Abbreviations

CLT - Central Limit Theorem

1

Background

In this section we present the theorectical concepts needed in order to understand this work.

1.1 Bayesian Probability

The Probability theory has its roots in the 16^{th} century with attempts to analyse games of chance by Cardano. It is not hard to understand why games of chance are in the foundations of the probability theory, as throughout History the concept of probability has fascinated the Human being. Luck, fate were words that reflect things we feel we have no control upon and are associated with those games, and almost paradoxically we evolved in a way that we do not feel comfortable around them.

1.1.1 Kolmogorov axioms

In spite of the fact that the problem of games of chance kept attracting numerous mathematicians (with some of the most influential ones being Fermat, Pascal and Laplace), it was not until the 20^{th} century that the Russian mathematician Kolmogorov laid the foundations of the modern probability theory (first published in 1933) introducing three axioms [1]:

1. The probability of an event is a non-negative real number:

   $$P(A) \in \mathbb{R} \ \wedge \ P(A) \geq 0 \tag{1.1}$$

   This number represents the likelihood of that event happening, the greater the probability the more certain is its associated outcome.

2. The sum of probabilities of all possible outcomes in a space is always 1 ($P(\Omega) = 1$). These first two axioms leave us with the corollary that probabilities are bounded:

   $$0 \leq P(A) \leq 1 \tag{1.2}$$

3. The probability of a sequence of pairwise disjoint events is the sum of the probabilitis of each of those events. A corollary of this axiom is:

   $$P(A_1 \vee A_2 \vee ... \vee A_n) = P(A_1) + P(A_2) + ... + P(A_n) \tag{1.3}$$

Another important corollary derived from the axioms is the addition law of probability:

$$P(A \vee B) = P(A) + P(B) - P(A \wedge B) \tag{1.4}$$

1.1.2 Conditional Probability

When some evidence is presented we have what we call conditional probability (or posterior probability). Conditional probability is represented as $P(A|B)$, that could be read as: the probability of A after evidence B is presented.

The product rule is used to calculate posterior probability:

$$P(A|B) = \frac{P(A \wedge B)}{P(B)} \tag{1.5}$$

1.1.3 Joint Distribution

When we need to deal with more than a one Boolean variable, the joint distribution is used to define events in terms of those variables. This distribution grows exponentially (for n variables, there are 2^n combinations), with the probability space of the various variables considered. If we consider the joint probability distribution between all the variables in a given domain we call this full joint probability.

Assuming that have three random variables X, Y and W representing medical inferences. X represents if a person has fever, x being the answer 1 (true) or 0 (false). Y represents if a person has been in a tropical region. W stands for if a person has headaches. Their joint distribution can be considered a vector with length 8 accounting for the multiple combinations of X, Y, and W. For example $P(101)$ represents the probability of having fever and seizures without having been in a tropical region. The notation $P(xyw)$ is equivalent to $P(x, y, w)$, and to $P(X = x \vee Y = y \vee W = w)$

The joint distribution can be calculated using conditional probabilities by the chain rule [2]. Given a set of variables $x_1, x_2, ..., x_n$:

$$P(x_1, ..., x_n) = P(x_1) \prod_{i=2}^{n} P(x_i | x_{i-1}, ..., x_1) \tag{1.6}$$

For the last example a chain rule to calculate the joint probability would be:

$$P(x, y, w) = P(x|y, w).P(y|w).P(w) \tag{1.7}$$

To calculate P(x) from a joint distribution we need the marginal distribution $P_X : x \rightarrow \sum_{w \in W} \sum_{y \in Y} P(x, y, w)$. This would require 2^{n-1} sums.

1.1.4 Conditional Independence

When variables are independent they are uncorrelated and their marginal distributions are equal to their prior probability distribution [3]. For example:

$$P_X(x) = \sum_{w \in W} \sum_{y \in Y} P(x, y, w) = P(x) \tag{1.8}$$

So, two variables are independent iif [4] [2]:

$$P(X|W) = P(X) \tag{1.9}$$

That means if we are presented with three variables X, Y and W that are conditional independent relatively to each other, their joint distribution would only account their probability prior probability functions, thus simplifying the chain rule (Equation (1.7)):

$$P(x, y, w) = P(x).P(y).P(w) \tag{1.10}$$

Although it isn't always possible to decouple variables and assume that they are conditionally independent, doing so is a way to counter the "curse of dimensionality".

1.1.5 Markov Chains

Markov Chains define a system in terms of states and the probabilistic transitions from one state to another.

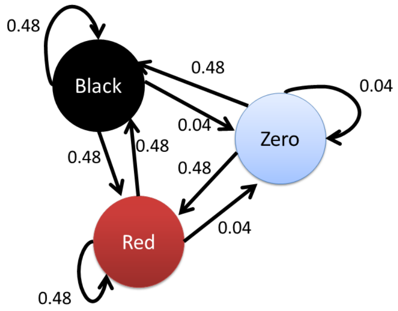


Figure 1.1: Markov Chain of a perspective on a roulette.

In 1913, at Monte Carlo Casino (Monaco), black came up twenty-six times in succession in roulette. Knowing that the probability of the ball landing on a red or on a black house is approximately 0.48 (the zero is a neutral house), many a gambler lost enormous sums while betting red as the they believed that the roulette was ripe with red, given the history. Players didn't want to believe that this and insisted that the roulette was biased; this became known as the Gamblers fallacy.

In the Markov Chain represented on Figure 1.1, we can see that in the state black the probability of transitioning to red is the same as to stay on the state black.

Given a system represented by the states $\{x_0, x_1, ..., x_i, ..., x_n\}$, and considering p_{ij} the probability of being in the state j and transitioning to the state i, the mixed state vector 1.11, which represents the probabilities of the system in the that i to transition to the other states. .

$$\overrightarrow{x_i} = \{p_{0i}x_0, p_{1i}x_1, ..., p_{ii}x_i, ..., p_{ni}x_n\} \tag{1.11}$$

The law of total probability is verified as $\sum_{j=0}^{n} p_{ji} = 1$, by specifying the every transition we get a stochastic matrix P, named the Markov matrix.

To illustrate how to construct a Markov Chain we will pick up on the example of Figure 1.1.

In this simplification of the Roulette we have 3 states:

- Black (B);
- Red (R);
- Zero (0).

We indifferently assign an index to each state, in order to construct the mixed state vector as in 1.11. Having the mixed state vectors defined the next step is to use them to create the stochastic matrix that has specified every transition 1.12.

$$R = \begin{bmatrix} p_{BB} & p_{BR} & p_{B0} \\ p_{RB} & p_{RR} & p_{R0} \\ p_{0B} & p_{0R} & p_{00} \end{bmatrix} = \begin{bmatrix} 0.48 & 0.48 & 0.04 \\ 0.48 & 0.48 & 0.04 \\ 0.48 & 0.48 & 0.04 \end{bmatrix} \tag{1.12}$$

1.2 Von Neumann Probability

In the beginning on the 20^{th} century the nature of light was on the spotlight of scientific investigation. The question whether light would be a particle (corpuscular theory), or a wave (undulatory theory), had been posed throughout History. Newton, notoriously, considered light to be a particle and presented arguments such as the fact that light travels in a straight like, not bending when presented with obstacles, unlike waves, and gave an interpretation of the diffraction mechanism by resorting to a special medium (aether), where the light corpuscles could create a localized wave [5].

The idea of light as a particle stood up until the 18^{th}century as many scientists (Robert Hooke, Christian Huygens and Leonhard Euler to name a few) tried to explain contradictions found in corpuscular theory. This brought back the idea that light behaves like a wave.

One of the most famous experiments that corroborates the undulatory theory is the Young's experiments $(19^{th}$ century), or the double-slit interferometer.

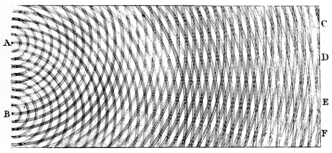


Figure 1.2: Thomas Young's sketch of two-slit diffraction of light.

The apparatus for the double-slit experiment can be seen in Figure 1.2. A light source is placed in such a way that "two portions" of light arrive at same time at the slits. Behind the barrier is a "wall" placed to intercept the light. The light captured at a wall will sport an interference pattern similar to the pattern when two waves interfere. The double-slit experiment was considered for a while the full stop on the discussion on the nature of light. However with experiments on the spectres of the light emitted by diverse substances and its relation with temperature, a new problem was posed.

The black body radiation problem was the theoretical problem where a body that absorbs light in all the electromagnetic spectrum, this makes the body acting as an natural vibrator, where each mode would have the same energy, according to the classic theory.

When a black body is at a determined temperature the frequency of the radiation it emits depends on the temperature. The classic theory predicted that most of the energy of the body would be in the high frequency part of the spectrum (violet part) where most modes would be found, this led to a prediction called the ultraviolet catastrophe. According to the classic theory the black body would emit radiation with an infinite power for temperatures above approximately $5000K$. Max Plank(1901), provided an explanation where the light was exchanged in discrete amounts called quanta, so that each frequency would only have specific levels of energy. Plank also determined through experimentation the value of the energy of the quanta that became known as photons later, that value became the physical constant called Plank constant:

$$h = 6.62606957(29) \times 10^{-34} J.s \tag{1.13}$$

In 1905, Einstein used the concept of quanta (photons) to explain the photoelectric effect. De Broglie(1924), suggested that all matter has a wave behaviour. This prediction was confirmed by studying the interference patterns caused by electron diffraction.

[0] Source: Young, Thomas: Probability. `http://en.wikipedia.org/wiki/File:Young_Diffraction.png(1803)`

In Quantum Mechanics, a particle is deemed as a system that is described by a wave-function (quantum system). Yhe wave function is usualy represented by the greek letter ψ. The main purpose of Quantum Mechanics is to give predictions for the outcomes derived from measurements on quantum systems [6].

1.2.1 Mathematical Foundations of Quantum Probability

As previously explained, Quantum Theory is a branch of physics that has evolved from the need to explain certain phenomena that could not be explained with the current classical theory. In the beginning of the 20th century Dirac and von Neumann helped to create the mathematical formalisms for this theory [7] [8].

Von Neumann's contributions revolved around the mathematical rigour that enabled to explain and manipulate the quantum phenomena. His framework is strongly based in Hilbert's theory of operators. Dirac's concerns were more of a practical nature, as his notation provided a compact way to represent quantum states. Their combined contributions were invaluable to establish this area.

A Hilbert Space is an extension of a vector space which requires the definition of inner product. A quantum system is represented by a n-dimensional complex Hilbert Space; a complex vector space in which the inner product is defined.

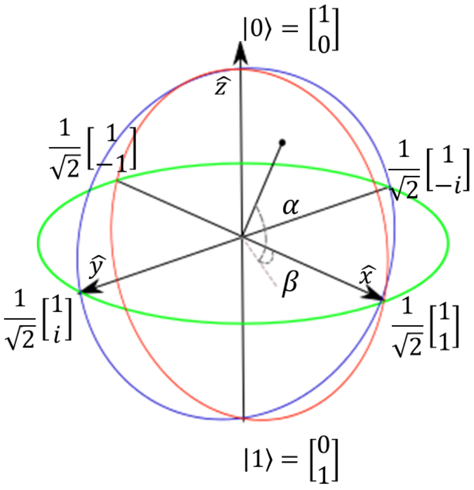

$|0\rangle = \begin{bmatrix}1\\0\end{bmatrix}$
$\hat{z}$
$\frac{1}{\sqrt{2}}\begin{bmatrix}1\\-1\end{bmatrix}$
$\frac{1}{\sqrt{2}}\begin{bmatrix}1\\-i\end{bmatrix}$
α
$\hat{y}$
β
$\hat{x}$
$\frac{1}{\sqrt{2}}\begin{bmatrix}1\\i\end{bmatrix}$
$\frac{1}{\sqrt{2}}\begin{bmatrix}1\\1\end{bmatrix}$
$|1\rangle = \begin{bmatrix}0\\1\end{bmatrix}$

Figure 1.3: Representation of a two-dimensional Hilbert Space($\mathcal{H}^2$)

A 2-dimensional Hilbert Space (corresponding to a qubit for example), can be represented by a Bloch Sphere. In the Figure 1.3 we have a representation of quantum state $|v\rangle$ in a two-dimensional Hilbert Space.

