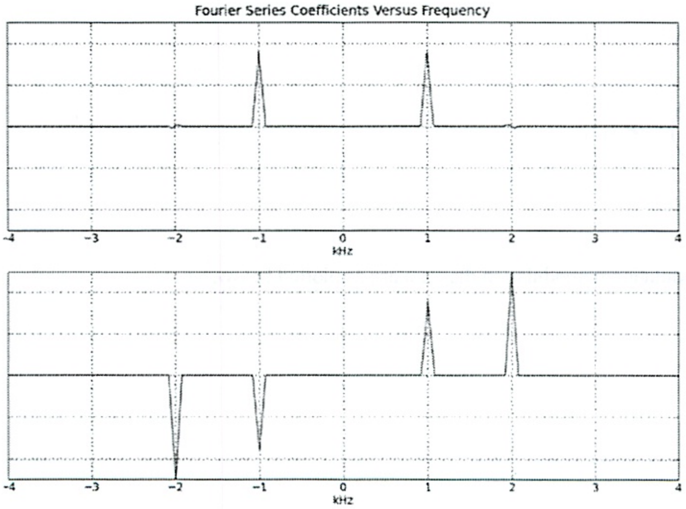
Consider the multiple delay system diagrammed below.



The input to the multiple delay system is a modulated signal that is periodic with period N = 8000 and sampling rate fs = 8000. The Fourier Series coefficients versus frequency for this modulated signal are plotted below.

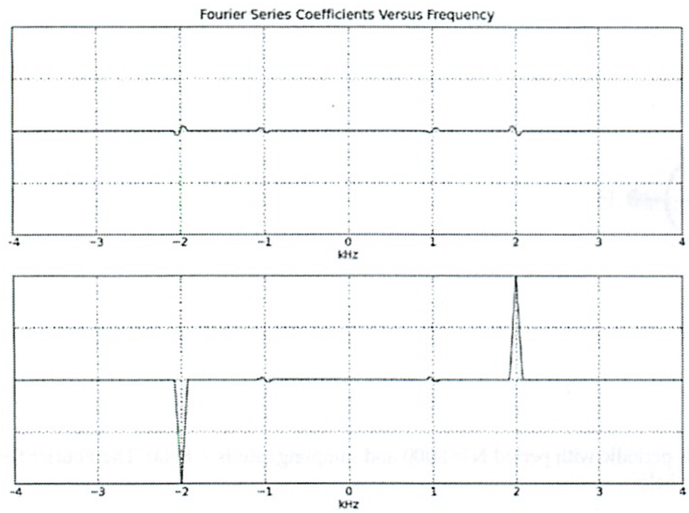


A. Below are plots of the real and imaginary parts of the Fourier coefficients for point A in the multiple delay system. Determine the numerical values for the six peaks in the plots.



Show Answer

B. Use the following plot of the Fourier series coefficients for the sum of the delayed signals (point B in the multiple delay diagram), to determine the smallest integer value for M, the number of samples in the second delay.



Show Answer

Problem 7.

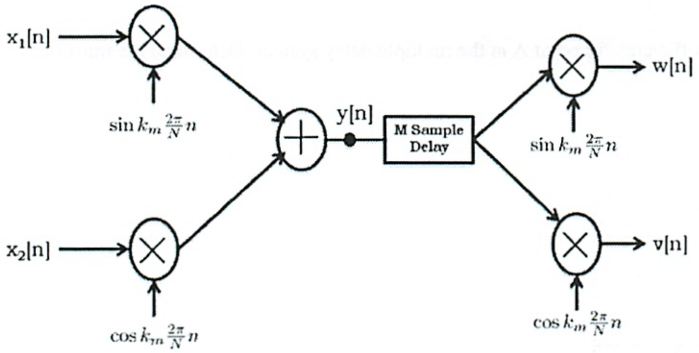
In this modulation problem you will be examining periodic signals and their associated discrete-time Fourier series (DTFS) coefficients. Recall that a periodic signal x[n] with period N has DTFS coefficients given by

$$a_k = (1/N)\,\Sigma_{n=<N>} x[n]\,e^{-jk(2\pi/N)n}$$

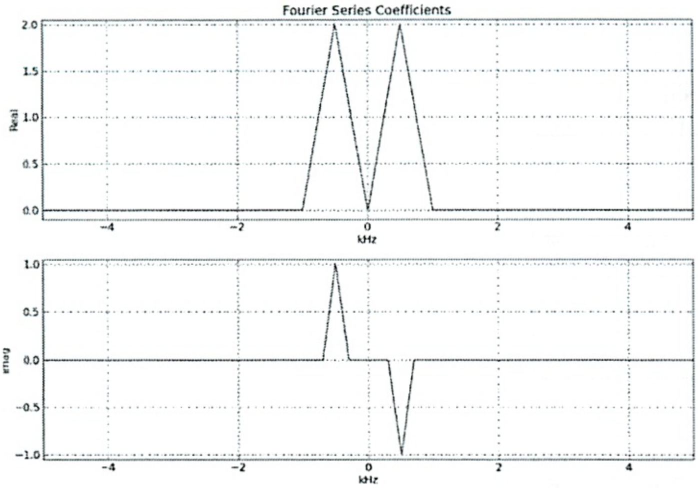
and that the signal x[n] can be reconstructed from the DTFS coefficients using

$$x[n] = \Sigma_{k=<N>} a_k e^{jk(2\pi/N)n}$$

All parts of this question pertain to the following modulation-demodulation system, where all signals are periodic with period N = 10000. Please also assume that the sample rate associated with this system is 10000 samples per second, so that k is both a coefficient index and a frequency. In the diagram, the modulation frequency, km, is 500.



A. Suppose the DTFS coefficients for the signal y[n] in the modulation/demodulation diagram are as plotted below.



Assuming that M = 0 for the M-sample delay (no delay), please plot the DTFS coefficients for the signals w and v in the modulation/demodulation diagram. Be sure to label key features such as values and coefficient indices for peaks.

Show Answer

B. Assuming the DTFS coefficients for the signal y[n] are the same as in part A, please plot the DTFS coefficients for the signal x1 in the modulation/demodulation diagram. Be sure to label key features such as values and coefficient indices for peaks.

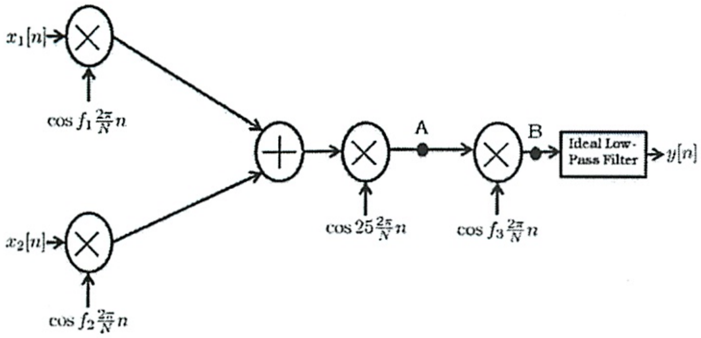
Show Answer

C. If the M-sample delay in the modulation/demodulation diagram has the right number of samples of delay, then it will be possible to nearly perfectly recover x1[n] by low-pass filtering v[n]. Please determine the smallest positive number of samples of delay that are needed and the cut-off frequency for the low-pass filter. Please be sure to justify your answer.

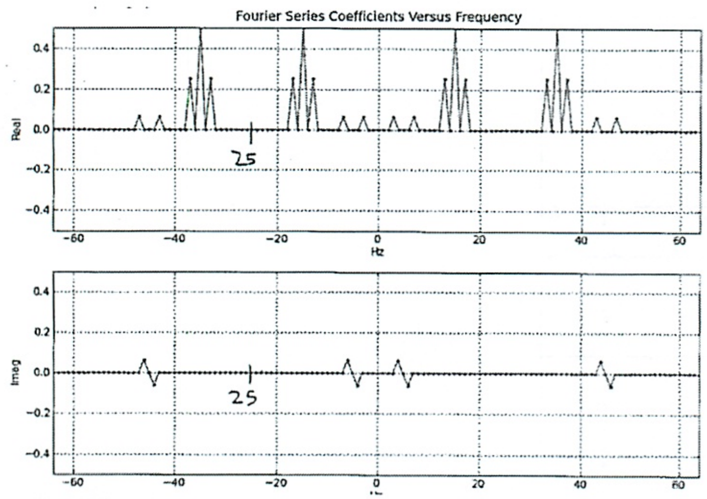
Show Answer

Problem 8.

In answering this question, please refer to the modulation-demodulation system diagrammed below. Assume the N=128 and that the sampling frequency, fs, is 128 samples per second.


$x_1[n]$
$\cos f_1 \frac{2\pi}{N} n$
A
B
Ideal Low-Pass Filter
$y[n]$
$\cos 25 \frac{2\pi}{N} n$
$\cos f_3 \frac{2\pi}{N} n$
$x_2[n]$
$\cos f_2 \frac{2\pi}{N} n$

A. Below are the plots of the real and imaginary parts of the Fourier coefficients vs. frequency for point A of the modulation-demodulation diagram. Please plot the real and imaginary parts of the Fourier coefficients vs. frequency for the signal at point B, assuming f3 = 25. You only need to plot the Fourier coefficents in the range -25 to 25. Please be sure to label critical frequencies and values in your graph.



Show Answer

B. Referring to plot given in part (A), if

```
x1[n] = α + βcos(2(2π/N)n) and
x2[n] = (1/2)cos(2(2π/N)n) + (1/2)sin((2π/N)n)
```

please determine the two modulation frequencies, f1 and f2, and the two amplitudes, α and β.

Show Answer

C. Now suppose

```
x1[n] = (1/2)cos(2(2π/N)n) + (1/2)sin((2π/N)n)
x2[n] = 0
```

and f1 = 15 (not one of the answers to part B!). For what values of f3 < 64 (there is more than one) will y[n] = x1[n], assuming that the low-pass filter has been designed correctly? In addition, what should the magnitude be for the low frequency response of the low-pass filter?

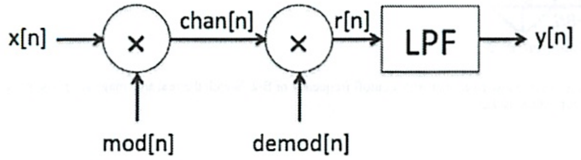
Show Answer

Show All Answers Hide All Answers

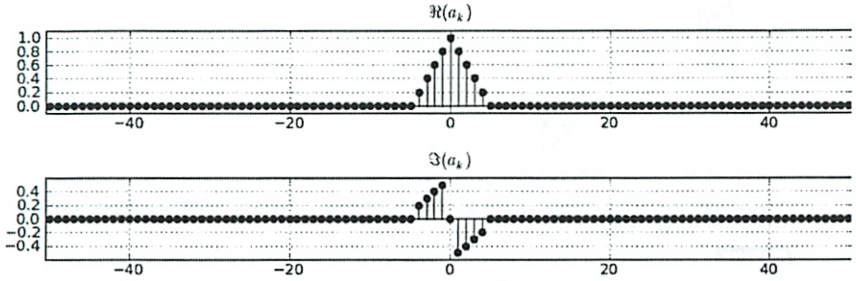
6.02 Tutorial Problems: Modulation

Problem 1.

Here's our "standard" modulation-demodulation system diagram: at the transmitter, signal x[n] is modulated by signal mod[n] and the result (chan[n]) to the receiver where the incoming signal is demodulated by demod[n] and sent through a low-pass filter to produce y[n].

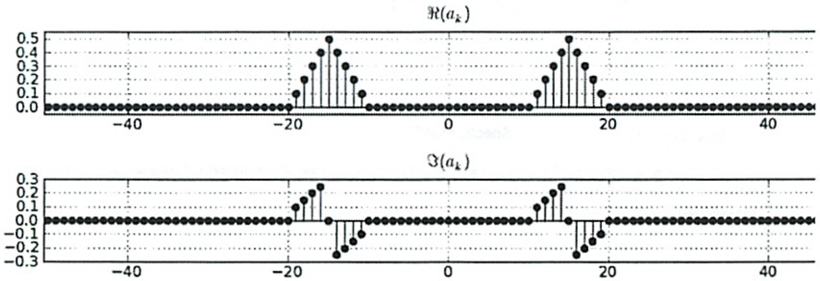


Suppose the band-limited signal x[n] has the spectral coefficients shown below, where N=101:



A. Let mod[n] = cos(15(2π/101)n). Sketch a plot of the spectral coefficients for chan[n], showing both the real and imaginary components. Please label center frequencies and peak amplitudes.

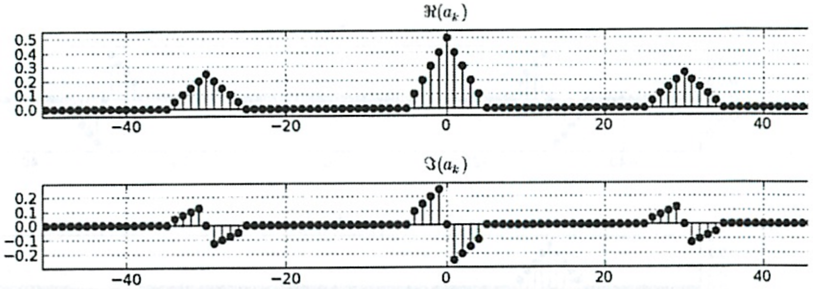
Hide Answer



B. Let demod[n] = cos(15(2π/101)n), i.e., a sinusoid of the same frequency and phase as mod[n]. Sketch a plot of the spectral coefficients for r[n], showing both the real and imaginary components. Please label center frequencies and peak amplitudes.

Hide Answer

Note how the waveforms around frequency 0 are added together to produce a center component that's twice the height of the two side components.



C. Give the appropriate cutoff frequency and gain for the low-pass filter LPF so that y[n] is identical to x[n].

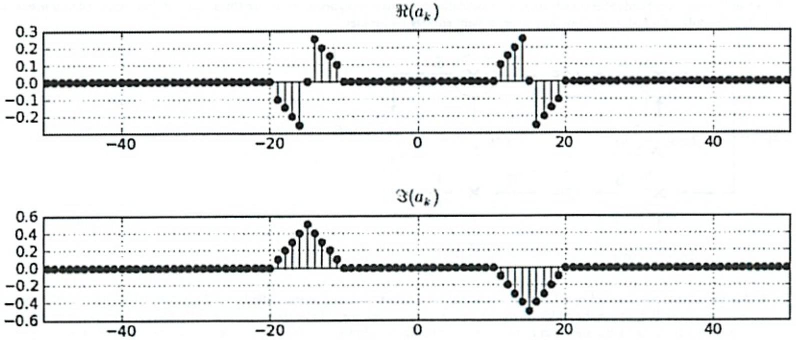
Hide Answer

The low-pass filter should have a gain of 2 and a cutoff frequency that equals the bandwidth of x[n], i.e., k = ±5.

D. Let mod[n] = sin(15(2π/101)n), i.e., like part (A) except the modulating signal is a sine instead of a cosine. Sketch a plot of the spectral coefficients for chan[n], showing both the real and imaginary components. Please label center frequencies and peak amplitudes.

Hide Answer

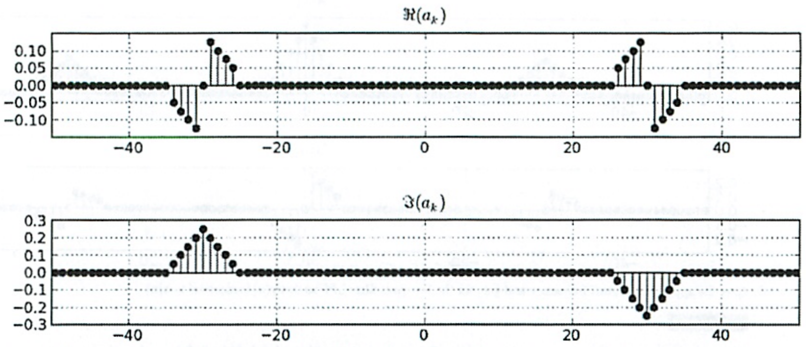
Note how the real and imaginary parts have been interchanged.



E. Let demod[n] = cos(15(2π/101)n), i.e., a sinusoid of the same frequency and but π/2 out of phase with mod[n]. Sketch a plot of the spectral coefficients for r[n], showing both the real and imaginary components. Please label center frequencies and peak amplitudes.

Hide Answer

Note how the real and imaginary parts have been interchanged.

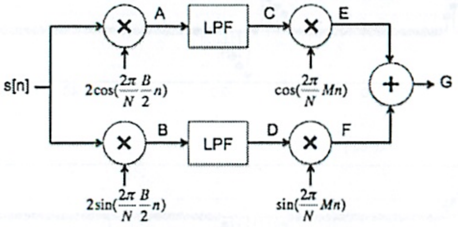


F. Assuming the parameters of the LPF are set as determined in part (C), describe y[n].

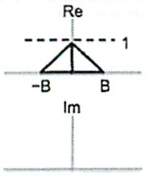
Hide Answer

y[n] = 0. Demodulating with a signal that's π/2 out of phase with the modulating signal produces destructive adding around frequency 0, so the signal in that region has zero amplitude.

Problem 2. Single-sideband (SSB) modulation is a modulation technique designed to minimize the amount of footprint used to transmit an amplitude modulated signal. Here's one way to implement an SSB transmitter.

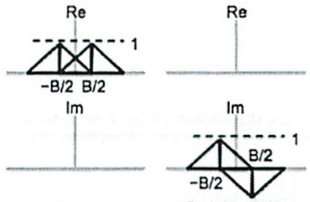

s[n]
A
B
C
D
E
F
G
LPF
LPF
$2\cos(\frac{2\pi}{N}\frac{B}{2}n)$
$2\sin(\frac{2\pi}{N}\frac{B}{2}n)$
$\cos(\frac{2\pi}{N}Mn)$
$\sin(\frac{2\pi}{N}Mn)$

A. Starting with a band-limited signal s[n], modulate it with two carriers, one phase shifted by π/2 from the other. The modulation frequency is chosen to be B/2, i.e., in the middle of the frequency range of the signal to be transmitted. Sketch the real and imaginary parts of the Fourier coefficients for the signals at points A and B. The figure below shows the Fourier coefficients for the signal s[n].



S(k)

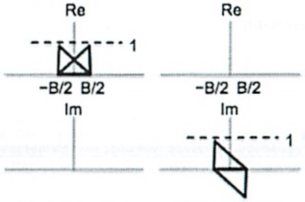
Hide Answer



Spectrum at A Spectrum at B

B. The modulated signal is now passed through a low-pass filter with a cutoff frequency of B/2. Sketch the real and imaginary parts of the Fourier coefficients for the signals at points C and D.

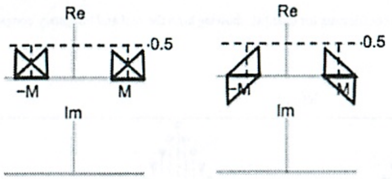
Hide Answer



Spectrum at C Spectrum at D

C. The signal is modulated once again to shift it up to the desired transmission frequency. Sketch the real and imaginary parts of the Fourier coefficients for the signals at points E and F.

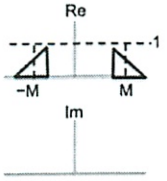
Hide Answer



Spectrum at E Spectrum at F

D. Finally the two signals are summed to produce the signal to be sent over the air. Sketch the real and imaginary parts of the Fourier coefficients for the signal at point G.

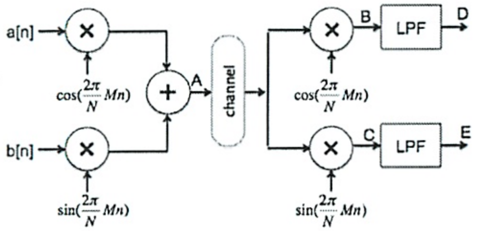
Hide Answer



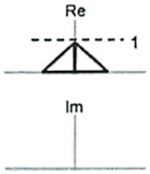
Spectrum at G

Problem 3.

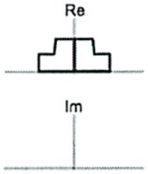
We learned in lecture that if we modulate a signal with a cosine of a particular frequency and then demodulate with a sine of the same frequency and pass the result through a low-pass filter, we get nothing! We can use this effect to our advantage -- here's a modulation/demodulation scheme that sends two independent signals over a single channel using the same frequency band:



A. The Fourier coefficients for signals a[n] and b[n] are shown below. Sketch the real and imaginary parts of the Fourier coefficients for the signal at point A.

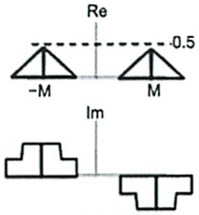


Spectrum of a[n]



Spectrum of b[n]

Hide Answer

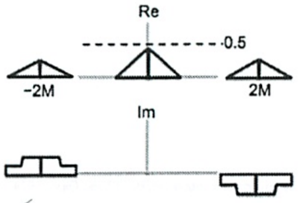


Spectrum at A

B. Sketch the real and imaginary parts of Fourier coefficients for signal at point B, right after we demodulate the combined signal using a cosine. The result is passed through a low-pass filter with a cutoff of M; compare the signal at point D to the two input signals and summarize your findings.

Hide Answer

The signal at point D is a scaled replica of a[n].

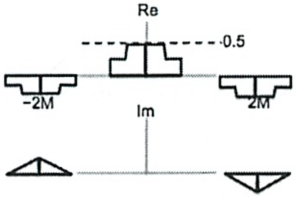


Spectrum at B

C. Sketch the real and imaginary parts of Fourier coefficients for signal at point C, right after we demodulate the combined signal using a sine. The result is passed through a low-pass filter with a cutoff of M; compare the signal at point E to the two input signals and summarize your findings.

Hide Answer

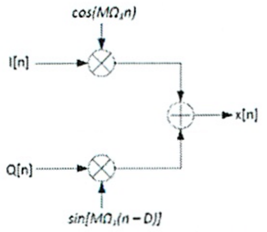
The signal at point E is a scaled replica of b[n].



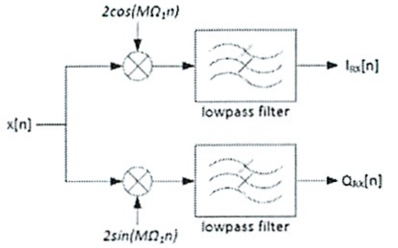
Spectrum at C

Problem 4.

The diagram for a broken IQ transmitter is shown below. Assume N=1024 and M=64.



The IQ receiver was designed assuming the transmitter was working correctly:



A. The broken transmitter sent the symbols I=1 and Q=1. However, the receiver received the symbols I_{RX}=0.617 and Q_{RX}=0.924. What is the value of D?

Hide Answer

$x[n]=I[n]\cos(M\Omega_1 n) + Q[n]\sin(M\Omega_1 n - M\Omega_1 D)$

Recall the identity $\sin(a-b) = \sin(a)\cos(b) - \cos(a)\sin(b)$:

$$x[n] = I[n]\cos(M\Omega_1 n) + Q[n] * [\cos(M\Omega_1 D)\sin(M\Omega_1 n) - \sin(M\Omega_1 D)\cos(M\Omega_1 n)]$$
$$x[n] =[I[n] - Q[n]\sin(M\Omega_1 D)] * \cos(M\Omega_1 n) + Q[n]\cos(M\Omega_1 D) \sin(M\Omega_1 n)$$

The receiver demodulates $x[n]$ back to baseband, so that

$$I_{RX}[n]= I[n] - Q[n]\sin(M\Omega_1 D)$$
$$Q_{RX}[n] = Q[n]\cos(M\Omega_1 D)$$

You can use either of the above formulas to solve for D: D = 1.

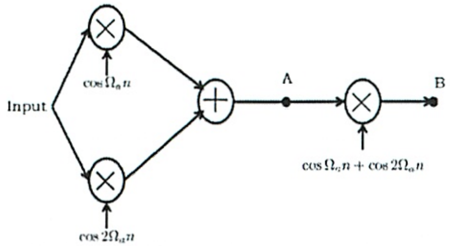
B. The broken transmitter sent the symbols I=1 and Q=1. However, the receiver received the symbols I_{RX}=0 and Q_{RX}=0. What is the value of D?

Hide Answer

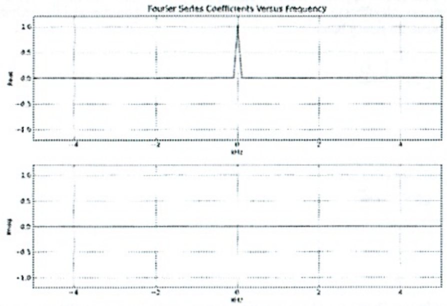
D = 4.

Problem 5.

