Treball final de grau

GRAU DE MATEMÀTIQUES

Facultat de Matemàtiques i Informàtica
Universitat de Barcelona

QUEUEING THEORY

Autor: Gerard Morro

Director: Carles Rovira
Realitzat a: Departament de Matemàtiques i Informàtica

Barcelona, June 26, 2018

Abstract

This project is aimed to study queueing theory and it is divided in three parts. In the first part we are going to take a theoretical approach to the stochastic processes, specially to the birth and death processes which are indispensable to understand queueing theory. In the second part, we are going to use that knowledge to analyse different types of queues analytically to find some of its characteristic values. Finally in the last part, we are going to do some simulations with the queues we have worked in the previous part to verify if the values we guessed previously were right.

2010 Mathematics Subject Classification. 60K25, 68M20, 90B22

Greetings

I would like to thank Carles Rovira for giving me the idea of this project and for his guidance and feedback through all this semester. I would also thank my parents who helped me during all the steps of my bachelor years. Finally, I greet my mathematician flatmate and friend Gerard Martinez who put up with me and my waves of anger.

Contents

Chapter 1

Introduction

Queues are not a strange thing to humans, everyone of us have got stuck in a traffic jam or waited in the supermarket to be attended. We are really used to wait in queues, but the study of this queues from a Mathematical point of view is not that usual and it is a branch of Mathematics still growing.

Queueing theory has its origins in research done by Agner Krarup Erlang who published a paper about it in 1909. Erlang was a engineer who worked for the Copenhagen Telephone Exchange when he was asked about how many circuits were needed to provide an acceptable telephone service. In modern terms, we could say that he was asked how may servers did the company need to handle all the calls made in Copenhagen.

Erlang proved in his paper in 1909 that Poisson distribution can be applied to the traffic intensity of calls. Poisson distribution is the most usual distribution for queues and we will work on it later.

It was a bit later, in 1940s when queueing theory became an area of interest to mathematicians. Later in 1957, Kendall introduced the modern notation for queues that we are still using today. Nowadays approaches using computers to make large simulations allows us to guess parameters for some queues without solving them analytically although that is not a demonstration it is useful to improve queues performances.

The research in this field have seen applications in traffic jams and supermarkets as it is obvious, but also in computing and telecommunications as web servers also queue their requests.

Nowadays, queueing theory can be seen as a branch of Birth-death Processes which is in turn a branch of stochastic processes. In order to work with queues we will need some basis on Stochastic Processes specifically concerning Markov Chains, as we are going to study queues that its distributions are memoryless.

Chapter 2

Stochastic Process

Queueing Processes are a particular case among Birth-death processes which are in turn a type of Markov Process. Markov processes are a type of stochastic process which satisfies the Markov property.
First of all, we are making a formal definition of a stochastic process:

Definition 1 (Stochastic Process)**.** *Suppose that* $(\Omega, \mathcal{F}, \mathbb{P})$ *is a probability space. Suppose that for each* $\alpha \in I$*, there is a random variable* $X_\alpha(\omega) : \Omega \to \mathbb{R}$ *defined on* $(\Omega, \mathcal{F}, \mathbb{P})$*. The function* $X : I \times \Omega \to \mathbb{R}$ *defined by* $X(\alpha, \omega) = X_\alpha(\omega)$ *is called a stochastic process with indexing set I, and is written* $X = \{X_\alpha, \alpha \in I\}$

This formal definition may hide the purpose of a stochastic process. Stochastic processes can be seen as the collection of random variables $\{X(t), t \in T\}$ where $X(t)$ is a random variable and t being a parameter, generally time. Depending on whether T is countable or not, the processes may be divided in discrete-time or continuous-time.
The set of all possible values of $X(t)$ is called the state space of the process and it can be either countable (discrete state space) or uncountable (continuous state space). A discrete state process is often referred to as a chain. This way we can have continuous or discrete time chains depending on whether T is countable or is not.

2.1 Markov's Chains

Now it's time to focus in a particular process: **Markov Chains**. As it is said before a chain is a stochastic process with a discrete state space. A **Markov Chain** is a chain that satisfies Markov property, this is: The future state of the system only depends on its present state. This property may be expressed as the system being memoryless. Formally:

$$P(X_n = x_n \mid X_{n-1} = x_{n-1}, \ldots, X_0 = x_0) = P(X_n = x_n \mid X_{n-1} = x_{n-1}).$$

The conditional probability:

$$P(X_n = j \mid X_{n-1} = i) \ ,$$

is called the transition probability from state i to state j, denoted by:

$$p_{ij}(n) = P(X_n = j \mid X_{n-1} = i).$$

The Markov chain will be called temporally homogeneous if the probability of transition from any i to any j is always the same regardless in which step the process is in:

$$P(X_n = j \mid X_{n-1} = i) = P(X_{n+m} = j \mid X_{n+m-1} = i) \qquad \forall m$$

when this happen, the transition probability is simply denoted by p_{ij}.
It may be useful also to define the n-step transition probability. It is the probability to pass from state i to j in n steps.

$$p_{ij}^{(n)} = P(X_{r+n} = j \mid X_r = i).$$

Now, it is the moment to define the Transition Probability Matrix (TPM). TPM is a non negative matrix build with every p_{ij}. It is obvious that every row sum is 1 because it is a sum of probabilities for each state of the system.
Also, as $(p_{ij}^{(n)}) = P^n$, it is possible to iterate the probability matrix in a homogeneous system to get the probabilities of a n-step transition.

2.2 Classification of states

Let X_n be a finite homogeneous Markov chain with TPM $P = (p_{ij})$. Then, i leads to state j if there exist $m \in \mathbb{Z}$ such that $p_{ij}^{(m)} > 0$. It is denoted by $i \to j$. It is also said that j is accessible from i. If m does not exist then states i does not lead to j ($i \nrightarrow j$). If two states are both leading each other then it is said that they communicate each other ($i \leftrightarrow j$). All this relations are transitive.
If $i \to j$ but $j \nrightarrow i$ then i is called inessential. Otherwise, i is essential. This way is possible to relate all essential states into a essential class. Essential classes are closed, meaning that if i, j, k belong to an essential class and l is outside it then $i \leftrightarrow j \leftrightarrow k$ but $i \nrightarrow l$, $j \nrightarrow l$ and $k \nrightarrow l$. A finite homogeneous Markov chain has at least one essential class of states. A Markov chain is said to be irreducible if it contains exactly one essential class, so every state communicates with every other state.

Suppose that $i \to i$, so it exists some m such that $p_{ii}^{(m)} > 0$. The greatest common divisor of all such m for which $p_{ii}^{(m)} > 0$ is called the period $d(i)$. If $d = 1$ then state i is aperiodic, and if $d > 1$ it is periodic with period d. Obviously if $p_{ii} > 0$ the state is aperiodic.
For an irreducible Markov chain, all the states may be periodic with the same period or all the states are aperiodic. In the first case it is said the chain is imprimitive and it is called primitive if it is aperiodic.
A Markov chain where all the essential states form a single essential class and they are primitive is said to be regular.

This notation works perfectly with finite chains but in case that our chain is not finite, the classification will not be adequate.

Definition 2 (Transient state and recurrent state)**.** *A state i is transient if*

$$P_i(X_n = i, \text{for infinitely many } n) = 0\ ,$$

elsewhere, a state i is recurrent (persistent) if

$$P_i(X_n = i, \text{for infinitely many } n) = 1.$$

Transience definition means that once the process arrives to some point is possible that it never comes back to the point i. In the other hand, if i is a recurrent point, sooner or later the process will come back.

An inessential state is transient and a recurrent state is essential. In the case of finite chains, i us transient iff it is inessential.
All the states of an irreducible chain are of the same type: all transient or all recurrent.
A finite Markov chain contains at least one persistent state.

Theorem 2.1 (General Ergodic Theorem)**.** *Let P be the TPM of a primitive Markov chain with a countable state space S. If the Markov chain is transient or null recurrent, then* $\forall i, j \in S$*;*

$$\lim_{n\to\infty} p_{ij}^{(n)} \to 0$$

if the chain is no null recurrent, then $\forall i, j \in S$*,*

$$\lim_{n\to\infty} p_{ij}^{(n)} = v_n$$

exists and is independent of i. The probability vector $V = (v_1, v_2, \dots)$ *is the unique invariant measure of P. Further if* μ_{jj} *is the mean recurrence time of state j, then*

$$v_j = (\mu_{jj})^{-1}.$$

2.3 Continuous-time Markov Chains

It is time to consider now continuous-time Markov Chains. Let $X(t)$ be a Markov process with countable state space S. The transition probability function given by:

$$p_{ij}(t) = P(X(t+u) = j \mid X(u) = i), \qquad t > 0 \qquad i, j \in S$$

is clearly independent of $u \geq 0$ Then $\forall t$, we denote the TPM as:

$$P(t) = (p_{ij}(t))) \qquad i, j \in S.$$

Setting $p_{ij}(0) = \delta_{ij}$ the initial condition can be put as $P(0) = I$ and the probability that the system is at state j at time t by:

$$\pi_j(t) = P\{X(t) = j\},$$

therefore the vector $\pi(t) = \{\pi_1(t), \pi_2(t), \dots\}$ is the probability vector of the state of the system at time t, then $\pi(0)$ is the initial probability vector.

It is possible to get $\pi_j(t)$ in terms of the initial probability vector and the transition probability matrix:
First of all we use the conditioned probability definition

$$\pi_j(t) = \sum_i P(X(t+u)) = j \mid X(u) = i)P(X(u) = i),$$

now we use that the probability is independent of u

$$\pi_j(t) = \sum_i p_{ij}(t)P(X(0) = i) = \sum_i p_{ij}(t)\pi_i(0)$$

which in matrix form results in:

$$\pi(t) = \pi(0)P(t) \ .$$

2.4 Sojourn time

The waiting time for a change from state i is called the sojourn time at state i and noted as τ_i. Then denote the cumulative distributive function of the sojourn (waiting) time as:

$$\bar{F}_i(u) = P(\tau_i > u \mid X(0) = i) \qquad u \geq 0.$$

If we impose the function to satisfy $g(t+s) = g(t)g(s)$ and to be right-continuous, the only solution it gives is: $\bar{F}_i(u) = e^{-a_i u}$, which is the exponential distribution.
This fact gives us another way of defining a continuous-time Markov Chain:

- The amount of time it spends at state i once it enters state i is exponentially distributed with parameter a_i
- When the process leaves state i, it next enters state j with probability p_{ij} satisfying: $\sum_j p_{ij} = 1, \quad j \in S$

This way a continuous-time Markov chain is completely defined giving all the parameters $a_i \ \forall i$ for every state the system has.