From Dirac it is important to point the Dirac's notation (also known as Bra-ket notation or $\langle Bra|c|ket\rangle$) (introduced in 1939 [7]), that is widely used in literature based on quantum theory. This notation uses angle brackets and vertical bars to represent quantum states (or abstract vectors) as it can be seen bellow:

$$|z\rangle = \begin{bmatrix} z_1 \\ z_2 \\ \ldots \\ z_n \end{bmatrix} \tag{1.14}$$

$$\langle z| = (|z\rangle)^* = \begin{bmatrix} z_1^* & z_2^* & ... & z_n^* \end{bmatrix} \tag{1.15}$$

This notation provides for an elegant representation of the inner product (1.16), and as the linearity that arises from the inner product, in equation (1.17). While the Bra-ket notation can be useful in terms of condensing information, using vectors and matrices to represent the states turns out to be a more approachable way to understand and manipulate data.

$$\langle z|z\rangle = \sum_{i=1}^{n} \bar{z}_i z_i \tag{1.16}$$

where $\bar{z_i}$ is the complex conjugate of z_i.

$$\langle z|(\alpha|x\rangle + \beta|y\rangle) = \alpha\langle z|x\rangle + \beta\langle z|y\rangle \tag{1.17}$$

1.2.2 Born rule

The Born rule was formulated by Born in 1926. This law allows to predict the probability that a measurement on a quantum system will yield a certain result. This law provides a link between the mathematical foundation of Quantum Mechanics and the experimental evidence [9] [10].

The Born rule states that if there is a system is in a state $|v\rangle$ (in a given n-dimensional Hilbert Space $\mathcal{H}$), and an Hermitian operator A is applied then the probability of measuring a specific eigenvalue λ_i associated with the i-th eigenvector of A (ψ_i), will be given by [9]:

$$P_v(\lambda_i) = \langle v|Proj_i|v\rangle \tag{1.18}$$

where $Proj_i$ is a projection matrix corresponding to ψ_i :

$$Proj_i = |\psi_i\rangle\langle\psi_i| \tag{1.19}$$

Given the properties of A, the set of eigenvectors $\{\psi_1, \psi_2, ..., \psi_i, ..., \psi_n\}$ forms a orthogonal basis of the n-dimensional Hilbert Space considered. Thus the state $|v\rangle$ can be written as a linear combination of the eigenvectors of A:

$$|v\rangle = \alpha_1\psi_1 + \alpha_2\psi_2 + ... + \alpha_i\psi_i + ... + \alpha_n\psi_n \tag{1.20}$$

The coefficients α_i are complex numbers called probability amplitudes, and their squared sum is equal to 1:

$$\sum_{i=0}^{n} |\alpha_i|^2 = 1 \tag{1.21}$$

this brings us to:

$$P_v(\lambda_i) = \langle v|\psi_i\rangle\langle\psi_i|v\rangle = |\langle v|\psi_i\rangle|^2 = |\alpha_i^*\alpha_i|^2 \tag{1.22}$$

So the determination of the probability of an event ($P(A)$), is made by projecting the quantum state on the eigenvectors corresponding to the operator A of the Hilbert Space and measuring the squared length of the projection. [11]

$$P(A) = (Proj_A|z\rangle)^2 \tag{1.23}$$

Applying an operator can be seen as applying a rotation matrix on the system ($|v\rangle$) and measuring the projection of $|v\rangle$ onto the imaginary axis and the real axis (Figure 1.3), or considering that we have a determined state vector and rotating the orthogonal basis of the Hilbert Space according to an operator and then to do a projection on the new chosen orthogonal basis.

According to Leiffer [12] “quantum theory can be thought of as a non-commutative, operator-valued, generalization of classical probability theory”.

As in the classical probability theory where from a random variable it is possible to establish a probability distribution, also known as density function, in the Hilbert space there is a equivalent density operator. The density operator (ρ) is a Hermitian operator that has the particularity of having its trace equal to 1 [9].

$$\rho = \sum_{i=0}^{n} \alpha_i |\psi_i\rangle\langle\psi_i| \tag{1.24}$$

$$tr(\rho) = 1 \tag{1.25}$$

1.2.3 Example of the double-slit experiment with electrons

Like the Young’s Experiment with light created an interference pattern similar to a wave, firing electrons one at the time produces a similar pattern. The unobserved fired electron behaved like a wave and after passing the slits the wavelets interfered with one another to create a interference pattern. However if a measuring device was active while the electron was fired the interference pattern wasn’t registered.

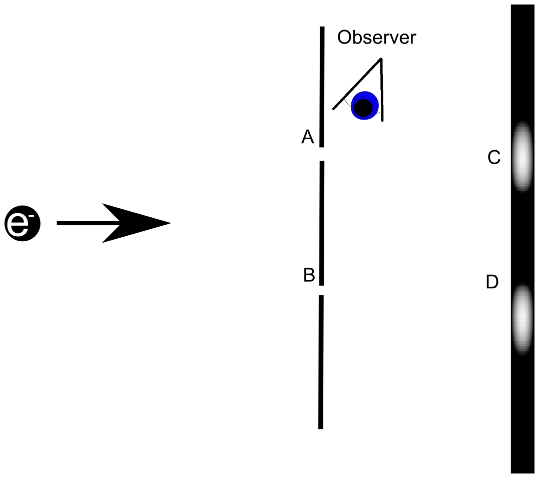


Figure 1.4: Double-slit experiment where there is a measuring device that allows to know through which slit the electron passed.

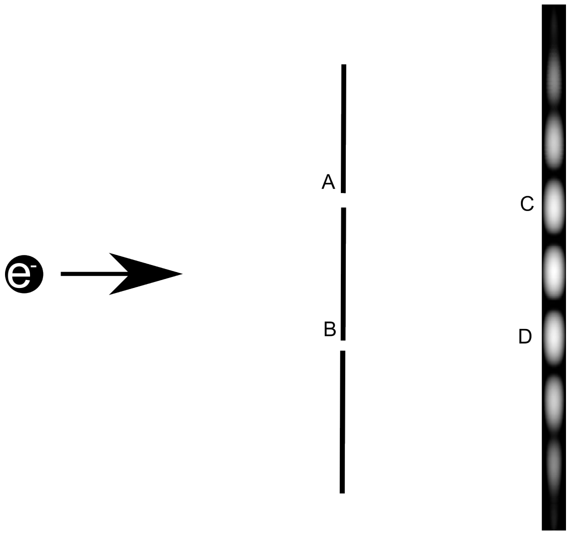


Figure 1.5: Double-slit experiment, where electrons exhibit the interference pattern characteristic in waves

The fact that the electron was measured while passing through a slit produced a particle behaviour, explained by the classical theory (Figure 1.4).

In this experiment a single electron is shoot at a time. So in the start of the experiment (S), we know the initial position of the electron.

A final measurement (F) is made when the electron hits the wall behind the slits, where we know the final position of the electron. [1]

If this experiment is observed, there is an intermediate measure that tells us whether the electron went through the slit A or B. The corresponding probability amplitudes related to this measurement are ω_A and ω_B, and:

$$\omega_A = \langle F|A\rangle\langle A|S\rangle \tag{1.26}$$

$$\omega_B = \langle F|B\rangle\langle B|S\rangle \tag{1.27}$$

If we consider the intermediate measurement the probability $P(F|S)$ will be:

$$P(F|S) = |\langle F|A\rangle\langle A|S\rangle|^2 + |\langle F|B\rangle\langle B|S\rangle|^2 \tag{1.28}$$

But if we only measure the position of the electron at the end of the experiment that probability will be:

$$P(F|S) = |\langle F|A\rangle\langle A|S\rangle + \langle F|B\rangle\langle B|S\rangle|^2 \tag{1.29}$$

The latter equation will be dependent on a interference coefficient that will be responsible the interference pattern observed in the unobserved experiment.

1.2.4 Example of the Polarization of Light

The photons in a beam of light don't vibrate all the same direction in most of the natural sources of light. To filter the light polaroids are used. A polaroid only allows the passage of light in a well-defined direction

[1] Mohrhoff, U.: Two Slits. `http://thisquantumworld.com/wp/the-mystique-of-quantum-mechanics/two-slit-experiment/#fn1back`

and thus reducing the intensity of the light. In the Figure 1.6 we can observe that the introduction of the oblique polaroid in the third situation led to a passage of light. Although there is a classical explanation to this phenomenon if we consider waves when we are considering a beam of light, if our light source emits one photon at the time a quantum mechanical explanation is needed [13].

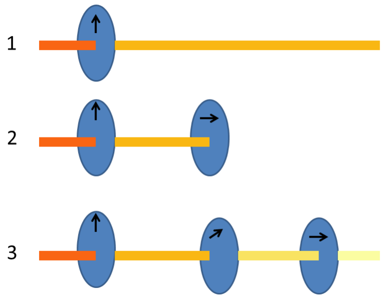


Figure 1.6: 1. With one vertical polaroid the unpolarized light is attenuated by a half. 2. Vertical polarization followed by a horizontal polarization will block all the passing light. 3. Inserting a oblique polaroid between the vertical and horizontal polaroids will allow light to pass.

To model the polarization of the photon in a quantum setting, we will use a vector $|v\rangle$ in a two-dimensional Hilbert Space:

$$|v\rangle = a\begin{bmatrix} 1 \\ 0 \end{bmatrix} + b\begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{1.30}$$

where $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$ would represent the vertical direction (could also be represented by the state vector $|\uparrow\rangle$), and $\begin{bmatrix} 0 \\ 1 \end{bmatrix}$ the horizontal one (another possible representation to this basis could be $|\rightarrow\rangle$). We can consider the Figure 1.3 as a graphical representation for this system.

In the first situation if $a = \frac{1}{\sqrt{2}}$, that would mean that the probability of passing the vertical polaroid would be $a = (\frac{1}{\sqrt{2}})^2 = 0.5$, that would light to the expected reduction of a half of the intensity of light. After passing through the vertical polaroid the photon will have a polarization of $|v\rangle = a\begin{bmatrix} 1 \\ 0 \end{bmatrix}$. Considering now the second situation after we have our photon polarized vertically (like in the end on the first situation), the probability of being vertically polarized is 1, thus making the probability of passing through the horizontal polaroid 0. In the third situation, after being vertically polarized the photon will pass through an oblique polaroid that makes its direction

$$|v\rangle = cos(\theta)\begin{bmatrix} 1 \\ 0 \end{bmatrix} + i.sin(\theta)\begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{1.31}$$

θ being the angle of the polaroid. The photon filtered by the vertical polaroid will pass this second polaroid with a probability of $(cos(\theta))^2$, becoming polarized according to the filter, as we can observe depending on the value of θ we will now have a horizontal component in the vector that describes the state of the photon. This will make the photon pass the horizontal polaroid with a probability of $sin(\theta)^2$.