Consider the simple modulation-demodulation system below, where all signals are assumed periodic with period N = 10000 and the sampling frequency, fs, is 10000 samples per second. In addition, Ωa = 2π(fa/fs) = (1000*2π)/10000.



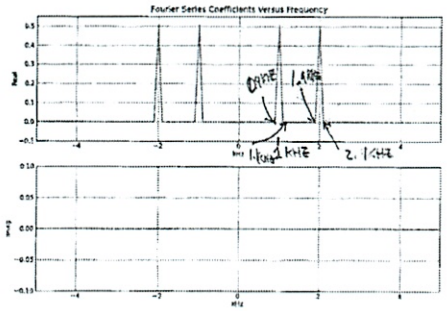
The Fourier Series coefficients versus frequency for the input to the modulation-demodulation system are plotted below for the case N=10000 and fs=10000. Note that the Fourier coefficients are nonzero only for -100 ≤ k ≤ 100.



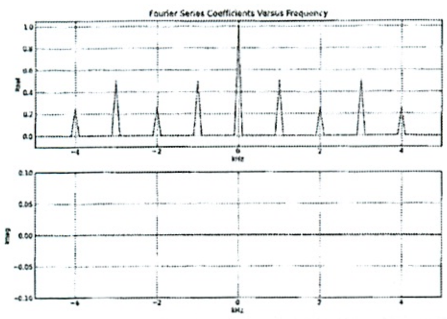
Please plot the Fourier series coefficients versus frequency for the signals at location A and B in the above diagram. Be sure to label key features such as values and coefficient indices for peaks.

Hide Answer

Plot of Fourier Coefficients of signal at Point A

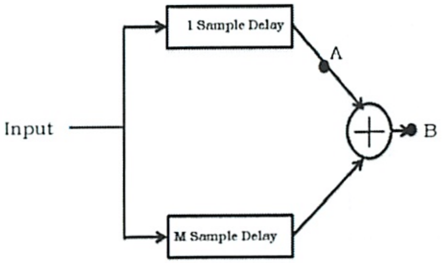


Plot of Fourier Coefficients of signal at Point B

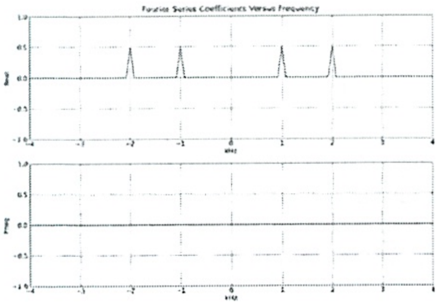


Problem 6.

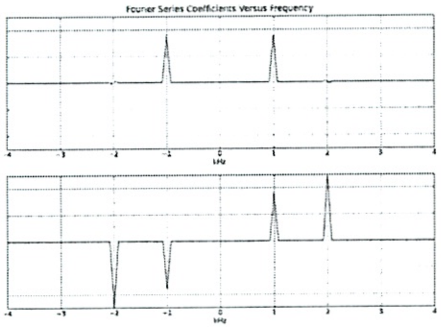
Consider the multiple delay system diagrammed below.



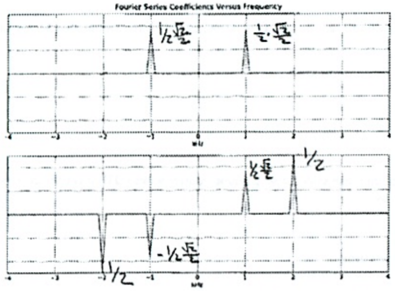
The input to the multiple delay system is a modulated signal that is periodic with period N = 8000 and sampling rate fs = 8000. The Fourier Series coefficients versus frequency for this modulated signal are plotted below.



A. Below are plots of the real and imaginary parts of the Fourier coefficients for point A in the multiple delay system. Determine the numerical values for the six peaks in the plots.



Hide Answer



At $f_k = 1000$ $\cos\left(2\pi\frac{1000}{8000}(n-1)\right) = \cos\left(\frac{2\pi 1000}{8000}n - \frac{\pi}{4}\right)$

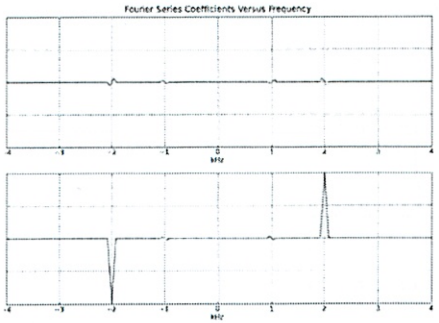
$= \frac{\sqrt{2}}{2}\cos\left(\frac{2\pi 1000}{8000}n\right)$

$+ \frac{\sqrt{2}}{2}\sin\left(\frac{2\pi 1000}{8000}n\right)$

At $f_k = 2000$ $\cos\left(2\pi\frac{2000}{8000}(n-1)\right) = \cos\left(\frac{2\pi 2000}{8000} - \pi/2\right)$

$= -\sin\left(\frac{2\pi 2000}{8000}\right)$

B. Use the following plot of the Fourier series coefficients for the sum of the delayed signals (point B in the multiple delay diagram), to determine the smallest integer value for M, the number of samples in the second delay.



Hide Answer

Must add a 1-sample delayed signal to an M-sample delayed signal and zero out the Ω = 1000(2π/8000) term. Noting that M-1 is the delay difference:

$e^{j(M-1)1000(2\pi/8000)} = -1 = e^{j\pi}$

So M-1 must be 4, which means M = 5.

Problem 7.

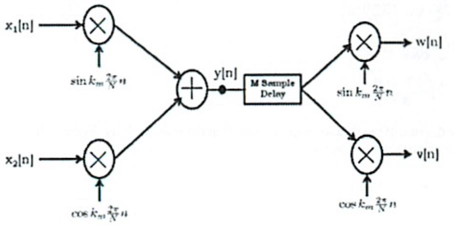
In this modulation problem you will be examining periodic signals and their associated discrete-time Fourier series (DTFS) coefficients. Recall that a periodic signal x[n] with period N has DTFS coefficients given by

$a_k = (1/N)\, \Sigma_{n=<N>} x[n]\, e^{-jk(2\pi/N)n}$

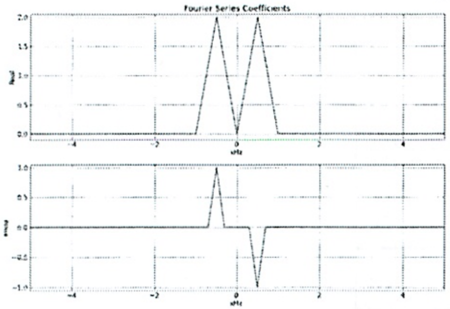
and that the signal x[n] can be reconstructed from the DTFS coefficients using

$x[n] = \Sigma_{k=<N>} a_k e^{jk(2\pi/N)n}$

All parts of this question pertain to the following modulation-demodulation system, where all signals are periodic with period N = 10000. Please also assume that the sample rate associated with this system is 10000 samples per second, so that k is both a coefficient index and a frequency. In the diagram, the modulation frequency, km, is 500.


$x_1[n]$
$\sin k_m \frac{2\pi}{N} n$
$x_2[n]$
$\cos k_m \frac{2\pi}{N} n$
y[n]
M Sample Delay
$\sin k_m \frac{2\pi}{N} n$
w[n]
$\cos k_m \frac{2\pi}{N} n$
v[n]

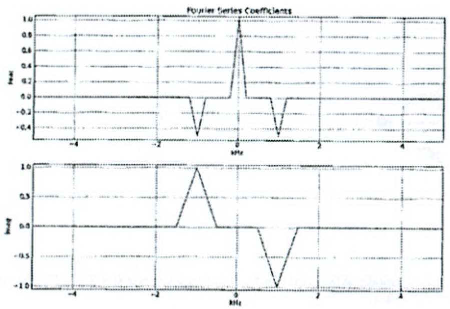
A. Suppose the DTFS coefficients for the signal y[n] in the modulation/demodulation diagram are as plotted below.



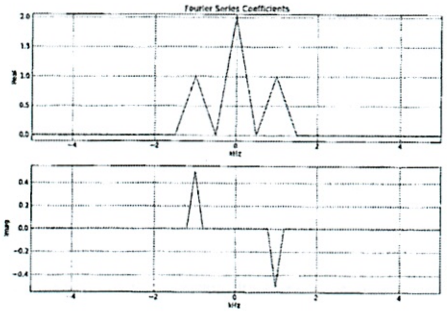
Assuming that M = 0 for the M-sample delay (no delay), please plot the DTFS coefficients for the signals w and v in the modulation/demodulation diagram. Be sure to label key features such as values and coefficient indices for peaks.

Hide Answer

Plot of DFTS coefficients for w

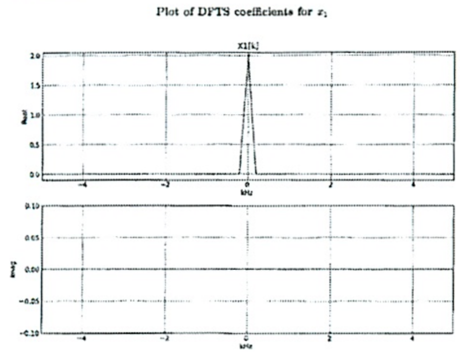


Plot of DFTS coefficients for v

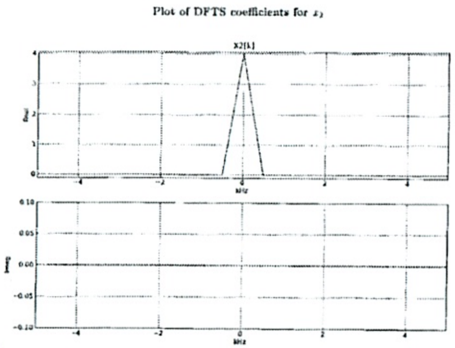


B. Assuming the DTFS coefficients for the signal y[n] are the same as in part A, please plot the DTFS coefficients for the signal x1 in the modulation/demodulation diagram. Be sure to label key features such as values and coefficient indices for peaks.

Hide Answer



For the record, here are the coefficients for x2[n]:



C. If the M-sample delay in the modulation/demodulation diagram has the right number of samples of delay, then it will be possible to nearly perfectly recover x1[n] by low-pass filtering v[n]. Please determine the smallest positive number of samples of delay that are needed and the cut-off frequency for the low-pass filter. Please be sure to justify your answer.

Hide Answer

We want to choose M such that

```
sin(kₘ(2π/N)(n-M)) &approx; cos(kₘ(2π/N)n)
```

i.e., M samples of delay should introduce a phase shift of +π/2, or, equivalently, -3π/2:

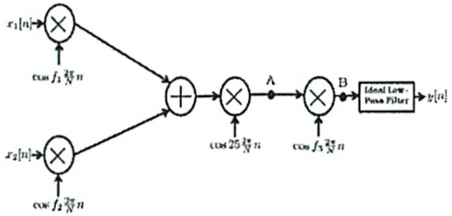
```
-3π/2 = 500(2π/10000)(-M) = (-M)π/10
```

So if M = 15 we get the desired result. Looking at the imaginary part of the plot in part (A), which shows x1[n] modulated by sin(...), we see that the bandwidth of x is 250 kHz.

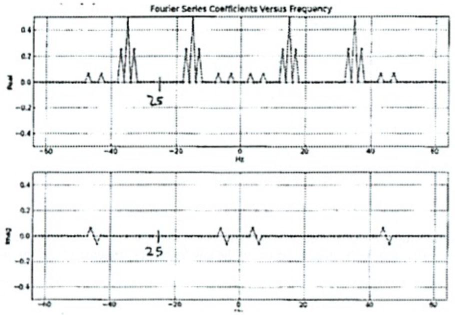
Note that if M=5, the phase shift is -π/2, which produces -cos(...), which we can convert to cos(...) if the LPF has a gain of -1.

Problem 8.

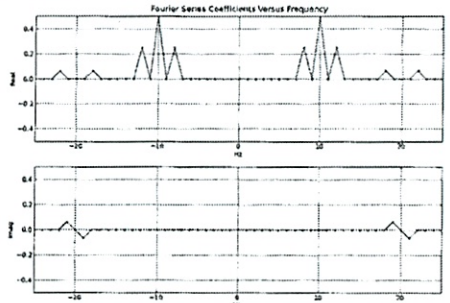
In answering this question, please refer to the modulation-demodulation system diagrammed below. Assume the N=128 and that the sampling frequency, fs, is 128 samples per second.


$x_1[n]$
$\cos f_1 \frac{2\pi}{N} n$
$x_2[n]$
$\cos f_2 \frac{2\pi}{N} n$
$\cos 25 \frac{2\pi}{N} n$
A
$\cos f_3 \frac{2\pi}{N} n$
B
Ideal Low-Pass Filter
$v[n]$

A. Below are the plots of the real and imaginary parts of the Fourier coefficients vs. frequency for point A of the modulation-demodulation diagram. Please plot the real and imaginary parts of the Fourier coefficients vs. frequency for the signal at point B, assuming f3 = 25. You only need to plot the Fourier coefficents in the range -25 to 25. Please be sure to label critical frequencies and values in your graph.



Hide Answer

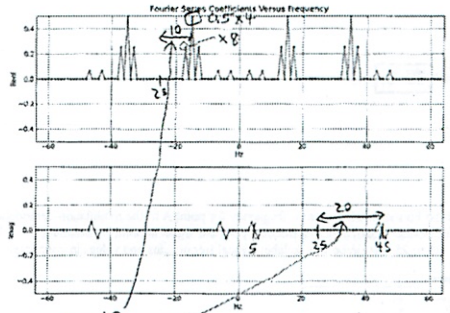


B. Referring to plot given in part (A), if

```
x1[n] = α + βcos(2(2π/N)n) and
x2[n] = (1/2)cos(2(2π/N)n) + (1/2)sin((2π/N)n)
```

please determine the two modulation frequencies, f1 and f2, and the two amplitudes, α and β.

Hide Answer



f_1 = 10

f_2 = 20

α = 2.0 (4 × 0.5)

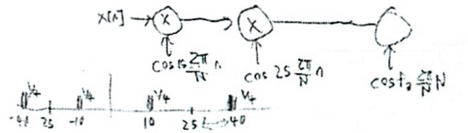
β = 2.0 (4 × 0.25)

C. Now suppose

```
x1[n] = (1/2)cos(2(2π/N)n) + (1/2)sin((2π/N)n)
x2[n] = 0
```

and f1 = 15 (not one of the answers to part B!). For what values of f3 < 64 (there is more than one) will y[n] = x1[n], assuming that the low-pass filter has been designed correctly? In addition, what should the magnitude be for the low frequency response of the low-pass filter?

Hide Answer



Values for f_3 = 10 and 40

Magnitude of the low-pass filter's low frequency response = 4

Show All Answers Hide All Answers

6.02 Tutorial Problems: Network Switching, Little's Law

Problem 1. Little's Law.

A. At the supermarket a checkout operator has on average 4 customers and customers arrive every 2 minutes. How long must each customer wait in line on average?

Show Answer

B. A restaurant holds about 60 people, and the average person will be in there about 2 hours. On average, how many customers arrive per hour? If the restaurant queue has 30 people waiting to be seated, how long does each person have to wait for a table?

Show Answer

C. A fast-food restaurant uses 3,500 kilograms of hamburger each week. The manager of the restaurant wants to ensure that the meat is always fresh, i.e., the meat should be no more than two days old on average when used. How much hamburger should be kept in the refrigerator as inventory?

Show Answer

Problem 2.

Calculate the latency (total delay from first bit sent to last bit received) for the following:

A. Sender and receiver are separated by two 1-Gigabit/s links and a single switch. The packet size is 5000 bits, and each link introduces a propagation delay of 10 microseconds. Assume that the switch begins forwarding immediately after it has received the last bit of the packet and the queues are empty.

Show Answer

B. Same as (A) with three switches and four links.

Show Answer

Problem 3.

Network designers generally attempt to deploy networks that don't have single points of failure, though they don't always succeed. Network topologies that employ redundancy are of much interest.

A. Draw an example of a six-node network in which the failure of a single link does not disconnect the entire network (that is, any node can still reach any other node).

B. Draw an example of a six-node network in which the failure of any single link cannot disconnect the entire network, but the failure of some single node does disconnect it.

C. Draw an example of a six-node network in which the failure of any single node cannot disconnect the entire network, but the failure of some single link does disconnect it.

Note: Not all the cases above may have a feasible example.

Show Answer

Problem 4.

Under what conditions would circuit switching be a better network design than packet switching?

Show Answer

Problem 5.

Circuit switching and packet switching are two different ways of sharing links in a communication network. Indicate True or False for each choice.

A. Switches in a circuit-switched network process connection establishment and tear-down messages, whereas switches in a packet-switched network do not.

Show Answer

B. Under some circumstances, a circuit-switched network may prevent some senders from starting new conversations.

Show Answer

C. Once a connection is correctly established, a switch in a circuit-switched network can forward data correctly without requiring data frames to include a destination address.

Show Answer

D. Unlike in packet switching, switches in circuit-switched networks do not need any information about the network topology to function correctly.

Show Answer

Problem 6.

Consider a switch that uses time division multiplexing (rather than statistical multiplexing) to share a link between four concurrent connections (A, B, C, and D) whose packets arrive in bursts. The link's data rate is 1 packet per time slot. Assume that the switch runs for a very long time.

A. The average packet arrival rates of the four connections (A through D), in packets per time slot, are 0.2, 0.2, 0.1, and 0.1 respectively. The average delays observed at the switch (in time slots) are 10, 10, 5, and 5. What are the average queue lengths of the four queues (A through D) at the switch?

Show Answer

B. Connection A's packet arrival rate now changes to 0.4 packets per time slot. All the other connections have the same arrival rates and the switch runs unchanged. What are the average queue lengths of the four queues (A through D) now?

Show Answer

Problem 7.

Alyssa P. Hacker has set up eight-node shared medium network running the Carrier Sense Multiple Access (CSMA) MAC protocol. The maximum data rate of the network is 10 Megabits/s. Including retries, each node sends traffic according to some unknown random process at an average rate of 1 Megabit/s per node. Alyssa measures the network's utilization and finds that it is 0.75. No packets get dropped in the network except due to collisions, and each node's average queue size is 5 packets. Each packet is 10000 bits long.

A. What fraction of packets sent by the nodes (including retries) experience a collision?

Show Answer

B. What is the average queueing delay, in milliseconds, experienced by a packet before it is sent over the medium?

Show Answer

Problem 8.

Little's law can be applied to a variety of problems in other fields. Here are some simple examples for you to work out.

A. F freshmen enter MIT every year on average. Some leave after their SB degrees (four years), the rest leave after their MEng (five years). No one drops out (yes, really). The total number of SB and MEng students at MIT is N. What fraction of students do an MEng?

Show Answer

B. A hardware vendor manufactures $300 million worth of equipment per year (= invoice$/year). On average, the company has $45 million in accounts receivable (= invoice$). How much time elapses between invoicing and payment?

Show Answer

C. While reading a newspaper, you come across a sentence claiming that "less than 1% of the people in the world die every year". Using Little's law (and some common sense!), explain whether you would agree or disagree with this claim. Assume that the number of people in the world does not decrease during the year (this assumption holds).

Show Answer

D. (This problem is actually almost related to networks.) Your friendly 6.02 professor receives 200 non-spam emails every day on average. He estimates that of these, 50 need a reply. Over a period of time, he finds that the average number of unanswered emails in his inbox that still need a reply is 100.

   i. On average, how much time does it take for the professor to send a reply to an email that needs a response?

   Show Answer

   ii. On average, 6.02 constitutes 25% of his emails that require a reply. He responds to each 6.02 email in 60 minutes, on average. How much time on average does it take him to send a reply to any non-6.02 email?

   Show Answer

Problem 9.

You send a stream of packets of size 1000 bytes each across a network path from Cambridge to Berkeley. You find that the one-way delay varies between 50 ms (in the absence of any queueing) and 125 ms (full queue), with an average of 75 ms. The transmission rate at the sender is 1 Mbit/s; the receiver gets packets at the same rate without any packet loss.

A. What is the mean number of packets in the queue at the bottleneck link along the path (assume that any queueing happens at just one switch).

Show Answer

You now increase the transmission rate to 2 Mbits/s. You find that the receiver gets packets at a rate of 1.6 Mbits/s. The average queue length does not change appreciably from before.

B. What is the packet loss rate at the switch?

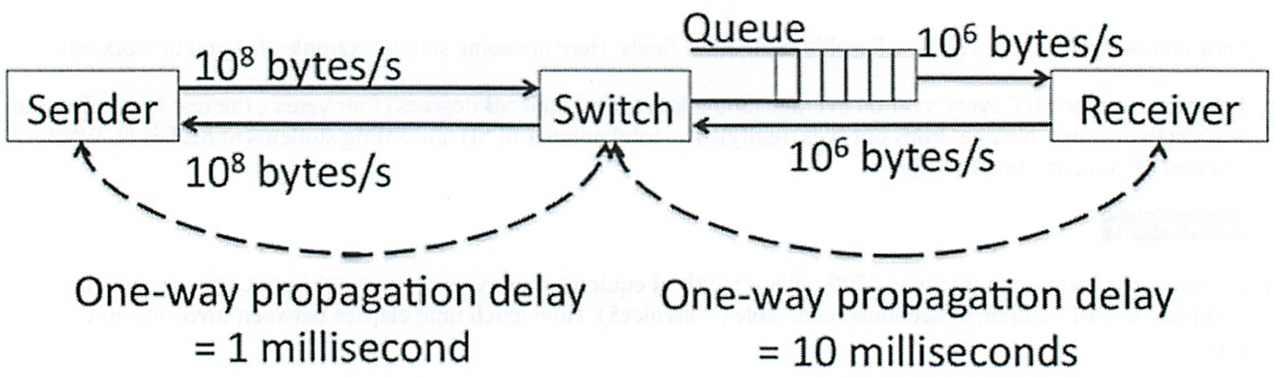
Show Answer

C. What is the average one-way delay now?

Show Answer

Problem 10.

Consider the network topology shown below. Assume that the processing delay at all the nodes is negligible.


Queue
10⁸ bytes/s
10⁶ bytes/s
Sender
Switch
Receiver
10⁸ bytes/s
10⁶ bytes/s
One-way propagation delay = 1 millisecond
One-way propagation delay = 10 milliseconds

A. The sender sends two 1000-byte data packets back-to-back with a negligible inter-packet delay. The queue has no other packets. What is the time delay between the arrival of the first bit of the second packet and the first bit of the first packet at the receiver?

Show Answer

B. The receiver acknowledges each 1000-byte data packet to the sender, and each acknowledgment has a size A = 100 bytes. What is the minimum possible round trip time between the sender and receiver? The round trip time is defined as the duration between the transmission of a packet and the receipt of an acknowledgment for it.

Show Answer

Problem 11.

Annette Werker has developed a new switch. In this switch, 10% of the packets are processed on the "slow path", which incurs an average delay of 1 millisecond. All the other packets are processed on the "fast path", incurring an average delay of 0.1 milliseconds. Annette observes the switch over a period of time and finds that the average number of packets in it is 19. What is the average rate, in packets per second, at which the switch processes packets?

Show Answer

Problem 12.

Alyssa P. Hacker designs a switch for a circuit-switched network to send data on a 1 Megabit/s link using time division multiplexing (TDM). The switch supports a maximum of 20 different simultaneous conversations on the link, and any given sender transmits data in frames of size 2000 bits. Over a period of time, Alyssa finds that the average number of conversations simultaneously using the link is 10. The switch forwards a data frame sent by a given sender every δ seconds according to TDM. Determine the value of δ

Show Answer

Show All Answers Hide All Answers

6.02 Tutorial Problems: Network Switching, Little's Law

Problem 1. Little's Law.

A. At the supermarket a checkout operator has on average 4 customers and customers arrive every 2 minutes. How long must each customer wait in line on average?

Hide Answer

Throughput time = 4 customers / 1/2 customer/minute = 8 minutes

B. A restaurant holds about 60 people, and the average person will be in there about 2 hours. On average, how many customers arrive per hour? If the restaurant queue has 30 people waiting to be seated, how long does each person have to wait for a table?

Hide Answer

Rate = 60 customers / 2 hrs = 30 customers / hr
Waiting time = 1 hour

C. A fast-food restaurant uses 3,500 kilograms of hamburger each week. The manager of the restaurant wants to ensure that the meat is always fresh, i.e., the meat should be no more than two days old on average when used. How much hamburger should be kept in the refrigerator as inventory?

Hide Answer

Rate = 3,500 kilograms per week (= 500 kilograms per day)
Average flow time = 2 days
Average inventory = Rate x Average flow time = 500 x 2 = 1,000 kilograms
(Note that the variables are all in the same time frame i.e. days)

Problem 2.

Calculate the latency (total delay from first bit sent to last bit received) for the following:

A. Sender and receiver are separated by two 1-Gigabit/s links and a single switch. The packet size is 5000 bits, and each link introduces a propagation delay of 10 microseconds. Assume that the switch begins forwarding immediately after it has received the last bit of the packet and the queues are empty.

Hide Answer

For each link, it takes 1 Gigabits/5 Kbits = 5 microseconds to transmit the packet on the link, after which it takes an additional 10 microseconds for the last bit to propagate across the link. Thus, with only one switch that starts forwarding only after receiving the whole packet, the total transfer delay is two transmit delays + two propagation delays = 30 microseconds.

B. Same as (A) with three switches and four links.

Hide Answer

For three switched and thus four links, the total delta is four transmit delays + four propagation delays = 60 microseconds.

Problem 3.

Network designers generally attempt to deploy networks that don't have single points of failure, though they don't always succeed. Network topologies that employ redundancy are of much interest.

A. Draw an example of a six-node network in which the failure of a single link does not disconnect the entire network (that is, any node can still reach any other node).

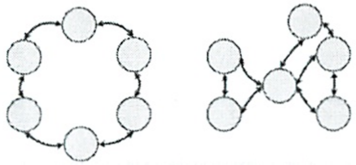
B. Draw an example of a six-node network in which the failure of any single link cannot disconnect the entire network, but the failure of some single node does disconnect it.

C. Draw an example of a six-node network in which the failure of any single node cannot disconnect the entire network, but the failure of some single link does disconnect it.

Note: Not all the cases above may have a feasible example.

Hide Answer

Here is an example topology for part A (left) and B (right).



C. Impossible. If the failure of some link disconnects the network, then pick a node adjacent to that link and fail it. The link in question will fail, and the network will be disconnected, violating the condition that the failure of any single node cannot disconnect the network.

Problem 4.

Under what conditions would circuit switching be a better network design than packet switching?

Hide Answer

Circuit switching offers

- guaranteed capacity. Once a circuit has been estabilished, a TDMA slot has been reserved in each cycle to service this connection.
- no reordering of packets. Since all packets travel the same path, the arrive in the same order as sent.
- bounded delay. Packets move at a fixed pace along the links in the circuit, so their arrival time can be predicted with accuracy.
- no lost packets. Since there are no collisions or queue overflows, packets don't get dropped by the network.

Various streaming applications (VOIP, video streaming, real-time monitoring, ...) would benefit from these features.

Problem 5.

Circuit switching and packet switching are two different ways of sharing links in a communication network. Indicate True or False for each choice.

A. Switches in a circuit-switched network process connection establishment and tear-down messages, whereas switches in a packet-switched network do not.

Hide Answer

True.

B. Under some circumstances, a circuit-switched network may prevent some senders from starting new conversations.

Hide Answer

True.

C. Once a connection is correctly established, a switch in a circuit-switched network can forward data correctly without requiring data frames to include a destination address.

Hide Answer

True.

D. Unlike in packet switching, switches in circuit-switched networks do not need any information about the network topology to function correctly.

Hide Answer

False.

Problem 6.

Consider a switch that uses time division multiplexing (rather than statistical multiplexing) to share a link between four concurrent connections (A, B, C, and D) whose packets arrive in bursts. The link's data rate is 1 packet per time slot. Assume that the switch runs for a very long time.

A. The average packet arrival rates of the four connections (A through D), in packets per time slot, are 0.2, 0.2, 0.1, and 0.1 respectively. The average delays observed at the switch (in time slots) are 10, 10, 5, and 5. What are the average queue lengths of the four queues (A through D) at the switch?

Hide Answer

With TDMA, each connection gets to send 1 packet every 4 time slots, or .25 packets/slot. And with TDMA, the behavior of each connection is independent of what's happening on the other connections. All of the arrival rates are less than this number, so the queue lengths are bounded.

Using Little's Law: N = λ*D, so

A: N = 0.2 * 10 = 2 packets
B: N = 0.2 * 10 = 2 packets
C: N = 0.1 * 5 = .5 packets
D: N = 0.1 * 5 = .5 packets

B. Connection A's packet arrival rate now changes to 0.4 packets per time slot. All the other connections have the same arrival rates and the switch runs unchanged. What are the average queue lengths of the four queues (A through D) now?

Hide Answer

Following the reasoning above, the queues for connections B, C and D are unaffected by the change in A's arrival rate, so the average queue lengths.

The arrival rate on connection A now exceeds the outgoing rate, so the queue length on A is unbounded.

Problem 7.

Alyssa P. Hacker has set up eight-node shared medium network running the Carrier Sense Multiple Access (CSMA) MAC protocol. The maximum data rate of the network is 10 Megabits/s. Including retries, each node sends traffic according to some unknown random process at an average rate of 1 Megabit/s per node. Alyssa measures the network's utilization and finds that it is 0.75. No packets get dropped in the network except due to collisions, and each node's average queue size is 5 packets. Each packet is 10000 bits long.

A. What fraction of packets sent by the nodes (including retries) experience a collision?

Hide Answer

Network capacity is 10^7 bits/s and each packet is 10^4 bits, so the network can transmit up to 1000 packets/s.

If the utilization is .75, then 750 packets are delivered each second. In that time the 8 nodes each send 100 packets, for a total of 800 packets. So 50 packets must be lost due to collision, which 50/800 = 6.25%.

B. What is the average queueing delay, in milliseconds, experienced by a packet before it is sent over the medium?

Hide Answer

N = 5 packets, λ = 100 packets/s
D = N/λ = 5/100 = .05 s.

Problem 8.

Little's law can be applied to a variety of problems in other fields. Here are some simple examples for you to work out.

A. F freshmen enter MIT every year on average. Some leave after their SB degrees (four years), the rest leave after their MEng (five years). No one drops out (yes, really). The total number of SB and MEng students at MIT is N. What fraction of students do an MEng?

Hide Answer

a = fraction of students who do MEng
λ = F
D = (1-a)*4 + a*5 = 4 + a
N = λ*D → N/F = 4 + a → a = N/F - 4

B. A hardware vendor manufactures $300 million worth of equipment per year (= invoice$/year). On average, the company has $45 million in accounts receivable (= invoice$). How much time elapses between invoicing and payment?

Hide Answer

λ = 300e6 invoice$/year
N = 45e6 invoice$
D = N/λ = 45/300 = .15 years = 55 days

C. While reading a newspaper, you come across a sentence claiming that "less than 1% of the people in the world die every year". Using Little's law (and some common sense!), explain whether you would agree or disagree with this claim. Assume that the number of people in the world does not decrease during the year (this assumption holds).

Hide Answer

N = people
λ = .01*N people/year
D = N/λ = N/(.01*N) = 100 years = life span!

Since our life span is less than 100, the stated λ must not be correct.

D. (This problem is actually almost related to networks.) Your friendly 6.02 professor receives 200 non-spam emails every day on average. He estimates that of these, 50 need a reply. Over a period of time, he finds that the average number of unanswered emails in his inbox that still need a reply is 100.

i. On average, how much time does it take for the professor to send a reply to an email that needs a response?

Hide Answer

λ = 50 emails/day
N = 100 emails
D = N/λ = 100/50 = 2 days

ii. On average, 6.02 constitutes 25% of his emails that require a reply. He responds to each 6.02 email in 60 minutes, on average. How much time on average does it take him to send a reply to any non-6.02 email?

Hide Answer

average D = 2 days = 48 hours = (.25)(1 hour) + (.75)(x hours)
x = (48 - .25)/.75 = 63.67 hours

Problem 9.

You send a stream of packets of size 1000 bytes each across a network path from Cambridge to Berkeley. You find that the one-way delay varies between 50 ms (in the absence of any queueing) and 125 ms (full queue), with an average of 75 ms. The transmission rate at the sender is 1 Mbit/s; the receiver gets packets at the same rate without any packet loss.

A. What is the mean number of packets in the queue at the bottleneck link along the path (assume that any queueing happens at just one switch).

Hide Answer

λ = 1 Mb/s = 1000 packets/s
D = 75 ms
N = λ*D = (1000 p/s)*(.075 s) = 75 packets

You now increase the transmission rate to 2 Mbits/s. You find that the receiver gets packets at a rate of 1.6 Mbits/s. The average queue length does not change appreciably from before.

B. What is the packet loss rate at the switch?

Hide Answer

Send 2000 packets/sec, receive 1600 packets/sec, so loss rate must be 400 packets/sec.

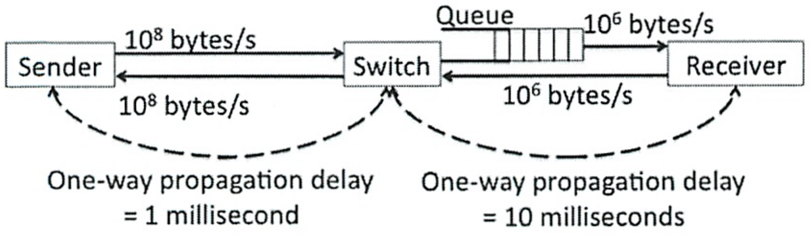
C. What is the average one-way delay now?

Hide Answer

N = 75 packets (unchanged from above)
λ = 1600 packets/sec
D = N/λ = 75/1600 = 47 ms

Problem 10.

Consider the network topology shown below. Assume that the processing delay at all the nodes is negligible.


Queue
10^8 bytes/s
10^6 bytes/s
Sender
Switch
Receiver
10^8 bytes/s
10^6 bytes/s
One-way propagation delay = 1 millisecond
One-way propagation delay = 10 milliseconds

A. The sender sends two 1000-byte data packets back-to-back with a negligible inter-packet delay. The queue has no other packets. What is the time delay between the arrival of the first bit of the second packet and the first bit of the first packet at the receiver?

Hide Answer

After the first bit of the first packet arrives at the receiver, it will take 1 millisecond for the rest of the packet to arrive (1000 bytes at 1 million bytes/sec).

Since the connection from the sender is much faster than the connection to the receiver, the second packet arrives in the queue before the first packet has been completely sent to the receiver. That means that the first bit of the second packet is available for sending right after the last bit of the first packet.

So the total delay is 1 ms.

B. The receiver acknowledges each 1000-byte data packet to the sender, and each acknowledgment has a size A = 100 bytes. What is the minimum possible round trip time between the sender and receiver? The round trip time is defined as the duration between the transmission of a packet and the receipt of an acknowledgment for it.

Hide Answer

The propagation time from the sender to the receiver is 11 ms, which is how long the first bit of a packet takes to arrive assuming no queueing delay in the switch. It then takes another 1 ms for the rest of the packet to arrive. So it takes 12 ms for the complete packet to arrive at the receiver.

After the complete packet has arrived, the receiver generates an acknowledgment, which takes another 10.1 ms to completely arrive at the switch. Assuming the switch doesn't forward the acknowledgment until it completely arrives, then it takes another 1.001 ms for the acknowledgment to completely arrive at the sender.

Round trip time = 12 + 10.1 + 1.001 = 23.101 ms. In less picky mode, we'd probably ignore the transmission times of the small packet (.101 ms) as being relatively immaterial.

Problem 11.

Annette Werker has developed a new switch. In this switch, 10% of the packets are processed on the "slow path", which incurs an average delay of 1 millisecond. All the other packets are processed on the "fast path", incurring an average delay of 0.1 milliseconds. Annette observes the switch over a period of time and finds that the average number of packets in it is 19. What is the average rate, in packets per second, at which the switch processes packets?

Hide Answer

N = 19 packets
D = (.1)(1 ms) + (.9)(.1 ms) = .19 ms
λ = N/D = 19/.00019 = 10000 p/s

Problem 12.

Alyssa P. Hacker designs a switch for a circuit-switched network to send data on a 1 Megabit/s link using time division multiplexing (TDM). The switch supports a maximum of 20 different simultaneous conversations on the link, and any given sender transmits data in frames of size 2000 bits. Over a period of time, Alyssa finds that the average number of conversations simultaneously using the link is 10. The switch forwards a data frame sent by a given sender every δ seconds according to TDM. Determine the value of δ

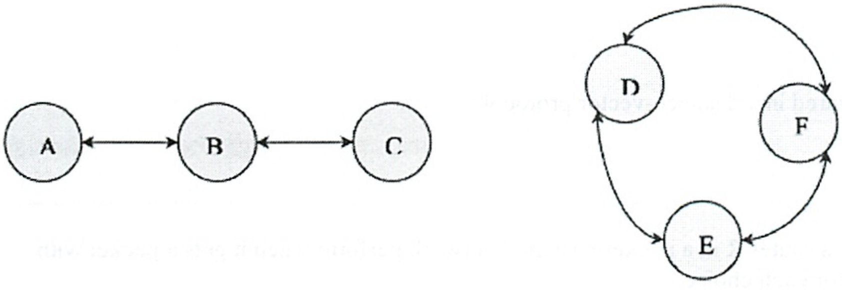
Hide Answer

Each frame (2000 bits) takes 2e3/1e6 = 2 ms. With 20 conversations, it takes 20*2 = 40 ms for one complete cycle (δ) through the TDM schedule.

Show All Answers Hide All Answers

6.02 Tutorial Problems: Routing

Problem 1. Consider the following networks: network I (containing nodes A, B, C) and network II (containing nodes D, E, F).



A. The Distance Vector Protocol described in class is used in both networks. Assume advertisements are sent every 5 time steps, all links are fully functional and there is no delay in the links. Nodes take zero time to process advertisements once they receive them. The HELLO protocol runs in the background every time step in a way that any changes in link connectivity are reflected in the next DV advertisement. We count time steps from t=0 time steps.

Please fill in the following table:

Event	Number of time steps
A's routing table has an entry for B	
A's routing table has an entry for C	
D's routing table has an entry for E	
F's routing table has an entry for D	

Show Answer

B. Now assume the link B-C fails at t = 51 and link D-E fails at t = 71 time steps. Please fill in this table:

Event	Number of time steps
B's *advertisements* reflect that C is unreachable	
A's routing table reflects C is unreachable	
D's routing table reflects a new route for E	

Show Answer

Problem 2. Alyssa P. Hacker manages MIT's internal network that runs link-state routing. She wants to experiment with a few possible routing strategies. Of all possible paths available to a particular destination at a node, a routing strategy specifies the path that must be picked to create a routing table entry. Below is the name Alyssa has for each strategy and a brief description of how it works.

MinCost: Every node picks the path that has the smallest sum of link costs along the path. (This is the minimum cost routing you implemented in the lab).

MinHop: Every node picks the path with the smallest number of hops (irrespective of what the cost on the links is).

SecondMinCost: Every node picks the path with the second lowest sum of link costs. That is, every node picks the second best path with respect to path costs.

MinCostSquared: Every node picks the path that has the smallest sum of squares of link costs along the path.

Assume that sufficient information (e.g., costs, delays, bandwidths, and loss probabilities of the various links) is exchanged in the link state advertisements, so that every node has complete information about the entire network and can correctly implement the strategies above. You can also assume that a link's properties don't change, e.g., it doesn't fail.

A. Help Alyssa figure out which of these strategies will work correctly, and which will result in routing with loops. In case of strategies that do result in routing loops, come up with an example network topology with a routing loop to convince Alyssa.

   Show Answer

B. How would you implement MinCostSquared in a distance-vector protocol?

   Show Answer

Problem 3. Which of the following tasks does a router R in a packet-switched network perform when it gets a packet with destination address D? Indicate True or False for each choice.

A. R looks up D in its routing table to determine the outgoing link.

   Show Answer

B. R sends out a HELLO packet or a routing protocol advertisement to its neighbors.

   Show Answer

C. R calculates the minimum-cost route to destination D.

   Show Answer

D. R may discard the packet.

   Show Answer

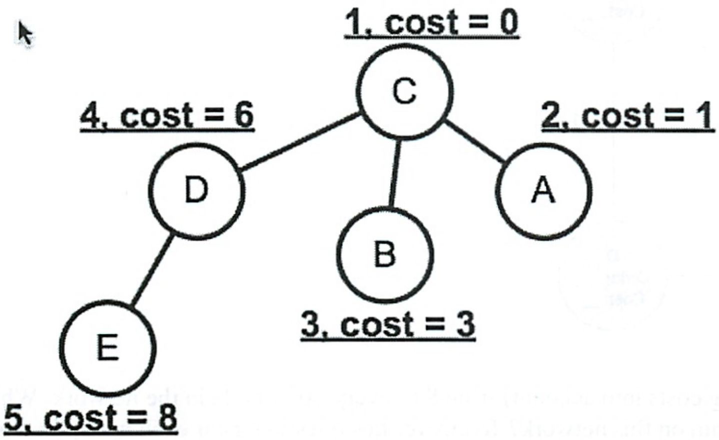
Problem 4. Alice and Bob are responsible for implementing Dijkstra's algorithm at the nodes in a network running a link-state protocol. On her nodes, Alice implements a minimum-cost algorithm. On his nodes, Bob implements a "shortest number of hops" algorithm. Give an example of a network topology with 4 or more nodes in which a routing loop occurs with Alice and Bob's implementations running simultaneously in the same network. Assume that there are no failures.

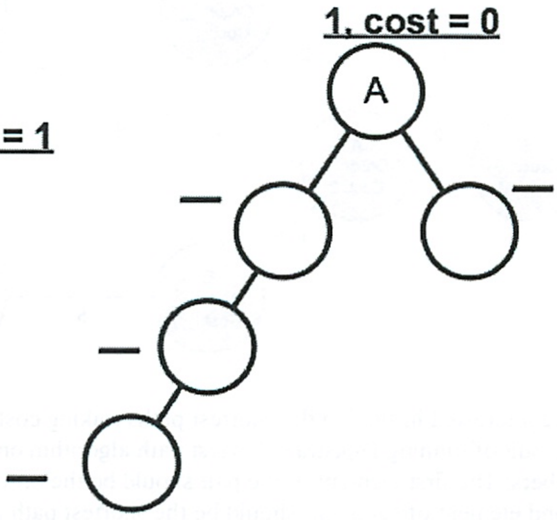
(Note: A routing loop occurs when a group of k ≥ 1 distinct nodes, n_0 , n_1 , n_2 , …, n_(k-1) have routes such that n_i's next-hop (route) to a destination is n_(i+1 mod k).

Show Answer

Problem 5. Consider the network shown below. The number near each link is its cost.

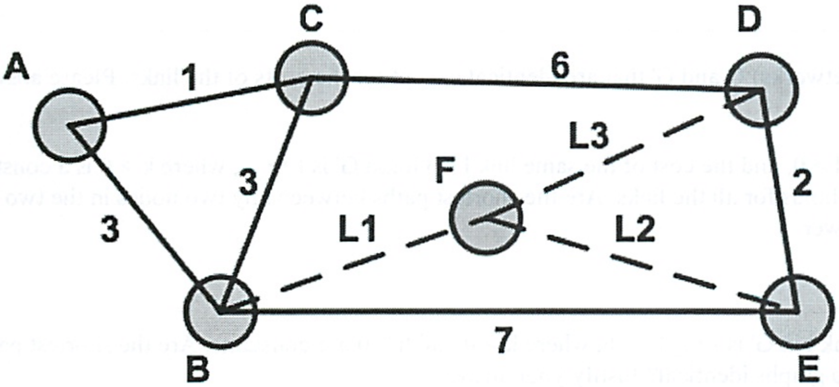
completed routing tree is shown as well.



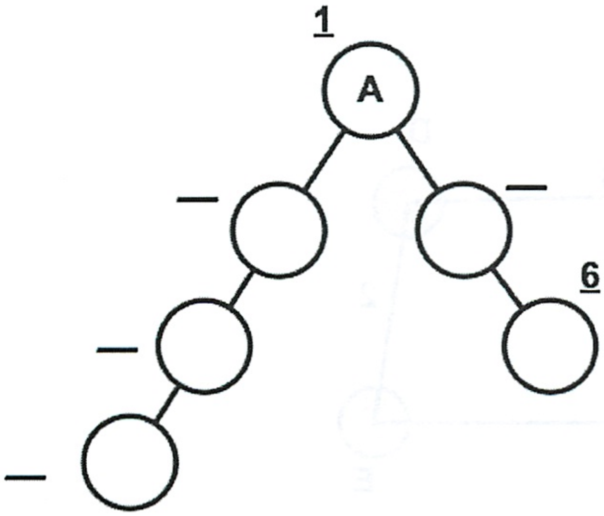


Show Answer

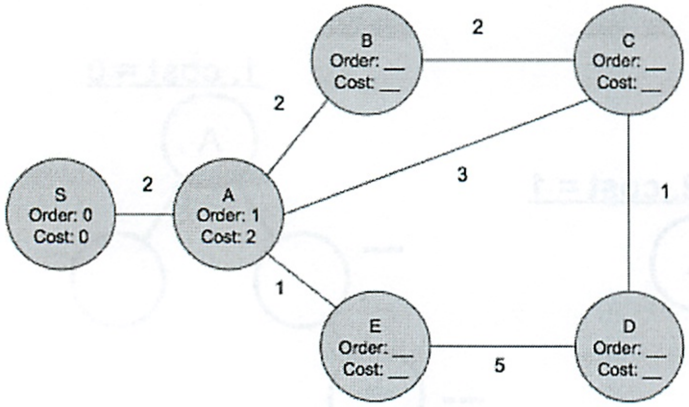
B. Now assume that node F has been added to the network along with links L1, L2 and L3.



What are the constraints on L1, L2 and L3 such that node A's routing tree matches the topology shown below, and it is known that node F is not the last node added when using Dijkstra's algorithm?



Show Answer



We're interested in finding the shortest paths (taking costs into account) from S to every other node in the network. What is the result of running Dijkstra's shortest path algorithm on this network? To answer this question, near each node, list a pair of numbers: The first element of the pair should be the order, or the iteration of the algorithm in which the node is picked. The second element of each pair should be the shortest path cost from S to that node.

To help you get started, we've labeled the first couple of nodes: S has a label (Order: 0, Cost: 0) and A has the label (Order: 1, Cost: 2).

Show Answer

Problem 6. Consider any two graphs(networks) G and G' that are identical except for the costs of the links. Please answer these questions.

A. The cost of link l in graph G is c_l > 0, and the cost of the same link l in Graph G' is `k*c_l`, where k > 0 is a constant and the same scaling relationship holds for all the links. Are the shortest paths between any two nodes in the two graphs identical? Justify your answer.

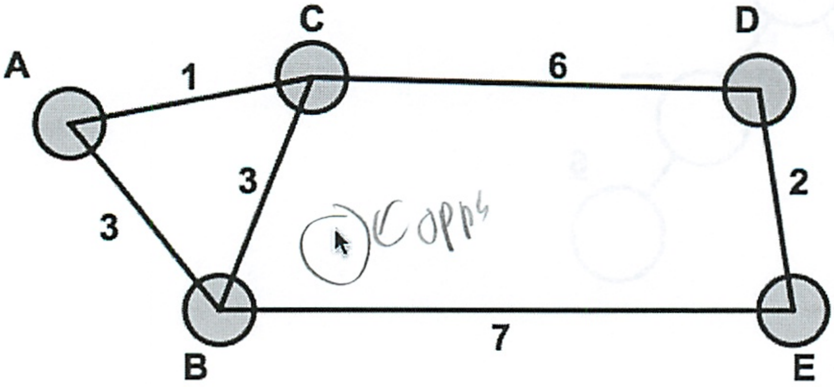
   Show Answer

B. Now suppose that the cost of a link l in G' is `k*c_l + h`, where k > 0 and h > 0 are constants. Are the shortest paths between any two nodes in the two graphs identical? Justify your answer.

   Show Answer

Problem 7. Dijkstra's algorithm

A. For the following network



an empty routing tree generated by Dijkstra's algorithm for node A (to every other node) is shown below. Fill in the missing nodes and indicate the order that each node was added and its associated cost. For reference, node C's

Problem 8. Under some conditions, a distance vector protocol finding minimum cost paths suffers from the "count-to-infinity" problem. Indicate True or False for each choice.

A. The count-to-infinity problem may arise in a distance vector protocol when the network gets disconnected.

Show Answer

B. The count-to-infinity problem may arise in a distance vector protocol even when the network never gets disconnected.

Show Answer

C. The "path vector" enhancement to a distance vector protocol always enables the protocol to converge without counting to infinity.

Show Answer

Problem 9. Ben Bitdiddle has set up a multi-hop wireless network in which he would like to find paths with high probability of packet delivery between any two nodes. His network runs a distance vector protocol similar to what you developed in the pset. In Ben's distance vector (BDV) protocol, each node maintains a metric to every destination that it knows about in the network. The metric is the nodeâ€™s estimate of the packet success probability along the path between the node and the destination.

The packet success proba bility along a link or path is defined as 1 minus the packet loss probability along the corresponding link or path. Each node uses the periodic HELLO messages sent by each of its neighbors to estimate the packet loss probability of the link from each neighbor. You may assume that the link loss probabilities are symmetric; i.e., the loss probability of the link from node A to node B is the same as from B to A. Each link L maintains its loss probability in the variable `L.lossprob` and `0 < L.lossprob < 1`.

A. The key pieces of the Python code for each node's `integrate()` function in BDV is given below. It has three missing blanks. Please fill them in so that the protocol will eventually converge without routing loops to the correct metric at each node. The variables are the same as in the pset: `self.routes` is the dictionary of routing entries (mapping destinations to links), `self.getlink(fromnode)` returns the link connecting the node self to the node fromnode, and the `integrate` procedure runs whenever the node receives an advertisement (`adv`) from node fromnode. As in the pset, `adv` is a list of (destination, metric) tuples. In the code below, `self.metric` is a dictionary storing the nodeâ€™s current estimate of the routing metric (i.e., the packet success probability) for each known destination. Please fill in the missing code.

```
# Process an advertisement from a neighboring node in BDV

def integrate(self, fromnode, adv):
  L = self.getlink(fromnode)
  for (dest, metric) in adv:
    my_metric = ______________________________
    if (dest not in self.routes
        or self.metric[dest] _____ my_metric
        or _________________________________):
      self.routes[dest] = L
      self.metric[dest] = my_metric

  # rest of integrate() not shown
```

Show Answer

Ben wants to try out a link-state protocol now. During the flooding step, each node sends out a link-state advertisement comprising its address, an incrementing sequence number, and a list of tuples of the form `(neighbor, lossprob)`, where the lossprob is the estimated loss probability to the `neighbor`.

B. Why does the link-state advertisement include a sequence number?

Show Answer

Ben would like to reuse, without modification, his implementation of Dijkstra's shortest paths algorithm from the pset, which takes a map in which the links have non-negative costs and produces a path that minimizes the sum of the costs of the links on the path to each destination.

C. Ben has to transform the lossprob information from the LSA to produce link costs so that he can use his Dijkstra implementation without any changes. Which of these transformations will accomplish this goal? Choose the BEST answer.

a. Use `lossprob` as the link cost.
b. Use `-1/log(1-lossprob)` as the link cost.
c. Use `log(1/(1-lossprob))` as the link cost.
d. Use `log(1 - lossprob)` as the link cost.

Show Answer

Problem 10. We studied a few principles for designing networks in 6.02.

A. State one significant difference between a circuit-switched and a packet-switched network.

Show Answer

B. Why does topological addressing enable large networks to be built?

Show Answer

C. Give one difference between what a switch does in a packet-switched network and a circuit-switched network.

Show Answer

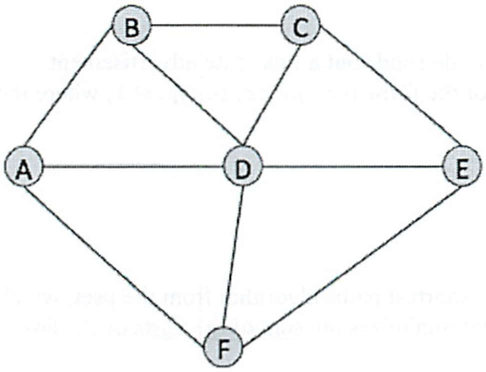
Problem 11. Eager B. Eaver implements distance vector routing in his network in which the links all have arbitrary positive costs. In addition, there are at least two paths between any two nodes in the network. One node, u, has an erroneous implementation of the integration step: it takes the advertised costs from each neighbor and picks the route corresponding to the minimum advertised cost to each destination as its route to that destination, *without adding the link cost to the neighbor*. It breaks any ties arbitrarily. All the other nodes are implemented correctly.

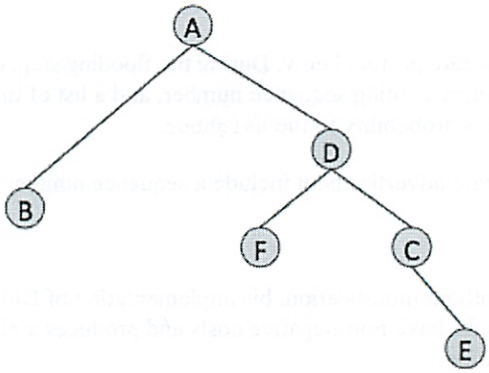
Let's use the term "correct route" to mean the route that corresponds to the minimum-cost path. Which of the following statements are true of Eager's network?

a. Only u may have incorrect routes to any other node.
b. Only u and u's neighbors may have incorrect routes to any other node.
c. In some topologies, all nodes may have correct routes.
d. Even if no HELLO or advertisements packets are lost and no link or node failures occur, a routing loop may occur.

Show Answer

Problem 12. Alyssa P. Hacker is trying to reverse engineer the trees produced by running Dijkstra's shortest paths algorithm at the nodes in the network shown in the picture on the left, below. She doesn't know the link costs, but knows that they are all positive. All link costs are symmetric (the same in both directions). She also knows that there is exactly one minimum-cost path between any pair of nodes in this network.





She discovers that the routing tree computed by Dijkstra's algorithm at node A looks like the picture on the right, above. Note that the exact order in which the nodes get added in Dijkstra's algorithm is not obvious from this picture.

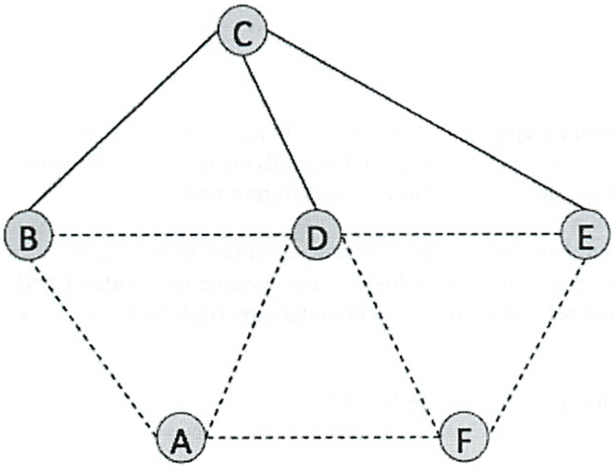
A. Which of A's links has the highest cost? If there could be more than one, tell us what they are.

Show Answer

B. Which of A's links has the lowest cost? If there could be more than one, tell us what they are.

Show Answer

Alyssa now inspects node C, and finds that it looks like the picture below. She is sure that the bold (not dashed) links belong to the shortest path tree from node C, but is not sure of the dashed links.



C. List all the dotted links that are *guaranteed* to be on the routing tree at node C.

Show Answer

D. List all the dotted links that are *guaranteed* not to be (i.e., surely not) on the routing tree at node C.

Show Answer

Problem 13. Consider a network running the link-state routing protocol as described in lecture and on the pset. How many copies of any given LSA are received by a given node in the network?

Show Answer

Problem 14. In implementing Dijkstra's algorithm in the link-state routing protocol at node u, Louis Reasoner first sets the route for each directly connected node v to be the link connecting u to v. Louis then implements the rest of the algorithm correctly, aiming to produce minimum-cost routes, but does not change the routes to the directly connected nodes. In this network, u has at least two directly connected nodes, and there is more than one path between any two nodes. Assume that all link costs are non-negative. Which of the following statements is true of u's routing table?

A. There are topologies and link costs where the *majority* of the routes to other nodes will be *incorrect*.

Show Answer

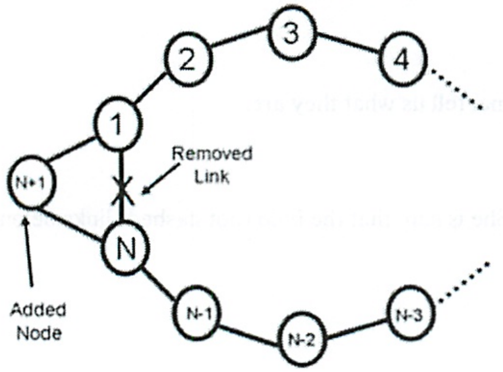
B. There are topologies and link costs where *no* routing table entry (other than from u to itself) will be *correct*.

Show Answer

C. There are topologies and link costs where *all* routing table entry (other than from u to itself) will be *correct*.

Show Answer

Problem 15. A network with N nodes and N bidirectional links is connected in a ring, and N is an even number.



The network runs a distance-vector protocol in which the advertisement step at each node runs when the local time is `T*i` seconds and the integration step runs when the local time is `T*i + T/2` seconds, (i=1,2,...). Each advertisement takes time δ to reach a neighbor. Each node has a separate clock and time is not synchronized between the different nodes.

Suppose that at some time t after the routing has converged, node N + 1 is inserted into the ring, as shown in the figure above. Assume that there are no other changes in the network topology and no packet losses. Also assume that nodes 1 and N update their routing tables at time t to include node N + 1, and then rely on their next scheduled advertisements to propagate this new information.

A. What is the minimum time before every node in the network has a route to node N + 1?

Show Answer

B. What is the maximum time before every node in the network has a route to node N + 1?

Show Answer

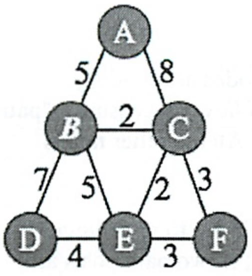
Problem 16. Louis Reasoner implements the link-state routing protocol discussed in 6.02 on a best-effort network with a non-zero packet loss rate. In an attempt to save bandwidth, instead of sending link-state advertisements periodically, each node sends an advertisement *only if* one of its links fails or when the cost of one of its links changes. The rest of the protocol remains unchanged. Will Louis' implementation always converge to produce correct routing tables on all the nodes?

Show Answer

Problem 17. Consider a network implementing minimum-cost routing using the distance-vector protocol. A node, S, has k neighbors, numbered 1 through k, with link cost c_i to neighbor i (all links have symmetric costs). Initially, S has no route for destination D. Then, S hears advertisements for D from each neighbor, with neighbor i advertising a cost of p_i. The node integrates these k advertisements. What is the cost for destination D in S's routing table after the integration?

Show Answer

Problem 18. Consider the network shown in the picture below. Each node implements Dijkstra's shortest path algorithm using the link costs shown in the picture.



A. Initially, node B's routing table contains only one entry, for itself. When B runs Dijkstra's algorithm, in what order are nodes added to the routing table? List all possible answers.

Show Answer

B. Now suppose the link cost for one of the links changes but all costs remain non-negative. For each change in link cost listed below, state whether it is possible for the route at node B (i.e., the link used by B) for any destination to change, and if so, name the destination(s) whose routes may change.

a. The cost of link(A, C) increases.

Show Answer

b. The cost of link(A, C) decreases.

Show Answer

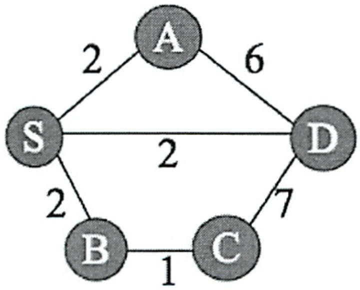
c. The cost of link(B, C) increases.

Show Answer

d. The cost of link(B, C) decreases.

Show Answer

Problem 19. Alyssa P. Hacker implements the 6.02 distance-vector protocol on the network shown below. Each node has its own local clock, which may not be synchronized with any other node's clock. Each node sends its distance-vector advertisement every 100 seconds. When a node receives an advertisement, it immediately integrates it. The time to send a message on a link and to integrate advertisements is negligible. No advertisements are lost. There is no HELLO protocol in this network.



A. At time 0, all the nodes except D are up and running. At time 10 seconds, node D turns on and immediately sends a route advertisement for itself to all its neighbors. What is the minimum time at which each of the other nodes is guaranteed to have a correct routing table entry corresponding to a minimum-cost path to reach D? Justify your answers.

Show Answer

B. If every node sends packets to destination D, and to no other destination, which link would carry the most traffic?

Show Answer

Alyssa is unhappy that one of the links in the network carries a large amount of traffic when all the nodes are sending packets to D. She decides to overcome this limitation with Alyssa's Vector Protocol (AVP). In AVP, *S lies*, advertising a "path cost" for destination D that is different from the sum of the link costs along the path used to reach D. All the other nodes implement the standard distance-vector protocol, not AVP.

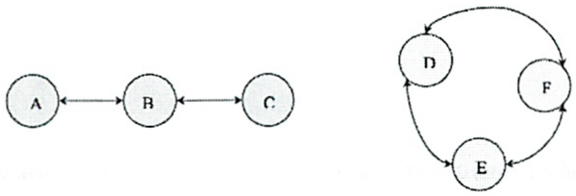
C. What is the smallest numerical value of the cost that S should advertise for D along each of its links, to *guarantee* that only its own traffic for D uses its direct link to D? Assume that all advertised costs are integers; if two path costs are equal, one can't be sure which path will be taken.

Show Answer

Show All Answers Hide All Answers

6.02 Tutorial Problems: Routing

Problem 1. Consider the following networks: network I (containing nodes A, B, C) and network II (containing nodes D, E, F).



A. The Distance Vector Protocol described in class is used in both networks. Assume advertisements are sent every 5 time steps, all links are fully functional and there is no delay in the links. Nodes take zero time to process advertisements once they receive them. The HELLO protocol runs in the background every time step in a way that any changes in link connectivity are reflected in the next DV advertisement. We count time steps from t=0 time steps.

Please fill in the following table:

Event	Number of time steps
A's routing table has an entry for B	
A's routing table has an entry for C	
D's routing table has an entry for E	
F's routing table has an entry for D	

Hide Answer

A's routing table has an entry for B: 5 time steps
A's routing table has an entry for C: 10 time steps
D's routing table has an entry for E: 5 time steps
F's routing table has an entry for D: 5 time steps

B. Now assume the link B-C fails at t = 51 and link D-E fails at t = 71 time steps. Please fill in this table:

Event	Number of time steps
B's *advertisements* reflect that C is unreachable	
A's routing table reflects C is unreachable	
D's routing table reflects a new route for E	

Hide Answer

B's advertisements reflect that C is unreachable: 55 time steps
A's routing table reflects C is unreachable: 55 time steps
D's routing table reflects a new route for E: 75 time steps

Problem 2. Alyssa P. Hacker manages MIT's internal network that runs link-state routing. She wants to experiment with a few possible routing strategies. Of all possible paths available to a particular destination at a node, a routing strategy specifies the path that must be picked to create a routing table entry. Below is the name Alyssa has for each strategy and a brief description of how it works.

> MinCost: Every node picks the path that has the smallest sum of link costs along the path. (This is the minimum cost routing you implemented in the lab).
>
> MinHop: Every node picks the path with the smallest number of hops (irrespective of what the cost on the links is).
>
> SecondMinCost: Every node picks the path with the second lowest sum of link costs. That is, every node picks the second best path with respect to path costs.
>
> MinCostSquared: Every node picks the path that has the smallest sum of squares of link costs along the path.

Assume that sufficient information (e.g., costs, delays, bandwidths, and loss probabilities of the various links) is exchanged in the link state advertisements, so that every node has complete information about the entire network and can correctly implement the strategies above. You can also assume that a link's properties don't change, e.g., it doesn't fail.

A. Help Alyssa figure out which of these strategies will work correctly, and which will result in routing with loops. In case of strategies that do result in routing loops, come up with an example network topology with a routing loop to convince Alyssa.

Hide Answer

Answer: All except SecondMinCost will work fine.

To see why SecondMinCost will not work: consider the triangle topology with 3 nodes A, B, D, and equal cost on all the links. The second route at A to D is via B, and the second best route at B to D is via A, resulting in a routing loop.

B. How would you implement MinCostSquared in a distance-vector protocol?

Hide Answer

To implement MinCostSquared, your integrate_announcement routine would add the square of the link cost (instead of just the link cost) to any route costs advertised over that link.

Problem 3. Which of the following tasks does a router R in a packet-switched network perform when it gets a packet with destination address D? Indicate True or False for each choice.

A. R looks up D in its routing table to determine the outgoing link.

Hide Answer

True.

B. R sends out a HELLO packet or a routing protocol advertisement to its neighbors.

Hide Answer

False.

C. R calculates the minimum-cost route to destination D.

Hide Answer

False.

D. R may discard the packet.

Hide Answer

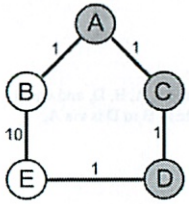
True.

Problem 4. Alice and Bob are responsible for implementing Dijkstra's algorithm at the nodes in a network running a link-state protocol. On her nodes, Alice implements a minimum-cost algorithm. On his nodes, Bob implements a "shortest number of hops" algorithm. Give an example of a network topology with 4 or more nodes in which a routing loop occurs with Alice and Bob's implementations running simultaneously in the same network. Assume that there are no failures.

(Note: A routing loop occurs when a group of k ≥ 1 distinct nodes, n_0 , n_1 , n_2 , ..., n_(k-1) have routes such that n_i's next-hop (route) to a destination is n_(i+1 mod k).

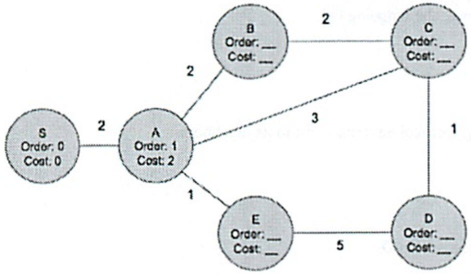
Hide Answer

In the picture below, the grey nodes (A in particular) run Bob's algorithm (shortest number of hops), while the white nodes (B in particular) run Alice's (minimum-cost).



Suppose the destination is E. A will pick B as its next hop because ABE is the shortest path. B will pick A as its next hop because BACDE is the minimum-cost path (cost of 4, compared to 11 for the ABE path). The result is a routing loop ABABAB...

Problem 5. Consider the network shown below. The number near each link is its cost.



We're interested in finding the shortest paths (taking costs into account) from S to every other node in the network. What is the result of running Dijkstra's shortest path algorithm on this network? To answer this question, near each node, list a pair of numbers: The first element of the pair should be the order, or the iteration of the algorithm in which the node is picked. The second element of each pair should be the shortest path cost from S to that node.

To help you get started, we've labeled the first couple of nodes: S has a label (Order: 0, Cost: 0) and A has the label (Order: 1, Cost: 2).

Hide Answer

A: order = 1, cost = 2
E: order = 2, cost = 3
B: order = 3, cost = 4
C: order = 4, cost = 5
D: order = 5, cost = 6

Problem 6. Consider any two graphs(networks) G and G' that are identical except for the costs of the links. Please answer these questions.

A. The cost of link l in graph G is c_l > 0, and the cost of the same link l in Graph G' is `k*c_l`, where k > 0 is a constant and the same scaling relationship holds for all the links. Are the shortest paths between any two nodes in the two graphs identical? Justify your answer.

   Hide Answer

   Yes, they're identical. Scaling all the costs by a constant factor doesn't change their relative size.

B. Now suppose that the cost of a link l in G' is `k*c_l + h`, where k > 0 and h > 0 are constants. Are the shortest paths between any two nodes in the two graphs identical? Justify your answer.

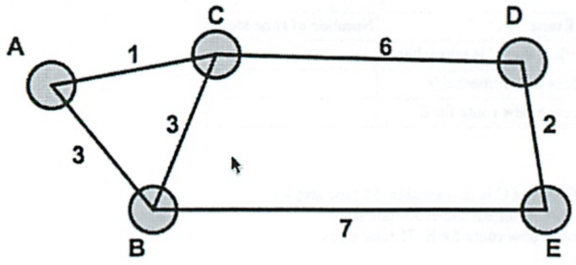
   Hide Answer

   No, they're not necessarily identical. Consider two paths between nodes A and B in graph G. One path takes 3 hops, each of cost 1, for a total cost of 3. The other path takes 1 hop, with a cost of 4. In this case, the shortest path between nodes A and B is the first one.

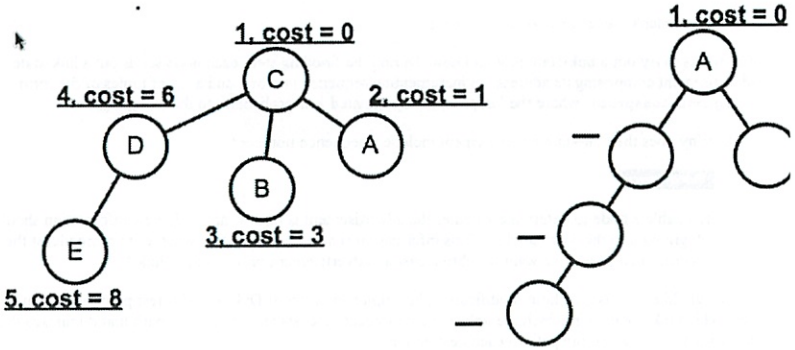
   Consider k=1 and h=1 and compute the costs and shortest paths in G'. Now the 3-hop path has cost 6 and the 1-hop path has cost 5. In G' the shortest path is the second path.

Problem 7. Dijkstra's algorithm

A. For the following network

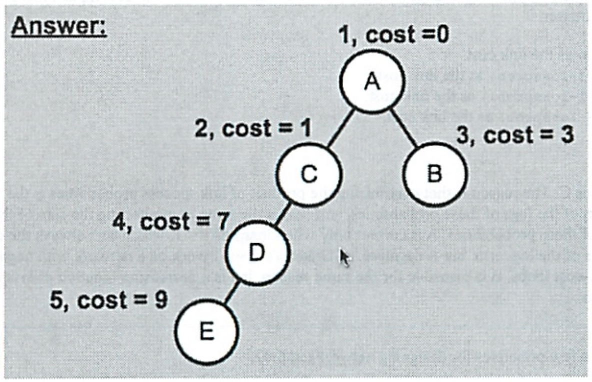


an empty routing tree generated by Dijkstra's algorithm for node A (to every other node) is shown below. Fill in the missing nodes and indicate the order that each node was added and its associated cost. For reference, node C's completed routing tree is shown as well.

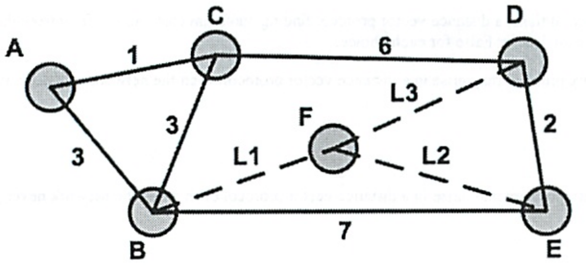


Hide Answer

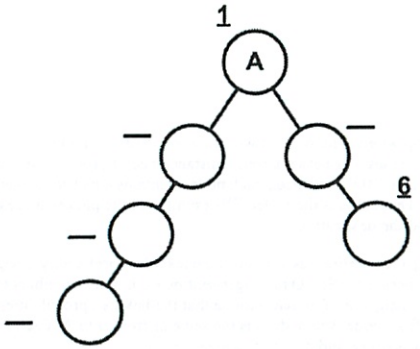
Answer:



B. Now assume that node F has been added to the network along with links L1, L2 and L3.



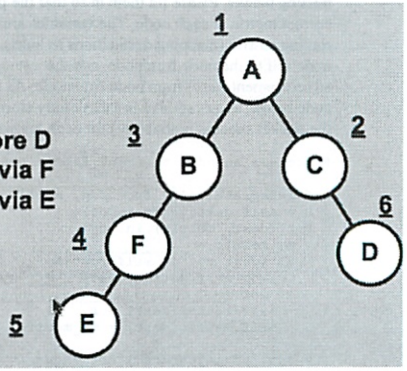
What are the constraints on L1, L2 and L3 such that node A's routing tree matches the topology shown below, and it is known that node F is not the last node added when using Dijkstra's algorithm?



Hide Answer

Answer:
The resulting graph must be:

In order for that to occur:
L1+L2 < 4 so that E is added before D
L1+L3 > 4 so that D is not added via F
L1+L2 > 2 so that D is not added via E
Thus, the real constraints are:
L1 + L2 = 3
L1 + L3 > 4



Problem 8. Under some conditions, a distance vector protocol finding minimum cost paths suffers from the "count-to-infinity" problem. Indicate True or False for each choice.

A. The count-to-infinity problem may arise in a distance vector protocol when the network gets disconnected.

Hide Answer

True.

B. The count-to-infinity problem may arise in a distance vector protocol even when the network never gets disconnected.

Hide Answer

False.

C. The "path vector" enhancement to a distance vector protocol always enables the protocol to converge without counting to infinity.

