**Stochastic processes 364-2-5431**

**Winter 2024**

**Final assignment**

Uriel Zaed 319435608

Q1

(a)

To write the transition probabilities matrix for this Markov chain, we need to consider the possible transitions from each state to other states. The states in this case are the nine points in the plane: (0, 0), (0, 1), (0, 2), (1, 0), (1, 1), (1, 2), (2, 0), (2, 1), (2, 2).

transition probabilities matrix:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | (0,0) | (0,1) | (0,2) | (1,0) | (1,1) | (1,2) | (2,0) | (2,1) | (2,2) |
| (0,0) | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| (0,1) | (1-p) \*0.5 | 0 | (1-p) \*0.5 | 0 | p | 0 | 0 | 0 | 0 |
| (0,2) | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| (1,0) | p\*0.5 | 0 | 0 | 0 | 1-p | 0 | p\*0.5 | 0 | 0 |
| (1,1) | 0 | p\*0.5 | 0 | (1-p) \*0.5 | 0 | (1-p) \*0.5 | 0 | p\*0.5 | 0 |
| (1,2) | 0 | 0 | p\*0.5 | 0 | 1-p | 0 | 0 | 0 | p\*0.5 |
| (2,0) | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| (2,1) | 0 | 0 | 0 | 0 | p | 0 | (1-p) \*0.5 | 0 | (1-p) \*0.5 |
| (2,2) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

(b)

We want to calculate the probability of returning to the initial point (1, 1) after two steps exactly.

Starting at point (1, 1), after one step, there are four possible outcomes:

1. Move horizontally down to (2, 1) with probability .
2. Move horizontally up to (0, 1) with probability .
3. Move vertically right to (1, 2) with probability .
4. Move vertically left to (1, 0) with probability

From each of these new positions, after the second step, we can analyze the probabilities of returning to the initial point.

1. If we moved horizontally down to (2, 1):

* To return to the initial point (1, 1), we must move up.
* The probability of moving up is .

1. If we moved horizontally up to (0, 1):

* To return to the initial point (1, 1), we must move down.
* The probability of moving down is .

1. If we moved vertically right to (1, 2):

* To return to the initial point (1, 1), we must go left.
* The probability of moving left is .

1. If we moved vertically left to (1, 0):

* To return to the initial point (1, 1), we must go right.
* The probability of moving right is .

Now, let's calculate the probabilities of these paths:

Probability of going from (1, 1) to (2, 1) and back to (1, 1):

Probability of going from (1, 1) to (0, 1) and back to (1, 1):

Probability of going from (1, 1) to (1, 2) and back to (1, 1):

Probability of going from (1, 1) to (1, 0) and back to (1, 1):

Now, add these probabilities together to get the total probability of returning to (1, 1) after two steps:

(c)

Y = the length of the game

We can write Y = 2X (because the length of the game is even)

The distribution of X:

This is since we need to end the game in 2 steps, and we have 8 distinct ways to arrive at any of the corners, with each path involving one specific horizontal and one specific vertical step. We count these paths as success, and we will try until we succeed. Meaning every 2 moves we got to one of the edges, or we came back to the initial state and the probability of succeeding remains due to the lack of history.

generating function of Y:

(By Geometric mean)

(d)

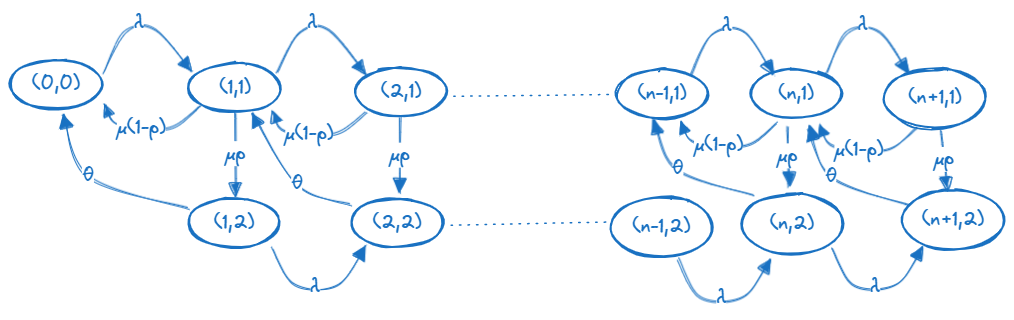
I did a simulation using python (you can see it in the code attached under "Quetion1\_Simulation.py"), and for and 100000 different runs the average number of steps was .

Now let calculate the when :

As we can see, the average steps in the simulation and the calculation of E(Y) are very close (0.0019 difference) as expected, and if we make more runs in the simulation, we will get even closer results.

Q2

(a)



Steady-state equations:

(b)

The stability condition is: .

Let's calculate :

So, the steady state condition is:

(C)

We start by multiplying (2) by , and multiplying (3) by and sum the equations.

Left side:

Right side, first component:

Right side, second component:

Right side, third component:

So' we get:

Now we will do the same for (4) and (5):

Left side:

Right side:

So' we get:

Now we got 2 equations of g1 and g2:

And by using (1) and (4) (And we know that ) we get:

Let's solve the equations of and , and by using the Wolfram Mathematica code attached called ("Quetion\_2\_Calc\_g1\_g2") and also attached in the appendix, we got:

(d)

We need to show that:

As we learned in class, the formula of (the generation function of the number of customers in the system):

First let's calculate the LST of Y:

Now let's add these two together:

From the previous section we know that:

Now we compare the values of and , and by using the Wolfram Mathematica code attached called ("Quetion\_2\_Calc\_g") and also attached in the appendix, we got that the equation is true.

