ASSIGNMENT GUIDE NUMBER 2

March 3, 2019

Bruno Mendes, 81850
André Cunha, 79969
P4G4

Contents

0.1 Exercise number 1

To solve this first part of the guide we need to take advantage of two formulas:

- Bayes' law: consider a set of mutually exclusive events F1, F2, ..., Fn such that its union is the set of all possible outcomes of a random experiment. Knowing that event E has occurred, the probability of event Fj, with j = 1, 2, ..., n, is given by:

$$P(Fj|E) = \frac{P(E|Fj) \cdot P(Fj)}{\sum_{i=1}^{n} P(E|Fi) \cdot P(Fi)}$$

- The probability function of a binomial random value with parameters n and p is:

$$f(i) = \binom{n}{i} \cdot p^i \cdot (1-p)^{n-i}$$

0.1.1 Line a

In order to determine the probability of the link being in the interference or normal state when one control frame is received with errors we need to use both the Bayes' Law and the probability function of a binomial random variable. We also need to bare in mind that:

- P(E) = probability of receiving a control frame with errors
- P(N) = probability of the link being in the normal state
- P(I) = probability of the link being in the interference state

To use the Bayes' Law we start by calculating the probability of receiving a control frame with errors when the link is in the normal state (i.e. P(E|N)). We can easily simplify that by calculating the probability of receiving a control frame with no errors when the link is in the normal state, and then subtract the result to one. Knowing beforehand that the bit error rate when the link is in the normal state is 10^{-7}, and considering a data of 64 bytes (512 bits), and using the probability function of a binomial random variable when i = 0 (i refers to the number of errors):

$$P(E|N) = 1 - \binom{512}{0} \cdot (10^{-7})^0 \cdot (1-10^{-7})^{512-0} = 1 - (1-10^{-7})^{512}$$

Analogously, we need to calculate the probability of receiving a control frame with errors when the link is in the interference state. In this case, the bit error rate is 10^{-3}.

$$P(E|I) = 1 - \binom{512}{0} \cdot (10^{-3})^0 \cdot (1 - 10^{-3})^{512-0} = 1 - (1 - 10^{-3})^{512}$$

The next step is to utilize the Bayes' law using the two values we just calculated. Knowing that the link has a probability p of being in the normal state and 1-p of being in the interference state, to calculate the probability of the link being in the normal state when receiving a control frame with errors:

$$P(N|E) = \frac{P(E|N) \cdot p}{(P(E|N) \cdot p) + (P(E|I) \cdot (1-p))}$$

And to calculate probability of the link being in the interference state when receiving a control frame with errors:

$$P(I|E) = \frac{P(E|I) \cdot (1-p)}{(P(E|I) \cdot (1-p)) + (P(E|N) \cdot p)}$$

Substituting p with the requested values, we get the following results:

	p(normal)	p(interference)
p = 99%	0.0125	0.9875
p = 99.9%	0.1132	0.8868
p = 99.99%	0.5608	0.4392
p = 99.999%	0.9274	0.0726

Concluding, when receiving a control frame with errors, and the higher the probability of the link being in the normal state is, the higher the probability of when receiving that control frame the link is in the normal state. And the lower the probability of the link being in the normal state, the higher the probability that the link is in the interference state when receiving a control frame with errors.

0.1.2 Line b

A false positive is when a station decides wrongly that the link is in interference state, meaning that it could receive n consecutive control frames with error and the link is in the normal state.

To determine the probability of n false positives, we need to determine the probability of receiving n consecutive bits with errors. Mathematically, this translates to, using the Bayes' law, powering to n P(E|N) and P(E|I):

$$P(N|E) = \frac{P(E|N)^n \cdot p}{(P(E|N)^n \cdot p) + (P(E|I)^n \cdot (1-p))}$$

Substituting p and n with the requested values, we get the following results:

Probability of false positives				
	n = 2	n = 3	n = 4	n = 5
p = 99%	1.6150e^{-6}	2.0627e^{-10}	2.6346e^{-14}	3.3649e^{-18}
p = 99.9%	1.6297e^{-5}	2.0815e^{-9}	2.6585e^{-13}	3.3955e^{-17}
p = 99.99%	1.6309e^{-4}	2.0834e^{-8}	2.6609e^{-12}	3.3986e^{-16}
p = 99.999%	0.0016	2.0835e^{-7}	2.6612e^{-11}	3.3989e^{-15}

0.1.3 Line c

A false negative is when a station decides wrongly that the link is in normal state, meaning that at least one of the n consecutive control frames is received without errors and the link is in the interference state.

