Random signal analysis I (ECE673)
Solution assignment 3

1. (Problem 4.41) (Problem 4.41) A two-state Markov chain modeling dependent Bernoulli trials has state transition probabilities $P[0|0] = 1/4$, $P[0|1] = 3/4$ and the initial state probability of $P[0] = 1/2$. Draw the Markov state diagram. What is the probability of the sequence (tuple) $(0, 1, 0, 1, 0)$?

   Solution: The Markov state diagram is shown in the figure below.

   The probability of the simple event of interest is

$$
\begin{aligned}
P[(0,1,0,1,0)] &= P[0] \cdot P[1|0] \cdot P[0|1] \cdot P[1|0] \cdot P[0|1] = \\
&= \frac{1}{2}\frac{3}{4}\frac{3}{4}\frac{3}{4}\frac{3}{4} = \frac{81}{512} = 0.1582.
\end{aligned}
$$

2. At a party a large barrel is filled with 99 gag gifts and 1 diamond ring, all enclosed in identical boxes. Each person at the party (say that there are 10 people) is given a chance to pick a box from the barrel, open the box to see if the diamond is inside, and, if not, to close the box and return it to the barrel. (*i*) Define the probabilistic model (sample space, probability assignment). (*ii*) Define a random variables that measures the number of lucky persons that draw the diamond ring. What is its probability mass function? (*iii*) Evaluate the probability that only one person finds the diamond ring. (*iv*) If the number of attendees and the number of gag gifts were very large, how could we approximate the probability mass function of the number of persons choosing the diamond ring?

   Solution: (i) The probabilistic model can be described as an independent Bernoulli sequence, where the probability of "success" (obtaining a diamond ring) is $p = 1/100 = 0.01$. In particular, the sample space can be written as

$$
\mathcal{S} = \{(z_1, ..., z_{10})\colon\ z_i \in \{0, 1\}\},
$$

   where "1" denotes "success" (i.e., diamond ring). The probability of any sequence $(z_1, ..., z_{10})$ can be obtained by exploiting the independence between subexperiments

$$
P[(z_1, ..., z_{10})] = p^{N_1}(1-p)^{N_0},
$$

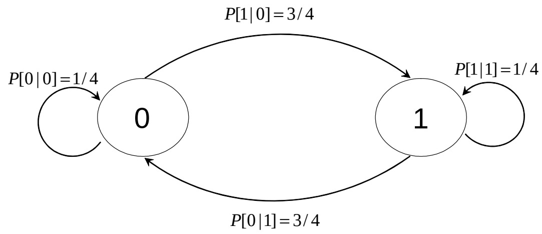
   with $N_1 = \sum_{i=1}^{10} z_i$ couting the number of 1's and $N_0 = 10 - N_1$ the number of 0's.

   (ii) The random variables that measures the number of 1's is

$$
X(s) = \sum_{i=1}^{10} z_i.
$$

   The probability mass function of X is binomial ($X \backsim bin(10, 0.01)$):

$$
p_X[k] = \binom{10}{k} 0.01^k (0.99)^{10-k}.
$$



(iii) Accordingly, the probability of only one person choosing the diamond ring is

$$p_X[1] = \binom{10}{1} 0.01(0.99)^9 = 9.13 \times 10^{-2}.$$

(iv) If the number of persons (N) was large and the probability of success (p) small (due to the large number of gag gifts), the binomial distribution could be approximated by a Poisson probability mass function $Pois(\lambda)$ with $\lambda = Np$.

3. This exercise is meant to illustrate once again the relationship between analysis and computer simulations.

(i) At first, write down the probability mass function (PMF) $p_X[k]$ of $X \sim bin(2, 0.5)$.
(ii) Then, write a MATLAB code that generates the random variable.X.
(iii) Modify your code so as to obtain an estimate of $p_X[k]$ through Monte Carlo iterations. In particualr, estimate $p_X[k]$ by evaluating the number of Monte Carlo iterations that yield any possible value in the range $\mathcal{S}_X$ of X (i.e, evaluate the histogram of X). Compare your result with point (i). Increasing the number of realizations (Monte Carlo simulations) improves the estimate?
Please include your code and numerical outcomes.
Solution: The probability density function of the binomial is

$$\begin{aligned} p_X[k] &= \binom{2}{k}(0.5)^k(0.5)^{2-k} \text{ for } k = 0, 1, 2 \\ &= \begin{cases} 0.25 & k = 0 \\ 0.5 & k = 1 \\ 0.25 & k = 2 \end{cases} \end{aligned}$$

possible MATLAB code to estimate the PMF of X is as follows:

```
N=1000; %number of Monte Carlo iterations
h=zeros(3,1); %contains the relative frequencies of values (0,1,2) for random variable X
for i=1:N %for each Monte Carlo iteration
 u=rand(1);
  if (u<=0.25)
    x=0;
  elseif (u>0.25)&(u<=0.75)
     x=1;
   elseif (u>0.75)
```

```
  x=2;
 end %if
 %updating the estimate of the PMF of X
 if (x==0) h(1)=h(1)+1;
  elseif (x==1) h(2)=h(2)+1;
  elseif (x==2) h(3)=h(3)+1;
 end %if

end
pxest=h/N
```

My MATLAB outcome is as follows:
pxest =
0.250
0.5040
0.2460
The estimates obtained through Monte Carlo iteration are pretty close to the real values.