The Document

João Morais

October 15, 2019

Abstract

This will be the document where I write everything I know about what I learn during my studies. I hope it serves me in the future as a "go to" document when I need to remind something.

Introduction

This document will focus on definitions, understanding concepts and transmitting intuitions/perspectives.

Each chapter will have the same structure. First the definitions and logic connections between everything with references to derivations and demonstrations that will be in attach at the end of the document. Second an overview of the derivations and how formula connect together to make a quick & easy way of finding the connections for hurry times. And third, simply one or two pages with the most important formulas from that chapter.

The first chapter will be about antennas. The second about microwaves. And the ones after that will consist of applications such as Microwave Links/ Hertzian Beams, Satellites and Radar. Maybe I'll join in a chapter about Probabilistic/Statistic Detection and Estimation of signals and one about some fundamentals of communications.

Please be aware that this is a document always at work. I'll write right here when I thing a certain chapter is closed, otherwise it might be target of modifications.

Enjoy

Contents

1 Linear Algebra

Before anything else, I strongly recommend watching a series of short videos that will give you a very graphical notion of Linear Algebra. That series of videos is "Essence of linear algebra" by 3Blue1Brown.

Here will be presented a concise analysis of Linear Algebra topics ranging from:

- various topics from one of the best resources to learn linear algebra, the MIT professor **Gilber Strang**'s lectures
- a summary of the above course can be found in here by Sho Nakagome ("A neuroengineer researching Brain Computer Interface (BCI)");
- geometric interpretations on a basic matrix, from here ;
- orthogonality, from MIT here ;
- important information for Signal Processing and various other applications.

1.1 What is a matrix?

We know what a matrix looks like, but what it is exactly?

Well, it can be seen as a group of vectors, as an equation and probably as many other things I haven't figured out yet.

1.1.1 A system of Equations

Let's start as seeing it as means of simplifying the process of solving a set of equations. Imagine that we have the following equations:

$$\begin{cases} 3x + 5y = 11 \\ x + 4y = 6 \end{cases}$$

These can be written in matrix from like this:

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 11 \\ 6 \end{bmatrix}$$

And solvable through the Row Echelon Form (REF):

$$\left[\begin{array}{cc|c} 3 & 5 & 11 \\ 1 & 4 & 6 \end{array}\right] = \left[\begin{array}{cc|c} 1 & 4 & 6 \\ 0 & -7 & -7 \end{array}\right] = \left[\begin{array}{cc|c} 1 & 4 & 6 \\ 0 & 7 & 7 \end{array}\right] \tag{1}$$

Is important to note that to obtain the REF, only elementary equation operations can be used. These are the sum/subtraction of two equations, the multiplication of a equation by a scalar different from 0 and change the place of the equations. In the particular case of (1) the operations that took place were: row1 - 3*row2, switch rows and multiply the last row by -1.

From the our new form, is trivial to go back to the equation formula and directly attribute values to x and y. However, a few calculations are still required. The absolute best way is using the Reduced Row Echelon Form (RREF). This consists of using the rows with less elements to "cut out" elements from the other rows, leading to a very simplified matrix (see (2))

$$\left[\begin{array}{cc|c} 1 & 4 & 6 \\ 0 & 7 & 7 \end{array}\right] = \left[\begin{array}{cc|c} 1 & 4 & 6 \\ 0 & 1 & 1 \end{array}\right] = \left[\begin{array}{cc|c} 1 & 0 & 2 \\ 0 & 1 & 1 \end{array}\right] \tag{2}$$

Note that this RREF is unique and characterised by 1's in the diagonal and in stairs, i.e the only non-zero entry in that column starting with the leftmost column. In this form, all results can be obtained immediately.

However, putting a matrix in this form isn't always possible. For instance, is impossible to do so if two rows are **linearly dependent** meaning that is possible to multiply a constant to one to obtain the other - seeing rows as equations, linear dependency can be thought of an equation that adds absolutely no other constraint to the solution set or, in other words, does the exact same as one equation that already exists thus can be discarded hence ending with more unknowns than equations.

Before further analysing the concept of linear dependency and all that derives from that, is pertinent to have yet another look at the matrix. A geometric interpretation is often useful.

1.1.2 Vectors in Space

Exactly, have you thought your matrix could be vectors in space?

Well, maybe this decomposition helps to thing about it:

$$\begin{bmatrix} 3 & 5 \\ 1 & 4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 11 \\ 6 \end{bmatrix}$$

$$\downarrow$$

$$x \begin{pmatrix} 3 \\ 1 \end{pmatrix} + y \begin{pmatrix} 5 \\ 4 \end{pmatrix} = \begin{pmatrix} 11 \\ 6 \end{pmatrix}$$

Thus we can imagine the vectors (5,4) and (3,1) is space. And one solution to obtain the vector (11, 6) is by multiplying 2 by the first vector and 1 by the second.

Note that we broke the matrix in half and gave it a meaning: "What do you have to multiply to this vector and to this vector to obtain the vector you want?" Where the vectors were the columns of the matrix. And the solution were the x and the y that were going to scale those vectors.

One can also think, despite being a bit more complicated, that what we want is to write (11, 6) in the basis of those two vectors (3,1) and (5,4). This is how the matrix represents a transformation. We are transforming the "normal" into the space defined by those two vectors. Further note that that space is given by the column vectors of the matrix! This will be important in the future.

Let's have a look at basis and spaces.

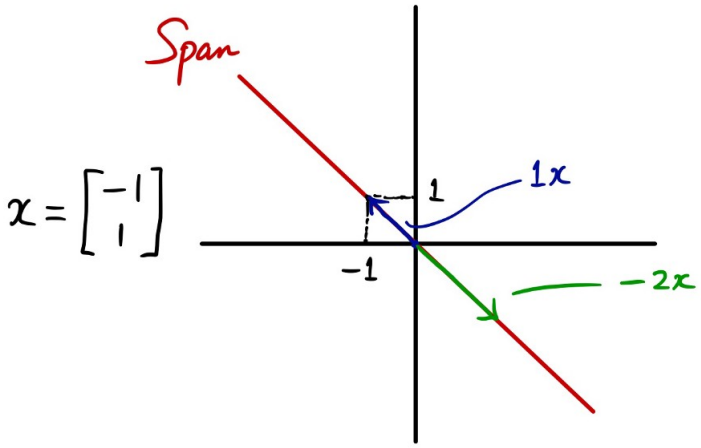
1.2 Basis, Spaces and Subspaces

A space is nothing more than a set of vectors that can be obtained from a linear combination from vectors in the basis.

If we want to take a subset of a space that is also a space we call that a subspace.

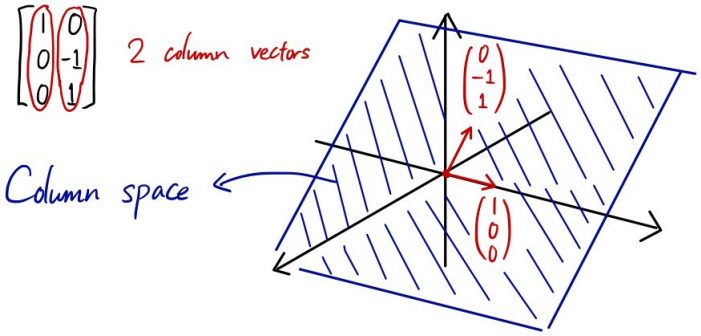
About basis: these are the vectors which when linearly combined generate a certain space. Note that one space can have infinite many basis. Actually, any set of N linearly independent vectors from a space with dimension N is a basis of that space. A set of vectors is only considered a basis if it spans a space (all linear combinations of its vectors create a space) and the vectors in it are linearly independent.

A good image to describe what "span" is:



1.2.1 Column Space or Range

The range of A (ran(A)) is the space spanned by the columns of A.



As expected, the row space is nothing more than the exact same thing for the rows.

1.2.2 Row Nullspace or Kernel

The kernel of A (ker(A)) is the space created by the vectors that when A is applied to them the result is 0.

Mathematically, the nullspace of a is nothing more than the vectors x that lead to $Ax = 0$.

Note that that thinking from before as "what vectors should be multiplied to get a certain solution" will now have a problem: there can be more than one answer to the null result. This only happens if there is a problem with the transformation. If the transformation is well done (if the matrix has full rank), then the column subspace will be the full space and anything multiplied to the matrix will have one and only one solution. If the problem/matrix is ill-conditioned (the rank is less than the dimension of the vectors of the columns), then there can be more than one solution to a give problem.

————From an answer in math.stackexchange.com: "Let's suppose that the matrix A represents a physical system. As an example, let's assume our system is a rocket, and A is a matrix representing the directions we can go based on our thrusters. So what do the null space and the column space represent?

Well let's suppose we have a direction that we're interested in. Is it in our column space? If so, then we can move in that direction. The column space is the set of directions that we can achieve based on our thrusters. Let's suppose that we have three thrusters equally spaced around our rocket. If they're all perfectly functional then we can move in any direction. In this case our column space is the entire range. But what happens when a thruster breaks? Now we've only got two thrusters. Our linear system will have changed (the matrix A will be different), and our column space will be reduced.

What's the null space? The null space are the set of thruster instructions that completely waste fuel. They're the set of instructions where our thrusters will thrust, but the direction will not be changed at all."

If you are someone with some linear algebra background you might have spotted some kind of connection between the null space and the column space. So it is worth to talk now about the rank and the relation these spaces.

1.3 Rank and Spaces relationships

The rank of the matrix is nothing more than the number of pivots in the Reduced Row Echelon Form.

Seeing a matrix as a set of equations with N variables, a full rank matrix has rank N meaning that is possible to discover the N variables without ambiguity. If the matrix is rank deficient or ill-ranked, that means we have too many unknowns for our equations.

Is common sense that if a matrix has more columns than rows, it won't have full rank as this is exactly the same as saying it will have more unknowns than equations.

In terms of the relations between the spaces: was it evident so far that the less precise is the matrix (the less rank it has compared to full rank), the more it will nullify vectors, in other words, take away their identity and the more vectors would the space that gets to zero have.

Therefore, the sum of the dimensions of the column span and nullsubspace should equal the rank that the matrix needs to be full rank, i.e the number of unknowns or simply the number of columns.

(there are a few more subspaces and we still need to explain the question that is asked every time.)

2 Antennas

Permeabilidade (magnética) do vácuo

$$\mu_0 = 4\pi \times 10^{-7} \ \text{H.m}^{-1}$$

Permitividade (eléctrica) do vácuo

$$\varepsilon_0 = \frac{10^{-9}}{36\pi} \ \text{F.m}^{-1}$$

Velocidade da luz no vácuo

$$c_0 = \frac{1}{\sqrt{\mu_0 \, \varepsilon_0}} = 3 \times 10^8 \ \text{m.s}^{-1}$$

The electric permittivity is what relates the Electric Displacement Vector (D) and the Electric Field (E). For a deeper meaning one has to look into the constitutive relations as well. Likewise, the magnetic permeability relates the Magnetic Induction Field (B) with the Magnetic Field (H).

As equações de Maxwell, as relações constitutivas e a força de Lorentz

$$\mathbf{F} = q(\mathbf{E} + \mathbf{v} \times \mathbf{B})$$

mostram que os campos fundamentais (básicos) são E e B . Excepto para o vácuo, D e H dependem de um modelo do meio.

$$\mathbf{D} = \varepsilon \mathbf{E} = \varepsilon_0 \mathbf{E} + \mathbf{P} \qquad \mathbf{P} = \varepsilon_0 \chi_e \mathbf{E} \qquad \varepsilon = \varepsilon_0 \varepsilon_r \qquad \varepsilon_r = 1 + \chi_e$$

$$\mathbf{B} = \mu \mathbf{H} = \mu_0 (\mathbf{H} + \mathbf{M}) \qquad \mathbf{M} = \chi_m \mathbf{H} \qquad \mu = \mu_0 \mu_r \qquad \mu_r = 1 + \chi_m$$

P – (vector) Polarização eléctrica — χ_e – susceptibilidade eléctrica

M – (vector) Magnetização — χ_m – susceptibilidade magnética

The Electric Polarization and Magnetization vectors show how the material changes when to being subjected to a Electric/Magnetic field. This change can be displacement of charges according due to the application of the Electric field or similar in relation to the magnetic induction field. The higher the permittivity, the most charges move with the application of the field. The higher the permeability, the higher the internal magnetic field.

3 Radio Wave Propagation

It is absolutely pivotal some nomenclature in order to understand each other.

- Vector $\mathbf{E}$ ou $\underset{\sim}{\mathrm{E}}$
- Versor $\hat{\mathbf{x}}$ ou $\underset{\sim}{\hat{\mathrm{x}}}$
- Tensor $\underset{\approx}{\mathrm{E}}$
- Amplitude Complexa $\overline{\mathrm{E}}$ ($\overline{\mathbf{E}}$ ou $\underset{\sim}{\overline{\mathrm{E}}}$)
- Produto interno $\mathbf{A} \cdot \mathbf{B}$
- Produto externo $\mathbf{A} \times \mathbf{B}$
- Operadores Diferenciais (*Nabla* $\nabla = \hat{\mathbf{x}}\frac{\partial}{\partial x} + \hat{\mathbf{y}}\frac{\partial}{\partial y} + \hat{\mathbf{z}}\frac{\partial}{\partial z}$)
  - Gradiente $\nabla \mathrm{A}$
  - Divergência $\nabla \cdot \mathbf{A}$
  - Rotacional $\nabla \times \mathbf{A}$
  - Laplaciano $\nabla^2 \mathbf{A}$

Mainly, everything that is bold is a vector and everything that has a bar on top is a complex amplitude. With these definitions out of the way, we can start:

A wave propagating across the z direction has the following (mathematical) shape:

$$e(z,t) = E_o cos(wt - kz)$$

We can only talk about waves when the field disturbance changes in time and propagates in space, thus the dependence with a spatial coordinate and time. Note further that we start off with a very simple wave: is sinusoidal, doesn't have attenuation, only propagates in one direction and only oscillates in one direction.

To write a (slightly) more general formula for a wave, while taking advantage of the complex notation that assumes already a sinusoidal wave - note that the real part of a exponential is a cossine -, we get:

$$e(z,t) = Re\{Ee^{jwt}\}, E = E_o e^{-jkz}$$
$$e(r,t) = Ecos(wt - \boldsymbol{k} \cdot \boldsymbol{r})$$

Something important to keep in mind:

In free space, electric and magnetic fields are mutually orthogonal and orthogonal to the propagation direction.

Note, however, that this is not true for propagation in matter that is anisotropic or when the waves are contained in a waveguide such like a metallic waveguide where there are TE/TM modes. But we won't consider those cases, at least for now.

Also, the vector $\boldsymbol{k} = k_x\hat{x} + k_y\hat{y} + k_z\hat{z}$ represents the direction of propagation because each of its components will contribute a phase to the oscillation of the wave.

