# RAE 555 Teletrafic theory.Course report

Salvador Moreno Carrillo StudentID:230AEB151

May 2024

# Contents

**1 Traffic and its metrics** **3**
1.1 Introduction and theory . . . 3
1.2 Problems . . . 3

**2 Queueing Systems** **5**
2.1 Definition . . . 5
2.2 Math description of the model . . . 6

**3 Analitical models** **6**
3.1 Kendall's notation . . . 6

**4 Markov Chains models** **8**
4.1 Definition . . . 8
4.2 Transition matrix . . . 8
4.3 Q matrix . . . 10

**5 Petri nets model** **10**
5.1 Definition . . . 10
5.2 Diference with Markov chains . . . 11
5.3 How they work . . . 11

**6 PRISM model checker** **11**

**7 RabbitMQ** **12**
7.1 Message broker software . . . 12
7.2 How RabbitMQ works . . . 12

# 1 Traffic and its metrics

## 1.1 Introduction and theory

### Definition of teletraffic

Teletraffic engineering is the aplication of traffic engineering to comunication networks.

**Traffic engineering** is a branch of civil engineering that uses engineering techniques to achieve the safe and efficient movement of people and goods on roadways; knowing this we can say that teletraffic studies the flow of information through a telecomunication network and its main objective is to optimise the data transmition.
All in all the objective of the teletraffic engeeniering is to deliver as much data as posible, as fast as posible,as reliable as posible and as inexpesive as posible.

### Traffic metrics

Traffic is defined by the following metrics.

•**Load** The amount of data that goes throug the system

•**Capacity** The maximum amount of data that can flow through a chanel or the whole syste.

•**Waiting time**

## 1.2 Problems

**Problem Nrº1** In the context of the image, Tele-Traffic is related to the flow and management of data or information associated with the parcel locker machine of the delivery platform company. It involves the communication and coordination aspects to ensure efficient parcel delivery.

**Problem Nº2** In this scenario, Tele-Traffic is evident in the entire process of parcel delivery. It involves the flow and exchange of information among the person at the first package locker, the parcel service truck, and the person at the second package locker. Tele-Traffic manages the communication and coordination needed for seamless parcel transfer between different locations.

**Problem Nº2** Arrival Rate ($\lambda$): The arrival rate indicates the pace at which entities enter a system. In the scenario of the parcel locker machine, an arrival occurs when a person arrives to utilize the locker, whether it's for depositing or retrieving a package.
So, in this scenario: Arrival: A person approaching the parcel locker machine to perform a delivery or retrieve a package.

Arrival Rate ($\lambda$): The frequency at which people come to use the parcel locker machine for deliveries or pickups.

**Problem Nº3** In this visual representation, the cloud at the center symbolizes a hub or distribution centre of a network.

The connected images around the cloud represent diferent serivices provided through the network.

• Phone: Provide voice and video calls in a mobile network.

• Shopping Cart: Representing e-commerce transactions and online shopping activities.

• Globe: Indicating global connectivity and international data exchange.

• Computer: Highlighting data processing and information exchange through computing devices.
•Lock: Suggesting security measures and encryption in telecommunication.

• Light Bulb: Could simbolize either an AI powered through the network or Smart house and domotics

• Web Browser Search: Representing online search and information retrieval.

**Problem Nº4** In this illustration, Tele-Traffic can be found by the connection between "Alice's PC" and "Bob's PC" where the internet is the network and the data is the traffic load.

**Problem Nº5** We may asume that the differnce in thickness can represent the difference of capacity in each Chanel.

**Problem Nº6** Here we see a Petri net representation of the previous scenario ehre

The traffic is in the entities that are connect by both, sender and responder ($P_1, P_2, P_3, P_4, P_5, P_6, P_7, P_8$).

$T_1, T_2, T_3, T_4, T_5, T_6, T_7, T_8$ are the components which work in transmission.

