ECS 452: Digital Communication Systems **2015/2**

HW 1 — Due: Feb 5

Lecturer: Asst. Prof. Dr. Prapun Suksompong

Instructions

(a) Must solve all non-optional problems. (5 pt)

  (i) Write your first name and the last three digit of your student ID on the upper-right corner of ***every*** submitted page.

  (ii) For each part, write your explanation/derivation and answer in the space provided.

(b) ONE part of a question will be graded (5 pt). Of course, you do not know which part will be selected; so you should work on all of them.

(c) Late submission will be rejected.

(d) **Write down all the steps** that you have done to obtain your answers. You may not get full credit even when your answer is correct without showing how you get your answer.

Problem 1 (HW1-2015-2)**.** Consider the code $\{0, 01\}$

(a) Is it nonsingular?

Yes, it is non singular. The two codewords are different.

(b) Is it uniquely decodable?

Yes, it is UD. Given a sequence of codewords, first isolate occurrences of 01 (i.e., find all the ones) and then parse the rest into 0's.

Remark: This code is suffix-free.
No codeword is a suffix of any other codeword.
To see this, use backward decoding. For any received sequence/string, we work backward from the end, and look for the reversed codewords. Since the codewords satisfy the suffix-free condition, the reversed codewords satisfy the prefix-free condition, and therefore we can uniquely decode the reversed sequence/string.

(c) Is it prefix-free?

No, the code is not prefix-free. The first codeword (0) is a prefix of the second codeword (01).

Reminder: prefix-free code is the same as prefix code.

This is more accurate in terms of meaning. **This is the name that has been traditionally used.**

Remark: Observe that we can specify whether a code is nonsingular, uniquely decodable, or prefix-free even when there is no information about the shared alphabet nor the pmf of the DMS symbols

Problem 2 (HW1-2015-2). Consider the random variable X whose support S_X contains seven values:

$$S_X = \{x_1, x_2, \ldots, x_7\}.$$

Their corresponding probabilities are given by

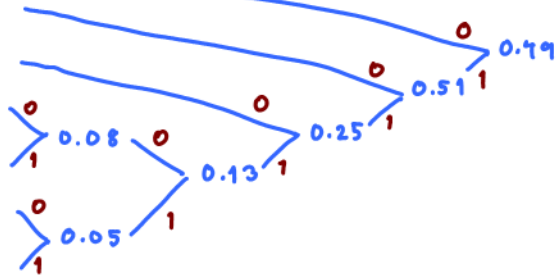
x	x_1	x_2	x_3	x_4	x_5	x_6	x_7
$p_X(x)$	0.49	0.26	0.12	0.04	0.04	0.03	0.02

(a) Find the entropy $H(X)$.

$H(X) = -\mathbb{E}\left[\log_2 p_X(X)\right] = -\sum_x p_X(x) \log_2 p_X(x) = 2.0128$ bits per symbol.

(b) Find a binary Huffman code for X.

x	$p_X(x)$	$c(x)$	$\ell(x)$
x_1	0.49	0	1
x_2	0.26	10	2
x_3	0.12	110	3
x_4	0.04	11100	5
x_5	0.04	11101	5
x_6	0.03	11110	5
x_7	0.02	11111	5



(c) Find the expected codelength for the encoding in part (b).

$$\mathbb{E}[\ell(X)] = 1\times 0.49 + 2\times 0.26 + 3\times 0.12 + 5\times\underbrace{(\underbrace{2\times 0.04}_{0.08} + 0.03 + 0.02)}_{0.13}$$

$= 2.02$ bits per symbol

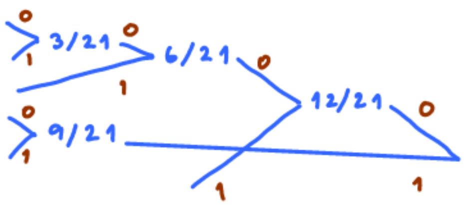
Problem 3 (HW1-2015-2). Find the entropy and the binary Huffman code for the random variable X with pmf

$$p_X(x) = \begin{cases} \frac{x}{21}, & x = 1, 2, \ldots, 6, \\ 0, & \text{otherwise.} \end{cases}$$

