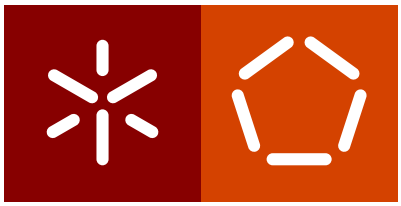


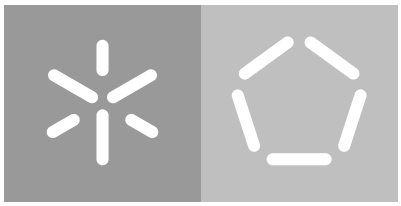
Universidade do Minho
Escola de Engenharia??
Departamento de Informática??

Jaime Santos

Staggered Quantum Walks in Qiskit?? Second Part of Title

First Part of Subtitle
Second part of Subtitle

July 2020



Universidade do Minho
Escola de Engenharia??
Departamento de Informática??

Jaime Santos

Staggered Quantum Walks in Qiskit?? Second Part of Title

First Part of Subtitle Second part of Subtitle

Master dissertation
Master Degree in Física da Informaçao

Dissertation supervised by
Luís Barbosa
Bruno Chagas

July 2020

ACKNOWLEDGEMENTS

Write acknowledgements here

ABSTRACT

Implementation of staggered quantum walks in qiskit.

RESUMO

Pensar no que escrever aqui.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1

INTRODUCTION

1.1 BRIEF HISTORY OF QUANTUM COMPUTING

1.1.1 *[TEMP ARTIGOS RELEVANTES]*

kendon2006 - Introducao historia CQ - Pasta introqw/Coin toffoli1980- Reversible Computing - Pasta intromain
gottesman1998- The Heisenberg Representation of Quantum Computers - Pasta intromain
rebuscado turing1936 - ON COMPUTABLE NUMBERS, WITH AN APPLICATION TO THE ENTSCHEIDUNGSPROBLEM - Pasta intromain
Random walks classicas -

1.2 CLASSICAL AND QUANTUM RANDOM WALKS

From classical random walks to quantum random walks.
Balu2017 introduçao
Quantum walks are typically classified into discrete- and continuous-time quantum walks with the main difference being whether free evolution is interrupted by quantum coin 'flips' followed by coin-dependent translations or whether evolution is continuous involving entangling between the walker and internal, or coin, degrees of freedom [42] **topological-walks.pdf**

1.3 STATE OF THE ART QUANTUM WALKS IMPLEMENTATIONS

Mencionar que o proximo capitulo tem implementacoes de outros algoritmos.
timeline da wikipedia
ler artigos das implementaçoes circuitos locke a e b; douglas wang
Quantum walks have become germane to quantum computation [20–25] and quantum simulation [26–29] and single-walker versions are amenable to experimental implementations including ion traps [30, 31], superconducting systems [32, 33], nuclear magnetic resonance

[34, 35], optical lattice [36, 37], and both freespace linear optics [38, 39] and on photonic chips [40, 41] **topologicalwalks.pdf**

1.4 TEXT OVERVIEW AND CONTRIBUTIONS

2

QUANTUM COMPUTING

qubits
esfera de bloch
rotacoes – implementacoes simples nos pcs da ibm.
grover exemplo. Artigo coles 2018 , livro do nielsen tambem.
isto nao seria state of the art ?
Fazer ao estilo do Nielsen pag. 174, meter exemplos do qiskit

2.0.1 *[TEMP ARTIGOS RELEVANTES]*

bennet1996- Strengths and Weaknesses of Quantum Computing - Pasta intromain
deutsch1992- Rapid solution of problems by quantum computation - Pasta intromain
gottesman1998- The Heisenberg Representation of Quantum Computers - Pasta intromain
shor1994- Algorithms for quantum computation: discrete logarithms and factoring - Pasta intromain
toffoli1980- REVERSIBLE COMPUTING - Pasta intromain
wootters1982- A single quantum cannot be cloned - Pasta intromain
zalka1999- Grover's quantum searching algorithm is optima - Pasta intromain
aharonov2001.pdf - Modelo computaçao quantica.