The connection between each component symbolizes the flow of data and interactions among different components. Traffic occurs within the pathways and interactions among these elements. Traffic could is defined by its load and it is usually measured in a

$Mem \rightarrow Gbps/MaxGbpsStorage \rightarrow GiB/MaxGiBCPU \rightarrow TFLOPS/MaxTFLOPS$

**Problem Nº7** The traffic in the HPC cluster anatomy is primarily distributed across the connections between different components. The flow of data occurs from the HPC user to the laptop, then to the login node through a secure connection. From the login node, the scheduler manages the distribution of tasks, and a dedicated network facilitates data exchange between the login node, computing nodes, and network storages.

**Problem Nr 8** Understanding the relationship between reality, model, and application is essential for effectively managing and optimizing network performance. Firstly, we have a definition of these concepts.

- Reality: it refers to the real world what is actually happening.
- Model: it is a simplified representation of reality that captures essential aspects of a telecommunications network.
- Application: refer to the practical use of models to address real-world problems.

**Problem Nº 9** In this scenario, data packets are transmitted from Alice's PC to Bob's PC via the web.

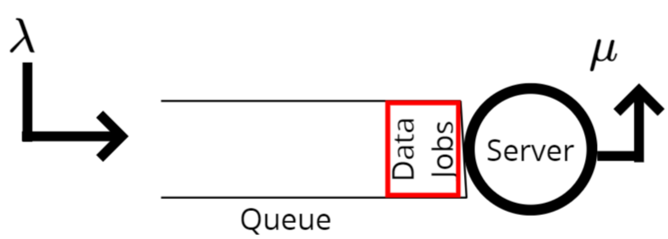
We can calculate $X(t)$ just subtracting $A(t)$ and $D(t)$. $X(t) = A(t)–D(t)$

# 2 Queueing Systems

## 2.1 Definition

Queueing theory studies the function of waiting lines for example the cash register at a shop where costumers arrive wait in line and then the cashier serves and dispatch them.

Here we can see a model of a queueing system. Where jobs/data wait to be processed or served.



### 2.2 Math description of the model

L=Load
$\lambda$=Arrival rate
$\mu$=Departure rate
W=Waiting time

$$\lambda = \frac{k}{T_{obs}} \tag{1}$$

$$\mu = \frac{k}{T_{bussy}} = \frac{1}{W} \tag{2}$$

**Little's formula**

$$L = \lambda * W \tag{3}$$

**Earlang's A formula**

$$\rho = \frac{\lambda}{\mu} \tag{4}$$

From here we can start describing the behaviour of the queues in different models.

# 3 Analitical models

The analitical model mainly developed by Agner Krarup Erlang explains the queues from a pure statistical and mathematical point of view.

*Erlang's model is mainly used in analog systems (ErlangB)or cellular networks(ErlangC). This is beyond the scope of this paper but it's recomended to learn more about these since its useful and interesting.*

### 3.1 Kendall's notation

Kendall's notation is the standard system used to describe and classify a queueing node.
Originaly it had three key elements A/S/c it was later extended to A/S/c/K/N/D.

**-A:**It indicates the kind of arrival rate it can be:
D:deterministic
M:Poisson distribution(aka Markovian)
$M^x$:Poisson process with a random variable X for the number of arrivals at one time
MMPP:Markov modulated poisson process
E:Erlang distribution

G:General or independet arrivals.(doesn't follow any distribution)

**-S:**It indicates the kind of service rate(same kinds as arrival rate).

**-c:**Number of servers

**-K:**Queue capacity

**-N:**Population(ex:nº of posible costumer)

**-D:**Queue modality:
-FiFo(First in First out):The customers are served in the order they arrived in
-LIFo(Last in First out):The customers are served in the reverse order to the order they arrived in.
-SiRo(Service in random order):The customers are served in a random order with no regard to arrival order.
-PQ(Priority queueing):There are several options: Preemptive Priority Queuing, Non Preemptive Queuing, Class Based Weighted Fair Queuing, Weighted Fair Queuing.

-PS(Processor sharing): The customers are served in the determine order with no regard of arrival order.

If only A/S/c are specified then $K = N = \infty$ and D=FIFO.
In the course we will focus in M/M/1 and M/M/s where $s\epsilon\mathbb{N}$
From the previous expresions(1-4) we can obtain the following useful formulas:

**For M/M/1**

$$P_0 = 1 - \rho \tag{5}$$

$$P_n = P_0 * \rho^n \tag{6}$$

$$L_q = \frac{\lambda^2}{\mu * (\mu - \lambda)} = \rho * L_s \tag{7}$$

$$L_s = \frac{\lambda}{\mu - \lambda} \tag{8}$$

$$W_q = L_q/\lambda \tag{9}$$

$$W_s = L_s/\lambda \tag{10}$$

**For M/M/S**

$$\rho = \frac{\lambda}{S * \mu} \tag{11}$$

$$P_0 = \frac{1}{\sum_{n=0}^{s} \frac{\frac{\lambda}{\mu}^n}{n!} + \frac{(\lambda/\mu)^s}{s! * (1-\rho)}} \tag{12}$$

If $0 \geq n \leq S$

$$P_n = P_0 * (\frac{(\frac{\lambda}{\mu})^n}{n!}) \tag{13}$$

If $n \geq S$

$$P_n = P_0 * (\frac{(\frac{\lambda}{\mu})^n}{S! * S^{n-S}}) \tag{14}$$

$$L_q = \frac{P_0 * (\frac{\lambda}{\mu})^S * \rho}{S! * (1-\rho)^2} \tag{15}$$

$$L_s = L_q + \frac{\lambda}{\mu} \tag{16}$$

$$W_q = L_q/\lambda \tag{17}$$

$$W_s = W_q + \frac{1}{\mu} \tag{18}$$

# 4 Markov Chains models

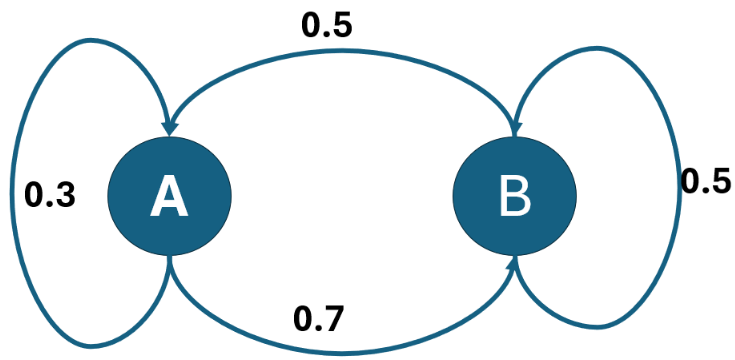
## 4.1 Definition

Markov chains are a stochastic process that helps us to understand how things might happen in sequence. There are two basic kinds of Markov chains DTMC(Discrete Time Markov Chain) and CTMC(Continuous Time Markov Chain).

**Fundamental concepts of a Markov Chain** The markov chain can be defined by:
**Number of States**:Either finite or infite.
**Periodicity**:A state is consider periodic if and only if in period $k \epsilon \mathbb{N}$ for DTMC or $k > 0$ $k \epsilon \mathbb{R}$ for CTMC con reach the same state only after k Transitions or k units of time (if all states have the same periodicity the chain is periodic with period $k$)
**Ergodic**An ergodic chain allows transitions between all states.
**Absorbing chain**:An absorbing chain has at least one state where the system can stay forever, with no probability of transitioning out.
Only ergodic chains are in the scope of this course.

## 4.2 Transition matrix

Each state has a transition probability to be reached from any other state and another one to reach any other state.




In this example the probability to transition from A to B is 0.7 and 0.3 to remain in A(transition from A to A) and 0.5 to transion from B to A and 0.5 to remain in B.
Using these we can create the transtion matrix

**Components: Rows and Columns:**The transition matrix has the same number of rows and columns, equal to the total number of states in the Markov chain. Each row represents the current state, and each column represents the possible next state.