Further note that the above wave equation is for a plane wave. Fortunately in the far-field all waves are plane waves or can be obtained from them therefore that expression will pop-up fairly often. However, most waves are spherical waves that are no more than plane waves that decay with the radius to the source, therefore the surfaces of equal amplitude are spheres.

$$\frac{A}{r}e^{j(wt-kr)}$$

The more general formula for the complex amplitude is:

$$\bar{\mathbf{E}}(\mathbf{r}) = \mathbf{E_0}\, e^{-j\mathbf{k}\cdot\mathbf{r}} = (E_{0x}\hat{\mathbf{x}} + E_{0y}\hat{\mathbf{y}} + E_{0z}\hat{\mathbf{z}})\, e^{-j(k_x x + k_y y + k_z z)}$$

$$k^2 - \omega^2 \mu_0\, \varepsilon_0 = 0 \qquad \textbf{Equação de dispersão}$$

$$\mathbf{k} = k\hat{\mathbf{n}} \qquad k = \sqrt{k_x^2 + k_y^2 + k_z^2} = \frac{\omega}{c_0}$$

$$\nabla = \frac{\partial}{\partial x}\hat{\mathbf{x}} + \frac{\partial}{\partial y}\hat{\mathbf{y}} + \frac{\partial}{\partial z}\hat{\mathbf{z}} = -j\mathbf{k} = -jk\hat{\mathbf{n}}$$

The last part is important because the expression for the free space impedance Z_0 comes directly from the Maxwell equations and it is easier to solve them in the frequency domain (with complex amplitudes/phasors).

$$\nabla \times \mathbf{H} = \mathbf{J} + \frac{\partial \mathbf{D}}{\partial t}$$ → Corrente de Deslocamento — Lei de Ampere

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}$$ — Lei de Faraday

$$\nabla \cdot \mathbf{D} = \rho$$ — Lei de Gauss

$$\nabla \cdot \mathbf{B} = 0$$

E – Campo Eléctrico [V.m^{-1}]

H – Campo Magnético [A.m^{-1}]

D – Deslocamento Eléctrico [C.m^{-2}]

B – Indução Magnética [T = Wb.m^{-2}]

J – Densidade de Corrente [A.m^{-2}]

ρ – Densidade de Carga [C.m^{-3}]

Resulting in the expression below, from where we can prove the previous statement of orthogonality between fields and direction of propagation.

$$-jk\hat{\mathbf{n}} \times \bar{\mathbf{H}} = j\omega\varepsilon_0 \bar{\mathbf{E}}$$

$$-jk\hat{\mathbf{n}} \times \bar{\mathbf{E}} = -j\omega\mu_0 \bar{\mathbf{H}}$$

$$-jk\hat{\mathbf{n}} \cdot \bar{\mathbf{D}} = 0$$

$$-jk\hat{\mathbf{n}} \cdot \bar{\mathbf{B}} = 0$$

$$\hat{\mathbf{n}} \perp \bar{\mathbf{E}} \perp \bar{\mathbf{H}}$$

$$\bar{\mathbf{E}} = -Z_0(\hat{\mathbf{n}} \times \bar{\mathbf{H}}) \qquad Z_0 = \sqrt{\frac{\mu_0}{\varepsilon_0}} = 120\pi \text{ Ohm}$$

$$\bar{\mathbf{H}} = \frac{1}{Z_0}(\hat{\mathbf{n}} \times \bar{\mathbf{E}})$$

Then many expressions come from the Maxwell equations when we consider losses. Because the actual Dispersion Equation is the following:

$$\gamma^2 - j\omega\mu(\sigma + j\omega\varepsilon) = 0 \qquad \textbf{Equação de dispersão}$$

$$jk = \gamma = \alpha + j\beta = \sqrt{j\omega\mu\,(\sigma + j\omega\varepsilon)} = j\omega\sqrt{\mu\varepsilon}\sqrt{1 - j\frac{\sigma}{\omega\varepsilon}}$$

$$Z = \sqrt{\frac{j\omega\mu}{\sigma + j\omega\varepsilon}} = \sqrt{\frac{\mu}{\varepsilon(1 - j\frac{\sigma}{\omega\varepsilon})}} = \sqrt{\frac{\mu(1 + j\frac{\sigma}{\omega\varepsilon})}{\varepsilon\left[1 + \left(\frac{\sigma}{\omega\varepsilon}\right)^2\right]}}$$

Tangente do ângulo de perdas

$\tan\theta = \sigma/\omega\varepsilon$

Note that the amplitudes of the field for each component can NOT be a positive number. They can be a complex number in the sense that they will influence the polarization of the field. We'll see that in the next section!

In summary:

$$Z=\sqrt{\frac{j\omega\mu}{\sigma+j\omega\varepsilon}} \qquad \textbf{Resumo} \qquad \beta=\frac{2\pi}{\lambda}$$

$$\gamma=\alpha+j\beta=\sqrt{j\omega\mu\,(\sigma+j\omega\varepsilon)} \qquad Z_0=\sqrt{\frac{\mu_0}{\varepsilon_0}}=120\pi\ \Omega \qquad c_0=\frac{1}{\sqrt{\mu_0\,\varepsilon_0}}=3\times10^8\ m\cdot s^{-1}$$

Meio	Z [Ω]	α [Np.m^{-1}]	β [rad.m^{-1}]
Dieléctrico Perfeito $\sigma=0$	$\sqrt{\frac{\mu}{\varepsilon}}=Z_0\sqrt{\frac{\mu_r}{\varepsilon_r}}$	0	$\frac{\omega}{c_0}\sqrt{\mu_r\,\varepsilon_r}$
Bom Dieléctrico $\tan\theta=\frac{\sigma}{\omega\varepsilon}<<1$	$\sqrt{\frac{\mu}{\varepsilon}}(1+j\frac{\sigma}{2\omega\varepsilon})$	$\frac{\sigma}{2}\sqrt{\frac{\mu}{\varepsilon}}$	
Bom Condutor $\tan\theta=\frac{\sigma}{\omega\varepsilon}>>1$	$\sqrt{\frac{\omega\mu}{2\sigma}}(1+j)$	$\sqrt{\frac{\omega\mu\sigma}{2}}=\frac{1}{\delta}$	

3.1 Polarization

The polarization is the shape the field oscillation describes while looking at the wave from propagation direction with the wave going away from us . An easier way of finding if it is rotating to the left or to the right, we can use the right-hand, with the thumb along the propagation. If it is rotating along the way our (right-)hand closes, then it is a right circular polarisation.

The reference for horizontal polarisation is the earth.

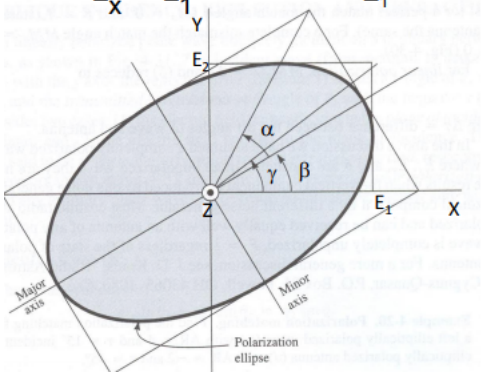
This derivation of The general Elliptic polarization shows why plane waves propagating freely can have its polarization described generally by an ellipse. Also shows that certain specific parameters of that ellipse can lead to certain more familiar polarizations, such as linear and circular.

$$\left(\frac{E_x}{E_1}\right)^2+\left(\frac{E_y}{E_2}\right)^2-2\left(\frac{E_x}{E_1}\right)\left(\frac{E_y}{E_2}\right)\cos\Phi=\text{sen}^2\Phi$$

Equação da curva (elipse) de polarização

$$p=\frac{\overline{E}_y}{\overline{E}_x}=\frac{E_2}{E_1}e^{j\Phi} \qquad \tan\gamma=\frac{E_2}{E_1}$$

p é a razão de polarização



Estado de Polarização (α,β) ou (γ,Φ)

$$\cos 2\gamma=\cos 2\alpha\cos 2\beta$$
$$\tan\Phi=\frac{\tan 2\alpha}{\text{sen}\,2\beta}$$

$$\text{sen}\,2\alpha=\text{sen}\,2\gamma\,\text{sen}\,\Phi$$
$$\tan 2\beta=\tan 2\gamma\cos\Phi$$

3.2 Reflection

All reflections' chapter is according to Snell Laws.

A few bullet points to take away from the first class on reflection are:

- a dry soil behaves as a metal for a very small angle with the ground, typical in very long distances. "Behaving as a metal" means that the reflection coefficient will be -1, for either polarisation;
- Brewster's Angle is of incidence for which the parallel component of the field is passes completely to the other side of the surface. The reflected wave does a 90° angle with the refracted wave and only has H

field. Note that this only happens for parallel polarisation. In perpendicular is the H field that will be suppressed on reflection. The perpendicular component is never fully absorbed.

- Horizontal polarisation always has bigger reflection coefficients therefore is worst to use because it will cause more variability on the arrival. Other word for incidences with very short angles with the ground is "grazing incidences".
- Dependences with frequency increase with dielectric conductivity. Else the losses will simply be too small.

One thing that needs to be introduced are the Fresnel coefficients:

$$\Gamma_H = \frac{\sin\psi - \sqrt{n^2 - \cos^2\psi}}{\sin\psi + \sqrt{n^2 - \cos^2\psi}}$$

$$\Gamma_V = \frac{n^2\sin\psi - \sqrt{n^2 - \cos^2\psi}}{n^2\sin\psi + \sqrt{n^2 - \cos^2\psi}}$$

$$n^2 = \frac{\varepsilon}{\varepsilon_0}\left(1 - j\frac{\sigma}{\omega\varepsilon}\right)$$

Note further that the last bullet point mentioned the tangent of the loss angle.

$$\tan\delta = \frac{\sigma}{\omega\epsilon}$$

Note that if the conductivity is too small, the whole expression will be controlled by that and the frequencies would have to be very small as well to make a difference. Since we don't use those frequencies in Radio Propagation, we may say that only for a sufficiently high conductivity, the frequency dependence increases.

3.3 Spherical Earth

Surprise: the Earth is not flat, it's spherical. And this curvature needs to be accounted specially because even the direct ray may be influenced by this curvature.

4 Telecommunication Networks - Overview

From the courses Internet Networks and Services (RSI in portuguese) and Telecommunication Networks (RTel in pt) I've had an insight about how the whole network is multiplexed into optic fibers and many other interesting topics such as the triple play services and a bunch of protocols that are used in today's world to make everything communicate with everything. Therefore, I propose to write a sum up of the slides and bibliography of RSI and RTel in this section. I'll mainly give importance to RTel since it is what I'm studying at the moment, but I hope to go through the slides of RSI as well.

1. Introduction
   (2 lessons)
2. Fundamentals of networks
   (7 lessons)
3. Ethernet and data centre networks
   (5 lessons)
4. SDH transport networks
   (4 lessons)
5. Optical transport networks
   (4 lessons)
6. Access networks
   (3 lessons)

1. The Internet
2. Quality of Service on the Internet
3. IP Network Models
4. Next Generation Networks
5. The Telephony Network
6. Technologies for data transport
7. MPLS - Multi-Protocol Label Switching

4.1 Introduction

Definition *Telecommunications* : is the transmission of information at a distance through the use of electromagnetic signals.

Definition *Telecom. Network* : collection of nodes and links with the purpose of interchanging these signals in order to have an information flow.

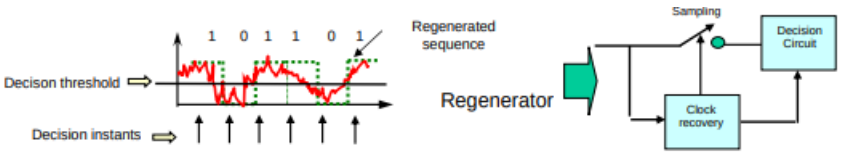
These Telecommunication Networks can be public, owned by Telecom. / Network Operators that use that network to provide services to the general public, or can be private, used by a company to connect infrastructures. Many of these private networks also rely on leased links by public networks.

There are mainly 3 layers in a network: the backbone or core, the metropolitan and the access layer. As expected, the access layer collects the traffic, connections to homes, offices, everywhere the internet is required. The metropolitan area connects different parts of the city that use that network, typically with a ring (made out of optic fiber). The core is the most extensive layer, with a mesh of nodes and very extensive links that connect cities of the whole world. Hundreds or thousands of km's is not atypical.

What makes possible to communicate with everyone connected to the internet is that the networks of both operators are also connected.

As a public service, the public networks must provide fidelity (transmit the information without loss of changes) and reliability (less than 3 minutes down per year).

Nowadays, most of transmission is digital. A series of pulses is transmitted through a channel with attenuation, dispersion, interference from other signals and noise. Therefore what reaches the other side is considerably different and has to be estimated what the original input was.



As having a dedicated physical infrastructure for each service would be far to expensive and messy, the big majority of services share the same channel, the optic fiber. It is easily shared because of the available bandwidth in it. Thus, the signals are multiplexed at the entrance, using different wavelengths (**Wavelength-Division**

Multiplexing (WDM)) and de-multiplexed at the other end, to follow each one to their device that is requiring the service.

Note that WDM is exactly like **Frequency Division Multiplexing (FDM)** but in the optical domain. Technically they are exactly the same as changing the wavelength is nothing more than changing the transmit frequency.

A single mode optical fiber can reach throughputs of 10 Terabits per second.

Remember that there are many ways of scheduling frames in a multiplexer. With time slots or doing it statistically are two ways. Also, there are 2 types of switching: packet switching and circuit switching. Circuit is when a channel is constantly reserved for a certain application even if it is not being used. Packet switching allows a much better share of the resources. Packet switching principle is based on sending packets whenever there's a packet to transmit and use all the resources to do so as fast as possible. Therefore, the "speed" of the internet depends a lot on the amount of people that are accessing it.

Regarding the physical infrastructures for the transmission of data, those go from satellites, well the open space in general as microwave links are also a thing, twisted pairs, optical fibers and even a few more that are less common.

Finally, a look at the tendencies is pertinent. The traffic is increasing constantly, at a rate of 30% a year, therefore the network must be upgraded as the time passes as well, or else it won't be able to handle the traffic of the future. Not only are the links being upgraded since now we have fiber to the home, terminating really in our router, but each node must be upgraded as well to cope with the traffic resulting in new switches, larger datacentres, ect... However, all of this must be standardised to guarantee compatibility between countries, operators, manufacturers and users and to ensure minimum quality of service for all users. This standardisation is done by the International Telecommunication Union (ITU) that has two main sectors of interest: the -T sector regarding telecommunications in general and the -R sector for radiocommunications that is more focused in point-to-point, mobile, satellite links, ect... Additionally, ETSI, ISO, OSI, ANSI, IEEE are some of the main organisations that standardise technologies. IEEE is the best :)

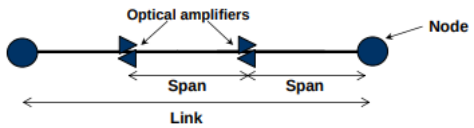
4.2 Networks Fundamentals

A network is composed of nodes and links and can be represented by graphs. However, a clear distinction between networks and graphs has been made in class: a network is a graph with a few more numbers that represent various network parameters. These parameters will be talked later, but can be delay of a link, distance, ...

The physical topology concerns the physical connections that are in place while the logical topology for a certain case concerns the actual flow of information. Even though every computer is connected in a network, maybe the information always flows from one to the others and the graph that represents that has much less links.

A link can be unidirectional or bidirectional. If the link is unidirectional, sometimes is referred to as an arc. $e_1 = (v_1, v_2)$ is the representation of a link, and the order of the nodes matter if it's an arc.

In optical fiber networks, or other networks that require amplifiers, the space between amplifiers (distance the signal has to go attenuating) is called a span.

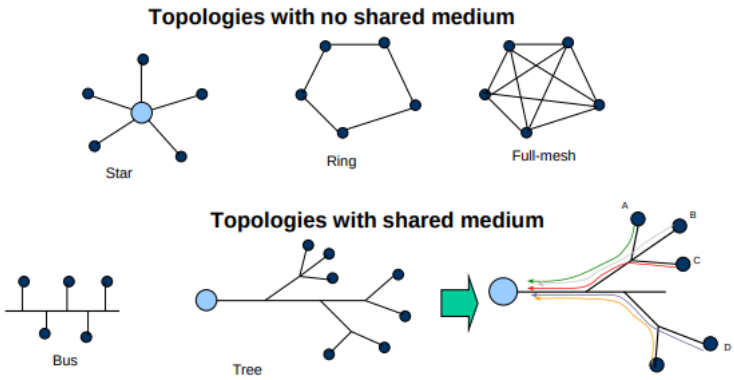


In a graph there's N number of Vertices(v_i), and L number of Edges(e_j). And the degree of the vertex is the number of edges it has. It's called the order of the graph, it's number of vertices, and the size of the graph it's number of edges.

Directed graphs only have unidirectional links, while undirected graphs only have bidirectional links.

The reason to make the distinction between directed and undirected graphs (unidirectional and bidirectional edges): in case of optical fiber, which is what connects most of long distance networking, is required to use more than one fiber. Because and optical emitter can't receive as well (at least in the same fiber). Also, if amplifiers are required, note that they are directed as well.

A path can be represented by a set of links, starting at some node. Source and sink are the names for the first and last vertex of that path.



Bus, Ring and Star are the main physical topologies. Tree as well.

A tree is simply a graph with no cycles.

A graph can be represented with an **Adjacency matrix (A)** , with $a_{ij} = 1$ if there's a direction from the vertices i to j.

The average node degree is given by the average of each node's degree which won't be more than the sum of all links, times 2 divided by the number of nodes. Times 2 because each link contributes for the degree twice, once at each end.

$$\delta_i = \sum_{j=1}^{N} a_{ij}$$

The average node degree is given by

$$<\delta>=\frac{1}{N}\sum_{i=1}^{N}\delta_i=\frac{2L}{N}$$

The Network diameter (D_R) is the maximum number of links between nodes through the shortest path between them. D_R is the longest of the shortest paths between every node.

Every link can have an associated cost (a function of distance, delay, reliability, actual cost, or other parameters) and a capacity (u_e denotes the capacity of node e).

The link capacity can be measured in any traffic unit that is appropriate for the problem. In packet networks: bit/s; in SDH networks: STM-N; in OTN networks: OTU-k; in WDM networks: number of wavelengths

Despite similar to an Adjacency Matrix, the **Demand matrix (D)** is slightly different. $d_{ij} = 1$ if the traffic flows from the vertex i to the vertex j.

Note that the diagonal of this matrix should be empty, or else it would mean that a certain node would receive information from himself, which makes no sense.

The mean number of demands is the number of demands divided by the number of nodes.

$$< d >=\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{N} d_{ij}$$

The number of unidirectional demands (edges) is $D_1 = N(N-1)$. N nodes x the other N-1 nodes, however, not every node is connected to every other node. One other way of seeing it is that the D matrix is $N \times N$ but we need to take N away due to the empty diagonal. $D_2 = \frac{D1}{2}$ is the number of bidirectional demands, which is one of the triangles of the D matrix.

Another interesting matrix is the **Traffic matrix (T)** and it's used to denote traffic intensities. It only has entries different from 0 in the exact same places the Demand matrix has. It's used for static traffic designs.

In transport networks the traffic units is the type of client signals: Ex: E3 (34 Mb/s), STM-1 (155.51 Mb/s), GbE (1 Gb/s), 10 GbE (10 Gb/s) ,etc. This traffic units must be converted to traffic units appropriate to be used in network design. These traffic units must be VC-n in SDH networks and in OTN networks ODU-k signals.

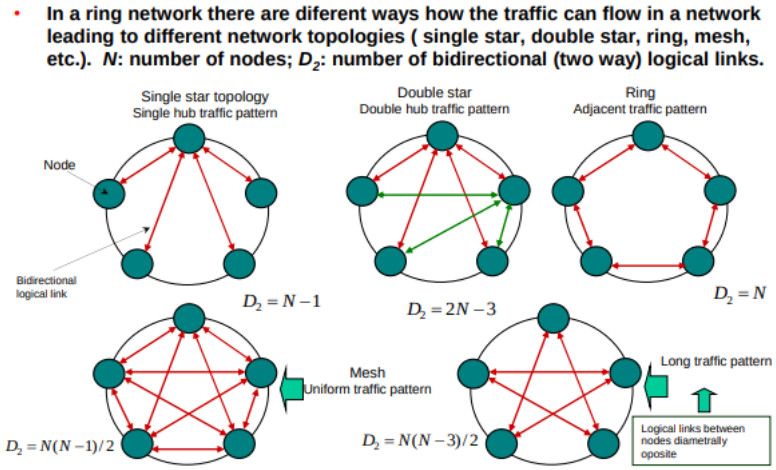
Considering now a Dynamic Traffic case, a few concepts arise:

- Average intensity of data flow between two nodes;
- Traffic bursts are time intervals where the flux of data is considerably higher than the average rate;

- Peak rate is the maximum instantaneous intensity.

Aggregation level or multiplexing level reduces the traffic *burstiness* because there less often flows get fully used and is more likely that the information can flow at a constant pace instead of in bursts.

Note one important distinction between Logical and Physical Topologies that we haven't done yet is that there can be many logical topologies on one physical topology. This can happen in the following way:



Routing is how a packet travels in a network. Therefore, routing ends up being the map between logical and physical topologies.

In optical networks the path between two nodes is usually called "lightpath".

We can also define a **Cost Matrix (C)** where each element c_{ij} represents the costs between nodes i and j.

The path can be performed manually (static routing - Demand matrix is time invariable) or dynamically, through routing algorithms (dynamic routing - Traffic matrix is time dependent, with constant arrival and termination of new demands).

Additionally, if the a given traffic demand(connection) is able to use more than one route, it is called a multipath routing process, else it is a mono-path process. Because there are usually many paths connecting two nodes, some metrics are taken into account when choosing which to follow:

- 1) Minimizing the network cost;
- 2) Minimizing the traffic in the most loaded link;
- 3) Minimizing the number of hops (number of links in the path);
- 4) Minimizing the path distance;
- 5) Maximizing the protection capacity, etc.

Most of the routing strategies incorporate some sort of shortest path algorithm to determine which path minimizes a particular metric, which can be for example 1), or 3), or 4). Note that the same algorithm used to 4) can be applied to 3) making all link distances identical.

From the physical topology, described by a graph G(V,E) and the traffic matrix T, describing all the traffic demands to be routed, one can perform shortest path algorithms such as Dijkstra's algorithm.

Order the demands according to a certain sorting strategy:

- Shortest-first: The demands with the lowest number of nodes in its path come first in the list;
- Longest- first: The demands with the highest number of nodes in its path come first in the list;
- Largest-first: The demands with the highest number of traffic units come first in the list;
- Random ordering: the demands are not known initially.

Route de demands according to the orderings. To break a tie choose the path that minimizes the load in the most loaded link.

Dijkstra's Algorithm

Consider a generic node *i* in a network with *N* nodes from where one wants to determine the shortest path to all the other nodes in the network. Lets l_{ij} be length (cost) of the link between node *i* and node *j* and d_{ij} the length of the shortest path between node *i* and *j*.

Algorithm:

1) Start with the source node i in the permanent list of nodes, i.e. $S = \{i\}$; all other nodes are put in the tentative list labeled S'. Set $d_{ii} = 0$ and $d_{ij} = \infty \quad \forall j \neq i$.
2) For all the adjacent nodes to i set $d_{ij} \leftarrow l_{ij} \quad \forall j$ adjacent to i .
3) Identify the adjacent node j (not in the current list S) with the minimum value of d_{ij} (permanent node), add it to the list S ($S = S \cup \{j\}$) and remove it from ($S' = S' \setminus \{j\}$). If S' is empty stop.
4) Consider the list of neighboring nodes of the intermediate node j (but not consider nodes already in S) to check for improvement in the minimum distance path by setting $d_{ik} \leftarrow \min(d_{ik}, d_{ij} + l_{ij})$. Go to Step 3.

Now is pertinent to introduce yet another matrix, the **Hop Matrix (H)** where each element h_{ij} denotes the minimum number of hops from node i to node j.

The average number of hops

5 Artificial Intelligence / Machine Learning

5.1 Artificial Intelligence

There are some goal states and one initial state. The objective is to find the goal state that is closer (with the shortest path to the root/ with the least search cost). Is given as the solution the steps necessary to perform that path.

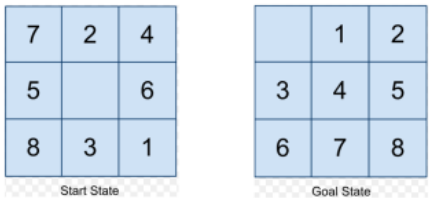
To keep the formulation as general as possible, abstractions are required, keeping in mind that they will need a correspondence when applied to a real problem.

Example of the application to a problem: a vacuum cleaner that needs to clean every square. In this case, only 2 squares are presented, no localization sensors are present, only "rubbish" sensors that tell if the current square is dirty or not.

Therefore:

- States : $< r, d_1, d_2 >$ where r is the robot position, d_1 and d_2 are binary, representing the existence of dirt in each of the rooms.
- Operators/Actions : L (go left), R (go right) or S (suck dirt)
- Goal Test : d_1 = False and d_2 = False. $((d_1 \text{ nor } d_2) == 1)$
- Initial State : $< r, d_1, d_2 >=< 1, T, T >$ is at square 1, and squares 1 and 2 are dirty
- Step Cost : the description of each action cost. In this case, 1 for each one.

8-puzzle



States: specify in a 3x3 matrix the location of the 8 numbered tiles plus the blank one

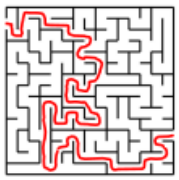
Initial state: any puzzle configuration

Operators: move "blank" to the left, right, up and down

Goal test: checks whether the state matches the goal configuration

Path cost: each step costs 1 (so path cost = number of steps)

Mazes



States: maze configuration plus current location

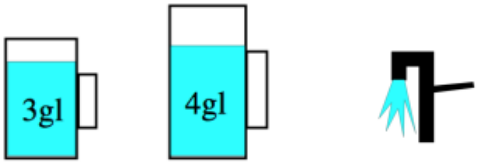
Initial state: any maze location

Operators: move to an adjacent and linked position

Goal test: current location = maze exit

Path cost: number of steps

Water jugs



States: amount of water in each jug (e.g., <1,4>)

Initial state: <0,0>

Operators: fill J4; fill J3;
empty J4; empty J3;
pour water from J4 into J3 until J3 is full or J4 is empty;
pour water from J3 into J4 until J4 is full or J3 is empty;

Goal test: <0,2>

Path cost: amount of water used

Hanoi towers

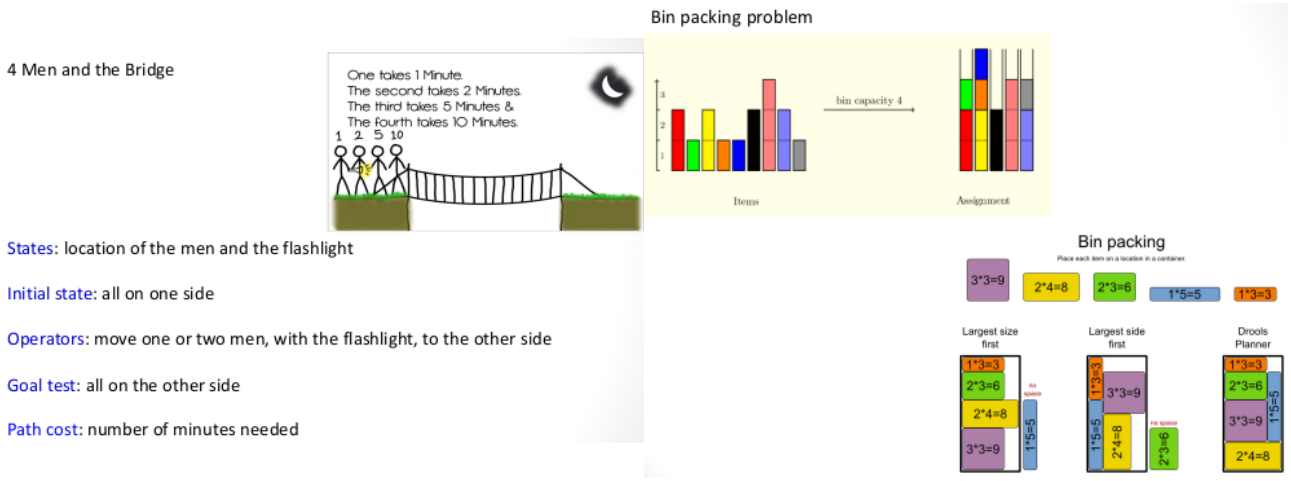
States: location of the disks in the three poles

Initial state: all disks correctly arrange in the left pole

Operators: move a free disk to another pole, which is empty or all disks in it are bigger

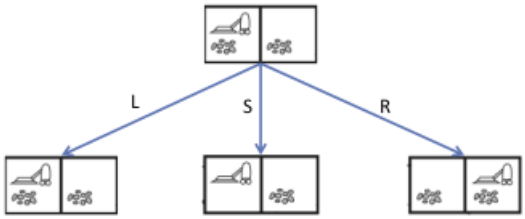
Goal test: all disks correctly arrange in the right pole

Path cost: number of disk moves

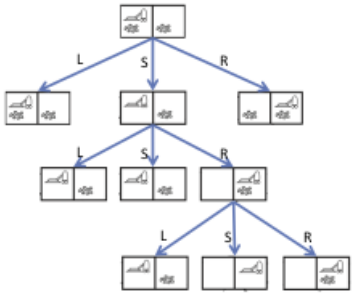


As you can see, every problem is very well defined in terms of the start, the end, the possible moves and what makes a solution better than other. Therefore, the conditions to put the computer thinking how to get from the possible steps to the best solution are assembled. One way the computer should be able to get to the solution is by exploring every move combination and check the state it ended up with.

Given the initial state, use the operators to generate successor states.



Then a choice of which is more "profitable" or more likely to lead to a goal state needs to be made and this cycle continues until the arrival at a goal state.



In this case, the solution would be $\{S, R, S\}$.

However, before diving directly in algorithms, is important to note the characteristics of the Environment and of the Agent. Based on these we'll be able to perform much better choosing the search algorithm that is best for us.

5.1.1 Environment

Observability: how much can the agent know about the environment.

An environment is fully observable if the agent can see everything. Or partial observable: - part of the state is occulted and you simply can't

Deterministic: the effect of the actions are predictable If I move one queen from one place to the other, I can predict the effects, what is attacking what, etc... Instead of Deterministic, it can be stochastic, where the outcomes are functions of probability functions. Therefore, you can't be 100% sure of the outcome. Therefore, you can use probability to make choices...

Neither the environment nor the agent performance change while the agent is deliberating. –¿ Static.

In a dynamic environment, the agent performance can change with time.

Semi-dynamic means that the world is static, but the performance is changing. ex: turn game where the game doesn't change while you are thinking but the more you take, the less point you get.

Continuous or discrete

Sequential or episodic. Episodic means that one episode doesn't influence the next one. It's very related with causality. In episodic environments, there's no influence in consequent problems.

E.g. one game of chess is sequential, but different games are episodic.

The outcomes for all actions are given. –¿ Known environment.

In a case where the less time you spend thinking before answering,

Knowing these is very important because it allows us to best choose the shelf of methods from which we take our algorithms from. Some are best from some things and some can only be used in certain situations as well.

Internally to the agent, you have a way of representing of the world. (Internal representation)

Example when you don't have an internal representation: random vacuum cleaning robots don't know anything about the environment, they just rotate randomly when they see an object.

- Fully observable vs. partially observable
  (partially observable because of noise, inaccurate or faulty sensors, or hidden parts)
- Single agent vs. multiagent (cooperative vs. competitive)
  (in multiagent environments, communication may be a key issue)
- Deterministic vs. stochastic
  (an environment is uncertain if it is not fully observable or not deterministic)
  (nondeterministic environment is when actions have different possible outcomes but no probabilities associated)
- Episodic vs. sequential
- Static vs. dynamic
  (if the environment doesn't change with the time but agent's performance does, it is **semidynamic**)
- Discrete vs. continuous (applies to states, time, percepts, and actions)
- Know vs. unknown
  (depends on the agent's knowledge about how the world evolves – the "laws of physics" of the environment)

	observable	deterministic	episodic	static	discrete
Chess	Y	Y	N	Y	Y
Poker	N	N	N	Y	Y
Taxi	N	N	N	N	N
Image analysis	Y	Y	Y	Semi	N

5.1.2 Agent

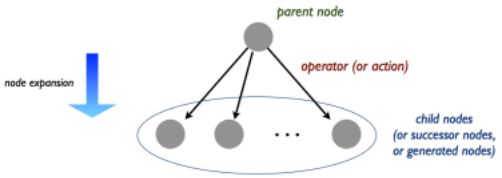
Model-based: typically, when you have partial observability.

Goal-based: when you aim for a goal. You have states and actions and can search for the goal.

Utility-based: are very similar to goal-based but are a bit more, not all goals are the same. There a preference between goals. Utility Theory handles how to translate preferences to numbers.

5.1.3 Search Problems

In essence, is necessary some **Search Terminology** to refer to certain things:



Successor function: given a node, returns the set of child nodes

Open list (or frontier or fringe): set of nodes not yet expanded

Closed (explored) list: set of nodes already expanded

Leaf node: a node without successors

State space can be:

- Tree-based – no repeated nodes in search
- Graph-based – directed cycle graph

General algorithm shape, starting with $open\_list = \{initial_node\}$, iterate over:

1. Select a node from the open_list;
2. Check if it's a goal node (in case it satisfies the goal test). If yes, return solution by backing up to the root;
3. If not, remote it from the open_list, expand it with the successor function and insert the nodes that come from there to the open_list.

Different selection criteria leads to a variety of search methods.

If it's tree based, then no nodes will be repeated and it's not necessary to have a list of the already visited nodes. In a graph however, is needed to have a list of these, or else a cycle is possible.

To evaluate the algorithm, many parameters may be enumerated:

Completeness: guarantee that a solution is found if there is one

Optimality: the solution found minimizes path cost over all possible solutions

Time complexity: how long does it take to find a solution (usually measured in terms of the number of nodes generated)

Space complexity: how much memory is needed to find a solution (usually measured in terms of the maximum number of nodes stored in memory)

Branching factor (b): maximum number of successors of any node

Depth (d) of the shallowest goal node

m = maximum length of any path in state space (may be infinite)

g(n): the cost of going from the root node to node n (path cost function)

Mentioning types of search strategies:

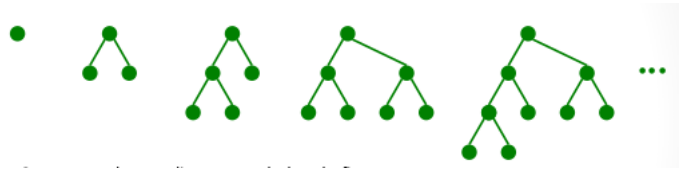
Uninformed (or blind) search: does not use additional (domain-dependent) information about states beyond that provided in the problem definition

Informed (or heuristic) search: uses problem-specific knowledge about the domain to "guide" the search towards more promising paths

5.1.4 Uniformed Search Strategies

A list of **Uniformed search strategies** we'll have a deeper look to:

- **Breadth-first Search** - Select earliest expanded node first - uses a FIFO queue (First In, First out). This leads to opening every node at the safe depth first before moving the deeper nodes.



Because order of depth is followed and every node checked, this search strategy is Complete and Optimal (if the path cost increases - or at least doesn't decreases - with depth). Being **d** the depth of the solution and **b** the branching factor(max number of successors of a node.) then in the worst case, the total number of nodes generated is: $1 + b + b^2 + b^3 + \cdots + b^d$

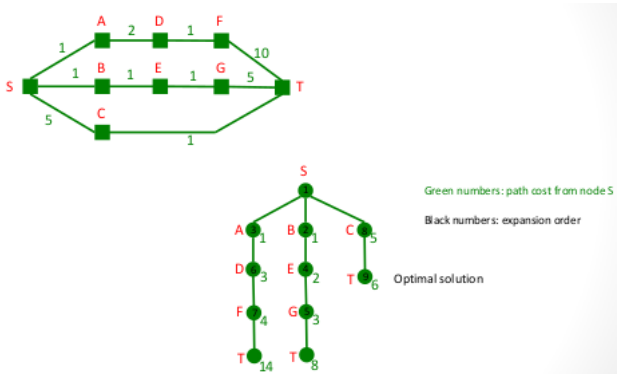
Time Complexity - O(b^d) Space Complexity - O(sum no of nodes) = O(b^d)

Depth	Nodes	Time	Memory
2	110	.11 milliseconds	107 kilobytes
4	11,110	11 milliseconds	10.6 megabytes
6	10^6	1.1 seconds	1 gigabyte
8	10^8	2 minutes	103 gigabytes
10	10^{10}	3 hours	10 terabytes
12	10^{12}	13 days	1 petabyte
14	10^{14}	3.5 years	99 petabytes
16	10^{16}	350 years	10 exabytes

Figure 3.13 Time and memory requirements for breadth-first search. The numbers shown assume branching factor $b = 10$; 1 million nodes/second; 1000 bytes/node.

Note that it makes a difference if you test the node before or after expanding it. If it's tested before, there's no need of expanding the node. If the test is made only after the expansion, the complexities grows to O(b^{d+1}).

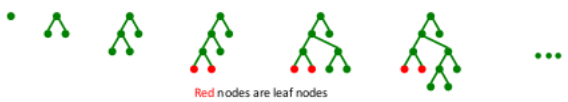
- **Uniform-cost Search** - expands the node that has the smallest cost from the root to it.



Note that each action generates one possible state from the state where the robot was previously.

This search strategy is only complete and optimal if the step costs are strictly positive. Else, it can give several steps that cost nothing to places far away from the solution.

- **Depth-first Search** - Exactly as the name suggests, goes until the deepest node first, and only then looks at the other nodes at the first level. Therefore, it can be very inefficient on a large or infinite tree. Open the last node added to the list (LIFO- Last In, First Out).



- **Backtrack Search** - a variation of depth-first, but expands one node at a time, only stores in memory that only node and the expansion is made by modifying the node, while backtrack is nothing more than undoing the modification..

  It's not complete nor it is optimal... but saves a lot of memory.

- **Depth-limited search** - another variation of depth-first where limiting the search tree to a depth L, is possible to contain the inefficiency. It's complete if the depth of the solution if smaller than L but still not optimal.

  Time complexity: O(b^L) Space complexity: O($b \times L$)

- **Iterative deepening depth-first search** - A variation of the previous one. In this one, the idea will be to run depth-limited search for an increasing L. Run for L=1, L=2, ...

This way, it is complete and it's optimal (if the path cost is a non-decreasing function of depth).

Time complexity: $O(b^d)$ Space complexity: $O(b \times d)$

Node expansion:

Breadth-first

$$1+b+b^2+b^3+\cdots+b^{d-1}+b^d$$

Iterative deepening

$$(d+1)1+(d)b+(d-1)b^2+(d-2)b^3+\cdots+(2)b^{d-1}+(1)b^d$$

Example: b = 10 and d = 5

breadth-first = 111 111

iterative deepening = 123 456 (+ 11%)

- **Bidirectional Search** - Search both from initial node and from the gold node. Note however that can only be used when a goal node is known and when the parent nodes can be computed given its child (through the sets of available actions). It's complete (if breath-first in both directions) and Optimal, if the step costs are equal.

  Time and space complexity: $O(b^{d/2})$

Example: $b = 10$ and $d = 5$

$b^d = 1\,111\,111$

$2\,b^{d/2} = 2\,222$

A summary of the above analysis:

Criterion	Breadth-first	Uniform-cost	Depth-first	Depth-limited	Iterative deepening	bidirectional
Complete?	✓	✓[1]	×	×	✓	✓[3]
Optimal?	✓[2]	✓	×	×	✓[2]	✓[2,3]
Time	$O(b^d)$	$O(b^{1+\lfloor C^*/\varepsilon \rfloor})$	$O(b^m)$	$O(b^L)$	$O(b^d)$	$O(b^{d/2})$
Space	$O(b^d)$	$O(b^{1+\lfloor C^*/\varepsilon \rfloor})$	$O(b.m)$	$O(b.L)$	$O(b.d)$	$O(b^{d/2})$

[1] for strictly positive step costs
[2] for path costs a non-decreasing function of depth
[3] for breadth-first in both directions

A **General Search Algorithm** can be formulated in the following way:

function General-search (***problem***, ***strategy***) **returns** a solution or failure
insert the root node into the open list
(the root node contains the initial state of problem)
loop do
if there are no candidate nodes for expansion **then return** failure
choose a node for expansion according to strategy (using strategy function)
if the node contains a goal state (using goal checking function) **then**
return the corresponding solution
else
for each operator in the list of operators (or successor function)
create a child node (for the new child state)
update child node path cost (using g-function)
add the resulting node to the open list [*unless… see graph search versions*]
end

domain (problem) independent

problem argument should include at least:

- initial state (using a specific state representation)
- successor function: new state = succ (current state, operator)
- path cost function (g-function)

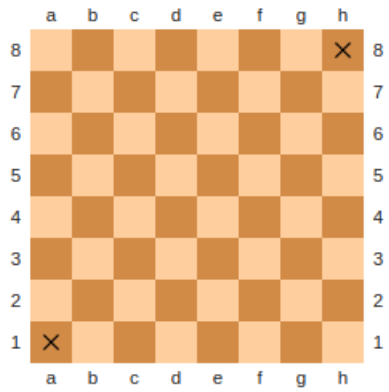
domain (problem) dependent

strategy argument (e.g., FIFO, LIFO, priority queue (by path cost), etc.)

algorithm selection

Finally, there are problem dependent and problem independent details. The search strategies are problem independent, but how we define the actions, states, ect… is problem dependent. And the way we see and think about the problem can be highly related to the way we represent it.

An incredibly well formulated example is the Mutilated Chess Board problem. If we take just the squares of the corners of a chess board, can we still fill the whole board with dominos (each domino takes 2 squares). It becomes slightly harder! If you keep removing squares it gets increasingly harder to figure out by head.



However, if you represent the chess board as the remaining black and white pieces and notice that a domino piece must always cover one black and one white squares, then by taking those two corners then 30 black and 32 white squares will be remaining making it impossible to fill with dominos. As a matter of fact, accordingly with Gomory's Theorem is also possible to say that if 2 square of opposite colours are removed then is always possible to fill the board with dominos! More in: Mutilated chessboard problem

The bottom line is that the representation (problem dependent details) matters.

With this is mind, consider the following example:

Initial state with: bedroom(3), living room(2), kitchen(3), hall(2), truck(0)

a) State: $< pos, n_{bedroom}, n_{living}, n_{kitchen}, n_{hall}, n_{truck} >$

b) Initial State: $< truck, 3, 2, 3, 2, 0 >$

c) m for move, p for push $< m_N, m_S, m_E, m_W, p_N, p_W, p_E, p_W >$. However, is only possible to push in directions where there are rooms (as for walking) and when there are boxes in the current room we are located in.

d) A goal condition could be $n_{truck} = 10$. Another could be the sum of all rooms to be 0. But the first one is more elegant and also seems more general... We don't want to throw boxes out of the window.

Sometimes, just finding the solution is enough. Some times, there's a best solution.

The heuristic (the only difference between informed and uninformed search) is a function of a state that gives us the appreciation we have for that state - how good that state is. If we want the best solution, we must have the heuristic function is key. If you just want a solution, that function can be much simpler or even nonexistent.

5.2 Supervised Learning

Extrema Conditions and Hessian Matrix

Videos: 87 - Warm up to the second partial derivative test to... 89 - Second partial derivative test intuition

This sub (...) sub section aims to explain why the conditions imposed on the Hessian matrix - the second derivative matrix - correspond to imposing in 2D what we know already.

In practice, in 2D, to find a extrema we need to guarantee that the first derivative is zero, which in N dimensions is correspondent to guaranteeing that the gradient is zero in that point, because the gradient is nothing more than all partial derivatives in that point. Therefore, setting $\nabla f = 0$ means that, in that point, the function is not increasing or decreasing in any of the N directions. So the first derivative makes sense.

However, when we go to the second derivative, the meanings get a bit more complicated.

In 2D, if the second derivative was 0, it was probably (not certainly) a saddle point, if it was > 0 or < 0 it was, respectively, a local minimum or maximum.

The conditions we should impose in 3D is to have a positive (for finding a minimum) or negative (for finding a maximum) definite Hessian Matrix. While second derivatives in order to just one variable is possible to attribute a sense to it, why do the cross derivatives play a role as well? And, why do they mean really?

Well, first the explanation on why it is needed: there are functions that across multiple dimensions still show that it is an extrema but then there's an inflexion along directions that are not along the axis. So, checking the axis is not enough. Why checking the cross partial derivatives makes it enough?

The Second Derivative Test

$$f_{xx}(x_o, y_o) f_{yy}(x_o, y_o) - f_{xy}(x_o, y_o)^2 \gtrless 0$$

If it is greater than 0, we have a maximum or a minimum and have to check the value of f_{xx} or f_{yy} to be sure. If it is less than 0, we have a saddle point. If it equals 0, then we don't know if it is a saddle point, but it is not a min or max therefore, at least for now, we certainly don't care.

Cross or Mixed partial derivatives can be switched? Yes if the function is C^2. (Boring to prove theorem called: Schwarz' Theorem)

Therefore, we just need to compute one of the cross derivatives.

Also this works because the second derivative test is nothing more than the determinant of the Hessian matrix. The determinant is the product of every eigenvalue of that matrix, therefore it can only be positive if they are both positive or both negative, in which cases there is, respectively, a minimum or a maximum.

But why do eigenvalues tell us this? Because they tell us how the eigenvectors are scaled! And the eigenvectors of such matrix will be the greatest and the least curvatures. Therefore, they either have the same signal / are scaled the same amount, or

Some other links that helped with this:

- Differential Geometry
- Criterior for critical points - Maximum, Minimum or Saddle?
- David Butler - Facts about Eigenvalues

5.2.1 Neural Networks - BackPropagation

Two of the most useful websites to check while trying to demonstrate the backpropagation algorithm:

- all backpropagation derivatives
- Error (deltas) derivation for backpropagation in neural networks
- 3Blue1Brown Neural Networks - Specially the last one, on backpropagation.

Why kernels are important? They facilitate a lot the mapping of features to higher dimensions!

Why kernels?

5.3 Unsupervised Learning

5.3.1 Lagrange Multipliers

Perfect Explanation why it works:

Quora - Intuition on Langrange Multipliers

And a video showing exactly how it's done:

Khan Academy - Lagrange Multipliers Example

5.4 Reinforcement Learning and Decision Making

An agent to make the wisest decision in its situation has to have knowledge on what state he is in, how the environment will evolve, how it will be like if he performs a certain action (taking into account stochastic environments) and a utility function to know how much that state contributes to its happiness.

6 Theory on Variate Topics

6.1 Taylor Series

Incredibly important for simplifications, approximations and to have a notion of the shape of the function.

The definition is as follows: A real or complex-valued function $f(x)$ that is infinitely differentiable at a real or complex number a can be written as a power series the following way

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!}(x-a)^n = \\ = f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f'''(a)}{3!}(x-a)^3 + \cdots$$

Some utilities of this formula:

- The approximation $(1+x)^n \approx 1 - x^n$, when x is much smaller than 1. (Binomial Series)
- When approximating $y(x) = cos(2x) + \epsilon$ with a polynomial using least squares, the coefficients fo that polynomial will tend for the coefficients of Taylor's polynomial. Many MacLaurin Series (Taylor Series at the origin) can be found Here in the Wiki .

6.2 Erlang Models B and C

These are statistical models to predict the necessary resources to satisfy a certain amount of traffic with the constraint of not serving at most a certain percentage of the incoming traffic at a given time.

Instead of thinking like:

> "" *Hey, it's simple arithmetic! We get 3 200 calls a day. That's 400 calls an hour. Each call lasts three minutes, so each person can handle 20 calls and hour. So we'll need 20 incoming lines and 20 people to answer the phones.*"

Good models were created because of this! This is completely wrong if the calls don't arrive evenly distributed across the day, which is very likely. There are busy hours where the traffic is much bigger than in any other time of the day.

A.K. Erlang derived nice formulas for us to use. Note that the terms "call" and "resource" will be used interchangeable because these models can be used for a immense variety of applications besides call-centres. For instance, to figure how many printers to buy so that people in the office can print decently.

One last note: these formulas were derived for "infinite sources" of incoming traffic. However, they work very very well for cases when there are about 10x more sources than trunks. For real world applications that tends to be the case.

The concepts:

- **Erlang** : Dimensionless unit. Regards the continuous use of a resource. Normally is referred to hours, therefore 90 minutes of traffic are regarded as 1.5 erlangs.
- **Grade of Service** : Probability of all servers/trunks being busy when a resource is requested for a given amount of traffic and for a certain amount of trunks.

Erlang B

Use if a call is really blocked, i.e if the person trying to use the resource doesn't wait to use it when all the trunks are busy at that time. Given 2 from Traffic , Grade of Service or Number of Trunks one may calculate the other.

Tables can be found here.

Erlang C

Use if a blocked call is delayed instead of the "come back later" that happens in the previous model. This one needs as well the average duration of each call .

7 English

Unbelievable how this got to this point, right? I know, I know... But,indeed English is important.

7.1 Commas

Always the same thought "should I put a comma here?" followed by the exercise rereading the sentence multiple times until one makes up his mind. Fluency requires well founded rules and this is what we are here for.

To do after: check this website, it seems to have cool rules Grammar Commas

7.1.1 Commas on adverbs like *therefore,* *however,* and *indeed*

When the adverb is essential to the meaning of the clause, or if no pause is intended or desired, commas are not needed.

Two sentences showing when commas aren't needed:

- If you cheat and are therefore disqualified, you may also risk losing your scholarship.
- That was indeed the outcome of the study. (this one goes both ways really, how'd you like it to sound?)

Two other showing when they fit well:

- A truly efficient gasoline-powered engine remains, however, a pipe dream.
- Indeed, not one test subject accurately predicted the amount of soup in the bowl.

7.1.2 Commas to separate Adjectives - Only if they are parallel

Consider the following sentence "The event is part of a catchy, public health message about the importance of emergency preparedness." Catchy and "public health" are not coordinate adjectives. The point is not that the message is catchy and public health; it's that the public health message is catchy. Therefore, no comma is necessary: "The event is part of a catchy public health message about the importance of emergency preparedness."

If, by contrast, the sentence read, for example, "The event is part of a catchy, quirky message about the importance of emergency preparedness," note that because catchy and quirky are parallel — they are coordinate adjectives — a comma should separate them.

7.2 British English vs American English

This probably was the main reason for the birth of this section. I never know when to use certain letters in both spelling ways. So this part will cover firstly that issue. The following website was helpful in many ways and I'll cite others as the list gets extensive.

Wikipedia - American and British English spelling differences

Moreover, I intend to organise this collection in many sub-sections, each presenting a ground truth or a general rule that allows one to replace the doubts with a short recall and reasoning exercise, which tends to be quicker and feel better than searching the thing in Google.

7.2.1 -ise, -ize (-isation, -ization)

This is the worst. But the pain will be over after a few paragraphs.

Generalise is British, I guess. That's how far I've gone so far.

8 MATLAB

8.1 Plots

First things first: to create a vector of equally spaced elements is best to use **linspace** . It guarantees the same amount of points in each array more easily.

```
t = linspace(-Ts, Ts, 200);
plot(t, 2*t);
xlabel('x axis, duh');
ylabel('y axis, hud');
title('guess?');
grid minor; %sometimes..
```

Use **meshgrid** and **surf**ace to plot functions with multiple variables:

```
[xx, yy] = meshgrid(x,y);
surf(xx,yy, 2 * xx, 'EdgeColor', 'none');
```

Meshgrid is necessary because matlab doesn't know how to take do value iteration! The **surf** function DOES know that should take one value from x, run for all values of y and get the (x,y) value for each iteration from z. THEREFORE, z must be an length(x) by length(y) matrix! This is why the mesh grid command is used. It allows for this matrix creation just in the same way as one creates values for a 2D plot.

What **surf** will do is: check the sizes of the x and y vectors to see if it should iterate or just choose the right values. If x and y are row vectors, it will have to manually calculate all the possible combinations. If they are matrices (like the ones returned by meshgrid), it is able to just choose the appropriate combinations: take each element of each vector and plot it in space - notice what is returned by meshgrid is exactly the same dimensions as the Z vector will be.

To conclude: is possible to give x and y or X and Y as inputs for the **surf** function of MATLAB, however, Z must be a matrix! And that will only happen with meshgrid, otherwise it will be a vector without all combinations of values in x an y.

8.1.1 Contour

In portuguese "contorno", returns the levels curves of the function.

contour(X1, X2, Z, 10.^(-20:3:0)') will plot the Z points of the coordinates X1, X2, on the levels 10.^[-20 -17 -14 -11 -8 -5 -2]. Note that if the points X1 and X2 are not provided, there is no "set" where to which is possible to compute the level curves, so Matlab chooses one and it may not fit your previous graph.

8.1.2 Extra Stuff for Graphs

Check all properties of a graph here. Some of them:

Subplots - allow several figures in the same window. Writing subplot(1,2,1) divides a window in a grid of 1x2 and will access the first element of it. Increasing the final number changes the division of the grid selected which is where the subsequent plots are going to end up on.

xlabel, ylabel - labels for the Graphs

xlim, ylim - limits for the Graphs

set current figure position (and size) - set(gcf,'position',[x0, y0, width, height]); x0 is measure from the left of the screen. y0 is measure from the bottom of the screen. width is along x, height along y.

imagesc - Image with scaled colours. imagesc(A); colorbar; shows quite fast the values that the matrix A has. A very useful tool to debug and to quickly have an insight on the matrix contents and patterns.

LineWidth : a number for the thickness of the line

Dashed line : '–' makes the line dashed.

DisplayName : To specify a specific legend

legend is possible to position the legend according to the cardinal points, "North", etc... Use legend('Location', 'North').It is also possible to put it in a desired location with legend('Position', [.25 .25 .25 .25]).

'HandleVisibility' 'on' by default. This doesn't show a legend for a certain graph, useful for when we need to plot thresholds or something that we don't want to have a legend about.

8.2 Functions

To make a function, create a new file and write in the first line:

```
function [o1,o2,o3] = nameOfTheFunction(i1,i2,i3)
    %func code here
end
```

Where the o's are the outputs and the i's are inputs.

Is also possible to pass a function as an argument! For instance, if that function is what is suppose to be used inside another function.

integral(@log, x) is passing the built-in function *log* as an argument.

8.3 Set and Matlab Objects

Since you've probably done something with Java or something with python oriented in that sense, you probably know what an object is. In Matlab, there are many object as well. Instead of changing its properties in the arguments of the instantiation of that object, one may create that object and then use the set method.

The following two pieces of code do exactly the same thing regarding legends.

```
    h = legend;
    rect = [0.25, 0.25, .25, .25];
    set(h, 'Position', rect);

    legend('Position', rect);
```

The rectangle should be [x0, y0, width, height], in percentages, given that 1 is the size of the figure.

check

Actually, I have the sensation that in this case, "Position" is nothing more than an internal variable.

8.4 Save images

```
fig = figure;
plot(t,x);
pause(0.2); %may be necessary so that the figures ends plotting, before it starts saving
print(fig,'-dpng','img_filename'); %as png
```

Or use the following for printing the current figure:

```
plot(t,x);
print('myfig.png', '-dpng');
```

OR EVEN imagining it was necessary to open a file to plot that picture. File named 'ola.bin' (binary file), for instance. If the image is going to be included in some report, using .eps is a good idea since it allows "infinite zoom" because it describes the picture by vectors, instead of saving the information of each pixel of compressed.

The code below will save as .eps (Encapsulated PostScript) file.

```
plot(t,x, 'r');
hold on;
plot(t, x_window, 'b');
print([file(1:end-4) '-hamming'], '-depsc')
```

Check more formats in print page of matlab.

8.5 Opening stuff

In CSV format:

```
file = csvread('file.csv');
```

But better is to open as a matrix, in a big variety of formats:

```
%Excel
A = readmatrix('File.xlsx','Sheet', 3,'Range','A1:AX5000');
A = readmatrix('file.xlsx','Sheet', "name of sheet3",'Range','A1:AX5000');
%More formats coming when I use it for them
```

8.6 Max and Min

Max function is incredibly useful. Returns the maximum of a vector and the place it appears in! Of course, the **min** flavour exists too.

```
[maximum, index] = max(1:10)
```

You can even specify the dimension on which you want the maximum. So, if you want the maximum of each row, max(A, 2) will do a column vector with as many maximums as there are rows, thus a column vector.

8.7 Other useful tools

- It is possible to see **EVERY** command written in matlab. Just write 'commandhistory' or press the above arrow. The commands there are OLD!
- **Load** , **save** , **exist** , **return** to stop execution and **clear** one variable:

```
try
    B = load("A.mat");
    B = B.A; %to get the matrix out of the structure
catch
    if ~exist('Traces.xlsx', 'file')
        return
    end
    %create A code here;
    clear aux;
    save('A.mat', 'A');
end
```

- **squeeze** to take away not needed dimensions. For instance in a 1x1x5000 vector, the following code will result in a 5000x1 vector.

```
squeeze(R(prb,1,:))
```

- **find** to find an element in an array! Simply returns a vector with the indexes of where that element appears.

```
find(a == 0)
find(a < 5)
find(a == b) %return all i that a(i) == b(i), i.e all elements that are in both vectors
```

"The relational operators (>, <, >=, <=, ==, ~=) impose conditions on the array, and you can apply multiple conditions by connecting them with the logical operators and, or, and not, respectively denoted by the symbols &, |, and ~." From the Matlab Docs .

Additionally: "[row,col] = find(X) returns the row and column subscripts of each non-zero element in array X " And this explains the above uses of this function.

- A cool way of **copying an array size** :

```
a = zeros(size(b)) %copies size of b to the use we want to give a
```

- To select random elements of an array: permute the indexes and select the first n elements of that permutation.

```
m = 100; n = 10;
a = 1:m;
randIndexes = randperm(m);
b = a(randIndexes(1:n));
```

- Cell to Logical to Double:

```
B = double(cell2mat(A));
```

Copy

The `cell2mat` call converts it from a `cell` to a `logical` array, and `double` converts it to a double array.

8.8 Label data in Scatter plots

By far the best way of getting a decent result is to use an already written function. Adam Danz published here a function that does this perfectly for you.

From the examples it becomes very evident the use. The function requires at least the coordinates of every point and the labels to put in every data point.

```
labelpoints(x, y, 'Color', [1 0.5 0.5], 'FontSize', 12);
```

8.9 Create Gif from plots

It can't get easier than this. Chad Greene wrote a miraculous script and published it here .

Before any plot, write:

```
gif('mygiffilename.gif');
gif(gif_name,'DelayTime',0.2,'LoopCount',1,'frame',gcf);
```

Then, to insert a frame simply write **gif** :

```
for i = 1:10
    plot(x,y);
    gif;
end

web('mygiffilename.gif'); %will open on matlab web.
```

That is it. It can't get simpler. To view the gif with controls you can use a video player.

8.10 Write table to Excel

A matrix looks like a table but will look like a line if you use the writematrix. Instead, use **writetable** and convert the matrix to a table before!

```
table = array2table(squeeze(avg_rates_perTest(1,:,:)))

for scheduler = 1:length(schedulers_for_testing)
    writetable(table, ...
               num2str(scheduler) +"-avg_rates_per_test"+ ".xlsx");
end
```

Additionally, is possible to add the correct names to the columns, but they have to have possible variable names... So there are some limitations: just letters, numbers and underscores and has to start with a letter.

```
table = array2table(array, 'VariableNames', {'first_name', 'second', 'etc'});
```

9 LaTeX

In this section, basic easy to search commands will be presented.

As a very honourable mention as a perfectly clear LaTeXtutorial, please visit latex-tutorial.com/tutorials/ . I couldn't like that website more. It has the symbols, the tools and the tutorials in the top bar.

Although I prefer much more to use tablesgenerator.com/ a table generator, the symbols and tutorial sections are simply perfect.

9.1 Symbols that you never remember

First of all, the best place to search is Detexify . You can draw a symbol and the latex correspondence appears right away. For other common options, here a list:

- $\sim$ **Tilde** : $ \sim $ (uses: textcomp)
- \ **Backslash** : \textbackslash
- $\ll$ and $\gg$: \ll and \gg
- $\widetilde{\mathbf{h}}$: \widetilde{h}
- $\circledast$: \circledast
- $\otimes$: \otimes

9.2 Important Packages

Import packages with:

```
\usepackage{packagename}
```

Some important package to keep around:

- verbatim: allows for "\comments" which are very useful.
- todo or todonotes: allows for "\todo" and that is the best way of having reminders in your code.
- bm: allows for bold math "$\mathbf{H}$" = $\mathbf{H}$

9.3 Margins

To change the margins of the document this is simply the best place to go. The example sets you up with a standard wide paper size. Overleaf - Margins

9.4 Code listings

[language=Tex]

Use package:

```
\usepackage{listings}
```

To input a complete file of code, use the expression below.
In this case, the file is assumed to be right outside the report folder. Note also that the code should not have non unicode characters like º or else LaTeXwon't do a proper job, even if they are in strings.

```
\lstinputlisting{../gen_data1.m}
```

To paste code in a certain language do:
(the "no numbers" serves to suppress the numbers on the left, to be easier to copy)

Create a list of all code listings with:

```
\lstlistoflistings
```

9.5 Images side by side

From this tex.stackexchange question :

- 2 images: Side by side with one legend for each and one for both

```
\begin{figure}
    \centering
    \begin{subfigure}{.5\textwidth}
      \centering
      \includegraphics[width=.4\linewidth]{image1}
      \caption{A subfigure}
      \label{fig:sub1}
    \end{subfigure}%
    \begin{subfigure}{.5\textwidth}
      \centering
      \includegraphics[width=.4\linewidth]{image1}
      \caption{A subfigure}
      \label{fig:sub2}
    \end{subfigure}
    \caption{A figure with two subfigures}
    \label{fig:test}
\end{figure}
```

- 2 images: Side by side with one legend for each:

```
\begin{figure}
    \centering
    \begin{minipage}{.5\textwidth}
      \centering
      \includegraphics[width=.4\linewidth]{image1}
      \captionof{figure}{A figure}
      \label{fig:test1}
    \end{minipage}%
    \begin{minipage}{.5\textwidth}
      \centering
      \includegraphics[width=.4\linewidth]{image1}
      \captionof{figure}{Another figure}
      \label{fig:test2}
    \end{minipage}
    \end{figure}
```

- 4 images: 2x2 config. Note that is possible to replicate the reasoning for this code and make any number of figures side by side.

```
\begin{figure}[ht]
    \label{ fig7}
    \begin{minipage}[b]{0.5\linewidth}
      \centering
      \includegraphics[width=.5\linewidth]{example-image-a}
      \caption{Initial condition}
      \vspace{4ex}
    \end{minipage}%%
    \begin{minipage}[b]{0.5\linewidth}
      \centering
      \includegraphics[width=.5\linewidth]{example-image-b}
      \caption{Rupture}
      \vspace{4ex}
    \end{minipage}
    \begin{minipage}[b]{0.5\linewidth}
      \centering
      \includegraphics[width=.5\linewidth]{example-image-c}
      \caption{DFT, Initial condition}
      \vspace{4ex}
    \end{minipage}%%
    \begin{minipage}[b]{0.5\linewidth}
      \centering
      \includegraphics[width=.5\linewidth]{example-image}
      \caption{DFT, rupture}
      \vspace{4ex}
    \end{minipage}
  \end{figure}
```

9.6 Equations and Math

As before, a list of useful commands. First the code, then the result.

- System of equations:

```
\begin{equation*}
    \begin{cases}
       a = b \\
       c = 0
    \end{cases}
```

```
    \end{equation*}
\end{itemize}
```

$$\begin{cases} a = b \\ c = 0 \end{cases}$$

- Matrixes: short story

```
\[
\begin{bmatrix}
    x_{11} & x_{12} & x_{13} & \dots & x_{1n} \\
    x_{21} & x_{22} & x_{23} & \dots & x_{2n} \\
    \vdots & \vdots & \vdots & \ddots & \vdots \\
    x_{d1} & x_{d2} & x_{d3} & \dots & x_{dn}
\end{bmatrix}
\]
```

$$\begin{bmatrix} x_{11} & x_{12} & x_{13} & \dots & x_{1n} \\ x_{21} & x_{22} & x_{23} & \dots & x_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{d1} & x_{d2} & x_{d3} & \dots & x_{dn} \end{bmatrix}$$

- Matrixes long story: from matrix environment stackexcange question, " amsmath: it defines 6 types of matrix environments: matrix (without any delimiter), pmatrix (delimiters: ( )), bmatrix ([ ]), Bmatrix ( ), vmatrix (— —), Vmatrix (—— ——)"

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

- If the matrix is coefficient extended, then some smart stuff needs to be used. Use the command below to implement the workaround, written by Stefan Kottwitz

```
\makeatletter
\renewcommand*\env@matrix[1][*\c@MaxMatrixCols c]{%
\hskip -\arraycolsep
\let\@ifnextchar\new@ifnextchar
\array{#1}}
\makeatother
```

After that you can use:

```
\begin{equation}
    \begin{bmatrix}[cccc|c]
     1 & 0 & 3 & -1 & 0 \\
     0 & 1 & 1 & -1 & 0 \\
     0 & 0 & 0 & 0 & 0 \\
  \end{bmatrix}
\end{equation}
```

$$\left[\begin{array}{cccc|c} 1 & 0 & 3 & -1 & 0 \\ 0 & 1 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{array}\right]$$

- **A good practice** (depending on the context) would be to only number referenced equations. For this, all equations should have a short label like "eq:eqlabel", be referenced with \eqref and you need to use this to enable that automatic behaviour of only display names if referenced:

```
\usepackage{mathtools}
\mathtoolsset{showonlyrefs}
```

- Funky stuff. Argmin and a way of doing the norm. If a norm that fits automatically is required, then Google it and put it here.

```
\begin{equation}
    \hat{\bb{S}} = \operatorname*{argmin}_\bb{S} \left\Vert\bb{X} - \bb{A}\bb{S}\right\Vert^2_F
\end{equation}
```

$$\hat{\mathbf{S}} = \underset{\mathbf{S}}{\operatorname{argmin}} \|\mathbf{X} - \mathbf{A}\mathbf{S}\|_F^2$$

9.7 Multicolumns

9.7.1 Multicolumns in Text

```
\setlength{\columnseprule}{1pt}
\def\columnseprulecolor{\color{blue}}

\begin{document}

\begin{multicols}{3}
[
All human things are subject to decay. And when fate summons, Monarchs must obey.
]

Hello, here is some text without a meaning. This text should show what
a printed text will look like at this place.

If you read this text, you will get no information. Really? Is there
no information? Is there.

\columnbreak

This will be in a new column, here is some text without a meaning. This text
should show what a printed text will look like at this place.

If you read this text, you will get no information. Really? Is there
no information? Is there...


A blind text like this give you information about the selected font, how the letters are written and an impression of the look.
     This text should contain all letters of the alphabet and it should be written in the original language. There is no need for
      special content, but the length of words should match the language.
\end{multicols}
```

Results in:

All human things are subject to decay. And when fate summons, Monarchs must obey.

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place.

If you read this text, you will get no information. Really? Is there no information? Is there.

This will be in a new column, here is some text without a meaning. This text should show what a printed text will look like at this place.

If you read this text, you will get no information. Really? Is there no information? Is there...

A blind text like this give you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in the original language. There is no need for special content, but the length of words should match the language.

9.7.2 Multicolumns in lists

```
    \usepackage{multicol}
    \begin{multicols}{2}
        \begin{itemize}
            \item item 1
            \item item 2
            \item item 3
            \item item 4
            \item item 5
            \item item 6
        \end{itemize}
    \end{multicols}
```

Results in:

- item 1
- item 2
- item 3
- item 4
- item 5
- item 6

9.8 Itemize, Enumerate and Lists

I just found this website that has a great tutorial about these.

The main and most interesting thing I wanted to show here is a thing that is also present in the link with further explanations: is possible to change the bullet point style very easily:

```
\begin{itemize}
    \item here is one item
    \item[--] here is another
    \item[$\ast$] here another
    \item[$-$] Here another and I can continue!
\end{itemize}
```

Just by putting the character inside the rectangular parentheses is possible to create a list with that character as a bullet. The result of the above code:

- here is one item

– here is another

∗ here another

– Here another and I can continue!

Another way, better in terms of verbose per list, is this one:

```
%requires \usepackage{enumitem}
\begin{itemize}[label=$\ast$]
    \item ok, now they are the same
    \item see?
\end{itemize}
```

∗ ok, now they are the same

∗ see?

9.9 How to insert images from files outside the report file

Don't know... Tell me if you find out. I just know from inside the folder.

9.10 Good Tables with that diagonal line

Make a table with the tablegenerator.com and then use this:

```
%uses \usepackage{diagbox}

\begin{table}[h]
    \small
    \begin{tabular}{c|c|c|c}\hline
        \backslashbox{Room}{Date} & Monday & Tuesday & Wednesday \\ \hline
        room1 & & & \\ \hline
        room2 & & & \\
    \end{tabular}
\end{table}
```

Room \ Date	Monday	Tuesday	Wednesday
room1			
room2			

9.11 Useful little things

9.11.1 Tables

- **Centre tables' captions** , works for figures and tables

```
\usepackage[center]{caption}
```

- **Set space between columns**

```
\setlength{\tabcolsep}{2.8pt}
```

- ...

9.11.2 Horizontal lines in a page

Like this:

“An important quote here should be placed.”

---

or even like this

Explain what ”I am hungry” means in Portuguese/Spanish if translated literally, i.e if translated to ”Eu sou esfomeado”/”Yo soy hambriento”.

I used, respectively these two pieces of code:

```
\hrule
``An important quote here should be placed.''
\hrule

%needs the definition before.
\def\Vhrulefill{\leavevmode\leaders\hrule height 0.7ex depth \dimexpr0.4pt-0.7ex\hfill\kern0pt}

\noindent\Vhrulefill \ \bb{or even like this} \Vhrulefill

Explain what "hello" means.

\noindent \Vhrulefill
```

9.11.3 Others

- \appendix - used to start an appendix. Right after that, put \section and a label inside the section and then the appendix will be referenced with a letter.

10 Python

10.1 Important Concepts

Python is a language with many characteristics, ones more obvious and easy to understand, like in-fix notation, others are more complex, like Dynamic Typing and Automatic Memory Allocation.

Moreover, many programming principles like anonymous functions, map, zip are important and should be addressed in order to achieve flexible and overall good programming skills.

Beforehand, check the Python Notebook and the slides from Rodrigo Ventura in the "Additional Material" Folder. They are very complete and give a proper insight on how everything works.

Tuples are faster than lists. So, if the idea is doing a very big iteration where there's no memory problem, converting to tuple before probably helps.

However, regarding speed, always use built-in functions and libraries! Almost always what is slowing the program down is a huge for loop... Libraries are compiled already and are very optimised, therefore by using them you are probably running very optimised C code, which couldn't be faster than it is.

10.1.1 Lambda and Anonymous functions

Lambda is the keyword used to make an anonymous function. The following 2 pieces of code do exactly the same.

```
    def my_key(x):
    return x[0]

    l.sort(key=my_key)


    OR

    l.sort(key = lambda x: x[1])
```

And a short example is:

```
    -----short example on how cool python is-----
    def make_multiplier(factor):
    return lambda x: factor*x

    f = make_multiplier(2)
    -----------------------------------------------
```

Therefore, lambda is nothing more than defining a function, without giving it a name. It helps keeping the code simple specially when the function is just going to be used once.

10.1.2 __main__

This can be perfectly understood in the following stackoverflow post .

In a nutshell, when a script is executed python assigns many names to certain variables like __main__. Is the script is executed in the terminal, then main will be the name of that script (without the .py). The *if __name__ == '__main__':* aims to distinguish between situation where the script was imported, so that the functions it contains can be used elsewhere (in this case the main name won't be the name of the script which is stored in __name__) and the situation where the script is directly called in the terminal. If the script is directly called, for it to do something, something has to execute and that's usually what goes inside that if. When the script is imported, most of the times only the definitions matter and the calls will be made in the script that is calling that one.

Therefore, is a good tool in case you want to make a script importable but also callable.

10.2 Some useful tools

To check if a variable points to a certain data type:

```
    isinstance(var, [list, tuple, int])
```

To define functions with optional arguments and call them in the incorrect order:

```
1
2     def draw_point(x, y, color='red', thickness=2):
3         print('x =', x, 'y =', y, 'color=', color, 'thickness=', thickness)
4
5     x = 1
6     y = 2
7     draw_point(x,y,'blue', 5)
8
9     draw_point(x,y,thickness=2, color='blue')
```

10.3 Pandas

Check this link :)

10.4 Jupyter Notebooks

Along with Anaconda comes a full installation of the most recent python version and the Jupyter Notebooks, which are awesome to write python.

Here are some very useful shortcuts to tame that beast:

There are 2 modes of shortcutting:

- The Command Mode (when border of the cell is blue). Press ESC to access this mode.
- The Edit Mode (when border of the cell is green). Press ENTER to access this mode.

The Edit Mode is clearly superior:

- Ctrl + Enter to run cell
- Shift + Enter to run cell and get directly to the next
- Ctrl + D to delete a whole line
- Ctrl + arrows jumps words (usual)
- Ctrl + Backspace deletes whole word (usual)

The Command Mode can be usefull sometimes, especially when adding cells is needed, but remember that the cell has to have a blue border:

- A - insert cell before
- B - insert cell after
- DD - delete current cell
- Z - undo cell deletion

10.5 From Python 2 to Python 3

In Linux, if 2to3 is installed, one may simply run:

```
1     2to3 -w -n file.py
```

This will write the file translated into python 3 in the same file(-w) and without creating a backup(-n).

11 Linux

11.1 How to build from source

1. Be sure you have everything needed:

```
sudo apt-get install build-essential
```

2. Download source. May come in various *.tar* formats. The flags **xvf** mean e**x**tract to a **f**ile with **v**erbose. When not specified, the file will have the same name. Then flag **z** for *.gz* or flag **j** for *.bz2*. If the extension is *xz* then no flag is needed because GNU tar recognizes the format by default. The code should be one of the following:

```
tar -xzvf file.tar.gz
tar -xjvf file.tar.bz2
tar -xvf file.tar.xz
```

   Note: sudo apt install xz-utils if tar gives error.

3. Check dependencies, i.e if you have everything necessary to build that package. It may require libraries specific to that package. Do the following code or check the README or INSTALL files.

```
./configure
```

4. Now various paths are possible. First of all, read the README and/or the INSTALL to look for instructions.

11.2 How to change Permissions and Ownership

Sadly, this happens very often. Anytime something like "permission denied" is presented in the terminal or similar but anywhere else, it is because permissions and ownerships are not set the right way. If they were, the owner of the file would be able to do what he wants with it.

First, what the permissions mean?

- **read** restricts or allows viewing the directories contents, i.e. ls command
- **write** restricts or allows creating new files or deleting files in the directory. (Caution: write access for a directory allows deleting of files in the directory even if the user does not have write permissions for the file!)
- **execute** restricts or allows changing into the directory, i.e. cd command

4 is read. 2 is write. 1 is executing. Therefore 6 is r+w, 7 is r+w+x-.

Additionally, running **ls -l** one gets something like **-rw-r–r–**. Each 3 refer to Owner, Group or Other. In this order.

Is possible to do the command in 2 different ways. The quickest is just using the numbers. One number per group. Like this:

```
chmod 754 filename
```

This way, the owner can rwx, group rw and other people/users can only read the file. Note: 770 is possible to not even allow the files to be discovered by other users.

Now we know how to alter what certain groups can do to a file. However, if the files doesn't belong to the correct group, those permissions changes may not do much regards the "permission denied" problem. Therefore, how to change ownership?

With **ch**ange **own**er duh!

```
sudo chown owner_name:group_name filename
```

A little small note: **whoami** gives the user name. Alternatevely, **users** and **groups** give the users with initiated sessions and the groups the current user belongs to.

To finalize, why running **./script.sh** and running **bash script.sh** require different permissions?
The first one needs executing permissions and the second just needs read permissions. This is because the first

we are saying to execute the shell script. In the latter it is said to 1º load bash, then give "script.sh" as an argument to it. Therefore, bash just needs to read what is written in the script.sh file.

Running scripts as ./script.sh is a more general way as it also works with ./script.py as long as the script has a *hashbang* (#!)inside to tell the program loader what should be used.

11.3 Formatting a partition as exFAT

exFat is readable in every computer. In linux is necessary to install some utilities:

```
sudo apt-get install exfat-fuse exfat-utils
```

First: HOW TO MANAGE PARTITIONS?

Ans: install gparted, it does a good job.

Second: HOW TO FORMAT AS exFAT?

Ans: gparted can't do this... supposing the devices has only one partition (all partitions can be deleted in gparted to achieve this) and is called mmcblk0 in the devices directory.

To list all connected storage devices

```
sudo fdisk -l
```

To wipe the filesystem (almost the same as gparted does):

```
sudo wipefs /dev/mmcblk0
```

To create only one partition. Note: in the fdisk utility is necessary to confirm several intermediate steps, because of that "all enters" means to press enter until other instruction in the normal utility prompt is asked.

```
sudo fdisk /dev/mmcblk0
n
"all enters"
t
7
w
"all enters"
```

I am pretty sure until lhere it could be done in gparted as well. But now comes the important part! After creating a partition, when doing sudo fdisk -l, there is a partition in the device we want to format. This is the partition we will format as exFAT. That partition in this case is called mmcblk0p1 (makes sense right?). Do the following to format is as exFAT:

```
sudo mkfs.exfat -n hardDisk /dev/mmcblk0p1
```

hardDisk is nothing more than the name you want to call the partition, doesn't matter.

You can check everything went fine with:

```
sudo fsck.exfat /dev/mmcblk0p1
```

Congratz, done :)

11.4 Install Custom ROM with Linux

The custom ROM is installed in the android device, as I hope you remember...

The major difference in the processes between Windows and Linux is in the initial part. After the initial part concerning installing TWRP in the phone, the rest is trivial. In fact, I've just noticed that in Windows is necessary to write several commands as is required in Linux, so by covering the part in Linux the might be some transferability to Windows.

The steps necessary (general, independent of Linux or Windows):

1. Get developer options
2. Install adb and fastboot utilities

3. Backup (optional)
4. Allow OEM (Original Equipment Manufacturer) bootloader unlocking
5. Boot into fastboot
6. Unlock bootloader
7. Install TWRP (Team Win Recovery Project)
8. Put ROMs, Gapps and Magisk in an SD card and clean & dirty flash as much as you like.

The difference between the two OS's is on how to perform the step 2 and step 6 may be marginally different also. Regarding step 2, Windows has an executable that lets you do everything at once while Linux has to be more paced and do everything separately. Regarding step 6, one is done in the terminal (wanna guess which?) and the other has a somewhat nice gui.

Maybe it wouldn't be such a bad idea do use a windows virtual machine for this actually... But oh well.

NOTE : If the phone can't transfer files to the pc, don't even start this! On linux MTP must be enabled and installed. Enabling can be done by simply pressing the pop up on the phone to MTP(Media Transfer Protocol) and installing on Linux:

```
1    sudo apt install mtp-tools libmtp mtpfs
```

ADB (Android Debugging Bridge) is the utility that allows a connection with the phone with a considerable degree of permissions, for debugging purposes.

After this, in order:

1 Enable Developer Options

Go to settings, system, tap 7 times in the build number of the phone.

2 Install ADB and fastboot utilities

In Linux, this is the hardest step! A list of the steps:

1. Go here to download the latest abd tool.
2. If the above doesn't download properly, is possible to get it from the apt repository from android-tools-adb. If it works, do one more thing to be installed system wide: change path to this adb tool by adding to .bashrc the following:
   export PATH=${PATH}:/home/joao/plataform-tools/adb
   , considering the extracted file was placed in my top user folder.
3. Run: sudo apt install android-tools-fastboot.
4. Now adb is "installed" and fastboot too.. If you want to make a FULL backup of your phone - worth it because TWRP will delete everything - do the following: **suicide** .

Obviously kidding, but this is uncertain territory. Now is necessary to be sure everything is 100%. The following set of procedures did work for me after trying TOO many things:

1. don't turn on usb debugging still and don't connect the phone yet to the pc: DELETE .android from home!
2. revoke all usb debugging authorizations just below that option (on phone)
3. kill adb server with adb kill-server
4. connect phone to pc, enable MTP on phone and then enable usb debbuging
5. start adb server on pc: adb start-server
6. with "adb devices" the output should now be "number device" instead of "number unauthorized"

3 Backup (optional)

If you still are with me, then we might be able to backup things now! The following will allow a complete backup to be placed in the Documents folder: (confirmation on phone required and a password for encryption can be used)

```
adb backup -apk -shared -all -f ~/Documents/backup.ab
```

To get everything back, the following can be done to restore the backup: (again confirmation needed and password requested if used before)

```
adb restore ~/Documents/backup.ab
```

4 Enable OEM Bootloader unlocking

Go to general settings again, developer options and activate: OEM unlocking and Advanced reboot.

5 Boot into fastboot

Reboot into bootloader/fastboot. Setting the advance reboot option makes this much easier. The other way would require booting into recovery and from recovery to fastboot.

6 Unlock Bootloader

Check if the fastboot is working:

```
sudo fastboot devices
```

If one shows up, cool.

Next, unlock bootloader with:

```
sudo fastboot oem unlock
```

Confirm on phone and then should be cool. Takes a while to boot back to the OS, but then all is fine.

7 Install TWRP

After booting to the OS, boot back to fastboot/bootloader with the restart method in number 5. It will be necessary to enable the previous options again.

When in fastboot, check if everything is fine

```
sudo fastboot oem device-info
```

On the line saying "Device unlocked" is should be "true". The other lines don't matter.

To install TWRP in the recovery partition, so that it runs when we try to recover, do:

```
sudo fastboot flash recovery recovery.img
```

8 Install custom ROMs, Gapps, Magisk and others

Now, put the ROMs you like in the SD card, Magisk and Gapps.

Some clarification on the above:

- ROMs can be downloaded by searching 'name_of_phone ROM
- Gapps is necessary with the given ROM doesn't come with google apps, like the playstore. I don't know if it happens always, but is very frequent. Can be downloaded here: opengapps.org. Note that is necessary to know which processor architecture your smartphone uses. Oneplus X is ARM.
- Magisk is the BEST way to rooting your phone and have full access to it inside the ROM. Can be downloaded here: magiskmanager.org and go to install magisk for non-rooted phones, which will give you a zip from github.

Some clarification on the differences between clean and dirty flash.
Clean flash is FORMATTING COMPLETELY EVERYTHING in TWRP, by doing Wipe-¿ format data and only after that installing something. This is only needed when flashing incompatible ROMs, which are all ROMs that are not direct patches from the previous one. ROMs for different versions of android (8/Oreo to 9/Pie) are not compatible and from 8 to 8.1 I have might doubts. Dirty flash is just erasing some cache and then flashing/installing. It can be done with simple programs that don't change all the things like installing a new ROM does.

Having an Oneplus X, a great developer for this phone is YumeMichi and he/she posts everything here: YumeMichi Repo.

11.5 Android Studio with Linux

The above section is related with this one because ADB is necessary to run apps in the phone quickly.

In practice, if "adb devices" detects a good "device", then is all good. this section is also related with the above because if you want to test on an older phone, let's say more than 3 years old, it may not be possible due to API constrains. Therefore, what I needed to do was installing a very stable custom ROM Omni, by YumeMichi, that can be found the repository mentioned above.

To get everything ready to be able to plug and unplug the phone from the pc and still be recognized by Android Studio is was necessary to disconnect the phone, put MTP has a default mode, revoke USB debugging authorizations, delete .android, open android studio (the current session can't have already tried to connect to the phone once or else it will lead to a adb loop), "adb kill-server", then connecting the phone WITHOUT usb debugging enabled, then enable usb debugging and wait a bit. During about 10 seconds wait, there can be a popup on the phone which represents great news. Mark as "always trust". Check "adb devices" and the devices should be "device" and not "unauthorized" or anything.

From here, android studio will almost certainly work.

From here on, I'll write EVERYTHING I learn about Android Studio. It is hard to structure the amount of random information I'll gather so my strategy will be not doing so. I'll just add titles with the objective task that I learned to perform and some related information.

Change text size

By adding:

```
1     android:textSize="25sp"
```

to activity_main.xml, near the part of "android:text="Hello World!" ", was possible to change the text size.

Additionally, by messing with some things on the design tab, layout window, was possible to adjust the text window to the text.

To end: WHAT'S THE DIFFERENCE BETWEEN SP AND DP?
One means scale-independent pixels and the other density-independent pixels. The density of pixels a screen has is know as dpi/ppi (dots per inch/ pixels per inch). Now the choice is: is better to disregard the various format rations of screens of their density, what changes more? The form factor changes a lot more and a lot more abruptly. So it influences the text much more and we should use sp for text almost every time. Also, the scale can change with the user given scale and be messed up. Using sp prevents that. (a better look at this topic wouldn't be a bad idea...): DP change from phone to phone, regarding the aspect ratio. SP shouldn't...right?

Java background

I've created a document with the essentials, very bare material of course, of Java. Additionally, I'll write in another google document the things I learn from experienced Java and Android programmers.

Git with Android Studio

This is one of the best things about Android Studio: you can go to VCS (Version Control System) and check your previous commits, commit and push, resolve merges, ect...)

11.6 Downloading videos from all over the web

The Linux way, with the terminal.

For this, a tool called youtube.dl.

```
pip install --upgradable youtube_dl
```

There are a TON of supported websites that can be consulted here.

Some usage examples:

- To downlaod description, metadata, annotations, subtitles and thumbnail

  ```
  youtube-dl --write-description --write-info-json --write-annotations --write-sub --write-thumbnail url
  ```

- for the best single file video and audio quality

  ```
  youtube-dl -f best url
  ```

  [language=bash, numbers = none]

- NOTE: the best quality may be really good... do you want 4k if you display if only FHD (1920x1080)? take the "best" out and choose the quality by number and just put it instead of the "best".

This topic came up because self english training required transcribing videos. From Ted is difficult to have the exact transcripts downloaded automatically, but they can be easily copied.

However, a good video player is key for this so that it becomes much easier to jump to different parts of the video.

11.7 MPV - The best video player

Download as usual, these will be some advices to work with it.

Check Manual for a complete and clear description of all commands.

For sound adjustments use 9 and 0, to decrease and increase the volume.

For previous and next frame, use **,** and **.** , respectively.

For controlling the speed use:

or {}. Backspace to revert the speed back to normal.

Ctrl + Shift + Left or Right to align subtitles in time.

To use subtitles is just needed to use the GUI.

All the rest is at the provided link.

11.8 Keybindings - Keyboard and Mouse

For the mouse: Configure mouse buttons to custom sets of commands - Use to detect only mouse button presses:

```
xev | grep -A2 ButtonPress
```

To detect anything, simply run 'xev' in the terminal.

Then do as the examples show, 'keydown Control_L', 'keyup', 'key z' for a short press. THEN: 'xbindkeys -p' to apply the .xbindkeysrc conf file.

And the Fn keys can't be used... Explanation here .

BEFORE : trying the same with the keyboard, like mapping the arrows to the middle of the keyboard, check Section 11.12.2 (clickable) .

For the keyboard, the answer is using xmodmap if what you really need it is chaning functionalities instead of adding them. If you need to add them, xbindkeys is the way to go. But I urge you, read 11.12.2 before doing anything, I've invested quite a bit of my life, speacially while I was the most busy, exploring no way out and very unefficient paths.

11.9 Linux Image Editor

If you want something like paint, don't bother. Use pixlr.com/editor/ .

11.10 Linux Video Editor & Instagram

Yes, one may use Instagram. If one does, one may need some tools. To resize photographs or videos, something that is needed in case one needs to upload multiple files at once, the kapwing resize tool does that perfectly.

Additionally, the general Kapwing Tool is very usefull for video and image editing. Quick and easy really.

11.11 Other Linux related stuff

- More than enough about bash here
- To download files from links: wget –user-agent="Mozilla" link
- To convert .ppt to .pdf with libreoffice installed:

```
libreoffice --headless --invisible --convert-to pdf *.ppt
```

- Matlab Continuous Distributions
- MATLAB Ubuntu fixNecessary for help displaying and no errors on terminal
- To search stuff on the terminal by name:
  find directory -name "filename", e.g: find . -name "error.png"
  Or add the following for files: find directory -type d -name "foldername"
- sudo rm /var/crash/* for removing error reports every time they appear, or else they will continue to appear every time the computer boots
- xdg-open . to open the file explorer in the terminal location. xdg-open helo.mp4 is will open with the default program the specified file.
- To create a File shortcut for Thunar, the beautiful file manager that XUBUNTU uses, visit this;

11.12 Linux Life Lessons

11.12.1 Wine and PlayOnLinux - Project: Kindle to PDF

Sometimes Linux is not ready for somethings. Or somethings are not ready for Linux. In either case, the things should be used in the correct place.

Lesson: Wine and PlayOnLinux are fun and are becoming increasingly less of a cancer, but for now... they still a very substantial cancer. So, if something is meant to be done in Windows, for the next 2 years, at least, do so.

It took me 10 minutes in windows to achieve what I couldn't after 4 hours in different days in Linux. Don't waste your life in stupid things like this.

The solution for achieve kindle to PDF convertion is below. Basicly, uses a older version of kindle to pc to download a file that is protected with a breakable cypher and is just needed to load a plug-in in a e-book reader called Calibre. Full instructrions are below and work flawlessly in windows. In Linux they don't: the plug-in loaded into Calibre is meant to run on windows, so it requires far more stuff and has too many windows dependencies, so it would take too much time to even have the possibility of making it work. Completly not worth the try.

Do this in windows and get the pdf after that.

Then you can use stuff like this to have a nice print out of it.

```
pdfnup --nup 2x1 --suffix test file.pdf
```

11.12.2 Keyboard keybindings

Know what you are looking for .

Simplify the problem.

Ask in Ubuntu Forums.

After more than a morning on this, the major difficulty with mapping keyboard keys is that most of them already have a mapping that completely interferes what any other function you want to give them! For instance, **Alt** is used when your mouse is f*cked and you need press menus in windows. **Control** is practically reserved for every application. Not all keys thought, but if you want to map the arrows to JKLI for instance, Ctrl is not a very good key to use because everytime you press an arrow it will count the Ctrl as a modifier of the arrow and jump a whole word, for instance. **Shift** similar. **Alt Gr** seemed to be the answer, but for some reason, it prints f*cked up characters when you try to use it... Combinations of these are out of question.

Then what other modifiers do you have that aren't being used, are close to your hands when you write and you never use them? **Caps Lock** ? Haven't you been paying attention to what I said: You have to remove the function that keypress has, or else it will still be identifier as a caps Lock.

Bottom line: if you can find a way of changing the event that a certain keypress sends, you can make it seem like that key is being pressed... For instance: when you press ctrl, you make it seem like you are presing Insert (that no one f*cking uses), then you can make combinations with Ctrl. **HOWEVER** you completely miss the ability of pressing control!

What you really want to do is changing the response to a set of key presses. If your keyboard is already taken, and most of those things don't serve any purpose, you want to change what happens when you press those things! **Alt Gr** is the perfect example! Do you know that when you press that cripple cousin of Alt that is Alt Gr, you end up with a modifier letter? Moreover, if you press Shift + AltGr you get yet another variation! **Here is the opportunity!**

You will change what happens when you press those combinations! Use **xmodmap** for this!

For knowing what you need to change, look this what for the key. The "h H h H" is just something that is uniquely in the line of the h key. Do this for every key you want to change.

```
1    xmodmap -pke | grep "h H h H"
```

From here the image below. You want to change that line because every key means what happens when it is pressed.

```
$ xmodmap -e "keycode <keycode> = <key1> <key2> <key3> <key4> <key5> <key6>"
```

```
<key1> ...              <Key>
<key2> ...        <Shift-Key>
<key3> ...          <Alt-Key>
<key4> ...    <Shift-Alt-Key>
<key5> ...        <AltGr-Key> (at least by Unity)
<key6> ... <Shift-AltGr-Key> (at least by Unity)
```

Step 2) Map the keys to your desire:

```
$ ##                      1 2 3        4        5      6
$ xmodmap -e "keycode 43 = h H NoSymbol NoSymbol Left   NoSymbol"
$ xmodmap -e "keycode 44 = j J NoSymbol NoSymbol Down   NoSymbol"
$ xmodmap -e "keycode 45 = k K NoSymbol NoSymbol Up     NoSymbol"
$ xmodmap -e "keycode 46 = l L NoSymbol NoSymbol Right  NoSymbol"
$ ##                      1          2          3         4        5
$ xmodmap -e "keycode 47 = odiaeresis Odiaeresis NoSymbol NoSymbol Home"
$ xmodmap -e "keycode 48 = adiaeresis Adiaeresis NoSymbol NoSymbol End"
```

The exact mappings can be different. I did something like:

```
joao@joaoPC:~$ xmodmap -e "keycode 44 = j J j J Left dead_hook dead_horn"
joao@joaoPC:~$ xmodmap -e "keycode 45 = k K k K Down ampersand kra"
```

```
joao@joaoPC:~$ xmodmap -e "keycode 46 = l L l L Right Lstroke lstroke"
joao@joaoPC:~$ xmodmap -e "keycode 31 = i I i I Up idotless rightarrow"
joao@joaoPC:~$ xmodmap -e "keycode 30 = u U u U Home downarrow uparrow"
joao@joaoPC:~$ xmodmap -e "keycode 32 = o O o O End oslash Oslash"
```

So, with AltGr JKLI I have the arrows and have the Home and End in U and O, respectively. And this way is not necessary to change any shortcuts anywhere, will work everywhere and is the fastest way possible of doing it.

The lesson to take away is that if something somewhat simple is already becoming a mess, you are probably asking the wrong questions and there is a much shorter and more pleasant path that you are not finding. Take a step back, look and think about the problem verbally... Ask a question in the Ubuntu forums because they sugest a load of different answers that most certainly will have your question!

One last thing: if you want to make everything "reboot proof", put everything into a script and run it at the beginning.

12 Database work - SQL

12.1 SQL commands

The basic syntax is:

SELECT {Column name } FROM {Table name} WHERE {condition}

From here, there are a lot of variety that can be added to be possible lot of flexibility.

There are also other types of commands, that I haven't used so I won't talk here. Search :)

12.2 Browsing Tool with Filters

To use filters while using DB browsing tool installed on linux from sqlitebrowser.org with:

```
1    sudo apt install sqlitebrowser
```

Consult their wiki on that, it can be found here .

Additionally, is possible to export the result of filters to a csv by selecting the table with the mouse, pasting it in the side window and selecting "Export", not forgetting to add the .csv extension to the file name.

One last important feature: is possible to import and merge tables: if there is a similar database with a table with the same name (and structure) is possible to open that database with the same program, press on the table and Export it to csv. Then when opening the database where we want to import a table or various rows to, we select import from the File menu and select the csv. Note that the columns names in that csv may be in the first row. If this happens, we must check the square box that says "columns in the first row" so that the table can be read appropriately. Then it will ask if we want to merge and the previous table with the same name as the imported table will have the new rows from the imported table. If we want to do some filtering, for instance, *if we just wanted to import certain rows* we can now filter the table, or we could've filtered the table that was imported before importing it.

13 Visual Studio Code: The Environment for Development

This whole document was created with LaTeX on Visual Studio Code.

I ran commands for installing the necessary LaTeX stuff, these can be seen below.

I installed vscode with the software center from the .deb package downloaded throught their website and then installed 2 extensions. The first extensions is Latex Workshop from the extensions market inside vscode. The second I can't remember which one was or even if it did something...

```
1     sudo add-apt-repository ppa:jonathonf/texlive
2     sudo apt update && sudo apt install texlive-full
```

In addition, I configured *Ctrl + .* for compilation opening every shortcut with *Ctrl + K + Ctrl + S* (ctrl + K then lift the K key and press S). However, with the dual window mode, just as I it *Ctrl + S* to save, it automatically compiles which is very pleasant. *Alt + Z* for word wrap is useful too. To finish is possible to have 2 windows side by side in the same vscode instance by right clicking the window on the top bar and selecting "Slit Left/Right".

Some useful things to know:

- F1 opens the command pallet. From there, type what you need :P
- Ctrl + Alt + V -¿ open on a window on the right the document that is being edited on the main window;
- Ctrl + Alt + H -¿ go to the same place on the LaTeXdocument as where the source is;
- You can commit and push and all that git stuff with vscode;
- Ctrl + Alt + A -¿ After selecting a word, adds it to user dictionary, therefore is not considered an error anymore.
- Ctrl + T + Ctrl + A -¿ Toggle Activity Side-bar (shows files, ect...)

14 GitHub

So far, what a pain. A globally used tool that takes so much to learn. Here's the short guide.

Crete or clone

It depends who starts it. If you start it, you have to create it. If someone else already created it, then clone.

In the creation part, it can be done in the terminal, with some kind of GUI/in an IDE (Visual Studio Code and Android Studio do this very nicely). Or it can be done on a Git website which is has been necessary for me, due to terminal problems.

Create on Website. Works all the times, then clone it.

How Git works

Other important files

They are: gitignore, README, license...

GitIgnore is made to make git ignore certain files in the directory, so that those aren't included in the repository. Examples of those files are the .log files generated by compilations, build auxiliary files, etc...

About README, there's a whole website about this practice HERE. Normally, markdown is used to do this and that is a fairly easy language that can be fully consulted in HERE.

Another very good guide done on GitHub is Cheatsheet.

Regarding the license, it should be included if the code is to be used by others. A good example of a classical license is the MIT License. The website Choose a License explains everything perfectly.

15 Interesting stuff and People

15.1 ArcXiv

It's pronounce "archive" because the X is read as a χ (Chi, the greek letter).

arxiv.org has more than 1.5 million papers, including the paper published by **Grigori Perelman** , the man who cracked the first millenium problem rejecting the one million euros price and won a Fields medal rejecting the medal as well because he didn't want fame:

"After 10 hours of attempted persuasion over two days, Ball gave up. Two weeks later, Perelman summed up the conversation as follows: "He proposed to me three alternatives: accept and come; accept and don't come, and we will send you the medal later; third, I don't accept the prize. From the very beginning, I told him I have chosen the third one ... [the prize] was completely irrelevant for me. Everybody understood that if the proof is correct, then no other recognition is needed."

Overall, a place to read freely about what is and what once was the state of the art in science.

15.2 The writings of IST president

He writes well. Very well. After hearing him once talk for less than 5 minutes I did see why he was president.

Some of his writings in Public, one of the most well known portuguese newspapers, can be found in publico.pt/autor/arlindo-ol
.

Just in case he gets retired or something, his name is **Arlindo Oliveira** .

15.3 YIFY/YST release group

A release group so famous for their quality content, speed of delivery and that due to their websites taken down somewhat simultaneously.

Most of movie related piracy would still have their names attached. In quite a few cases, they don't have anything to do with it anymore.

In conclusion, search [YSF][YIFY] when you are searching a movie, it will be found much much faster.

Likewise, searching for yify subtitles will give great results very fast as well.

A good way of organizing movies would be: movie, jpeg with credits for release (yify jpeg), subtitle for yify and torrent source. Then ship all this to the drive.

About formats, in essence BLU ¿ WEB ¿ everyhing else. You can read everything about them in the Wikipedia: Pirated movie release types.

15.4 Interesting links

- https://www.sciencedirect.com/browse/journals-and-books a website with many scientific articles

16 Books

Every book I find interesting, reasons why I want to read it and what I thought after reading it.

16.1 Emotional Inteligence - Daniel Goleman

16.2 The Digital Mind - Arlindo Oliveira

16.3 Inteligência Artificial - Arlindo Oliveira

16.4 12 Rules for Life: An Antidote to Chaos - Jordan Peterson

16.5 Maps of Meaning - Jordan Peterson

16.6 Enlightenment Now: The Case for Reason, Science, Humanism, and Progress - Steven Pinker

16.7 The Better Angels of Our Nature: Why Violence Has Declined - Steven Pinker

16.8 The Beginning of Infinite - David Deutsch

16.9 How We Know What Isn't So - Thomas Gilovic

Another similar to this one is "Thinking, Fast and Slow" by Daniel Kahneman and both of them are important scientists that evaluate to what extent we (humans) don't make the logical decisions every time and what other factors influence ours decision making. We are probably not making the most rational decision because of those factors...

17 A few lessons

17.1 Be professional & make up your mind

Make your decisions. Sometimes in life you have to know what you want to do to make effective choices! Don't let the time pass, don't be a fucking passive cunt that only watches a mess unroll in front of you. Be there, be conscious of the decisions you are making and the impact they'll have and make them anyway because if you don't, you'll just get a random result and you'll probably piss-off and disrespect the people around you that are dependent on your decisions.

17.2 Insure properly

Insurance is the best way of risk transferring. Ensure until the money you would get back is preferable to the contents of the package. The money you are spending in the first place is probably completely irrelevant to make sure you get out of the situation winning.

This lesson came from insuring the package for 100 euro and paying for the shipping 33.5 euro while I could have insure it for 500 euro paying only 36. Of course the package got lost and I totally regret because it had important things that costed much more than that and that I valued much much more than that. Also I am in a phase I would like to buy a 500 euro headphones for motor cortex stimulation during workouts/trainings and I just sold a huge amount of hours of work in notes in TU Delft for 100 euro. Looking back, 500 euro wouldn't even be enough.

Be fucking sure you value your time properly.

17.3 Read

...