2.0.2 *Grover's Algorithm*

Searching through an unstructured database is a task classically achieved by exhaustively evaluating every element in the database. Assume there exists a black box (oracle) that can be asked to find out if two elements are equal. Since we're looking for a specific element in a database of size N, we'd have to query the oracle on average $\frac{N}{2}$ times, or in the worst case N times.

Grover's algorithm, presented in Grover (1996), comes as a quantum alternative to this type of problems, taking advantage of superposition by increasing desirable states' amplitudes through a process called *amplitude amplification*. This method has a quadratic gain over the

classical counterpart Boyer et al. (1998), being able to find a target element with $\mathcal{O}(\sqrt{N})$ oracle complexity.

Let us now expand on the inner workings of the black box. We start by focusing on searching indexes instead of directly evaluating the element and we assume $N = 2^n$, n being a positive integer. We can now define a function $f : \{0, 1, ..., N-1\}$ that returns 1 when evaluating the desired (marked) element and 0 otherwise. Since this function is to be applied to a quantum system, we must build a unitary operator $\mathcal{F}$

$$\mathcal{F} |x\rangle |i\rangle = |x\rangle |i \oplus f(x)\rangle . \tag{1}$$

where $|x\rangle$ is the index register, $\oplus$ is the binary sum operation and $|i\rangle$ is a qubit that is flipped if $f(x) = 1$.

The action of the oracle on state $|0\rangle$ will be

$$\mathcal{F} |x\rangle |0\rangle = \begin{cases} |x_0\rangle |1\rangle , & \text{if } x = x_0 \\ |x\rangle |0\rangle , & \text{otherwise.} \end{cases} \tag{2}$$

where x_0 is the marked element. More generically, $\mathcal{F}$ can be written as

$$\mathcal{F} |x\rangle = (-1)^{f(x)} |x\rangle . \tag{3}$$

This offers a bit of insight into the oracle, it *marks* the solutions to the search problem by applying a phase shift to the solutions. As shown in **?**, for an N item search with K solutions, we must only apply the oracle $\mathcal{O}(\sqrt{N/K})$ times, which will be useful later when we describe the *single shot Grover*.

The question now is, what is the procedure that determines a solution x_0 using $\mathcal{F}$ the minimum number of times? The answer lies in the amplitude amplification section of Grover's search, starting with the creation of a uniform superposition

$$|\psi\rangle = H^{\otimes n} |0\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle . \tag{4}$$

where $H^{\otimes n}$ is the *Hadamard* operator, whose action on states $|0\rangle$ and $|1\rangle$ is

$$H |0\rangle |x\rangle = \frac{|0\rangle |x\rangle + |1\rangle |x\rangle}{\sqrt{2}} \tag{5}$$

$$H |1\rangle |x\rangle = \frac{|0\rangle |x\rangle - |1\rangle |x\rangle}{\sqrt{2}}. \tag{6}$$

If we were to measure $|x\rangle$ at this point, the superposition would collapse to any of the base states with the same probability $\frac{1}{N} = \frac{1}{2^n}$, which means that on average, we'd need to

try $N = 2^n$ times to guess the correct item. This is where amplitude amplification comes into effect, by means of a second unitary operator

$$\mathcal{D} = (2\left|\psi\right\rangle\left\langle\psi\right| - I) = H^{\otimes n}(2\left|0\right\rangle\left\langle 0\right| - I)H^{\otimes n} \tag{7}$$

This operator applies a conditional phase shift, with every computational basis state except $|0\rangle$ receiving a phase shift. This can also be described as the *inversion about the mean*, for a state of arbitrary amplitudes

$$\left|\phi\right\rangle = \sum_{k=0}^{N-1} \alpha_k \left|k\right\rangle \tag{8}$$

the action of D on state ϕ will be

$$\mathcal{D}\left|\phi\right\rangle = \sum_{k=0}^{N-1} (-\alpha_k + 2\langle\alpha\rangle)\left|k\right\rangle \tag{9}$$

where $\langle\alpha\rangle$ is the average of α_k

$$\langle\alpha\rangle = \frac{1}{N}\sum_{k=0}^{N-1} \alpha_k \left|k\right\rangle \tag{10}$$