Further, the sojourn times τ_i and τ_j are independent. This leads us to Chapman-Kolgomorov equation;

$$p_{ij}(T+t) = \sum_k p_{ik}(T)p_{kj}(t) \qquad i,j,k \in S$$

or in Matrix form:

$$P(T+t) = P(T)P(t).$$

2.5 Transition density matrix

Let's denote the right-hand derivative at $t = 0$, by

$$q_{ij} = \lim_{h \to 0} \frac{p_{ij}(h) - p_{ij}(0)}{h} = \lim_{h \to 0} \frac{p_{ij}(h)}{h}, \quad i \neq j$$

where we used that $p_{ij}(0) = 0$ because at $t = 0$ the system can't be at state j, and also:

$$q_{ii} = \lim_{h \to 0} \frac{p_{ii}(h) - p_{ii}(0)}{h} = \lim_{h \to 0} \frac{p_{ij}(h) - 1}{h}$$

where obviously $p_{ii}(0) = 1$ because it is certain that the state of the system at $t = 0$ is i.
We are writing $q_i = -q_{ii}$
Writing $Q = q_{ij}$, we get a matrix that can be denoted in matrix notation as:

$$Q = \lim_{h \to 0} \frac{P(h) - I}{h},$$

we got that:

$$\sum_j p_{ij}(h) = 1,$$

$$\sum_{j \neq i} p_{ij}(h) + \underbrace{p_{ii} - 1}_{q_{ii}} = 0,$$

whence we get:

$$\sum_{j \neq i} q_{ij} + q_{ii} = 0.$$

Finally:

$$\sum_{j \neq i} q_{ij} = q_i.$$

The $Q - matrix$ is called also **transition density matrix** or **rate matrix**. It is a matrix satisfying that its diagonal elements are negative while non-diagonal are positive, and also the sum of each row is 0. Q-matrix is used to describe the rate a continuous time Markov chain moves between states.

2.6 Limiting behaviour

Continuous Markov Chains are very similar to discrete-time chains when related to classification. A state j is said to be accessible from $i (i \to j)$ if $p_{ij}(t) > 0$ for some t. Both states communicate if $i \to j$ and $j \to i$. If every state can be reached from any other one, then it is said that the chain is irreducible.

As before, we can also divide states between persistent and transient. If we denote α_{ij} as the first entrance time from state i to state j and use F as its distribution function, we get:

$$F_{ij}(t) = P(\alpha_{ij} < t).$$

Then state i is called persistent if:

$$\lim_{t\to\infty} F_{ii}(t) = 1$$

and transient otherwise. The definition means that if i is persistent, the system is always coming back to i whether if i is transient, there is a path that makes the system to never come back to i, even if $t \to \infty$.

Using Chapman-Kolgomorov equation we get:

$$p_{ij}(h+t) = \sum_k p_{ik}(h)p_{kj}(t) =$$

$$= \sum_{k\neq i} p_{ik}(h)p_{kj}(t) + p_{ii}(h)p_{ij}(t)$$

where we separated i from k. So that:

$$\frac{p_{ij}(h+t) - p_{ij}(t)}{h} = \sum_{k\neq i} \frac{p_{ik}(h)}{h} p_{kj}(t) + \left(\frac{p_{ii}(h)-1}{h}\right) p_{ij}(t)$$

where we used the last equality and the definition of derivative, when $h \to 0$. Now if we operate the limits and the summations:

$$\lim_{h\to 0} \frac{p_{ij}(h+t) - p_{ij}(t)}{h} = \sum_{k\neq i} \underbrace{\left[\lim_{h\to 0} \frac{p_{ik}(h)}{h}\right]}_{q_{ik}} p_{kj}(t) + \underbrace{\left[\lim_{h\to 0} \frac{p_{ii}(h)-1}{h}\right]}_{q_i} p_{ik}(t).$$

In the other hand, using $p'_{ij}(t)$ as the derivative of p and q_{ik} as the derivative of p_{ik}, we are able to write:

$$p'_{ij}(t) = \sum_{k\neq i} q_{ik}p_{kj}(t) + q_i p_{ij}(t)$$

which is another form of the Chapman-Kolgomorov equation that can be written in matrix form using Q-matrix as:

$$P'(t) = QP(t).$$

Note that this is one way to look at the transition, analogously we can obtain the other point of view of the system change, expressed as:

$$p'_{ij}(t) = \sum_{k\neq i} p_{ik}q_{kj}(t) + q_j p_{ij}(t),$$

$$P'(t) = P(t)Q.$$

2.7 Transient solution

Considering a finite $(m+1)$ state chain with a rate matrix Q, if we solve the Chapman-Kolgomorov equation (with $P(0) = I$) we obtain:

$$P(t) = e^{Qt} = I + \sum_{n=1}^{\infty} \frac{Q^n t^n}{n!}$$

which can be noted using probability vector as:

$$\pi(t) = \pi(0) \left(I + \sum_{n=1}^{\infty} \frac{Q^n t^n}{n!} \right).$$

As seen in Linear Algebra, if d_i, eigenvalues of Q are all distinct, being D the diagonal matrix that have d_i as its elements. Then there exists a base change that transforms Q into D using matrices H:

$$Q = HDH^{-1}$$

this fact is useful to make the power of matrix Q as:

$$Q^n = HD^nH^{-1}.$$

As $\sum_{n=1}^{\infty} \frac{t^n}{n!}$ is the definition of the exponential function, then we can substitute both expressions in the the equality: $P(t) = H \wedge (t)H^{-1}$ where $\wedge$ is the diagonal matrix formed by $e^{d_i t}$. It can be also noted using the probability vector:

$$\pi(t) = \pi(0)P(t),$$

this way it is possible to obtain a transient solution (analytic) once we have $P(t)$ isolated. As it is said before, continuous-time Markov Chains can be seen as stochastic process such that the sojourn time in state i is an exponential distribution with mean a_i depending on its current state and the probabilities to jump to any other state in S. This way it is possible to get a relation between a_i and q_{ij}.

We defined q_{ij} as $\lim_{h \to 0} \frac{p_{ij}(h)}{h} = q_{ij}$. Isolating $p_{ij}(h)$ as the probability that the state of the process changes from i to j in an infinitesimal interval h we obtain:

$$p_{ij}(h) = hq_{ij} + o(h).$$

If we think of $p_{ij}(h)$ as the probability that the sojourn time being lesser than h we can also express $p_{ij}(h)$ as:

$$p_{ij}(h) = ha_i p_{ij} + o(h).$$

Moving h and taking the limit as $h \to 0$, we obtain:

$$\underbrace{\lim_{h \to 0} \frac{p_{ij}(h)}{h}}_{q_{ij}} = a_i p_{ij}.$$

Finally:

$$q_{ij} = a_i p_{ij}.$$

Now, for the rest of states:

$$1 - p_{ii}(h) = a_i h \underbrace{\sum_j p_{ij}}_{1} + o(h) = a_i h + o(h),$$

Moving h:

$$\underbrace{\lim_{h \to 0} \frac{1 - p_{ii}(h)}{h}}_{q_i} = a_i$$

$$q_i = a_i.$$

Now, Q-matrix can be written as:

$$\begin{bmatrix} -a_0 & a_0 p_{01} & \dots & a_0 p_{0m} \\ a_1 p_{10} & -a_1 & \dots & a_1 p_{1m} \\ \dots & \dots & \dots & \dots \\ a_m p_{m0} & a_m p_{m1} & \dots & -a_m \end{bmatrix}$$

Chapter 3

Birth/Death Processes

In order to study queues, the first step is to study some Markov Chains properties. Once we already done that part, now it is time to start studying one of the most important subclasses: Birth-and-death processes. Queues are a type of birth-and-death processes so it is important to look at them before we even start, analysing queues. They are called birth and death processes because there is only two types of transitions: births which increase the state of the variable by one and deaths which decrease it by one.

Definition 3 (Birth-and-Death Processes)**.** *A birth-death process refers a Markov process with space state* $\mathbb{N}$ *or a subset of* $\mathbb{N}$*, where transitions can only be to a neighbouring state.*

If there is a continuous Markov Chain $\{X(t), t \in T\}$ with a discrete state space $S \in \mathbb{N}$ and with rates:

$$\begin{array}{lcc} q_{i,i+1} = \lambda_i & \forall i & \text{(rate of growth)} \\ q_{i,i-1} = \mu_i & \forall i > 0 & \text{(rate of loss)} \\ q_{i,j} = 0 & j \neq i \pm 1 & \forall i \\ q_i = (\lambda_i + \mu_i) & & \forall i \end{array}$$

is called:

- a pure birth process if $\mu_i = 0 \quad \forall i$
- a pure death process if $\lambda_i = 0 \quad \forall i$
- a birth-and-death process if there are $\exists i, j$ such that $\lambda_i > 0$ and $\mu_j > 0$.

Now, we can get Chapman-Kolgomorov equations for the birth-and-death processes. We previously saw that: $p'_{ij}(t) = \sum_{k \neq i} p_{ik}(t) q_{kj} + q_j p_{ij}(t)$. But now $q_{ij} = 0$ when i and j are not neighbouring states so we can rewrite the sum using only the boundary states.

$$p'_{ij}(t) = -(\lambda_j + \mu j) p_{ij}(t) + \lambda_{j-1} p_{i,j-1}(t) + \mu_{j+1} p_{i,j+1}(t) \quad \forall j \geq 1 \quad j \in \mathbb{N}$$

but then when $j = 0$:

$$p'_{i,0}(t) = -\lambda_0 p_{i,0}(t) + \mu_1 p_{i,1}(t).$$

For any state j in a given time t, we denote $P_j(t) = P(X(t) = j)$. If we assume that at time $t = 0$ system starts at state i (we can do that because the process is memoryless), then we can write:

$$P_j'(t) = -(\lambda_j + \mu_j)P_j(t) + \lambda_{j-1}P_{j-1}(t) + \mu_{j+1}P_{j+1}(t),$$

$$P_0'(t) = -\lambda_0 P_0(t) + \mu_1 P_1(t).$$

If λ_i and μ_i are not 0 for any i, then the Markov Chain is irreducible, so all the states can communicate between them given enough steps. This chain is persistent (recurrent) as $P(X_n = i, \textit{for infinitely many } n) = 1$ and non-null as the expected time for visiting state i is in $\mathbb{R}$. This is the reason that we can assure that $\lim_{t\to\infty} p_{ij}(t)$ exists and that is independent from initial state i. We are going to denote it as p_j, which is clearly independent of t.

Now we can make limits to the expressions of P' stated above. As we have saw that the limits of p_{ij} are independent from t, the left hand side will be 0:

$$\boxed{0 = -(\lambda_j + \mu_j)p_j + \lambda_{j-1}p_{j-1} + \mu_{j+1}p_{j+1}}$$

$$\boxed{0 = -\lambda_0 p_0 + \mu_1 p_1}$$

with this notation we are able to define:

$$\pi_j = \frac{\lambda_0\lambda_1 \dots \lambda_{j-1}}{\mu_1\mu_2 \dots \mu_j}, \quad j \geq 1, \quad \text{and}$$

$$\pi_0 = 1.$$

This definition of π is useful to write probabilities in function of p_0, doing induction in j:

$$p_1 = \left(\frac{\lambda_0}{\mu_1}\right) p_0 = \pi_1 p_0.$$

If we assume that $p_k = \pi_k p_0$, $\forall k \leq j$, then for $j+1$ using the expression boxed above, we get:

$$\mu_{j+1}p_{j+1} = (\lambda_j + \mu_j) \underbrace{p_j}_{\pi_j p_0} - \lambda_{j-1} \underbrace{p_{j-1}}_{\pi_{j-1} p_0},$$

dividing into μ_j:

$$\frac{\mu_{j+1}p_{j+1}}{\mu_j} = \frac{\lambda_j}{\mu_j}\pi_j p_0 + \pi_j - \underbrace{\frac{\lambda_{j-1}}{\mu_j}\pi_{j-1}}_{\pi_j} p_0$$

Finally, we get the result we were looking for:

$$p_{j+1} = \underbrace{\frac{\lambda_j}{\mu_{j+1}}\pi_j}_{\pi_{j+1}} p_0,$$

$$\boxed{p_{j+1} = \pi_{j+1} p_0}$$

Now we want to prove that if $\sum_{k=0}^{\infty} \pi_k < \infty$ then, $p_j = \frac{\pi_j}{\sum_{k=0}^{\infty} \pi_k}$. We know that $\sum_{k=0}^{\infty} \pi_k p_0 = 1$, as the sum of probabilities has to be 1. Dividing the right hand side of the boxed equation by $\sum_{k=0}^{\infty} \pi_k p_0$ and the left hand side by 1, we get the result we wanted.

$$\boxed{p_j = \frac{\pi_j}{\sum_{k=0}^{\infty} \pi_k}}$$

The condition used, $\sum_{k=0}^{\infty} \pi_k < \infty$ is sufficient for a birth-death process to have all the state non-null persistent.

3.1 Poisson Processes

There is an important type of birth-death processes that has to be studied to work with queueing theory: the Poisson Processes.

Definition 4 (Poisson Process)**.** *A pure birth process with a constant rate of growth λ is called a Poisson Process.*

Using Chapman-Kolgomorov equation with $\mu_i = 0$ and $\lambda_i = \lambda$ for all i, we get the following expressions:

$$\boxed{P_j'(t) = -\lambda[P_j(t) - P_{j-1}(t)] \qquad j \geq 1}$$

$$\boxed{P_0'(t) = -\lambda P_0(t)}$$

Now we want to solve this equation to get an expression of $P_j(t)$. In order to do this we define:

$$P(s,t) = \sum_{j=0}^{\infty} P_j(t) s^j.$$

When $t = 0$, the only term of the sum is the current state i and $P_i(0) = 1$, then we have that $P(s,0) = s^i$.Differentiating term by term $P(s,t)$ expression:

$$\frac{\partial}{\partial t} P(s,t) = \sum_{j=0}^{\infty} \frac{\partial}{\partial t} P_j(t) s^j = P_0'(t) + \sum_{j}^{\infty} P_j'(t) s^j.$$

Now we isolate P_j' and then we multiply the expression boxed above of P_j' by s^j, cancelling P_j with $-P_{j-1}$. We get:

$$\frac{\partial}{\partial t} P(s,t) - P_0'(t) = -\lambda(P(s,t) - P_0(t) - sP(s,t)).$$

We use now the above boxed expression $P_0' = -\lambda P_0$ and it cancels with λP_0 in the right hand side:

$$\frac{\partial}{\partial t}P(s,t) = \lambda P(s,t)(s-1).$$

Solving the separable differential equation and using $P(s,0) = s^i$, we get:

$$P(s,t) = Ce^{\lambda(s-1)t} = s^i e^{\lambda(s-1)t}.$$

As P_j is the coefficient of s^j in $P(s,t)$ this way P_j can be:

$$\begin{cases} e^{-\lambda t}\dfrac{(\lambda t)^{j-i}}{(j-i)!} & if \quad j \geq i \\ 0 & if \quad j > i \end{cases}$$

So P_j is 0 when j is an state previous than i, and it is the exponential of the states that i has to transit to get j if not. With this expression we can get finally the probabilities of a Markov Process, using the first expression in the last equality:

$$P(X(t+s) - \underbrace{X(s)}_{i} = k | X(s) = i) = P(X(t+s) = i + k | X(s) = i) = \boxed{\frac{(\lambda t)^k}{k!}e^{-\lambda t}}$$

There is another way to define a Poisson process. We denote by $N(t)$ the number of times an event happens in an interval of length t, and $P_n(t) = P(N(t) = n)$. Now we make the following postulates:

- **Independence:** The number of events occurring in 2 disjoint intervals are independent. That is, the increments of $N(t)$ are independent random variables.
- **Homogeneity in time:** The random variables $N(t)$ depend only on the length of t.
- **Regularity:** In an interval of infinitesimal length h the probability of exactly one occurrence is $P_1(h) = \lambda h + o(h)$ and the probability of more occurrences is $\sum_{k=2}^{\infty} P_k(h) = o(h)$.

From the last postulate we get that the probability of 0 occurrences is:

$$P_0(h) = 1 - \lambda h + o(h).$$

As we have assumed independence, we can write:

$$P_0(t+h) = P_0(t)P_0(h) = P_0(t)(1 - \lambda h + o(h)),$$

and dividing all the terms by h and taking limits when $h \to 0$, we get:

$$\underbrace{\lim_{h\to 0}\frac{P_0(t+h) - P_0(t)}{h}}_{P_0'(t)} = -\lambda P_0(t) + \underbrace{\lim_{h\to 0}\frac{o(h)}{h}}_{0}$$

$$\boxed{P_0'(t) = -\lambda P_0(t)}$$

Now we want to get an expression of $P_j'(t)$. We use:

$$P_j(t+h) = P_j(t)P_0(h) + P_{j-1}(t)P_1(h) + \cdots + P_1(t)P_{j-1}(h) + P_0(t)P_j(h)$$

where the j occurrences happen in one of the two periods of time, t or h and we use that they are both independent. But we already knew the probabilities of P in an infinitesimal length interval h so:

$$P_j(t+h) = P_j(t)[1 - \lambda h + o(h)] + P_{j-1}(t)[\lambda h + o(h)] + o(h).$$

Finally doing the same step as before, this is, dividing all terms by h and taking limits when $h \to 0$, we get:

$$\underbrace{\lim_{h\to 0} \frac{P_j(t+h) - P_j(t)}{h}}_{P_j'(t)} = -\lambda P_j(t) + \lambda P_{j-1}(t) + \underbrace{\lim_{h\to 0} \frac{o(h)}{h}}_{0},$$

$$\boxed{P_j'(t) = -\lambda[P_j(t) - P_{j-1}(t)]} \quad \text{when } j \geq 1$$

As both boxed expressions are the same as Chapman-Kolgomorov equations we found when considering the Poisson process as a pure birth process with constant growth rate λ, we can see that they will follow the same probability, thus:

$$P_j(t) = e^{-\lambda t}\frac{(\lambda t)^j}{j!}, \qquad \forall j \in \mathbb{Z}.$$

Thus, $N(t)$ is a Poisson Process with parameter λt.
Poisson Processes are important in order to do queueing models. We can consider a interval of time when the system is studied. The moment when a customer enters the queue (the occurrence of the event) is indeed random and can be considered independent from the other consumers. We have λ as the fixed rate of arrival of consumers. It can also be done with the servicing times with another parameter μ. This way we are able to do a model to calculate the probabilities of the system to be in any state in a given moment.

Chapter 4

Queueing models

We have already seen that queues are birth and death processes, a type of Markov processes. We have also defined Poisson processes and we have found some useful theorems as Chapman-Kolgomorov. Now, it is time to look at queueing models ans its own notation and parameters.

4.1 Kendall's notation

A queue model can be used to model a lot of different types of queues: telecommunications, traffic engineering, computing, design, waiting lines...
Independently of the usage of the model, all the queues are characterized by some items such as:

- **The arrival pattern of customers:** It is the manner in which arrivals occur. A good measure is the mean time between two consecutive arrivals or the number of arrivals per unit of time. It can be deterministic (always the same time between arrivals) but usually is stochastic with some distribution specified. Poisson distribution is usually used to describe them.
- **Pattern of service times:** Service times can be deterministic but usually they will be stochastic and independent between them and from the arrival time. Average time of service is a good way of describing it.
- **Number of servers:** Number of servers the system has. If a customer finds more than one free server will choose one randomly and if all of them are busy he will enter the queue. Usually we will consider that there is only one queue independently of the number of servers.
- **Capacity of the system:** A system may have a limitation of the length of the queue. If this happens, new arrivals will be lost until the system has available places.
- **Queue discipline:** The order in which customers leave the queue to get the service: FIFO (First In First Out) is the most usual but there are others such as LIFO (Last In First Out), random order, priority orders...

Kendall introduced a notation to describe some these parameters. It is a 3 part code $(X/Y/Z)$. First letter stands for interarrival time distribution and the second one is used to describe service time distribution. If the distributions are exponential or Poisson we use M, for memoryless. G is for general distribution time and D is for deterministic times.

Last part is a number or a letter which describes the number of servers in the system. Generally, we assume that customers are always allowed to enter the system, there is enough waiting room and that service discipline is FIFO without priority rules.
This notation can be extended, adding another parameter to show the capacity of the system or if the service discipline varies if the assumptions made do not hold.

4.2 Performance measures

There are a few performance measures that describe the queueing models. In this section, we are going to use again λ and μ, rates of growth (birth) and loss (death) of Birth/Death processes, to find some of them. This performance measures such as traffic intensity or server utilization are important in order to understand what is happening in a queue.

Traffic intensity is expressed as $a = \lambda/\mu$ and it is the mean number of arrivals per unit of time. It can also be called *offered load* and is a measure of what the customers want.
Server utilization is also called *utilization factor* or *carried load* and it stands for $\rho = a/c$ where c is the number of servers.

The analysis of queueing systems when arrival and service times are deterministic are very simple and do not present any difficulty. That is the reason why from now on, we will always speak of models where arrival and service times are stochastic.

In order to describe the system, we will need some more performance measures such as:

- $N(t)$: Number of customers in the system (in the queue and being served) at time t. We are interested in knowing its distribution.
- W_n is the waiting time of the nth arrival. It can be interesting to know this distribution.
- Distribution of time when the server is busy.
- Distribution of $W(t)$: the time a customer has to wait if he has arrived at time t

Knowing all the distributions stated above we will be able to study problems related to the nature of solutions and the method needed to find it.

4.3 Transients and Steady-State behaviour

We said that $N(t)$ is the number of customers both waiting and being served in the system at time t. Now let's define the probability distribution of it:

$$p_n(t) = P(N(t) = n) \qquad n \in \mathbb{N},$$

$$p_i(0) = 1.$$

This expression implies that in the initial time $t = 0$ there where i customers in the system. In order to obtain a complete description of the queue, we should found a time dependent solution $p_n(t)$ for all times. As this can be hard to do, for practical situations the equilibrium behaviour is actually enough. This means that we are interested in knowing (if it exists):

$$p_n = \lim_{t\to\infty} p_n(t) \qquad n \in \mathbb{N}\ .$$

Thus, $p_n(t)$ is the limiting probability that there are n customers at equilibrium. Obviously it is independent of the initial number of customers at $t = 0$. If this limit exists we say the system reaches an equilibrium or a steady state, p_n is called the steady-state probability that there are n customers in the system. Then it is clear that if the system reaches an equilibrium, we get the normalizing condition:

$$\sum_{n=0}^{\infty} p_n = 1\ .$$

We need that the limit p_n exists for the system to reach an equilibrium. It is necessary to know when this limit exists, so we need to consider two other limiting probabilities:

- a_n is the probability that arriving customers find n customers when they arrive.
- d_n is the probability that leaving customers leave n customers when they depart.

Theorem 4.1 (Burke's Theorem)**.** *In any queueing system in which arrivals and departures occur one by one and that has reached equilibrium state,*

$$a_n = d_n \quad \forall n \geq 0\ .$$

Proof: Consider a system in equilibrium where we see an arrival. Then the number of the system will be $n + 1$. After that, as the system is in equilibrium, a departure will leave the system so the number will decrease from $n + 1$ to n. In any interval of time T the number of transitions A from n to $n + 1$ and the number of transitions B from $n + 1$ to n will differ at most by 1, so $A = B$ or $A - B = \pm 1$. When T is large, then the ratio of transitions will be the same $\frac{A}{T} = \frac{B}{T}$. Thus, on average, arrivals and departures always see the same number of customers so $a_n = d_n$ for every n as we wanted to prove. □

Following this theorem we can state an equality principle which holds for systems in steady state. It is true for systems in steady state that the rate at which a process enters the state n is equal to the rate at which the process leaves state n. In other words, rates of entering and leaving a particular state are equal.

4.4 Little's Law

Little's law gives a very important relation between L,expected units in the system; W, expected waiting time and λ, the mean arriving time:

$$L = \lambda W.$$

It is assumed that the capacity of the system is sufficient to deal with the customers at all time. If we denote by L_Q and W_Q the number of customers in the queue and their queueing time we have a similar relationship:

$$L_Q = \lambda W_Q.$$

Proof: Suppose that we have a system which is already in equilibrium. Consider a time interval T and define:

- $A(T)$: Total number of arrivals during T
- $B(T)$: Total waiting time of all the customers during T
- $\lambda(T)$: mean arrival rate during T. As we defined λ, we have that: $\lambda(T) = \dfrac{A(T)}{T}$.

The mean waiting time, will be $W(T) = \dfrac{B(T)}{A(T)}$.

We can found the mean number of customers in the system during T: $L(T) = \dfrac{B(T)}{T}$.
Multiplying and dividing the expression of $L(T)$ by $A(T)$, we got:

$$L(T) = \frac{B(T)}{T} = \underbrace{\frac{B(T)}{A(T)}}_{W(T)} \underbrace{\frac{A(T)}{T}}_{\lambda(T)} = W(T)\lambda(T) \ .$$

Now, as limits when $T \to \infty$ exist and are given by:

$$\lim_{T\to\infty} \lambda(T) = \lambda \ ,$$

$$\lim_{T\to\infty} W(T) = W \ ,$$

$$\lim_{T\to\infty} L(T) = L \ .$$

We get the relation we wanted:

$$L = \lambda W.$$

Similarly we can get the same formula for L_Q and W_Q defining $B(T)$ as the total time the customers is in the queue rather than in the system.

□

4.5 Poison's arrivals: PASTA property

As we said before many arrival processes can be taken as Poisson processes, as they are memoryless. These are, for example, the cases of the queues in supermarkets, telephone calls received at a switchboard, orders for modems...

Poisson arrivals also hold a interesting property: arriving customers find on average the same situation in the queueing system as an outside observer looking at the system at an arbitrary point of time. This property of Poisson arrivals is called PASTA, which is the acronym for Poisson Arrivals See Time Average. We already had defined a_n as the probability that the customer who arrives into the system finds n customers there. Using this definition, PASTA property can be written as:

$$a_n = p_n \quad \forall n \geq 1 \, .$$

Let's try to proof it. Let $A(t, t+\delta)$ be the number of arrivals in the infinitesimal interval $(t, t+\delta)$, and $a_n(t)$ the probability of a_n happening at time t:

$$\begin{aligned} a_n(t) &= \lim_{\delta \to 0} P(N(t) = n \mid \text{ an arrival occurred just after t}) \\ &= \lim_{\delta \to 0} P(N(t) = n \mid A(t, t+\delta) = 1) \\ &= \lim_{\delta \to 0} \frac{P(N(t) = n, A(t, t+\delta) = 1)}{P(A(t, t+\delta) = 1)} \\ &= \lim_{\delta \to 0} \frac{P(A(t, t+\delta) = 1 \mid N(t) = n) P(N(t) = n)}{P(A(t, t+\delta) = 1)} \end{aligned}$$

but as Poisson porcesses are memoryless:

$$P(A(t, t+\delta) = 1 \mid N(t) = n) = P(A(t, t+\delta) = 1),$$

so:

$$a_n(t) = \lim_{\delta \to 0} P(N(t) = n) = p_n(t).$$

and in steady state,

$$\boxed{a_n = p_n \quad \forall n \geq 0}$$

□

This way we have seen that in a queue with Poisson arrivals, the proportion of arrivals that find the system in a stet n is the same as the proportion of time the state spends in state n.

4.6 M/M/1 queue

4.6.1 Introduction

This is the first queue we are going to study and it is the most simple case possible. It stands for a situation of only one server where interarrival times are independent exponentially distributed with mean $\frac{1}{\lambda}$. Customers are served in order of arrival and their service time is also exponentially distributed with mean μ. The carried load (utlization factor) is $\rho = \lambda/\mu$ and it is the same as the traffic intensity (a) as there is only one server. Which conditions should we impose to the system to reach a steady state? If we assume this steady state exists:

$$p_n = \lim_{t\to\infty} P(N(t) = n), \quad \forall n \in \mathbb{N}$$

where $N(t)$ is the number of people in the system at instant t and p_n is both the probability of the system of being at state n when $t \to \infty$ and, by the PASTA property, the proportion of time the process is in state n. Our goal is to find p_n.

4.6.2 Steady state distribution

Now we are assuming that the system is in steady state. Let's consider state n. At this point the system can go up to the next state $n+1$ (that is a new customer enters the system) at rate λp_n or it can come down from state $n+1$ to state n (a client has been served and leaves) at rate μp_{n+1}. Any other different change of state is not possible as we assumed that the system is in steady state. As we are in a steady state is obvious that all the customers entering the system have to be the same amount as the customers who leave it, once they are already served:

$$\lambda p_n = \mu p_{n+1} \ ,$$

$$p_{n+1} = \frac{\lambda}{\mu} p_n = a p_n = a^2 p_{n-1} = \ldots = a^{n+1} p_0 \ ,$$

$$p_n = a^n p_0 \ .$$

We know that $\sum_{n=0}^{\infty} p_n = 1$, which means that: $a_0 p_0 + a_1 p_0 + a_2 p_0 + \ldots + a_{n-1} p_0 + a_n p_0 = 1$.
Now removing common factor and assuming that $a < 1$:

$$1 = \sum_{n=0}^{\infty} p_n = p_0(a_0 + a_1 + \ldots) = \frac{p_0}{1-a} \ ,$$

$$p_0 = (1-a) \ ,$$

where we used the formula for summing geometrical series.
Now using the expression of p_n above, we get:

$$p_n = (1-a)a^n \ ,$$

but $\rho = a$ as we said before, so we finally got an expression for p_n in terms of carried load:

$$\boxed{p_n = (1-\rho)\rho^n}$$

As we can see, the steady distribution depends only of the ratio between $\frac{\lambda}{\mu}$ which is ρ.We need that $\rho < 1$, otherwise the system will never get to a steady state.

4.6.3 Number of the system

Let's find some performance measures, beginning with the expected value of the number of people in the system $E(N)$.

$$E(N) = \sum_{n=0}^{\infty} np_n = \sum_{n=1}^{\infty} n \underbrace{(1-\rho)\rho^n}_{p_n} = \rho(1-\rho) \underbrace{\sum_{n=1}^{\infty} n\rho^{n-1}}_{\frac{1}{1-\rho^2}},$$

$$\boxed{E(N) = \frac{\rho}{1-\rho}}$$

where in the sum of the series we assumed that $\rho < 1$(the system won't get to a steady state if it doesn't) and derived the solution from: $\frac{1}{1-\rho} \sum_{n=0}^{\infty} \rho^n$.

If we find $E(N^2)$ we will be able to compute variance.

$$E(N^2) = \sum_{n=0}^{\infty} n^2 p_n = \sum_{n=1}^{\infty} n^2(1-\rho)\rho^n = (1-\rho) \sum_{n=1}^{\infty} n^2\rho^n .$$

To solve this series we should assume again that $\rho < 1$ and the deriving twice and multiplying by ρ twice, we get that:

$$\sum_{n=1}^{\infty} n^2\rho^n = \frac{\rho^2+\rho}{(1-\rho)^3},$$

$$\boxed{E(N^2) = \frac{\rho^2+\rho}{(1-\rho)^2}}$$

Computing variance:

$$var(N) = E(N^2) - (E(N))^2 = \frac{\rho+\rho^2}{(1-\rho)^2} - \frac{\rho^2}{(1-\rho)^2}$$

$$\boxed{var(N) = \frac{\rho}{(1-\rho)^2}}$$

Now we can use Little's formula, $L = \lambda W$, to compute the expected waiting time in the system:

$$E(W) = \frac{E(N)}{\lambda} = \frac{\rho}{\lambda(1-\rho)},$$

$$\boxed{E(W) = \frac{1}{\mu(1-\rho)}}$$

where in the last equality we used that $\rho = \frac{\lambda}{\mu}$.

4.6.4 Sojourn time

Now we want to derive the distribution of the sojourn time W. We denoted a_n the number of customers in the system before an arrival, and v_k the service time of the kth customer. As the service time follows an exponential distribution, which is memoryless, we get that all the random variables v_k are independent and distributed with mean $1/\mu$. We also defined W_q as the time a customer is waiting in the queue. Seems logical that the time when n customers are served is equal to the time that customer n will have to wait, $W_q = S_n$.
As $S_n = v_1 + v_2 + \ldots + v_n$, S_n is a sum of exponential distributions with mean $1/\mu$; it is a gamma distribution having a density function:

$$\frac{\mu^n x^{n-1} e^{-\mu x}}{\Gamma(n)} \ .$$

Hence conditioning it to the number of units in the system at certain moment x:

$$w_q(x)dx = \sum_{n=1}^{\infty} \frac{\mu^n x^{n-1} e^{-\mu x}}{\Gamma(n)} dx(a_n) \ ,$$

and by PASTA property, we know that the fraction of customers that find n customers in the system is equal to the fraction of time that there are n customers in the system so $a_n = p_n = (1-\rho)\rho^n$:

$$w_q(x) = \sum_{n=1}^{\infty} \frac{\mu^n x^{n-1} e^{-\mu x}}{\Gamma(n)} (1-\rho)\rho^n \ .$$

Now subtracting some terms from the sum:

$$w_q(x) = \mu e^{-\mu x}(1-\rho)\rho \underbrace{\sum_{n=1}^{\infty} \frac{(\mu\rho x)^{n-1}}{(n-1)!}}_{e^{\mu\rho x}} =$$

$$= \mu\rho(1-\rho)e^{\mu x(\rho-1)} \ .$$

Thus probability density function of W_q is given by:

$$w_q(x) = \begin{cases} p_0 = 1-\rho, & \text{if } x = 0 \\ \mu\rho(1-\rho)e^{\mu x(\rho-1)} & \text{if } x > 0 \ . \end{cases}$$

For $x = 0$ is a different probability because if there is nobody in the system, the customer won't need to do any queue. Now, if we want to know the mean sojourn time a customer

stays in the system:

$$W(x) = P(W(x) > x) = P\left(\sum_{k=1}^{a_n+1} S_n > x\right) = \underbrace{\sum_{n=0}^{\infty} P\left(\sum_{k=1}^{n+1} S_n\right)}_{\text{time waited for each customer}} \underbrace{(1-\rho)\rho^n}_{p_n} .$$

Using PASTA property and the fact that S_n is a sum of exponential distributions, a Gamma distribution, we get:

$$W(x) = \sum_{n=0}^{\infty}\sum_{k=0}^{n} \frac{(\mu x)^k}{k!} e^{-\mu x}(1-\rho)\rho^n =$$

$$= \sum_{k=0}^{\infty}\sum_{n=k}^{\infty} \frac{(\mu x)^k}{k!} e^{-\mu x}(1-\rho)\rho^n =$$

when $\rho < 1$:

$$= \underbrace{\sum_{k=0}^{\infty} \frac{(\mu\rho x)^k}{k!}}_{e^{\mu\rho x}} e^{-\mu x} ,$$

$$\boxed{W(x) = e^{-\mu x(1-\rho)}}$$

We can see that the waiting time is exponentially distributed with parameter $\mu(1-\rho)$.

4.7 M/M/1/K queue

4.7.1 Introduction

For the queue M/M/1 we had assumed that the system has enough room for any number of customers. Now in M/M/1/K model we are going to consider that there is only space for K customers in the system. This restriction in the waiting space is known as *finite buffer*. Customers arrive and are served following a Poisson distribution as in the previous case but now if there are more than K customers in the system they will be lost. This model could be applied in ATM queues: people can wait some time but if they see that there's too much people in the queue they will leave without withdrawing money.

4.7.2 Steady state distribution

First of all, we want to know about the steady state of the queue. The system behaves as a simple M/M/1 while the number of the system is less than K. Once it gets there, only departures are possible. In steady state we have:

$$\lambda p_n = \mu p_{n+1} \qquad n = 0, 1, \ldots K-1 .$$

If we use $a = \lambda/\mu$ and express p_n as a function of p_0 like we did before in the M/M/1 case:

$$p_n = p_0 a^n .$$

Now we use that $\sum_{i=0}^{K} p_i = 1$ in the expression above to get:

$$p_0 \sum_{n=0}^{K} a^n = 1 \, .$$

Solving the geometrical series:

$$p_0 = \frac{1}{\sum_{n=0}^{K} a^n} = \frac{1-a}{1-a^{K+1}} \quad \text{if } \lambda \neq \mu \, ,$$

$$p_0 = \frac{1}{K+1} \quad \text{if } \lambda = \mu \, .$$

Thus for p_n, with $n \leq K$ (otherwise it would be 0):

$$p_n = p_0 a^n = \frac{(1-a)a^n}{1-a^{K+1}} \quad \text{if } \lambda \neq \mu \, ,$$

$$p_n = \frac{1}{K+1} \quad \text{if } \lambda = \mu \, .$$

Now we can find its probability generating function:

$$G(s) = \sum_{n=0}^{K} p_n s^n = \frac{1-a}{1-a^{K+1}} \left(\frac{1-(as)^{K+1}}{1-as} \right) \quad \text{for } a \neq 1 \, .$$

As we can see the distribution of the number in the system is truncated geometric when $a \neq 1$ and uniform if $a = 1$.

4.7.3 Number in the system

Now we want to guess the expected number in the system. If $\lambda = \mu$:

$$N_K = \sum_{n=0}^{K} n p_n = \sum_{n=0}^{K} \frac{n}{K+1}$$

Using the formula for summing arithmetic series:

$$N_K = \frac{(K+1)(\frac{K}{K+1}) + 0}{2}$$

$$\boxed{N_K = \frac{K}{2}}$$

If $\lambda \neq \mu$ then:

$$N_K = \sum_{n=0}^{K} n \underbrace{\frac{(1-a)a^n}{1-a^{K+1}}}_{p_n} =$$

$$= \frac{(1-a)a}{1-a^{K+1}} \sum_{n=0}^{K} n a^{n-1} .$$

Using the formula for summing finite geometric series and deriving the result:

$$N_K = \frac{(1-a)a}{1-a^{K+1}} \frac{1-(K+1)a^K + Ka^{K+1}}{(1-a)^2} ,$$

$$\boxed{N_K = \frac{a}{1-a} \frac{1-(K+1)a^K + Ka^{K+1}}{1-a^{K+1}}} .$$

It is important to note that as the finite series $\sum_{n=0}^{K} a^n$ gives a value for any a, the steady solution will always exists in this type of queue.

4.7.4 System loss

It is important to guess the number of arrivals our system is not able to manage and get lost. The average number of customers that the system loses can be noted as the arrivals once the system is in state K, so:

$$\boxed{p_K = \frac{(1-a)a^K}{1-a^{K+1}} \quad \text{if } a \neq 1} .$$

$$\boxed{p_K = \frac{1}{K+1} \quad \text{if } a = 1} .$$

In this type of queue, even when arrivals occur in a Poisson distribution, PASTA property does not hold. This happens because the distribution is truncated. In this case the probability of an arrival finding n in the system is obtained through Baye's theorem:

$$a_n = P(\text{n in the system} \mid \text{an arrival is about to occur}) =$$

$$= \frac{P(\text{an arrival is about to occur} \mid \text{n in the system}) p_n}{\sum_{k=0}^{K-1} P(\text{an arrival is about to occur} \mid \text{k in the system})} =$$

$$= \frac{\lambda p_n}{\sum_{k=0}^{K-1} \lambda p_k} = \frac{p_n}{1-p_K}$$

where in the last equality we used that $\sum_{n=0}^{K} p_k = 1$.

4.8 M/M/∞ queue

4.8.1 Introduction

In this section, we are considering an exponential model with an infinite number of servers. This model is appropriate to study situations like a self-service buffet. As there is always an idle server, all the customers will be served once they arrive, so there won't be any waiting time. Like all the queues done before it is a birth-death model where:

$$\lambda_n = \lambda \quad \forall n \in \mathbb{N} \, ,$$

$$\mu_n = n\mu \quad \forall n \in \mathbb{N} \backslash \{0\}$$

4.8.2 Steady state distribution

Our goal here is to find an expression for p_n once the system gets to an steady state, so:

$$p_n = \prod_{k=0}^{n-1} \frac{\lambda_k}{\mu_{k+1}} = p_0 \prod_{k=0}^{n-1} \frac{\lambda}{(k+1)\mu} =$$

$$= p_0 \frac{\lambda^n}{(\mu)(2\mu)(3\mu)\dots(n\mu)} = p_0 \frac{(\lambda/\mu)^n}{n!} \quad \forall n \in \mathbb{N}$$

We need to find the value of p_0. We know that $1 = \sum_{n=0}^{\infty} p_n$ so:

$$1 = \underbrace{\sum_{n=0}^{\infty} \frac{(\lambda/\mu)^n}{n!}}_{e^{\lambda/\mu}} p_0 \, ,$$

$$p_0 = e^{-\lambda/\mu} \, .$$

Using this in the expression of p_n we got:

$$\boxed{p_n = \frac{e^{-\lambda/\mu}(\lambda/\mu)^n}{n!} \quad \forall n \in \mathbb{N}} \, .$$

As we can see the distribution is a Poisson with mean λ/μ.

4.8.3 Number in the system

Here we want to guess the expected number of customers in the system. In order to do it we should solve the expected value of n:

$$N = \sum_{n=0}^{\infty} n p_n = \sum_{n=0}^{\infty} n \frac{e^{-\lambda/\mu}(\lambda/\mu)^n}{n!} =$$

$$= e^{-\lambda/\mu} \sum_{n=0}^{\infty} \frac{(\lambda/\mu)^n}{(n-1)!} =$$

$$= e^{-\lambda/\mu}\frac{\lambda}{\mu}\sum_{n=0}^{\infty}\frac{(\lambda/\mu)^{n-1}}{(n-1)!} = e^{-\lambda/\mu}\frac{\lambda}{\mu}\underbrace{\sum_{n=1}^{\infty}\frac{(\lambda/\mu)^{n}}{(n)!}}_{e^{\lambda/\mu}},$$

$$\boxed{N = \frac{\lambda}{\mu}}\ .$$

As it is obvious the expected response time will be $1/\mu$ as it is the average service time because no customer will have to wait if there are infinite servers.

4.9 M/M/c queue

4.9.1 Introduction

This queue model is suitable for some supermarket queues. We consider a Poisson imput arrivals with mean λ and c service channels each of them has an equal service rate μ. Every customer will be served by the first idle server. If none is idle, then the customer enters the queue until one of the current customers leaves the system. Thus, if there are $n \leq$ servers busy at some point, their rate of leaving service will be $n\mu$. On the other hand, if there are more than n people in the system, their rate of leaving service will be $c\mu$ as all the servers will be already busy. As always the system is a birth-death process but while the birth rate is constantly equal λ, death rate is now:

$$\mu_n = n\mu \quad \text{if } n \in \mathbb{N} \quad \text{and } n \leq c\ ,$$

$$\mu_n = c\mu \quad \text{if } n \in \mathbb{N} \quad \text{and } n > c\ .$$

4.9.2 Steady state distribution

Let's denote $\rho = \frac{\lambda}{c\mu}$. Assume that the steady state exists and the system is in it. Now we can express p_n for $n \leq c$ as it follows, using λ and μ :

$$\boxed{p_n = \prod_{k=0}^{n-1}\frac{\lambda}{(k+1)\mu}p_0 = \frac{(\lambda/\mu)^n}{n!}p_0}$$

when $n > c$, then p_n:

$$p_n = \prod_{k=0}^{c-1}\frac{\lambda}{(k+1)\mu}\prod_{k=c-1}^{n}\frac{\lambda}{c\mu}\ ,$$

$$p_n = \frac{\lambda^c}{c!\mu^c}\frac{\lambda^{n-c}}{c^{n-c}\mu^{n-c}}p_0 = \frac{\lambda^n}{\mu^n c! c^{n-c}}p_0\ ,$$

$$\boxed{p_n = \frac{(\lambda/\mu)^n}{c!c^{n-c}}p_0}\ .$$

We can express p_n depending on p_{n-1} as it follows for any n:

$$\{min(n,c)\}\mu p_n = \lambda p_{n-1}\ .$$

Now we want to use the normalizing condition $\sum_{n=0}^{\infty} p_n = 1$ to get an expression of p_0

$$1 = p_0 \left(\sum_{n=0}^{c-1} \frac{(\lambda/\mu)^n}{n!} + \sum_{n=c}^{\infty} \frac{(\lambda/\mu)^n}{c!c^{n-c}} \right) ,$$

$$\frac{1}{p_0} = 1 + \sum_{n=1}^{c-1} \frac{(\lambda/\mu)^n}{n!} + \sum_{n=c}^{\infty} \frac{(\lambda/\mu)^n}{c!c^{n-c}} = \sum_{n=0}^{c-1} \frac{(\lambda/\mu)^n}{n!} + \frac{1}{c!c^{-c}} \sum_{n=c}^{\infty} \left(\frac{\lambda}{c\mu} \right)^n .$$

In order for this series to be convergent, we need that $\frac{\lambda}{c\mu} < 1$. It is a condition needed for the system to reach a steady state. Thus, when it happens, we use the geometric formula for summing series:

$$p_0 = \left[\sum_{n=0}^{c-1} \frac{(\lambda/\mu)^n}{n!} + \frac{(\lambda/\mu)^c}{c!(1-\frac{\lambda}{c\mu})} \right]^{-1} .$$

Finally the steady state distribution is given by:

$$\boxed{p_n = \frac{(\lambda/\mu)^n}{n!} \left[\sum_{n=0}^{c-1} \frac{(\lambda/\mu)^n}{n!} + \frac{(\lambda/\mu)^c}{c!(1-\frac{\lambda}{c\mu})} \right]^{-1}} \quad \text{if } n < c ,$$

$$\boxed{p_n = \frac{(\lambda/\mu)^n}{c!c^{n-c}} \left[\sum_{n=0}^{c-1} \frac{(\lambda/\mu)^n}{n!} + \frac{(\lambda/\mu)^c}{c!(1-\frac{\lambda}{c\mu})} \right]^{-1}} \quad \text{if } n \geq c .$$

4.9.3 Number in the system

We have seen that p_n behaves like a Poisson distribution for $n \leq c$ and like a geometric distribution for $n > c$. If we want to know the probability C that an arriving unit has to wait, it is equal to the probability the number in the system is greater than c. This way and using the server utilization $\rho = \frac{\lambda}{c\mu}$:

$$C(c, \frac{\lambda}{\mu}) = P(N \geq c) = \sum_{n=c}^{\infty} p_n =$$

$$= \frac{(\lambda/\mu)^c}{c!(1-\frac{\lambda}{c\mu})} p_0 = \underbrace{\frac{(\lambda/\mu)^c}{c!} p_0}_{p_c} \frac{1}{(1-\rho)} ,$$

$$\boxed{C(c, \frac{\lambda}{\mu}) = \frac{p_c}{1-\rho}} .$$

This is known as Erlang's C formula or Erlang delay probability. It can also be interpreted as the fraction of time when all the servers are busy, using PASTA property.
In order to understand if our servers are enough to give an efficient service, it is useful to know the expected number of busy servers $E(B)$.

$$E(B) = \sum_{n=0}^{c-1} n p_n + \sum_{n=c}^{\infty} c p_n = \sum_{n=0}^{c-1} \frac{(n\lambda/\mu)^n}{n!} p_0 + \frac{c(\lambda/\mu)^c}{c!(1-\rho)} p_0 ,$$

where in the second equality we used that $\rho = \lambda/(c\mu) < 1$. Removing p_0 and λ/μ as a common factor:

$$E(B) = \frac{\lambda}{\mu} p_0 \left[\sum_{n=1}^{c-1} \frac{(\lambda/\mu)^{n-1}}{(n-1)!} + \frac{(\lambda/\mu)^{c-1}}{(c-1)!(1-\rho)} \right] ,$$

transforming the indices $m = n - 1$, we got:

$$= \frac{\lambda}{\mu} p_0 \left[\sum_{m=0}^{c-2} \frac{(\lambda/\mu)^{m}}{(m)!} + \frac{(\lambda/\mu)^{c-1}}{(c-1)!(1-\rho)} \right] ,$$

adding and subtracting ρ to the second term:

$$= \frac{\lambda}{\mu} p_0 \left[\sum_{m=0}^{c-2} \frac{(\lambda/\mu)^{m}}{(m)!} + \frac{\cancel{(1-\rho)}(\lambda/\mu)^{c-1}}{(c-1)!\cancel{(1-\rho)}} + \frac{\rho(\lambda/\mu)^{c-1}}{(c-1)!(1-\rho)} \right]$$

now the second term can be added to the sum, while in the third term we can write $\rho = \lambda/c\mu$, so it becomes:

$$= \frac{\lambda}{\mu} p_0 \underbrace{\left[\sum_{m=0}^{c-1} \frac{(\lambda/\mu)^{m}}{(m)!} + \frac{(\lambda/\mu)^{c}}{(c)!(1-\rho)} \right]}_{p_0^{-1}} ,$$

$$\boxed{E(B) = \frac{\lambda}{\mu} = c\rho} .$$

We used in the last equality the formula for p_0^{-1} that we had found before. So, the expected number of idle servers will be:

$$E(I) = E(c - B) = c - E(B) = c - \underbrace{\frac{\lambda}{\mu}}_{c\rho} ,$$

$$\boxed{E(I) = c(1-\rho)} .$$

Finally, we want to guess the expected number of the system. It seems pretty obvious that it will be sum of the expected number of people being served $E(B)$, that we already have found, plus the expected number in the queue: $E(N) = E(B) + E(Q)$. To find $E(Q)$ we should think that the people that are queueing are people that can't be served, so they are $(n - c)$:

$$E(Q) = \sum_{n=c}^{\infty} (n-c) p_n = \sum_{n=c}^{\infty} (n-c) \frac{(\lambda/\mu)^n}{c! c^{n-c}} p_0 =$$

Using $m = n - c$ we get:

$$= \sum_{m=0}^{\infty} m \frac{(\lambda/\mu)^{m-c}}{c! c^m} p_0 = \frac{(\lambda/\mu)^c}{c!} \sum_{m=0}^{\infty} m \left(\frac{\lambda}{c\mu} \right)^{m-1} p_0 =$$

$$= \underbrace{\frac{(\lambda/\mu)^c}{c!} p_0}_{p_c} \frac{\rho}{(1-\rho)^2} = \frac{p_c \rho}{(1-\rho)^2} ,$$

where in the last equality we used the formula of the sum of geometrical series and derived it once. Using this expression in $E(N)$:

$$\boxed{E(N) = c\rho + \frac{p_c \rho}{(1-\rho)^2}} \ .$$

4.10 Table of results

We are going to do a table of the theoretic results we have found in this section:

	M/M/1	**M/M/1/K**	**M/M/∞**	**M/M/c**
p_0	$1-\rho$	$\frac{1-\rho}{1-\rho^{K+1}}$	$e^{-\lambda/\mu}$	$\left[\sum_{n=0}^{c-1} \frac{(\lambda/\mu)^n}{n!} + \frac{(\lambda/\mu)^c}{c!(1-\frac{\lambda}{c\mu})}\right]^{-1}$
p_n	$(1-\rho)\rho^n$	$\frac{(1-\rho)\rho^n}{1-\rho^{K+1}}$	$\frac{e^{-\lambda/\mu}(\lambda/\mu)^n}{n!}$	$\frac{(\lambda/\mu)^n}{c!c^{n-c}} \left[\sum_{n=0}^{c-1} \frac{(\lambda/\mu)^n}{n!} + \frac{(\lambda/\mu)^c}{c!(1-\frac{\lambda}{c\mu})}\right]^{-1^{(1)}}$
$E(N)$	$\frac{\rho}{1-\rho}$	$\frac{\rho}{1-\rho}\frac{1-(K+1)\rho^K+K\rho^{K+1}}{1-\rho^{K+1}}$	λ/μ	$c\rho + \frac{p_c\rho}{(1-\rho)^2}$

Note that this formula (1) is only right when $n \geq c$. Otherwise, if $n < c$ we had that $p_n = \frac{(\lambda/\mu)^n}{n!}\left[\sum_{n=0}^{c-1} \frac{(\lambda/\mu)^n}{n!} + \frac{(\lambda/\mu)^c}{c!(1-\frac{\lambda}{c\mu})}\right]^{-1}$.

Note also that we have been using for this table the most general formulas we got, that is $\lambda \neq \mu$

Chapter 5

Simulations

5.1 Introduction

We have been studying analytically some queues, so far. Now it is time to simulate this queues to prove that the empiric results are close to that ones we predicted analytically. In order to do it, I will use R which makes it possible to plot results on each type of queue depending on the parameters. As it is a simulation, done with a finite amount of data, it is possible that there are some amount of error among the results the simulation produces and the true value of the parameter we want to know. To reduce the possible errors that an small amount of data can provide, we will run 10000 simulations for 1000 units of time each one.

In this simulation we will be interested in values such as interarrival and service times, number in system, completions in a given time, amount of time servers are busy, mean queue length, mean sojourn time... As we already know the analytic results we will be able to compare them.

5.2 M/M/1 queue

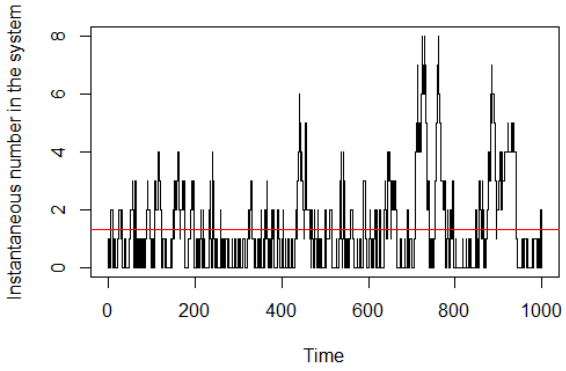
Let's begin with the simplest queue we have done. M/M/1 has only one server and both interarrival ans service times follow an exponential distribution, with mean $1/\lambda$ and μ respectively. Note that if we want that the system gets to an steady state, we saw that we need $\rho = \frac{\lambda}{\mu} < 1$. In a simulation we don't have this limitation because we simulate a finite amount of time. In the other hand, if $\rho \geq 1$ queue will grow a lot and it will be harder to compare results to the ones we found analytically.

The simulation of M/M/1 will give us data about the number in the system and mean sojourn time that we can compare to the ones we did analytically. We can also plot them in a graph. We will be able to observe the utilization factor also.

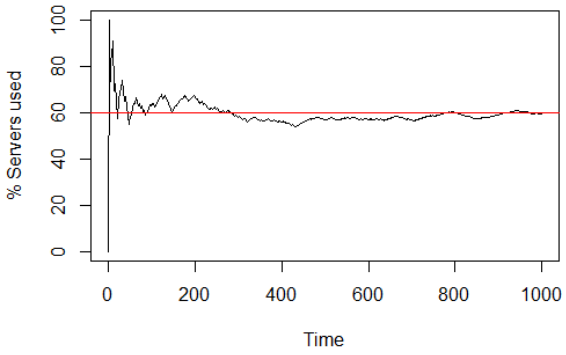
Let's start by simulating queues with $\lambda = 1/4$ and $\mu = 1/2$. That gives us a $\rho = 1/2$. We guessed that $E(N)$ should be $\frac{\rho}{1-\rho}$ so in this case $E(N) = 1$. First of all we are going to do small simulations, where we can observe that the values will not be as close as

we could expect. After that, we are going to make bigger simulations to watch that the number of the system approaches the one we expected.

Let's start with a time of 1000, the red line is the mean number in system. Using different seeds, we get:

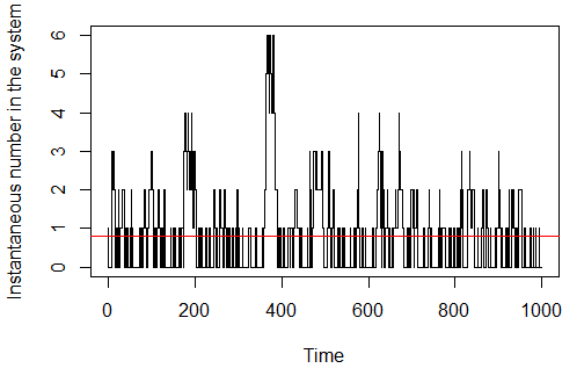


(a) Mean: $N = 1.3018$

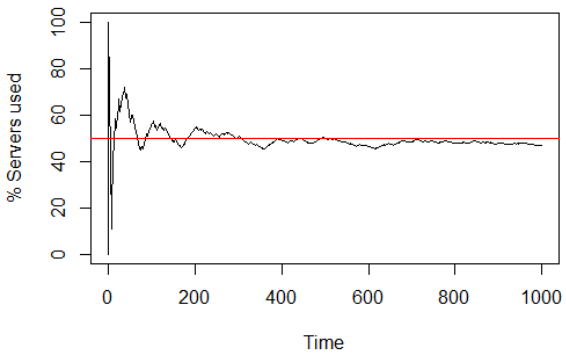


(b) Mean: U= 59.77%

Figure 5.1: Simulation 1 with time=1000, $\lambda = 1/4$ and $\mu = 1/2$



(a) Mean: $N = 0.7857$

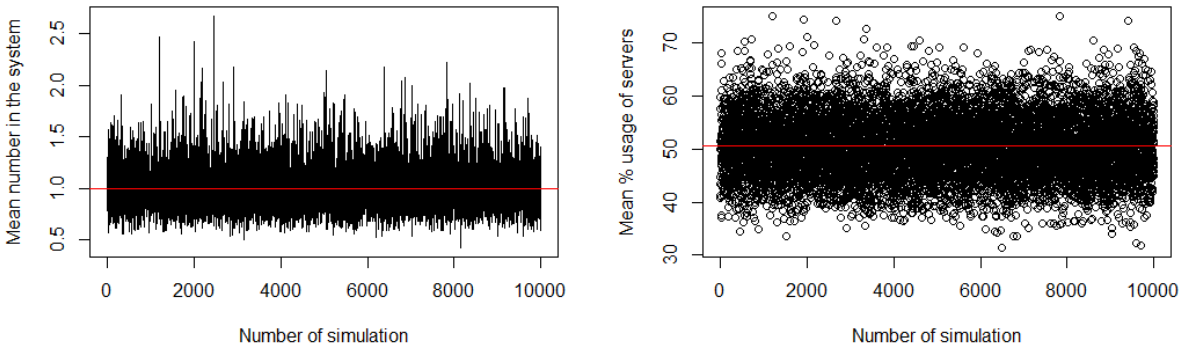


(b) Mean: U= 50.03%

Figure 5.2: Simulation 2 with time=1000, $\lambda = 1/4$ and $\mu = 1/2$

9998 more simulations were done but I won't include their plots here. Code used to simulate is included in appendix.

The mean number for all 10000 simulations is $N = 0.9960$ and the mean usage of the servers is $U = 50.55\%$. As we can see it is close enough from the $E(N) = 1$ we predicted. That is because when we have done 10000 simulations like we have done, the data is already enough to give us a reliable idea of what is happening in the system. The small amounts of errors data can contain in relation with expected values are compensated when large simulations are done. The graphics of all the simulations are shown below



(a) Mean: $N = 0.9960$

(b) Mean: U= 50.55%

Figure 5.3: 10000 Simulations with time=10^3, $\lambda = 1/4$ and $\mu = 1/2$

5.3 M/M/1/K queue

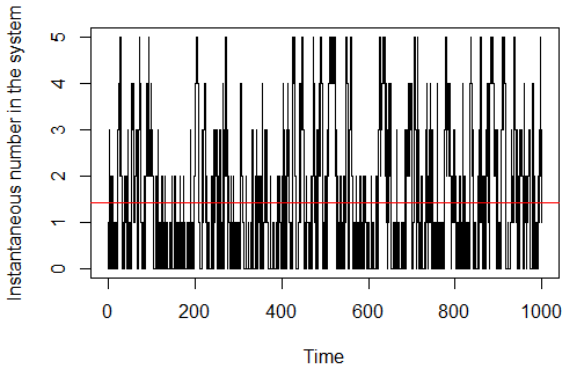
In this system, there is only one server like in the previous case but this time there is a limitation in the buffer: there can not be more than K people in the system any time. We have already computed analytically the expected number in the system:

$$N_K = \frac{\rho}{1-\rho}\frac{1-(K+1)\rho^K + K\rho^{K+1}}{1-\rho^{K+1}}$$

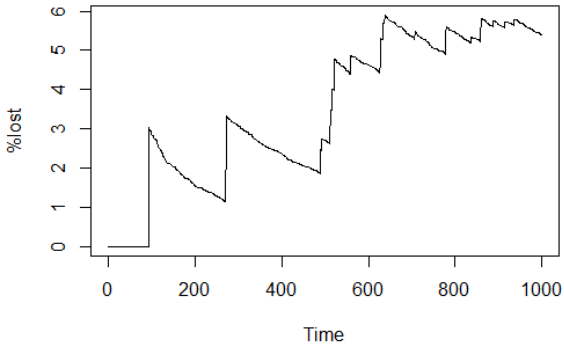
and also the probability of the system to have a loss:

$$L_K = \frac{\rho^K(1-\rho)}{1-\rho^{K+1}}$$

In this section we are going to simulate results focused in this 2 performance measures. We choose $\lambda = 3/4$ and $\mu = 1$ and $K = 5$. Substituting the above expressions with these values we got that $E(N) = 1.7001$, while $E(L_K) = 0.07217$.This are the expected values we should get analytically. We are going to make 10000 simulations again with 1000 units of time being simulated in each one. As we did before, only the first two simulations have plots for their particular performance. The code used to do simulations is in the appendix.

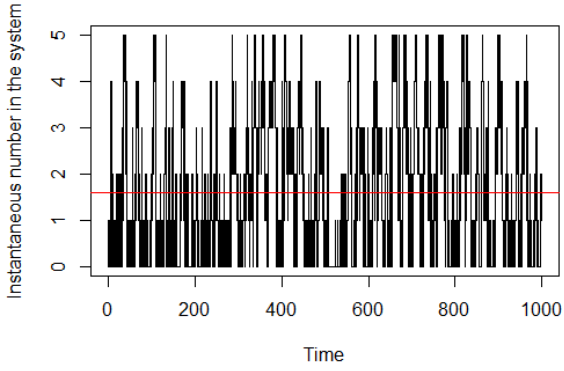


(a) Mean: $N = 1.4240$

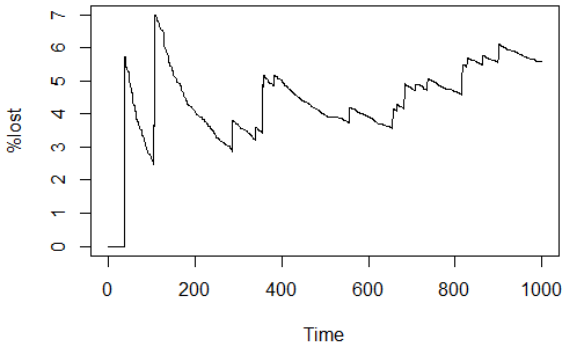


(b) $\%L_K = 5.4017\%$

Figure 5.4: Simulation 1 with time=10^3, $\lambda = 3/4$, $\mu = 1$, $K = 5$



(a) Mean: $N = 1.5899$

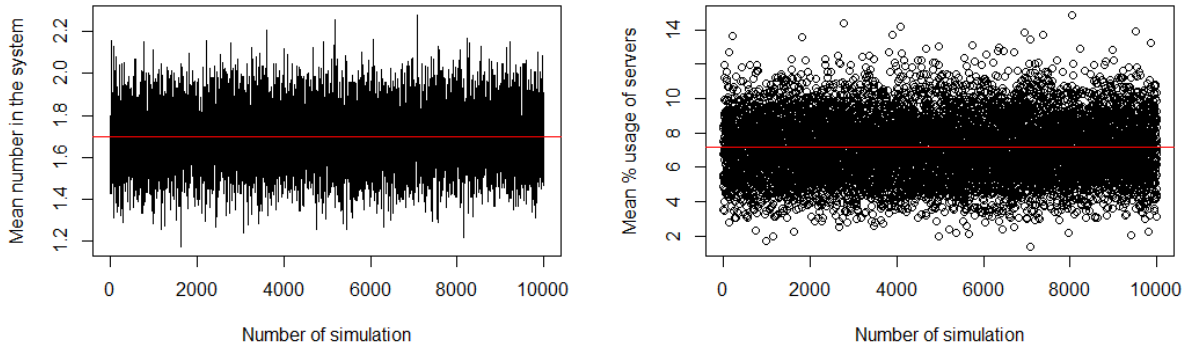


(b) $\%L_K = 5.5703\%$

Figure 5.5: Simulation 2 with time=10^3, $\lambda = 3/4$, $\mu = 1$, $K = 5$

Plots of N are not very clear, as all the lines mix when t is big.

Finally the mean for all 10^4, simulations gives us that the mean number of the systems is $N = 1.6969$, while the mean percentage lost is $\%L_K = 7.1504\%$. We can see that the number of the system is pretty accurate respect what we expected $E(N) = 1.7001$. In the other hand, percentage lost is sensibly smaller respect the one we found analytically= 7.2117%. This is because the percentage of customers lost is small in absolute terms, so a small change produces big changes in it. We could solve this by simulating bigger periods of time or by doing even more simulations, but that would require more time or a more powerful computer.



(a) Mean: $N = 1.6969$

(b) $\%L_K = 7.1504\%$

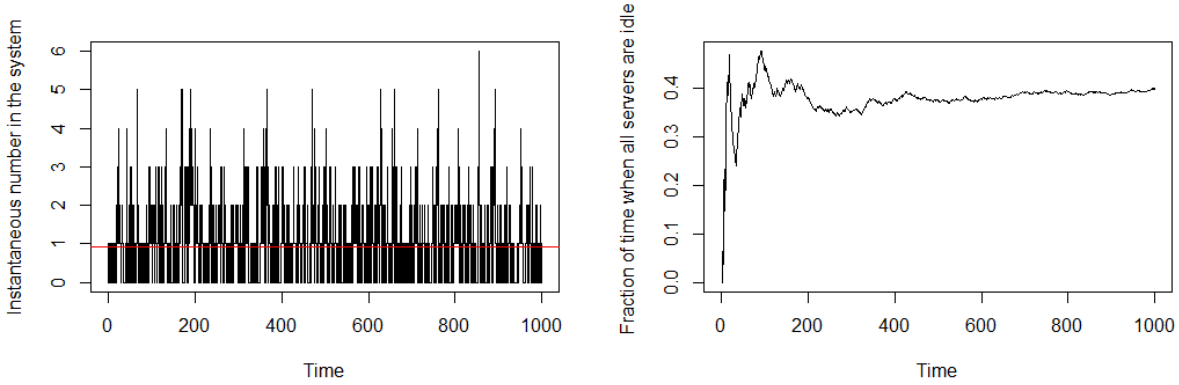
Figure 5.6: 10^4 Simulations with time=10^3, $\lambda = 3/4$, $\mu = 1$, $K = 5$

Percentage of customers lost is an important factor in real life problems. For instance, on-line servers only can handle a certain amount of petitions at once; if they get more they can either refuse them or the server can even go down. If we want that a server can handle enough petitions, we will need a buffer (K) large enough for all the waiting petitions to wait.

5.4 M/M/∞ queue

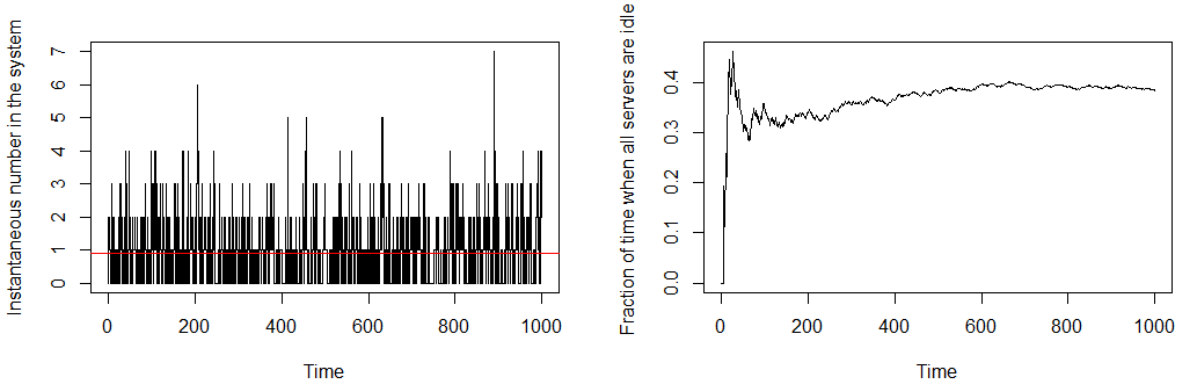
In this section we are going to model a queue that has infinite servers. This is a proper model to use in self-service systems. Analytically we have seen that the expected number in the system is $E(N) = \lambda/\mu$. We can also observe p_0 that is the proportion of time in which all the servers are idle. As we know $p_n = \dfrac{e^{-\lambda/\mu}(\lambda/\mu)^n}{n!}$, then $p_0 = e^{-\lambda/\mu} = e^{-\rho}$

To do our simulations we will choose $\lambda = 2/3$ and $\mu = 3/4$, which gives us $\rho = 8/9$. This values gives us an expected number in the system $E(N) = 8/9 = 0.8889$ and all the servers will be idle in $p_0 = e^{-8/9} = 0.4111$. Let's start to simulate doing 10^5 units of time simulations.