1.3 Quantum Computing

Quantum Computing is an area that tries to take advantage of Quantum Mechanics phenomena to perform calculations. This idea of using the properties of a wave function to compute was first introduced by the works of Feynman [14]. Shor (at Bell Labs), developed an algorithm that, running on a quantum computer, could factor large numbers that can provide theoretical speed-ups over classical algorithms [13].

The quantum equivalent of a bit (the basic unit of information in computers), is a qubit (or quantum bit). A qubit is a two-state quantum system that can be interpreted as normalized vectors in a 2-dimensional Hilbert space. This Hilbert Space($\mathcal{H}^2$), an element in that space can be uniquely specified by resorting to a base ($\mathcal{B}$), with two orthonormal basis (also known as pure states). In quantum computing the base used is $\mathcal{B} = \{|0\rangle, |1\rangle\}$ or $\mathcal{B} = \{\begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}\}$.

1.3.1 Superposition

The qubit can be described in terms of linear transformations of pure states as in

$$|\psi\rangle = \omega_0|0\rangle + \omega_1|1\rangle = \begin{bmatrix} \omega_0 \\ \omega_1 \end{bmatrix} \tag{1.32}$$

where ω_0 and ω_1 are complex numbers called probability amplitudes, because when their module is squared they represent a probability or a probability density function associated with the system being in a certain state. When a system is in a mixture of pure states, meaning ω_0 and ω_1 are both different than 0, it is in a state known as a superposition. Measuring forces the system to collapse and assume one of the pure states with a certain probability.

In the example stated in the representation 1.32 the probability of the system falling into the state $|0\rangle$ would be $|\omega_0|^2$. The probability of the system falling into $|1\rangle$ would be $|\omega_1|^2$. The second axiom of Kolmogorov (law of total probability) is verified (1.33).

$$|\omega_0|^2 + |\omega_1|^2 = 1 \tag{1.33}$$

ω_0 and ω_1 (1.32) are complex numbers, the so-called probability amplitudes. When squared, the probability amplitude represents a probability.

1.3.2 Operators

In order to perform transformations in the qubit systems, we can define linear operators in the Hilbert space. One of the most important classes of operators being the self-adjoint operators, $A = A^*$, that have the property stated in Equation 1.34. The Hermitian operator is one that satisfies the property of being equal to its conjugate transposed, $A = A^{*T} = A^\dagger$. In a finite-dimensional Hilbert space defined by a set of orthonormal basis every self-adjoint operator is Hermitian.

$$\langle A^* z|x\rangle = \langle z|Ax\rangle \tag{1.34}$$

Unitary and Hermitian operators are used in the majority of quantum algorithms because they can be reversible, and because they insure that no rule of quantum mechanics is violated while applying the transformations. For that matter we arrive at an interesting difference between classical computing and quantum computing: it is impossible to copy or clone unknown quantum states [13]. This is known as: The No-Cloning Principle.

We can prove the "The No-Cloning Principle" by *reductio ad impossibilem.* Suppose we have an operator U which is unitary and clones quantum states, and two unknown quantum states $|a\rangle$ and $|b\rangle$. The transformation U means that if we apply it to $|a\rangle$ we have $U(|a\rangle|0\rangle) = |a\rangle|a\rangle$. The result when applying to $|b\rangle$ is $U(|b\rangle|0\rangle) = |b\rangle|b\rangle$.

If we consider a state $|c\rangle = \frac{1}{\sqrt{2}}(|a\rangle+|b\rangle)$, by the principle of linearity $U(|c\rangle|0\rangle) = \frac{1}{\sqrt{2}}(U(|a\rangle|0\rangle)+U(|b\rangle|0\rangle))$, giving the final result $U(|c\rangle|0\rangle) = \frac{1}{\sqrt{2}}(|a\rangle|a\rangle + |b\rangle|b\rangle)$. However if U is a cloning operator then $U(|c\rangle|0\rangle) = |c\rangle|c\rangle$.

$|c\rangle|c\rangle = \frac{1}{\sqrt{2}}(|a\rangle|a\rangle + |a\rangle|b\rangle + |b\rangle|a\rangle + |b\rangle|b\rangle)$ is different from $\frac{1}{\sqrt{2}}(|a\rangle|a\rangle + |b\rangle|b\rangle)$, thus we can affirm there is no unitary operator U that can clone unknown quantum states [13].

When presented with q qubits, a q-dimensional square matrix is called a Quantum Gate.

Some important Quantum Gates that operate on a single qubit are:

- The Identity Matrix (1.35).

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{1.35}$$

- The Pauli Operators. This operators are used to perform the NOT operation.
  - The Bit-Flip Operator

$$\delta_x = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \tag{1.36}$$

  - The Phase-Flip Operator

$$\sigma_z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \tag{1.37}$$

  - The Bit and Phase-Flip Operator

$$\sigma_y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \tag{1.38}$$

- The Hadamard Gate, belongs to a general class of Fourier Transforms. This 2×2 particular case is also a Discrete Fourier Transform matrix.

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{1.39}$$

The Pauli operators with the identity matrix I, form an orthogonal basis for the complex Hilbert space of all 2×2 matrices, also known as Special Unitary Group 2 - $\mathrm{SU}(2)$ [15].

1.3.3 Compound Systems

The representation of a system comprising multiple qubits grows exponentially. If to represent a single qubit system there is a 2-dimensional Hilbert space($\mathcal{H}^2$), to represent a system with m qubits a 2^m-

dimension space would be required. To represent a higher dimension multiple-qubit system composed by single-qubits, one can perform a tensor product of single-qubit systems.

The tensor product is an operation denoted by the symbol $\otimes$. Given two vector spaces V and W with basis (1.40) and (1.41) respectively, their tensor product would be the mn-dimensional vector space with a basis with elements from the set (1.42) [13].

$$A = \{|\alpha_1\rangle, |\alpha_2\rangle, ..., |\alpha_m\rangle\} \tag{1.40}$$

$$B = \{|\beta_1\rangle, |\beta_2\rangle, ..., |\beta_n\rangle\} \tag{1.41}$$

$$C = \{|\alpha_i\rangle \otimes |\beta_j\rangle\} \tag{1.42}$$

For example, if we consider two Hilbert spaces $\mathcal{H}^2$ with basis $A = \{|0\rangle, |1\rangle\}$ and $B = \{|-\rangle, |+\rangle\}$, their tensor product would be a $\mathcal{H}^4$ with basis (1.43).

$$AB = \{|0-\rangle, |0+\rangle, |1-\rangle, |1+\rangle\} \tag{1.43}$$

Now taking the former Hilbert space, supposing we have the qubits

$$|v\rangle = a_0|0\rangle + a_1|1\rangle \tag{1.44}$$

and

$$|w\rangle = b_0|-\rangle + b_1|+\rangle \tag{1.45}$$

then their tensor product would be

$$|v\rangle \otimes |w\rangle = a_0b_0|0-\rangle + a_0b_1|0+\rangle + a_1b_0|1-\rangle + a_1b_1|1+\rangle \tag{1.46}$$

The Bra-ket notation provides a way to prevent the escalation of the basis notation. When specified the vector space the basis can be specified in base 10 for simplicity sake. According to this the basis of the last example would be $AB = \{|0\rangle, |1\rangle, |2\rangle, |3\rangle\}$ and a system with 3 qubit could be represented in a Hilbert space, $\mathcal{H}^8$, with basis $\mathcal{B} = \{|0\rangle, |1\rangle, |2\rangle, |3\rangle, |4\rangle, |5\rangle, |6\rangle, |7\rangle\}$, however it is important to have the Hilbert dimension (the cardinal of $\mathcal{B}$), in mind.

When working with a compound system wich holds n particles represented by the quantum state $|\psi\rangle$ we often want to be able to separate its components $\psi_0, \psi_1, ..\psi_{n-1}$. This way we use the following notations as equivalent $|\psi\rangle = |\psi_0, \psi_1, ...\psi_{n-1}\rangle = |\psi_0\rangle|\psi_1\rangle...|\psi_{n-1}\rangle = |\psi_0\rangle \otimes |\psi_1\rangle \otimes ... \otimes |\psi_{n-1}$ [13]

1.3.4 Entanglement

In multiple qubit systems, qubits can interfere with each other, thus making impossible to determine the state of part of the system without "disturbing" the whole. In other words, there are states in a multi-qubit system that cannot be described as a probabilistic mixture of the tensor product of single-qubit systems; when this happens this state is not separable (with respect to the tensor product decomposition), this phenomenon is called quantum entanglement [13]. If a mixed quantum state ψ in a quantum system constructed by $V_1, V_2, ..., V_n$ is separable it can be written as:

$$\rho = \sum_{j=1}^{m} p_j |\varphi_j^1\rangle\langle\varphi_j^1| \otimes ... \otimes |\varphi_j^n\rangle\langle\varphi_j^n|, \sum_i p_i = 1, |\varphi_j^i\rangle \in V_i \tag{1.47}$$

Quantum entanglement is one of the main differences from the classical theory [13]. In an entangled system each member has a quantum state that is described with relation to the other members, thus cannot being described independently. This property is not local as transformations that act separately in different parts of an entangled system cannot break the entanglement. However, if we measure a part of the entangled system, the system collapses, and if we measure the otherpart in any point of time from that moment we will find a correlation with the outcome of the first measurement.

For example, supposing we consider the following quantum states, known as Bell states:

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|0\rangle_A \otimes |0\rangle_B + |1\rangle_A \otimes |1\rangle_B) = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \tag{1.48}$$

$$|\Phi^-\rangle = \frac{1}{\sqrt{2}}(|0\rangle_A \otimes |0\rangle_B - |1\rangle_A \otimes |1\rangle_B) = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle) \tag{1.49}$$

$$|\Psi^+\rangle = \frac{1}{\sqrt{2}}(|0\rangle_A \otimes |1\rangle_B + |1\rangle_A \otimes |0\rangle_B) = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle) \tag{1.50}$$

$$|\Psi^-\rangle = \frac{1}{\sqrt{2}}(|0\rangle_A \otimes |1\rangle_B - |1\rangle_A \otimes |0\rangle_B) = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle) \tag{1.51}$$

These states form a particular basis in $\mathcal{H}^4$ as they are all entangled states and they are maximally entangled. If we have a system of two qubits in a Hilbert space in the mixed state $|\Phi^+\rangle$ and we measure the qubit A (by deciding the outcome 0 or 1 with a probability of 0.5 for each),we are automatically uncovering the value of the qubit B. In this state we know the second qubit will always yield the same value of the first measured qubit. The second value is correlated with the first one.

1.4 Game Theory

Game Theory did not exist as a field of own right before von Neumann published the article " On the Theory of Games of Strategy". This field deals mainly with the study of interactions between rational decision-makers [16]. Game Theory knows numerous applications in areas such as Economics, Political Science, Biology, and Artificial Intelligence.

1.4.1 Definition of a Game

A game (Γ), is a model of conflict between players characterized by [17] [18] [19]:

- A set of Players: $P = \{1, 2, .., N\}$.
- For each player there is set of Actions: X_i
- There are preferences , for each player, over the set of Actions.

The preferences in the previous definitions are defined with resort to the concept of utility, expected utility, or payoff derives from the theory that we can use real numbers (Equation 1.52), to model and represent the wants and needs of the players.

$$u : X \to \mathbb{R} \tag{1.52}$$

This simplified mathematical model allows to compare different states and rank preferred outcomes. If $u_i(x)$ is a utility function for player i, and $u_i(A) > u_i(B)$, that would mean that the player would strictly prefer A over B. The concept of expected utility is fundamental to analyse games, as a rational player would try to maximize her expected utility [16] [17] [20].