Also calculate $\mathbb{E}[\ell(X)]$ when Huffman code is used.

$$H(X) = -\sum_{x} p_X(x) \log_2 p_X(x) \overset{\text{MATLAB}}{=} 2.3983 \text{ bits per symbol}$$

x	$p_X(x)$	$c(x)$	$\ell(x)$
1	1/21	0000	4
2	2/21	0001	4
3	3/21	001	3
4	4/21	10	2
5	5/21	11	2
6	6/21	01	2



$$\mathbb{E}[\ell(X)] = \frac{17}{7} \approx 2.43 \text{ bits per symbol}$$

Problem 4 (HW1-2015-2). These codes cannot be Huffman codes. Why?

(a) $\{00, 01, 10, 110\}$

(b) $\{01, 10\}$

(c) $\{0, 01\}$

(a) The code {00,01,10,110} can be shortened to {00,01,10,11}. The prefix-free property is preserved; so the shortened code is still UD. Regardless of the probabilty associated with each source symbol (as long as the last symbol has non-zero probability), the shortened code will have smaller expected length. Because Huffman code is optimal, the original code can't be Huffman.

(b) The code {01,10) can be shortened to {0,1}. The prefix-free property is preserved; so the shortened code is still UD. Regardless of the probabilty associated with each source symbol, the shortened code will have smaller expected length. Because Huffman code is optimal, the original code can't be Huffman.

(c) The code {0,01} is not prefix-free. Any Huffman code must be prefix-free. Therefore, it cannot be a Huffman code.

Alternatively, one can also use the same reasoning in part (a) and (b): The code {0,01) can be shortened to {0,1}. The new code is prefix-free and hence still UD. Regardless of the probabilty associated with each source symbol, the shortened code will have smaller expected length. Because Huffman code is optimal, the original code can't be Huffman.

Problem 5 (HW1-2015-2)**.** A memoryless source emits two possible message Y(es) and N(o) with probability 0.9 and 0.1, respectively.

(a) Determine the entropy (per source symbol) of this source.

$$H(X) = -\sum_{x} p_X(x)\log_2 p_X(x) = 0.469 \text{ bits per symbol}$$

MATLAB

(b) Find the expected codeword length per symbol of the Huffman binary code for the third-order extensions of this source.

x_1	x_2	x_3	$p_{X_1,X_2,X_3}(x_1,x_2,x_3)$	$\ell(x_1,x_2,x_3)$
Y	Y	Y	0.729	1
Y	Y	N	0.081	3
Y	N	Y	0.081	3
Y	N	N	0.009	5
N	Y	Y	0.081	3
N	Y	N	0.009	5
N	N	Y	0.009	5
N	N	N	0.001	5

0.162 0.018 0.028 0.01 0.109 0.271

$\mathbb{E}[\ell(X_1,X_2,X_3)] = 1.5980$ bits per three source symbols $\Rightarrow L_3 = 0.5327$ bits per source symbol

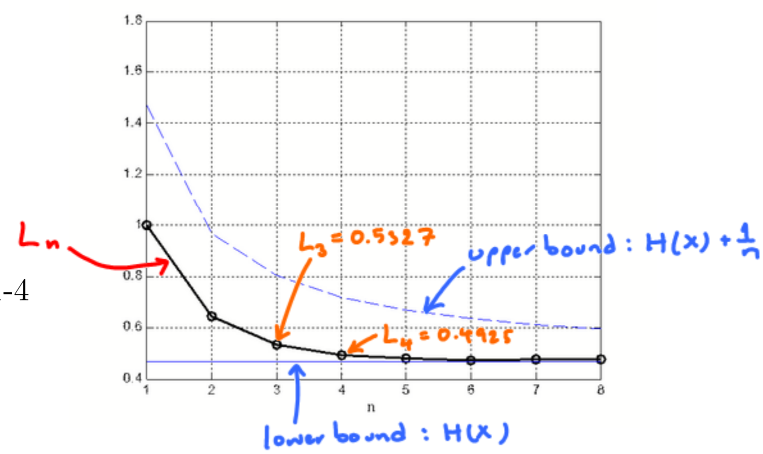
(c) Use `MATLAB` to find the expected codeword length **per (source) symbol** of the Huffman binary code for the fourth-order extensions of this source.