The evolution operator that performs one step of the algorithm is then

$$\mathcal{U} = \mathcal{D}\mathcal{F} \tag{11}$$

and after t steps the state of the system is

$$\left|\psi(t)\right\rangle = \mathcal{U}^t\left|\psi\right\rangle. \tag{12}$$

The optimal number of steps is, as aforementioned, $\sqrt{\frac{N}{K}}$ where K is the number of solutions to the problem.

3

QUANTUM WALKS

Falar dos varios tipos de caminhada quantica sem circuito
talvez simulaçoes em python??
ambainis2004 - Pasta introqw/Coin
introducao com random walks classicas
Meter resultados simulaçoes de python.

3.1 COINED QUANTUM WALK

3.1.1 *[TEMP Artigos relevantes]*

MAIS MOEDAS. aharonov1993 - Pasta introqw/Coin
aharonov2001 - Pasta introqw/Coin - definiçao da QW ;shift operator mod n
ambainis2001 - Pasta introqw/Coin
carteret2005- Implementacao - Pasta introqw/Coin
inui2003 - - Pasta introqw/Coin
kendon2006 - Introducao historia CQ - Pasta introqw/Coin
shenvi2002 - Relacao com grover pg. 10
Resto da pasta.

3.1.2 *Texto*

Consider a one dimensional walker on top of a line, whose decision to go left or right depends on a coin toss. As shown in Portugal (2018), the outcome after many iterations of the coin toss is a binomial distribution centered in the starting position with a standard deviation of $\sqrt{t}$, where t is the number of time steps. The one dimensional walk can be abstracted to a graph of any dimension, that if allowed to be weighted and directed could also be called a Markov chain.

In the quantum case, the walker is a quantum system whose position x on the line is described by a vector $|x\rangle$ in Hilbert Space. The next position of the system is determined by a unitary operator, which can be viewed as a quantum coin. The analogy is, if the coin is tossed and rolls "heads", for example, the system transitions to position $|x+1\rangle$, otherwise it advances to $|x-1\rangle$. The Hilbert space of the system is $\mathscr{H} = \mathscr{H}_C \otimes \mathscr{H}_P$, where $\mathscr{H}_C$ is the two-dimensional Hilbert space associated with the "coin" and $\mathscr{H}_P$ is the one-dimensional lattice's Hilbert space. One of the most commonly used coins is the Hadamard operator, which is the one chosen for this example.

The transition from $|x\rangle$ to either $|x+1\rangle$ or $|x-1\rangle$ must be described by a unitary operator, the *shift operator*

$$\mathcal{S}\,|0\rangle\,|x\rangle = |0\rangle\,|x-1\rangle \tag{13}$$

$$\mathcal{S}\,|1\rangle\,|x\rangle = |1\rangle\,|x+1\rangle \tag{14}$$

that can also be described by

$$S = |0\rangle\langle 0| \otimes \sum_{x=-\infty}^{x=\infty} |x+1\rangle\langle x| + |1\rangle\langle 1| \otimes \sum_{x=-\infty}^{x=\infty} |x-1\rangle\langle x|\,. \tag{15}$$

It follows that the operator that describes the dynamics of the quantum walk will be given by

$$U = S(H \otimes I). \tag{16}$$

Consider a quantum system located at $|x=0\rangle$ with coin state $|0\rangle$, for $t=0$. It's state will be described by

$$|\psi(0)\rangle = |0\rangle\,|x=0\rangle\,, \tag{17}$$

and after t steps the walker's state will be

$$|\psi(t)\rangle = U^t\,|\psi(0)\rangle\,, \tag{18}$$

more explicitly

$$|\psi(0)\rangle \xrightarrow{U} |\psi(1)\rangle \xrightarrow{U} |\psi(2)\rangle \xrightarrow{U} (...) \xrightarrow{U} |\psi(t)\rangle\,. \tag{19}$$