Q3

(a)

Understanding the Events:

The System is Full: This is our starting point. The system has n customers. The Next Event: After the system is full, the next event could be:

* An arrival (which is governed by the Poisson process with rate )
* A departure (which is governed by the exponential service time with rate )

Breakdown of where :

Z is the time until next event:We have 2 possible events: arrival, departure. Z follows an exponential distribution with rate , because we need this is the minimal between the 2 (next event).

is a Binary Variable that equals 0 if the next event it's arrival, because in this case the time where the system is back to full is just Z with no addition. If the next event is departure will equal 1, In this case we now have n-1 customers, so we need to add the (The time until the system returns to n-1 customers) and then add the again.

(b)

Let's compute :

(c)

Let's compute the LST of :

This is because if we have only 1 customer in the system, he is in the server. So, if the next event is departure we need to wait until someone arrives, to be back with 1 customer in the system. And if the next event is arrival, we are back in at least 1 customer in the system.

(d)

Let's calculate the LST recursion of :

Q4

(a)

To find James's average cost per time unit, we need to consider the expected cost per time unit over the lifetime of the car, given his strategy.

Let's break down the problem:

1. Cost of a new car: b
2. Time until the car breaks down:
3. Selling price when the car breaks down at time :
4. Buying a new car when the old one breaks down.

Let's define as the probability for lifetime of a car.

First, we need to calculate the expected period length, by calculation the expectancy of the . Because one period ends or in time meaning that the car broke down or ends in time meaning James sold the car.

The reason for the last equality is:

The cost of the car per period is , and the expected period time is . So:

(b)

Now we assume that , and we want to choose that will minimize the average cost per time unit.

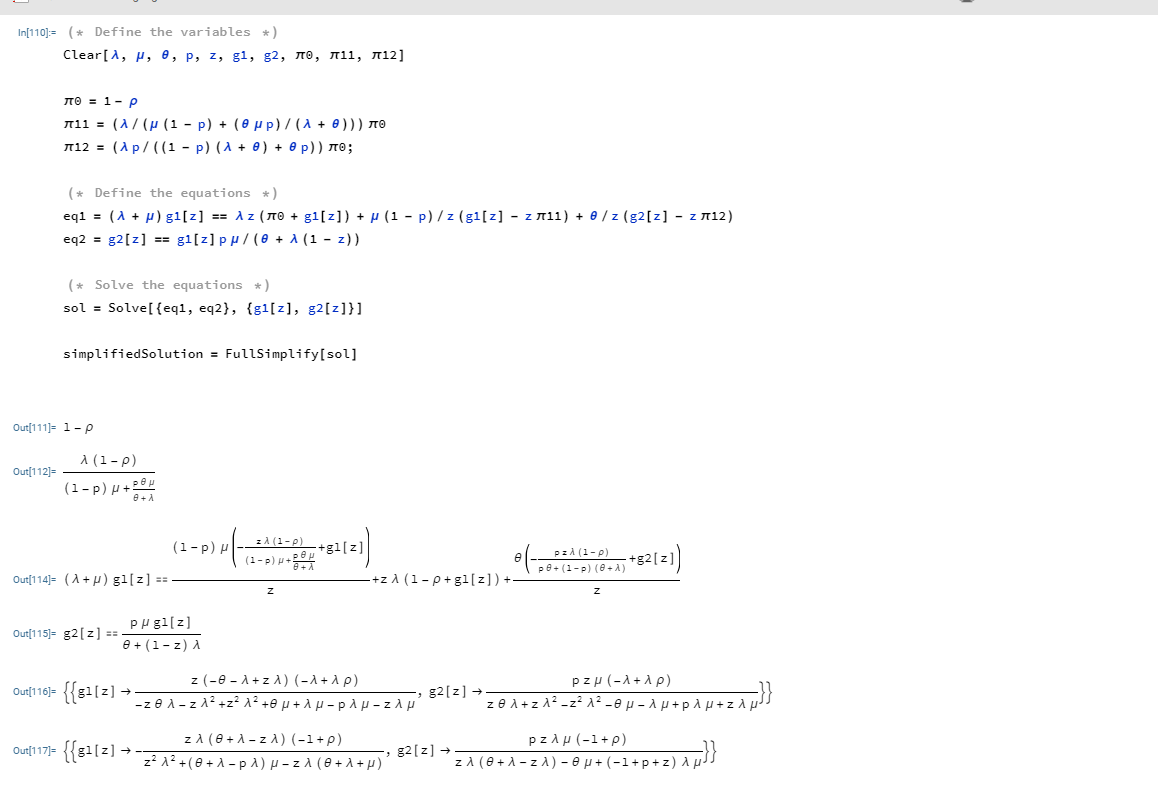
To calculate the of , we could look at it like the probability of Poisson distribution with parameter 1 to be less than 2 in time t. Because Gamma is a sum of exponential distribution with parameter 1\*t, so it's the same to ask what the probability of getting less than 2 events until time t (in Poisson).

So,

Now let's rewrite the average cost per time unit under gamma distribution:

As we can see b is a constant and doesn’t play a role in minimization, so we will ignore it.

After using the code attached ("Qetion\_4\_Calculation.py"), using scipy library (scipy.optimize and scipy.integrate) we find that that time units is the optimal time for James to sell the car when .

Appendix:

