University of Pennsylvania
Department of Electrical and System Engineering
Digital Audio Basics

ESE150, Spring 2021 — Midterm Solutions — Wednesday, March 3

See exam as given for figures and question details.

1. Frequency of waveform in Hz? 625
   5 cycles in 8ms, or 5/0.008 Hz
2. Sample rate in Hz? 500Hz
   Every 2ms = 500 per second.
3. The waveform is correctly sampled? False
   Signal changes direction between samples.
   Sampling at 500 Hz, we can only accurately capture signals below 250 Hz.
4. Inferred frequency from samples in Hz? $625 \bmod 500 = 125$
5. Frequency of waveform in Hz? 800
   2 cycles in 2.5 ms = 8 cycles per 10ms = 8/0.01=800 Hz
6. Sample rate in Hz? 2000
   Every 0.5ms = 2000 per second.
7. The waveform is correctly sampled? True

   Signal (800 Hz) is less than half the sample rate (2000 Hz).
8. Inferred frequency from samples in Hz? 800
   No aliasing; reconstruct original signal perfectly.
9. If we assign the same number of bits to each of the 13 symbols, how many bits will we need to encode the 27 symbol quote? $\lceil \log_2(13) \rceil \cdot 27 = 108$

10. According to Shannon Entropy, what is the lower bound on the number of bits needed to represent the collection of 27 symbols for the entire quote?

			Bits		
Occur	**What**	**Prob.**	**symbolic**	**per sym**	**ext.**
4	4 characters occur once (k,n,r,. (period)	1/27	$\log_2(1/27)$	4.75	19
10	5 characters occur 2 times (a, c, h, s, t)	2/27	$\log_2(2/27)$	3.75	37.5
9	3 characters occur 3 times (e, m, o)	3/27	$\log_2(3/27)$	3.17	28.53
4	1 character occurs 4 times (space)	4/27	$\log_2(4/27)$	2.75	11
27	13 unique characters				96.03

96

11. Select an encoding for each symbol to minimize the length of the encoded quote.

Since it doesn't matter which encoding of a particular length is used, the exact assignment doesn't matter. Nonetheless, it does matter that we use the shortest encodings for the most frequent symbols and the longest encodings for the most frequent. And, each symbol needs a unique encoding.

symbol	**encoding**	**freedom/requirements**
a	0010	could get length 3 or 4
e	100	must get length 3
k	01000	must get longest, length 5
n	01001	must get longest, length 5
r	01110	must get longest, length 5
t	111	could get length 3 or 4
c	0011	could get length 3 or 4
h	0101	could get length 3 or 4
m	101	must get length 3
o	110	must get length 3
s	0110	could get length 3 or 4
. (period)	01111	must get longest, length 5
(space)	000	must get a shortest code – length 3

12. For your selected encoding, how many bits are required to represent the quote?

symbol	**encoding**	**occurrences**	**bits**
a	0010	2	8
e	100	3	9
k	01000	1	5
n	01001	1	5
r	01110	1	5
t	111	2	6
c	0011	2	8
h	0101	2	8
m	101	3	9
o	110	3	9
s	0110	2	8
. (period)	01111	1	5
(space)	000	4	12
Total			97

97

13. What is the minimum sample rate in Hz for accurately capturing a Tuba solo?

$2 \times 349 = 698$

Also took $2 \times 392 = 784$

14. What is the minimum sample rate in Hz for capturing a piccolo solo?

$2 \times 3951 = 7902$

15. If you only need to represent notes on the scale (corresponding to notes for which there are piano keys), how many bits do you need to represent the frequency of single notes in a Tuba-Piccolo duet?

There are more than 64 and less than 128 notes in that range, so 7 bits will suffice.

Exploiting the gap, maybe can be less than 64 notes, so also accepted 6 bits.

16. How much smaller can a recording of the notes play in the duet be compared to 16b PCM time-sample recording taken at a 44KHz sample rate?

16b PCM samples for 100ms: $44,000 \times 16 \times 0.1 = 70{,}400$b

8b for loudness plus 7b for frequency is 15b per note. Two notes at a time (one from Tuba, one from Piccolo) = 30b.

Note recording can be $\frac{70400}{30} = 2,347$ times smaller than PCM samples.

17. This signal is f(t) $=A \cdot \sin(2\pi \cdot 250 \cdot t) + B \cdot \sin(2\pi \cdot C \cdot t)$

    What is A?

    Take a dot product with the 250 Hz sine samples. Remember to scale by $2/N$.

i	0	1	2	3	4	5	6	7	8	9	**sum**
$f(i \cdot 0.001)$	0	0.88	0.23	-0.73	-0.38	0.5	0.38	-0.26	-0.23	0.12	
$\sin(2\pi \cdot 250 \cdot i \cdot 0.001)$	0	1	0	-1	0	1	0	-1	0	1	
product	0	0.88	0	0.73	0	0.5	0	0.26	0	0.12	2.49

Sum above is before multiply by 2/10.

A=0.498 $\approx$ 0.5

18. This signal is f(t) $=A \cdot \sin(2\pi \cdot 250 \cdot t) + B \cdot \sin(2\pi \cdot C \cdot t)$

    What is C?