(a) Mean: $N = 0.9206$ (b) $p_0 = 0.3989$

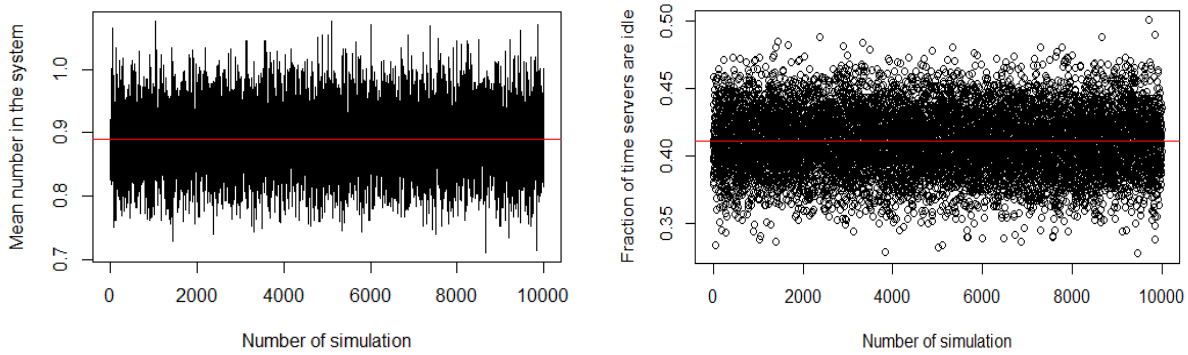
Figure 5.7: Simulation 1 with time=10^3, $\lambda = 2/3$, $\mu = 3/4$



(a) Mean: $N = 0.9142$ (b) $p_0 = 0.3845$

Figure 5.8: Simulation 2 with time=10^3, $\lambda = 2/3$, $\mu = 3/4$

All the other simulations aren't plotted but code is also in appendix. We now can plot the mean number in the queue for all the different simulations.



(a) Mean: $N = 0.8891$

(b) $p_0 = 0.4107$

Figure 5.9: 10^4 Simulations with time=10^3, $\lambda = 2/3$, $\mu = 3/4$

As we can see, the mean number in the system $N = 0.8891$ is really close to the one we had found analytically ($E(N) = 0.8889$). The fraction of idle time servers have is $p_0 = 0.4107$ which is also close to the expected $p_0 = 0.4111$. Again, if we could do more simulations, results would be even more close.

5.5 M/M/c queue

This queue has c servers, but only a queue. Customers in the queue will enter one server once it becomes idle. We have seen that analytically the expected number in the server is:

$$E(N) = c\rho + \frac{p_c \rho}{(1-\rho)^2}$$

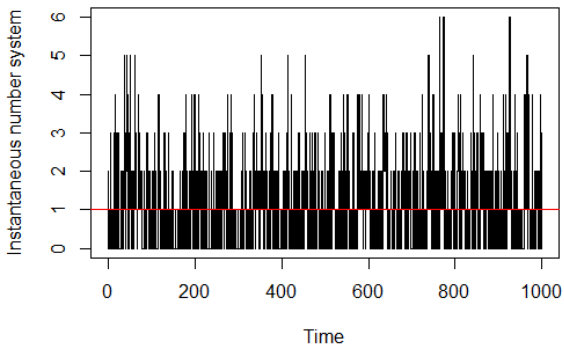
where p_c is the steady probability of being n people in the system:

$$p_c = \frac{(\lambda/\mu)^c}{c!}\left[\sum_{n=0}^{c-1}\frac{(\lambda/\mu)^n}{n!} + \frac{(\lambda/\mu)^c}{c!(1-\frac{\lambda}{c\mu})}\right]^{-1}$$

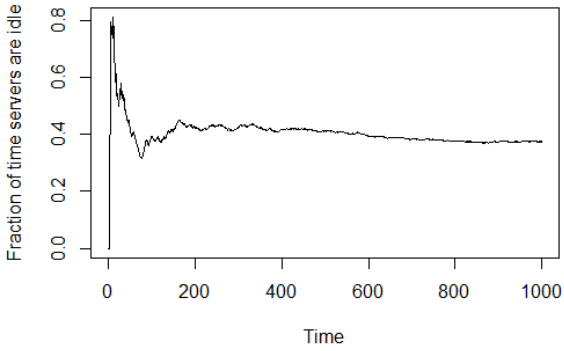
In our simulations, we are going to choose $c = 3$, and $\lambda = \mu = 1$. Substituting in the expressions above by these values, we get an expected number of the system of $E(N) = 1.0455$. If we want to know the fraction of time where all the servers are idle, we can use the expression of p_0:

$$p_0 = \left[\sum_{n=0}^{c-1}\frac{(\lambda/\mu)^n}{n!} + \frac{(\lambda/\mu)^c}{c!(1-\frac{\lambda}{c\mu})}\right]^{-1} = \frac{1}{1+1+1/2+1/4} = 4/11 = 0.3636$$

Again, we are doing 10^4 simulations of 1000 units of time each one:

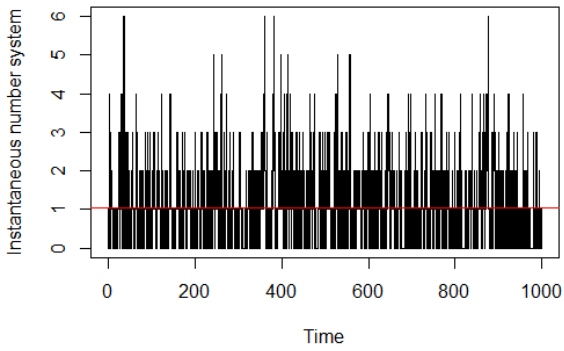


(a) Mean: $N = 1.0166$

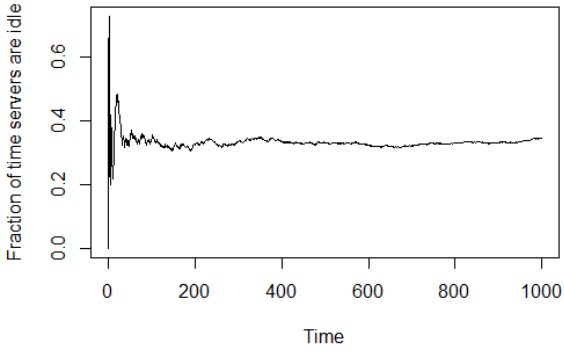


(b) $p_0 = 0.3740$

Figure 5.10: Simulation 1 with time=10^3, $\lambda = 1$, $\mu = 1$, $c = 3$



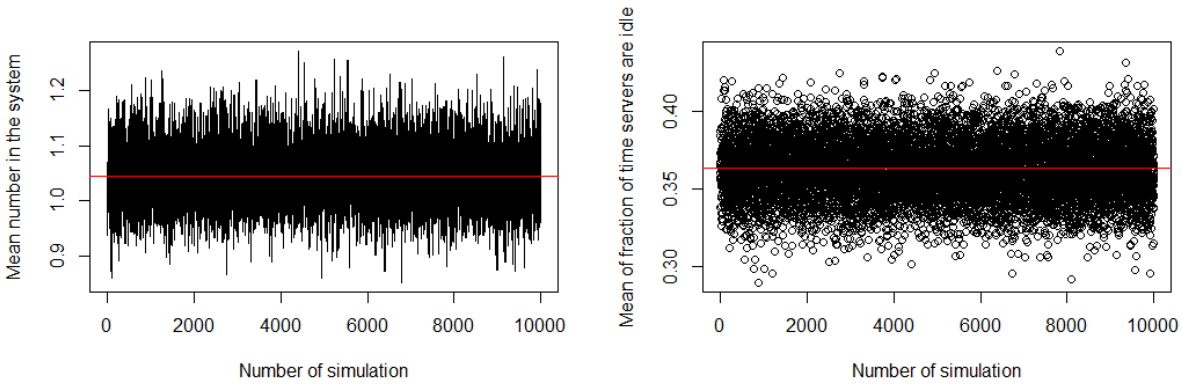
(a) Mean: $N = 1.0495$



(b) $p_0 = 0.3458$

Figure 5.11: Simulation 2 with time=10^3, $\lambda = 1$, $\mu = 1$, $c = 3$

Doing now the rest of simulations and plotting their mean into a graph we get:



(a) Mean: $N = 1.0455$

(b) $p_0 = 0.3632$

Figure 5.12: 10^4 simulations with time=10^3, $\lambda = 1$, $\mu = 1$, $c = 3$

As you can see, the mean number of people in the system is $N = 1.0455$ which is exactly the result we got analytically. Moreover $p_0 = 0.3632$ is close to the value we found analytically, which was $p_0 = 0.3636$.

Chapter 6

Conclusions

This project was aimed to work in some basics about queueing theory, as this area is not included in the study program for a Bachelors degree in Maths. In order to do that, in the first chapters we have done some theoretical approach about stochastic processes and Markov chains. About Markov chains, stochastic processes characterized for being memoryless, we have learned some properties to describe them and their different states and also some useful tools to work with them such as the transition density matrix or Chapman-Kolgomorov equations.

After that part, we got into a subclass of Markov chains: Birth/Death processes. In this type of processes transitions can only happen to a neighbouring state. Queues are a type Birth/Death processes, so we calculated probabilities of each state depending on growth and loss rates.

Once the theoretical foundations were laid, was time to enter the core of this project: queues. First of all we needed to describe in which queues we were working so we explained Kendall's notation. Afterwards, some performance measures were described. Some of them were later used in simulations. Finally, a proof of Burke's Theorem, Little's Law and PASTA property were given.

Then, with all properties we have learned, was time to start analysing queues. We started with the simplest case: M/M/1. After that, we worked with $M/M/1/K$, $M/M/\infty$ and $M/M/c$. We have found for every one of them their steady-state distribution (p_i) and their expected number in the system $E(N)$. Sojourn time was only found analytically in the simplest case, as for the other ones were beyond the objectives of this project. However, the fact that they are not found analytically can be compensated when we have done the last part of this thesis.

Finally in the last part, we have written some codes using R to simulate all the queues we previously had worked in an analytic way. To do that we have simulated 10^4 different queues for each type of queue to reduce randomness. Once we did the mean of all simulations, results were very similar to the ones we had expected to see in all the cases. Sojourn times are not included in the thesis, as we only wanted to show that simulations match with the analytic results, but the annexed codes compute them.

Bibliography

[1] Mehdi, J.: *Stochastic Models in Queueing Theory* Academic Press, Boston, 2003.

[2] Adan, Ivo and Resing, Jacques: *Queueing Systems*, Eindhoven University, Eindhoven, 2015.

[3] Sztrik, Janos: *Basic Queueing Theory* University of Debrecen Faculty of Informatics, Debrecen, 2012.

[4] Gross, D. and Harris, C.M.: *Fundamentals of queueing theory*, Wiley, Chichester, 1985.

[5] Corcuera, J.M.: *A Course in Stochastic Processes* Lecture Notes, 2016.

[6] Virtamo, J.: *Queueing Theory / Poisson Process*, Helsinki Universtiy of Technology, Helsinki, 2007.

[7] Chee-Hock NG. *Queueing Modelling Fundamentals*, John Wiley & Sons, Chichester, 2002.

[8] Palm, Francisco: *Simulacion en R*, URL: https://rpubs.com/map0logo/simED01