The options a player i has when choosing an action, when the outcome depends not only on their choice (but also on the actions taken by the other players), is referred as strategy s_i. The set of strategies available to a player is represented by the set S_i.

When a strategy gives strictly a higher expected utility in comparison to other strategies, we have a strategy that dominates, or a dominant strategy. The mathematical definition of dominance is represented in Equation (1.53); where, for a player i, in spite of the strategies chosen by the other players (denoted by s_{-i}), there is a strategy s^* which gives always an higher expected utility in comparison with other strategies $s^{'}$ available to player i.

$$\forall s_{-i} \in S_{-i}[u_i(s^*, s_{-i}) > u_i(s^{'}, s_{-i})] \tag{1.53}$$

The strategy that leads to the most favourable outcome for a player, taking into account other players strategies, is known as best response.

A pure strategy defines deterministically how the player will play the game. In the game represented in Table1.1 , in a pure strategy, the players either they choose "Cooperate" (C), or "Defect" (D).

If there is a probability distribution associated with probability of playing with a determined pure strategy, we have a mixed strategy.

There are two standard representations of games:

- Normal Form - lists what payoffs the players get as a function of their actions as if they all make their moves simultaneously. This games are usually represented by a matrix, for example 1.1.
- Extensive Form - extensive form games can be represented by a tree and represent sequential actions.

	Player 2: C	Player 2: D
Player 1: C	(2,2)	(0,3)
Player 1: D	(3,0)	(1,1)

Table 1.1: Example of a Normal Form game.

A finite game is a game that has a finite set of actions, a finite number or players, and it does not go on indefinitely.

A zero-sum game is a mathematical representation of a system where the gains of the players are completely evened out by the losses of the others; this means that the sum of the utilities of all players will always be zero.

1.4.2 Nash Equilibrium

If we observe the representation of the classic game "Prisoner's Dilemma", we can observe that each player has two strategies; either they choose "Cooperate" (C), or "Defect" (D). The "Defect" strategy, for both players (the game is symmetrical), always yield a higher payoff in spite of the other player's strategy, this means that it is a dominant strategy and also constitutes a best response to the game. If both players chose their best response the final outcome (D, D) becomes an equilibrium solution, more specifically a Nash Equilibrium.

When all players cannot improve their utility by changing their strategy unilaterally, we have an equilibrium point. This equilibrium point is named after John Nash, who proved that it exists at least one mixed strategy Nash equilibrium in a finite game [21] [22]. This concept is used to analyse game where several decision makers interact simultaneously and the final outcome depends on the players strategy [17].

In order to compute a mixed strategy Nash Equilibrium in a 2-player 2 actions game, we need to define the probability distribution $P(C) = q$, $P(D) = 1 - q$ that makes player 1 indifferent to the whether player 2 decides to Cooperate or to Defect, and the probability distribution $P(C) = p$, $P(D) = 1 - p$ that makes player 2 indifferent to whether player 1 decides to Cooperate or to Defect. In the game represented in Table 1.2 the mixed strategies $(\frac{1}{3}, \frac{2}{3})$ and $(\frac{2}{3}, \frac{1}{3})$ are Nash Equilibria.

When there is at least a Nash Equilibrium when the players choose pure strategies we have a a strictly determined game [20]

	Player 2: C	Player 2: D
Player 1: C	(2,1)	(0,0)
Player 1: D	(0,0)	(1,2)

Table 1.2: Example of a Normal Form game. The mixed strategies $(\frac{1}{3}, \frac{2}{3})$ and $(\frac{2}{3}, \frac{1}{3})$ are Nash Equilibria for this game.

In an extensive form game we have a concept of sub-game perfect Nash equilibrium, when a strategy is a Nash equilibrium for all sub-games in the original game [20].

1.4.3 Prisoner's Dilemma

The Prisoner's Dilemma is a classic example of a game that can be represented in normal form [17]. This problem has received a great deal of attention because, in its simple form, rational individuals will seem to deviate from solutions that would represent the best social interest, the Pareto Optinal solution. In this game the Pareto Optimal solution is not a Nash Equilibrium. The Prisoner's Dilemma can be formulated as it follows:

> Two suspects of being partners in a crime are arrested. The police needs more evidence in order to prosecute the prisoners. So each prisoner is locked in solitary confinement and has no means of communicating with the other suspect. The police will then try to extort a confession from the prisoners. A bargain will be proposed to the suspects:
>
> - If the suspect testifies against the other suspect (Defects) and the other denies, he will go free and the second will get three years sentence.
> - If they both testify against one another (both Defect), both will be convicted and they will get two years.
> - In case both suspects deny the involvement (both Cooperate) of the other, they will get a one year sentence.

A matrix representation of the problem is in Table 1.3, here the payoff represented by the letter R would be the standard reward for the game, T would be the temptation to deviate from a cooperation profile, P represents the punishment when both entities do not cooperate, and finally S would be a sucker's payoff. A particular case for this problem is represented in Table 1.4. In each cell of the matrix we have a pair of the expected utility for the players for every outcome. A higher utility represents a more desirable state.

In this game the Pareto Optimal solution happens when both players chose to Cooperate (the pair (2,2) in Table 1.4). However both players have the incentive to Defect, because regardless what the opponent chooses they will have always a strictly higher payoff. Defecting becomes a dominant strategy in the Prisioner's Dillema and the outcome $(Defect, Defect)$ is a Nash Equilibrium to the game.

	Player 2: C	Player 2: D
Player 1: C	(R,R)	(S,T)
Player 1: D	(T,S)	(P,P)

Table 1.3: The canonical normal form representation for the Prisoner's Dilemma must respect $T > R > P > S$.

	Player 2: C	Player 2: D
Player 1: C	(2,2)	(0,3)
Player 1: D	(3,0)	(1,1)

Table 1.4: One possible normal form representation of Prisoner's Dilemma.

1.4.4 Pareto Optimal

From the point of view of an observer outside the game system some outcomes may seem better than others. For example on the game represented in Table 1.1 (Prisioners' Dilemma), the outcome (C, C) seems better than the outcome (D, D) because it provides a strictly higher utility to the players. However we know that the outcome (D, D) is the Nash Equilibrium of the game. If both players use their best response their outcome might not be the best outcome for both players.

When it is impossible to improve the payoff of a player without lowering another player's expected utility we have a Pareto Optimal solution.

For example if we have two players and 10 units of a finite resource to distribute among the players, a Pareto optimal solution is $(5, 5)$. Any other attempt to redistribute where the sum of the resource isn't 10 -$(1, 4)$, $(2, 3)$, etc...- could still be improved.

1.5 Quantum Game Theory

In the article “Quantum information approach to normal representation of extensive games” [23], the authors propose a representation for both normal and finite extensive form games [19]. This definition is based on the premise that any strategic game can be represented as an extensive form game where all the players have no knowledge about the actions taken by other players.

The representation assumes that for each action that a player can make in the game there are only two possible measurable outcomes. For example while voting on a referendum in which the question is ”Should the state declare war on country X?” despite the method using for voting the final answer is revealed as a ””yes”” or ””no””. In Quantum Game Theory we model an action as a qubit which is manipulated by the player.

A game in this form is represented by a six-tuple(1.54), where:

$$\Gamma = (\mathcal{H}^{2^a},\, N,\, |\psi_{in}\rangle,\, \xi,\, \{\mathcal{U}_j\},\, \{E_i\}) \tag{1.54}$$

- a is the number of actions (qubits), in the game;
- $\mathcal{H}^{2^a}$ is a 2^a-dimensional Hilbert space constructed as $\otimes_{j=1}^{a}\mathbb{C}^2$, with basis $\mathcal{B}$;
- N is the number of players;
- $|\psi_{in}\rangle$ is the initial state of the compound-system composed by a qubits: $|\varphi_1\rangle,\, |\varphi_2\rangle, ..., |\varphi_j\rangle, ..., |\varphi_a\rangle$;
- ξ is a mapping function that assigns each action to its respective player;
- For each qubit $j, \mathcal{U}_j$ is a subset of unitary operators from $\mathsf{SU}(2)$ (the general for the 2 dimensional Special Unitary Group is presented in Equation (1.55)). These operators can be used by the player to manipulate her qubit(s);

$$\begin{gathered}\mathcal{U}_j(w,x,y,z) = w.I + ix.\sigma_x + iy.\sigma_y + iz.\sigma_z,\\ w,x,y,z \in \mathbb{R} \wedge w^2 + x^2 + y^2 + z^2 = 1\end{gathered} \tag{1.55}$$

- Finally, for each player i, E_i is a utility functional that specifies her payoff. This is done by attributing a real number (representing a expected utility $u_i(b)$, in (1.56)), to the measurement for the projection of the final state ((1.57)), on a basis from the $\mathcal{B}$(1.56).

$$E_i = \sum_{b\in\mathcal{B}} u_i(b)|\langle b|\psi_{fin}\rangle|^2, u_i(b) \in \mathbb{R} \tag{1.56}$$

The strategy of a player i is a map τ_i which assigns a unitary operator U_j to every qubit j that is manipulated by the player ($j\in\, \xi^{-1}(i)$). The simultaneous move is represented in (1.57).

$$|\psi_{fin}\rangle = \otimes_{i=1}^{N} \otimes_{j\in\xi^{-1}(i)} \mathcal{U}_j|\psi_{in}\rangle \tag{1.57}$$

The tensor product of all the operators chosen by the players is referred as a super-operator, which act upon the game system.

$\bigotimes$	C-$\|0\rangle$	D-$\|1\rangle$
C-$\|0\rangle$	$\|0,0\rangle$	$\|0,1\rangle$
D-$\|1\rangle$	$\|1,0\rangle$	$\|1,1\rangle$

Table 1.5: Construction of the basis for the game space; $\mathcal{H}^4$.

1.5.1 Quantum Prisoner's Dilemma

The importance of the Prisoner's Dilemma for the study of Game Theory made it a prime target for investigation in Quantum Game Theory. The problem has been modelled several times [24] [15]. Therefore we will use it in order to exemplify and consolidate the definition described above, in Section 1.5 [23].

Each player i in the quantum version of Prisoner's Dilemma will be able to manipulate one qubit (φ_1 and φ_2) in Equations (1.58) and (1.59), with the unitary operators (shown in Equation (1.60)). The classical strategies: Cooperate (C), and Defect (D). C is represented in the sub-set $\mathcal{U}_j$ (Equation (1.61)) . D, also known as the Bit-flip operator (σ_x), is not represented in the restricted space proposed in [24] [23]. D^y is an alternative to D in the sub-set $\mathcal{U}$.

$$\varphi_1 = a_0|0\rangle + a_1|1\rangle, \sum_{i=0}^{1} |a_i|^2 = 1 \tag{1.58}$$

$$\varphi_2 = b_0|0\rangle + b_1|1\rangle, \sum_{j=0}^{1} |b_j|^2 = 1 \tag{1.59}$$

$$\mathcal{U}_j(\theta,\phi) = \begin{bmatrix} cos(\frac{\phi}{2}) & e^{i\phi}sin(\frac{\phi}{2}) \\ -e^{-i\phi}sin(\frac{\phi}{2}) & cos(\frac{\phi}{2}) \end{bmatrix}, j \in \{1,2\}, \theta \in (0,\pi), \phi \in (0,\frac{\pi}{2}) \tag{1.60}$$

$$\begin{cases} C = U_j(0,0) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \\ D = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \\ D^y = U_j(\pi,0) = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \end{cases}, j \in \{1,2\} \tag{1.61}$$

The system that holds the game is represented in a $\mathcal{H}^4$. Each basis ($|1\rangle, |2\rangle, |3\rangle, |4\rangle$), represents a final outcome as Table 1.5 suggests.

The fundamental difference from the classical version lies in the way the initial state is formulated in Equation and the strategies the players might use(1.63). We will entangle our state by applying the gate $\mathcal{J}$ [15] [24].

The parameter γ becomes a way to measure the entanglement in the system [24].

$$\mathcal{J} = exp\left\{i\frac{\gamma}{2}\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \otimes \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\right\} \tag{1.62}$$

$$\begin{aligned}|\psi_{in}(\gamma)\rangle &= exp\left\{i\frac{\gamma}{2}\left[\begin{array}{cc}0 & 1\\ 1 & 0\end{array}\right]\otimes\left[\begin{array}{cc}0 & 1\\ 1 & 0\end{array}\right]\right\}|00\rangle \\ &= cos(\frac{\gamma}{2})|00\rangle + isin(\frac{\gamma}{2})|11\rangle, \gamma \in (0,\pi)\end{aligned} \tag{1.63}$$

$$|\psi_{fin}\rangle = \mathcal{J}^{\dagger} \otimes_{i=1}^{2} \mathcal{U}_i|\psi_{in}\rangle \tag{1.64}$$

The entanglement in quantum game theory can be viewed as an intrinsic unbreakable contract. Furthermore measuring an entangled state will cause the wave function that describes the state to collapse. Before measuring the final result we will de-entangle the system by applying the operator $\mathcal{J}^{\dagger}$, as shown in Equation (1.64).

The utility functions for each player is calculated by projecting the final state in each base and attributing a real number to each measurement, as in equation. In order to compare a classical version with this quantum model, the real numbers assigned are those in the classical example in Table 1.3, and the basis are presented in Table 1.5.

$$E_0(|\psi_{fin}\rangle) = 2 \times |\langle 00|\psi_{fin}\rangle|^2 + 3 \times |\langle 10|\psi_{fin}\rangle|^2 + 1 \times |\langle 11|\psi_{fin}\rangle|^2 \tag{1.65}$$

$$E_1(|\psi_{fin}\rangle) = 2 \times |\langle 00|\psi_{fin}\rangle|^2 + 3 \times |\langle 01|\psi_{fin}\rangle|^2 + 1 \times |\langle 11|\psi_{fin}\rangle|^2 \tag{1.66}$$

The gate $\mathcal{J}$ is chosen to be commutative with the super-operators created by the tensor product of the classical strategies C and D: $[\mathcal{J}, C \otimes C] = 0$, $[\mathcal{J}, C \otimes D] = 0$, $[\mathcal{J}, D \otimes C] = 0$, and $[\mathcal{J}, D \otimes D] = 0$.

This condition implies that any pair of strategies in the sub-set $S_0 = \{\mathcal{U}(\theta, 0), \theta \in (0, \pi)\}$ is the equivalent of a classical mixed strategy when $\gamma = 0$; the joint probability associated with measuring a outcome $(\delta_1, \delta_2) \in \{(C,C),(C,D),(D,C),(D,D)\}$ ($P(\delta_1, \delta_2) = |\langle\delta_1, \delta_2|\psi_{fin}\rangle|^2$), becomes $P(\delta_1, \delta_2) = P(\delta_1)P(\delta_2)$, with $P(C) = cos^2(\frac{\theta}{2})$, and $P(D) = 1 - P(C)$ [24].

If the parameter ϕ in the operator $U_j(\theta, \phi)$ and the entanglement coefficient differ from 0 we are able to explore quantum strategies that have no counterpart in the classical domain.

[24] finds that the pair of strategies $(\mathcal{U}(0, \frac{\pi}{2}), \mathcal{U}(0, \frac{\pi}{2}))$ for $\gamma = \frac{\pi}{2}$ yield a payoff of $(2, 2)$, which is a Nash equilibrium and it is Pareto Optimal when we have a restricted space described by $\mathcal{U}$(1.60). If we allow the operator D(1.61), however, according to [15] there is no quantum pure strategy Nash Equilibrium, when the entanglement is maximal, $\mathcal{U}(\pi, 0)$ is the optimal counter-strategy for C (represented by the identity matrix), $\mathcal{U}(0, \frac{\pi}{2})$ is the optimal counter strategy for $\mathcal{U}(\pi, 0)$, D becomes the optimal counter-strategy for $\mathcal{U}(0, \frac{\pi}{2})$, and C becomes an optimal counter strategy for D [25].

1.6 Overview

In this chapter we provided a theoretical background that can help the reader to grasp the contents presented in the subsequent chapters.

We started by laying down a comparison between the classical probability theory and the quantum probability theory. The von Neumann probability is the mathematical foundation behind Quantum Mechanics.

The von Neumann probability differs from the classical mainly because mutually exclusive events can interfere, this happens in the Double-slit Experiment, this means that the third axiom of Kolmogorov does not hold true in this probability theory. The concept of probability is deeply intertwined with Quantum Mechanics.

In the Quantum Computing we presented the fundamental concepts. The book "'Quantum Computing - A Gentle Introduction" [13] provides a more in-depth resource on this subject; including the Shor's algorithm that is used to find prime factorizations, and the Deutsch-Jozsa algorithm, that with a single query decides if an unknown function is constant or balanced. These algorithms fall outside the scope of this document.

The Game Theory section contains a brief description of some fundamental concepts from Game Theory needed in order to understand this work. A comprehensive reference such as [17] or [20] can be consulted for a deeper insight on the domain.

2

Related Work

In this section we present relevant work that it is related with the problem. This constitutes an in-depth analisys of some important problems but also tries to present a broad representative view on the path towards building our solution.

2.1 Quantum Walk on a Line

Quantum Walk is the quantum version of random walks, which are mathematical formalisms that describe a path composed of random steps. A Markov Chain might be used to describe these processes.

We can define the Discrete Quantum Walk on a Line as a series of Left/Right decisions. Understanding this algorithm is important towards being able to define and design more complex algorithms that make use of quantum properties.

We followed an approach suggested by [26] towards simulating n-steps of a Quantum Walk on a Line. The Matlab algorithm can be consulted on Appendix A. In a discrete quantum walk in a line we want to preserve the fact that the probability of turning left is equal to the probability of turning right. To represent a state in this algorithm we will need the number of the node and a direction (identified as L,R) (2.1).

$$|\psi\rangle = |n, L\rangle \tag{2.1}$$

With two equally possible direction choices in each step, we can use a coin metaphor [26] [27] to approach the decision. We toss a coin and go either Left or Right depending on the result.

In a quantum version we need to define a Coin Operator (Coin Matrix), which is responsible to imprint a direction to the current state. This operator is a unitary matrix in a 2-dimension Hilbert space. Some examples of Coin Operators are the Hadamard matrix(2.2) and a symmetric unitary matrix (2.3).

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{2.2}$$

$$\begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{i}{\sqrt{2}} \\ \frac{i}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} \tag{2.3}$$

Taking the Hadamard matrix as an example(2.2), the coin matrix will operate on the state in the following way (2.4)(2.5) [26].

$$C|n, L\rangle = \frac{1}{\sqrt{2}}|n, L\rangle + \frac{1}{\sqrt{2}}|n, R\rangle \tag{2.4}$$

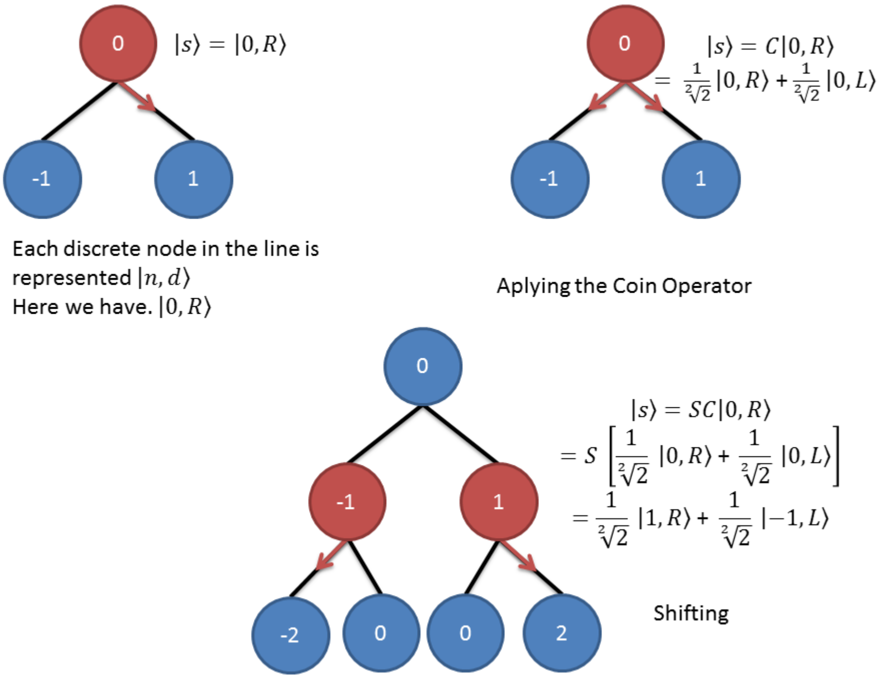
$$C|n, R\rangle = \frac{1}{\sqrt{2}}|n, L\rangle - \frac{1}{\sqrt{2}}|n, R\rangle \tag{2.5}$$

The Coin Matrix obtains its name by being the quantum equivalent of flipping a classic coin. After tossing a coin comes an operator that will move the node in the direction assigned. The operator responsible for this modification is commonly referred as Shift Operator (2.6)(2.7).

$$S|n,L\rangle = \frac{1}{\sqrt{2}}|n-1,L\rangle \tag{2.6}$$

$$S|n,R\rangle = \frac{1}{\sqrt{2}}|n+1,R\rangle \tag{2.7}$$

These matrices (Coin Matrix and Shift Operator) are used conceptually in various algorithms [13], therefore it is important to be familiar with them. A single step of the algorithm A is illustrated in Figure 2.1.



0
-1
1
$|s\rangle = |0,R\rangle$
Each discrete node in the line is represented $|n,d\rangle$
Here we have. $|0,R\rangle$
0
-1
1
$|s\rangle = C|0,R\rangle$
$= \frac{1}{\sqrt[2]{2}}|0,R\rangle + \frac{1}{\sqrt[2]{2}}|0,L\rangle$
Aplying the Coin Operator
0
-1
1
-2
0
0
2
$|s\rangle = SC|0,R\rangle$
$= S\left[\frac{1}{\sqrt[2]{2}}|0,R\rangle + \frac{1}{\sqrt[2]{2}}|0,L\rangle\right]$
$= \frac{1}{\sqrt[2]{2}}|1,R\rangle + \frac{1}{\sqrt[2]{2}}|-1,L\rangle$
Shifting

Figure 2.1: Simulating a step of a discrete quantum walk on a line. In the beginning we have a state characterized by the position (0) and a direction (either Left or Right).