(i) Put your answer here.

$\mathbb{E}[\ell(X_1X_2X_3X_4)] = 1.9702$ bits per 4 source symbols $\Rightarrow L_4 = 0.4925$ bits per source symbol.

(ii) Don't forget to attach the **printout** of your MALTAB script (highlighting the modified parts if you start from the provided class example) and the expressions/results displayed in the command window. See the next page for ides.

(d) Use `MATLAB` to plot the expected codeword length **per (source) symbol** of the Huffman binary code for the nth-order extensions of this source for $n = 1, 2, \ldots, 8$. Attach the **printout** of your plot.



Two ideas for solving part c (and d) of Q5

We first need to list all the probabilities for the (extended) source string (vector/block.)

Idea 1: There are $\binom{n}{k}$ source strings that has k Y and $n-k$ N.

The probability for each of these strings is

$p^k(1-p)^{n-k} = 0.9^k (0.1)^{n-k}$

#Y

k	#	$p^k(1-p)^{n-k}$
0	1	0.0001
1	4	0.0009
2	6	0.0081
3	4	0.0729
4	1	0.6561

we put all these 16 probabilities in one pmf vector.

Idea 2: Use MATLAB's "kron" command

Kronecker product

Defn. $\text{kron}(A,B) = \begin{bmatrix} a_{11}B & a_{12}B & \cdots \\ a_{21}B & a_{22}B & \cdots \\ \vdots & & \ddots \end{bmatrix}$

Observe that the pmf used for the n^{th}-order source extension is the Kronecker product of

the pmf used for the 1^{st}-order extension

and

the pmf used for the $(n-1)^{th}$-order extension

Problem 6 (HW1-2015-2)**.** (Optional) The following claim is sometimes found in the literature:

> "It can be shown that the length $\ell(x)$ of the Huffman code of a symbol x with probability $p_X(x)$ is always less than or equal to $\lceil -\log_2 p_X(x) \rceil$".

Even though it is correct in many cases, this claim is not true in general.

Find an example where the length $\ell(x)$ of the Huffman code of a symbol x is greater than $\lceil -\log_2 p_X(x) \rceil$.

Hint: Consider a pmf that has the following four probability values $\{0.01, 0.30, 0.34, 0.35\}$.

Following the hint, we create a Huffman code from the suggested probability values

$p_X(x)$	$c(x)$	$\ell(x)$		$\lceil -\log_2 p_X(x) \rceil$
0.01	000	3	<	7
0.30	001	3	>	2
0.34	01	2	=	2
0.35	1	1	<	2

This column is added for comparison

Use ceil(-log2(·)) in MATLAB

The claim says $\ell(x) \le \lceil -\log_2 p_X(x) \rceil$ for any x.

Here, we found a case where $\ell(x) > \lceil -\log_2 p_X(x) \rceil$.

Therefore, the claim is not true in general.

Problem 7 (HW1-2015-2)**.** (Optional) Construct a random variable X (by specifying its pmf) whose corresponding Huffman code is $\{0, 10, 11\}$.

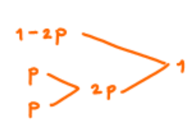
$\{0,10,11\}$ is a Huffman code for

$$p_X(x) = \begin{cases} 1/2, & x = 1 \\ 1/4, & x = 2,3 \\ 0, & \text{otherwise.} \end{cases}$$

$p_X(x)$	$c(x)$
$\frac{1}{2}$	0
$\frac{1}{4}$	10
$\frac{1}{4}$	11

Note that the answer for this question is not unique. You may check that

$$p_X(x) = \begin{cases} 1-2p, & x = 1 \\ p, & x = 2,3 \\ 0, & \text{otherwise} \end{cases}$$



works when $1-2p \ge p \Leftrightarrow p \le \frac{1}{3}$

this condition is here to guarantee that $x=2$ and $x=3$ are grouped first.

Note also that when $p = \frac{1}{3}$, we have uniform pmf.