After applying the Hadamard operator followed by the shift operator twice the respective system states are

$$|\psi(1)\rangle = \frac{|0\rangle\,|x=-1\rangle + |1\rangle\,|x=1\rangle}{\sqrt{2}} \tag{20}$$

$$|\psi(2)\rangle = \frac{|0\rangle\,|x=-2\rangle + |1\rangle\,|x=0\rangle + |0\rangle\,|x=0\rangle - |1\rangle\,|x=2\rangle}{2} \tag{21}$$

$$\tag{22}$$

If one where to measure the system after each application of $\mathcal{U}$, it would be expected to see the walker at $x = 1$ 50% of the time and at $x = -1$ for the remainder, after the first iteration (20). Measure the system t times and the result is a binomial probability distribution. The conclusion is that repetitive measurement of a coined quantum walk system reduces to the classical case, which means that any desired quantum behaviour is lost.

It is possible, however, to make use of the quantum correlations between different positions to generate constructive or destructive interference, by applying the Hadamard and shift operators successively without intermediary measurements. The consequences of interference between states are very apparent in

$$|\psi(3)\rangle = \frac{|1\rangle\,|x=-3\rangle - |0\rangle\,|x=-1\rangle + 2(|0\rangle + |1\rangle)\,|x=1\rangle + |0\rangle\,|x=3\rangle}{2\sqrt{2}}. \tag{23}$$

Even though an unbiased coin was used, this state is not symmetric to the origin and the probability distributions will not be centered in the origin. Moreover, the standard deviation will not be $\sqrt{t}$.

The quantum walk is said to be *ballistic* since its standard deviation is proportional to t Venegas-Andraca (2012) meaning exponentially faster hitting times in certain graphs Childs et al. (2002); Farhi and Gutmann (1998) which can be advantageous in problems that involve visiting certain vertices in a graph. There are also studies that show that a quantum walk may have advantages in element distinctness Ambainis (2007) and spatial search Childs and Goldstone (2004) problems.

In order to study this distribution, a simulation of the coined quantum walk was coded in *Python*, **[PRESENTE ONDE?]**. Using the Hadamard coin and the aforementioned initial condition, the resulting probability distribution is:

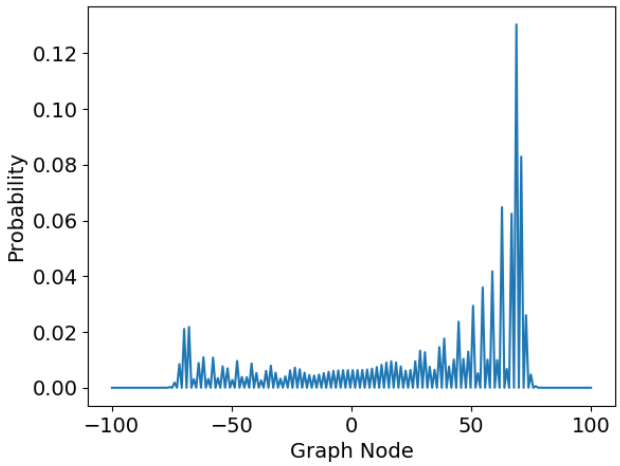


Figure 1: Probability distribution for the coined quantum walk on a line, after 100 steps, with initial condition $|\psi(0)\rangle = |0, x = 0\rangle$ and the Hadamard coin.

Analyzing the plot, it is noticeable that the distribution is asymmetric, the probability amplitude of finding the walker at the right-hand side is much larger than at the left, with a peak around $x = \frac{100}{\sqrt{2}}$ **[COMO POSSO PROVAR ISTO? Confirmar atraves de graficos]**. Regardless of number of steps, this peak is always present (albeit in varying positions), which is to say that the walker can always be found moving in a uniform fashion away from the origin, implying that the quantum walk has *ballistic* behaviour.

Another interesting case study is to find if this behaviour is preserved for a symmetric distribution around the origin. For this purpose, one must first understand where the asymmetry comes from. The Hadamard operator induces a sign flip **[in OU on ?]** state $|1\rangle$, hence more terms are cancelled when the coin state is $|1\rangle$ **[FAZER DEMONSTRAÇAO?]**. Since $|0\rangle$ was defined to induce movement to the right, the result is as shown in 1. It then follows that the initial condition

$$|\psi(0)\rangle = -|1, x = 0\rangle \tag{24}$$

Will return a mirror image of 1:

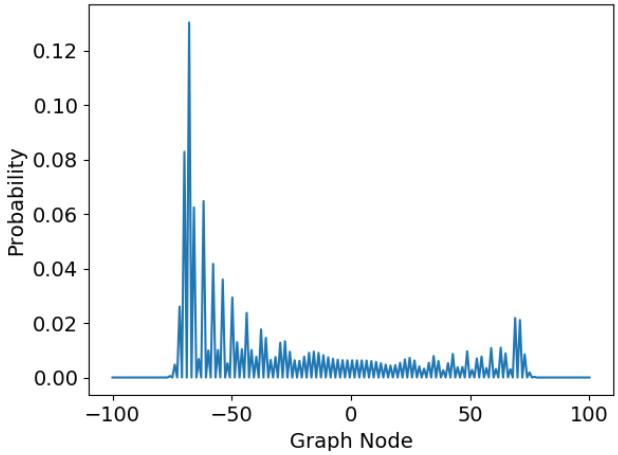


Figure 2: Probability distribution for the coined quantum walk on a line, after 100 steps, with initial condition $|\psi(0)\rangle = -|1, x = 0\rangle$ and the Hadamard coin.

In order to obtain a symmetrical distribution, one must superpose 17 and 43 carefully, as to not cancel terms before the calculation of the probability distribution, by multiplying 43 by the imaginary unit. This works because the entries of the Hadamard operator are real numbers; terms with the imaginary unit will not cancel out with terms without it**[FAZER DEMONSTRAÇAO?]**, thus the walk can proceed to both left and right. The initial condition is then:

$$|\psi(0)\rangle = \frac{|0\rangle - i\,|1\rangle}{\sqrt{2}}\,|x = 0\rangle \tag{25}$$

With a correspoding plot:

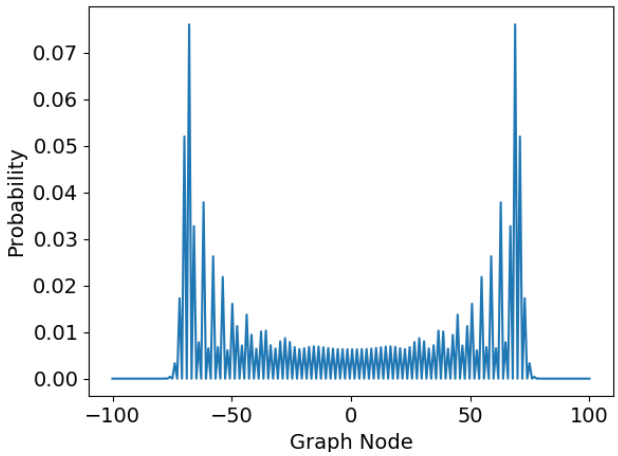


Figure 3: Probability distribution for the coined quantum walk on a line, after 100 steps, with initial condition $|\psi(0)\rangle = \frac{|0\rangle - i|1\rangle}{\sqrt{2}} |x = 0\rangle$ and the Hadamard coin.

3.2 CONTINUOUS-TIME QUANTUM WALK

3.2.1 *[TEMP Artigos relevantes]*

Loke2016b- Pasta introqw childs2001 childs2004 dam2002 pg 204 nielsen fazer mais conds inciais

3.2.2 *[Texto]*

The transition from the classical random walk model (A.2) to the coined quantum model was done by replacing the probability vector by a state vector of probability amplitudes and the transition matrix by a unitary matrix. The position Hilbert space was also extended with the coin Hilbert space, through the tensor product. The same will be done for A.3 resulting in the *continuous-time Quantum Walk*.

In this case, the nodes can be viewed as quantum states that form the basis for the Hilbert space, and the first step is to create an Hamiltonian with non-zero matrix elements only between states that are adjacent (connected) in the graph. Matrix $\hat{H}$ is Hermitan and matrix M is not necessarily unitary, which can be fixed by replacing the generating matrix $\hat{H}$ by $i\hat{H}$.

The quantum system is then started in the state corresponding to the topmost node and has a time evolution determined by $\hat{H}$ so that the unitary time evolution operator is

$$U(t) = e^{-iHt} \tag{26}$$

The quantum state at time t, for the corresponding state $|\psi(0)\rangle$, will be

$$|\psi(t)\rangle = U(t)\,|\psi(0)\rangle \tag{27}$$

With a probability distribution

$$p_k = |\,\langle k|\psi(t)\rangle\,|^2 \tag{28}$$

where k is a vertex and $|k\rangle$ is the state of the computational basis relative to k.

Considering the line example, matrix H will reduce to

$$H_{ij} = \begin{cases} 2\gamma, & \text{if } i = j; \\ -\gamma, & \text{if } i \neq j \text{ and adjacent;} \\ 0, & \text{if } i \neq j \text{ and not adjacent.} \end{cases} \tag{29}$$

And, when applied to a position:

$$H\,|x\rangle = -\gamma\,|x-1\rangle + 2\gamma\,|x\rangle - \gamma\,|x+1\rangle \tag{30}$$

Coding this into Python **[METER AQUI A REFERENCIA]**, the following plot was obtained setting the transition rate to $\gamma = \frac{1}{2\sqrt{2}}$ the initial condition to $|\psi(0)\rangle = |0\rangle$:

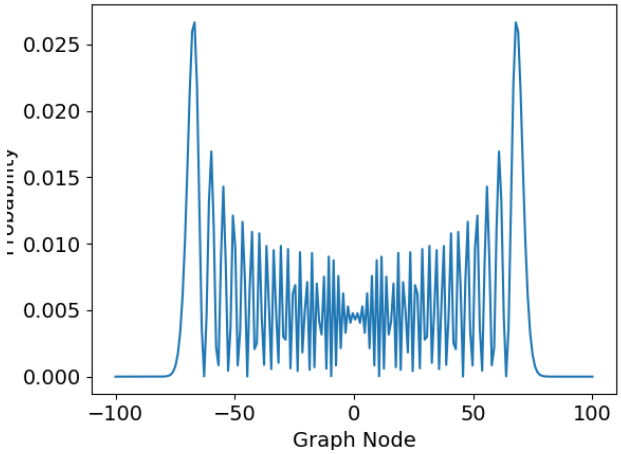


Figure 4: Probability distribution for the continuous-time quantum walk on a line, after 100 steps, with initial condition $|\psi(0)\rangle = |0\rangle$ and the Hadamard coin.

3.3 STAGGERED QUANTUM WALK

Similarly to the continuous-time quantum walk, the staggered case aims to spread a transition probability to neighboring vertices. The notion of adjacency now comes from cliques[1], and the initial stage of this walk consists in partitioning the graph in several different cliques. This is called a tesselation, and it is defined as the division of the set of vertices into disjoint cliques. An element of tesselation $\mathscr{T}$ is called a polygon, and it's only valid if if all of its vertices belong to the clique in $\mathscr{T}$. The set of polygons of each tesselation must cover all vertices of the graph, and the set of tesselations ($\mathscr{T}_1$,$\mathscr{T}_2$,...,$\mathscr{T}_k$) must cover all edges.

These definitions allow the construction of operators H_1,H_2,...,H_k that will be used to propagate the probability amplitude locally, in each polygon. The state associated to each polygon is

$$|u_j^k\rangle = \frac{1}{\sqrt{|\alpha_j^k|}} \sum_{l \in \alpha_j^k} |l\rangle \tag{31}$$

where α_j^k is the $j-th$ polygon in the $k-th$ tesselation.