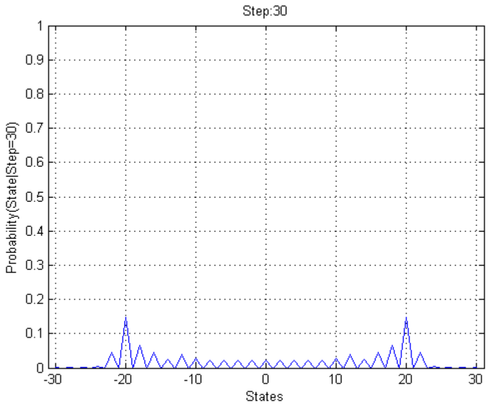


Figure 2.2: 30 Step of the Simulation A using Matrix (2.2) as a Coin Operator.

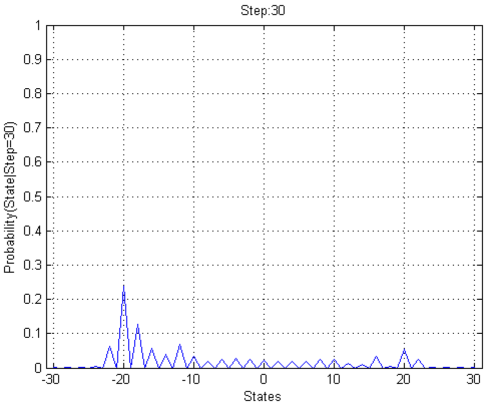


Figure 2.3: 30 Step of the Simulation A using a Hadamard Matrix (2.2) as a Coin Operator.

If we took the classical approach in which we tossed a fair coin to decide to go either left or right, and after n-steps we measured the final node repeatedly, by the Central Limit Theorem (CLT) the final distribution would converge to a normal distribution. However, in the quantum approach, depending on the Coin Matrix we can get different distributions. In this simple problem we have the basis for some quantum algorithms.

In Figures 2.3 and 2.2, depending on the coin operator we get two different results. Despite that, on one moment the probability of shifting left, is always equal to the probability of shifting right. The main difference from the classical approach lies on the fact there is a quantum interference during the walk, when we measure the results after N steps, somewhat like the electrons interfere in the “double-slit experiment”, presented in Section 1.2.3.

2.2 Quantum Models

The rationale behind building a quantum version of a Game Theory and/or Statistics problem lays in bringing phenomena like quantum superposition, and entanglement into known frameworks. Converting known classical problems into quantum games is relevant to the familiarize with the potential differences these models bring.

2.2.1 Quantum Roulette

In the arbitrary N-State quantum roulette, [28] presented a N-State roulette model using permutation matrices.

This model is interesting because in captures the usage of permutation matrices to manipulate and change the state of the system.

To verify this model with two players we developed a Matlab simulation C, that followed the steps taken in [28].

The game in represented in a N-Dimensional Hilbert Space. There is a basis in the space that represents each of the equally probable entries as shown in (2.8). In a sense this is a generalization of a quantum coin flip that is also used in Section 2.1.

$$|1\rangle = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, |2\rangle = \begin{bmatrix} 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \ldots, |N\rangle = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix} \tag{2.8}$$

Each state transition is obtained using a permutation matrix denoted by P^i. There are $N!$ permutation matrices, so in the particular case of having a 3-State roulette, there are 6 possible transition choices. The classical strategy considered will rely on choosing an arbitrary probability distribution, that verifies(2.10), and that maps the usage of the permutation matrices. This step will not affect the density matrix (ρ) of the roulette(2.9).

$$\rho = \frac{1}{N!} \sum_{i=0}^{N!-1} P^i \tag{2.9}$$

$$\sum_{i=0}^{N!-1} p_i = 1 \tag{2.10}$$

The density matrix is diagonalizable by a Discrete Fourier Transform because it is a kind of circulant matrix [29], as we can see in (2.11). In (2.11) λ_k are eigenvalues of ρ. $\lambda_1 = 1$ while $\lambda_2 = \lambda_3 = \lambda_k = \lambda_{N-1} = 0$. Each column i of the Fourier matrix will represent a eigenvector $|\lambda_i\rangle$. If we construct the diagonalizing matrix by rotating the columns of the Fourier Matrix we can obtain the projection states as in (2.12).

$$F^{\dagger}\rho F = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_{N-1} \end{bmatrix} \tag{2.11}$$

$$|1\rangle\langle 1| = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix} = F^{\dagger}\rho F \tag{2.12}$$

The quantum strategy advantage in this case is that the first player will not alter the density matrix(2.13).

$$\rho = \sum_{i=0}^{N!-1} p_i P^i \rho P^{i\dagger}, \sum_{i=0}^{N!-1} p_i = 1 \tag{2.13}$$

This means that if the second player knows the initial state and the first player plays with a classical strategy, thus never modifying the system density matrix, the second player will be able to manipulate the game under optimal conditions. This result is confirms the demonstration done in [30]; on Quantum Strategies, where in a classical 2 player zero-sum game, if one player adopts a quantum strategy, she increases her chances of winning the game.

2.2.2 Ultimatum Game

The ultimatum game is an example of an extensive form game where two players interact in order to divide a sum of money.

A finite amount of money (or other finite resource), is given to the players, and player 1 must propose how the money will be divided between the two players. If the second player agrees with the proposal, the resource will be split accordingly. When the player 2 rejects the proposal, neither player will receive the money.

If we consider that we have 100 coins, the number of coins received can be considered the expected utility associated with the proposal. The first player can either present a fair division (F), where the coins are split evenly, or an unfair division (U) game tree that represents the Ultimatum Game is shown in Figure 2.4.

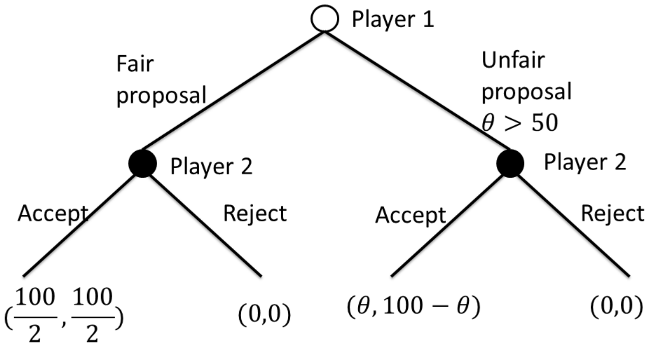


Figure 2.4: Ultimatum Game representation in the extensive form.

2.2.2.A Quantum Model

In a “Quantum information approach to the ultimatum game” [19] we are presented with a quantization scheme for the ultimatum game that uses the definition of quantum game in Section 1.5. They compare that approach to other quantum game definitions, and proposes and compares also quantum extensive form of the game.

If we present the game in Figure 2.4 in the normal form we get the matrix represented in Table 2.1. The player 2 has 4 possible strategies. The strategy $A_F R_U$ means that the player 2 will accept a fair division proposed by player 1 but will reject a unfair division.

The quantum game representation for this game is $\Gamma_{Ultimatum} = (\mathcal{H}^{2^3},\ 2,\ |\psi_{in}\rangle,\ \xi,\ \{\mathcal{U}_j\},\ \{E_i\})$. The game system will consist in 3 qubit, which correspond to the number of actions in the game. Player 1 will be able to manipulate the qubit 1, the player 2 can manipulate the remaining qubits.

	Player 2: $A_F A_U$	Player 2: $A_F R_U$	Player 2: $R_F A_U$	Player 2: $R_F R_U$
Player 1: F	$(50, 50)$	$(50, 50)$	$(0, 0)$	$(0, 0)$
Player 1: U	$(\theta, 100 - \theta)$	$(0, 0)$	$(\theta, 100 - \theta)$	$(0, 0)$

Table 2.1: Normal form representation of the ultimatum game.

The extensive form approach in [19] allows the differentiation between simultaneous moves and sequential moves. This is accomplished by measuring the game state in order to separate game stages. This ”Sequential procedure” uses the Lüders Rule, a quantum analogous of conditional probability.

The main points discussed in [19] Quantum Ultimatum approach are that the game definition presented in Section 1.5 makes the game more convenient to analyse than the extensive form approach.

2.3 Overview

There are more examples of games that have attracted interest in the Quantum domain. In this overview we are going to present a general picture of the work already done in this field.

For example various Models have been proposed to describe a quantum version of the Monty Hall problem [31] [32]. This popular problem [33] is known for its counter-intuitiveness. The problem can be posed as a contest where the player must choose a door (from a set of 3), and has $\frac{1}{3}$ of probability of getting a prize. After the player has chosen the door, one of the remaining 2 doors which does not have a prize is opened. The contestant is asked whether is to her advantage to switch her initial choice.

As the host reveals information, the initial set-up is modified. This is an interesting property. Despite being a counter-intuitive problem, a quantum approach to this problem allows an in-depth comparison between the classical measurement and the quantum measurement. The classic Monty Hall problem is modelled using conditional probability and Baye's Rule to learn that it is to the advantage of the contestant to switch doors. In the quantum version, measuring the outcome of the final state yields the result, instead of taking into account the intermediate actions [19]. In [34] we can observe the attempt to stick as closely to the classical formulation as possible, the host has a system that is correlated to the game system.

The principal information taken from this problems is that there is not a unique way to model a classical problem [34]. Therefore, when modelling a classical problem, we need to select properties that could potentially benefit from a quantum approach.

From the point of view of Quantum Cognition (a domain that seeks to introduce Quantum Mechanics Concepts in the field of Cognitive Sciences), these games are approached from the perspective of trying to model the mental state of the players in a quantum manner. The Prisoner's Dilemma is an example of a problem that has been modelled in order to explain discrepancies from the theoretical results of the classical Game Theory approach and the way humans play the game [35].

One last example worth mention is the quantum approach of the Stackelberg Duopoly problem [36] [37]. This is an Economics Game Theory model that seeks to represent the interactions of two companies, a market leader and a follower which play sequentially; the leader makes a decision and the follower responds.

Bibliography

[1] A. Hájek, "Interpretations of probability," in *The Stanford Encyclopedia of Philosophy*, summer 2012 ed., E. N. Zalta, Ed., 2012.

[2] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 2nd ed. Pearson Education, 2003.

[3] J. Pearl, *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2000. [Online]. Available: http://www.worldcat.org/isbn/0521773628

[4] W. Feller, *An Introduction to Probability Theory and Its Applications*. Wiley, January 1968, vol. 1. [Online]. Available: http://www.amazon.ca/exec/obidos/redirect?tag=citeulike04-20{&}path=ASIN/0471257087

[5] J. Dias de Deus, M. Pimenta, A. Noronha, T. Peña, and P. Brogueira, *Introdução à Física (in portuguese)*, ser. Colecção ciência e técnico. Lisboa: McGraw-Hill, 1992.

[6] H. Nikoli, "Quantum mechanics : Myths and facts," *Foundation of Physics*, vol. 37, pp. 1563–1611, 2007.

[7] F. Kronz and T. Lupher, "Quantum theory: von neumann vs. dirac," in *The Stanford Encyclopedia of Philosophy*, summer 2012 ed., E. N. Zalta, Ed., 2012.

[8] S. J. Summers, "Quantum Probability Theory," pp. 1–29, 2006.

[9] C. J. van Rijsbergen, *The Geometry of Information Retrieval*. Cambridge: Cambridge University Press, 2004. [Online]. Available: http://ebooks.cambridge.org/ref/id/CBO9780511543333

[10] N. Landsman, "Algebraic quantum mechanics," in *Compendium of Quantum Physics*, 2009, vol. 1, no. 2, ch. The Born r, pp. 6–9. [Online]. Available: http://www.math.ru.nl/~landsman/http://www.springerlink.com/index/L0M12Q6075H014RJ.pdf

[11] J. S. Trueblood and J. R. Busemeyer, "Quantum Information Processing Theory," pp. 1–11.

[12] M. Leifer and D. Poulin, "Quantum graphical models and belief propagation," *Annals of Physics*, 2008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0003491607001509

[13] E. Rieffel and W. Polak, *Quantum Computing, A Gentle Approach*. Cambridge: Massachusetts Institute of Technology Press, 2011.

[14] R. Feynman, "Simulating physics with computers," *International Journal of Theoretical Physics*, vol. 21, no. 6-7, pp. 467–488, 1982. [Online]. Available: http://dx.doi.org/10.1007/BF02650179

[15] J. Du, X. Xu, H. Li, X. Zhou, and R. Han, "Playing prisoner's dilemma with quantum rules," *Fluctuation and Noise Letters*, vol. 2, no. 4, 2002. [Online]. Available: http://www.worldscientific.com/doi/pdf/10.1142/S0219477502000993

[16] J. v. Neumann and O. Morgenstern, *Theory of Games and Economic Behavior*. Princeton University Press, 1944.

[17] M. Osborne, *Theory of Games and Economic Behavior*. Oxford University Press, 2004.

[18] M. Osborne and A. Rubinstein, *A Course in Game Theory*. MIT Press, 1994.

[19] P. Fra, “Quantum information approach to the ultimatum game,” pp. 1–14, 2011.

[20] K. Leyton-Brown and Y. Shoham, *Essentials of Game Theory: A Concise Multidisciplinary Introduction*. Morgan and Claypool Publishers, 2008.

[21] J. Nash, “Equilibrium points in n-person games,” *Proceedings of the National Academy of Sciences of the United States of America*, vol. 36, no. 1, pp. 48–49, January 1950.

[22] ——, “Non-cooperative games,” 1951.

[23] P. Fra, “Quantum information approach to normal representation of extensive games,” no. 1, pp. 1–14, 2011.

[24] J. Eisert, M. Wilkens, and M. Lewenstein, “Quantum games and quantum strategies,” *arXiv preprint quant-ph/9806088*, vol. 2, no. 1, pp. 2–5, 1998. [Online]. Available: http://arxiv.org/abs/quant-ph/9806088

[25] J. Du, H. Li, X. Xu, X. Zhou, and R. Han, “Phase-transition-like behaviour of quantum games,” *Journal of Physics A: . . .*, 2003. [Online]. Available: http://iopscience.iop.org/0305-4470/36/23/318

[26] A. Ambainis and J. Kempe, “Quantum Walks on Graphs.”

[27] A. Ambainis, “Quantum walks and their algorithmic applications,” pp. 1–11, 2008.

[28] S. Salimi and M. M. Soltanzadeh, “Investigation of quantum roulette arXiv : 0807 . 3142v3 [quant-ph] 30 Apr 2009,” 2009.

[29] P. J. Davis, “Circulant Matrices,” 1994.

[30] D. A. Meyer, “Quantum Strategies,” *Phisical Review Letters*, vol. 82, pp. 1052–1055, February 1999.

[31] R. D. Gill, M. Keyl, R. F. Werner, and H. Maassen, “The Quantum Monty Hall Problem,” pp. 1–8.

[32] A. P. Flitney and D. Abbott, “Quantum version of the Monty Hall problem,” pp. 1–5, 2008.

[33] M. vos Savant, “Ask Marilyn,” *Parade Magazine*, p. 16, September 1990.

[34] R. D. Gill, M. Keyl, R. F. Werner, and H. Maassen, “The Quantum Monty Hall Problem,” pp. 1–8, 2002.

[35] E. M. Pothos and J. R. Busemeyer, “A quantum probability explanation for violations of ’rational’ decision theory.” *Proceedings. Biological sciences / The Royal Society*, vol. 276, no. 1665, pp. 2171–8, Jun. 2009. [Online]. Available: http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2677606&tool=pmcentrez&rendertype=abstract

[36] S. Khan and M. K. Khan, “Quantum Stackelberg Duopoly in a Noninertial Frame,” *Chinese Physics Letters*, vol. 28, no. 7, July 2011. [Online]. Available: http://stacks.iop.org/0256-307X/28/i=7/a=070202?key=crossref.99b40803901ef6b332c1b5b78d9b9002

[37] A. Iqbal and A. H. Toor, “Backwards-induction outcome in a quantum game.” pp. 1–16, 2008.

A

Matlab Simulation: Discrete Quantum Walk on a Line

```


%-- number of steps in the simulation
steps= 30;

%-- hadamard matrix
H = [1/sqrt(2) 1/sqrt(2); 1/sqrt(2) -1/sqrt(2) ];
%-- symetric matrix
M = [1/sqrt(2) 1i/sqrt(2); 1i/sqrt(2) 1/sqrt(2) ];
%-- coin flip unitary operator
C = H;
C=M;

%-- coin flip matrix
CM= zeros((steps*2+1),2);
%-- shift matrix
SM= zeros(steps*2+1,2);

%-- middle index
i0= steps+1;

%-- initialize flip probability amplitudes (coin has 1/2 chance of going +1 or -1)
CM(i0,1)=1/sqrt(2);
CM(i0,2)=1/sqrt(2);

for i=1:steps
   %-- clean SM
   SM= zeros(steps*2+1,2);

   for j=1:(steps*2+1)
       if CM(j, 1)≠0
           SM(j-1,1)=CM(j, 1);
       end
       if CM(j, 2)≠0
           SM(j+1,2)=CM(j, 2);
       end
   end
   SM;
   %disp('-------');
   %-- clean CM
   CM= zeros(steps*2+1,2);
   for j=1:(steps*2+1)
       if SM(j, 1)≠0
           CM(j, 1)= CM(j, 1)+C(1,1)*SM(j, 1);
           CM(j, 2)= CM(j, 2)+C(1,2)*SM(j, 1);
       end
       if SM(j, 2)≠0
           CM(j, 1)= CM(j, 1)+C(2,1)*SM(j, 2);
           CM(j, 2)= CM(j, 2)+C(2,2)*SM(j, 2);
       end
   end
   CM;

   %Display

   figure

   probability=zeros(steps*2+1);

   for j=1:steps*2+1
      probability(j)=abs(SM(j,1)).^2+ abs(SM(j,2)).^2;
   end
   axisP = -steps : steps;
   plot(axisP,probability)
   title(strcat('Step: ' ,num2str(i)))
   axis([-steps-1 , steps+1, 0, 1]);
   grid on
   ylabel(strcat('Probability(State|Step=',num2str(i), ')'));
   xlabel('States');

   %Display
end
```

B

Quantum Prisioner’s Dillema

```
% Quantum Prisioner's Dillema
function out = quantumprisionersdillema(g,T,R,P,S,U1,U2)
%quantumprisionersdillema(g)
%
%                 Simulates the payoff of 2-player prisioner's dillema game
%
%         IN:
%                 g   : entanglement coefitient, by defaut g=0 (no
%                 entanglement)
%                 T   : temptation, by defaut T=3
%                 R   : reward, by defaut R=2
%                 P   : punishment, by defaut P=1
%                 S   : suckers, by defaut S=0
%                 U1   : Player 1 strategy
%                 U2   : Player 2 strategy
%
% To have a prisoner's dilemma game according to the cannonical form,
% it must respect:
% T > R > P > S
%             P      Player 2
%             l       C       D
%             a  C: (R, R)   (S, T)
%             y  D: (T, S)   (P, P)
%             e
%             r
%             1
%
%         OUT:
%                 out: 5x2 matrix, column 1 has the expected utility for
%                      player 1, column 2 has the expected utility for
%                      player 2. A possible outcome corresponds to each
%                      line like [CC;CD;DC;DD;U1U2]
%
% All parameters are optional
%------------------------%

%-- entanglement coeficient (checks if exists)
%-- Check variables and set to defaults
if exist('g','var')≠1, g=0; end

%-- Utility
%-- Check variables and set to defaults
if exist('T','var')≠1, T=3; end
if exist('R','var')≠1, R=2; end
if exist('P','var')≠1, P=1; end
if exist('S','var')≠1, S=0; end

%-- Strategies
if exist('U1','var')≠1, U1=[0 1i;1i 0]; end
if exist('U2','var')≠1, U2=[0 1i;1i 0]; end

%-- Actions
%   cooperate= [1 0;0 1]
C= eye(2);
%   defect= [0 1;1 0]
D= ones(2)-eye(2);

%-- Building the initial state
ini= cos(g)*kron([1 0]',[1 0]') + 1i*sin(g)*kron([0 1]',[0 1]');
%-- Deentangles to produce a final state
Jt = ctranspose( expm(1i*(g)*kron(D,D)));
out= zeros(5,2);
%-- Simulation a outcome where player1 (C)ooperates
%   and player 2 (C)ooperates
   finCC= Jt*kron(C,C)*ini;
   out(1,1)=payofffunc_player1(finCC, T, R, P, S);
   out(1,2)=payofffunc_player2(finCC, T, R, P, S);
%-- Simulation a outcome where player1 (C)ooperates
%   and player 2 (D)efects
   finCD= Jt*kron(C,D)*ini;
   out(2,1)=payofffunc_player1(finCD, T, R, P, S);
   out(2,2)=payofffunc_player2(finCD, T, R, P, S);
%-- Simulation a outcome where player1 (D)efects
```

```
%   and player 2 (C)ooperates
   finDC= Jt*kron(D,C)*ini;
   out(3,1)=payofffunc_player1(finDC, T, R, P, S);
   out(3,2)=payofffunc_player2(finDC, T, R, P, S);
%-- Simulation a outcome where player1 (D)efects
%   and player 2 (D)efects
   finDD= Jt*kron(D,D)*ini;
   out(4,1)=payofffunc_player1(finDD, T, R, P, S);
   out(4,2)=payofffunc_player2(finDD, T, R, P, S);
%-- Simulation a outcome where player1 plays U1
%   and player 2 U2
   finUU= Jt*kron(U1,U2)*ini;
   out(5,1)=payofffunc_player1(finUU, T, R, P, S);
   out(5,2)=payofffunc_player2(finUU, T, R, P, S);
end

function u = payofffunc_player1(fin, T, R, P, S)
%payofffunc_player1(fin)
%
%                 Calculates the payoff for player 1
%               IN
%                    fin: final state
%                    T, R, P, S: payoff numbers
%               OUT
%                    u: expected utility for player 1
%(norm(conj([0 1 1 1 0 1 1 1]').*fin2CC))^2
    u= R*(norm(conj([1 0 0 0]').*fin))^2 + T*(norm(conj([0 0 1 0]').*fin))^2+ ...
        P*(norm(conj([0 0 0 1]').*fin))^2;
end

function u = payofffunc_player2(fin, T, R, P, S)
%payofffunc_player1(fin)
%
%                 Calculates the payoff for player 2
%               IN
%                    fin: final state
%                    T, R, P, S: payoff numbers
%               OUT
%                    u: expected utility for player 2
    u= R*(norm(conj([1 0 0 0]').*fin))^2 + T*(norm(conj([0 1 0 0]').*fin))^2+ ...
        P*(norm(conj([0 0 0 1]').*fin))^2;
end
```