    Compute $f(i) - 0.5\sin(2\pi \cdot 250t) = B \cdot \sin(2\pi \cdot C \cdot t)$

i	0	1	2	3	4	5	6	7	8	9
$f(i \cdot 0.001)$	0	0.88	0.23	-0.73	-0.38	0.5	0.38	-0.26	-0.23	0.12
$\sin(2\pi \cdot 250 \cdot i \cdot 0.001)$	0	1	0	-1	0	1	0	-1	0	1
$B \cdot \sin(2\pi \cdot C \cdot i \cdot 0.001)$	0	0.38	0.23	-0.232	-0.38	0.00	0.38	0.24	-0.23	-0.378

This starts at 0,
crosses 0 from above between 2 and 3,
crosses 0 from below around 5,
crosses 0 from above between 7 and 8,
returns to zero at 10

So, we have 2 full cycles in 10 ms, or 5 ms per cycle.

B=200 Hz

19. This signal is f(t) $=A \cdot \sin(2\pi \cdot 250 \cdot t) + B \cdot \sin(2\pi \cdot C \cdot t)$

    What is C?

    Now, we can either take another dot product, or we can solve for any one of the points.

i	0	1	2	3	4	5	6	7	8	9	**sum**
$f(i \cdot 0.001)$	0	0.88	0.23	-0.73	-0.38	0.5	0.38	-0.26	-0.23	0.12	
$\sin(2\pi \cdot 200 \cdot i \cdot 0.001)$	0	0.95	0.59	-0.59	-0.95	0	0.95	0.59	-0.59	-0.95	
product	0	0.84	0.14	0.43	0.36	0	0.36	-0.15	0.14	-0.11	1.99

Sum above is before multiply by 2/10.

B=0.398 $\approx$ 0.4

20. Consider critical band 2 from 100–200 Hz.

    Assume humans can only distinguish frequencies as different in this range if they differ by > 3Hz.

    Assume humans can only notice amplitude difference as small as 1 dB.

    How many bits do you need to represent sounds in this band for humans?

    We need to capture the amplitude of 34 quantized frequencies. Each amplitude can be represented with 7 bits (0 to 120dB at 1 dB quanta). Could also say 8b if wanted to go up above 127 dB.

    $7 \times 34 = 238$

    One could also argue need 7b for frequency and 7b for amplitude, then need to keep a number of 14b (frequency, amplitude) pairs. If you want to be able to capture all 34, that's 14×34 which is larger than the above. But, we would accept that.

21. How does this new group at your left impact your ability to hear the interesting conversation on your right?

    It will tend to mask the interesting conversation, preventing you from hearing it. The new group will likely be talking in the same frequency range as the previous group. Since their sound is louder, your ear will only be able to pickup and interpret the louder signal.

22. Would it help to record the audio at your table to an MP3? Why or why not?

    It would not help. The MP3 encoding is modeling the masking that occurs in the human auditory system. As such, it will be throwing away the masked frequencies that you cannot hear, so it provides no additional information.

23. Would it help to record the audio at your table in PCM? Why or why not?

    It would help. The PCM audio is not discarding any information. The lower amplitude frequencies that may be masked in the MP3 are still recoverable from the PCM as outlined in Q24.

24. Sketch how you could program audio processing (perhaps for your phone) to recover more of the interesting conversation on the right than you were able to hear unaided?

    Idea: remove the loud frequencies so that it is possible to hear the lower amplitude frequencies.

    - start with standard anti-alias filtered (22KHz) and PCM sampled (44KHz) sound capture
    - convert to frequency domain with DFT
    - Remove frequencies with amplitudes above a threshold (maybe 40dB, but can vary the threshold to extract different version to listen to)
    - convert back to time domain with inverse DFT
    - playback time-domain samples to listen

25. Which animals can communicate with each other directly?

    - chicken and human
    - check and little brown bat These cannot. The chicken's range only goes up to 2 KHz while the bat's starts above 10 KHz
    - human and little brown bat

26. Consider designing a zoo-session (zoom optimized for communication among animals).

    How could this zoo session allow any animals (like the chicken, human, and little brown bat) to communicate with each other?

    Assume each animal is at its own, personal zoo-client terminal.

    Sketch the basic processing your zoo-client could perform to allow the animals to communicate.

    - Assume the client knows the animal at the zoo client (at least its audio range).
    - Anti-alias filter at the limit of the animal's audio range.
    - Sample the anti-alias filter signal at twice the animals highest frequency.
    - Convert to frequency domain with DFT.
    - Scale frequencies captured to a standard range (like 0 to 1, where 0 is at the lower limit of frequencies the animal can produce/hear and the 1 is at the upper limit).
    - At the client shift and scale the sound to the receiving animal.
      - scale the 0 to 1 range to the 0 to (animal upper frequency minus animal lowest frequency)
      - shift the scaled range to the animal's lower frequency
      - this makes sure that all frequencies are in the audible range for the receiving animal and spread out across as many different frequency bands as possible.
    - convert scaled and shifted frequencies back to time domain using a DFT.
    - play time-domain samples for receiver.