The unitary and Hermitian operator H_k, associated to each tesselation is defined in Portugal et al. (2017) as:

$$H_k = 2 \sum_{j=1}^{p} |u_j^k\rangle \langle u_j^k| - I \tag{32}$$

1 A clique is defined as the subset of vertices of an undirected graph such that every two distinct vertices in each clique are adjacent.

The authors also show that the evolution operator can be generalized with Hamiltonians

$$U = e^{i\theta_k H_k} ... e^{i\theta_2 H_2} e^{i\theta_1 H_1} \tag{33}$$

where

$$e^{i\theta_k H_k} = \cos(\theta_k) I + i \sin(\theta_k) H_k \tag{34}$$

and $H_k^2 = I$.

The simplest use case of this quantum walk model is the one-dimensional lattice, where the minimum tessalations are:

$$\mathscr{T}_\alpha = \{\{2x, 2x+1\} : x \in Z\} \tag{35}$$

$$\mathscr{T}_\beta = \{\{2x+1, 2x+2\} : x \in Z\} \tag{36}$$

Each element of the tesselation has a corresponding state, and the uniform superposition of these states is:

$$|\alpha_x\rangle = \frac{|2x\rangle + |2x+1\rangle}{\sqrt{2}} \tag{37}$$

$$|\beta_x\rangle = \frac{|2x+1\rangle + |2x+2\rangle}{\sqrt{2}} \tag{38}$$

One can now define operators H_α and H_β as:

$$H_\alpha = 2 \sum_{x=-\infty}^{+\infty} |\alpha_x\rangle \langle\alpha_x| - I \tag{39}$$

$$H_\beta = 2 \sum_{x=-\infty}^{+\infty} |\beta_x\rangle \langle\beta_x| - I \tag{40}$$

The next step is to define the Hamiltonian evolution operator as

$$U = e^{i\theta H_\beta} e^{i\theta H_\alpha} \tag{41}$$

and apply it to an initial condition $|\psi(0)\rangle$, resulting in the time evolution operator

$$U |\psi(t)\rangle = U^t |\psi(0)\rangle \tag{42}$$

The following graphs, using a time evolution operator with initial condition $|\psi(0)\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}}$, will shed further light into the behaviour of this quantum walk:

3.4 DESVIO PADRAO

-grafico com a media dependendo da cond inicial - grafico do quadrado da media

4

IMPLEMENTATIONS AND APPLICATIONS

meter aqui os circuitos.
Pagerank? paparo2012 tem coisas
Simulaçoes em python?
mencionar PC IBMQ

5

DISCUSSIONS AND CONCLUSION

discutir os resultados encontrados, o future work, o que falta fazer etc.

5.1 CONCLUSIONS

5.2 PROSPECT FOR FUTURE WORK

BIBLIOGRAPHY

Andris Ambainis. Quantum walk algorithm for element distinctness. *SIAM Journal on Computing*, 37(1):210–239, 2007.

Michael Boyer, Gilles Brassard, Peter Høyer, and Alain Tapp. Tight bounds on quantum searching. *Fortschritte der Physik*, 46(4-5):493–505, 1998.

Andrew M. Childs and Jeffrey Goldstone. Spatial search by quantum walk. *Physical Review A*, 70(2):022314, 2004.

Andrew M. Childs, Richard Cleve, Enrico Deott, Edward Farhi, Sam Gutmann, and Daniel A. Spielman. Exponential algorithmic speedup by quantum walk. *Proc. 35th ACM Symposium on Theory of Computing (STOC 2003), pp. 59-68*, 2002. doi: 10.1145/780542.780552.

Edward Farhi and Sam Gutmann. Quantum computation and decision trees. *Physical Review A, 58:915-928*, 1998. doi: 10.1103/PhysRevA.58.915.

Lov K. Grover. A fast quantum mechanical algorithm for database search. *STOC '96 Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 212–219, 1996.