C

Quantum Roullete

```
function []= quantum_roullete3()
%%
% Simulation based on
% S. Salimi and M. M. Soltanzadeh, "Investigation of quantum roulette arXiv : 0807 . ...
    3142v3 [ quant−ph ] 30 Apr 2009," 2009.

%%

N=3
%% make states
I = eye(N);
D = (1/N)*ones(N);


%% permutation matrices change between states
 X0 = circshift(I, 0);
%X0 = I;
 X1 = circshift(I, 1);
%X1 = [0 1 0; 1 0 0; 0 0 1];
 X2 = circshift(I, 2);
%X2 = [0 0 1; 0 1 0; 1 0 0];
 X3 = circshift(I, 3);
%X3 = [1 0 0; 0 0 1; 0 1 0];
X4 = circshift(I, 4);
%X4 = [0 0 1 ; 1 0 0; 0 1 0];
 X5 = circshift(I, 5);
%X5 = [0 1 0; 0 0 1; 1 0 0];


%%Fourier Matrix
F= (1/sqrt(N))*fft(I)

T0 = circshift(F,[0 0]);
T1 = circshift(F,[0 1]);
T2 = circshift(F,[0 2]);
%%

%%Step1

%Assuming Alice places the roullete in state 2 were both player can see
Ro0= [0; 1; 0]*[0; 1; 0]'

%%Step2

%As Alice chose state 2, Bob will select T1 to rotate the state
Ro1 = T1 * Ro0 * T1'

%step 3 —
Ro1 = T2 * Ro1 * T2'

%%Step3
%Alice will play again
%

Ro2 = (2/6)*X0*Ro1*X0' + (1/12)*X1*Ro1*X1' +(1/12)*X2*Ro1*X2' +(1/6)*X3*Ro1*X3' ...
    +(1/6)*X4*Ro1*X4' +(1/6)*X5*Ro1*X5'

%%Step4
%Bob can choose which state he wants

Ro3 = T0'*Ro2*T0

Ro3 = T1'*Ro2*T1

Ro3 = T2'*Ro2*T2

Ro4 = (2/6)*X0*Ro3*X0' + (1/12)*X1*Ro3*X1' +(1/12)*X2*Ro3*X2' +(1/6)*X3*Ro3*X3' ...
    +(1/6)*X4*Ro3*X4' +(1/6)*X5*Ro3*X5'

Ro3 = T0'*Ro3*T0
end
```

D

Results: Pirate Game

D.1 2 Player Game

D.1.1 Simulation

```
% Quantum Pirate Game
function out = quantum2piratesanalise()
    C= eye(2);
%   defect= [0 1;1 0]
D= ones(2)-eye(2);

Dy= D;
Dy(2,1)=-1;
Dz= [1i 0;0 -1i];

g=pi/2;

u1=zeros(4,4);
u2=zeros(4,4);

for s1=1:1:4


    switch s1
    case 1
        U1=C;
    case 2
        U1=D;
    case 3
        U1=Dy;
    otherwise
        U1=Dz;
end
for s2=1:1:4

   switch s1
    case 1
        U2=C;
    case 2
        U2=D;
    case 3
        U2=Dy;
    otherwise
        U2=Dz;

   end

   out=quantum2pirates(g,U1,U2);
    u1(s1,s2)=out(1);
    u2(s1,s2)=out(2);

end

end

figure


h=bar3(u1);

% set(h(1),'facecolor','green');
% set(h(2),'facecolor','blue');
% set(h(3),'facecolor','red');
%mesh(prob);
%

set(gca,'yTickLabel',{'C', 'D',  'U(\pi ,0)', 'U(0, \pi /2)'})
% set(gca,'xTick',0:length(t)/4:length(t))
set(gca,'xTickLabel',{'C', 'D',  'U(\pi ,0)', 'U(0, \pi /2)'})
ylabel('U_2');
```

```
xlabel('U_1');
zlabel('E_1');

title('\gamma = \pi /2 Expected utility for player 1 (E_1)')

figure


h=bar3(u2);

% set(h(1),'facecolor','green');
% set(h(2),'facecolor','blue');
% set(h(3),'facecolor','red');
%mesh(prob);
%

set(gca,'yTickLabel',{'C', 'D',  'U(\pi ,0)', 'U(0, \pi /2)'})
% set(gca,'xTick',0:length(t)/4:length(t))
set(gca,'xTickLabel',{'C', 'D',  'U(\pi ,0)', 'U(0, \pi /2)'})
ylabel('U_2');
xlabel('U_1');
zlabel('E_2');

title('\gamma = \pi /2 Expected utility for player 2 (E_2)')

end

function out = quantum2pirates(g,U1,U2)
%quantumprisionersdillema(g)
%
%                 Simulates the payoff of 2-player prisioner's dillema game
%
%        IN:
%                 g   : entanglement coefitient, by defaut g=0 (no
%                 entanglement)
%
%                 U1   : Player 1 strategy
%                 U2   : Player 2 strategy
%
% All parameters are optional
%-------------------------%

%-- entanglement coeficient (checks if exists)
%-- Check variables and set to defaults
if exist('g','var')≠1, g=0; end

%-- Utility
%-- Check variables and set to defaults
a22=100;
a23=0;
death=-200;
incentive=0.5;


%-- Strategies
if exist('U1','var')≠1, U1=[0 1i;1i 0]; end
if exist('U2','var')≠1, U2=[0 1i;1i 0]; end

%-- Actions
%   cooperate= [1 0;0 1]
C= eye(2);
%   defect= [0 1;1 0]
D= ones(2)-eye(2);

%-- Building the initial state
ini= cos(g/2)*kron([1 0]',[1 0]') + 1i*sin(g/2)*kron([0 1]',[0 1]');
%-- Deentangles to produce a final state
Jt = ctranspose( expm(1i*(g/2)*kron(D,D)));

   fin= Jt*kron(U1,U2)*ini;
   u1=payofffunc_player1(fin, a22, a23, death, incentive);
   u2=payofffunc_player2(fin, a22, a23, death, incentive);
    out= [u1 u2];
```

```
end

function u = payofffunc_player1(fin, a22, a23, death, incentive)
%payofffunc_player1(fin)
%
%                 Calculates the payoff for player 1
%               IN
%                    fin: final state
%                    a22, a23, death, incentive: payoff coefficients
%               OUT
%                    u: expected utility for player 1
%(norm(conj([0 1 1 1 0 1 1 1]').*fin2CC))^2
    u= a22*(measure2([1 0 0 0]',fin)+measure2([0 1 0 0]',fin)+measure2([0 0 1 0]',fin) ...
        )−200*measure2([0 0 0 1]',fin) ;
end

function u = payofffunc_player2(fin, a22, a23, death, incentive)
%payofffunc_player1(fin)
%
%                 Calculates the payoff for player 2
%               IN
%                    fin: final state
%                    a22, a23, death, incentive: payoff coefficients
%               OUT
%                    u: expected utility for player 2
    u= a23*(measure2([1 0 0 0]',fin)+measure2([0 1 0 0]',fin)+measure2([0 0 1 0]',fin) ...
        )+(a22+a23+incentive)*measure2([0 0 0 1]',fin) ;
end

function m = measure2(b,fin)
m=abs( sum(b.*conj(fin)))^2;
end
```

D.2 3 Player Game

D.2.1 Simulation

```


function out = simulationQuantumPirateGame3Players(a_11,a_12,a_13,a_22,a_23, gamma, ...
    U1,U2,U3,U4,U5)
%
%         IN:
%−− alocation proposal a_ij −i number of the player that proposes a_ij coins
%   to player j
%−− gamma entanglemen coefitient
%−− U1, U2, U3, U4, U5 player's strategies; default CDCCD
%
%        OUT:
%−− expected utility for players 1 ,2 ,3:  [u1 u2 u3]

%−− Check variables and set to defaults
if exist('a_11','var')≠1, a_11=99; end
if exist('a_12','var')≠1, a_12=0; end
if exist('a_13','var')≠1, a_13=1; end
if exist('a_22','var')≠1, a_22=100; end
if exist('a_23','var')≠1, a_23=0; end
if exist('gamma','var')≠1, gamma=0; end

%−− Classic Pure Strategy Operators
%   cooperate= [1 0;0 1]
C= eye(2);
%   defect= [0 1;1 0]
D= ones(2)−eye(2);
D(2,1)=1;
```

```

if exist('U1','var')≠1, U1=C; end
if exist('U2','var')≠1, U2=D; end
if exist('U3','var')≠1, U3=C; end
if exist('U4','var')≠1, U4=C; end
if exist('U5','var')≠1, U5=D; end

%-- Matrix containing all pure-states for hilbert space 32
H32B=eye(32);

step=0.1;
t=0:step:pi;

i=gamma;

%-- varying the entanglement parameter

%-- Building the initial state, for the entanglement parameter i
ini= cos(i/2)*kron([1 0]', kron([1 0]',[1 0]'))+1i*sin(i/2)*kron([0 1]', kron([0 1]',[0 ...
    1]'));
%-- Entanglement Gate J
J= expm(1i*(i/2)*kron(D,kron(D,kron(D,kron(D,D)))));
%-- Alternative way to build the initial state
ini= J*H32B(:,1);
%-- Deentangles to produce a final state
Jd = ctranspose( J);

H= fft(eye(2))/sqrt(2);
fin = kron(U1,kron(U2,kron(U3,kron(U4,U5))))*ini;
fin= Jd*fin;
out = expectedUtility(fin, a_11,a_12,a_13,a_22,a_23);

end


function out = expectedUtility(fin, a_11,a_12,a_13,a_22,a_23)
%payofffunc_player1(fin)
%
%                 Calculates the payoff for player 1
%               IN
%                    fin: final state
%

%-- Check variables and set to defaults
if exist('a_11','var')≠1, a_11=99; end
if exist('a_12','var')≠1, a_12=0; end
if exist('a_13','var')≠1, a_13=1; end
if exist('a_22','var')≠1, a_22=100; end
if exist('a_23','var')≠1, a_23=0; end

%-- Matrix containing all pure-states for hilbert space 32
H32B=eye(32);

prob_proposal_1_accepted=0;
for accepted=[1 2 3 4 5 6 7 8 9 10 11 12 17 18 19 20]
    prob_proposal_1_accepted= prob_proposal_1_accepted + measure(H32B(:,accepted),fin);
end
prob_proposal_1_accepted;

prob_proposal_1_rejected=0;
for rejected=[13 14 15 16 21 22 23 24 25 26 27 28 29 30 31 32]
    prob_proposal_1_rejected= prob_proposal_1_rejected + measure(H32B(:,rejected),fin);
end
prob_proposal_1_rejected;

prob_proposal_2_accepted=0;
for accepted=[13 14 15 21 22 23 25 26 27 29 30 31]
    prob_proposal_1_accepted= prob_proposal_1_accepted + measure(H32B(:,accepted),fin);
end
prob_proposal_2_accepted;

prob_proposal_2_rejected=0;
for rejected=[16 24 28 32]
```

```
    prob_proposal_1_rejected= prob_proposal_1_rejected + measure(H32B(:,rejected),fin);
end
prob_proposal_2_rejected;

u1 = a_11*prob_proposal_1_accepted -200* prob_proposal_1_rejected;

u2 = a_12*prob_proposal_1_accepted +(0.5 + a_22)*prob_proposal_2_accepted ...
    -199.5*prob_proposal_2_rejected;

u3 = a_13*prob_proposal_1_accepted +(0.5 + a_23)*prob_proposal_2_accepted + ...
    100.5*prob_proposal_2_rejected;
out =[ u1 u2 u3];
end

function m = measure(b,fin)
m= (norm(conj(b).*fin))^2;
end


function u= U_thetha_phi(theta,phi)
u= [exp(1i*phi)cos(theta/2) *sin(theta/2);sin(theta/2) -exp(-i*phi)*cos(theta/2)];
end
```