Hide Answer

True.

Problem 9. Ben Bitdiddle has set up a multi-hop wireless network in which he would like to find paths with high probability of packet delivery between any two nodes. His network runs a distance vector protocol similar to what you developed in the pset. In Ben's distance vector (BDV) protocol, each node maintains a metric to every destination that it knows about in the network. The metric is the nodeâ€™s estimate of the packet success probability along the path between the node and the destination.

The packet success proba bility along a link or path is defined as 1 minus the packet loss probability along the corresponding link or path. Each node uses the periodic HELLO messages sent by each of its neighbors to estimate the packet loss probability of the link from each neighbor. You may assume that the link loss probabilities are symmetric; i.e., the loss probability of the link from node A to node B is the same as from B to A. Each link L maintains its loss probability in the variable `L.lossprob` and `0 < L.lossprob < 1`.

A. The key pieces of the Python code for each node's `integrate()` function in BDV is given below. It has three missing blanks. Please fill them in so that the protocol will eventually converge without routing loops to the correct metric at each node. The variables are the same as in the pset: `self.routes` is the dictionary of routing entries (mapping destinations to links), `self.getlink(fromnode)` returns the link connecting the node self to the node fromnode, and the `integrate` procedure runs whenever the node receives an advertisement (`adv`) from node fromnode. As in the pset, `adv` is a list of (destination, metric) tuples. In the code below, `self.metric` is a dictionary storing the nodeâ€™s current estimate of the routing metric (i.e., the packet success probability) for each known destination. Please fill in the missing code.

```
# Process an advertisement from a neighboring node in BDV

def integrate(self, fromnode, adv):
  L = self.getlink(fromnode)
  for (dest, metric) in adv:
    my_metric = ______________________________
    if (dest not in self.routes
        or self.metric[dest] _____ my_metric
        or _________________________________):
      self.routes[dest] = L
      self.metric[dest] = my_metric

  # rest of integrate() not shown
```

Hide Answer

First blank: `(1 - L.lossprob)*metric`

Second blank: < (<= is also fine since we said that lossprob is strictly > 0)

Third blank: `self.routes[dest] == L`

Ben wants to try out a link-state protocol now. During the flooding step, each node sends out a link-state advertisement comprising its address, an incrementing sequence number, and a list of tuples of the form `(neighbor,lossprob)`, where the lossprob is the estimated loss probability to the `neighbor`.

B. Why does the link-state advertisement include a sequence number?

Hide Answer

To enable a node to determine whether the advertisement is new or not; only new information should be integrated into the routing table. (This information is also used to decide whether to rebroadcast the advertisement, since we want to rebroadcast an advertisement only once per link.)

Ben would like to reuse, without modification, his implementation of Dijkstra's shortest paths algorithm from the pset, which takes a map in which the links have non-negative costs and produces a path that minimizes the sum of the costs of the links on the path to each destination.

C. Ben has to transform the lossprob information from the LSA to produce link costs so that he can use his Dijkstra implementation without any changes. Which of these transformations will accomplish this goal? Choose the BEST answer.

a. Use `lossprob` as the link cost.
b. Use `-1/log(1-lossprob)` as the link cost.
c. Use `log(1/(1-lossprob))` as the link cost.
d. Use `log(1 - lossprob)` as the link cost.

Hide Answer

The correct choice is C. The reason is that maximizing the product of link success probabilities is the same as maximizing the sum of the logs of these probabilities, and that is the same as minimizing the sum of the logs of the reciprocals of these probabilities. A is correct only when lossprob << 1, which isn't always the case. D is plausible because of the log term, but is negative, so Dijkstra's doesn't work on a network with negative costs with negative-cost loops. B is plausible for the same reason, but is a decreasing function of lossprob, and so can't be right.

Problem 10. We studied a few principles for designing networks in 6.02.

A. State one significant difference between a circuit-switched and a packet-switched network.

Hide Answer

In a packet-switched network, packets carry information in the header that tells the switches about the destination. Circuit-switched networks donâ€™t carry any destination information in the data frames.

The abstraction provided by a circuit-switched network is that of a dedicated link of a fixed rate; a packet-switched network provides no such guarantee to the communicating end points.

B. Why does topological addressing enable large networks to be built?

Hide Answer

It reduces the size of the routing tables and the amount of information that must be exchanged in the routing protocol.

C. Give one difference between what a switch does in a packet-switched network and a circuit-switched network.

Hide Answer

Switches in a circuit-switched network participate in a connection set-up/teardown protocol, but not in a packet-switched network.

Switches in a packet-switched network look-up the destination address of a packet during forwarding, but not in a circuit-switched network.

Problem 11. Eager B. Eaver implements distance vector routing in his network in which the links all have arbitrary positive costs. In addition, there are at least two paths between any two nodes in the network. One node, u, has an erroneous implementation of the integration step: it takes the advertised costs from each neighbor and picks the route corresponding to the minimum advertised cost to each destination as its route to that destination, *without adding the link cost to the neighbor*. It breaks any ties arbitrarily. All the other nodes are implemented correctly.

Let's use the term "correct route" to mean the route that corresponds to the minimum-cost path. Which of the following statements are true of Eager's network?

a. Only u may have incorrect routes to any other node.
b. Only u and u's neighbors may have incorrect routes to any other node.
c. In some topologies, all nodes may have correct routes.
d. Even if no HELLO or advertisements packets are lost and no link or node failures occur, a routing loop may occur.

Hide Answer

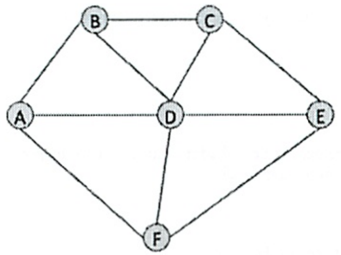
A and B are false, C is true, and D is false.

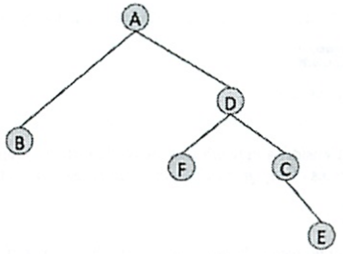
A is false because u could propagate an incorrect cost to its neighbors causing the neighbor to have an incorrect route. In fact, u's neighbors could do the same.

C is correct; a simple example is where the network is a tree, where there is exactly one path between any two nodes.

D is false; no routing loops can occur under the stated condition. We can reason by contradiction. Consider the shortest path from any node s to any other node t running the flawed routing protocol. If the path does not traverse u, no node on that path can have a loop because distance vector routing without any packet loss or failures is loop-free. Now consider the nodes for which the computed paths go through u; all these nodes are correctly implemented except for u, which means the paths between u and each of them is loop-free. Moreover, the path to u is itself loop-free because u picks one of its neighbors with smaller cost, and there is no possibility of a loop.

Problem 12. Alyssa P. Hacker is trying to reverse engineer the trees produced by running Dijkstra's shortest paths algo- rithm at the nodes in the network shown in the picture on the left, below. She doesn't know the link costs, but knows that they are all positive. All link costs are symmetric (the same in both directions). She also knows that there is exactly one minimum-cost path between any pair of nodes in this network.





She discovers that the routing tree computed by Dijkstra's algorithm at node A looks like the picture on the right, above. Note that the exact order in which the nodes get added in Dijkstra's algorithm is not obvious from this picture.

A. Which of A's links has the highest cost? If there could be more than one, tell us what they are.

Hide Answer

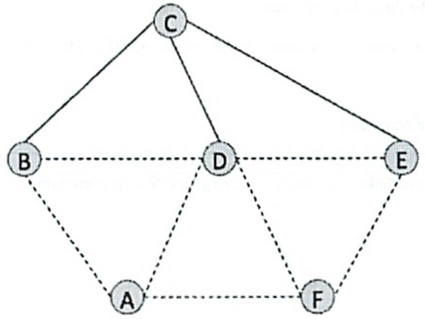
AF or AB could be the highest cost link; AD clearly has lower cost than AF.

B. Which of A's links has the lowest cost? If there could be more than one, tell us what they are.

Hide Answer

Either AB or AD could be the lowest cost link; AF clearly has a higher cost than AD.

Alyssa now inspects node C, and finds that it looks like the picture below. She is sure that the bold (not dashed) links belong to the shortest path tree from node C, but is not sure of the dashed links.



C. List all the dotted links that are *guaranteed* to be on the routing tree at node C.

Hide Answer

AD is guaranteed to be on the routing tree because AD is on the shortest path tree from node A. No other dotted link is guaranteed to be on a shortest path from C.

D. List all the dotted links that are *guaranteed* not to be (i.e., surely not) on the routing tree at node C.

Hide Answer

BD, BA, AF, DE.

Problem 13. Consider a network running the link-state routing protocol as described in lecture and on the pset. How many copies of any given LSA are received by a given node in the network?

Hide Answer

When there are no packet losses, it is equal to the number of neighbors of the node.

Problem 14. In implementing Dijkstra's algorithm in the link-state routing protocol at node u, Louis Reasoner first sets the route for each directly connected node v to be the link connecting u to v. Louis then implements the rest of the algorithm correctly, aiming to produce minimum-cost routes, but does not change the routes to the directly connected nodes. In this network, u has at least two directly connected nodes, and there is more than one path between any two nodes. Assume that all link costs are non-negative. Which of the following statements is true of u's routing table?

A. There are topologies and link costs where the *majority* of the routes to other nodes will be *incorrect*.

Hide Answer

True. For example, all the neighbors but one could have very high cost, and all the other links have low cost, so all the routes could in fact be just one link.

B. There are topologies and link costs where *no* routing table entry (other than from u to itself) will be *correct*.

Hide Answer

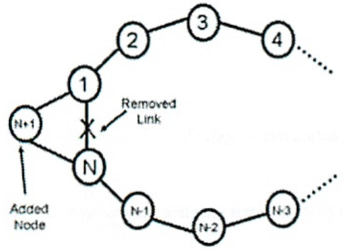
False. The lowest-cost neighbor's route will be the direct link, of course!

C. There are topologies and link costs where *all* routing table entry (other than from u to itself) will be *correct*.

Hide Answer

True. A trivial example is when all the links have equal cost.

Problem 15. A network with N nodes and N bidirectional links is connected in a ring, and N is an even number.



The network runs a distance-vector protocol in which the advertisement step at each node runs when the local time is `T*i` seconds and the integration step runs when the local time is `T*i + T/2` seconds, (i=1,2,...). Each advertisement takes time δ to reach a neighbor. Each node has a separate clock and time is not synchronized between the different nodes.

Suppose that at some time t after the routing has converged, node N + 1 is inserted into the ring, as shown in the figure above. Assume that there are no other changes in the network topology and no packet losses. Also assume that nodes 1 and N update their routing tables at time t to include node N + 1, and then rely on their next scheduled advertisements to propagate this new information.

A. What is the minimum time before every node in the network has a route to node N + 1?

Hide Answer

The key insight to observe is that each introduces a delay of at least T/2 because it takes that long between the integration and advertisement steps. Given this fact, the answer would be

```
(N/2 - 1)*(T/2 + δ)
```

However, there is a small "fence-post error" in this argument. As stated in the problem, the nodes labeled 1 and N update their routing tables at time t to include node N + 1. In the best case, these two nodes could both immediately send out advertisements, and nodes 2 and N â˜' 1 could run their integration steps immediately after receiving these advertisements. Because of that, the delay of T/2 only starts applying to the other nodes in the network. Hence, the answer is

```
(N/2 - 2)*T/2 + (N/2 - 1)*δ
```

B. What is the maximum time before every node in the network has a route to node N + 1?

Hide Answer

The key insight is that it takes in the worst case T + T/2 seconds per hop because each node may get the information about the new node *just after* it completes the previous integration step. So it has to wait T for the next integration, and then another T/2 to advertise. The correct answer is

```
(N/2 -1)*(3T/2 + δ)
```

Problem 16. Louis Reasoner implements the link-state routing protocol discussed in 6.02 on a best-effort network with a non-zero packet loss rate. In an attempt to save bandwidth, instead of sending link-state advertisements periodically, each node sends an advertisement *only if* one of its links fails or when the cost of one of its links changes. The rest of the protocol remains unchanged. Will Louis' implementation always converge to produce correct routing tables on all the nodes?

Hide Answer

No, because the LSA could be lost on all of the links connected to some one node (or more than one node), causing that node to not necessarily have correct routes. In fact this protocol doesn't converge even if packets are not lost. Consider a network where the failure of one link disconnects the network into two connected components, each with multiple nodes and links. Suppose the cost of one or more of the links in some component changes. When the network heals because the failed link recovers, the previously discon- nected component will not have the correct link costs for one or more links in the other component. However, its routes will still be correct because all paths to the other component go via the failed link. However, if we were to add another link between the two components at this time, the routing would never converge correctly.

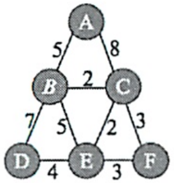
Problem 17. Consider a network implementing minimum-cost routing using the distance-vector protocol. A node, S, has k neighbors, numbered 1 through k, with link cost c_i to neighbor i (all links have symmetric costs). Initially, S has no route for destination D. Then, S hears advertisements for D from each neighbor, with neighbor i advertising a cost of p_i. The node integrates these k advertisements. What is the cost for destination D in S's routing table after the integration?

Hide Answer

This question asks for the update rule in the Bellman-Ford integration step. The cost in S's routing table for D should be set to

```
min_i{c_i + p_i}
```

Problem 18. Consider the network shown in the picture below. Each node implements Dijkstra's shortest path algorithm using the link costs shown in the picture.



A. Initially, node B's routing table contains only one entry, for itself. When B runs Dijkstra's algorithm, in what order are nodes added to the routing table? List all possible answers.

   Hide Answer

   B,C,E,A,F,D and B,C,E,F,A,D.

B. Now suppose the link cost for one of the links changes but all costs remain non-negative. For each change in link cost listed below, state whether it is possible for the route at node B (i.e., the link used by B) for any destination to change, and if so, name the destination(s) whose routes may change.

   a. The cost of link(A, C) increases.

      Hide Answer

      No effect. The edge AC is not in any shortest path.

   b. The cost of link(A, C) decreases.

      Hide Answer

      Can affect route to A. If cost_AC ≤ 3, then we can start using this edge to go to A instead of the edge BA.

   c. The cost of link(B, C) increases.

      Hide Answer

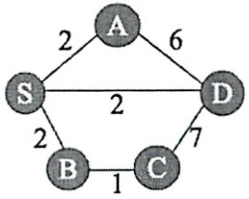
      Can affect route to C,F,E. If cost_BC ≥ 7, then we can use BE-EC to go to C instead of BC. If cost BC > 5, then we can use BE-EF to go to F. If cost_BC ≥ 3, can use BE to go to B.

   d. The cost of link(B, C) decreases.

      Hide Answer

Can affect route to D. If cost_BC ≤ 1, then we can use BC-CE-ED to go to D instead of BD.

Problem 19. Alyssa P. Hacker implements the 6.02 distance-vector protocol on the network shown below. Each node has its own local clock, which may not be synchronized with any other node's clock. Each node sends its distance-vector advertisement every 100 seconds. When a node receives an advertisement, it immediately integrates it. The time to send a message on a link and to integrate advertisements is negligible. No advertisements are lost. There is no HELLO protocol in this network.



A. At time 0, all the nodes except D are up and running. At time 10 seconds, node D turns on and immediately sends a route advertisement for itself to all its neighbors. What is the minimum time at which each of the other nodes is guaranteed to have a correct routing table entry corresponding to a minimum-cost path to reach D? Justify your answers.

   Hide Answer

   Node S: 10 seconds.
   Node A: 110 seconds.
   Node B: 110 seconds.
   Node C: 210 seconds.

   At time t = 10, D advertises to S, A, and C. They integrate this advertisement into their routing tables, so that cost(S,D) = 2, cost(A,D) = 2, cost(C,D) = 7. Note that only Sss route is correct. In the worst case, we wait 100s for the next round of advertisements. So at time t = 110, S, A, and C all advertise about D, and everyone integrates. Now cost(A, D) = 4 (via S), cost(B, D) = 4 (via S), and cost(C, D) = 7 still. A and B's routes are correct; C's is not. Finally, after 100 more seconds, another round of advertisements is sent. In particular, C hears about B's route to D, and updates cost(C, D) = 5 (via B).

B. If every node sends packets to destination D, and to no other destination, which link would carry the most traffic?

   Hide Answer

   S→D. Every node's best route to D is via S.

Alyssa is unhappy that one of the links in the network carries a large amount of traffic when all the nodes are sending packets to D. She decides to overcome this limitation with Alyssa's Vector Protocol (AVP). In AVP, *S lies*, advertising a "path cost" for destination D that is different from the sum of the link costs along the path used to reach D. All the other nodes implement the standard distance-vector protocol, not AVP.

C. What is the smallest numerical value of the cost that S should advertise for D along each of its links, to *guarantee* that only its own traffic for D uses its direct link to D? Assume that all advertised costs are integers; if two path costs are equal, one can't be sure which path will be taken.

   Hide Answer

   7. S needs to advertise a high enough cost such that everyone's path to D via S will no longer be the best path.

In particular, since B's cost to D without going through S is the highest (8), S must advertise a cost so that linkcost(B, S) + advertisedcost(S, D) > 8. Hence, S advertises a cost of 7.

Show All Answers Hide All Answers

6.02 Tutorial Problems: Reliable Data Transport

Problem 1. Consider the following chain topology:

A ---- B ----- C ---- D ---- E

A is sending packets to E using a reliable transport protocol. Each link above can transmit one packet per second. There are no queues or other sources of delays at the nodes (except the transmission delay of course).

A. What is the RTT between A and E?

Show Answer

B. What is the throughput of a stop-and-wait protocol at A in the absence of any losses at the nodes?

Show Answer

C. If A decides to run a sliding window protocol, what is the optimum window size it must use? What is the throughput achieved when using this optimum window size?

Show Answer

D. Suppose A is running a sliding window protocol with a window size of four. In the absence of any losses, what is the throughput at E? What is the utilization of link B-C?

Show Answer

E. Consider a sliding window protocol running at the optimum window size found in part 3 above. Suppose nodes in the network get infected by a virus that causes them to drop packets when odd sequence numbers. The sliding window protocol starts numbering packets from sequence number 1. Assume that the sender uses a timeout of 40 seconds. The receiver buffers out-of-order packets until it can deliver them in order to the application. What is the number of packets in this buffer 35 seconds after the sender starts sending the first packet?

Show Answer

Problem 2. Ben Bitdiddle implements a reliable data transport protocol intended to provide "exactly once" semantics. Each packet has an 8-bit incrementing sequence number, starting at 0. As the connection progresses, the sender "wraps around" the sequence number once it reaches 255, going back to 0 and incrementing it for successive packets. Each packet size is S = 1000 bytes long (including all packet headers).

Suppose the link capacity between sender and receiver is C = 1 Mbyte per second and the round-trip time is R = 100 milliseconds.

A. What is the highest throughput achievable if Ben's implementation is stop-and-wait?

Show Answer

B. To improve performance, Ben implements a sliding window protocol. Assuming no packet losses, what should Ben set the window size to in order to saturate the link capacity?

Show Answer

C. Ben runs his protocol on increasingly higher-speed bottleneck links. At a certain link speed, he finds that his implementation stops working properly. Can you explain what might be happening? What threshold link speed causes this protocol to stop functioning properly?

Show Answer

Problem 3. A sender S and receiver R are connected over a network that has k links that can each lose packets. Link i has a packet loss rate of p_i in one direction (on the path from S to R) and q_i in the other (on the path from R to S). Assume that each packet on a link is received or lost independent of other packets, and that each packet's loss probability is the same as any other's (i.e., the random process causing packet losses is indepedent and identically distributed).

A. Suppose that the probability that a data packet does not reach R when sent by S is p and the probability that an ACK packet sent by R does not reach S is q. Write expressions for p and q in terms of the p_i's and q_i's.

Show Answer

B. If all p's are equal to some value $\alpha << 1$ (much smaller than 1), then what is p (defined above) approximately equal to?

Show Answer

C. Suppose S and R use a stop-and-wait protocol to communicate. What is the expected number of transmissions of a packet before S can send the next packet in sequence? Write your answer in terms of p and q (both defined above).

Show Answer

Problem 4. Consider a 40 kbit/s network link connecting the earth to the moon. The moon is about 1.5 light-seconds from earth.

A. 1 Kbyte packets are sent over this link using a stop-and-wait protocol for reliable delivery, what data transfer rate can be achieved? What is the utilization of the link?

Show Answer

B. If a sliding-window protocol is used instead, what is the smallest window size that achieves the maximum data rate? Assume that error are infrequent. Assume that the window size is set to achieve the maximum data transfer rate.

Show Answer

C. Consider a sliding-window protocol for this link with a window size of 10 packets. If the receiver has a buffer for only 30 undelivered packets (the receiver discards packets it has no room for, and sends no ACK for discarded packets), how bits of sequence number are needed?

Show Answer

Problem 5. Consider a best-effort network with variable delays and losses. Here, Louis Reasoner suggests that the receiver does not need to send the sequence number in the ACK in a correctly implemented stop-and-wait protocol, where the sender sends packet `k+1` only after the ACK for packet `k` is received. Explain whether he is correct or not.

Show Answer

Problem 6. The 802.11 (WiFi) link-layer uses a stop-and-wait protocol to improve link reliability. The protocol works as follows:

A. The sender transmits packet k + 1 to the receiver as soon as it receives an ACK for the packet k.

B. After the receiver gets the entire packet, it computes a checksum (CRC). The processing time to compute the CRC is Tp and you may assume that it does not depend on the packet size.

C. If the CRC is correct, the receiver sends a link-layer ACK to the sender. The ACK has negligible size and reaches the sender instantaneously.

The sender and receiver are near each other, so you can ignore the propagation delay. The bit rate is R = 54 Megabits/s, the smallest packet size is 540 bits, and the largest packet size is 5,400 bits.

What is the maximum processing time Tp that ensures that the protocol will achieve a throughput of at least 50% of the bit rate of the link in the absence of packet and ACK losses, for any packet size?

Show Answer

Problem 7. Consider a sliding window protocol between a sender and a receiver. The receiver should deliver packets reliably and in order to its application.

The sender correctly maintains the following state variables:

`unacked_pkts` -- the buffer of unacknowledged packets

`first_unacked` -- the lowest unacked sequence number (undefined if all packets have been acked)

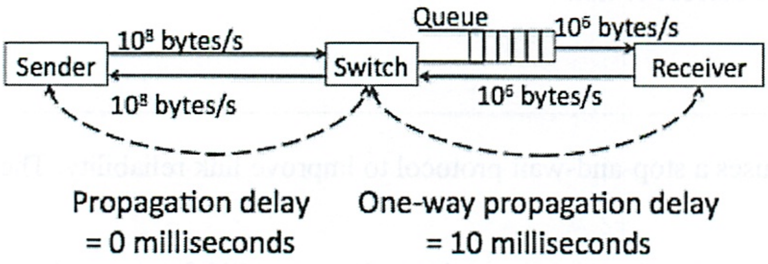
`last_unacked` -- the highest unacked sequence number (undefined if all packets have been acked)

`last_sent` -- the highest sequence number sent so far (whether acknowledged or not)

If the receiver gets a packet that is strictly larger than the next one in sequence, it adds the packet to a buffer if not already present. We want to ensure that the size of this buffer of packets awaiting delivery *never exceeds* a value $W \geq 0$. Write down the check(s) that the sender should perform before sending a new packet in terms of the variables mentioned above that ensure the desired property.

Show Answer

Problem 8. Ben decides to use the sliding window transport protocol we studied in 6.02 and implemented in the pset on the network below. The receiver sends end-to-end ACKs to the sender. The switch in the middle simply forwards packets in best-effort fashion.



Max queue size = 30 packets
Packet size = 1000 bytes
ACK size = 40 bytes
Initial sender window size = 10 packets

A. The sender's window size is 10 packets. Selecting the best answer from the choices below, at what approximate rate (in packets per second) will the protocol deliver a multi-gigabyte file from the sender to the receiver? Assume that there is no other traffic in the network and packets can only be lost because the queues overflow.
   a. Between 900 and 1000.
   b. Between 450 and 500.
   c. Between 225 and 250.
   d. Depends on the timeout value used.

Show Answer

B. You would like to double the throughput of this sliding window transport protocol running on the network shown on the previous page. To do so, you can apply one of the following techniques alone:
   a. Double the window size.
   b. Halve the propagation time of the links.
   c. Double the speed of the link between the Switch and Receiver.

   For each of the following sender window sizes, list which of the above techniques, if any, can approximately double the throughput. If no technique does the job, answer "None". There might be more than one answer for each window size, in which case you should list them all. Note that each technique works in isolation. Explain your answers.

   1. W = 10.

Show Answer

2. W = 50.

Show Answer

3. W = 30.

Show Answer

Problem 9. Consider the sliding window protocol described in lecture and implemented in the pset. The receiver sends "ACK k" when it receives a packet with sequence number `k`. Denote the window size by W. The sender's packets start with sequence number 1. Which of the following is true of a correct implementation of this protocol over a best-effort network?

A. Any new (i.e., previously unsent) packet with sequence number *greater than* W is sent by the sender if, and only if, a new (i.e., previously unseen) ACK arrives.

Show Answer

B. The sender will never send more than one packet between the receipt of one ACK and the next.

Show Answer

C. The receiver can discard any new, out-of-order packet it receives after sending an ACK for it.

Show Answer

D. Suppose that no packets or ACKs are lost and no packets are ever retransmitted. Then ACKs will arrive at the sender in non-decreasing order.

Show Answer

E. The sender should retransmit any packet for which it receives a duplicate ACK (i.e., an ACK it has received earlier).

Show Answer

Problem 10. In his haste in writing the code for the exponential weighted moving average (EWMA) to estimate the smoothed round-trip time, `srtt`, Ben Bitdiddle writes

```
srtt = alpha * r + alpha * srtt
```

where `r` is the round-trip time (RTT) sample, and `0 < alpha < 1`.

For what values of alpha does this buggy EWMA over-estimate the intended `srtt`? You may answer this question assuming any convenient non-zero sequence of RTT samples, r.

Show Answer

Problem 11. A sender S and receiver R communicate reliably over a series of links using a sliding window protocol with some window size, W packets. The path between S and R has one bottleneck link (i.e., one link whose rate bounds the throughput that can be achieved), whose data rate is C packets/second. When the window size is W, the queue at the bottleneck link is always full, with Q data packets in it. The round trip time (RTT) of the connection between S and R during this data transfer with window size W is T seconds. There are no packet or ACK losses in this case, and there are no other connections sharing this path.

A. Write an expression for W in terms of the other parameters specified above.

Show Answer

B. We would like to reduce the window size from W and still achieve high utilization. What is the minimum window size, Wmin, which will achieve 100% utilization of the bottleneck link? Express your answer as a function of C, T, and Q.

Show Answer

C. Now suppose the sender starts with a window size set to Wmin. If all these packets get acknowledged and no packet losses occur in the window, the sender increases the window size by 1. The sender keeps increasing the window size in this fashion until it reaches a window size that causes a packet loss to occur. What is the smallest window size at which the sender observes a packet loss caused by the bottleneck queue overflowing? Assume that no ACKs are lost.

Show Answer

Problem 12. A sender A and a receiver B communicate using the stop-and-wait protocol studied in 6.02. There are n links on the path between A and B, each with a data rate of R bits per second. The size of a data packet is S bits and the size of an ACK is K bits. Each link has a physical distance of D meters and the speed of signal propagation over each link is c meters per second. The total processing time experienced by a data packet and its ACK is T_p seconds. ACKs traverse the same links as data packets, except in the opposite direction on each link (the propagation time and data rate are the same in both directions of a link). There is no queueing delay in this network. Each link has a packet loss probability of p, with packets being lost independently.

What are the following four quantities in terms of the given parameters?

A. Transmission time for a data packet on one link between A and B.

Show Answer

B. Propagation time for a data packet across n links between A and B.

Show Answer

C. Round-trip time (RTT) between A and B?. (The RTT is defined as the elapsed time between the start of transmission of a data packet and the completion of receipt of the ACK sent in response to the data packetâ€™s reception by the receiver.)

Show Answer

D. Probability that a data packet sent by A will reach B.

Show Answer

Problem 13. Ben Bitdiddle gets rid of the timestamps from the packet header in the 6.02 stop-and-wait transport protocol running over a best-effort network. The network may lose or reorder packets, but it never duplicates a packet. In the protocol, the receiver sends an ACK for each data packet it receives, echoing the sequence number of the packet that was just received. The sender uses the following method to estimate the round-trip time (RTT) of the connection:

1. When the sender transmits a packet with sequence number k, it stores the time on its machine at which the packet was sent, t_k. If the transmission is a retransmission of sequence number k, then t_k is updated.

2. When the sender gets an ACK for packet k, if it has not already gotten an ACK for k so far, it observes the current time on its machine, a_k, and measures the RTT sample as a_k - t_k.

If the ACK received by the sender at time ak was sent by the receiver in response to a data packet sent at time tk, then the RTT sample a_k - t_k is said to be correct. Otherwise, it is incorrect.

Indicate which of the following statements is true:

A. If the sender never retransmits a data packet during a data transfer, then all the RTT samples produced by Benâ€™s method are correct.

Show Answer

B. If data and ACK packets are never reordered in the network, then all the RTT samples produced by Benâ€™s method are correct.

Show Answer

C. If the sender makes no spurious retransmissions during a data transfer (i.e., it only retransmits a data packet if all previous transmissions of data packets with the same sequence number did in fact get dropped before reaching the receiver), then all the RTT samples produced by Ben's method are correct.

Show Answer

Opt E. Miser implements the 6.02 stop-and-wait reliable transport protocol with one modification: being stingy, he replaces the sequence number field with a 1-bit field, deciding to reuse sequence numbers across data packets. The first data packet has sequence number 1, the second has number 0, the third has number 1, the fourth has number 0, and so on. Whenever the receiver gets a packet with sequence number s (= 0 or 1), it sends an ACK to the sender echoing s. The receiver delivers a data packet to the application if, and only if, its sequence number is different from the last one delivered, and upon delivery, updates the last sequence number delivered.

D. He runs this protocol over a best-effort network that can lose packets (with probability less than

1) or reorder them, and whose delays may be variable. Does the modified protocol always provide correct reliable, in-order delivery of a stream of packets?

Show Answer

Problem 14. Consider a reliable transport connection using the 6.02 sliding window protocol on a network path whose RTT in the absence of queueing is RTTmin = 0.1 seconds. The connection's bottleneck link has a rate of C = 100 packets per second, and the queue in front of the bottleneck link has space for Q = 20 packets.

Assume that the sender uses a sliding window protocol with fixed window size. There is no other connection on the path.

A. If the size of the window is 8 packets, then what is the throughput of the connection?

Show Answer

B. If the size of the window is 16 packets, then what is the throughput of the connection?

Show Answer

C. What is the smallest window size for which the connection's RTT exceeds RTTmin?

Show Answer

Problem 15. TCP, the standard reliable transport protocol used on the Internet, uses a sliding window. Unlike the protocol studied in 6.02, however, the size of the TCP window is variable. The sender changes the size of the window as ACKs arrive from the receiver; it does not know the best window size to use a priori.

TCP uses a scheme called *slow start* at the beginning of a new connection. Slow start has three rules, R1, R2, and R3, listed below (TCP uses some other rules too, which we will ignore).

In the following rules for slow start, the sender's current window size is W and the last in-order ACK received by the sender is A. The first packet sent has sequence number 1.

R1. Initially, set W ← 1 and A ← 0.
R2. If an ACK arrives for packet A+1, then set W←W+1, and set A←A+1.
R3. When the sender retransmits a packet after a timeout, then set W←1.

Assume that all the other mechanisms are the same as the 6.02 sliding window protocol. Data packets may be lost because packet queues overflow, but assume that packets are not reordered by the network.

We run slow start on a network with RTTmin = 0.1 seconds, bottleneck link rate = 100 packets per second, and bottleneck queue = 20 packets.

A. What is the smallest value of W at which the bottleneck queue overflows?

Show Answer

B. Sketch W as a function of time for the first 5 RTTs of a connection. The X-axis marks time in terms of multiples of the connectionâ€™s RTT. (Hint: Non-linear!)

Show Answer

Show All Answers Hide All Answers

6.02 Tutorial Problems: Reliable Data Transport

Problem 1. Consider the following chain topology:

A ---- B ----- C ---- D ---- E

A is sending packets to E using a reliable transport protocol. Each link above can transmit one packet per second. There are no queues or other sources of delays at the nodes (except the transmission delay of course).

A. What is the RTT between A and E?

Hide Answer

8 seconds

B. What is the throughput of a stop-and-wait protocol at A in the absence of any losses at the nodes?

Hide Answer

1/8 pkts/s

C. If A decides to run a sliding window protocol, what is the optimum window size it must use? What is the throughput achieved when using this optimum window size?

Hide Answer

optimum window = 8 pkts, optimum throughput = 1 pkt/s

D. Suppose A is running a sliding window protocol with a window size of four. In the absence of any losses, what is the throughput at E? What is the utilization of link B-C?

Hide Answer

throughput=0.5 pkts/s, utilization=0.5

E. Consider a sliding window protocol running at the optimum window size found in part 3 above. Suppose nodes in the network get infected by a virus that causes them to drop packets when odd sequence numbers. The sliding window protocol starts numbering packets from sequence number 1. Assume that the sender uses a timeout of 40 seconds. The receiver buffers out-of-order packets until it can deliver them in order to the application. What is the number of packets in this buffer 35 seconds after the sender starts sending the first packet?

Hide Answer

Answer: 7.

With a window size of 8, the sender sends out packets 1--8 in the first 8 seconds. But it gets back only 4 ACKs because packets 1,3,5,7 are dropped. Therefore, the sender transmits 4 more packets (9--12) in the next 8 seconds, 2 packets (13--14) in the next 8 seconds, and 1 (sequence number 15) packet in the next 8 seconds. Note that 32 seconds have elapsed so far. Now the sender gets no more ACKs because packet 15 is dropped, and it stalls till the first packet times out at time step 40. Therefore, at time 35, the sender would have transmitted 15 packets, 7 of which would have reached the receiver. But because all of these packets are out of order, the receiver's buffer would have 7 packets.

Problem 2. Ben Bitdiddle implements a reliable data transport protocol intended to provide "exactly once" semantics. Each packet has an 8-bit incrementing sequence number, starting at 0. As the connection progresses, the sender "wraps around" the sequence number once it reaches 255, going back to 0 and incrementing it for successive packets. Each packet size is S = 1000 bytes long (including all packet headers).

Suppose the link capacity between sender and receiver is C = 1 Mbyte per second and the round-trip time is R = 100 milliseconds.

A. What is the highest throughput achievable if Ben's implementation is stop-and-wait?

Hide Answer

The highest throughput for stop-and-wait is (1000 bytes)/(100ms) = 10 Kbytes/s.

B. To improve performance, Ben implements a sliding window protocol. Assuming no packet losses, what should Ben set the window size to in order to saturate the link capacity?

Hide Answer

Set the window size to the bandwidth-delay product of the link, 1 Mbyte/s * 0.1 s = 100 Kbytes.

C. Ben runs his protocol on increasingly higher-speed bottleneck links. At a certain link speed, he finds that his implementation stops working properly. Can you explain what might be happening? What threshold link speed causes this protocol to stop functioning properly?

Hide Answer

Sequence number wraparound causes his protocol to stop functioning properly. When this happens, two packets with different content but the same sequence number are in-flight at once, and so an ack for the firstt spuriously acknowledges the second as well, possibly causing data loss in Ben's "reliable" protocol. This happens when

(255 packets)(1000 bytes/1 packet) = (C bytes/s)(0.1 s)

Therefore C = 2.55 Mbytes/s.

Problem 3. A sender S and receiver R are connected over a network that has k links that can each lose packets. Link i has a packet loss rate of p_i in one direction (on the path from S to R) and q_i in the other (on the path from R to S). Assume that each packet on a link is received or lost independent of other packets, and that each packet's loss probability is the same as any other's (i.e., the random process causing packet losses is indepedent and identically distributed).

A. Suppose that the probability that a data packet does not reach R when sent by S is p and the probability that an ACK packet sent by R does not reach S is q. Write expressions for p and q in terms of the p_i's and q_i's.

Hide Answer

p = 1 - (1 - p_1)(1 - p_2)...(1 - p_k) and
q = 1 - (1 - q_1)(1 - q_2)...(1 - q_k)

B. If all p's are equal to some value $\alpha << 1$ (much smaller than 1), then what is p (defined above) approximately equal to?

Hide Answer

$p = 1 - (1 - \alpha)^k \approx 1 - (1 - k\alpha) = k\alpha$

C. Suppose S and R use a stop-and-wait protocol to communicate. What is the expected number of transmissions of a packet before S can send the next packet in sequence? Write your answer in terms of p and q (both defined above).

Hide Answer

The probability of a packet reception from S to R is 1-p and the probability of an ACK reaching S given that R sent an ACK is 1-q. The sender moves from sequence number k to k + 1 if the packet reaches and the ACK arrives. That happens with probability (1-p)(1-q). The expected number of transmissions for such an event is therefore equal to 1/((1-p)(1-q)).

Problem 4. Consider a 40 kbit/s network link connecting the earth to the moon. The moon is about 1.5 light-seconds from earth.

A. 1 Kbyte packets are sent over this link using a stop-and-wait protocol for reliable delivery, what data transfer rate can be achieved? What is the utilization of the link?

Hide Answer

Stop-and-wait sends 1 packet per round-trip-time so the data transfer rate is 1 Kbyte/3 seconds = 333 bytes/s = 2.6 Kbit/s. The utilization is 2.6/40 = 6.5%.

The estimate above omits the transmission time of the packet. If we include the transmission time (8 kbit/(40 kbit/s) = 0.2 s), the result is 1 kbyte/3.2 seconds = 312 bytes/s = 2.5 Kbits/s.

B. If a sliding-window protocol is used instead, what is the smallest window size that achieves the maximum data rate? Assume that error are infrequent. Assume that the window size is set to achieve the maximum data transfer rate.

Hide Answer

Achieving full rate requires a send window of at least

bandwidth-delay product = 5 packets/s * 3 s = 15 packets.

C. Consider a sliding-window protocol for this link with a window size of 10 packets. If the receiver has a buffer for only 30 undelivered packets (the receiver discards packets it has no room for, and sends no ACK for discarded packets), how bits of sequence number are needed?

Hide Answer

The window size determines the number of unacknowledged packets the transmitter will send before stalling, but there can be arbitrarily many acknowledged but undelivered (because of one lost packet) packets at the receiver. But if only 30 packets are held at the receiver, after which it stops acknowledging packets except the one it's waiting for, the total number of packets in transit or sitting in the receivers buffer is at most 40.

So a 6-bit sequence number will be sufficent to ensure that all unack'ed and undelivered packets have a unique sequence number (avoiding the sequence number wrap-around problem).

Problem 5. Consider a best-effort network with variable delays and losses. Here, Louis Reasoner suggests that the receiver does not need to send the sequence number in the ACK in a correctly implemented stop-and-wait protocol, where the sender sends packet `k+1` only after the ACK for packet `k` is received. Explain whether he is correct or not.

Hide Answer

(Not surprisingly,) Louis is wrong. Imagine that the sender sends packet `k` and then retransmits `k`. However, the original transmission and the retransmission get through to the receiver. The receiver sends an ACK for `k` when it gets the original transmission, and in response the sender sends packet `k+1`. Now, when the sender gets an ACK, it cannot tell whether the ACK was for packet `k` (the retransmission), or for packet `k+1`!

Problem 6. The 802.11 (WiFi) link-layer uses a stop-and-wait protocol to improve link reliability. The protocol works as follows:

A. The sender transmits packet k + 1 to the receiver as soon as it receives an ACK for the packet k.

B. After the receiver gets the entire packet, it computes a checksum (CRC). The processing time to compute the CRC is Tp and you may assume that it does not depend on the packet size.

C. If the CRC is correct, the receiver sends a link-layer ACK to the sender. The ACK has negligible size and reaches the sender instantaneously.

The sender and receiver are near each other, so you can ignore the propagation delay. The bit rate is R = 54 Megabits/s, the smallest packet size is 540 bits, and the largest packet size is 5,400 bits.

What is the maximum processing time Tp that ensures that the protocol will achieve a throughput of at least 50% of the bit rate of the link in the absence of packet and ACK losses, for any packet size?

When W = 10, the throughput is about 476 packets/s. If we double the window size, throughput would double to 952 packet/s. If we reduce the propagation time of the links, throughput would roughly double as well. The new throughput would still be smaller than the bottleneck capacity of 1000 packets/s.

2. W = 50.

Hide Answer

C.

When W = 50, throughput is already 1000 packets/s. At this stage, doubling the window or halving the RTT does not increase the throughput. If we double the speed of the link between the Switch and Receiver, the bottleneck becomes 2000 packets/s. A window size of 50 packets over an RTT of 20 or 21 milliseconds has a throughput of more than 2000 packet/s. Hence, doubling the bottleneck link speed will double the throughput when W = 50 packets. With a queue size of 30 packets and a window size of 50, the initial window of packets sent back-to-back would indeed cause the queue to overflow. However, that doesn't cause the throughput to drop in the steady state, so for a long data transfer, the throughput will be as described above.

3. W = 30.

Hide Answer

None.

When W = 30, throughput is already 1000 packets/s. Now, if we double the window or halve the RTT, the throughput won't change. An interesting situation occurs when we double the link speed, because the bottleneck link would now be capable of delivering 2000 packets/s. But our window size is 30 and RTT about 20 milliseconds, giving a throughput of about 1500 packets/s (or if we use 21 milliseconds, we get 1428 packet/s). That's an improvement of about 50%, far from the doubling we wanted. None of the techniques work.

Problem 9. Consider the sliding window protocol described in lecture and implemented in the pset. The receiver sends "ACK k" when it receives a packet with sequence number `k`. Denote the window size by W. The sender's packets start with sequence number 1. Which of the following is true of a correct implementation of this protocol over a best-effort network?

A. Any new (i.e., previously unsent) packet with sequence number *greater than* W is sent by the sender if, and only if, a new (i.e., previously unseen) ACK arrives.

Hide Answer

True.

B. The sender will never send more than one packet between the receipt of one ACK and the next.

Hide Answer

False. The sender could time-out and retransmit.

C. The receiver can discard any new, out-of-order packet it receives after sending an ACK for it.

Hide Answer

False. The sender thinks the receiver has delivered this packet to the application.

D. Suppose that no packets or ACKs are lost and no packets are ever retransmitted. Then ACKs will arrive at the sender in non-decreasing order.

Hide Answer

False. Packets or ACKs could get reordered in the network.

E. The sender should retransmit any packet for which it receives a duplicate ACK (i.e., an ACK it has received earlier).

Hide Answer

False. Duplicate ACKs can be ignored by the sender.

Problem 10. In his haste in writing the code for the exponential weighted moving average (EWMA) to estimate the smoothed round-trip time, `srtt`, Ben Bitdiddle writes

```
srtt = alpha * r + alpha * srtt
```

where `r` is the round-trip time (RTT) sample, and `0 < alpha < 1`.

For what values of alpha does this buggy EWMA over-estimate the intended `srtt`? You may answer this question assuming any convenient non-zero sequence of RTT samples, r.

Hide Answer

This buggy EWMA over-estimates the true `srtt` when alpha > 0.5. The true `srtt` is equal to

```
alpha * r + (1-alpha) * srtt
```

So

```
alpha * r + alpha * srtt > alpha * r + (1-alpha) * srtt
```

implies `alpha > 0.5`.

Problem 11. A sender S and receiver R communicate reliably over a series of links using a sliding window protocol with some window size, W packets. The path between S and R has one bottleneck link (i.e., one link whose rate bounds the throughput that can be achieved), whose data rate is C packets/second. When the window size is W, the queue at the bottleneck link is always full, with Q data packets in it. The round trip time (RTT) of the connection between S and R during this data transfer with window size W is T seconds. There are no packet or ACK losses in this case, and there are no other connections sharing this path.

Hide Answer

Because T_p is independent of packet size, and smaller packets have a lower transmission time over the link, what matters for this question is the processing time for the smallest packet. The maximum throughput of the stop-and-wait protocol is 1 packet every round-trip time (RTT), which in our case is the sum of the transmission time and T_p. The transmission time for a 540 bit packet at 54 Megabits/s is 10 microseconds. Hence, if Tp* is the maximum allowable processing time, we have:

```
540 bits/(10 microseconds + Tp*) = 27 Megabits/s,
```

giving us Tp* = 10 microseconds.

Problem 7. Consider a sliding window protocol between a sender and a receiver. The receiver should deliver packets reliably and in order to its application.

The sender correctly maintains the following state variables:

`unacked_pkts` -- the buffer of unacknowledged packets

`first_unacked` -- the lowest unacked sequence number (undefined if all packets have been acked)

`last_unacked` -- the highest unacked sequence number (undefined if all packets have been acked)

`last_sent` -- the highest sequence number sent so far (whether acknowledged or not)

If the receiver gets a packet that is strictly larger than the next one in sequence, it adds the packet to a buffer if not already present. We want to ensure that the size of this buffer of packets awaiting delivery *never exceeds* a value $W \geq 0$. Write down the check(s) that the sender should perform before sending a new packet in terms of the variables mentioned above that ensure the desired property.

Hide Answer

The largest sequence number that a receiver could have *possibly* received is `last_sent`. The size of the receiver buffer can become as large as `last_sent - first_unacked`. One might think that we need to add 1 to this quantity, but observe that the only reason any packets get added to the buffer is when some packet is lost (i.e., at least one of the packets in the sender's unacked buffer must have been lost).

We also need to handle the case when all the packets sent by the sender have been acknowledged -- clearly, in this case, the sender should be able to send data.

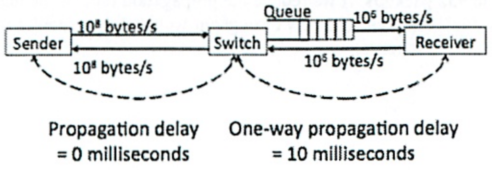
Hence, if the sender sends a new packet only if

```
if len(unacked_packets) == 0 or last_sent - first_unacked < W
```

the desired requirement is satisfied.

Problem 8. Ben decides to use the sliding window transport protocol we studied in 6.02 and implemented in the pset on the network below. The receiver sends end-to-end ACKs to the sender.

The switch in the middle simply forwards packets in best-effort fashion.



Max queue size = 30 packets
Packet size = 1000 bytes
ACK size = 40 bytes
Initial sender window size = 10 packets

A. The sender's window size is 10 packets. Selecting the best answer from the choices below, at what approximate rate (in packets per second) will the protocol deliver a multi-gigabyte file from the sender to the receiver? Assume that there is no other traffic in the network and packets can only be lost because the queues overflow.
   a. Between 900 and 1000.
   b. Between 450 and 500.
   c. Between 225 and 250.
   d. Depends on the timeout value used.

Hide Answer

Choice b. The RTT, which is the time taken for a single packet to reach the receiver and the ACK to return, is about 20 milliseconds plus the transmission time, which is about 1 millisecond (1000 bytes at a rate of 1 Megabyte/s). Hence, the throughput is 10 packets / 21 milliseconds = 476 packets per second. If one ignored the transmission time, which is perfectly fine given the set of choices, one would estimate the throughput to be about 500 packets per second.

B. You would like to double the throughput of this sliding window transport protocol running on the network shown on the previous page. To do so, you can apply one of the following techniques alone:
   a. Double the window size.
   b. Halve the propagation time of the links.
   c. Double the speed of the link between the Switch and Receiver.

For each of the following sender window sizes, list which of the above techniques, if any, can approximately double the throughput. If no technique does the job, answer "None". There might be more than one answer for each window size, in which case you should list them all. Note that each technique works in isolation. Explain your answers.

1. W = 10.

Hide Answer

A and B.

Hide Answer

True. If there are no retransmissions ever made, t_k gets set once and never updated, and the ACK for k can be unambiguously associated with the corresponding packet transmission, and the RTT sample will be correct.

B. If data and ACK packets are never reordered in the network, then all the RTT samples produced by Benâ€™s method are correct.

Hide Answer

False. If the sender retransmits a packet, it can no longer unambiguously associate a packet's ACK reception with a particular transmission or retransmission of a packet with the same sequence number.

C. If the sender makes no spurious retransmissions during a data transfer (i.e., it only retransmits a data packet if all previous transmissions of data packets with the same sequence number did in fact get dropped before reaching the receiver), then all the RTT samples produced by Ben's method are correct.

Hide Answer

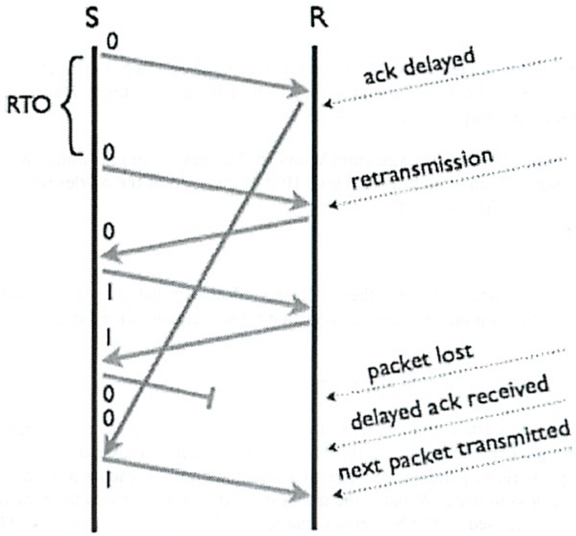
True. Given that there are no spurious retransmissions, *at most* one packet with a given sequence number, k can reach the receiver, and the sender can get at most one ACK for k. If the sender gets an ACK for k, that ACK *must* correspond to the last packet transmission of that sequence number, k. The RTT samples in this case will be produced correctly.

Opt E. Miser implements the 6.02 stop-and-wait reliable transport protocol with one modification: being stingy, he replaces the sequence number field with a 1-bit field, deciding to reuse sequence numbers across data packets. The first data packet has sequence number 1, the second has number 0, the third has number 1, the fourth has number 0, and so on. Whenever the receiver gets a packet with sequence number s (= 0 or 1), it sends an ACK to the sender echoing s. The receiver delivers a data packet to the application if, and only if, its sequence number is different from the last one delivered, and upon delivery, updates the last sequence number delivered.

D. He runs this protocol over a best-effort network that can lose packets (with probability less than 1) or reorder them, and whose delays may be variable. Does the modified protocol always provide correct reliable, in-order delivery of a stream of packets?

Hide Answer

No. For example, see the picture below.



Problem 14. Consider a reliable transport connection using the 6.02 sliding window protocol on a network path whose RTT in the absence of queueing is RTTmin = 0.1 seconds. The connection's bottleneck link has a rate of C = 100 packets per second, and the queue in front of the bottleneck link has space for Q = 20 packets.

Assume that the sender uses a sliding window protocol with fixed window size. There is no other connection on the path.

A. If the size of the window is 8 packets, then what is the throughput of the connection?

Hide Answer

The bandwidth-delay product of the connection is 10 packets (bottleneck rate times the minimum RTT). With a window size of 8, queues will not yet have built up, so the throughput is 80 packets/second.

B. If the size of the window is 16 packets, then what is the throughput of the connection?

Hide Answer

The bandwidth-delay product of the network is 10 packets, so if W >= 10, there will be 10 packets in flight. With W=16, 6 of these packets will be in the queue. The queuing delay will be 6/100 = 0.06 seconds. Then RTT = RTTmin + queuing delay = .1 + .06 = 0.16 and the throughput is W/RTT = 16/.16 = 100 pkts/s.

A. Write an expression for W in terms of the other parameters specified above.

Hide Answer

W = C*T. Note that some students may interpret T as the RTT without any queueing. That's wrong, but we ought to still give them credit as long as they have consistently made this mistake in all the parts. With this interpretation, W = CT + Q.

B. We would like to reduce the window size from W and still achieve high utilization. What is the minimum window size, Wmin, which will achieve 100% utilization of the bottleneck link? Express your answer as a function of C, T, and Q.

Hide Answer

Clearly, W = Wmin + Q, where Wmin is the smallest window size that gives 100% utilization. A smaller window than that would keep the network idle some fraction of the time. Hence, Wmin = C*T - Q.

With the flawed interpretation of T, Wmin = C*T.

C. Now suppose the sender starts with a window size set to Wmin. If all these packets get acknowledged and no packet losses occur in the window, the sender increases the window size by 1. The sender keeps increasing the window size in this fashion until it reaches a window size that causes a packet loss to occur. What is the smallest window size at which the sender observes a packet loss caused by the bottleneck queue overflowing? Assume that no ACKs are lost.

Hide Answer

Packets start getting dropped when the window size is W+1, i.e., when it is equal to C*T+1.

With the flawed interpretation of T, the window size at which packets start being dropped is W+1 = C*T + Q + 1.

Problem 12. A sender A and a receiver B communicate using the stop-and-wait protocol studied in 6.02. There are n links on the path between A and B, each with a data rate of R bits per second. The size of a data packet is S bits and the size of an ACK is K bits. Each link has a physical distance of D meters and the speed of signal propagation over each link is c meters per second. The total processing time experienced by a data packet and its ACK is T_p seconds. ACKs traverse the same links as data packets, except in the opposite direction on each link (the propagation time and data rate are the same in both directions of a link). There is no queueing delay in this network. Each link has a packet loss probability of p, with packets being lost independently.

What are the following four quantities in terms of the given parameters?

A. Transmission time for a data packet on one link between A and B.

Hide Answer

S/R. Each data packet has size S bits, and the speed of the link is R bits per second.

B. Propagation time for a data packet across n links between A and B.

Hide Answer

nD/C . Total distance to be travelled is nD since each link has length D meters, and there are n such links. The propagation speed is C meters/second.

C. Round-trip time (RTT) between A and B?. (The RTT is defined as the elapsed time between the start of transmission of a data packet and the completion of receipt of the ACK sent in response to the data packetâ€™s reception by the receiver.)

Hide Answer

```
nS/R + nK/r + 2nD/C + T_p
```

We need to consider the following times:

- Transmit data across n links: nS/R using result from part A.
- Transmit ACK across n links: nK/R also using result from part A.
- Propagate data across n links and ACKS across n links: 2nD/C
- Total time to process the data and the ACK: T_p

D. Probability that a data packet sent by A will reach B.

Hide Answer

$(1- p)^n$. Probability of loss in a link is p, so probability of no loss in a link is 1-p. Since link losses are independent, probability of no loss in n links is $(1-p)^n$. No loss in n links means the data gets from A to B successfuly.

Problem 13. Ben Bitdiddle gets rid of the timestamps from the packet header in the 6.02 stop-and-wait transport protocol running over a best-effort network. The network may lose or reorder packets, but it never duplicates a packet. In the protocol, the receiver sends an ACK for each data packet it receives, echoing the sequence number of the packet that was just received. The sender uses the following method to estimate the round-trip time (RTT) of the connection:

1. When the sender transmits a packet with sequence number k, it stores the time on its machine at which the packet was sent, t_k. If the transmission is a retransmission of sequence number k, then t_k is updated.
2. When the sender gets an ACK for packet k, if it has not already gotten an ACK for k so far, it observes the current time on its machine, a_k, and measures the RTT sample as a_k - t_k.

If the ACK received by the sender at time ak was sent by the receiver in response to a data packet sent at time tk, then the RTT sample a_k - t_k is said to be correct. Otherwise, it is incorrect.

Indicate which of the following statements is true:

A. If the sender never retransmits a data packet during a data transfer, then all the RTT samples produced by Benâ€™s method are correct.

C. What is the smallest window size for which the connection's RTT exceeds RTTmin?

Hide Answer

11 packets. The bandwidth-delay product is 10 packets. It's probably reasonable to accept an answer of 10 packets too.

Problem 15. TCP, the standard reliable transport protocol used on the Internet, uses a sliding window. Unlike the protocol studied in 6.02, however, the size of the TCP window is variable. The sender changes the size of the window as ACKs arrive from the receiver; it does not know the best window size to use a priori.

TCP uses a scheme called *slow start* at the beginning of a new connection. Slow start has three rules, R1, R2, and R3, listed below (TCP uses some other rules too, which we will ignore).

In the following rules for slow start, the sender's current window size is W and the last in-order ACK received by the sender is A. The first packet sent has sequence number 1.

R1. Initially, set W ← 1 and A ← 0.
R2. If an ACK arrives for packet A+1, then set W←W+1, and set A←A+1.
R3. When the sender retransmits a packet after a timeout, then set W←1.

Assume that all the other mechanisms are the same as the 6.02 sliding window protocol. Data packets may be lost because packet queues overflow, but assume that packets are not reordered by the network.

We run slow start on a network with RTTmin = 0.1 seconds, bottleneck link rate = 100 packets per second, and bottleneck queue = 20 packets.

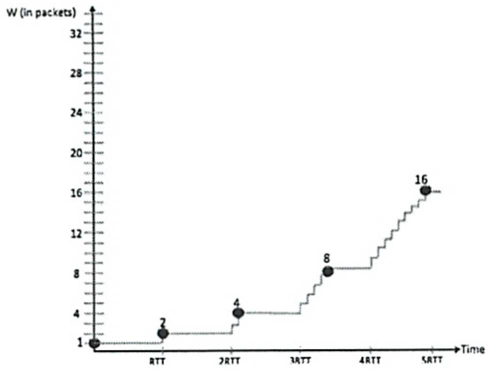
A. What is the smallest value of W at which the bottleneck queue overflows?

Hide Answer

The smallest W for which the queue overflows is 10 + 20 + 1 = 31 packets. The 10 is because that's the bandwidth-delay product; the 20 is the maximum size of the queue. And we need one more packet to cause an overflow.

B. Sketch W as a function of time for the first 5 RTTs of a connection. The X-axis marks time in terms of multiples of the connectionâ€™s RTT. (Hint: Non-linear!)

Hide Answer



6.02 Problems

5/14

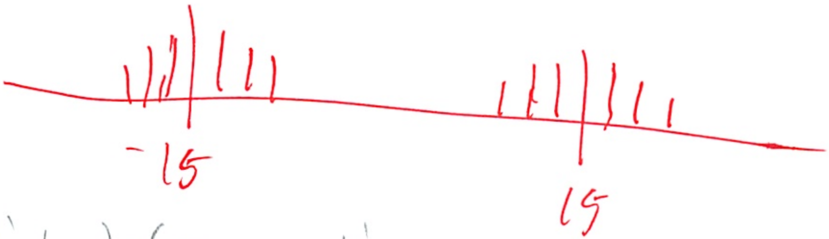
1. Modulation

a) $\cos 15 \frac{2\pi}{101} n$

Sketch spectral coeffs!?!

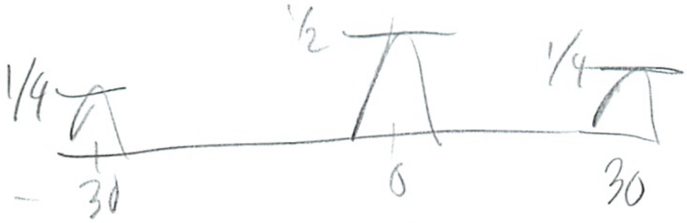
Oh ~~[illegible]~~ the above is spectral coeffs

Just place them



Think before peaking at the answer!

b) Now its back, well kinda

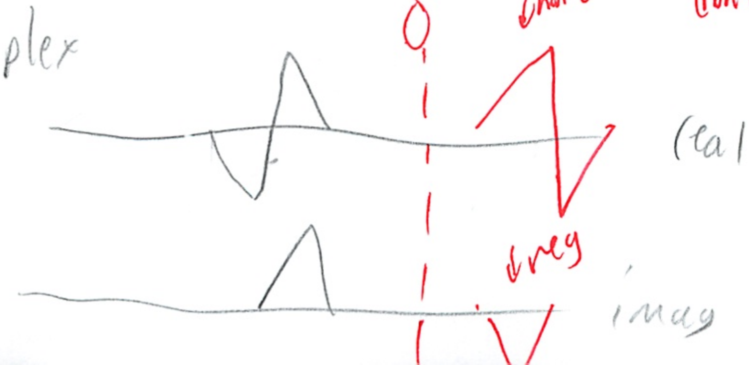


↑

c) gain 2

Cut off ±5

d) Sin is more complex



don't forget other side is on cheat sheet

②

F) Middle is gone
empty

2. SSM
- minimize footprint

(I just don't feel like doing this - can't concentrate

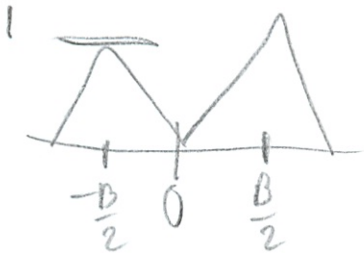
Phase shifted means sin becomes cos?

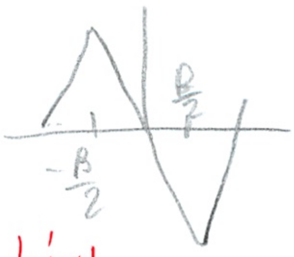
So on top of each other

But out of phase

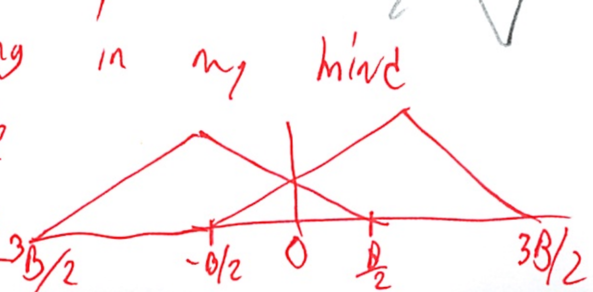
Points A B

they show s[n]





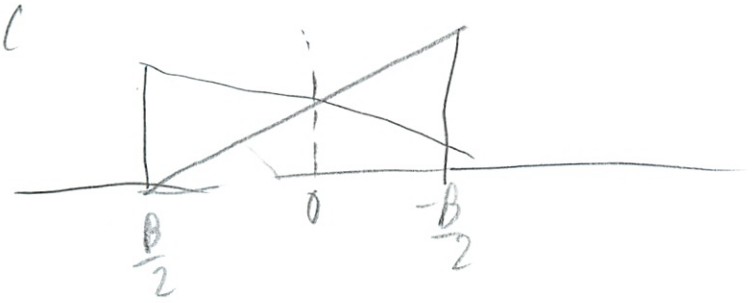
Got pic wrong in my mind
- is B wide
So $\frac{B}{2}$ overlaps



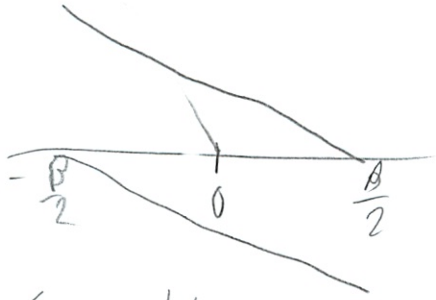
③

b) Now LPF

C

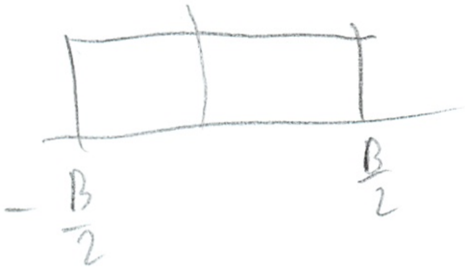


D



So adds to nothing

So adds to

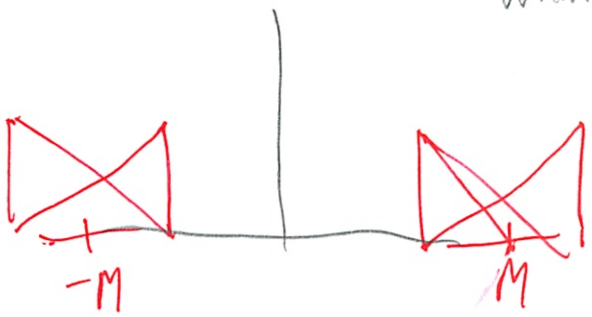


why don't they care what it adds to?

c) Now modulated again

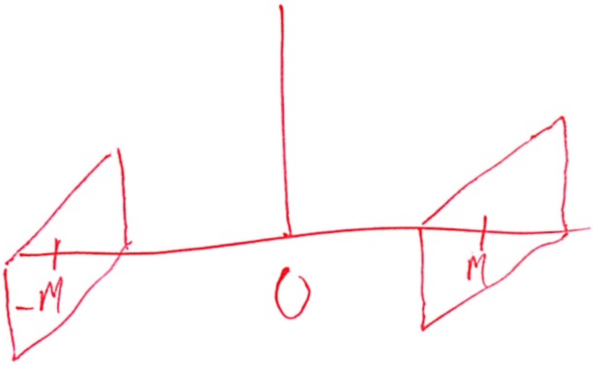
What is M?

keep as variable



? just spread + duplicate what you have

a)



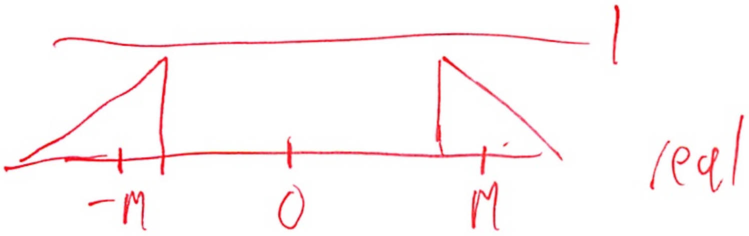
D) Now combine

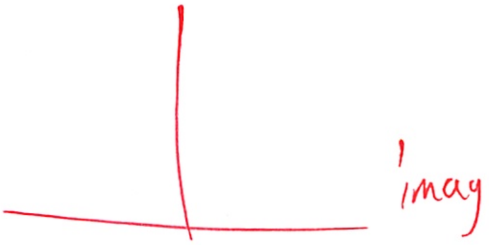
On one real and one img - so just on top of each other

(But how is this scheme supposed to demodulate?

No both are real

get



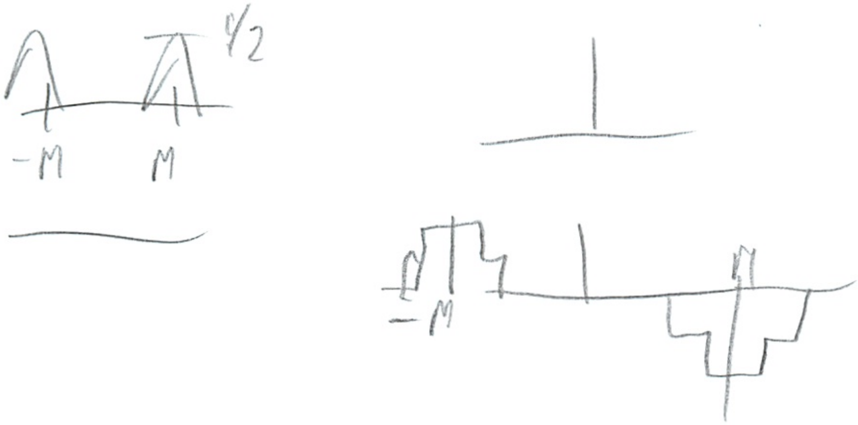


(pay closer attention if for grade

5

(I know the stuff - its a matter of figuring out what they want and doin it right!

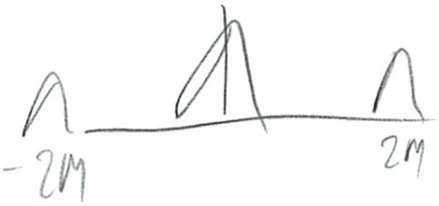
3a.

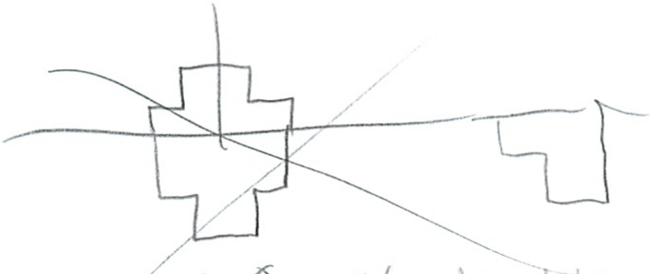


✓

but combine on top of each other to make channel

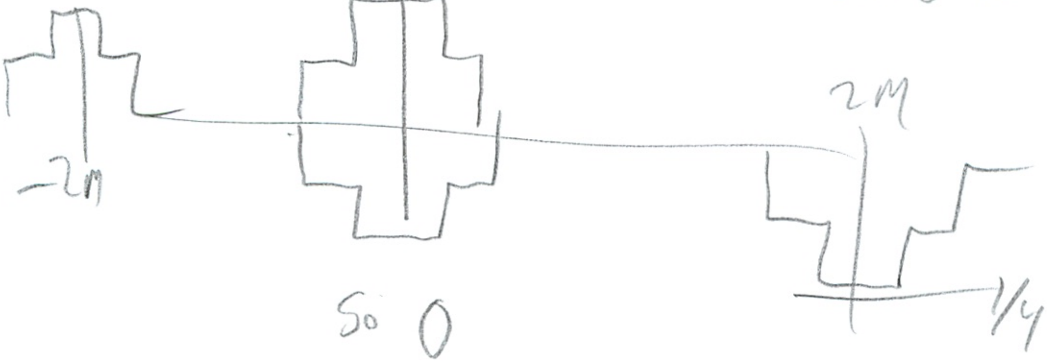
b) Just split again





So 0 ~~We when this splits not like that~~

no it is because its sin

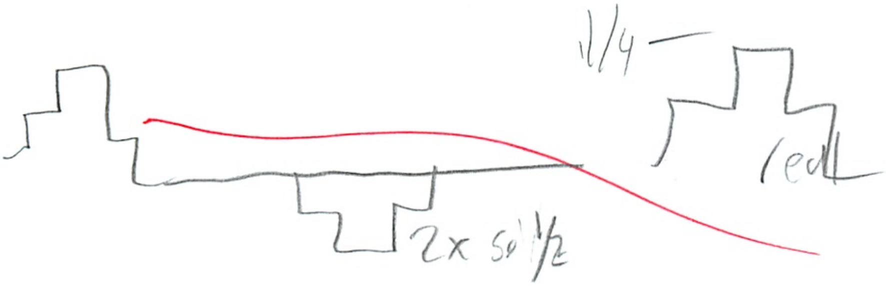


✓

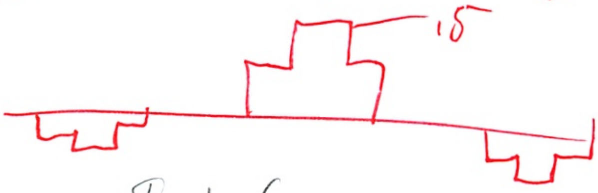
So 0

⑥

Now sin - it flips

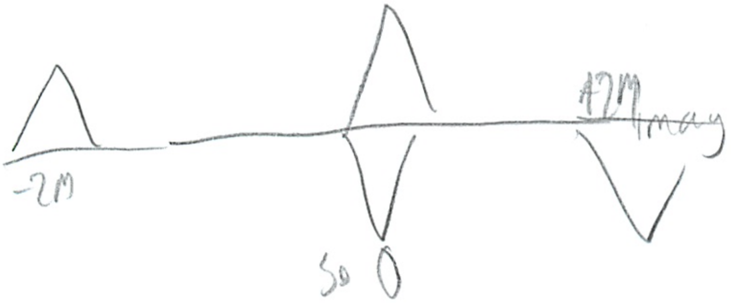


This is actually all flipped



? why ?

real, img are flipped at diff times



✓

The one on the left should have flipped

~~but right on~~

Thats not what is in notes

why did they flip like that???

I don't know unless something w/ j

Someone wrote in online

$\frac{1}{2j}$ when $z = a + jb$ is divided by j what happens?

complex #

$$z = \frac{b+a}{j} = b - ja$$

So b moves real and a moves a and sin flips

Ahhh Still don't really get why only one part flips if have -j's - ? how do you know you have j?

Try some other questions

Switching, Little's Law

1. 4 cust 2 min

$4 = 2D$

~~$D = 2$~~ look closly

Cust arrives 2 min means $\frac{1}{2}$ cust a min

$4 = \frac{1}{2} D$

$D = 8$ min ✓

2. Restaurant 60 people 2 hrs

How manny arrive per hr?

$60 = \lambda 2$

$\lambda = 30$

So 30 people in queu

wait avg 30 min?

they say 1 hr

I think we are assuming diff things

I assume ~~th~~ that 30 people got in line

They assume the people have been there

8

c) 3580 kilb HB

No more than 2 days old on avg

How much help?

(I think the on avg
is nothing extra special
I need to do

3500 ← rate

508 kg/day

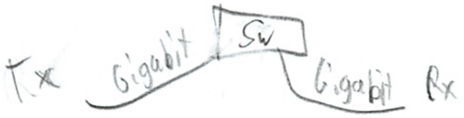
$N = 500 \cdot 2$

$N = 1000$ ✓

(I still got tricked up by these - shameful!)

2. Latency

—— 1GB/sec links 5000 bits packet 10 $\underset{\text{micro}}{^\wedge}$ sec delay



what does gigabit mean?

10^9 bits in a sec

This is what I am confused on - is it
max capacity is 10^9 bits in a sec
and/or does it say anything about the actual speed
of a packet?