Renato Portugal. *Quantum Walks and Search Algorithms*. Springer, 2018.

Renato Portugal, M. C. de Oliveira, and J. K. Moqadam. Staggered quantum walks with hamiltonians. *Physical Review A*, 95(1):012328, 2017.

Salvador Elías Venegas-Andraca. Quantum walks: a comprehensive review. *Quantum Information Processing*, 11(5):1015–1106, 2012.

A

SUPPORT MATERIAL

A.1 GRAPHS

A.2 CLASSICAL DISCRETE-TIME MARKOV CHAINS

A *classical Markov Chain* is a stochastic process that assumes discrete values and whose next state depends only on the current state, which is ruled by a deterministic or random rule based only on the current state. This can be viewed as a directed graph where the states are represented by *nodes* and the transitions between states by *edges*. In the case of the classical discrete-time Markov chain, each transition has an associated probability distribution and, after choosing an order for the states, this distribution is described by a vector:

$$\vec{p}(t) = \begin{bmatrix} p_1(t) \\ p_2(t) \\ \vdots \\ p_n(t) \end{bmatrix}$$

where $p_i(t)$ is the probability of the walker being on vertex x_i at time t. Due to the probabilistic nature of the transitions, one cannot tell the exact node the walker will be at a future time. One can instead determine the probability distribution, which will require the *transition matrix* M. **[POR ACABAR]**

A.3 CLASSICAL CONTINUOUS-TIME MARKOV CHAINS

Previously in A.2, the *classical discrete-times Markov chains* were defined in order to make the transition from the *classical random walk* to the *coined quantum walk* (3.1). This appendix comes as an additional tool to better understand 3.2.

Continuous time can be considered a stochastic process where time can be any non-negative real number, and the walker can now transition from vertex x_i to x_j at any time.

Eventually, the probability of the walker being found in adjacent vertices will increase and the probability of staying on x_i decrease. This is called the *transition rate*, γ, and it is homogeneous and uniform[1]. The probability of this transition is γ over time.

In order to deal with continuous variables, one must consider an *infinitesimal* time interval, τ, and solve the problem's differential equation. The probability of a transition during this time interval is thus γ_τ. The degree of vertex x_j is defined as d_j, will also be a component of the probability since it relates to how many other vertices it is connected to. After τ time, it is expected to see a transition with probability $d_j\gamma_\tau$ and a standstill with $1 - d_j\gamma_\tau$ probability. It is now possible to define a transition matrix as

$$M_{ij} = \begin{cases} 1 - d_j\gamma_\tau + O(\tau^2), & \text{if } i = j; \\ \gamma_+ O(\tau^2), & \text{if } i \neq j; \end{cases} \tag{43}$$

[Perceber o O] And an auxiliary generating matrix as

$$H_{ij} = \begin{cases} d_j\gamma, & \text{if } i = j; \\ -\gamma, & \text{if } i \neq j \text{ and adjacent;} \\ 0, & \text{if } i \neq j \text{ and not adjacent.} \end{cases} \tag{44}$$

Since the next state of a Markov state only depends on the current state, the probability of two independent events is given by their multiplication. The transition matrix at different times will then be

$$M_{ij}(t + \tau) = \sum_k M_{ik}(t) M_{kj}(\tau) \tag{45}$$

By noticing that one needs only to calculate the transitions for vertices adjacent to x_j and after some manipulation of 46, as was done in Portugal (2018), one arrives at the following differential equation

$$\frac{dM_{ij}(t)}{dt} = -\sum_k H_{kj} M_{ik}(t) \tag{46}$$

Setting the initial condition $M_{ij}(0) = \delta_{ij}$**[Perceber a cond inicial - e simplesmente a identidade]**, the solution for the differential equation is

$$M(t) = e^{-Ht} \tag{47}$$

With the transition matrix defined, the corresponding probability distribution will be

$$\vec{p}(t) = M(t)\vec{p}(0) \tag{48}$$

[Tentar completar mais isto e alterar um bocado o texto.]

1 It is the same for all vertices and times, respectively.