To determine the probability of n false positives, we need to determine the probability of receiving n consecutive bits without errors. Mathematically, this translates to, using the Bayes' law, subtract P(E|N) and P(E|I) to 1, and then powering it to n:

$$P(I|E) = \frac{(1-P(E|I))^n \cdot (1-p)}{((1-P(E|I))^n \cdot (1-p)) + ((1-P(E|N))^n \cdot p)}$$

Substituting p and n with the requested values, we get the following results:

Probability of false negatives				
	n = 2	n = 3	n = 4	n = 5
p = 99%	8.4066e^{-3}	9.3619e^{-3}	9.7443e^{-3}	9.8975e^{-3}
p = 99.9%	8.3945e^{-4}	9.3565e^{-4}	9.7420e^{-4}	9.8966e^{-4}
p = 99.99%	8.3933e^{-5}	9.3559e^{-5}	9.7418e^{-5}	9.8965e^{-5}
p = 99.999%	8.3931e^{-6}	9.3559e^{-6}	9.7418e^{-6}	9.8965e^{-6}

0.1.4 Line d

Concluding, when there is a higher probability of the link being in the normal state, the probability of occurring false positives rise and false negatives diminish. Also, the number of control frames considered affects the probability of false positives and negatives: the higher the number of control frames, the lower the probability of false positives and the higher the probability of false negatives.

Furthermore, as the probability of the link being in the normal state rises, and since the probability of receiving a control frame with errors is the same (i.e. P(E|N)), the probability of the link being in the normal state and receiving a control frame with errors (i.e. $P(E \cap N)$) rises as well (since P(N) rises). Because of this, there will be more false positives and less false negatives, since there is a higher probability of receiving control frames with errors when the link is in the normal state (i.e. $P(E \cap N)$).

On the other hand, as the number of control frames rises, the probability of one of those control frames having no errors also rises, even with the link being in the interference state, resulting in a higher probability of false negatives and lower false positives.

0.2 Exercise number 2

To solve this second part of the guide we need to take advantage of two formulas:

- Birth-dead Markov chain: if λ_i is the birth rate of state i and μ_i is the dead rate of state i, then the steady-state probability of state 0 is:

$$\pi_0 = \frac{1}{1 + \sum_{i=1}^{\infty} \frac{\lambda_0\lambda_1\cdots\lambda_{i-1}}{\mu_1\mu_2\cdots\mu_n}}$$

- And the steady-state probability of state n > 0 is:

$$\pi_n = \frac{\lambda_0\lambda_1\cdots\lambda_{i-1}}{\mu_1\mu_2\cdots\mu_n} \cdot \pi_0$$

0.2.1 Line a

In order to determine the average percentage of time the link is on each of the five possible states, we need to use the Birth-dead Markov chain. With this system we need to know that we have n states, and to arrive to a new state there is a rate λ_i, and to leave one state there is a rate of μ_n. So we can easily calculate the average time percentage for the state 0, being this the sum between all the rates of arriving divided by the sum of all the rates of leaving. We need to bare in mind that the states are in order beginning with state 0 being 10^{-6} and so on.

$$\pi_0 = \frac{1}{1 + \sum_{i=1}^{\infty} \frac{\lambda_0\lambda_1\cdots\lambda_{i-1}}{\mu_1\mu_2\cdots\mu_n}} = \frac{1}{1 + \frac{1*5*10*10}{5*20*40*195}} = 0.9994$$

$$\pi_1 = \frac{\lambda_0}{\mu_1} \cdot \pi_0 = \frac{1}{195} * \pi_0 = 0.0051$$

$$\pi_2 = \frac{\lambda_0\lambda_1}{\mu_1\mu_2} \cdot \pi_0 = \frac{1*5}{195*40} * \pi_0 = 6.4061e^{-4}$$

$$\pi_3 = \frac{\lambda_0\lambda_1\lambda_2}{\mu_1\mu_2\mu_3} \cdot \pi_0 = \frac{1*5*10}{195*40*20} * \pi_0 = 3.2031e^{-4}$$

$$\pi_4 = \frac{\lambda_0\lambda_1\lambda_2\lambda_3}{\mu_1\mu_2\mu_3\mu_4} \cdot \pi_0 = \frac{1*5*10*10}{195*40*20*5} * \pi_0 = 6.4061e^{-4}$$

0.2.2 Line b

In order to calculate the average time duration in minutes we just need to do 1 divide by the sum of the rates of leaving that determinate state, and then multiply by 60 to get the result in minutes.

$$T_0^{Avg} = \frac{1}{1} * 60 = 60min$$

$$T_1^{Avg} = \frac{1}{5+195} * 60 = 0.3min$$
$$T_2^{Avg} = \frac{1}{10+40} * 60 = 1.2min$$
$$T_3^{Avg} = \frac{1}{10+20} * 60 = 2min$$
$$T_4^{Avg} = \frac{1}{5} * 60 = 12min$$