- emailed in - should actually research

9

Last looked here May 4

Looking at solution also refers to packet speed

5 micro seconds

I think that makes sense since channel can only do 1 thing at a time

- not like a water ~~pip~~ pipe

Actually Little's Law might help

D = time

λ = the throughput

N = # of bits sending

~~N~~ 5000 bits = 1 GBit · D

$$D = \frac{5{,}000}{10^9} = 5 \cdot 10^{-6}$$

(Answer my own question)

(Studying is much more inteesting when you actually learn Something - not just make stupid mistakes

10 Pack to qu

5 micro sec

¿do I have to do anything about start + end

- I think so

5 μsec for first packet to arrive

No 1 bit would be ~~a~~ pico second or something

So then just (10 + 5) + (10 + 5) so recieve last bit

30 ✓

b) (10+5)·4 = 60 ✓

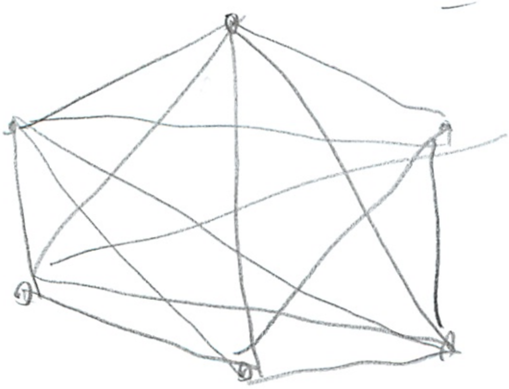
units: microseconds

3. Need some redundancy so no 1 point of failure

- 6 node network where 1 failure does not disconnect

- No need to be conservative

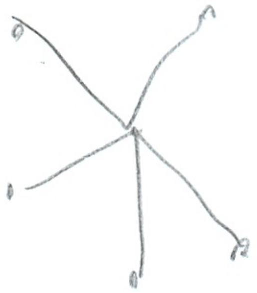
- do fully connected



(11)

b) Some failure - disconnects node

Star



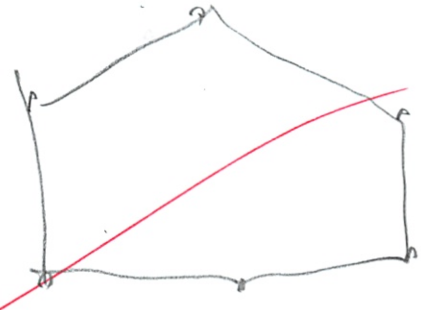
c) Node can't disconnect but link does

it i means what

Says some things not possible

~~1 do 2 line~~

1 line



~~actually~~ link failure - this still works

this is impossible

- what I was thinking
- node and link basically connected

12

4. When is circuit switched better

- Contineas communication
- like voice
- when certain amt bandwith guarentee needed
- cost no object ✓

5. T or F:

a) T - unless you count syn, ack

They say T - but that is diff layer

b) T - all circuits busy ✓

c) T ✓

d) F - still need to know where to switch! ✓

6. TDM circuit switching

	λ		D	N:	
A	.2	A	10	2	✓
B	.2	B	10	2	✓
C	.1	C	5	.5	✓
D	.1	D	5	.5	✓

: seperate right

B) A .4 So N = 4 X Unbounded

: others unchained ✓

Arrival rate > outbounded rate

: pay attention to!

13

Skip Some qu - This is boring

Online responded

- talled about RTT + propagation
- he was talking about usable, good throughput when sending "reliable" packets - that is handing windows + timeouts
- This useful throughput is diff than the raw throughput I taked about

9. 1000 bytes from Cabridge to Berky

50 ms - no queue

125 ms - queue max

75 - avg

1 Mbit/sec send

a) Mean # in queue

25 is lenght = D

$\lambda = \frac{1 \text{ Megabit}}{1000 \text{ bits}}$ ← Since want packets $\frac{10^6}{10^3} = 10^3$ packets/sec

(4)

So $N = \lambda D$

$= 10^3 \cdot 25$ micro sec

$= 10^3 \cdot 25 \cdot 10^{-6}$

$= .025$

↑ partial packets?

Does not seem right

So 1000 packets/sec - Avg queue lenght $25 \cdot 10^{-6}$ sec

So avg # packets in queue could be that...,
if often empty

Ⓧ They used $D = 75$ mc

$N = 1000 \cdot .075 = 75$ packets

also I confused mili and micro secounds

But also they used 75 sec #

I thoght 50 sec is transmission delay - other wise

B) New Tx 2 Megabit /sec

Rx gets 1.6

Packet loss rate

$= \frac{\text{packets lost}}{\text{sent}}$ So 1sec ~~1.6·10~~ $\frac{.4 \cdot 10^6}{2 \cdot 10^6} = \frac{.4}{2} = .2$

400 packets /sec ✓

C) What is avg 1 way delay now

Packets still trying to be Txed I am guessing

— are dropped later

Or dropped when queue full

So 125 ms

No they want math ans

$D = \frac{75}{1600} = 47 \text{ ms}$

~~Yeah~~ Yeah it depends where it is dropped

They could have written qu better

(16)

Do another section i Routing

Could also do P-set 9 again - got so many wrong

Only do ones I screwed up

2a Min RTT

$3 \cdot 10^8$ meters per sec

36000 km

36,000,000 m

$$\frac{\text{meters}}{\text{sec}} \cdot \frac{\text{meters}}{\text{sec}} = m$$

dimensional analysis

$$\frac{\text{meters}}{\frac{m}{s}} = m \cdot \frac{s}{m} = s$$

$$4 \cdot \frac{36{,}000{,}000}{3 \cdot 10^8} = 4 \cdot .12 = .48 ✓$$

M

I just did half one time

(17)

3. Chain Net

DV on power up + prop 100 sec

a) At what time has info for A, D

Why did I put those original answers in?

t=0 A

A
A

t=1 A ← B

A	B
A	B
B	

t=2 A ? B ← C

A	B	C
A	B	C
B	C	

I think I assumed here too - don't know why

t=3 A B C ← D

A	B	C	D
A	B	C	D
B	C	D	

t=100 A → B C D

A	B	C	D
A	A	C	D
B	B	D	
	C		

t=101 A ← B → C D

A	B	C	D
A	A	A	D
B	B	B	
C	C	C	
		D	

18

t=102

A B ← C → D

A	A	A	D
B	B	B	B
C	C	C	C
	D	D	A

✓

A = 103

A B C ← D

A	A	A	A
B	B	B	B
C	C	C	C
	D	D	D

t = 200

A → B C D

no change

t = 201

A ← B C D

A
B
C
D

✓

So simple - why did I screw up before?

(19)

When can't connect sets to ∞

C fails and C detects at T

No TTL

c) will protocal converge?

t=T	A	B	C × D	
	A	A	A	A
	B	B	B	B
	C	C	C	C
	D	D	~~D~~	D

t=T+1	A	B ← C × D		
	A	A	A	
	B	B	B	
	C	C	C	
	D	~~D~~	~~D~~	

?what happens to D?

it also sees that link is down

Should drop A, B, C link

? they should have clarified

t=T+2 A ← B

So yes

Don't forget about advert order

-made this mistake before

B → C

I've got D! But then a count to ∞ loop!

100 defined as ∞

20

D) There exists some time T_2 that A -> D bounces back

This is the one I got wrong - it will happen

E) Bounce back A - B

No - still not is my intuition

Can happen

C updates

C Sends D∞ to B

B ~~does not~~ has D, 101, right

A updates to B D, 4, left

Don't make this mistake again!

F) Can it go A -> B - C - B - A

How do you tell?

Think of scenarios

Its tough b/c its proving a lack of something

Not possible

a

4. Fish Net

Want dark lines to have = load

From S_1 S_2

to A, B, D

Dikstra

$R_A + R_D = R_B$

So weights for w1

Want D traffic to go w1, so smaller

$w_2 + w_4 = 6$

$w1 + w_3 = 7$ 1 smaller

$5 - w_3 = 5 - 2 = 3$ ✓

b) Now w_2 - want 1 bigger than $w_1 + w_3 = 8$

So $9 - w_4 =$ [illegible]

$9 - ~~4~~ 3 = 6$ is ~~largest~~ smallest ✓

largest - ∞? No since B traffic would use

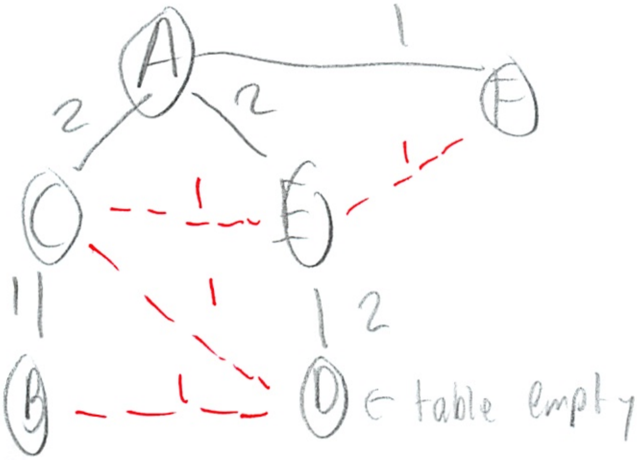
1 smaller than $w_1 + w_3 + w_4$

$5 + 3 + 3 - 1 = 10$ ✓

Why did I get that wrong originally

5. Got right, originally skipping

6. 6 node connected



Add possible lines

Smallest 2 + 2 + 1 = 5?

Or can something cute happen?

Don't think so

I made the same mistakes

How would you have known?

7. Bandwidth from DV ads

n = 200 nodes

m = 500 links

Oh I remember hating this one

(23)

How may linkes per node
It can be configured weirdy
Or will it balance out?
each off the

Each node ~~R~~ $\frac{(4+4+6n)}{T}$ ← assuming fully connected

Total bandwith $\frac{6m(6n+8)}{T}$

~~[illegible]~~ $= \frac{1000 \cdot 1204}{30}$ = 40267 bytes sec

Oh no - it has the full db of other nodes

↑ I keep forgetting this

This is after it is fully established
- usually don't think about there

(24)

Coding Qu After

How long does it take to update?

I totally got this wrong

2 Hellos in both

DV: in next cost 2 · Hello + n · Advert

^ longest path

LS: in one advert - since interval chosen so tables

update each interval

2 · Hello + i · Advert

^ 1 or 2

But still takes time for advert to spread

- but this is ~~D~~ generally smaller than advert intervals

Disconnected

DV Count to ∞ problem

So ~~As~~ ~~Ad~~

Infinity limit · advert slots

LS timing same

^ What does this mean? same as what?

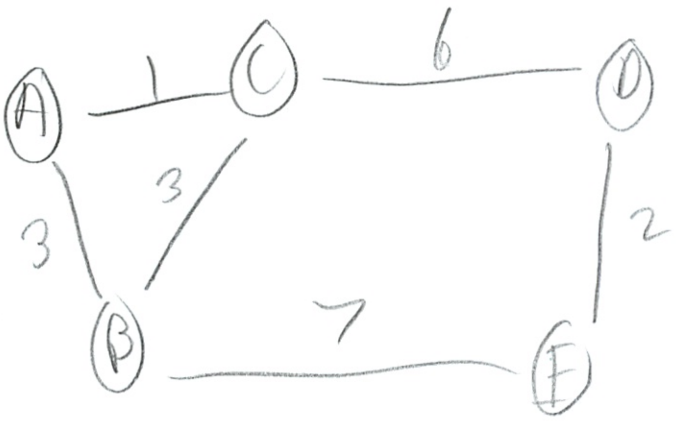
I think above LS, not this DV

25

Should I do some routing tutor problems?
I will try !

7. Dijkstar

(I think the tutors are harder than exams)
+ P-sets



for A — C's table shown

fill in table

C = 1

D = 7 link AC

B = 4 link AC

B = 3

C = 6 - don't count

E = 10 link AB

26

D = 7

E = 9 link AC ← replace

E = 9

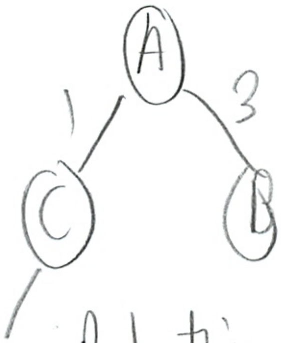
B = 16 don't care

(I finally get why they say $\overset{\text{route}[v]}{=}$ rate[v]

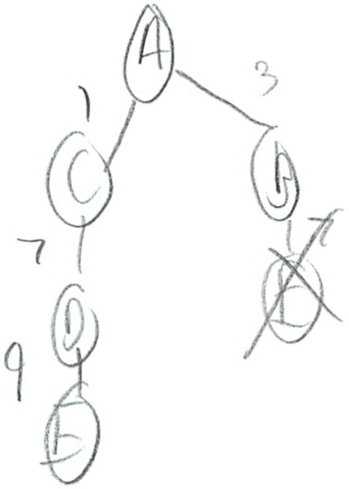
and how it works)

So what format they want ans in?

That table



But this is not how Dijstra operates

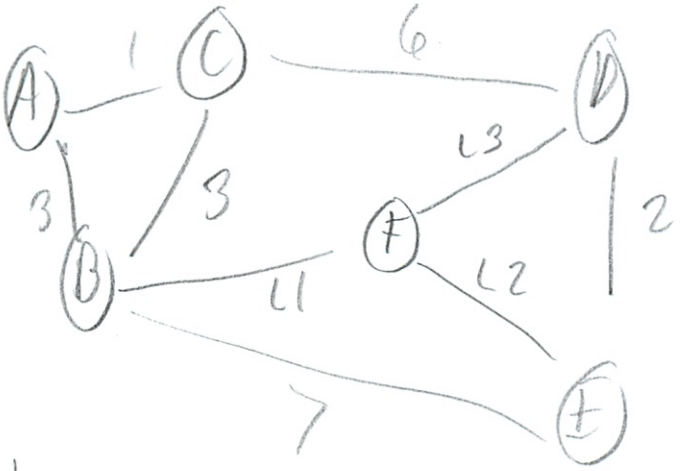


gets replaced later

✓

(27)

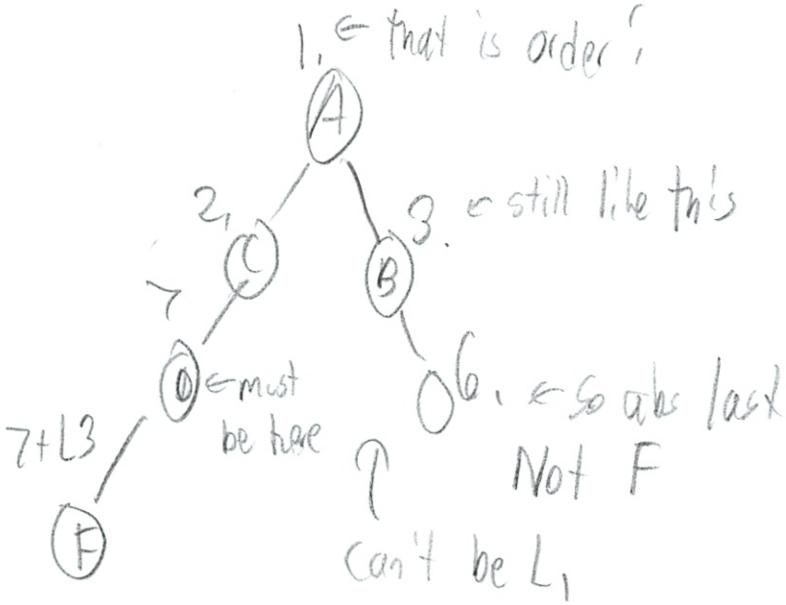
B) Now add F



What are constraints topology shown below

F was not last node added

(This qu is such a pain)



can't be L_1

~~m~~ L_1 must be $> L_3 + 7$

E must be last

$9 < L_3 + L_2$

$9 < L_1 + L_2$

20

$L_1 > L_3 + 7$

$9 < L_3 + L_2$

$9 < L_1 + L_2$

? anything else?

$-7 > L_3 - L_1$

Oh $L_1 > 0$ $L_2 > 0$ $L_3 > 0$

$L_1 > 7$

Oh just asks for constraints

Can do more

L_2 b/w $0, 2$

$0 \underset{\cancel{-}}{<} L_2 \underset{\cancel{-}}{<} 2$

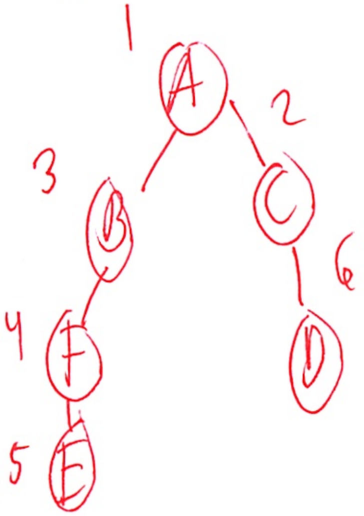
$L_3 < 9$ or > 7

$7 \underset{\cancel{-}}{<} L_3 \underset{\cancel{-}}{<} 9$

? is it or = to No!

– ~~word~~ should think about more

(29)



$L_1 + L_2 < 4$

$L_1 + L_3 > 4$

$L_1 + L_2 > 2$

So $L_1 + L_2 = 3$

$L_1 + L_3 > 4$

This question is just stupid!

Ok that was my qu, now final unit : windows

1. A - B - C - D - E

1 packet per second

No delays

A) RTT

1 (what is delay 1sec?

___ 1 ___ 1 ___ 1 ___

8 sec ✓

3

B) Throughput $= \frac{1}{T} = \frac{1}{8}$ packets/sec ✓

C) Optimal window $W = B \cdot RTT$

↑ 1 packet/sec ↑ 8 sec

↑ 8 packets ✓

Don't forget the other qu : throughput

$$\frac{W}{RTT} = \frac{8}{8} = 1 \text{ pk/sec}$$ ✓

d) W=4 Throughput at E

$$\frac{W}{RTT} \quad \frac{4}{8} = .5 \text{ pts/sec}$$ ✓

$$\text{util} = \frac{\text{Throughput}}{\text{max throughput}} \quad \frac{.5}{1} = 1/2$$ ✓

e) W=8

drop packets w/ odd seq #

timeout = 40 sec

What is # packets in buffer at t=35 sec

(31)

Starts at #1

~~t=1 t~~

~~1 2~~

t=1	~~1~~
t=2	2
	~~3~~
↓	4
	~~5~~
	6
	~~7~~
t=8	8
t=9	—
t=10	~~9~~
t=11	—
t=12	10

RTT = 8

↓ then every other one

— no gets worse (I think I saw this before

t=14	~~11~~
t=16	12
t=18	X
t=20	~~13~~
t=24	14

(32)

$t =$ ~~[illegible]~~ 28 x

$t = 32$ ~~15~~

So sent ~~[illegible]~~ 2 rec 1-14

rec 7 of them ✓

~~[illegible]~~ 2. randomly skip since got previous

3. Sender S
Recieve R

k links p_i in 1 dir
q_i other dir) per link

a) So write expression P, q

~~[illegible]~~

p_i = packet loss rate

So $\frac{1}{5}$ for example

So $2 \; \frac{1}{5} \cdot \frac{1}{5}$

So for whole network

$$\underbrace{\cancel{1 - n \cdot P}}_{\text{want Prob (does not reach)}}{}^{n} \quad \cancel{1 - qn}$$

I also assumed they were all the same

So p_i^2 is prob

- No $2 p_i$

~~It sums - since is ind!~~

↑ I knew this!

But why $1 -$

So p_i is prob it fails

But what we want $P(\text{failure})$

Actually

$$\cancel{1-(1-p_i)}$$

~~$1 - \sum(1-p_i)$~~

↑ $1 - \frac{1}{5} = \frac{4}{3}$ is prob suceeds

Needs to $\frac{4}{5}$.

No it multiplies

$$1 - \prod(1-p_i).$$

Then $1 - \left(\frac{4}{5} \cdot \frac{4}{5} \cdot \frac{4}{5}\right)$ to go back

(34)

So why can't I do that originally?

$\frac{1}{5} \cdot \frac{1}{5} \cdot \frac{1}{5}$

? gets very small - not what we want

B) If $a << 1$

Then p is very small

A) They want a specific ans

$$1-(1-a)^k$$

$$\approx 1-(1-ka) = ka$$

Oh that is what I orignally had !

C) Suppose stop + wait

E[# transmissions before S can send

On my cheat sheet

[illegible] ~~(1-L) RTT + L(timeout + T)~~ ← time

$$X = (1-L) \cdot 1 + L \cdot (X+1)$$

Solve for X

$$X = 1-L + LX + L$$

$$X = 1 + LX$$

$X - LX = 1$

$X(1-L) = 1$

$X = \frac{1}{1-L}$

$\frac{1}{(1-p)(1-q)}$

I think thats the same

---

4. 40 kb/s to moon

1.5 light-sec

? what is light ?

1 kb using stop + wait

What rate

~~1000 = λ~~ 1.5

We know this 40 kb

2·1.5 $\frac{1}{40}$ sec = RTT = .15

? can't do that

thanks — but wasn't that the ans last time

transfer rate $\frac{1 \text{ kbyte}}{3 \text{ sec}} = 333 \frac{\text{bytes}}{\text{sec}} = 2.6$ kbit/sec

Util $2.6/40 = 6.5\%$

I didn't even answer the right qu!

But yes prop delay matters I think

& in addition to throughput

Look at Fabian's response

—which was for this reliable

$$RTT = \frac{\text{data size}}{\text{link speed}} + \frac{\text{ack size}}{\text{link speed}} + \underset{\text{roundtrip}}{\text{prop delay}}$$

$$= \frac{10^6}{10^9} + \frac{100}{10^9} + 1$$

or is this "processing time"?

$$\approx 1.001 \text{ sec}$$

So need to count both!

Say rate = 1 packet/sec

takes 1 sec to transmit

So this problem

40 kb transmitted in $\frac{1}{40}$ + 1.5 prop delay

I'm guessing $4(\frac{1}{40} + 1.5) = RTT$

(37) Now back to regularly scheduled problem

B) Smallest window for max data rate?

4·40 ?, ← intuition

Its $W = B \cdot RTT_{min}$

$\overset{?}{\cancel{\frac{1}{40}}}$ $\overset{?}{=}$ $4(1.5 + \frac{1}{40})$

40
↑rate

$W = 244$

They say 5 packets/sec · 3 sec = 15 packets
↑
where they get this ???

C) w=10

Only 30 buffer

How many bits seq # needed

So if 0 is missing has 1-30

~~40 in the~~ air

~~30-40~~ ← 0 _ 3~~8~~-~~40~~39 in air

Then needs to discard from
40, 41, 42 etc untill it gets 1

? So ∞ ?

Says at most 4 in transit
So 6-bit for $2^6 = 64$

38

Oh read carefully - it stops acking packets except one its looking for

- yeah won't send more acks
- so no more than 40

Need to think clear about

One more problem

Randomly picking 9

9. Sliding window w

ACK k

a) True ✓

b) No - start ✓

c) what is it - that packet - yeah - Oh no must send an ack for - tripped up in my coding

False - but I don't get their explination

d) ~~D~~. No can go out of order ✓

e) No ✓

That was too - short one more

39

13. Ben has no timestamp

- stores tk on local machie

¿But what about Retx ?

a) Yeah ✓

b) Yeah ? what diff than above False

↓I didn't really notice this

Can retx here

c) Oh I realize - can spuriouse re-tx and then hears original back

So yeah ✓

d) Now reusing seq # 1 or 0

Hell no ✓

14. Sliding window $RTT_{min} = .1$ sec

$C = 100$ pkts/sec

$Q = 20$ pkts

A. If $W_i = 8$, what is Throughput

- Cheat sheet

$= \frac{1-L}{RTT}$ ¿but how w/ queues

40

If bandwidth delay = 10 packets
↑ TB · RTT

Oh what window should be, if done right

With w = 8 - no queue built up
80 pts/sec

(This tripped me up before - pay close attention to!)

B) w = 16

So now w ≥ 10

6 will be in queue

Queue delay = $\frac{6}{100}$ = .06 sec

RTT = RTT$_{min}$ + queue delay = .1 + .06 = .16

Throughput = ~~[illegible]~~ $\frac{w}{RTT}$ = $\frac{16}{.16}$ = 100 pts/sec

Make sure this is on cheat sheet

6 will always be in queue!

4

c) Smallest window where RTT > RTT min

11

↑ one more than 10

Fairly obvious in retrospect

I thought next qu will ↑w to past queue size

then throughput = 0 I think???

? Depends where packets are dropped

If keep adjusting W from this RTT would be feedback loop

Throughput is just B!

5/17

Ohhh

sin - when in negitive area?
not flipped
Since 2x negitive?

$j \cdot j = -1$

emailed in

Asked Prof in person
- I think he said this was correct