**Probabilities as Entries:** The entries within the matrix represent the probability of transitioning from the state in the corresponding row (current state) to the state in the corresponding column (next state). For example, an entry (i, j) in the matrix shows the probability of moving from state i to state j. Properties of a Transition Matrix:

**Key characteristics: Non-negative Values:** All the entries (probabilities) in the transition matrix must be non-negative numbers between 0 and 1 (inclusive). A value of 0 indicates there's no chance of transitioning from a specific state to another. A value of 1 indicates a guaranteed transition.

**Rows Sum to 1:** The values in each row of the transition matrix must sum to 1. This ensures that the probability of transitioning from a current state to some next state is certain. Since the system must move to some state after the current state, the probabilities across all possible next states must sum up to 1.

**Example:** Using the previous chain this would be it transtion matrix:

$$P = \begin{bmatrix} 0.3 & 0.7 \\ 0.5 & 0.5 \end{bmatrix}$$

**State vector**$X(t)$ **or** $X[n]$: It indicates the probability of being in each state at a given time or step.

Tanks to the transition matrix we can obtain the probabilties of being in each state using the following expresion

$$X[n+m] = P^m * X[n] \tag{19}$$

**equilibrium state**After a number of transitions the probabilities of the state vector converge to a value called the stationary probability.
In other words $X[n+\infty] = P^{\infty} * X[n]$ as this is$X \approx P * X$ then we can asume that the probabilities of the stationary state vector are just the eigenvalues of $P$

### 4.3 Q matrix

The Q matrix, also known as the infinitesimal generator matrix or intensity matrix, plays a similar role to the transition matrix in discrete-time Markov chains. Unlike discrete-time chains where transitions happen instantaneously, continuous-time chains involve transitions that occur over infinitesimal time intervals. The Q matrix captures the instantaneous rates of transition between states.

The diagonal elements(i,i) of the Q matrix, denoted by $q_{ii}$, typically have special interpretations. They often represent the total rate at which the system leaves state i.
**Properties of the Q Matrix:**

-The sum of each row in the Q matrix must be zero. This mathematically expresses that the total rate of leaving any state must be balanced by the total rate of entering that state.

-While the Q matrix deals with instantaneous rates, transition probabilities in continuous-time Markov chains can be derived from it.
By integrating the elements of the Q matrix over a specific time interval, you can calculate the probability of transitioning between states within that time interval.

## 5 Petri nets model

### 5.1 Definition

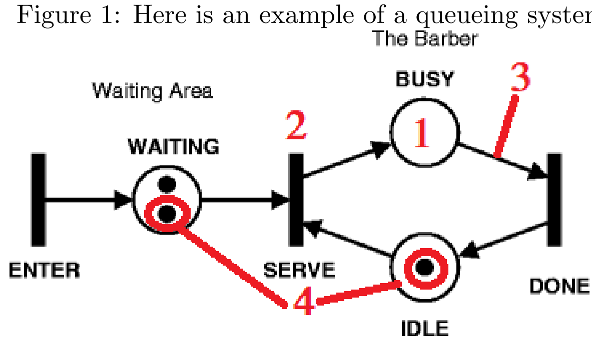
Petri nets are a graphical and mathematical modeling tool used to describe and analyze discrete event systems.
These are systems where the state changes occur abruptly at specific points in time, rather than continuously.
They are comprised of the following elements:
**Places(1):**Denote the conditions or states that the system can be in, they can hold tokens

**Transitions(2):**Transitions depict events or actions that cause the system to change from one state to another.

**Arcs(3):**Directed arcs connect places and transitions. They show the flow of tokens between places and how transitions are enabled

**Tokens(4):**Tokens indicate that a specific condition or state is active.

Figure 1: Here is an example of a queueing system rpresented with a petrinet




### 5.2 Diference with Markov chains

The main diference between petri nets and markov chains are the transitions.They represent abrupt changes in the system and not continuous like in markov chains. Other diferences are the nature of the events markov events are stochastic by nature and events in petri nets can be deterministic

### 5.3 How they work

Petri nets function in a sequence of simple steps.
Tokens initially reside in specific places, representing the starting state of the system. A transition can fire only if there are sufficient tokens present in all its input places.
When a transition fires, it consumes a token from each of its input places and deposits tokens in its output places (as specified by the outgoing arcs).The firing of transitions and the movement of tokens reflect the state changes and events occurring in the system.

## 6 PRISM model checker

A model checker is a software that let us put to the test and sometimes validate our propabilistc models. PRISM is a tool we can use for this purpose models are described using the PRISM language and then we can analise it and obtain useful measures.

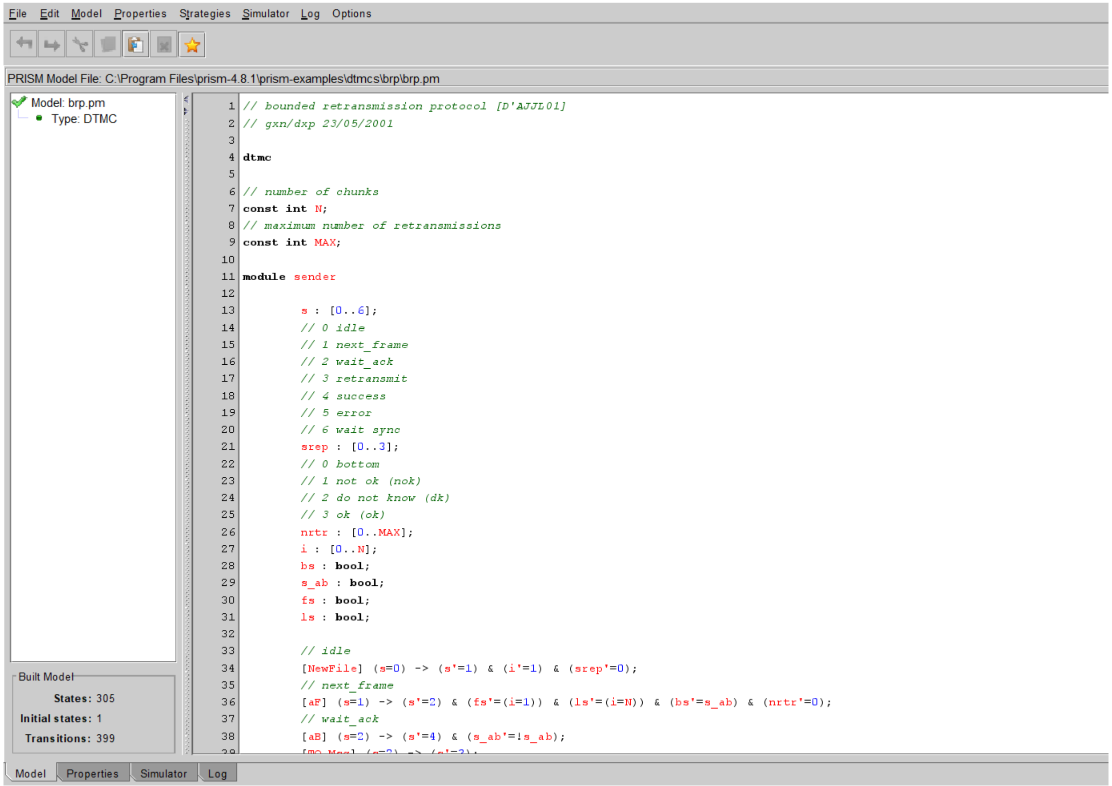



Figure 2: Snapshot of PRISM for DTMC model

# 7 RabbitMQ

RabbitMQ is an open-source message broker software.

## 7.1 Message broker software

A message broker software is like a post office for applications. It acts as a central hub that enables applications, systems, and services to communicate with each other by exchanging messages.

## 7.2 How RabbitMQ works

RabbitMQ works in 3 simple steps **sending**,**routing** and **delivering**. An application sends a message to RabbitMQ. This message can contain any kind of data, like a product update or a stock notification.RabbitMQ doesn't just deliver the message directly. It acts like a postman sorting mail, ensuring the message gets to the right recipient. It can route messages based on pre-defined rules. RabbitMQ then delivers the message to the intended recipient application. The configuration of the service has to be done in the server and client side.