0.2.3 Line c

The probability of the link being in the interference state, knowing that the link is in that state when the bit error rate is equal to 10^{-3} or higher, we just need to sum the average percentage of time that the link is in that state.

$$\pi_{int} = \pi_3 + \pi_4 = 9.6092e^{-4}$$

0.2.4 Line d

To calculate the average bit error, we need to multiply the bit error of one state by the average percentage of time being in that state of interference, dividing all of that by the probability of being in one of the interference states.

$$AvgBr = \frac{10^{-3} * \pi_3 + 10^{-2} * \pi_4}{\pi_{int}} = 7e^{-3}$$

0.2.5 Line e

When the system is in the state 3, it can either "bounce" between state 2 and 4. Knowing that the state 4 is in the interference state and state 2 is the normal state, theoretically the system can be in the interference state forever, since there is a probability, a really low one, of the system bouncing between state 3 and 4 forever.

In order to calculate the average time that the system is in the interference state we have to calculate the average of leaving the interference state (i.e. state 3 to state 2), and the average of staying in the interference state (i.e. state 3 to state 4 and vice-versa).

$$\pi_{32} = \frac{20}{10+20} = \frac{2}{3}$$

$$\pi_{34} = \frac{10}{20+10} = \frac{1}{3}$$

Finally, to calculate the average time the link is in the interference state, we can apply the following formula, with i being the number of times the system bounces from states 3 and 4, and t_3 and t_4 are the time in minutes that the system is in each state.

$$avgt = \sum_{i=0}^{\infty} \pi_{34}^{i} \cdot \pi_{32} \cdot (i \cdot (t_3 + t_4) + t_3)$$

To calculate the final result we need to stop the sum giving it a real limit, and for our result we used the value 15.

$$avgt = \sum_{i=0}^{15} \pi_{34}^{i} \cdot \pi_{32} \cdot (i \cdot (t_3 + t_4) + t_3) = 9min$$

0.3 Matlab code used

0.3.1 Exercise 1

```
% 1)
p = [0.99,0.999,0.9999,0.99999];
bErrorRateNormal = 10^-7;
bErrorRateInterference = 10^-3;
data = 64 * 8 ;

%P(e|n) = probabilidade de haver erro sabendo que est  no estado normal
pen = 1-((1-bErrorRateNormal)^data);


%P(e|i) = probabilidade de haver erro sabendo que esta no estado de ...
    interderencia
pei = 1-((1-bErrorRateInterference)^data);

%% a)

% P(n|e) = probabilidade de esta no estado normal sabendo que h  erro
pne = pen * p./((pen * p) + (pei* (1-p)))

% P(i|e) = probabilidade de esta no estado interferencia sabendo que ...
    h  erro
pie = pei * (1-p)./((pei * (1-p)) + (pen* (p)))


%% b)
  % pne2 = probabilidade de estar no estado normal e haver n erros ...
    seguidos
n = [2,3,4,5];
p = [0.99,0.999,0.9999,0.99999];
for ni=1:numel(n)
    pne2 = pen ^ n(ni) * p./((pen ^ n(ni) * p) + (pei ^ n(ni)* (1-p)));
    pne2(1);
    pne2(2);
    pne2(3);
    pne2(4);

end
```

```
%% c)
 % pie2 = probabilidade de estar no estado interferencia e haver n ...
     erros seguidos
n = [2,3,4,5];
p = [0.99,0.999,0.9999,0.99999];
for ni=1:numel(n)
    pie2 =  (1 - pei^n(ni)) * (1-p)./(((1-pei^n(ni)) * (1-p)) + ...
        ((1-pen^n(ni))* p));
    pie2(1)
    pie2(2)
    pie2(3)
    pie2(4)

end
```

0.3.2 Exercise 2

```
% 2)

%% a)

pi0 = 1/(1+((1*5*10*10)/(5*20*40*195)))
pi1 = (1/195) * pi0
pi2 = (1*5)/(195*40) * pi0
pi3 = (1*5*10)/(195*40*20) * pi0
pi4 = (1*5*10*10)/(195*40*20*5) * pi0


%% b)

t0 = 1/1 * 60
t1 = 1/(5+195) * 60
t2 = 1/(10+40) * 60
t3 = 1/(10+20) * 60
t4 = 1/5 * 60



```

```
%% c)

p_int = pi3+pi4

%% d)
 avgbr = (10^-3 * pi3 + 10^-2 *pi4)/ p_int



%% e)


t_interferencia = t3 + t4

p32 = 20/(10+20)
p34 = 10/(20+10)

avg_t = 0;
for i=0:20
    avg_t = avg_t + p34^i * p32 * (i * (t3 + t4) + t3)
end
```